

Online Programmer Manual



**TDS7000B Series
Digital Phosphor Oscilloscopes**

**CSA7000B Series
Communication Signal Analyzers**

PHP024310

Adapted from the TDS7000B Series and CSA7000B Series Online
Programmer Guide Version 1.0, October 20, 2003.

Table of Contents

Copyright and Version Information	1
---	---

Getting Started

Introduction	2
Setting Up Remote Communications	4
Documentation	8

Command Syntax

Syntax Overview	9
Command and Query Structure	9
Clearing the Instrument	11
Command Entry	12
Constructed Mnemonics	14
Argument Types	15

Command Groups

Acquisition Command Group	18
Alias Command Group	19
Calibration Command Group	20
Cursor Command Group	20
Diagnostics Command Group	22
Display Control Command Group	24
File System Command Group	26
Hard Copy and Export Command Group	26
Histogram Command Group	27
Horizontal Command Group	28
Mask Command Group	31
Math Command Group	36
Measurement Command Group	38
Miscellaneous Command Group	42
Save and Recall Command Group	43
Status and Error Command Group	44
Trigger Command Group	45
Vertical Command Group	51
Waveform Transfer Command Group	53
Zoom Command Group	58

Commands Listed in Alphabetical Order

*CAL?	59
*CLS	60
*DDT	61
*ESE	62
*ESR?	64
*IDN?	65
*LRN?	66
*OPC	73
*OPT?	75
*PSC	76
*PUD	77
*RCL	78
*RST	79
*SAV	80
*SDS	81
*SRE	82
*STB?	83
*TRG	84
*TST?	85
*WAI	86
ACQuire?	87
ACQuire:MODe	88
ACQuire:NUMACq?	91
ACQuire:NUMAVg	92
ACQuire:NUMEnv	93
ACQuire:NUMSAMPles	94
ACQuire:SAMPlingmode	95
ACQuire:STATE	96
ACQuire:STOPAfter	98
ALlas	99
ALlas:CATalog?	100
ALlas:DEFine	101
ALlas:DELEte	103
ALlas:DELEte:ALL	104
ALlas:DELEte:NAME	105
ALlas:STATE	106
ALLEV?	107
AUTOSet	108
AUXOut?	110
AUXOut:EDGE	111
AUXOut:SOUrce	112
BELI	113
BUSY?	114
CALibrate?	115

CALibrate:CALProbe:CH<x>?	116
CALibrate:PROBEstate:CH<x>?	117
CALibrate:RESults?	118
CALibrate:RESults:SPC?	119
CH<x>?	120
CH<x>:BANdwidth	121
CH<x>:COUPling	123
CH<x>:DESKew	124
CH<x>:LABEL:NAME	125
CH<x>:LABEL:XPOS	126
CH<x>:LABEL:YPOS	127
CH<x>:OFFSet	128
CH<x>:POSition	130
CH<x>:PROBE?	132
CH<x>:PROBECal?	133
CH<x>:PRObe:GAIN?	134
CH<x>:PRObe:ID?	135
CH<x>:PRObe:ID:SERnumber?	136
CH<x>:PRObe:ID:TYPE?	137
CH<x>:PRObe:RESistance?	138
CH<x>:PRObe:SET	139
CH<x>:PRObe:UNIts?	141
CH<x>:PROBEFunc:EXTatten	142
CH<x>:PROBEFunc:EXTDBatten	144
CH<x>:PROBEFunc:EXTUnits	146
CH<x>:SCALE	147
CH<x>:TERmination	149
CLEARMenu	150
CMDBatch	151
CURSor?	152
CURSor:FUNcTION	153
CURSor:HBArs?	155
CURSor:HBArs:DELTA?	156
CURSor:HBArs:POSITION<x>	157
CURSor:HBArs:UNIts?	158
CURSor:MODE	159
CURSor:SCREEN:STYle	160
CURSor:SOUrce	161
CURSor:STATE	162
CURSor:VBArS	163
CURSor:VBArS:DELTA?	164
CURSor:VBArS:POSITION<x>	165
CURSor:VBArS:UNIts	166
CURSor:WAVEform:STYle	167
CURVe	168
CURVEStream	171

DATA	173
DATA:DESTINATION	174
DATA:ENCdG	175
DATA:FRAMESTART	178
DATA:FRAMESTOP	180
DATA:SOURce	182
DATA:START	184
DATA:STOP	185
DATE	187
DELEte:SETUp	188
DELEte:WAVEform	189
DESE	190
DIAg:CONTROL:HALT	192
DIAg:CONTROL:LOOP	193
DIAg:EXECUTE	194
DIAg:ITEM?	195
DIAg:ITEM:FAILURES?	196
DIAg:ITEM:NAME?	197
DIAg:ITEM:RESULT?	198
DIAg:ITEM:SUBITEMS?	199
DIAg:LEVEL	200
DIAg:LOOPS?	201
DIAg:NAME?	202
DIAg:NAME:AREA?	203
DIAg:NAME:SUBSYS?	204
DIAg:NAME:TEST?	205
DIAg:NUMITEMS?	206
DIAg:RESULts?	207
DIAg:RESULts:VERBose?	208
DIAg:SELEct:ALL	209
DIAg:SELEct:AREA	210
DIAg:SELEct:LAST	211
DIAg:SELEct:SUBSYS	212
DIAg:SELEct:TEST	213
DIAg:STATE	214
DIAg:STOP	215
DISplay?	216
DISplay:CLOCK	217
DISplay:COLOR?	218
DISplay:COLOR:MATHCOLOR	219
DISplay:COLOR:PALEtte	220
DISplay:COLOR:PALEtte:IMAGEView	222
DISplay:COLOR:PALEtte:RECORDView	224
DISplay:COLOR:PALEtte:USER?	226
DISplay:COLOR:PALEtte:USER:CARet	227
DISplay:COLOR:PALEtte:USER:CH<x>	228

DISplay:COLOr:PALEtte:USER:GRATicule	229
DISplay:COLOr:PALEtte:USER:HIStogram	230
DISplay:COLOr:PALEtte:USER:MASK	231
DISplay:COLOr:PALEtte:USER:MASKHighlight	232
DISplay:COLOr:PALEtte:USER:MATH<x>	233
DISplay:COLOr:PALEtte:USER:REF<x>	234
DISplay:COLOr:REFCOLOr	235
DISplay:DESKew	236
DISplay:FILTer	237
DISplay:FORMat	238
DISplay:GRATicule	240
DISplay:INTENSITy?	241
DISplay:INTENSITy:AUTOBright	242
DISplay:INTENSITy:SCREENSAVER	244
DISplay:INTENSITy:SCREENSAVERDELAY	246
DISplay:INTENSITy:WAVEform	247
DISplay:INTENSITy:WAVEform:IMAGEView	248
DISplay:INTENSITy:WAVEform:RECORDView	249
DISplay:PERSiStence	250
DISplay:PERSiStence:RESEt	252
DISplay:SCREENTExt?	253
DISplay:SCREENTExt:LABel<x>?	254
DISplay:SCREENTExt:LABel<x>:NAME	255
DISplay:SCREENTExt:LABel<x>:XPOS	256
DISplay:SCREENTExt:LABel<x>:YPOS	257
DISplay:SCREENTExt:STATE	258
DISplay:STYle	259
DISplay:TRIGBar	260
DISplay:TRIGT	262
DISplay:VARPersist	263
EVENT?	265
EVMsg?	266
EVQty?	267
EXPort	268
EXPort:FILEName	269
EXPort:FORMat	270
EXPort:IMAGe	271
EXPort:PALEtte	272
EXPort:READOuts	273
EXPort:VIEW	274
FACTory	275
FASTAcq?	277
FASTAcq:STATE	278
FILESystem?	280
FILESystem:COpy	281
FILESystem:DELEte	283

FILESystem:DIR?	284
FILESystem:MKDir	285
FILESystem:PRInt	286
FILESystem:READFile	288
FILESystem:REName	290
FILESystem:RMDir	291
FILESystem:WRITEFile	292
HARDCopy	293
HARDCopy:FILEName	294
HARDCopy:IMAGe	295
HARDCopy:LAYout	296
HARDCopy:PALEtte	297
HARDCopy:READOuts	298
HARDCopy:VIEW	299
HDR	300
HEADer	302
HISTogram?	304
HISTogram:Box	305
HISTogram:BOXPcnt	307
HISTogram:COUNT	309
HISTogram:DATA?	310
HISTogram:DISplay	312
HISTogram:FUNCTion	313
HISTogram:MODE	314
HISTogram:SIZE	316
HISTogram:SOURce	317
HISTogram:STATE	319
HORizontal?	320
HORizontal:DIVisions?	321
HORizontal:FASTframe:COUNT	322
HORizontal:FASTframe:REF:FRAME	323
HORizontal:FASTframe:LENGth	324
HORizontal:FASTframe:MULTipleframes:MODE	325
HORizontal:FASTframe:MULTipleframes:FRAMEStart: CH<x>	326
HORizontal:FASTframe:MULTipleframes:FRAMEStart: MATH<x>	328
HORizontal:FASTframe:MULTipleframes:FRAMEStart: REF<x>	330
HORizontal:FASTframe:MULTipleframes:NUMFRames: CH<x>	332
HORizontal:FASTframe:MULTipleframes:NUMFRames: MATH<x>	333
HORizontal:FASTframe:MULTipleframes:NUMFRames: REF<x>	335
HORizontal:FASTframe:REF:SOURce	337
HORizontal:FASTframe:SELECTED:<wfm>	338

HORizontal:FASTframe:STATE	340
HORizontal:FASTframe:TIMEStamp:ALL:<wfm>?	342
HORizontal:FASTframe:TIMEStamp:BETWEEN:<wfm>?	344
HORizontal:FASTframe:TIMEStamp:DELTA:<wfm>?	346
HORizontal:FASTframe:TIMEStamp:FRAME:<wfm>?	347
HORizontal:FASTframe:TIMEStamp:REF?	348
HORizontal:FASTframe:TIMEStamp:SELECTED:<wfm>?	349
HORizontal:FASTframe:TRACK	350
HORizontal:MAIn?	352
HORizontal[:MAIn]:DELay:POSition	353
HORizontal[:MAIn]:DELay:MODE	354
HORizontal[:MAIn]:DELay:TIME	355
HORizontal:MAIn:INTERPRatio?	356
HORizontal[:MAIn]:POSition	357
HORizontal:MAIn:SAMPLERate	358
HORizontal[:MAIn]:SCALE	359
HORizontal:MAIn:UNIts	360
HORizontal:MAIn:UNIts:STRing	361
HORizontal:POSition	362
HORizontal:RECOrdlength	363
HORizontal:RESOLution	364
HORizontal:ROLL	365
HORizontal:SCALE	366
HORizontal:TRIGger:POSition	367
HORizontal:TIMEStamp:CHX?	368
HORizontal:TIMEStamp:REFX?	369
ID?	370
LOCK	371
MASK?	372
MASK:AUTOAdjust	373
MASK:AUTOAdjust:HDELTA	374
MASK:AUTOAdjust:VDELTA	375
MASK:AUTOSet:AUTOAdjust	376
MASK:AUTOSet:HPOS	377
MASK:AUTOSet:HSCALE	378
MASK:AUTOSet:MODE	379
MASK:AUTOSet:OFFSEtAdj	380
MASK:AUTOSet:STANdard	381
MASK:AUTOSet:TRIGger	389
MASK:AUTOSet:VPOS	390
MASK:AUTOSet:VSCALE	391
MASK:COpy:USER	392
MASK:COUNT:FAILURES?	393
MASK:COUNT:HITS?	394
MASK:COUNT:RESET	395
MASK:COUNT:SEG<m>:HITS?	396

MASK:COUNT:STATE	397
MASK:COUNT:TESTS?	398
MASK:COUNT:TOTAl?	399
MASK:COUNT:VIOLATIONS?	400
MASK:COUNT:WAVEFORMS?	401
MASK:DISPlay	402
MASK:FILTer	403
MASK:HIGHLIGHTHits	404
MASK:INVert	405
MASK:LOCK	406
MASK:MARgin:PERCent	407
MASK:MARgin:STATE	408
MASK:MASKPRE:AMPlitude	409
MASK:MASKPRE:HSCALE	410
MASK:MASKPRE:HTRIGPOS	411
MASK:MASKPRE:PATTERNBITS	412
MASK:MASKPRE:PRESAMPBITS	413
MASK:MASKPRE:RECOrdlength	414
MASK:MASKPRE:SERIALTRIG	415
MASK:MASKPRE:TRIGTOSAMP	417
MASK:MASKPRE:VOFFSet	418
MASK:MASKPRE:VPOS	419
MASK:MASKPRE:VSCALE	420
MASK:MASKPRE:WIDth	421
MASK:POLarity	422
MASK:SEG<m> DELEte	423
MASK:SEG<m>:NR_Pt?	424
MASK:SEG<m>:POINTS	425
MASK:SOUrce	426
MASK:STANdard	427
MASK:STOPOnviolation	435
MASK:TEST:AUX:COMPLetion	436
MASK:TEST:AUX:FAILure	437
MASK:TEST:BEEP:COMPLetion	438
MASK:TEST:BEEP:FAILure	439
MASK:TEST:DELay	440
MASK:TEST:HARDCopy	441
MASK:TEST:LOG:FAILure	442
MASK:TEST:REPeat	443
MASK:TEST:SAMple	444
MASK:TEST:SAMple:THReshold	445
MASK:TEST:SAVEWFM	446
MASK:TEST:SAVEWFM:FILEName	447
MASK:TEST:SRQ:COMPLetion	448
MASK:TEST:SRQ:FAILure	449
MASK:TEST:STATE	450

MASK:TEST:STATUS?	451
MASK:TEST:STOP:FAILure	452
MASK:TEST:THReshold	453
MASK:TEST:WAVEform	454
MASK:USER:AMPlitude	455
MASK:USER:BITRate	456
MASK:USER:HSCAlE	457
MASK:USER:HTRIGPOS	458
MASK:USER:PATTERNBITS	459
MASK:USER:PRESAMPBITS	460
MASK:USER:RECOrdlength	461
MASK:USER:SEG<m> DELEte	462
MASK:USER:SEG<m>:NR_Pt?	463
MASK:USER:SEG<m>:POINTS	464
MASK:USER:SERIALTRIG	465
MASK:USER:TRIGTOSAMP	467
MASK:USER:VOFFSet	468
MASK:USER:VPOS	469
MASK:USER:VSCAlE	470
MASK:USER:WIDth	471
MATH<x>	472
MATH<x>:DEFine	473
MATH<x>:LABEL:NAME	475
MATH<x>:LABEL:XPOS	476
MATH<x>:LABEL:YPOS	477
MATH<x>:NUMAVg	478
MATH<x>:POSition	479
MATH<x>:SCAlE	481
MATH<x>:SPECTral?	483
MATH<x>:SPECTral:CENTER	484
MATH<x>:SPECTral:GATEPOS	485
MATH<x>:SPECTral:GATEWIDTH	486
MATH<x>:SPECTral:LOCK	487
MATH<x>:SPECTral:MAG	489
MATH<x>:SPECTral:PHASE	490
MATH<x>:SPECTral:REFLEVEL	491
MATH<x>:SPECTral:REFLEVELOffset	492
MATH<x>:SPECTral:RESBw	494
MATH<x>:SPECTral:SPAN	495
MATH<x>:SPECTral:SUPPpress	497
MATH<x>:SPECTral:UNWRap	498
MATH<x>:SPECTral:WINDow	499
MATHVAR:VAR<n>	501
MEASUrement?	502
MEASUrement:ANNOtation:IMMEDState	504
MEASUrement:ANNOtation:STATE	505

MEASUrement:ANNOtation:TYPe	506
MEASUrement:ANNOtation:NUMX?	507
MEASUrement:ANNOtation:NUMY?	508
MEASUrement:ANNOtation:X<x>?	509
MEASUrement:ANNOtation:Y<x>?	510
MEASUrement:GATing	511
MEASUrement:IMMed?	513
MEASUrement:IMMed:DELaY?	514
MEASUrement:IMMed:DELaY:DIREction	515
MEASUrement:IMMed:DELaY:EDGE[1]	517
MEASUrement:IMMed:DELaY:EDGE2	518
MEASUrement:IMMed:NOISe	519
MEASUrement:IMMed:REFLevel?	520
MEASUrement:IMMed:REFLevel:ABSolute:HIGH	521
MEASUrement:IMMed:REFLevel:ABSolute:LOW	522
MEASUrement:IMMed:REFLevel:ABSolute:MID[1]	523
MEASUrement:IMMed:REFLevel:ABSolute:MID2	524
MEASUrement:IMMed:REFLevel:METhod	525
MEASUrement:IMMed:REFLevel:PERCent:HIGH	526
MEASUrement:IMMed:REFLevel:PERCent:LOW	527
MEASUrement:IMMed:REFLevel:PERCent:MID[1]	528
MEASUrement:IMMed:REFLevel:PERCent:MID2	529
MEASUrement:IMMed:SOURCE[1]	530
MEASUrement:IMMed:SOURCE2	532
MEASUrement:IMMed:TYPe	534
MEASUrement:IMMed:UNIts?	542
MEASUrement:IMMed:VALue?	544
MEASUrement:MEAS<x>?	545
MEASUrement:MEAS<x>:COUnT?	546
MEASUrement:MEAS<x>:DELaY?	547
MEASUrement:MEAS<x>:DELaY:DIREction	548
MEASUrement:MEAS<x>:DELaY:EDGE[1]	550
MEASUrement:MEAS<x>:DELaY:EDGE2	551
MEASUrement:MEAS<x>:MAXimum?	552
MEASUrement:MEAS<x>:MEAN?	553
MEASUrement:MEAS<x>:METhod	554
MEASUrement:MEAS<x>:NOISe	555
MEASUrement:MEAS<x>:MINImum?	556
MEASUrement:MEAS<x>:REFLevel?	557
MEASUrement:MEAS<x>:REFLevel:ABSolute:HIGH	558
MEASUrement:MEAS<x>:REFLevel:ABSolute:LOW	559
MEASUrement:MEAS<x>:REFLevel:ABSolute:MID[1]	560
MEASUrement:MEAS<x>:REFLevel:ABSolute:MID2	561
MEASUrement:MEAS<x>:REFLevel:METhod	562
MEASUrement:MEAS<x>:REFLevel:PERCent:HIGH	563
MEASUrement:MEAS<x>:REFLevel:PERCent:LOW	564

MEASUrement:MEAS<x>:REFLevel:PERCent:MID[1]	565
MEASUrement:MEAS<x>:REFLevel:PERCent:MID2	566
MEASUrement:MEAS<x>:SOURCE[1]	567
MEASUrement:MEAS<x>:SOURCE2	568
MEASUrement:MEAS<x>:STATE	570
MEASUrement:MEAS<x>:STDdev?	572
MEASUrement:MEAS<x>:TYPe	573
MEASUrement:MEAS<x>:UNIts?	581
MEASUrement:MEAS<x>:VALue?	583
MEASUrement:METhod	584
MEASUrement:NOISe	586
MEASUrement:REFLevel:ABSolute:HIGH	587
MEASUrement:REFLevel:ABSolute:LOW	588
MEASUrement:REFLevel:ABSolute:MID	589
MEASUrement:REFLevel:ABSolute:MID2	590
MEASUrement:REFLevel:METhod	591
MEASUrement:REFLevel:PERCent:HIGH	592
MEASUrement:REFLevel:PERCent:LOW	593
MEASUrement:REFLevel:PERCent:MID[1]	594
MEASUrement:REFLevel:PERCent:MID2	595
MEASUrement:STATIstics:COUNT	596
MEASUrement:STATIstics:MODE	597
MEASUrement:STATIstics:WEIghting	598
NEWpass	599
PASSWord	600
RECAIl:MASK	601
RECAIl:SETUp	602
RECAIl:WAVEform	604
REF<x>:HORizontal:POSition	605
REF<x>:LABel?	606
REF<x>:LABel:NAME	607
REF<x>:LABel:XPOS	608
REF<x>:LABel:YPOS	609
REF<x>:VERTical:POSition	610
REF<x>:VERTical:SCAle	612
REM	614
ROSc:SOUrce	615
ROSc:STate?	617
SAVE:MASK	618
SAVE:SETUp	619
SAVE:WAVEform	621
SAVE:WAVEform:FILEFormat	623
SET?	626
SElect?	633
SElect:<wfm>	634
SElect:CONTRol<wfm>	635

SETUp:NAME	637
TEKSecure	638
TEST	639
TEST:RESults?	640
TEST:RESults:VERBoSe?	641
TEST:STOP	642
TIMe	643
TRIGger	644
TRIGger:A	646
TRIGger:A:COMMunication:AMI...PULSEForm	648
TRIGger:A:COMMunication:AMI...THReShold:HIGH	649
TRIGger:A:COMMunication:AMI...THReShold:LOW	650
TRIGger:A:COMMunication:BITRate	651
TRIGger:A:COMMunication:CMI:PULSEForm	652
TRIGger:A:COMMunication:CODe	653
TRIGger:A:COMMunication:CLOCK:POLarity	654
TRIGger:A:COMMunication:SOURce	655
TRIGger:A:COMMunication:SOURce:TYPe	656
TRIGger:A:COMMunication:STANdard	657
TRIGger:A:EDGE?	661
TRIGger:A:EDGE:COUPling	662
TRIGger:A:EDGE:SLOpe	664
TRIGger:A:EDGE:SOURce	665
TRIGger:A:HOLDoff?	666
TRIGger:A:HOLDoff:ACTUal?	667
TRIGger:A:HOLDoff:BY	668
TRIGger:A:HOLDoff:TIMe	670
TRIGger:A:LEVel	671
TRIGger:A:LEVel:CH<x>	672
TRIGger:A:LOGIc?	673
TRIGger:A:LOGIc:CLAss	674
TRIGger:A:LOGIc:FUNcTION	676
TRIGger:A:LOGIc:INPut?	677
TRIGger:A:LOGIc:INPut:CH<x>	678
TRIGger:A:LOGIc:PATtern?	679
TRIGger:A:LOGIc:PATtern:INPut:CH4	680
TRIGger:A:LOGIc:PATtern:WHEn	682
TRIGger:A:LOGIc:PATtern:WHEn:LESSLimit	684
TRIGger:A:LOGIc:PATtern:WHEn:MORELimit	685
TRIGger:A:LOGIc:SETHold?	686
TRIGger:A:LOGIc:SETHold:CLOCK?	687
TRIGger:A:LOGIc:SETHold:CLOCK:EDGE	688
TRIGger:A:LOGIc:SETHold:CLOCK:SOURce	689
TRIGger:A:LOGIc:SETHold:CLOCK:THReShold	690
TRIGger:A:LOGIc:SETHold:DATA?	691
TRIGger:A:LOGIc:SETHold:DATA:SOURce	692

TRIGger:A:LOGIc:SETHold:DATA:THReshold	693
TRIGger:A:LOGIc:SETHold:HOLDTime	694
TRIGger:A:LOGIc:SETHold:QUALify	695
TRIGger:A:LOGIc:SETHold:SETTime	696
TRIGger:A:LOGIc:STATE?	697
TRIGger:A:LOGIc:STATE:INPut:CH4	698
TRIGger:A:LOGIc:STATE:WHEn	699
TRIGger:A:LOGIc:THReshold?	700
TRIGger:A:LOGIc:THReshold:CH<x>	701
TRIGger:A:LOWerthreshold:CH<x>	702
TRIGger:A:MODE	703
TRIGger:A:PULse?	704
TRIGger:A:PULse:CLAss	705
TRIGger:A:PULse:GLItch?	707
TRIGger:A:PULse:GLItch:POLarity	708
TRIGger:A:PULse:GLItch:QUALify	710
TRIGger:A:PULse:GLItch:TRIGIF	711
TRIGger:A:PULse:GLItch:WIDth	713
TRIGger:A:PULse:RUNT?	714
TRIGger:A:PULse:RUNT:POLarity	715
TRIGger:A:PULse:RUNT:QUALify	717
TRIGger:A:PULse:RUNT:THReshold?	718
TRIGger:A:PULse:RUNT:THReshold:BOTH	719
TRIGger:A:PULse:RUNT:THReshold:HIGH	720
TRIGger:A:PULse:RUNT:THReshold:LOW	721
TRIGger:A:PULse:RUNT:WHEn	722
TRIGger:A:PULse:RUNT:WIDth	723
TRIGger:A:PULse:SOUrce	724
TRIGger:A:PULse:TIMEOut?	725
TRIGger:A:PULse:TIMEOut:POLarity	726
TRIGger:A:PULse:TIMEOut:QUALify	728
TRIGger:A:PULse:TIMEOut:TIME	729
TRIGger:A:PULse:TRANSition?	730
TRIGger:A:PULse:TRANSition:DELTATime	731
TRIGger:A:PULse:TRANSition:POLarity	732
TRIGger:A:PULse:TRANSition:QUALify	733
TRIGger:A:PULse:TRANSition:THReshold?	734
TRIGger:A:PULse:TRANSition:THReshold:BOTH	735
TRIGger:A:PULse:TRANSition:THReshold:HIGH	736
TRIGger:A:PULse:TRANSition:THReshold:LOW	737
TRIGger:A:PULse:TRANSition:WHEn	738
TRIGger:A:PULse:WIDth?	739
TRIGger:A:PULse:WIDth:HIGHLimit	740
TRIGger:A:PULse:WIDth:LOWLIMIT	741
TRIGger:A:PULse:WIDth:POLarity	742
TRIGger:A:PULse:WIDth:QUALify	743

TRIGger:A:PULse:WIDth:WHEn	744
TRIGger:A:PULse:WINDow:QUALify	746
TRIGger:A:PULse:WINDow:THReshold:BOth	747
TRIGger:A:PULse:WINDow:THReshold:HIGH	748
TRIGger:A:PULse:WINDow:THReshold:LOW	749
TRIGger:A:PULse:WINDow:TYPe	750
TRIGger:A:PULse:WINDow:WHEn	751
TRIGger:A:PULse:WINDow:WIDth	752
TRIGger:A:SERial:BITRate	753
TRIGger:A:SERial:CLOCK:LEVel	754
TRIGger:A:SERial:CLOCK:POLARITY	755
TRIGger:A:SERial:CLOCK:SOURce	756
TRIGger:A:SERial:CODe	757
TRIGger:A:SERial:DATA:FORMat	758
TRIGger:A:SERial:DATA:PATtern	760
TRIGger:A:SERial:SOURce	763
TRIGger:A:SERial:STANDard	764
TRIGger:A:TYPe	766
TRIGger:A:UPPerthreshold:CH<x>	768
TRIGger:AUXLeVel	769
TRIGger:B	770
TRIGger:B:BY	771
TRIGger:B:EDGE?	773
TRIGger:B:EDGE:COUPling	774
TRIGger:B:EDGE:SLOpe	775
TRIGger:B:EDGE:SOURce	776
TRIGger:B:EVENTS?	777
TRIGger:B:EVENTS:COUNT	778
TRIGger:B:LEVel	779
TRIGger:B:LEVel:CH<x>	780
TRIGger:B:RESET:SOURce	781
TRIGger:B:RESET:STATE	782
TRIGger:B:RESET:TIMEOut	783
TRIGger:B:RESET:THReshold	784
TRIGger:B:RESET:TRANSition	785
TRIGger:B:RESET:TYPe	786
TRIGger:B:STATE	788
TRIGger:B:TIME	789
TRIGger:B:TYPe	790
TRIGger:LVLsrcpreference	791
TRIGger:STATE?	792
UNLock	794
VERBoSe	795
WAVFRm?	796
WAVFRMStream?	799
WFMinpre?	801

WFMInpre:BIT_Nr	802
WFMInpre:BN_Fmt	803
WFMInpre:BYT_Nr	804
WFMInpre:BYT_Or	805
WFMInpre:ENCdg	806
WFMInpre:NR_FR?	807
WFMInpre:NR_Pt	808
WFMInpre:PT_Fmt	809
WFMInpre:PT_Off	811
WFMInpre:WFId	812
WFMInpre:XINcr	813
WFMInpre:XUNit	814
WFMInpre:XZEro	815
WFMInpre:YMUIt	816
WFMInpre:YOFF	817
WFMInpre:YUNit	818
WFMInpre:YZEro	819
WFMOutpre?	820
WFMOutpre:BIT_Nr	821
WFMOutpre:BN_Fmt	822
WFMOutpre:BYT_Nr	823
WFMOutpre:BYT_Or	824
WFMOutpre:ENCdg	825
WFMOutpre:NR_FR?	827
WFMOutpre:NR_Pt?	828
WFMOutpre:PT_Fmt?	829
WFMOutpre:PT_Off?	830
WFMOutpre:PT_OR?	831
WFMOutpre:WFId?	833
WFMOutpre:XINcr?	835
WFMOutpre:XUNit?	836
WFMOutpre:XZEro?	837
WFMOutpre:YMUIt?	838
WFMOutpre:YOFF?	839
WFMOutpre:YUNit?	840
WFMOutpre:YZEro?	841
WLENgth:VALue	842
ZOOM	843
ZOOM:GRAticule:SIZE	845
ZOOM:GRAticule:SPLIT	846
ZOOM:MODE	848
ZOOM:SCROLL:DIREction	849
ZOOM:SCROLL:LOCK	850
ZOOM:SCROLL:SPEED	851
ZOOM:ZOOM<x>	852
ZOOM:ZOOM<x>:SCROLLLock	853

ZOOM:ZOOM<x>:STATE	854
ZOOM:ZOOM<x>:<wfm>:HORIZONTAL:POSITION	855
ZOOM:ZOOM<x>:<wfm>:HORIZONTAL:SCALE	856
ZOOM:ZOOM<x>:<wfm>:VERTICAL:POSITION	857
ZOOM:ZOOM<x>:<wfm>:VERTICAL:SCALE	858

Status and Events

Status and Events	859
Registers	860
Queues	864
Event Handling Sequence	865
Synchronization Methods	866
Messages	874

Programming Examples

Overview of Programming Examples	881
Compiling the Example GPIB Programs	881
Compiling and Linking Example Visual C++ Programs	882

Miscellaneous

Character Charts	883
Reserved Words	884
Factory Default Setup Values	888
GPIB Interface Specifications	895

Copyright and Version Information

Copyright © Tektronix, Inc. All rights reserved. Licensed software products are owned by Tektronix or its suppliers and are protected by United States copyright laws and international treaty provisions.

Use, duplication or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, or subparagraphs (c)(1) and (2) of the Commercial Computer Software – Restricted Rights clause at FAR 52.227-19, as applicable.

Tektronix products are covered by U.S. and foreign patents, issued and pending. Information in this documentation supersedes that in all previously published material. Specifications and price change privileges reserved.

Tektronix, Inc. P.O. Box 500, Beaverton, OR 97077

TEKTRONIX, TEK, and TEKPROBE are registered trademarks of Tektronix, Inc.

FrameScan is a trademark of Tektronix, Inc.

Getting Started

Introduction

This online programmer guide provides you with the information you need to use GPIB commands for remotely controlling your instrument. With this information, you can write computer programs that will perform functions such as setting the front-panel controls, taking measurements, performing statistical calculations, and exporting data for use in other programs, such as spreadsheets.

Besides the traditional GPIB electronic interface, (referred to as the *physical* GPIB interface), your instrument is provided with a TekVISA GPIB-compatible interface, (referred to as the *virtual* GPIB interface). This is software Application Programming Interface (API) that enables you to communicate with the instrument in a variety of ways, including via the Internet.

With the following two exceptions, these interfaces are completely independent:

- **HEADER.** Command headers enabled or disabled on one interface are correspondingly enabled or disabled on the other interface. Refer to the command descriptions for more detailed information.
- **VERBOSE.** Verbosity enabled or disabled on one interface is correspondingly enabled or disabled on the other interface. Refer to the command description for more detailed information.

Refer to Documentation for information on related manuals and documents.

The programmer guide is divided into the following major topics (books):

- **Getting Started.** This topic introduces you to the online help and provides basic information about setting up your instrument for remote control.
- **Command Syntax.** This topic provides an overview of the command syntax that you will use to communicate with the instrument and other general information about commands, such as how commands and queries are constructed, how to enter commands, constructed mnemonics, and argument types.

-
- **Command Groups.** This topic contains all the commands listed in functional groups. Each group consists of an overview of the commands in that group and a table that lists all the commands and queries for that group. You can click a command in the listing and a detailed description of the command will be displayed.
 - **Status and Events.** This topic discusses the status and event reporting system for the GPIB interface. This system informs you of certain significant events that occur within the instrument. Topics that are discussed include registers, queues, event handling sequences, synchronization methods, and messages that the instrument may return, including error messages.
 - **Miscellaneous.** This topic contains miscellaneous information, such as a list of reserved words, a table of the factory initialization (default) settings, and GPIB interface specifications that may be helpful when using GPIB commands to remotely control the instrument.

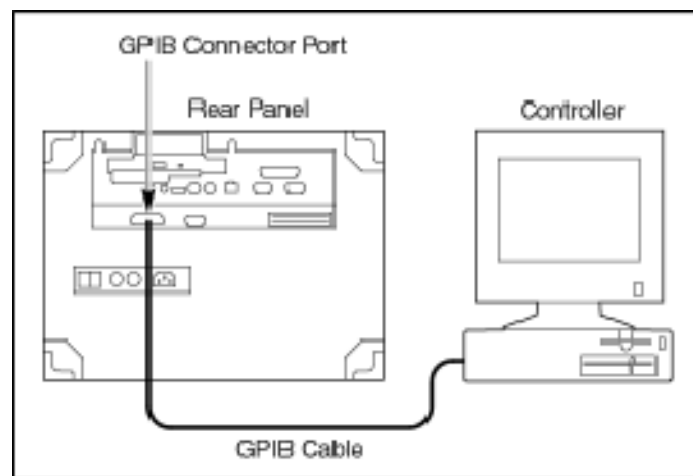
Setting Up Remote Communications

Before setting up your instrument for remote communications using the electronic (*physical*) GPIB interface, you should familiarize yourself with the following GPIB requirements:

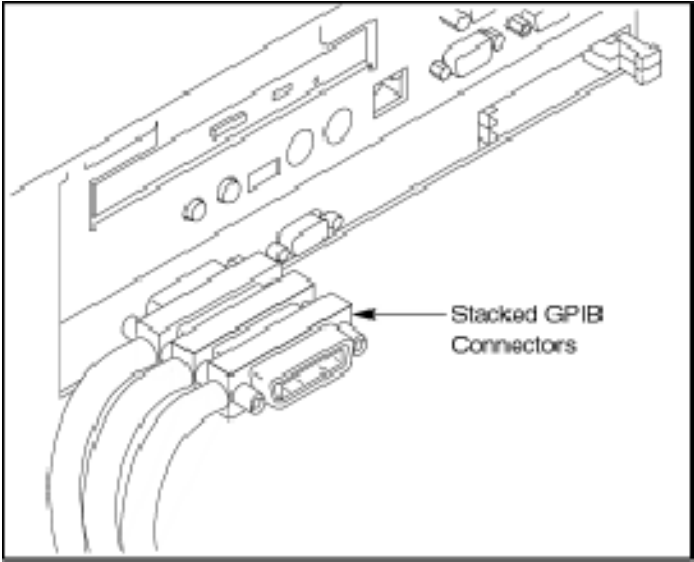
- A unique device address must be assigned to each device on the bus. No two devices can share the same device address.
- No more than 15 devices can be connected to any one line.
- One device should be connected for every 6 feet (2 meters) of cable used.
- No more than 65 feet (20 meters) of cable should be used to connect devices to a bus.
- At least two-thirds of the devices on the network should be powered on while using the network.
- Connect the devices on the network in a star or linear configuration. Do not use loop or parallel configurations.

Connecting to the Instrument

Your instrument has a 24-pin GPIB connector on its rear panel. This connector has a D-type shell and conforms to IEEE Std 488.1–1987. Attach an IEEE Std 488.1–1987 GPIB cable to this connector and to your controller as shown in the following figure.



If necessary, the GPIB connectors can be stacked as shown in the figure below.



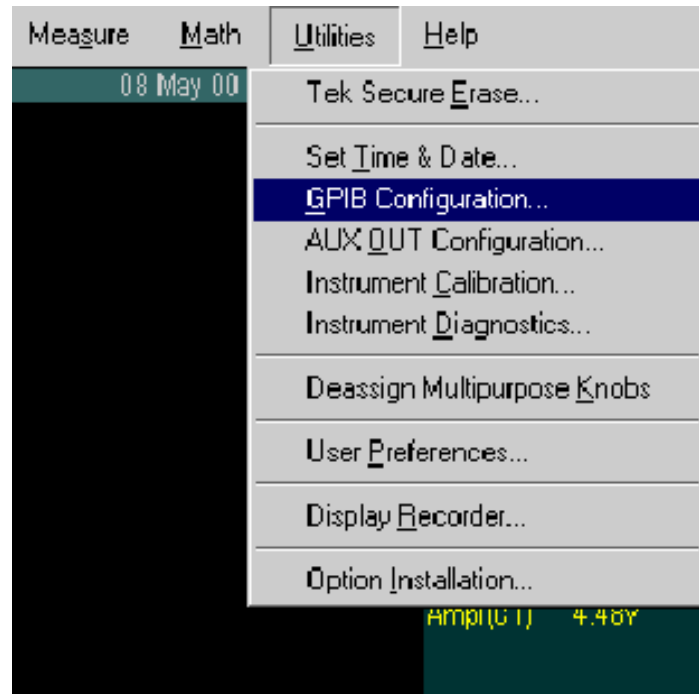
Setting the GPIB Address

To function correctly, your instrument must have a unique device address. The default settings for the GPIB configuration are:

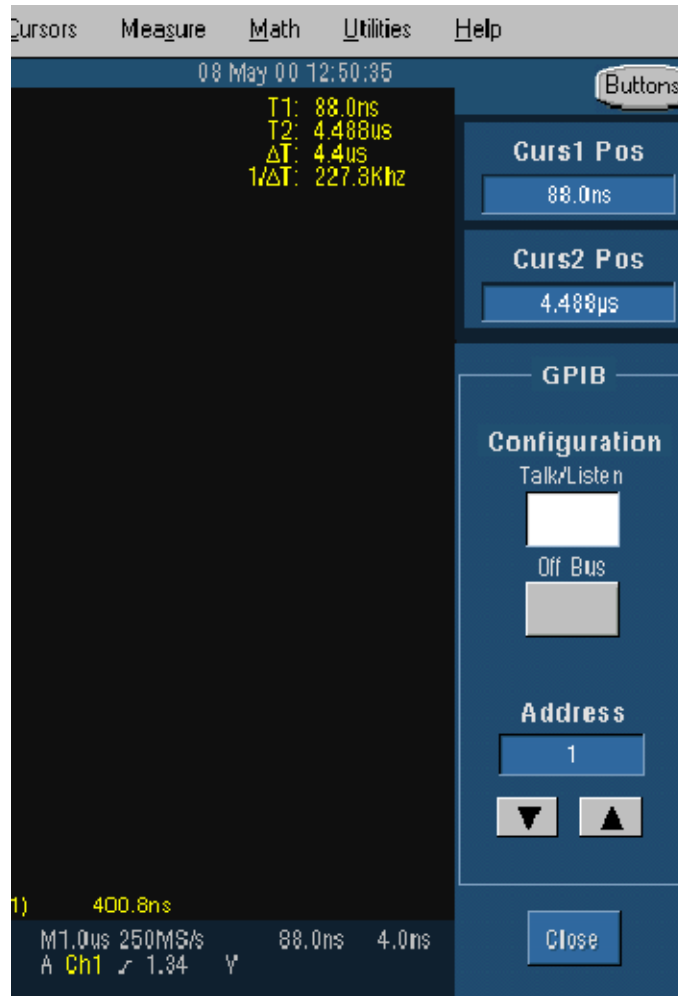
- GPIB Address: 1
- GPIB Mode: GPIB Talk/Listen

To change either of the GPIB settings, do the following:

1. Select GPIB Configuration... from the Utilities menu



2. Click the Configuration Talk/Listen button.



3. Change the GPIB Address to a unique address.
4. Click the Close button.

The instrument is set up for bi-directional communication with your controller.

Documentation



In addition to this *TDS7000B & CSA7000B Series Programmer Online Guide*, the following documentation is included with this instrument.

User Manual. The User Manual has information about installing and operating the Oscilloscope. It also provides concepts and theories about using the oscilloscope that are not covered in the online help.

Quick Reference Guide. A quick reference guide provides basic information on connecting a signal to the oscilloscope and displaying the results.

TDS7000B Series Online Help. This is an online help system that is integrated with the User Interface application that ships with this product.

Optional Service Manual. A service manual is available as an optional accessory. It includes procedures to service the oscilloscope to module levels. It also includes performance verification procedures so you can verify performance to advertised specifications.

Oscilloscope Analysis and Connectivity Made Easy. A book that explores some options for getting data from your instrument into any one of several available analysis tools.

TekVISA Programmer Manual. This manual is available as a printable PDF file on the *TDS7000B and CSA7000B Series Product Software* CD-ROM. The manual describes TekVISA, the Tektronix implementation of the VISA Application Programming Interface (API). TekVISA is industry-compliant software for writing interoperable instrument drivers in a variety of Application Development Environments (ADEs).

Command Syntax

Syntax Overview

You can control the operations and functions of the instrument through the GPIB interface using commands and queries. The related topics listed below describe the syntax of these commands and queries. The topics also describe the conventions that the instrument uses to process them. See the *Command Groups* topic in the table of contents for a listing of the commands by command group or use the index to locate a specific command.

Backus-Naur Form Notation

This documentation describes the commands and queries using Backus-Naur Form (BNF) notation. Refer to the following table for the symbols that are used.

Symbols for Backus-Naur Form	
Symbol	Meaning
< >	Defined element
::=	Is defined as
	Exclusive OR
{ }	Group; one element is required
[]	Optional; can be omitted
...	Previous element(s) may be repeated
()	Comment

Command and Query Structure

Overview

Commands consist of set commands and query commands (usually called commands and queries). Commands modify instrument settings or tell the instrument to perform a specific action. Queries cause the instrument to return data and status information.

Most commands have both a set form and a query form. The query form of the command differs from the set form by its question mark on the end. For example, the set command ACQUIRE:MODE has a query form ACQUIRE:MODE?. Not all commands have both a set and a query form. Some commands have set only and some have query only.

Messages

A command message is a command or query name followed by any information the instrument needs to execute the command or query. Command messages may contain five element types, defined in the following table.

Command Message Elements	
Symbol	Meaning
<Header>	This is the basic command name. If the header ends with a question mark, the command is a query. The header may begin with a colon (:) character. If the command is concatenated with other commands, the beginning colon is required. Never use the beginning colon with command headers beginning with a star (*).
<Mnemonic>	This is a header subfunction. Some command headers have only one mnemonic. If a command header has multiple mnemonics, a colon (:) character always separates them from each other.
<Argument>	This is a quantity, quality, restriction, or limit associated with the header. Some commands have no arguments while others have multiple arguments. A <space> separates arguments from the header. A <comma> separates arguments from each other.
<Comma>	A single comma is used between arguments of multiple-argument commands. Optionally, there may be white space characters before and after the comma.
<Space>	A white space character is used between a command header and the related argument. Optionally, a white space may consist of multiple white space characters.

Commands

Commands cause the instrument to perform a specific function or change one of the settings. Commands have the structure:

```
[ : ]<Header> [ <Space><Argument> [ <Comma><Argument> ] . . . ]
```

A command header consists of one or more mnemonics arranged in a hierarchical or tree structure. The first mnemonic is the base or root of the tree and each subsequent mnemonic is a level or branch off the previous one. Commands at a higher level in the tree may affect those at a lower level. The leading colon (:) always returns you to the base of the command tree.

Queries

Queries cause the instrument to return status or setting information. Queries have the structure:

- [:]<Header>?
- [:]<Header>? [<Space><Argument> [<Comma><Argument>] ...]

You can specify a query command at any level within the command tree unless otherwise noted. These branch queries return information about all the mnemonics below the specified branch or level. For example, `HISTogram:STATistics:STDdev?` returns the standard deviation of the histogram, while `HISTogram:STATistics?` returns all the histogram statistics, and `HISTogram?` returns all the histogram parameters.

Headers

You can control whether the instrument returns headers as part of the query response. Use the `HEADer` command to control this feature. If header is on, the query response returns command headers, then formats itself as a valid set command. When header is off, the response includes only the values. This may make it easier to parse and extract the information from the response. The table below shows the difference in responses.

Comparison of Header Off and Header On Responses		
Query	Header Off	Header On
<code>TIME?</code>	<code>"14:30:00</code>	<code>:TIME"14:30:00</code>
<code>ACQuire:NUMAVg?</code>	<code>100</code>	<code>:ACQUIRE:NUMAVG 100</code>

Clearing the Instrument

You can clear the Output Queue and reset the instrument to accept a new command or query by using the selected Device Clear (DCL) GPIB function. Refer to your GPIB library documentation for further details about the selected Device Clear operation.

Command Entry

Rules

The following rules apply when entering commands:

- You can enter commands in upper or lower case.
- You can precede any command with white space characters. White space characters include any combination of the ASCII control characters 00 through 09 and 0B through 20 hexadecimal (0 through 9 and 11 through 32 decimal).
- The instrument ignores commands consisting of any combination of white space characters and line feeds.

Abbreviating

You can abbreviate many instrument commands. Each command in this documentation shows the abbreviations in capitals. For example, you can enter the command ACQuire:NUMAvg simply as ACQ:NUMA or acq:numa.

Abbreviation rules may change over time as new instrument models are introduced. Thus, for the most robust code, use the full spelling.

If you use the HEADer command to have command headers included as part of query responses, you can further control whether the returned headers are abbreviated or are full-length with the VERBose command.

Concatenating

You can concatenate any combination of set commands and queries using a semicolon (;). The instrument executes concatenated commands in the order received.

When concatenating commands and queries, you must follow these rules:

1. Separate completely different headers by a semicolon and by the beginning colon on all commands except the first one. For example, the commands TRIGger:MODE NORMAl and ACQuire:NUMAVg 10, can be concatenated into the following single command:
TRIGger:MODE NORMAl;:ACQuire:NUMAVg 10
2. If concatenated commands have headers that differ by only the last mnemonic, you can abbreviate the second command and eliminate the beginning colon. For example, you can concatenate the commands ACQuire:MODE ENVelope and ACQuire:NUMAVg 10 into a single command:
ACQuire:MODE ENVelope; NUMAVg 10

The longer version works equally well:

ACQuire:MODE ENVelope;:ACQuire:NUMAVg 10

3. Never precede a star (*) command with a semicolon (or colon):

ACQuire:MODE ENVelope;*OPC

Any commands that follow will be processed as if the star command was not there so the commands, ACQuire:MODE ENVelope;*OPC;NUMAVg 10 will set the acquisition mode to envelope and set the number of acquisitions for averaging to 10.

4. When you concatenate queries, the responses to all the queries are concatenated into a single response message. For example, if the display background color is white and the display foreground color is black, the concatenated query DISPlay:COLor:BACKGround?;FOREGround? will return the following.

If the header is on:

```
:DISPLAY:COLOR:BACKGROUND 7;;DISPLAY:COLOR:FOREGROUND 0
```

If the header is off:

```
7;0
```

5. Set commands and queries may be concatenated in the same message. For example,

ACQuire:MODE SAMple;NUMAVg?;STATE?

is a valid message that sets the acquisition mode to sample. The message then queries the number of acquisitions for averaging and the acquisition state. Concatenated commands and queries are executed in the order received.

Here are some invalid concatenations:

- DISPlay:STYle:NORMAl;ACQuire:NUMAVg 10
(no colon before ACQuire)
- DISPlay:COLor:CURSor1 1;;CURSor2 5
(extra colon before CURSor2; use DISPlay:COLor:CURSor1 1:CURSor2 5 instead)
- DISPlay:STYle:NORMAl;: *OPC
(colon/semicolon before a star (*) command)
- DISPlay:COLor:CURSor1 1;COLor:CURSor2 5
(levels of the mnemonics are different; either remove the second use of COLor or place :DISPlay: in front of COLor:CURSor2 5)

Terminating

This documentation uses <EOM> (End of message) to represent a message terminator.

End of Message Terminator	
Symbol	Meaning
<EOM>	Message terminator

For messages sent to the instrument, the end-of-message terminator must be the END message (EOI asserted concurrently with the last data byte). The instrument always terminates messages with LF and EOI. It allows white space before the terminator. For example, it allows CR LF.

Constructed Mnemonics

Some header mnemonics specify one of a range of mnemonics. For example, a channel mnemonic can be CH1, CH2, CH3, or CH4. You use these mnemonics in the command just as you do any other mnemonic. For example, there is a CH1:POSITION command, and there is also a CH2:POSITION command. In the command descriptions, this list of choices is abbreviated as CH<x>.

Cursor Position Mnemonics

When cursors are displayed, commands may specify which cursor of the pair to use.

Cursor Mnemonics	
Symbol	Meaning
CURSOR<x>	A cursor selector; <x> is either 1 or 2.
POSITION<x>	A cursor selector; <x> is either 1 or 2.
HPOS<x>	A cursor selector; <x> is either 1 or 2.

Math Specifier Mnemonics

Commands can specify the mathematical waveform to use as a mnemonic in the header.

Math Specifier Mnemonics	
Symbol	Meaning
MATH<x>	A math waveform specifier; <x> is 1 through 4.

Measurement Specifier Mnemonics

Commands can specify which measurement to set or query as a mnemonic in the header. Up to eight automated measurements may be displayed.

Measurement Specifier Mnemonics	
Symbol	Meaning
MEAS<x>	A measurement specifier; <x> is 1 through 8.

Channel Mnemonics

Commands specify the channel to use as a mnemonic in the header.

Channel Mnemonics	
Symbol	Meaning
CH<x>	A channel specifier; <x> is 1 through 4.

Reference Waveform Mnemonics

Commands can specify the reference waveform to use as a mnemonic in the header.

Reference Waveform Mnemonics	
Symbol	Meaning
REF<x>	A reference waveform specifier; <x> is 1 thru 4.

Argument Types

Numeric

Many instrument commands require numeric arguments. The syntax shows the format that the instrument returns in response to a query. This is also the preferred format when sending the command to the instrument though any of the formats will be accepted. This documentation represents these arguments as follows:

Numeric Arguments	
Symbol	Meaning
<NR1>	Signed integer value
<NR2>	Floating point value without an exponent
<NR3>	Floating point value with an exponent

Most numeric arguments will be automatically forced to a valid setting, either by rounding or truncating, when an invalid number is input unless otherwise noted in the command description.

Quoted String

Some commands accept or return data in the form of a quoted string, which is simply a group of ASCII characters enclosed by a single quote (') or double quote ("). For example: "this is a quoted string". This documentation represents these arguments as follows:

Quoted String Argument	
Symbol	Meaning
<QString>	Quoted string of ASCII text

A quoted string can include any character defined in the 7-bit ASCII character set. Follow these rules when you use quoted strings:

1. Use the same type of quote character to open and close the string. For example: "this is a valid string".
2. You can mix quotation marks within a string as long as you follow the previous rule. For example, "this is an 'acceptable' string".
3. You can include a quote character within a string simply by repeating the quote. For example: "here is a "" mark".
4. Strings can have upper or lower case characters.
5. If you use a GPIB network, you cannot terminate a quoted string with the END message before the closing delimiter.
6. A carriage return or line feed embedded in a quoted string does not terminate the string, but is treated as just another character in the string.
7. The maximum length of a quoted string returned from a query is 1000 characters.

Here are some invalid strings:

- "Invalid string argument" (quotes are not of the same type)
- "test<EOI>" (termination character is embedded in the string)

Block

Several instrument commands use a block argument form (see the following table).

Block Arguments	
Symbol	Meaning
<NZDig>	A nonzero digit character in the range of 1–9
<Dig>	A digit character, in the range of 0–9
<DChar>	A character with the hexadecimal equivalent of 00 through FF (0 through 255 decimal)
<Block>	A block of data bytes defined as: <pre><Block> ::= {#<NZDig><Dig>[<Dig>...][<DChar>...] #0[<DChar>...]<terminator>}</pre>

<NZDig> specifies the number of <Dig> elements that follow. Taken together, the <Dig> elements form a decimal integer that specifies how many <DChar> elements follow.

Command Groups

Acquisition Command Group

You use the commands in the Acquisition Command Group to set up the modes and functions that control how the instrument acquires the signals you input to the channels and processes them into waveforms.

Using these commands for acquiring waveforms, you can do the following:

- Start and stop acquisitions.
- Control whether each waveform is simply acquired, averaged, or enveloped over successive acquisitions of that waveform.
- Set the controls or conditions that start and stop acquisitions.
- Determine the action the system takes upon completing an acquisition, such as saving all waveforms and taking a measurement when the acquisition is stopped.
- Control acquisition of acquired channel waveforms.
- Set acquisition parameters.

Command	Description
ACQuire?	Returns acquisition parameters
ACQuire:MODE	Sets and returns the acquisition mode
ACQuire:NUMACq?	Returns the number of acquisitions that have occurred
ACQuire:NUMAVg	Sets and returns the number of acquisitions for an averaged waveform
ACQuire:NUMEnv	Sets and returns the number of acquisitions for envelope waveform
ACQuire:NUMSAMples	Sets or returns the number of waveform database (WfmDB) points the oscilloscope can acquire before it stops a single sequence acquisition or stops running a mask test
ACQuire:SAMPLingmode	Sets or returns the sampling mode parameter.
ACQuire:STATE	Starts or stops acquisitions or returns the acquisition state
ACQuire:STOPAfter	Sets or returns whether the acquisition is continuous or single sequence
FASTAcq?	Returns the Fast Acquisition state
FASTAcq:STATE	Enables, disables, or returns Fast Acquisition mode

Alias Command Group

Alias commands allow you to define new commands as a sequence of standard commands. You may find this useful when repeatedly using the same commands to perform certain tasks like setting up measurements.

Aliases are similar to macros but do not include the capability to substitute parameters into alias bodies. The alias mechanism obeys the following rules:

- The alias name must consist of a valid IEEE 488.2 message unit, which may not appear in a message preceded by a colon, comma, or a command or query program header.
- The alias name may not appear in a message followed by program date, a colon, comma, or question mark.
- An alias name must be distinct from any keyword or keyword short form.
- An alias name cannot be redefined without first being deleted using one of the alias deletion functions.
- Alias names do not appear in response messages.
- The Alias commands are defined in Tektronix Standard Codes and Formats. Deviations between that standard and what is specified here will be considered errors unless specifically noted in the command description in this document.

Command

ALias

ALIAS:CATalog?

ALias:DEFIne

ALias:DELEte

ALias:DELEte:ALL

ALias:DELEte:NAME

ALias:STATE

Description

Sets or returns the alias state

Returns a list of the currently defined alias labels

Assigns a sequence of program messages to an alias label

Removes a specified alias

Deletes all existing aliases

Removes a specified alias

Sets or returns the alias state

Calibration Command Group

The calibration commands provide information about the current state of calibration for the mainframe and all resident sampling head channels.

Command	Description
*CAL?	Instructs the oscilloscope to perform a self-calibration
CALibrate?	Returns the calibration status of all the calibration subsystems
CALibrate:CALProbe:CH<x>?	Performs a probe calibration and returns the results
CALibrate:PROBEstate:CH<x>?	Returns the calibration status of the attached probe
CALibrate:RESults?	Returns status of all calibration subsystems without performing an SPC (Signal Path Calibration)
CALibrate:RESults:SPC?	Returns the results of the last SPC operation

Cursor Command Group

You use the commands in the Cursor Command Group to control the cursor display and readout. You can use these commands to control the setups for cursor 1 and cursor 2, such as waveform source, cursor position, and cursor color.

You can also use the commands to select one of the following cursor functions:

- Off—shuts off the display of all cursors.
- Vertical Bars—displays vertical bar cursors, which provide traditional horizontal unit readouts for Cursor 1 (bar1), Cursor 2 (bar2), the delta between them, and 1/delta (results in frequency when the horizontal unit is time).
- Horizontal Bars—displays horizontal bar cursors, which provide traditional vertical unit readouts for Cursor 1 (bar1), Cursor 2 (bar2), and the delta between them.
- Waveform Cursors—displays measurement of horizontal (time) and vertical (voltage) difference between cursors; voltage at the vertical position of the specified cursor; horizontal position of the specified cursor; and measurement units for the cursors. These cursors cannot be moved off the waveform.
- Screen Cursors—displays measurement of horizontal (time) and vertical (voltage) difference between

cursors; voltage at the vertical position of the specified cursor; horizontal position of the specified cursor; and measurement units for the cursors. These cursors can be moved off the waveform.

Command	Description
CURSor?	Returns all cursor settings
CURSor:FUNctIon	Sets or returns cursor type
CURSor:HBArs?	Returns H bars cursor settings
CURSor:HBArs:DELTA?	Returns H bars vertical difference
CURSor:HBArs:POSITIon<x>	Sets or returns H bars cursor<x> vertical position
CURSor:HBArs:UNIts?	Returns H bars cursor units
CURSor:MODE	Sets or returns whether cursors move in unison or separately
CURSor:SCREEN:STYle	Sets or returns the cursor type for screen mode
CURSor:SOUrce	Sets or returns the source for cursors
CURSor:STATE	Sets or returns the cursor state
CURSor:VBArS	Sets or returns the position of the vertical bar cursor
CURSor:VBArS:DELTA?	Returns difference between V bars cursors
CURSor:VBArS:POSITIon<x>	Sets or returns V bars and paired cursor<x> positions
CURSor:VBArS:UNIts	Sets or returns the units for V bars cursors
CURSor:WAVEform:STYle	Sets or returns the cursor type for waveform mode

Diagnostics Command Group

You use the commands in the Diagnostic Command Group to control the selection and execution of diagnostic tests. You can use Test functions to select and execute an item at any level of the test hierarchy. Control functions allow controls for diagnostic execution to be set. RSI-like functions provide the same capability of selecting and running diagnostic tests that are normally available from the RSI.

Command	Description
DIAG:CONTROL:HALT	Enables or disables halting on first diagnostic failure
DIAG:CONTROL:LOOP	Enables or disables looping of diagnostics
DIAG:EXECUTE	Executes currently selected set of diagnostics
DIAG:ITEM?	Returns all data associated with a particular item of the current RSI menu
DIAG:ITEM:FAILURES?	Returns the total number of failures that occurred
DIAG:ITEM:NAME?	Returns the name of the selected menu item
DIAG:ITEM:RESULT?	Returns the result from the last execution of the item
DIAG:ITEM:SUBITEMS?	Returns the number of sub items associated with this item
DIAG:LEVEL	Sets the current level of diagnostic test hierarchy
DIAG:LOOPS?	Returns the number of times that diagnostics were completed during the last execution
DIAG:NAME?	Returns the subsystem name, area and test name of the current diagnostic test
DIAG:NAME:AREA?	Returns the selected area of the current diagnostic test
DIAG:NAME:SUBSYS?	Returns the subsystem of the current diagnostic test
DIAG:NAME:TEST?	Returns the name of the current diagnostic test
DIAG:NUMITEMS?	Returns the number of items on the currently selected level of test hierarchy
DIAG:RESULTS?	Returns a brief pass or fail status of the last diagnostic execution
DIAG:RESULTS:VERBOSE?	Returns a more explanatory message about the results of the last diagnostic execution
DIAG:SELECT:ALL	Selects all available diagnostics
DIAG:SELECT:AREA	Selects one of the available diagnostic areas
DIAG:SELECT:LAST	Sets the last item of a group of items from the same level of test hierarchy
DIAG:SELECT:SUBSYS	Selects one of the available diagnostic subsystems

DIAG:SElect:TEST	Selects one of the available diagnostic tests
DIAG:STATE	Sets the instrument operating state
DIAG:STOP	Terminates the execution of diagnostics
TEST	Selects and executes any item at any level of the test hierarchy
TEST:RESults?	Returns a brief pass or fail status of the last test execution
TEST:RESults:VERBose?	Returns a more explanatory message about the results of the last test execution
TEST:STOP	Terminates the execution of the test

Display Control Command Group

You use the commands in the Display Control Command Group to change the graticule style, the displayed intensities, and to set the characteristics of the waveform display.

You can set the following:

- Background and foreground color as well as cursor, histogram, mask, and measurement annotation colors.
- Display of date and time; cursor, histogram, mask, and measurement readouts; measurement annotations, and the mode in which waveforms are displayed.
- Interpolation to increase sample density of waveform for high acquisition rates

Use the commands to set the style that best displays your waveforms and graticule display properties. Note that the mode you choose globally affects all displayed waveforms.

Command	Description
DISplay?	Returns current display settings
DISplay:CLOCK	Sets or returns the display of the date/time stamp
DISplay:COLOr?	Returns color group settings
DISplay:COLOr:MATHCOLOR	Sets or returns the color to be used for math traces
DISplay:COLOr:PALETTE	Sets or returns the palette to be used for trace display
DISplay:COLOr:PALEtte:IMAGEView	Sets or returns the color palette for image-view waveforms
DISplay:COLOr:PALEtte:RECORDView	Sets or returns the color palette for recordview waveforms
DISplay:COLOr:PALEtte:USER	Sets or returns the user palette group settings
DISplay:COLOr:PALEtte:USER:CARet	Sets or returns the user caret color
DISplay:COLOr:PALEtte:USER:CH<x>	Sets or returns the user palette channel colors
DISplay:COLOr:PALEtte:USER:GRAticule	Sets or returns the user palette graticule
DISplay:COLOr:PALEtte:USER:HISto-gram	Sets or returns the user palette histogram color
DISplay:COLOr:PALEtte:USER:MASK	Sets or returns the user palette mask color
DISplay:COLOr:PAL- Ette:USER:MASK:Highlight	Sets or returns the user palette mask hits color
DISplay:COLOr:PAL- Ette:USER:MATH<x>	Sets or returns the user palette math colors
DISplay:COLOr:PALEtte:USER:REF<x>	Sets or returns the user palette reference colors
DISplay:COLOr:REFCOLOR	Sets or returns the color to be used for reference traces

DISPLAY:DESKew	Sets or returns the Deskew feature
DISplay:FILTer	Sets or returns the type of interpolation to use for the display
DISplay:FORMat	Sets or returns the display format
DISplay:GRAticule	Sets or returns the type of graticule that is displayed
DISplay:INTENSITy?	Returns the waveform saturation level, autobright state and screensaver settings
DISplay:INTENSITy:AUTOBRIGHT	Sets or returns the autobright state
DISplay:INTENSITy:SCREENSAVER:DELAY	Sets or returns the display protection timeout
DISplay:INTENSITy:WAVEform	Sets or returns the intensity saturation of the waveforms
DISplay:INTENSITy:WAVEform:IMAGEView	Sets or returns the waveform saturation level for imageview waveforms
DISplay:INTENSITy:WAVEform:RECORDView	Sets or returns the waveform saturation level for recordview waveforms
DISplay:PERsistence	Sets or returns the display persistence setting
DISplay:PERsistence:RESET	Sets or returns the clearing of persistence data
DISplay:SCREENTExt?	Return all screen text settings
DISplay:SCREENTExt:LABel<x>?	Return screen text setting for a given label
DISplay:SCREENTExt:LABel<x>:NAME	Sets or returns the text to be displayed for a given label
DISplay:SCREENTExt:LABel<x>:XPOS	Sets or returns the horizontal position of a given label
DISplay:SCREENTExt:LABel<x>:YPOS	Sets or returns the vertical position of a given label
DISplay:SCREENTExt:STATE	Sets or returns the state of the display of screen text
DISplay:STYLE	Sets or returns the data display style
DISplay:TRIGBar	Sets or returns the display of the trigger level indicator bar(s)
DISplay:TRIGT	Sets or returns the display of the trigger point indicator
DISplay:VARPersist	Sets or returns the persistence decay time

File System Command Group

You use the commands in the File System Command Group to help you use the built-in hard disk and floppy drive. You can use the commands to do the following:

- List the contents of the default directory
- Create and delete directories
- Create, copy, read, rename, or delete a file

Command	Description
FILESystem?	Returns the file system state
FILESystem:COpy	Copies one or more files to a new file
FILESystem:DELEte	Deletes a named file or directory
FILESystem:DIR?	Returns a list of directory contents
FILESystem:MKDir	Makes a new directory
FILESystem:PRInt	Prints a named file to the named port
FILESystem:READFile	Copies the named file to the GPIB port
FILESystem:REName	Assigns a new name to an existing file
FILESystem:RMDir	Deletes the named directory
FILESystem:WRITEFile	Copies the GPIB port block data to a named file

Hard Copy and Export Command Group

Hard copy commands enable you to make hard copies of data files or send the data to a specified file. In addition, these commands provide you with information about (and the ability to set) file path settings.

Export commands enable you to format and capture waveforms as data files.

Command	Description
EXPort	Copies a waveform to a specified file or returns formatting and file information
EXPort:FILEName	Sets or returns the export file path
EXPort:FORMat	Sets or returns the export image format
EXPort:IMAGe	Sets or returns the export inksaver mode
EXPort:PALEtte	Sets or returns the export color palette
EXPort:READOuts	Sets or returns the position of the readouts for export
EXPort:VIEW	Sets or returns the export screen capture area
HARDCopy	Sends a screen copy that can be stored or printed or returns format, layout, and file information
HARDCopy:FILEName	Sets or returns the hard copy file path
HARDCopy:IMAGe	Sets or returns the hard copy inksaver mode
HARDCopy:LAYout	Sets or returns the hard copy page orientation

HARDCopy:PALEtte	Sets or returns the hard copy color palette
HARDCopy:READOuts	Sets or returns the position of the readouts for hardcopy
HARDCopy:VIEW	Sets or returns the screen capture area

Histogram Command Group

Histogram commands let you select the type of histogram, what part of the waveform should go into the histogram, and histogram statistics. You can use commands from this group to do the following:

- Create a histogram of vertical or horizontal values for a channel, math or reference waveform
- Adjust the limits of the waveform area from which the histogram data is obtained.
- Clear histogram count and restart.

Note: Histograms are not available in FastFrame, Record View XY, or Zoom modes.

Command	Description
HIStogram?	Return all histogram parameters
HIStogram:BOX	Sets or returns the left, top, right, and bottom positions of the histo box, in source waveform coordinates
HIStogram:BOXPct	Sets or returns same as HIStogram:BOX, but in percentage coordinates, with 0,0 upper left and 100,100 lower right
HIStogram:COUNt	Clears histogram count source data and restarts counting
HIStogram:DATA?	Returns an ASCII, comma-separated list of histogram data numbers
HIStogram:DISplay	Sets or returns whether histogram data is displayed and what display type to use
HIStogram:FUNCTion	Sets or returns the type of histogram to create
HIStogram:MODE	Sets or returns the type of histogram to create or the histogram's on-off state
HIStogram:SIZE	Sets or returns the width (or height) of the histogram on the screen in divisions
HIStogram:SOUrce	Sets or returns source waveform and time base for histogram
HIStogram:STATE	Enables or disables histogram calculations or returns their state

Horizontal Command Group

Horizontal commands control the time bases of the instrument. You can set the time per division (or time per point) of the main time base. You can use the Horizontal commands to do the following:

- Set the scale, horizontal position and reference, and units of the time base
- Get the screen resolution, time of first point and time of last point, or get all the horizontal settings
- Enable or disable the display of the time base

You may substitute SECdiv for SCALE in the horizontal commands. This provides program compatibility with earlier models of Tektronix oscilloscopes.

Horizontal Fastframe (also known as memory segmentation) commands are designed for situations where you want to capture a series of triggered acquisitions with minimal dead time between them. You can use the Horizontal Fastframe commands to do the following:

- Maintain Fastframe acquisitions with time stamp information for channel-based waveforms (in Sample, Peak Defect, and Hi Res modes)
- Display a list of time stamp and basic delta information on a Fastframe waveform, or Fastframe waveforms within a single frame (in either YT or XY format)
- Zoom Fastframe waveforms within a single frame

Command	Description
HORizontal?	Returns all learnable settings for the horizontal commands
HORizontal:DIVisions?	Returns the number of graticule divisions over which the waveform is displayed
HORizontal:FASTframe:COUNT	Sets or returns FastFrame frame count
HORizontal:FASTframe:LENGth	Sets or returns the horizontal record length to the number of sample points per frame
HORizontal:FASTframe:MULTiple-frames:MODE	Sets or returns FastFrame mode options
HORizontal:FASTframe:MULTiple-frames:FRAMEStart:CH<x>	Sets or returns the FastFrame start frame for the specified channel
HORizontal:FASTframe:MULTiple-frames:FRAMEStart:MATH<x>	Sets or returns the FastFrame start frame for a Math waveform
HORizontal:FASTframe:MULTiple-frames:FRAMEStart:REF<x>	Sets or returns the FastFrame start frame for a Reference waveform
HORizontal:FASTframe:MULTiple-frames:NUMFRames:CH<x>	Sets or returns the number of frames for the specified channel of FastFrame mode

HORizontal:FASTframe:MULTiple-frames:NUMFRames:MATH<x>	Sets or returns the number of frames for a FastFrame Math waveform
HORis:NUMFRames:REF<x>	Sets or returns the number of frames for a FastFrame Reference waveform
HORizontal:FASTframe:REF:FRame	Sets or returns the FastFrame reference frame number
HORizontal:FASTframe:REF:Source	Sets or returns the FastFrame Reference waveform source
HORizontal:FASTframe:SE-LECTED:<wfm>	Sets or returns the FastFrame Selected frame number on the specified waveform
HORizontal:FASTfzonal:FAST-frame:MULTipleframerame:STATE	Sets or returns the state of the FastFrame acquisition
HORizontal:FASTframe:TIME-Stamp:All:<wfm>?	Returns the frame number and time stamp for each frame between requested frames within the specified waveform
HORizontal:FASTframe:TIMESTAMP:BE-Tween:<wfm>?	Returns the relative trigger for the delta time between the specified frames
HORizontal:FASTframe:TIME-Stamp:DELTA:<wfm>?	Returns the relative time between the triggers of the FastFrame Selected and the FastFrame Reference, within the specified waveform
HORizontal:FASTframe:TIME-Stamp:FRame:<wfm>?	Returns the absolute trigger date and time for the specified frame and waveform.
HORizontal:FASTframe:TIME-Stamp:REF?	Returns the absolute trigger date and time for FastFrame reference
HORizontal:FASTframe:TIMESTAMP:SE-LECTED:<wfm>?	Returns the absolute trigger date and time for FastFrame Selected, within the specified waveform
HORizontal:FASTframe:TRACK	Sets or returns the state of the FastFrame tracking feature
HORizontal:MAIn?	Returns the time per division of the main time base
HORizontal[:MAIn]:DELay:POSition	Sets or returns the main time base position when Horizontal Delay Mode is turned off
HORizontal[:MAIn]:DELay:MODE	Sets or returns the main time base trigger delay mode
HORizontal[:MAIn]:DELay:TIME	Sets or returns the main time base trigger delay time
HORizontal:MAIn:INTERPRatio?	Returns the main horizontal time base interpolation ratio
HORizontal[:MAIn]:POSition	Sets or returns the waveform horizontal position on the display
HORizontal:MAIn:SAMPLERate	Sets or returns the horizontal sample rate to the desired number of samples per second
HORizontal[:MAIn]:SCALE	Sets or returns the time per division for the main time base
HORizontal:MAIn:UNIts	Sets or returns the units for the horizontal main time base
HORizontal:MAIn:UNIts:STRing	Sets or returns the units string for the horizontal main time base trigger delay
HORizontal:POSition	Positions the waveform horizontally on the display or returns its position or returns the position

HORizontal:RECOrdlength	Sets or returns the horizontal record length to the number of data points in each frame and returns the current horizontal record length
HORizontal:RESOLution	Sets (or returns) the horizontal record length to the number of data points in each frame and simultaneously adjusts the sample rate to maintain a constant time/division
HORizontal:ROLL	Sets or returns the horizontal roll mode
HORizontal:SCALE	Sets or returns the time per division for the main time base
HORizontal:TIMESTamp:CHX?	Returns the absolute trigger date and time for the specified frame and waveform.
HORizontal:TIMESTamp:REFX?	Returns the absolute trigger date and time for FastFrame reference
HORizontal:TRIGger:POSition	Sets or returns the position of the horizontal trigger

Mask Command Group

You use the commands in the Mask Command Group to compare incoming waveforms against standard or user-defined telecommunications masks. When the oscilloscope finds the waveforms that fall inside or outside the mask limits, it can perform tasks such as generating hard copy, emitting a beep or an SRQ, stopping and waiting for your input, or some combination of these actions.

A mask is a set of polygonal regions on the screen. Every vertical line of the screen intersects the polygon in zero, one, or two places, but never in more than two places. (A vertical line that intersects a vertical mask border is counted.) You have to break up more complicated polygons into two separate segments. Unlike limit testing, the inside of a mask is the region where waveform data would not normally fall. A telecommunications standard requires up to eight of these masks. Pulse standards always have two masks. Standards with eye patterns usually have three masks, but some have four.

Mask commands work with TDS7000 Series Oscilloscopes and all CSA7000 Series Communications Signal Analyzers. Some commands for TDS7000 Series oscilloscopes require installation of Option SM.

You use the commands in the Mask Command Group to do the following:

- Specify the waveform source to test and mask to use
- Specify whether to use, and the size of, mask margins, which allow you to shrink or expand an existing set of polygons by a specified percentage.
- Specify whether to display a readout of hits and the mask on screen. Options also exist for auto fitting the incoming waveforms to match the mask you choose
- Select industry-standard masks that support a variety of electrical and optical communication standards
- Define and edit your own custom mask; create an entirely new mask, or use a standard mask as a starting reference, and edit it to meet your needs
- Enable, disable, or reset the mask counts. Once you turn on mask counting, it remains on until you explicitly turn it off

Command	Description
MASK?	Returns the states of all settable mask parameters
MASK:AUTOAdjust	Optimizes signal position within the mask to minimize hits
MASK:AUTOAdjust:HDELTA	Sets or returns how far autofit searches horizontally
MASK:AUTOAdjust:VDELTA	Sets or returns how far autofit searches vertically
MASK:AUTOSet:AUTOAdjust	Controls what happens at the end of a mask autose. If ON, the horz/vert auto adjustment is run ONCE to potentially improve the result of mask autose
MASK:AUTOSet:HPOS	Controls whether the autose algorithm will attempt to change the horizontal position of the signal while attempting to autose
MASK:AUTOSet:HSCALE	Controls whether the autose algorithm will attempt to change the horizontal scale while attempting to autose
MASK:AUTOSet:MODE	Controls whether or not an autose will be automatically done after a standard mask is selected
MASK:AUTOSet:OFFSETAdj	Sets autose to not enforce the rule that, for pulse standards, requires 0V to be in a certain place in the mask. Instead, autose will attempt to measure the DC offset in the signal and use scope offset or position controls to compensate for the DC offset in the signal
MASK:AUTOSet:STANDARD	Selects the standard mask in autose
MASK:AUTOSet:TRIGGER	Controls if a MASK:AUTOSet affects the trigger level
MASK:AUTOSet:VPOS	Controls whether the autose algorithm will affect the vertical position (or offset) of the signal while attempting to autose
MASK:AUTOSet:VSCALE	Controls whether the autose algorithm will attempt to change the vertical scale while attempting to autose
MASK:COPY:USER	Copies the currently displayed mask to the User mask
MASK:COUNT:FAILURES?	Returns the number of pass/fail tests that have failed
MASK:COUNT:HITS?	Returns the total number of mask hits on all mask segments
MASK:COUNT RESET	Resets to zero the number of hits for all mask segments on the specified source
MASK:COUNT:SEG<m>:HITS?	Returns the number of hits in mask segment <m>
MASK:COUNT:STATE	Sets or returns the mask hits count state
MASK:COUNT:TESTS?	Returns the number of pass/fail tests that have occurred
MASK:COUNT:TOTAL?	Returns the sum of all hits in all mask segments
MASK:COUNT:VIOLATIONS?	Returns the number of test violations that have occurred in the current pass/fail test

MASK:COUNT:WAVEFORMS?	Returns the number of waveforms that have been acquired and processed during pass/fail testing
MASK:DISPlay	Sets or returns whether or not defined masks are displayed on the screen
MASK:FILTer	Sets or returns whether the selected optical reference receiver mask filter is enabled
MASK:HIGHLIGHTHits	Sets or returns whether or not hits in a mask are highlighted in different colors than other waveform data
MASK:INVert	Controls whether the mask is drawn inverted
MASK:LOCK	Sets or returns the mask lock to waveform mode
MASK:MARgin:PERCent	Sets or queries the mask margin percentage
MASK:MARgin:STATE	Sets or returns the state of the mask margins
MASK:MASKPRE:AMPitude	Sets or returns the current mask's nominal pulse amplitude in volts.
MASK:MASKPRE:HSCale	Sets or returns the nominal timing resolution used to draw the current mask, in time/division
MASK:MASKPRE:HTRIGPOS	Sets or returns the nominal tripper position.(pulse leading edge) used to draw the current mask as a fraction of the display width
MASK:MASKPRE:PATTERNBITS	Sets or returns the number of bits used for serial trigger for the current mask standard
MASK:MASKPRE:PRESAMPBITS	Sets or returns the number of bits before the (isolated one) pulse leading edge in the serial trigger pass/fail testing
MASK:MASKPRE:RECOrdlength	Sets or returns the nominal record length for pulse mask testing
MASK:MASKPRE:SERIALTRIG	Sets or returns the type of triggering used in pass/fail testing of the current mask
MASK:MASKPRE:TRIGTOSAMP	Sets or returns the time, in seconds, from the (leading edge) trigger position to the pulse bit sampling position
MASK:MASKPRE:VOFFSet	Sets or returns the nominal vertical offset, in volts, used to vertically offset the input channels
MASK:MASKPRE:VPOS	Sets or returns the nominal vertical position, in divisions, used to vertically position the input channels
MASK:MASKPRE:VSCALE	Sets or returns the nominal vertical scale, in volts per division, used to vertically scale the input channels
MASK:MASKPRE:WIDth	Sets or returns the nominal bit width in seconds
MASK:POLarity	Sets or returns the input waveform polarity for the pass/fail test
MASK:SEG<m> DELEte	Deletes the specified mask segment from the Current mask

MASK:SEG<m>:NR_Pt?	Returns the number of points that make up the specified mask segment of the Current mask
MASK:SEG<m>:POINTS	Sets or returns the X-Y user coordinates of all points in the Current mask segment
MASK:SOUrce	Sets or reports which source will be compared against the mask(s) when counting is turned on
MASK:STANdard	Deletes the existing mask and sets the selected standard mask
MASK:STOPOnviolation	Sets or returns the stop on violation setting
MASK:TEST:AUX:COMPLetion	Sets or returns the test:aux on pass/fail test completion mode
MASK:TEST:AUX:FAILure	Sets or returns the test:aux status on pass/fail test failure mode
MASK:TEST:BEEP:COMPLetion	Sets or returns the beep on pass/fail test completion mode
MASK:TEST:BEEP:FAILure	Sets or returns the beep status on pass/fail test failure mode
MASK:TEST:DELay	Sets or returns the amount of time, in seconds, to wait after starting pass/fail testing to evaluate waveforms
MASK:TEST:HARDCopy	Sets or returns the hard copy in pass/fail failure mode
MASK:TEST:LOG:FAILure	Sets or returns the log status on pass/fail test failure mode
MASK:TEST:REPeat	Sets or returns the repeat pass/fail test on completion mode
MASK:TEST:SAMple	Sets or returns the minimum number of waveform database (WfmDB) points the oscilloscope can acquire before it stops a single sequence acquisition or stops running a mask test
MASK:TEST:SAMple:THReshold	Sets or returns the minimum number of hits in mask regions needed to cause the pass/fail status to change from PASSING to FAILING
MASK:TEST:SAVEWFM	Sets or returns the save waveform to file on failure setting
MASK:TEST:SAVEWFM:FILEName	Sets or returns the name of the directory and file to use with the MASK:TEST:SAVEWFM command
MASK:TEST:SRQ:COMPLetion	Sets or returns an SRQ on pass/fail test completion mode
MASK:TEST:SRQ:FAILure	Sets or returns the SRQ status on pass/fail test failure mode
MASK:TEST:STATE	Sets or returns the state of mask pass/fail testing
MASK:TEST:STATUS?	Returns the pass/fail test status
MASK:TEST:STOP:FAILure	Sets or returns the stop status on pass/fail test failure mode
MASK:TEST:THReshold	Sets or returns the number of failed tested waveforms needed in each pass/fail test to cause the pass/fail status to change from PASSING to FAILING
MASK:TEST:WAVEform	Sets or returns the number of waveforms to test during pass/fail testing

MASK:USER:AMplitude	Sets or returns the current mask's nominal pulse amplitude in volts
MASK:USER:BITRate	Sets or returns the user mask's bit rate
MASK:USER:HSCAle	Sets or returns the nominal timing resolution used to draw the mask, in time/division
MASK:USER:HTRIGPOS	Sets or returns the nominal tripper position.(pulse leading edge) used to draw the mask as a fraction of the display width
MASK:USER:PATTERNBITS	Sets or returns the number of bits used for serial trigger for the User mask standard
MASK:USER:PRESAMPBITS	Sets or returns the number of bits before the (isolated one) pulse leading edge in the serial trigger pass/fail testing
MASK:USER:RECOrdlength	Sets or returns the nominal record length for pulse mask testing
MASK:USER:SEG<m> DELEte	Deletes the specified mask segment from the User mask
MASK:USER:SEG<m>:NR_Pt?	Returns the number of points that make up the specified User mask segment
MASK:USER:SEG<m>:POINTS	Sets or returns the X-Y user coordinates of all points in the specified User mask segment
MASK:USER:SERIALTRIG	Sets or returns the type of triggering used in pass/fail testing of the User mask
MASK:USER:TRIGTOSAMP	Sets or returns the time, in seconds, from the (leading edge) trigger position to the pulse bit sampling position
MASK:USER:VOFFSet	Sets or returns the nominal vertical offset, in volts, used to vertically offset the input channels
MASK:USER:VPOS	Sets or returns the nominal vertical position, in divisions, used to vertically position the input channels
MASK:USER:VSCAle	Sets or returns the nominal vertical scale, in volts per division, used to vertically scale the input channels
MASK:USER:WIDth	Sets or returns the nominal bit width in seconds
RECALL:MASK	Recalls a stored mask as the User mask.
SAVE:MASK	Saves the User mask to a mass storage file

Math Command Group

You use the commands in the Math Command Group to create and define math waveforms. You can define and display up to four math waveforms simultaneously. Use the available math functions to define your math waveform.

The math waveform you create depends on sources listed in the math expression. If you change these sources, the math waveforms you previously defined will be affected.

Math expressions can be simple, containing no mathematical computation (such as CH1, which specifies that a waveform shows the signal source of channel 1). Math expressions can also be complex, consisting of 100 plus characters and comprising many sources, functions, and operands.

The acquisition of a live waveform can stop for several reasons: You can turn off the channel, stop the waveform (via Run/Stop from the Horiz/Acq menu), or stop the trigger (via Run/Stop from the Trig menu). When you turn off the channel, math continues and data is acquired but not displayed. When you stop either the waveform or the trigger, the math no longer calculates but the last math calculation performed is displayed.

When a live waveform update or reference waveform is altered, math waveforms containing those waveforms as sources are also updated to reflect the changes. Also, sources must exist but do not need to be displayed to be used in and to update math waveforms.

Command	Description
MATH<x>?	Returns the specified math waveform settings
MATH<x>:DEFine	Sets or returns the math<x> waveform definition
MATH<x>:LABEL:NAME	Sets or returns the label string used for annotating the displayed math waveform
MATH<x>:LABEL:XPOS	Sets or returns the X screen offset where the math waveform label is displayed
MATH<x>:LABEL:YPOS	Sets or queries the Y screen offset at which the label attached to the displayed math waveform is displayed
MATH<x>:NUMavg	Sets or returns the acquisition number at which the averaging algorithm will begin exponential averaging
MATH<x>:POSition	Sets or returns the specified math waveform vertical position
MATH<x>:SCALE	Sets or returns the specified math waveform vertical scale (per div)
MATH<x>:SPECTral?	Returns the specified math waveform spectral setups

MATH<x>:SPECTral:CENTER	Sets or returns the center frequency of the spectral analyzer output data span
MATH<x>:SPECTral:GATEPOS	Sets or returns the position of the center of the gate
MATH<x>:SPECTral:GATEWIDTH	Sets or returns the gate width input to the spectral analyzer
MATH<x>:SPECTral:LOCK	Sets or returns the state of the math spectral locking for the specified math waveform
MATH<x>:SPECTral:MAG	Sets or returns the units of the Spectral-Mag function in the specified math string
MATH<x>:SPECTral:PHASE	Sets or returns the units of the Spectral-Phase function in the specified math string
MATH<x>:SPECTral:REFLEVEL	Sets or returns the vertical position of the output data from the spectral analyzer
MATH<x>:SPECTral:REFLEVELOffset	Sets or returns the spectral reference level offset used for calculating the dB value
MATH<x>:SPECTral:RESBw	Sets or returns the resolution bandwidth of the spectral analyzer
MATH<x>:SPECTral:SPAN	Sets the top of the span to a value that is closest to the specified value or returns the current spectral span value
MATH<x>:SPECTral:SUPPress	Sets or returns the phase suppression threshold
MATH<x>:SPECTral:UNWRap	Enables or disables phase unwrap of the spectral analyzer output data or returns whether the phase unwrap of the spectral analyzer output data is enabled
MATH<x>:SPECTral:WINDow	Sets or returns the window function used to multiply the input data to the spectral analyzer
MATHVAR:VAR<n>	Sets or returns the value of VAR<n>

Measurement Command Group

You use the commands in the Measurement Command Group to control the automated measurement system.

Up to eight automated measurements can be displayed on the screen. In the commands, these measurement read-outs are named MEAS<x>, where <x> can be 1 through 8.

In addition to the eight displayed measurements, the measurement commands let you specify an additional measurement, IMMEd. The immediate measurement has no front-panel equivalent. Immediate measurements are never displayed. Because they are computed only when needed, immediate measurements slow the waveform update rate less than displayed measurements.

Whether you use displayed or immediate measurements, you use the VALue? query to obtain measurement results.

Measurement commands can set and query measurement parameters. You can assign some parameters, such as waveform sources, differently for each measurement. Other parameters, such as reference levels, have only one value, which applies to all measurements.

Command	Description
MEASUrement?	Returns all measurement parameters
MEASUrement:ANNOtation:IMMEDStAte	Assigns measurement annotation to the immediate measurement
MEASUrement:ANNOtation:StAte	Assigns measurement annotation to the specified measurement
MEASUrement:ANNOtation:TYpe	Sets the number of annotation markers displayed on screen
MEASUrement:ANNOtation:NUMX?	Returns the maximum number of horizontally-related annotation values defined for the measurement
MEASUrement:ANNOtation:X<x>?	Returns the value of the Xth horizontally-related measurement annotation
MEASUrement:ANNOtation:NUMY?	Returns the maximum number of vertically-related annotation values defined for the measurement
MEASUrement:ANNOtation:Y<x>?	Returns the value of the Xth vertically-related measurement annotation
MEASUrement:GAtING	Sets or returns the measurement gating setting
MEASUrement:IMMed?	Returns all immediate measurement setup parameters
MEASUrement:IMMed:DElay?	Returns information about the immediate delay measurement
MEASUrement:IMMed:DElay:DiREction	Sets or returns the search direction to use for immediate delay measurements
MEASUrement:IMMed:DElay:EDGE[1]	Sets or returns the slope of the edge used for immediate delay "from" waveform measurements

MEASUrement:IMMed:DElay:EDGE2	Sets or returns the slope of the edge used for immediate delay "to" waveform measurements
MEASUrement:IMMed:NOISe	Sets or returns the location of the noise measurement on the eye waveform
MEASUrement:IMMed:REFLevel?	Returns the reference levels
MEASUrement:IMMed:REFLevel:ABSolute:HIGH	Sets or returns the high reference level
MEASUrement:IMMed:REFLevel:ABSolute:LOW	Sets or returns the low reference level
MEASUrement:IMMed:REFLevel:ABSolute:MID[1]	Sets or returns the middle reference level
MEASUrement:IMMed:REFLevel:ABSolute:MID2	Sets or returns the middle reference level for delay measurements
MEASUrement:IMMed:REFLevel:METhod	Sets or returns the method used to calculate measurements
MEASUrement:IMMed:REFLevel:PERCent:HIGH	Sets or returns the top level for rise time (90% level)
MEASUrement:IMMed:REFLevel:PERCent:LOW	Sets or returns the low level for rise time (10% level)
MEASUrement:IMMed:REFLevel:PERCent:MID[1]	Sets or returns the mid level for measurements
MEASUrement:IMMed:REFLevel:PERCent:MID2	Sets or returns the mid level for delay measurements
MEASUrement:IMMed:SOURCE[1]	Sets or returns the "from" source for all single channel immediate measurements
MEASUrement:IMMed:SOURCE2	Sets or returns the source to measure "to" for phase or delay immediate measurements
MEASUrement:IMMed:TYPE	Sets or returns the type of the immediate measurement
MEASUrement:IMMed:UNIts?	Returns the units of the immediate measurement
MEASUrement:IMMed:VALue?	Returns the value of the immediate measurement
MEASUrement:MEAS<n>:COUNt?	Returns the number of values accumulated since the last statistical reset
MEASUrement:MEAS<n>:MAXimum?	Returns the maximum value found since the last statistical reset
MEASUrement:MEAS<n>:MEAN?	Returns the mean value accumulated since the last statistical reset
MEASUrement:MEAS<n>:MINimum?	Returns the minimum value found since the last statistical reset
MEASUrement:MEAS<n>:STDdev?	Returns the standard deviation of values accumulated since the last statistical reset
MEASUrement:MEAS<x>?	Returns all measurement parameters for the displayed measurement
MEASUrement:MEAS<x>:DElay?	Returns the delay measurement parameters for specified measurement
MEASUrement:MEAS<x>:DElay:DIREction	Sets or returns the search direction to use for delay measurements
MEASUrement:MEAS<x>:DElay:EDGE[1]	Sets or returns the slope of the edge to use for delay "from" waveform measurements

MEASUrement:MEAS<x>:DELLay:EDGE2	Sets or returns the slope of the edge to use for delay "to" waveform measurements
MEASUrement:MEAS<x>:METHod	Sets or returns the method used to calculate the 0% and 100% reference level
MEASUrement:MEAS<x>:NOISe	Sets or returns the location of the noise measurement on the eye waveform for MEAS<x>
MEASUrement:MEAS<x>:REFLevel?	Returns the reference levels
MEASUrement:MEAS<x>:REFLevel:ABSolute:HIGH	Sets or returns the top level for rise time (90% level)
MEASUrement:MEAS<x>:REFLevel:ABSolute:LOW	Sets or returns the low level for rise time (10%)
MEASUrement:MEAS<x>:REFLevel:ABSolute:MID[1]	Sets or returns the mid level for measurements
MEASUrement:MEAS<x>:REFLevel:ABSolute:MID2	Sets or returns the mid level for delay measurements
MEASUrement:MEAS<x>:REFLevel:METHod	Sets or returns the HIGH and LOW levels to either % or units
MEASUrement:MEAS<x>:REFLevel:PERCent:HIGH	Sets or returns the top level for rise time (90% level)
MEASUrement:MEAS<x>:REFLevel:PERCent:LOW	Sets or returns the low level for rise time (10% level)
MEASUrement:MEAS<x>:REFLevel:PERCent:MID[1]	Sets or returns the mid level for measurements
MEASUrement:MEAS<x>:REFLevel:PERCent:MID2	Sets or returns the mid level for delay measurements
MEASUrement:MEAS<x>:SOURCE[1]	Sets or returns the channel from which measurements are taken
MEASUrement:MEAS<x>:SOURCE[2]	Sets or returns the source to measure "to" for phase or delay measurements
MEASUrement:MEAS<x>:STATE	Sets or returns whether the specified measurement slot is computed and displayed
MEASUrement:MEAS<x>:TYPE	Sets or returns the measurement<x> type
MEASUrement:MEAS<x>:UNIts?	Returns measurement<x> units
MEASUrement:MEAS<x>:VALue?	Returns the value of measurement<x>
MEASUrement:METHod	Sets or returns the method used for calculating reference levels
MEASUrement:NOISe	Sets or returns the location of the noise measurement on the eye waveform
MEASUrement:REFLevel:ABSolute:HIGH	Sets or returns the top reference level for risetime
MEASUrement:REFLevel:ABSolute:LOW	Sets or returns the low reference level for rise time
MEASUrement:REFLevel:ABSolute:MID	Sets or returns the mid reference level for measurements
MEASUrement:REFLevel:ABSolute:MID2	Sets or returns the mid reference level for delay "to" measurements
MEASUrement:REFLevel:METHod	Sets or returns the method for assigning high and low reference levels
MEASUrement:REFLevel:PERCent:HIGH	Sets or returns the top reference percent level for rise time

MEASUrement:REFLevel:PERCent:LOW	Sets or returns the low reference percent level for rise time
MEASUrement:REFLevel:PERCent:MID	Sets or returns the mid reference percent level for waveform measurements
MEASUrement:REFLevel:PERCent:MID2	Sets or returns the mid reference percent level for second waveform measurements
MEASUrement:STATIstics:COUNt	Clears existing measurement statistics from memory
MEASUrement:STATIstics:MODE	Turns management statistics on or off and sets or returns which pair of statistical accumulations is displayed
MEASUrement:STATIstics:WEIghing	Sets or returns the 'time constant' for mean and standard deviation statistical accumulations

Miscellaneous Command Group

Miscellaneous commands do not fit into other categories.

Several commands and queries are common to all 488.2–1987 devices on the GPIB bus. The 488.2–1987 standard defines these commands. The common commands begin with an asterisk (*) character.

Command	Description
*DDT	Sets or returns the commands to be executed by the group execute trigger
*IDN?	Returns the oscilloscope identification code
*LRN?	Returns a listing of oscilloscope settings
*TRG	Performs the group execute trigger (GET)
*TST?	Tests the GPIB interface and returns status
AUTOSet	Adjusts vertical, horizontal and trigger controls to provide a stable display of the selected waveform. The UNDO form attempts to return the instrument to a pre-Autoset state
AUXOut?	Returns the auxiliary out setup
AUXOut:EDGE	Sets or returns the polarity of the trigger output signal
AUXOut:SOURce	Sets or returns the trigger source at the BNC connection
BELI	In previous TDS models, beeped the audio indicator. Provided for backward compatibility.
CMDBatch	Turns command batching on or off or returns the state of command batching
DATE	Sets or returns the date that the oscilloscope can display
HDR	Sets or returns the Response Header Enable State
HEADer	Sets or returns the Response Header Enable State
ID?	Returns identifying information about the instrument and its firmware
LOCK	Sets or returns the front panel lock state
NEWpass	Changes the password for user protected data
PASSWord	Provides access for changing user protected data
REM	Specifies a comment, which is ignored by the oscilloscope
ROSc:SOURce	Sets or returns the source for the 10 MHz timebase
ROSc:STate?	Returns the state of the 10 MHz timebase
SET?	Returns a listing of oscilloscope settings
TEKSecure	Initializes both waveform and setup memories
TIME	Sets or returns the time displayed by the oscilloscope
UNLock	Unlocks the front panel
VERBose	Sets or returns the verbose state

Save and Recall Command Group

You can use the commands in the Save and Recall Command Group to store and retrieve internal waveforms and settings. When you save a setup, you save all the settings of the oscilloscope. When you recall a setup, the oscilloscope restores itself to the state that it was in when you originally saved that setting.

Reference File Format

The Tektronix wfm file format was designed for the internal save and restore of waveform data and the associated display. As a consequence, more parameters are saved than are required to transfer and create an external Time vs. Voltage value list. The format will handle XY waveform data, XYZ data, and pixel maps, such as the waveform database. FastFrame waveforms, multiple acquisitions with individual timestamps, can also be represented.

Note: The Tektronix wfm file format does not allow you to create a waveform and reintroduce it back into the instrument. To export waveforms for use outside the instrument, use the Export and Waveform Transfer commands.

Command	Description
*RCL	Restores the state of the oscilloscope from a copy of the setting stored in memory.
*SAV	Stores the current oscilloscope state to a specified memory location
*SDS	Changes the specified setup to reference the factory setup instead of the user setup
DELEte:SETUp	Removes stored setups from memory and initializes the location with the factory default setups
DELEte:WAVEform	Deletes (one or all of) the stored reference waveforms from memory
FACtory	Resets the oscilloscope to factory default settings
RECALL:SETUp	Recalls saved instrument settings
RECALL:WAVEform	Recalls a stored waveform to a reference location
SAVE:SETUp	Saves the current front-panel setup to a specified memory location or file
SAVE:WAVEform	Saves a waveform to one of four reference memory locations or a mass storage file
SAVE:WAVEform:FILEFormat	Sets or returns the format used for saved waveforms
SETUp:NAME	Sets or returns the user-defined setup label

Status and Error Command Group

You use the commands in the Status and Error command Group to determine the status of the oscilloscope and control events.

Several commands and queries used with the oscilloscope are common to all devices on the GPIB bus. The IEEE Std 488.2–1987 defines these commands and queries. The common commands begin with an asterisk (*) character.

Command	Description
*CLS	Clears status
*ESE	Sets or returns the bits in the Event Status Enable Register
*ESR?	Returns the contents of the Standard Event Status Register
*OPC?	Returns "1" when all current operations are finished
*OPC	Generates the operation complete message in the standard event status register when all pending operations are finished
*OPT?	Returns a list of options installed in the oscilloscope
*PSC	Sets or returns the power on status flag state
*PUD	Sets or returns a string of protected user data
*RST	Resets the oscilloscope to factory default settings
*SRE	Sets or returns the bits in the Service Request Enable Register
*STB?	Returns the contents of the Status Byte Register
*WAI	Prevents the oscilloscope from executing further commands until all pending operations finish
ALLEv?	Returns all events and their messages
BUSY?	Returns oscilloscope status
DESE	Sets or returns the bits in the Device Event Status Enable Register
EVENT?	Returns event code from the event queue
EVMsg?	Returns event code, message from the event queue
EVQty?	Return number of events in the event queue

Trigger Command Group

You use the commands in the Trigger Command Group to control all aspects of triggering for the oscilloscope.

There are two triggers: A and B. Where appropriate, the command set has parallel constructions for each trigger.

You can set the A or B triggers to edge mode. Edge triggering lets you display a waveform at or near the point where the signal passes through a voltage level of your choosing.

You can also set A triggers to pulse and logic modes. Pulse triggering lets the oscilloscope trigger whenever it detects a pulse of a certain width or height. Logic triggering lets you logically combine the signals on one or more channels. The oscilloscope then triggers when it detects a certain combination of signal levels.

Communications Trigger commands work with TDS7000 Series Oscilloscopes with option SM and all CSA7000 Series Communications Signal Analyzers.

Serial Trigger commands work with TDS7000 Series Oscilloscopes with option ST and all CSA7000 Series Communications Signal Analyzers.

Command	Description
TRIGger?	Returns current trigger parameters for the oscilloscope
TRIGger	Forces a trigger event to occur
TRIGger:A	Sets A trigger level to 50% or returns current A trigger parameters
TRIGger:A:COMMunication:AMI. . . PULSEForm	Sets the AMI pulse form
TRIGger:A:COMMunication:AMI. . . THResh-old:HIGH	Sets the AMI threshold high level
TRIGger:A:COMMunication:AMI. . . THResh-old:LOW	Sets the AMI threshold low level
TRIGger:A:COMMunication:BITRate	Sets and returns the bit rate
TRIGger:A:COMMunication:CMI:PULSEForm	Selects the CMI pulse form
TRIGger:A:COMMunication:CLOCK:POLarity	Sets or returns the polarity of the clock signal
TRIGger:A:COMMunication:CODE	Sets the signal code
TRIGger:A:COMMunication:SOURce	Sets the source channel
TRIGger:A:COMMunication:SOURce:TYPE	Sets the type of source channel
TRIGger:A:COMMunication:STANdard	Selects the standard that identifies the code and bit rate
TRIGger:A:EDGE?	Returns the source, coupling and source for the A edge trigger
TRIGger:A:EDGE:COUPling	Sets or returns the type of coupling for the A edge trigger

TRIGger:A:EDGE:SLOpe	Sets or returns the slope for the A edge trigger
TRIGger:A:EDGE:SOUrce	Sets or returns the source for the A edge trigger
TRIGger:A:HOLDoff?	Returns the A trigger holdoff parameters
TRIGger:A:HOLDoff:ACTUal?	Returns the holdoff time actually used by the A trigger
TRIGger:A:HOLDoff:BY?	Returns the type of holdoff for the A trigger
TRIGger:A:HOLDoff:BY	Sets the type of holdoff for the A trigger
TRIGger:A:HOLDoff:TIME	Sets or returns the A trigger holdoff time
TRIGger:A:LEVel	Sets or returns the level for the A trigger
TRIGger:A:LEVel:CH<x>	Sets or returns the A trigger level for TRIGger:LVL:Srcpreference
TRIGger:A:LOGic?	Returns all of the A trigger logic parameters
TRIGger:A:LOGic:CLAss	Sets or returns the type of A trigger logic
TRIGger:A:LOGic:FUNCTion	Sets or returns the logical combination of the input channels for the A logic trigger
TRIGger:A:LOGic:INPut?	Returns the A logic trigger input for channels 1-3.
TRIGger:A:LOGic:INPut:CH<x>	Specifies or returns the input settings for the specified logic trigger channel
TRIGger:A:LOGic:PATtern?	Returns the conditions for generating an A logic pattern trigger
TRIGger:A:LOGic:PATtern:INPut:CH4	Sets or returns the A logic trigger input for channel 4
TRIGger:A:LOGic:PATtern:WHEN	Sets or returns the condition for generating the A logic pattern trigger
TRIGger:A:LOGic:PATtern:WHEN:LEssLimit	Sets or returns the maximum time that the selected pattern may be true and still generate an A logic pattern trigger
TRIGger:A:LOGic:PATtern:WHEN:MORELimit	Sets or returns the minimum time that the selected pattern may be true and still generate an A logic pattern trigger
TRIGger:A:LOGic:SETHold?	Returns clock edge polarity, voltage threshold and source; data voltage threshold and source; and setup/hold times for violation triggering
TRIGger:A:LOGic:SETHold:CLOCK?	Returns clock edge polarity, voltage threshold and source input for setup/hold triggering
TRIGger:A:LOGic:SETHold:CLOCK:EDGE	Sets or returns the clock edge polarity for setup and hold triggering

TRIGger:A:LOGic:SETHold:CLOCK:SOUrce	Sets or returns the clock source for the A logic trigger setup and hold input
TRIGger:A:LOGic:SETHold:CLOCK:THReshold	Sets or returns the clock voltage threshold for setup and hold trigger
TRIGger:A:LOGic:SETHold:DATA?	Returns the voltage threshold and data source for the setup/hold trigger
TRIGger:A:LOGic:SETHold:DATA:SOUrce	Sets or returns the data source for the setup and hold trigger
TRIGger:A:LOGic:SETHold:DATA:THReshold	Sets or returns the data voltage threshold for setup and hold trigger
TRIGger:A:LOGic:SETHold:HOLDTime	Sets or returns the hold time for the setup and hold violation triggering
TRIGger:A:LOGic:SETHold:QUALify	Sets or returns the Setup and Hold trigger qualifications
TRIGger:A:LOGic:SETHold:SETTime	Sets or returns the setup time for setup and hold violation triggering
TRIGger:A:LOGic:STATE?	Returns the data input and trigger criteria for the A logic trigger
TRIGger:A:LOGic:STATE:INPut:CH4	Sets or returns the slope for channel 4 of the A logic state trigger
TRIGger:A:LOGic:STATE:WHEn	Sets or returns the condition for generating an A logic state trigger
TRIGger:A:LOGic:THReshold?	Returns the threshold voltage for all channels in A logic trigger
TRIGger:A:LOGic:THReshold:CH<x>	Sets or returns the A logic trigger threshold voltage for the specified channel
TRIGger:A:LOWerthreshold:CH<x>	Sets or returns CH<x> lower trigger level for TRIGger:LVLsrcpreference
TRIGger:A:MODE	Sets or returns the A trigger mode
TRIGger:A:PULse?	Returns the A pulse trigger parameters
TRIGger:A:PULse:CLAss	Sets or returns the type of pulse on which to trigger
TRIGger:A:PULse:GLItch?	Returns the current A glitch pulse trigger parameters
TRIGger:A:PULse:GLItch:POLarity	Sets or returns the polarity for the A pulse glitch trigger
TRIGger:A:PULse:GLItch:QUALify	Sets or returns the glitch trigger qualifications
TRIGger:A:PULse:GLItch:TRIGIF	Sets or returns the acceptance or rejection of pulse glitch trigger, based on width.
TRIGger:A:PULse:GLItch:WIDth	Sets or returns the width of the A pulse glitch trigger
TRIGger:A:PULse:RUNT?	Returns the current A runt pulse trigger parameters

TRIGger:A:PULse:RUNT:POLarity	Sets or returns the polarity for the A pulse runt trigger
TRIGger:A:PULse:RUNT:QUALify	Sets or returns the runt trigger qualifications
TRIGger:A:PULse:RUNT:THReshold?	Returns the upper and lower thresholds for the A pulse runt trigger
TRIGger:A:PULse:RUNT:THReshold:BOTH	Sets the upper and lower switching thresholds for the A pulse runt trigger
TRIGger:A:PULse:RUNT:THReshold:HIGH	Sets or returns the upper limit for the A pulse runt trigger
TRIGger:A:PULse:RUNT:THReshold:LOW	Sets or returns the lower limit for the A pulse runt trigger
TRIGger:A:PULse:RUNT:WHEn	Sets or returns the type of pulse width the trigger checks for when it uncovers a runt
TRIGger:A:PULse:RUNT:WIDth	Sets or returns the minimum width for A pulse runt trigger
TRIGger:A:PULse:SOURce	Sets or returns the source for the A pulse trigger
TRIGger:A:PULse:TIMEOut?	Returns the parameters for the A pulse timeout trigger
TRIGger:A:PULse:TIMEOut:POLarity	Sets or returns the polarity for the A pulse timeout trigger
TRIGger:A:PULse:TIMEOut:QUALify	Sets or returns the Timeout trigger qualifications
TRIGger:A:PULse:TIMEOut:TIME	Sets or returns the pulse timeout trigger time
TRIGger:A:PULse:TRANSition?	Returns the delta time, polarity, and both upper and lower threshold limits for the transition time trigger
TRIGger:A:PULse:TRANSition:DeltaTime	Sets or returns the delta time used in calculating the transition value
TRIGger:A:PULse:TRANSition:POLarity	Sets or returns the polarity for the A pulse transition trigger
TRIGger:A:PULse:TRANSition:QUALify	Sets or returns the transition trigger qualifications
TRIGger:A:PULse:TRANSition:THReshold:BOTH	Sets or returns the upper and lower thresholds for the pulse transition trigger
TRIGger:A:PULse:TRANSition:THReshold:HIGH	Sets or returns the upper transition trigger threshold
TRIGger:A:PULse:TRANSition:THReshold:LOW	Sets or returns the lower transition trigger threshold
TRIGger:A:PULse:TRANSition:WHEn	Sets or returns the relationship of delta time to transitioning signal
TRIGger:A:PULse:WIDth?	Returns the trigger A pulse width parameters
TRIGger:A:PULse:WIDth:HIGHLimit	Sets or returns the upper limit for the A pulse width trigger
TRIGger:A:PULse:WIDth:LOWLIMIT	Sets or returns the lower limit for the A pulse width trigger
TRIGger:A:PULse:WIDth:POLarity	Sets or returns the polarity for the A pulse width trigger

TRIGger:A:PULse:WIDth:QUALify	Sets or returns the width trigger qualifications
TRIGger:A:PULse:WIDth:WHEn	Sets or returns the criteria for width specification of pulse width trigger events
TRIGger:A:PULse:WINDow:QUALify	Sets or returns the window trigger qualifications
TRIGger:A:PULse:WINDow:THReshold:BOth	Sets both window thresholds
TRIGger:A:PULse:WINDow:THReshold:HIGH	Sets or returns the high window threshold
TRIGger:A:PULse:WINDow:THReshold:LOW	Sets or returns the low window threshold
TRIGger:A:PULse:WINDow:TYPE	Sets or returns the type of window trigger
TRIGger:A:PULse:WINDow:WHEn	Sets or returns the condition for use of window violation
TRIGger:A:PULse:WINDow:WIDth	Sets or returns the minimum width of a window violation
TRIGger:A:SERial:BITRate	Set the clock/data bit rate
TRIGger:A:SERial:CLOCK:LEVel	Sets the serial clock level
TRIGger:A:SERial:CLOCK:POLARITY	Sets the serial clock polarity
TRIGger:A:SERial:CLOCK:SOUrce	Select serial clock source channel
TRIGger:A:SERial:CODe	Select the signal code
TRIGger:A:SERial:DAta:FORMat	Sets the Pattern string format
TRIGger:A:SERial:DAta:PATtern	Allow up to 32 bit serial pattern
TRIGger:A:SERial:SOUrce	Select serial data source channel
TRIGger:A:SERial:STANdard	Select the standard that identifies the code and bit rate
TRIGger:A:TYPE	Sets or returns the type of A trigger
TRIGger:A:UPPPerthreshold:CH<x>	Sets or returns CH<x> upper trigger level for TRIGger:LVLScPreference
TRIGger:AUXLeVel	Sets or returns the auxiliary level for edge trigger
TRIGger:B	Sets the B trigger level to 50% or returns the B trigger parameters
TRIGger:B:BY	Sets or returns B trigger time or event qualifiers
TRIGger:B:EDGE?	Returns B trigger edge type parameters
TRIGger:B:EDGE:COUPling	Sets or returns the type of B trigger coupling
TRIGger:B:EDGE:SLOpe	Sets or returns the B edge trigger slope
TRIGger:B:EDGE:SOUrce	Sets or returns the B edge trigger source
TRIGger:B:EVENTS?	Returns the current B trigger events parameter
TRIGger:B:EVENTS:COUNt	Sets or returns the number of events that must occur before the B trigger occurs

TRIGger:B:LEVel	Sets or returns the level for the B trigger
TRIGger:B:LEVel:CH<x>	Sets or returns the B trigger level for TRIGger:LVLSrcpreference
TRIGger:B:RESET:SOUrce	Sets or returns the source for the trigger reset
TRIGger:B:RESET:STATE	Sets or returns the type of state for a state trigger reset
TRIGger:B:RESET:TIMEOut	Sets or returns the reset time for the timeout trigger reset
TRIGger:B:RESET:THREshold	Sets or returns the threshold for the trigger reset
TRIGger:B:RESET:TRANSition	Sets or returns the type of transition required for a transition trigger reset
TRIGger:B:RESET:TYPe	Sets or returns the type of A→B sequential trigger reset
TRIGger:B:STATE	Sets or returns the state of the B trigger
TRIGger:B:TIME	Sets or returns the B trigger delay time
TRIGger:B:TYPe	Sets or returns the type of B trigger
TRIGger:LVLSrcpreference	Sets or returns how the trigger source and levels interact
TRIGger:STATE?	Returns the current state of the triggering system

Vertical Command Group

You use the commands in the Vertical Command Group to control the vertical setup of all live (channel) waveforms for acquisition and to control the display of channel, reference, and math waveforms. This group contains commands to set up vertical parameters for electrical or optical signals, depending on the sampling module in use.

The `SElect:<wfm>` command also selects the waveform that many commands in other command groups use.

You may replace `VOLts` with `SCALE` in the vertical commands. This provides program compatibility with earlier models of Tektronix oscilloscopes.

Command	Description
<code>CH<x>?</code>	Returns vertical parameters for the specified channel
<code>CH<x>:BANdwidth</code>	Sets or returns the bandwidth of the specified channel
<code>CH<x>:COUPling</code>	Sets or returns the coupling for the specified channel
<code>CH<x>:DESKew</code>	Sets or returns the deskew time for the specified channel
<code>CH<x>:LABEL:NAME</code>	Defines or returns the label for the channel waveform
<code>CH<x>:LABEL:XPOS</code>	Sets or returns the X display coordinate for the channel waveform label
<code>CH<x>:LABEL:YPOS</code>	Sets or queries the Y screen offset at which the label (attached to the displayed waveform of the specified channel) is displayed
<code>CH<x>:OFFSet</code>	Sets or returns the channel offset
<code>CH<x>:POSition</code>	Sets or returns the channel vertical position
<code>CH<x>:PRObe?</code>	Returns the gain, resistance, units, and ID of the probe that is attached to the specified channel
<code>CH<x>:PROBECal?</code>	Returns the probe calibration status
<code>CH<x>:PRObe:GAIN?</code>	Returns the gain of the probe that is attached to the specified channel
<code>CH<x>:PRObe:ID?</code>	Returns the type and serial number of the probe that is attached to the specified channel
<code>CH<x>:PRObe:ID:TYPE</code>	Returns the type of probe that is attached to the specified channel
<code>CH<x>:PRObe:ID:SERnumber?</code>	Returns the serial number of the probe that is attached to the specified channel
<code>CH<x>:PRObe:RESistance?</code>	Returns the resistance of the probe that is attached to the specified channel
<code>CH<x>:PRObe:SET</code>	Sets or returns the P7260 probe attenuation
<code>CH<x>:PRObe:UNITS?</code>	Returns the units of the probe that is attached to the specified channel

CH<x>:PROBEFunc:EXTatten	Sets or returns the attenuation value for the specified channel to the specified scale factor
CH<x>:PROBEFunc:EXTDBatten	Sets or returns the attenuation value for the specified channel to the specified value, in decibels
CH<x>:PROBEFunc:EXTUnits	Sets or returns the unit of measurement for the external attenuator of the specified channel
CH<x>:SCALE	Sets or returns the vertical scale (per div) of the specified channel
CH<x>:TERmination	Sets or returns the channel input termination
MATH<x>:POSition	Sets or returns the vertical position for math display
MATH<x>:SCALE	Sets or returns the vertical scale (per div) for math display
REF<x>:HORizontal:POSition	Sets or returns the horizontal position of the specified reference waveform
REF<x>:LABel?	Returns a branch query containing the waveform label name and the coordinates at which the label is displayed
REF<x>:LABel:NAME	Sets or returns the label of the designated waveform
REF<x>:LABel:XPOS	Sets or returns the position of the reference waveform label on the X axis
REF<x>:LABel:YPOS	Sets or queries the Y screen offset at which the label (attached to the displayed waveform of the specified reference) is displayed
REF<x>:VERTical:POSition	Sets or returns the vertical position of the specified reference waveform
REF<x>:VERTical:SCALE	Sets or returns the vertical scale of the specified reference waveform
SElect?	Return information on which waveforms are ON/OFF and which is the selected waveform.
SElect:<wfm>?	Returns whether the specified channel is on or off
SElect<wfm>	Turns on the specified waveform
SElect:CONTRol?	Returns the waveform and time base selected for front-panel control.
SElect:CONTRol <wfm>	Sets the waveform controlled by front panel.
WLEnGth:VALue	Sets or returns the wavelength calibration standard

Waveform Transfer Command Group

You use the commands in the Waveform transfer Command Group to transfer waveform data points to and from the instrument. Waveform data points are a collection of values that define a waveform. One data value usually represents one data point in the waveform record. When working with envelope waveforms, each data value is either the minimum or maximum of a min/max pair. Before you transfer waveform data, you must specify the data format, record length, and waveform source.

Data Formats

Acquired waveform data uses eight or more bits to represent each data point. The number of bits used depends on the acquisition mode specified when you acquired the data. Data acquired in SAMple or ENVELOpe mode uses eight bits per waveform data point. Data acquired in AVERAGE mode uses up to 14 bits per point.

The oscilloscope can transfer waveform data in either ASCII or binary format. You specify the format with the DATA:ENCdg command. The instrument uses signed, 4 byte integers and floating point values; it does not support unsigned floating point values.

ASCII data is represented by signed integer or floating point values. An example ASCII waveform data string may look like this:

```
CURVE<space>-110,-109,-110,-110,-109,-107,  
-109,-107,-106,-105,-103,-100,-97,-90,-84,  
-80
```

Use ASCII to obtain more readable and easier to format output than binary. However, ASCII may require more bytes to send the same values than it does with binary. This may reduce transmission speeds.

Binary data can be represented by signed integer or floating point values. The range of the values depends on the byte width specified. When the byte width is one, signed integer data ranges from -128 to 127, and positive integer values range from 0 to 255. When the byte width is two, the values range from -32768 to 32767. When a MATH (or REF that came with a MATH) is involved, 32 bit floating point values are used that are four bytes in width.

The defined binary formats specify the order in which the bytes are transferred. The following are the four binary formats:

-
- RIBinary specifies signed integer data-point representation with the most significant byte transferred first.
 - SRIBinary is the same as RIBinary except that the byte order is swapped, meaning that the least significant byte is transferred first. This format is useful when transferring data to IBM-compatible PCs
 - RFBinary specifies floating point data-point representation with the most significant byte transferred first.
 - SRFBinary is the same as RFBinary except that the byte order is swapped, meaning that the least significant byte is transferred first. This format is useful when transferring data to IBM-compatible PCs.

Waveform Data and Record Lengths

You can transfer multiple points for each waveform record. You can transfer a portion of the waveform or you can transfer the entire record. You can use the DATA:START and DATA:STOP commands to specify the first and last data points of the waveform record.

When transferring data into the instrument, you must first specify the record length of the destination waveform record. You do this with the WFMInopre:NR_Pt command. Next, specify the first data point within the waveform record. For example, when you set DATA:START to 1, data points will be stored starting with the first point in the record. The instrument will ignore the value set by DATA:STOP when reading in data. It will stop reading data when there is no more data to read or when it has reached the specified record length.

When transferring data from the instrument, you must specify the first and last data points in the waveform record. Setting DATA:START to 1 and DATA:STOP to the record length will always return the entire waveform.

Waveform Data Locations and Memory Allocation

The DATA:SOURce command specifies the waveform source when transferring a waveform from the oscilloscope. You can only transfer one waveform at a time. Waveforms sent to the instrument are always stored in one of the four reference memory locations. You use the DATA:DESTination command to specify a reference memory location.

Waveform Preamble

Each waveform that you transfer has an associated waveform preamble that contains information such as the hori-

zontal scale, the vertical scale, and other settings in effect when the waveform was created. Refer to the individual WFMInpre and WFMOupre commands for more information.

Scaling Waveform Data

Once you transfer the waveform data to the controller, you can convert the data points into voltage values for analysis using information from the waveform preamble.

Transferring Waveform Data from the Oscilloscope

You can transfer waveforms from the oscilloscope to an external controller using the following sequence:

1. Select the waveform source(s) using DATa:SOURce.
2. Specify the waveform data format using DATa:ENCdg.
3. Specify the number of bytes per data point using WFMOupre:BYT_Nr.

Note: MATH waveforms (and REF waveforms that came from a MATH) are always set to four bytes.

4. Specify the portion of the waveform that you want to transfer using DATa:STARt and DATa:STOP.
5. Transfer waveform preamble information using WFMOupre?.
6. Transfer waveform data from the oscilloscope using CURVe?.

Transferring Waveform Data to the Oscilloscope

1. Specify waveform reference memory using DATa:DES-Tination.
2. Set WFMInpre:NR_Pt to equal the number of data points to be sent.
3. Specify the waveform data format using WFMInpre:ENCdg.
4. Specify the number of bytes per data point using WFMInpre:BYT_Nr.
5. Specify first data point in the waveform record using DATa:STARt.
6. Transfer waveform preamble information using WFMInpre.

Note: Neither FastAcq nor WFMDB waveforms can be returned to the scope via the CURVe command. Refer to Reference File Format for additional information.

Command	Description
CURve?	Transfers waveform data from instrument specified by the DATA:SOURce command
CURVe	Transfer waveform data to the instrument (reference memory location specified by DATA:DESTination)
CURVEStream	Sets or returns continuous transfer of waveform data from the instrument
DATA	Sets or returns the format and location of the waveform data that is transferred with the CURVe? command
DATA:DESTination	Sets or returns the reference memory location for storing waveform data sent to the oscilloscope
DATA:ENCdg	Sets or returns the format of outgoing waveform data
DATA:FRAMESTART	Sets or returns the starting data point for waveform transfer
DATA:FRAMESTOP	Sets or returns the last data frame that will be transferred when using the CURVe? query
DATA:SOURce	Sets or returns the location of waveform data transferred from the instrument
DATA:START	Sets or returns the starting point in waveform transfer
DATA:STOP	Sets or returns the ending data point in waveform transfer
WAVFrm?	Returns a branch query containing waveform data in either binary or ASCII format, waveform formatting data, and the location of the waveform data source
WAVFRMStream?	Returns WFMOutpre? and CURVEStream? data for DATA:SOURce waveforms
WFMInpre?	Returns the waveform formatting specification to be applied to the next incoming CURVE command data
WFMInpre:BIT_Nr	Sets or returns the number of bits per binary waveform point
WFMInpre:BN_Fmt	Sets or returns the format of binary data for the incoming waveform
WFMInpre:BYT_Nr	Sets or returns the binary field data width for the first ordered waveform
WFMInpre:BYT_Or	Sets or returns the byte order of waveform points
WFMInpre:ENCdg	Sets or returns the type of encoding for incoming waveform data
WFMInpre:NR_FR?	Returns one frame, indicating the number of frames in the transmitted waveform record
WFMInpre:NR_Pt	Sets or returns the number of points in the transmitted waveform record
WFMInpre:PT_Fmt	Sets or returns the point format of incoming waveform data
WFMInpre:PT_OFF	Sets or returns the trigger point within the incoming waveform data
WFMInpre:WFID	Provided to allow a waveform extracted from the oscilloscope to be more easily sent back

WFMInpre:XINcr	Sets or returns the horizontal sampling interval between incoming waveform points
WFMInpre:XUNit	Sets or returns the horizontal units of the incoming waveform
WFMInpre:XZEr0	Sets or returns the (sub-sample) time between the trigger sample and the occurrence of the actual incoming waveform trigger
WFMInpre:YMUlt	Sets or returns the vertical scale factor, per digitizing level, of the incoming waveform points
WFMInpre:YOFF	Sets or returns the vertical position of the incoming waveform in digitizing levels
WFMInpre:YUNit	Sets or returns the vertical units of the incoming waveform
WFMInpre:YZEr0	Sets or returns the offset of the incoming waveform
WFMOutpre?	Returns the waveform formatting data for the waveform specified by the DATA:SOURCE command
WFMOutpre:BIT_Nr	Sets or returns the number of bits per waveform point that outgoing waveforms contain
WFMOutpre:BN_Fmt	Sets or returns the format of binary data for the waveform
WFMOutpre:BYT_Nr	Sets or returns the data width for the waveform
WFMOutpre:BYT_Or	Sets or returns the byte order of waveform points
WFMOutpre:ENCdg	Sets or returns the type of encoding for outgoing waveforms
WFMOutpre:NR_FR?	Returns the number of frames for the DATA:SOURce waveform transmitted in response to a CURVe? query
WFMOutpre:NR_Pt?	Returns the number of points for the waveform transmitted in response to a CURVe? query
WFMOutpre:PT_Fmt?	Returns the point format for the waveform
WFMOutpre:PT_OFF?	Returns the trigger point relative to DATA:START for the waveform
WFMOutpre:PT_OR?	Returns the point ordering for the reference waveform
WFMOutpre:WFId?	Returns a string describing the acquisition parameters for the waveform
WFMOutpre:XINcr?	Returns the horizontal sampling interval
WFMOutpre:XUNit?	Returns the horizontal units for the waveform
WFMOutpre:XZEr0?	Returns the (sub-sample) time between the trigger sample and the occurrence of the actual waveform trigger
WFMOutpre:YMUlt?	Returns the vertical scale factor per digitizing level for the waveform
WFMOutpre:YOFF?	Returns the vertical offset in digitizing levels for the waveform
WFMOutpre:YUNit?	Returns the vertical units for the waveform
WFMOutpre:YZEr0?	Returns the vertical offset for the waveform

Zoom Command Group

Zoom commands let you expand and position the waveform display horizontally and vertically without changing the time base or vertical settings.

Command	Description
ZOOm	Sets the zoom transforms to default values for either live traces or all traces Or returns the current vertical and horizontal positioning and scaling of the display
ZOOm:GRAticule:SIZE	Sets or returns the size of the zoom graticule
ZOOm:MODE	Sets or returns the zoom mode
ZOOm:ZOOM<x>	Sets or returns zoom factors for the specified zoom
ZOOm:ZOOM<x>:STATE	Specifies or returns a trace as zoomed, on or off
ZOOm:ZOOM<x>:SCROLLLock	Sets or returns Scroll Lock for the specified zoom
ZOOm:ZOOM<x>:<wfm>:HORIzonal:POSition	Sets or returns the horizontal zoom position for the specified waveform in the specified zoom
ZOOm:ZOOM<x>:<wfm>:HORIzonal:SCAle	Sets or returns the horizontal zoom scale of the specified waveform in the specified zoom
ZOOm:ZOOM<x>:<wfm>:VERTi-cal:POSition	Sets or returns the vertical zoom position for the specified waveform in the specified zoom
ZOOm:ZOOM<x>:<wfm>:VERTi-cal:SCAle	Sets or returns the vertical zoom scale for the specified waveform in the specified zoom
ZOOM:GRAticule:SPLIT	Sets or returns the sizes of the acquisition and zoom windows
ZOOM:SCROLL:DIREction	Sets or returns the direction of AutoScroll for zoomed waveforms
ZOOM:SCROLL:LOCK	Sets or returns the state of Scroll Lock, on or off
ZOOM:SCROLL:SPEED	Sets or returns the speed of AutoScroll for zoomed waveforms

Commands Listed in Alphabetical Order

*CAL?

Description

This query-only command instructs the digitizing oscilloscope to perform a Signal Path Compensation (SPC) and return its calibration status. This command is equivalent to selecting Instrument Calibration from the Utilities menu and then clicking the Calibrate button. The SPC will not run until the instrument has warmed up for at least 20 minutes.

Note: This self-calibration can take several minutes to respond. No other commands will be executed until calibration is complete.

Group

Calibration

Related Commands

CALibrate:RESults:SPC?

Syntax

*CAL?

Returns

- -1

Indicating the calibration failed.

- 0

Indicating the calibration passed.

- 3

Indicating the 20-minute warm up period has not completed and the calibration is terminated.

Example

*CAL?

This query performs an internal self-calibration that may return

0

Indicating the SPC calibration was successful.

*CLS

Description

This command (no query form) clears the following:

- Event Queue
- Standard Event Status Register
- Status Byte Register (except the MAV bit)

If the *CLS command immediately follows an <EOI>, the Output Queue and MAV bit (Status Byte Register bit 4) are also cleared. MAV indicates that information is in the output queue. The device clear (DCL) GPIB control message will clear the output queue and thus MAV. *CLS does not clear the output queue or MAV.

*CLS can suppress a Service Request that is to be generated by an *OPC. This will happen if a single sequence acquisition operation is still being processed when the *CLS command is executed.

Group

Status and Error

Related Commands

DESE, *ESE, *ESR?, EVENT?, EVMsg?, *SRE, *STB?

Syntax

*CLS

Example

*CLS

This command clears the oscilloscope status data structures.

*DDT

Description

This command allows you to specify a command or a list of commands that are executed when the oscilloscope receives a *TRG command or the GET GPIB interface message. Define Device Trigger (*DDT) is a special alias that the *TRG command uses.

Group

Miscellaneous

Related Commands

ALias, *TRG

Syntax 1

```
*DDT {<Block>|<Qstring>}
```

Syntax 2

```
*DDT?
```

Arguments

■ <Block>

This is a complete sequence of program messages. The messages can contain only valid commands that must be separated by semicolons and must follow all rules for concatenating commands. The sequence must be less than or equal to 80 characters. The format of this argument is always returned as a query.

■ <QString>

This is a complete sequence of program messages. The messages can contain only valid commands that must be separated by semicolons and must follow all rules for concatenating commands. The sequence must be less than or equal to 80 characters.

Example

```
*DDT #OACQUIRE:STATE RUN
```

This command specifies that the acquisition system will be started each time a *TRG command is sent.

*ESE

Description

This command sets and queries the bits in the Event Status Enable Register (ESER). The ESER prevents events from being reported to the Status Byte Register (STB). For a more detailed discussion of the use of these registers, see Registers.

Group

Status and Error

Related Commands

*CLS, DESE, *ESR?, EVENT?, EVMsg?, *SRE, *STB?

Syntax 1

```
*ESE <NR1>
```

Syntax 2

```
*ESE?
```

Argument

■ <NR1>

This specifies the binary bits of the ESER according to this value, which ranges from 0 through 255.

The power-on default for ESER is 0 if *PSC is 1. If *PSC is 0, the ESER maintains the previous power cycle value through the current power cycle.

Note: Setting the DESER and the ESER to the same values allows only those codes to be entered into the Event Queue and summarized on the ESB bit (bit 5) of the Status Byte Register. Use the DESE command to set the DESER.

Example 1

```
*ESE 209
```

This command sets the ESER to binary 11010001, which enables the PON, URQ, EXE and OPC bits.

Example 2

*ESE?

This query form of this command returns

186

showing that the ESER contains the binary value
10111010.

***ESR?**

Description

This query-only command returns the contents of the Standard Event Status Register (SESR). *ESR? also clears the SESR (since reading the SESR clears it). For a more detailed discussion of the use of these registers, see Registers.

Group

Status and Error

Related Commands

ALLEv?, *CLS, DESE, *ESE, EVENT?, EVMsg?, *SRE, *STB?

Syntax

*ESR?

Example

*ESR?

This query might return

213

showing that the SESR contains the binary value 11010101.

*IDN?

Description

This query-only command returns the oscilloscope identification code.

Group

Miscellaneous

Related Commands

ID?

Syntax

IDN?

Example

*IDN?

This query might return

```
:TEKTRONIX,TDS7404,0,CF:91.1CT  
FV:01.00.912
```

indicating the oscilloscope model number, configured number, and firmware version number.

*LRN?

Description

This query-only command returns the commands that list the oscilloscope settings (except for configuration information for the calibration values), allowing you to record or "learn" the current oscilloscope settings. You can use these commands to return the oscilloscope to the state it was in when you made the *LRN? query.

Group

Miscellaneous

Related Commands

SET?

Syntax

*LRN?

Example

*LRN?

This query might return the following response:

```
:ACQUIRE:STOPAFTER RUNSTOP;STATE 1;MODE
SAMPLE;NUMENV
10;NUMAVG 16;REPET 1;:FASTACQ:STATE 1;:AP-
PLICATION
:GPKNOB1:ACTIVE 0;:APPLICATION:GPKNOB2:AC-
TIVE 0;
:APPLICATION:WINDOW:HEIGHT 0;WIDTH 0;:AP-
PLICATION
:SCOPEAPP:STATE NOTRUNNING;WINDOW FULL-
SCREEN;
:APPLICATION:EXTAPP:STATE NOTRUNNING;:AUX-
OUT:SOURCE
ATRIGGER;EDGE FALLING;:CMDBATCH 1;:HEADER
0;:LOCK
NONE;:VERBOSE 1;:ALIAS:STATE 0;:DIS-
PLAY:CLOCK 1
;COLOR:PALETTE NORMAL;MATHCOLOR DE-
FAULT;REFCOLOR
DEFAULT;:DISPLAY:FILTER SINX;FORMAT
YT;GRATICULE
FULL;INTENSITY:WAVEFORM 60.0000;AUTOBRIGHT
1
;SCRENSAVER 1;SCRENSAVERDELAY
28800;:DISPLAY
```

```

:PERSISTENCE OFF;STYLE VECTORS;TRIGBAR
SHORT;TRIGT 1
;VARPERSIST 500.0000E-3;:HARDCOPY:FILENAME
"";FORMAT
BMP;LAYOUT PORTRAIT;PALETTE HARDCOPY;PORT
FILE;
:HORIZONTAL:DELAY:MODE 1;POSITION
50.0000;TIME
0.0000;:HORIZONTAL:MAIN:SCALE
200.0000E-9;POSITION
50.0000;:HORIZONTAL:RECORDLENGTH 2500;ROLL
AUTO
;FASTFRAME:STATE 0;LENGTH 2500;COUNT 2;SE-
LECTED:CH1 2
;CH2 2;CH3 2;CH4 2;MATH1 2;MATH2 2;MATH3
2;MATH4 2
;REF1 2;REF2 2;REF3 2;REF4 2;:HORIZON-
TAL:FASTFRAME
:REF:SOURCE CH1;FRAME 1;:HORIZONTAL:FAST-
FRAME:TRACK
LIVE;:DIAG:LEVEL SUBSYS;:TRIGGER:A:MODE
AUTO;TYPE
EDGE;LEVEL 0.0000;HOLDOFF:BY DEFAULT;TIME
250.0000E-9;
:TRIGGER:A:EDGE:SOURCE CH1;COUPLING
DC;SLOPE RISE;
:TRIGGER:A:LOGIC:CLASS PATTERN;FUNCTION
AND;WHEN TRUE
;THRESHOLD:CH1 1.4000;CH2 1.4000;CH3
1.4000;CH4 1.4000;
:TRIGGER:A:LOGIC:INPUT:CH1 HIGH;CH2 X;CH3
X;:TRIGGER:A
:LOGIC:PATTERN:INPUT:CH4 X;:TRIGGER:A:LOG-
IC:PATTERN
:WHEN TRUE;WHEN:LESSLIMIT 5.0000E-9;MORE-
LIMIT 5.0000E-9;
:TRIGGER:A:LOGIC:SETHOLD:CLOCK:EDGE
RISE;THRESHOLD 1.4000
;SOURCE CH2;:TRIGGER:A:LOGIC:SE-
THOLD:DATA:THRESHOLD
1.4000;SOURCE CH1;:TRIGGER:A:LOGIC:SE-
THOLD:HOLDTIME
2.0000E-9;SETTIME 3.0000E-9;:TRIG-
GER:A:LOGIC:STATE:INPUT
:CH4 RISE;:TRIGGER:A:LOGIC:STATE:WHEN
TRUE;:TRIGGER:A
:PULSE:CLASS GLITCH;SOURCE
CH1;GLITCH:WIDTH 2.0000E-9
;TRIGIF ACCEPT;POLARITY POSITIVE;:TRIG-
GER:A:PULSE:RUNT
:POLARITY POSITIVE;THRESHOLD:HIGH

```

```

1.2000;LOW 800.0000E-3;
:TRIGGER:A:PULSE:RUNT:WHEN OCCURS;WIDTH
2.0000E-9;
:TRIGGER:A:PULSE:TRANSITION:DELTATIME
2.0000E-9;POLARITY
  POSITIVE;THRESHOLD:HIGH 1.2000;LOW
800.0000E-3;:TRIGGER
:A:PULSE:TRANSITION:WHEN SLOWERTHAN;:TRIG-
GER:A:PULSE
:WIDTH:LOWLIMIT 2.0000E-9;HIGHLIMIT
2.0000E-9;WHEN WITHIN
;POLARITYPOSITIVE;:TRIGGER:A:PULSE:TIME-
OUT:POLARITY
STAYSHIGH;TIME 2.0000E-9;:TRIGGER:B:STATE
0;TYPE EDGE
;LEVEL 0.0000;BY EVENTS;EDGE:SOURCE
CH1;SLOPE RISE
;COUPLING DC;:TRIGGER:B:TIME
16.0000E-9;EVENTS:COUNT 2;
:MATH1:DEFINE " ";NUMAVG 2;SCALE
1.0000;POSITION 0.0000
;LABEL:NAME "";XPOS 5;YPOS 65;:MATH1:SPEC-
TRAL:MAG DB
;PHASE DEGREES;GATEPOS -800.0000E-12;GATE-
WIDTH 1.9992E-6
;REFLEVEL 20.0000;REFLEVELOFFSET
223.6000E-3;SPAN
600.0000E+6;CENTER 325.0000E+6;RESBW
1.0004E+6;WINDOW
GAUSSIAN;SUPPRESS -35000.0000E-3;UNWRAP
0;LOCK 0;
:MATH2:DEFINE " ";NUMAVG 2;SCALE
1.0000;POSITION 0.0000
;LABEL:NAME "";XPOS 5;YPOS 80;:MATH2:SPEC-
TRAL:MAG DB
;PHASE DEGREES;GATEPOS -800.0000E-12;GATE-
WIDTH 1.9992E-6
;REFLEVEL 20.0000;REFLEVELOFFSET
223.6000E-3;SPAN
600.0000E+6;CENTER 325.0000E+6;RESBW
1.0004E+6;WINDOW
GAUSSIAN;SUPPRESS -35000.0000E-3;UNWRAP
0;LOCK 0;
:MATH3:DEFINE " ";NUMAVG 2;SCALE
1.0000;POSITION 0.0000
;LABEL:NAME "";XPOS 5;YPOS 95;:MATH3:SPEC-
TRAL:MAG DB
;PHASE DEGREES;GATEPOS -800.0000E-12;GATE-
WIDTH 1.9992E-6
;REFLEVEL 20.0000;REFLEVELOFFSET
223.6000E-3;SPAN

```

```

600.0000E+6;CENTER 325.0000E+6;RESBW
1.0004E+6;WINDOW
GAUSSIAN;SUPPRESS -35000.0000E-3;UNWRAP
0;LOCK 0;
:MATH4:DEFINE " ";NUMAVG 2;SCALE
1.0000;POSITION 0.0000
;LABEL:NAME "";XPOS 5;YPOS
110;;MATH4:SPECTRAL:MAG DB
;PHASE DEGREES;GATEPOS -800.0000E-12;GATE-
WIDTH 1.9992E-6
;REFLEVEL 20.0000;REFLEVELOFFSET
223.6000E-3
;SPAN 600.0000E+6;CENTER 325.0000E+6;RESBW
1.0004E+6
;WINDOW GAUSSIAN;SUPPRESS
-35000.0000E-3;UNWRAP 0;LOCK 0
;:HISTOGRAM:BOXPCNT
30.0000,25.1000,70.0000,75.2000
;DISPLAY LINEAR;STATE 0;FUNCTION HORIZON-
TAL;SIZE 2.0000
;SOURCE CH1;;CH1:BANDWIDTH 1.0000E+9;COU-
PLING DC;DESKEW
0.0000;OFFSET 0.0000;POSITION 0.0000;SCALE
100.0000E-3
;TERMINATION 1.0000E+6;PROBEFUNC:EXTATTEN
1.0000
;EXTUNITS "None";:CH1:LABEL:NAME "";XPOS
5;YPOS 5;
:CH2:BANDWIDTH 1.0000E+9;COUPLING DC;DES-
KEW 0.0000
;OFFSET 0.0000;POSITION 0.0000;SCALE
100.0000E-3
;TERMINATION 1.0000E+6;PROBEFUNC:EXTATTEN
1.0000
;EXTUNITS "None";:CH2:LABEL:NAME "";XPOS
5;YPOS 20;
:CH3:BANDWIDTH 1.0000E+9;COUPLING DC;DES-
KEW 0.0000
;OFFSET 0.0000;POSITION 0.0000;SCALE
100.0000E-3
;TERMINATION 1.0000E+6;PROBEFUNC:EXTATTEN
1.0000
;EXTUNITS "None";:CH3:LABEL:NAME "";XPOS
5;YPOS 35;
:CH4:BANDWIDTH 1.0000E+9;COUPLING DC;DES-
KEW 0.0000
;OFFSET 0.0000;POSITION 0.0000;SCALE
100.0000E-3
;TERMINATION 1.0000E+6;PROBEFUNC:EXTATTEN
1.0000
;EXTUNITS "None";:CH4:LABEL:NAME "";XPOS

```

```

5;YPOS 50;
:SELECT:CH1 1;CH2 0;CH3 0;CH4 0;MATH1
0;MATH2 0
;MATH3 0;MATH4 0;REF1 0;REF2 0;REF3 0;REF4
0;CONTROL
CH1;:CURSOR:STATE OFF;FUNCTION VBARS;MODE
INDEPENDENT
;SOURCE CH1;VBARS:UNITS SECONDS;POSITION1
-800.0000E-9
;POSITION2 800.0000E-9;:CURSOR:HBARS:POSI-
TION1
300.0000E-3;POSITION2 -300.0000E-3;:CUR-
SOR:PAIRED
:POSITION1 -800.0000E-9;POSITION2
800.0000E-9;:CURSOR
:SPLIT:POSITION1 -800.0000E-9;POSITION2
800.0000E-9
;SOURCE2 CH1;UNITS BASE;MEASUREMENT:GATING
OFF;METHOD
HISTOGRAM;IMMED:TYPE UNDEFINED;SOURCE1
CH1;SOURCE2 CH1
;DELAY:EDGE1 RISE;EDGE2 RISE;DIRECTION
FORWARDS;
:MEASUREMENT:MEAS1:STATE 0;TYPE UNDE-
FINED;SOURCE1 CH1
;SOURCE2 CH1;DELAY:EDGE1 RISE;EDGE2
RISE;DIRECTION
FORWARDS;:MEASUREMENT:MEAS2:STATE 0;TYPE
UNDEFINED
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1
RISE;EDGE2 RISE
;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS3:STATE 0
;TYPE UNDEFINED;SOURCE1 CH1;SOURCE2
CH1;DELAY:EDGE1
RISE;EDGE2 RISE;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS4
:STATE 0;TYPE UNDEFINED;SOURCE1
CH1;SOURCE2 CH1
;DELAY:EDGE1 RISE;EDGE2 RISE;DIRECTION
FORWARDS;
:MEASUREMENT:MEAS5:STATE 0;TYPE UNDE-
FINED;SOURCE1 CH1
;SOURCE2 CH1;DELAY:EDGE1 RISE;EDGE2
RISE;DIRECTION
FORWARDS;:MEASUREMENT:MEAS6:STATE 0;TYPE
UNDEFINED
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1
RISE;EDGE2 RISE
;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS7:STATE 0;TYPE

```



```

UNDEFINED;SOURCE1 CH1;SOURCE2
CH1;DELAY:EDGE1 RISE
;EDGE2 RISE;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS8
:STATE 0;TYPE UNDEFINED;SOURCE1
CH1;SOURCE2 CH1;DELAY
:EDGE1 RISE;EDGE2 RISE;DIRECTION FOR-
WARDS;:MEASUREMENT
:REFLEVEL:METHOD PERCENT;ABSOLUTE:HIGH
0.0000;LOW 0.0000
;MID1 0.0000;MID2 0.0000;:MEASUREMENT:RE-
FLEVEL:PERCENT
:HIGH 90.0000;LOW 10.0000;MID1
50.0000;MID2 50.0000;
:MEASUREMENT:STATISTICS:MODE OFF;WEIGHTING
32;:ZOOM
:MODE 0;GRATICULE:SIZE 50;:ZOOM:HORIZON-
TAL:LOCK ALL;
:ZOOM:CH1:HORIZONTAL:POSITION
50.0000;SCALE 2;:ZOOM
:CH1:VERTICAL:POSITION 0.0000;SCALE
1.0000;:ZOOM:CH2
:HORIZONTAL:POSITION 50.0000;SCALE
2;:ZOOM:CH2:VERTICAL
:POSITION 0.0000;SCALE
1.0000;:ZOOM:CH3:HORIZONTAL
:POSITION 50.0000;SCALE 2;:ZOOM:CH3:VER-
TICAL:POSITION
0.0000;SCALE 1.0000;:ZOOM:CH4:HORIZON-
TAL:POSITION
50.0000;SCALE 2;:ZOOM:CH4:VERTICAL:POSI-
TION 0.0000
;SCALE 1.0000;:ZOOM:MATH1:HORIZONTAL:POSI-
TION 50.0000
;SCALE 2;:ZOOM:MATH1:VERTICAL:POSITION
0.0000;SCALE 1.0000;:ZOOM:MATH2:HORIZON-
TAL:POSITION 50.0000;SCALE 2;
:ZOOM:MATH2:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:MATH3:HORIZONTAL:POSITION
50.0000;SCALE 2;:ZOOM
:MATH3:VERTICAL:POSITION 0.0000;SCALE
1.0000;:ZOOM
:MATH4:HORIZONTAL:POSITION 50.0000;SCALE
2;:ZOOM:MATH4
:VERTICAL:POSITION 0.0000;SCALE
1.0000;:ZOOM:REF1
:HORIZONTAL:POSITION 50.0000;SCALE
2;:ZOOM:REF1:VERTICAL
:POSITION 0.0000;SCALE
1.0000;:ZOOM:REF2:HORIZONTAL

```

```
:POSITION 50.0000;SCALE 2;:ZOOM:REF2:VER-
TICAL:POSITION
0.0000;SCALE 1.0000;:ZOOM:REF3:HORIZON-
TAL:POSITION
50.0000;SCALE 2;:ZOOM:REF3:VERTICAL:POSI-
TION 0.0000
;SCALE 1.0000;:ZOOM:REF4:HORIZONTAL:POSI-
TION 50.0000
;SCALE 2;:ZOOM:REF4:VERTICAL:POSITION
0.0000;SCALE
1.0000;:REF1:LABEL:NAME "";XPOS 5;YPOS
125;:REF1:SCALE
1.0000;POSITION 0.0000;:REF2:LABEL:NAME
"";XPOS 5;YPOS
140;:REF2:SCALE 1.0000;POSITION
0.0000;:REF3:LABEL:NAME
"";XPOS 5;YPOS 155;:REF3:SCALE
1.0000;POSITION 0.0000;
:REF4:LABEL:NAME "";XPOS 5;YPOS
170;:REF4:SCALE 1.0000
;POSITION 0.0000;:DATA:DESTINATION
REF1;ENCDG RIBINARY
;SOURCE CH1;START 1;STOP 500;WIDTH 8
```

***OPC**

Description

This command allows you to synchronize the operation of the oscilloscope with your application program. For more information, see Synchronization Methods.

The *OPC command generates the operation complete message in the Standard Event Status Register (SESR) when all pending commands are complete.

The *OPC? query places the ASCII character "1" into the output queue when all such commands are complete. The *OPC? response is not available to read until all pending operations finish.

For a complete discussion of the use of these registers and the output queue, see Registers and Queues.

Group

Status and Error

Related Commands

BUSY?, *WAI

Syntax 1

*OPC

Syntax 2

*OPC?

Example 1

*OPC

Upon completion of all pending (OPC related) operations, this command generates the operation complete message in the SESR.

Example 2

*OPC?

This query might return

1

to indicate that all pending (OPC related) operations are finished.

***OPT?**

Description

This query-only command returns a list of the options installed in your oscilloscope.

Group

Status and Error

Syntax

*OPT?

Example

*OPT?

This query might return

:3M:16 Meg Max 4M/CH

*PSC

Description

This command sets and queries the power-on status flag that controls the automatic power-on handling of the DESER, SRER, and ESER registers. When *PSC is true, the DESER register is set to 255 and the SRER and ESER registers are set to 0 at power-on. When *PSC is false, the current values in the DESER, SRER, and ESER registers are preserved in nonvolatile memory when power is shut off and are restored at power-on.

Group

Status and Error

Related Commands

DESE, *ESE, FACtory, *RST, *SRE

Syntax 1

*PSC <NR1>

Syntax 2

*PSC?

Argument

- <NR1>

A 0 sets the power-on status clear flag to false, disables the power-on clear and allows the oscilloscope to possibly assert SRQ after power-on; any other value sets the power-on status clear flag to true, enabling the power-on status clear and prevents any SRQ assertion after power on.

Example 1

*PSC 0

This command sets the power-on status clear flag to false.

Example 2

*PSC?

The query form of this command might return

1

to indicate that the power-on status clear flag is set to true.

*PUD

Description

This command sets or queries a string of Protected User Data. This data is protected by the PASSWord command. You can modify it only by first entering the correct password. This password is not necessary to query the data.

Group

Status and Error

Related Commands

PASSWord

Syntax 1

```
*PUD {<Block>|<String>}
```

Syntax 2

```
*PUD?
```

Argument

- <Block>

This is a block containing up to 100 characters.

- <String>

This is a string containing up to 100 characters.

Example 1

```
*PUD #229This oscilloscope belongs to me
```

This command stores the string "This oscilloscope belongs to me" in the user protected data area.

Example 2

```
*PUD?
```

This query might return

```
#221Property of Company X
```

***RCL**

Description

This command (no query form) restores the state of the oscilloscope from a copy of the settings stored in memory (The settings are stored using the *SAV command). If 'factory' is referenced (by specifying '0'), the factory default values will be restored. This command is equivalent to RECALL:SETUp and performs the same function as selecting Instrument Setup from the File menu and then choosing the Recall Setups tab.

Group

Save and Recall

Related Commands

DELEte:SETUp, FACtory, *LRN?, RECAll:SETUp, *RST, *SAV, SAVE:SETUp

Syntax

*RCL <NR1>

Argument

- <NR1>

This specifies a setup storage location value, ranging from 0 through 10. Using an out-of-range value causes an execution error.

Example

*RCL 3

This command restores the oscilloscope from a copy of the settings stored in memory location 3.

*RST

Description

This command (no query form) resets the oscilloscope to the factory default settings. The *RST command does not alter the following:

- The state of the IEEE Std 488.1-1987 interface
- The selected IEEE Std 488.1-1987 address of the oscilloscope
- Calibration data that affect device specifications
- The Output Queue
- The Service Request Enable Register setting
- The Power-on status clear flag setting
- Alias definitions
- Stored settings
- The *PUD? Response

This command is equivalent to pressing the **DEFAULT SETUP** button on the front-panel.

Group

Status and Error

Related Commands

FACtory, RECAI:SETUp, SAVE:SETUp

Syntax

*RST

Argument

None.

Example

*RST

This command resets the oscilloscope settings to factory defaults.

***SAV**

Description

This command (no query form) stores the state of the oscilloscope to a specified memory location. You can later use the *RCL command to restore the oscilloscope to this saved state. This is equivalent to selecting Instrument Setup from the File menu and then choosing the Save Setups tab.

Group

Save and Recall

Related Commands

*RCL, RECALL:SETUp, SAVE:SETUp

Syntax

*SAV <NR1>

Argument

- <NR1>

This specifies a location in which to save the state of the scope. Location values range from 1 through 10. Using an out-of-range location value causes an execution error. Any settings that have been stored previously at this location will be overwritten.

Example

*SAV 2

This command saves the current oscilloscope state in memory location 2.

*SDS

Description

This command (no query form) changes the specified setup to reference the factory setup instead of the specific user setup slot. The content of the setup slot is unchanged, but the data will no longer be accessible to you. This command is equivalent to selecting Instrument Setups from the File menu, choosing the Recall Setups tab and then clicking the Default button.

Group

Save and Recall

Related Commands

DELEte:SETUp

Syntax

*SDS <NR1>

Argument

- <NR1>

This specifies a user setup location that will be redirected. The setup in that position, which ranges from 1 through 10, will now refer to the Factory setup when recalled.

Example

*SDS 2

This command changes setup slot 2 to reference the factory setup.

*SRE

Description

The *SRE (Service Request Enable) command sets and queries the bits in the Service Request Enable Register. For more information, refer to Registers.

Group

Status and Error

Related Commands

*CLS, DESE, *ESE, *ESR?, EVENT?, EVMsg?, FACtory, *STB?

Syntax 1

*SRE <NR1>

Syntax 2

*SRE?

Argument

■ <NR1>

This is a value in the range from 0 through 255. The binary bits of the SRER are set according to this value. Using an out-of-range value causes an execution error. The power-on default for SRER is 0 if *PSC is 1. If *PSC is 0, the SRER maintains the previous power cycle value through the current power cycle.

Example 1

*SRE 48

This command sets the bits in the SRER to 00110000 binary.

Example 2

*SRE?

This query might return

32

showing that the bits in the SRER have the binary value of 00100000.

***STB?**

Description

The *STB? (Read Status Byte) query returns the contents of the Status Byte Register (SBR) using the Master Summary Status (MSS) bit. For more information, refer to Registers.

Group

Status and Error

Related Commands

*CLS, DESE, *ESE, *ESR?, EVENT?, EVMsg?, FACtory, *SRE?

Syntax

*STB?

Returns

<NR1>

Example

*STB?

This query might return

96

showing that the SBR contains the binary value 01100000.

***TRG**

Description

This command (no query form) performs the group execute trigger on commands defined by *DDT.

Group

Miscellaneous

Related Commands

*DDT

Syntax

*TRG

Example

*TRG

This command immediately executes all commands that have been defined by *DDT.

***TST?**

Description

This query-only command tests (self-test) the GPIB interface and returns a 0.

Group

Miscellaneous

Syntax

*TST?

Example

*TST?

This query always returns 0

***WAI**

Description

The *WAI (Wait) command (no query form) prevents the oscilloscope from executing further commands or queries until all pending commands are complete. This command allows you to synchronize the operation of the oscilloscope with your application program. For more information, refer to Synchronization Methods.

Group

Status and Error

Related Commands

BUSY?, *OPC

Syntax

*WAI

Example

*WAI

This command prevents the oscilloscope from executing any further commands or queries until all pending commands are complete.

ACQUIRE?

Description

This query-only command returns the following current acquisition parameters:

- Stop after
- Acquisition state
- Mode
- Number of envelopes
- Number of averages
- Repetitive signals

Group

Acquisition

Related Commands

ACQUIRE:MODE, ACQUIRE:NUMACQ, ACQUIRE:NUMAVG, ACQUIRE:NUMENV, ACQUIRE:STATE, ACQUIRE:STOPAFTER

Syntax

ACQUIRE?

Example

ACQUIRE?

This query might return the following string for the current acquisition parameters:

```
:ACQUIRE:STOPAFTER RUNSTOP;STATE 1;MODE  
SAMPLE;NUMENV 10;NUMAVG 16;REPET ON
```

ACQuire:MODE

Description

This command sets or queries the acquisition mode of the oscilloscope. This affects all live waveforms. This command is equivalent to selecting Horizontal/Acquisition from the Horiz/Acq menu and then choosing the desired mode from the Acquisition Mode group box.

Waveforms are the displayed data point values taken from acquisition intervals. Each acquisition interval represents a time duration set by the horizontal scale (time per division). The oscilloscope sampling system always samples at the maximum rate and so an acquisition interval may include more than one sample.

The acquisition mode (which you set using this ACQuire:MODE command) determines how the final value of the acquisition interval is generated from the many data samples.

Group

Acquisition

Related Commands

ACQuire:NUMAVg, ACQuire:NUMenv, CURVe?

Syntax 1

```
ACQuire:MODE{SAMPle | PEAKdetect | HIRes | AVEr-  
age | ENVelope | WFMDB}
```

Syntax 2

```
ACQuire:MODE?
```

Arguments

- SAMPle

Specifies that the displayed data point value is simply the first sampled value that is taken during the acquisition interval. In sample mode, all waveform data has 8 bits of precision. You can request 16 bit data with a CURVe? query but the lower-order 8 bits of data will be zero. SAMPle is the default mode.

- PEAKdetect

Specifies the display of high-low range of the samples taken from a single waveform acquisition. The high-low range is displayed as a vertical column that extends from

the highest to the lowest value sampled during the acquisition interval. `PEAKdetect` mode can reveal the presence of aliasing or narrow spikes.

- **HIRes**

Specifies Hi Res mode where the displayed data point value is the average of all the samples taken during the acquisition interval. This is a form of averaging, where the average comes from a single waveform acquisition. The number of samples taken during the acquisition interval determines the number of data values that compose the average.

- **AVERage**

Specifies averaging mode, in which the resulting waveform shows an average of `SAMple` data points from several separate waveform acquisitions. The oscilloscope processes the number of waveforms you specify into the acquired waveform, creating a running exponential average of the input signal. The number of waveform acquisitions that go into making up the average waveform is set or queried using the `ACQUIRE:NUMAVg` command.

- **ENVELOpe**

Specifies envelope mode, where the resulting waveform shows the `PEAKdetect` range of data points from several separate waveform acquisitions. The number of waveform acquisitions that go into making up the envelope waveform is set or queried using the `ACQUIRE:NUMENV` command.

- **WFMDB**

Specifies waveform database (`WfmDB`) mode, where the oscilloscope acquires source waveform data in three dimensions. In addition to acquiring amplitude and timing information, it acquires count information. The count represents the number of times a specific sample point (time and amplitude) has been acquired. The value at each point on the waveform is a counter that reflects the hit intensity. The oscilloscope acquires data after each trigger event using `Sample` mode. It then determines the location of each sample point and accumulates it with shared data from previous acquisitions. Color-graded displays based on counts can be used to highlight waveform activity.

Example 1

```
ACQUIRE:MODE ENVELOpe
```

Sets the acquisition mode to display a waveform that is an envelope of many individual waveform acquisitions.

Example 2

ACQUIRE:MODE?

This command might return

:ACQUIRE:MODE AVERAGE

indicating that the displayed waveform is the average of the specified number of waveform acquisitions.

ACQUIRE:NUMACQ?

Description

This query-only command returns the number of waveform acquisitions that have occurred since starting acquisition with the ACQUIRE:STATE RUN command. This value is reset to zero when any acquisition, horizontal, or vertical arguments that affect the waveform are changed. The maximum number of acquisitions that can be counted is $2^{63}-1$. The oscilloscope stops counting when this number is reached. This is the same value that displays in the upper center of the screen when the acquisition system is stopped.

Note: In FastFrame or InstaVu modes, the oscilloscope updates the acquisition count in groups of counts.

Group

Acquisition

Related Commands

ACQUIRE:STATE

Syntax

ACQUIRE:NUMACQ?

Example

ACQUIRE:NUMACQ?

This command might return

```
:ACQUIRE:NUMACQ 350
```

indicating that 350 acquisitions have occurred since executing an ACQUIRE:STATE RUN command.

ACQuire:NUMAVg

Description

This command sets or queries the number of waveform acquisitions that make up an averaged waveform. Use the ACQuire:MODE command to enable the Average mode. Sending this command is equivalent to selecting Horizontal/Acquisition Setup from the Horiz/Acq menu, selecting the Acquisition tab, and then choosing Average from the Acquisition Mode group box. Then enter the desired number of waveforms that will be used to make up an averaged waveform in the # of Wfms box.

Group

Acquisition

Related Commands

ACQuire:MODE

Syntax 1

```
ACQuire:NUMAVg <NRf>
```

Syntax 2

```
ACQuire:NUMAVg?
```

Arguments

- NRf

This is the number of waveform acquisitions to average.

Example 1

```
ACQuire:NUMAVg 10
```

This command specifies that 10 waveform averages will be performed before exponential averaging starts.

Example 2

```
ACQuire:NUMAVg?
```

This command might return

```
:ACQUIRE:NUMAVG 75
```

indicating that there are 75 acquisitions specified for averaging.

ACQUIRE:NUMENV

Description

This command sets or queries the number of waveform acquisitions that make up an envelope waveform. Sending this command is equivalent to setting the Envelope count in the Acquisition Mode side menu.

Group

Acquisition

Syntax 1

```
ACQUIRE:NUMENV {<NRf> | INFINITE}
```

Syntax 2

```
ACQUIRE:NUMENV?
```

Arguments

- NRf

This is the number of waveform acquisitions to be enveloped.

Example 1

```
ACQUIRE:NUMENV 10
```

This command specifies that an enveloped waveform will show the result of combining 10 separately acquired waveforms.

Example 2

```
ACQUIRE:NUMENV?
```

This command might return

```
:ACQUIRE:NUMENV 0
```

indicating that acquisitions are acquired infinitely for enveloped waveforms.

ACQUIRE:NUMSAMPLES

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or returns the number of waveform database (WfmDB) points the oscilloscope can acquire before it stops a single sequence acquisition or stops running a mask test.

Hint: a bigger sample size may allow a greater throughput (more waveforms to get acquired at a time). However, a very large sample size may slow down the display update. So if you find yourself wanting to speed up the display rate, consider reducing the sample size.

This command works the same as the MASK:TEST:SAMPLE command.

Group

Acquisition

Syntax 1

```
ACQUIRE:NUMSAMPLES {<NR1>}
```

Syntax 2

```
ACQUIRE:NUMSAMPLE?
```

Arguments

- <NR1>

The number of points to sample.

Example

```
ACQUIRE:NUMSAMPLES 100000
```

Sets the oscilloscope to acquire 100,000 WfmDB points.

ACQUIRE:SAMPLINGMODE

Description

This command sets or queries the Sampling Mode parameter. This command replaces the ACQUIRE:REPEt command.

Group

Acquisition

Related Commands

ACQUIRE:STATE, ACQUIRE:STOPAfter

Syntax 1

```
ACQUIRE:SAMPLINGMODE {RT|IT|ET}
```

Syntax 2

```
ACQUIRE:SAMPLINGMODE?
```

Arguments

- RT

Sets the oscilloscope sampling mode to Real Time Only.

- IT

Sets the oscilloscope sampling mode to Interpolated Real Time.

- ET

Sets the oscilloscope sampling mode to Equivalent Time.

Example

```
ACQUIRE:SAMPLINGMODE IT
```

Sets the oscilloscope to Interpolated Real Time sampling mode.

ACQuire:STATE

Description

This command starts or stops acquisitions. When state is set to ON or RUN, a new acquisition will be started. If the last acquisition was a single acquisition sequence, a new single sequence acquisition will be started. If the last acquisition was continuous, a new continuous acquisition will be started.

If RUN is issued in the middle of completing a single sequence acquisition (for examples, averaging or enveloping), the acquisition sequence is restarted, and any accumulated data is discarded. Also, the oscilloscope resets the number of acquisitions. If the RUN argument is issued while in continuous mode, a reset occurs and acquired data continues to acquire.

Sending this command is equivalent to pressing the front-panel RUN/STOP button.

Group

Acquisition

Related Commands

ACQuire:STOPAfter,

Syntax 1

```
ACQuire:STATE{OFF|ON|RUN|STOP|<NR1>}
```

Syntax 2

```
ACQuire:STATE?
```

Arguments

- OFF

This argument stops acquisitions.

- STOP

This argument stops acquisitions.

- ON

This argument starts acquisitions.

- RUN

This argument starts acquisitions.

- NR1

0 stops acquisitions; any other value starts acquisitions.

Example 1

ACQuire:STATE RUN

This command starts the acquisition of waveform data and resets the count of the number of acquisitions.

Example 2

ACQuire:STATE?

This query might return

:ACQUIRE:STATE 0

indicating that the acquisition is stopped

ACQUIRE:STOPAFTER

Description

This command sets or queries whether the oscilloscope continually acquires acquisitions or acquires a single sequence. This command is equivalent to pressing **SINGLE** from the front-panel.

Group

Acquisition

Related Commands

ACQUIRE:STATE

Syntax 1

```
ACQUIRE:STOPAFTER {RUNSTOP|SEQUENCE}
```

Syntax 2

```
ACQUIRE:STOPAFTER?
```

Arguments

- RUNSTOP

Specifies that the oscilloscope will continually acquire data, if ACQUIRE:STATE is turned on.

- SEQUENCE

Specifies that the next acquisition will be a single-sequence acquisition.

Example 1

```
ACQUIRE:STOPAFTER RUNSTOP
```

This command sets the oscilloscope to continually acquire data.

Example 2

```
ACQUIRE:STOPAFTER?
```

This query might return

```
:ACQUIRE:STOPAFTER SEQUENCE
```

indicating that the next acquisition the oscilloscope makes will be of the single-sequence type.

ALias

Description

This command sets or queries the state of alias functionality.

Group

Alias

Related Commands

ALias:STATE

Syntax 1

```
ALias {OFF | ON | <NR1>}
```

Syntax 2

```
ALias?
```

Arguments

- OFF

This turns Alias expansion off. If a defined alias is sent when ALias:STATE is off, a command error (102) will be generated.

- ON

This turns Alias expansion on. When a defined alias is received, the specified command sequence is substituted for the alias and executed.

- NR1

A 0 disables Alias mode; any other value enables Alias mode.

Example 1

```
ALias ON
```

This command turns the alias feature on. When a defined alias is received, the specified command sequence is substituted for the alias and executed.

Example 2

```
ALias?
```

This query might return the following string for the current alias state:

```
:ALIAS:STATE 1
```

ALias:CATalog?

Description

This query-only command returns a list of the currently defined alias labels, separated by commas. If no aliases are defined, the query returns the string "".

Group

Alias

Syntax

```
ALias:CATalog?
```

Example

```
ALias:CATalog?
```

This query might return the string `:ALIAS:CATALOG "SETUP1","TESTMENU1","DEFAULT"`, showing that there are three aliases named SETUP1, TESTMENU1, and DEFAULT.

ALias:DEFine

Description

This command assigns a sequence of program messages to an alias label. These messages are then substituted for the alias whenever it is received as a command or query, provided that ALias:STATE has been turned on. The query form of this command returns the definitions of a selected alias.

Note: Attempting to give two aliases the same name causes an error. To give a new alias the name of an existing alias, the existing alias must first be deleted.

Group

Alias

Related Commands

ALias:STATE

Syntax 1

```
ALias:DEFine  
<Qstring><,>{<Qstring>|<Block>}
```

Syntax 2

```
ALias:DEFine?
```

Arguments

- The first <Qstring> is the alias label.

This label cannot be a command name. Labels must start with a letter and can contain only letters, numbers, and underscores; other characters are not allowed. The label must be less than or equal to 12 characters.

- The second <Qstring> or <Block> is a complete sequence of program messages.

The messages can contain only valid commands that must be separated by semicolons and must follow all rules for concatenating commands. The sequence must be less than or equal to 256 characters.

Example 1

```
ALIAS:DEFINE "ST1",":RECALL:SETUP 5;:AUTO-  
SET EXECUTE;:SELECT:CH1 ON"
```

This command defines an alias named "ST1" that sets up the oscilloscope.

Example 2

ALIAS:DEFINE? "ST1"

This command returns :ALIAS:DEFINE "ST1",#246
:RECALL:SETUP 5;:AUTOSET EXECUTE;:SE-
LECT:CH1 ON

ALias:DELEte

Description

This command removes a specified alias and is identical to ALias:DELEte:NAME. An error message is generated if the named alias docs do not exist.

Group

Alias

Related Commands

*ESR?, ALias:DELEte:ALL

Syntax

```
ALias:DELEte <Qstring>
```

Argument

- <Qstring>

This is the name of the alias to be removed. Using ALias:DELEte without specifying an alias causes an execution error. <Qstring> must be a previously defined value.

Example

```
ALias:DELEte "SETUP1"
```

This command deletes the alias named SETUP1.

ALias:DELEte:ALL

Description

This command deletes all existing aliases.

Group

Alias

Related Commands

ALias:DELEte, ALias:DELEte:NAME

Syntax

```
ALias:DELEte:ALL
```

Example

```
ALias:DELEte:ALL
```

This command deletes all existing aliases.

ALias:DELEte:NAME

Description

This command removes a specified alias. An error message is generated if the named alias does not exist. This command is identical to ALias:DELEte.

Group

Alias

Syntax

```
ALias:DELEte:NAME <Qstring>
```

Argument

- <Qstring>

This is the name of the alias to remove. Using ALias:DELEte:NAME without specifying an alias causes an exception error. <Qstring> must be a previously defined alias.

Example

```
ALias:DELEte:NAME "STARTUP"
```

This command deletes the alias named STARTUP.

ALias:STATE

Description

This command turns aliases on or off and is identical to the ALias command. The query form of this command returns the state of the aliases.

Group

Alias

Syntax 1

```
ALias:STATE {OFF|ON|<NR1>}
```

Syntax 2

```
ALias:STATE?
```

Arguments

- OFF

This turns alias expansion off. If a defined alias is sent when ALias:STATE is OFF, the instrument generates a command error (102).

- ON

This turns alias expansion on. When a defined alias is received, the specified command sequence is substituted for the alias and executed.

- <NR1>

A 0 turns off aliases; any other value turns on aliases.

Example 1

```
ALias:STATE OFF
```

This command turns off the alias feature.

Example 2

```
ALias:STATE?
```

This query might return :ALIAS:STATE ON, indicating that alias expansion is currently turned on.

ALLEV?

Description

This query-only command prompts the oscilloscope to return all events and their messages (delimited by commas), and removes the returned events from the Event Queue. Use the *ESR? query to enable the events to be returned. This command is similar to repeatedly sending *EVMsg? queries to the oscilloscope.

Group

Status and Error

Related Commands

*ESR?, *EVMsg?

Syntax

ALLev?

Example

ALLev?

This query might return

```
:ALLEV 2225,"Measurement error, No wave-  
form to measure; "420,"Query UNTERMI-  
NATED;"
```

AUTOSet

Description

AUTOSet EXECute causes the digitizing oscilloscope to adjust its vertical, horizontal, and trigger controls to provide a stable display of the selected waveform. This is equivalent to pressing the front-panel **AUTOSET** button. For a detailed description of autoset functionality, see *Autoset* in the index of your oscilloscope online help.

Group

Miscellaneous

Syntax

AUTOSet EXECute

Argument

- EXECute

This autosets the displayed waveform.

- UNDo

This initiates an instrument Autoset undo. It attempts to return the instrument to a pre-Autoset state. Only instrument settings that Autoset directly changes will be undone.

The following examples explain how it would work in different situations:

If the user alters Horizontal Position prior to Autoset, executes Autoset, and then executes Autoset Undo, the Horizontal Position value should remain constant throughout. Autoset Undo will not change Horizontal Position and other settings not directly changed by Autoset.

If the user executes Autoset, alters Vertical Scale, and then executes Autoset Undo, the Vertical Scale will be reset to the value it had prior to executing Autoset, regardless of its new value. Autoset directly changes Vertical Scale, thus Autoset Undo will restore it to the pre-Autoset value.

Restoring pre-Autoset state includes restoring FastAcq Inhibited status. For example, if the user has inhibited FastAcq by enabling Zoom mode, executes Autoset, and then executes Autoset Undo, the instrument will be returned to the inhibited state. If the user then disables Zoom, FastAcq mode will re-enable.

Only the first Autoset Undo executed after an Autoset will cause an undo of the relevant instrument state. Subsequent Autoset Undo executions, regardless of when executed, have no effect.

Example 1

AUTOSET EXECUTE

This command adjusts the vertical, horizontal, and trigger controls to provide a stable display of the selected waveform.

Example 2

AUTOSET UNDO

This command initiates an instrument Autoset undo.

AUXOut?

Description

This query-only command returns the auxiliary out setup. This query command is equivalent to selecting AUX OUT Configuration from the Utilities menu and then viewing the current settings.

Group

Miscellaneous

Related Commands

AUXOut:EDGe, AUXOut:SOUrce

Syntax

AUXOut?

Example

AUXOUT?

This query might return

```
:AUXOUT:SOURCE ATRIGGER;EDGE RISING
```

indicating that the source at the BNC connector is set to the A trigger and that the polarity is set to the rising edge of the trigger output signal.

AUXOut:EDGE

Description

This command sets or queries the direction in which the trigger output signal transitions when a trigger occurs. This command is equivalent to selecting AUX OUT Configuration from the Utilities menu and then clicking the desired Polarity setting.

Group

Miscellaneous

Related Commands

AUXOut?, AUXOut:SOURce

Syntax 1

```
AUXOut:EDGE {RISing|FALLing}
```

Syntax 2

```
AUXOut:EDGE?
```

Arguments

- RISing

This sets the polarity to the rising edge of the trigger output signal.

- FALLing

This sets the polarity to the falling edge of the trigger output signal.

Example 1

```
AUXOUT:EDGE RISING
```

This command sets the polarity to the rising edge of the trigger output signal.

Example 2

```
AUXOUT:EDGE?
```

This query might return

```
:AUXOUT:EDGE RISING
```

indicating that the polarity is set to the rising edge of the trigger output signal.

AUXOut:SOURce

Description

This command sets or queries the trigger source at the BNC connection. This command is equivalent to selecting AUX OUT Configuration from the Utilities menu and then selecting the desired Configuration setting.

Group

Miscellaneous

Related Commands

AUXOut?, AUXOut:EDGE

Syntax 1

```
AUXOut:SOURce {ATRIGger|BTRIGger|EVENT}
```

Syntax 2

```
AUXOut:SOURce?
```

Arguments

- ATRIGger

This sets the source at the BNC connector to the main trigger.

- BTRIGger

This sets the source at the BNC connector to the delayed trigger.

- EVENT

Specifies that the Aux Out BNC will deliver a TTL signal when certain testing states are reached. Specifically, this will occur when MASK:TEST:AUX:COMPLETION or MASK:TEST:AUX:FAILURE is set to ON.

Similarly, setting AUXOUT:SOURCE to anything other than EVENT will turn MASK:TEST:AUX:COMPLETION and MASK:TEST:AUX:FAILURE to OFF.

Example

```
AUXOut:SOURce?
```

This query might return

```
:AUXOUT:SOURCE ATRIGGER
```

indicating that the source at the BNC connector is set to the A trigger.

BELI

Description

This command was previously used to beep an audio indicator and is provided for backward compatibility.

Group

Miscellaneous

Syntax

BEL1

Example

BEL1

This command is accepted but does nothing.

BUSY?

Description

This query-only command returns the status of the oscilloscope. This command allows you to synchronize the operation of the oscilloscope with your application program.

Group

Status and Error

Related Commands

*OPC, *WAI

Syntax

BUSY?

Returns

<NR1>

If 0 is returned, it means that the oscilloscope is not busy processing a command whose execution time is extensive. If 1 is returned, it means that the oscilloscope is busy processing one of the commands listed in the table below.

Commands that Affect BUSY? Response	
Operation	Command
Single sequence acquisition	ACQUIRE:STATE ON or ACQUIRE:STATE RUN (when ACQUIRE:STOPAFTER is set to SEQUENCE)

Example

BUSY?

This query might return

:BUSY 1

indicating that the oscilloscope is currently busy.

CALibrate?

Description

This query-only command returns the results of the most recent calibration queries without performing an SPC (Signal Path Compensation). The returned results include the calibration status of the attached probes.

Group

Calibration

Related Commands

*CAL

Syntax

CALibrate?

Returns

■ SPC Calibration Results:

PASS

FAIL

WARMUP

RUNNING

■ Probestate Calibration Results:

0 Indicating the calibration passed.

-1 Indicating the calibration failed.

1 Indicating the calibration initialized.

2 Indicating the calibration is running.

Example

CAL?

This query might return

```
CALIBRATE:RESULTS:SPC PASS;:CALIBRATE:PRO-  
BESTATE:CH1 0;CH2 1;CH3 1;CH4 1
```

Indicating that the oscilloscope is warmed up, the Signal Path Compensation passed, the Channel 1 probe calibration passed, and the Channel 2,3, and 4 probe calibrations are initialized.

CALibrate:CALProbe:CH<x>?

Description

This query-only command instructs the digitizing oscilloscope to perform a probe calibration for the selected channel and returns the calibration status. The Channel <x> range is 1 through 4. This command is equivalent to selecting Probe Cal from the Vertical menu. You must warm up the oscilloscope for at least 20 minutes before running this command.

Note: A probe query takes approximately 30 seconds to complete. No other commands will be executed until calibration is complete.

Group

Calibration

Related Commands

CALibrate:PRObestate:CH<x>?

Syntax

CALibrate:CALProbe:CH<x>?

Returns

- -1

Indicating the probe calibration failed for the selected channel.

- 0

Indicating the probe calibration passed for the selected channel.

- 1

Indicating the probe calibration is initialized for the selected channel.

Example

CALibrate:CALProbe:CH2?

This query might return :CALIBRATE:CALPROBE:CH2 0

Indicating the probe calibration passed for Channel 2.

CALibrate:PROBEstate:CH<x>?

Description

This query-only command returns the probe calibration status for the probe of the selected channel, <1 through 4>.

Group

Calibration

Related Commands

CH<x>:PROBECal?

Syntax

CALibrate:PROBEstate:CH<x>?

Returns

- -1

Indicating the probe calibration failed for the selected channel.

- 0

Indicating the probe calibration passed for the selected channel.

- 1

Indicating the probe calibration initialized for the selected channel.

- 2

Indicating the probe calibration for the selected channel is running.

Example

```
CALIBRATE:PROBESTATE:CH1?
```

This query might return :CALIBRATE:PROBESTATE:CH1
0

Indicating that the probe calibration for Channel 1 passed.

CALibrate:RESults?

Description

This query-only command returns the status of all the calibration subsystems, without performing an SPC (Signal Path Compensation). The results returned do not include the calibration status of the attached probes. The CALibrate:RESults? Query is intended to support GO/NoGO testing of the oscilloscope calibration readiness: all returned results should indicate Pass status if the oscilloscope is "fit for duty." It is quite common, however, to use uncalibrated probes (particularly when the oscilloscope inputs are connected into a test system with coax cables). Not including the probe cal results prompts the instrument to perform a simple "every returned status field must be Pass" test.

This command is equivalent to selecting Instrument Calibration from the Utilities menu and then viewing the contents of the Status field.

Group

Calibration

Related Commands

*CAL?

Syntax

```
CALibrate:RESults?
```

Returns

- PASS
- FAIL
- WARMUP
- RUNNING

Example

```
CALibrate:RESults?
```

```
The query might return :CALIBRATE:RESULTS:SPC  
PASS
```

Indicating that the Signal Path Compensation passed.

CALibrate:RESults:SPC?

Description

This query-only command returns the results of the last SPC (Signal Path Compensation). This query does not cause an SPC to run.

Group

Calibration

Related Commands

*CAL?

Syntax

```
CALibrate:RESults:SPC?
```

Returns

- PASS
- FAIL
- WARMUP
- RUNNING

Example

```
CALibrate:RESults:SPC?
```

This query might return :CALIBRATE:RESULTS:SPC
PASS

Indicating the Signal Path Compensation passed.

CH<x>?

Description

This query-only command returns the vertical parameters for the specified channel. The channel is specified by x, which ranges from 1 through 4. This command is similar to selecting Vertical Setup from the Vertical menu.

Group

Vertical

Syntax

CH<x>?

Example

CH1?

This query might return

```
:CH1:BANDWIDTH 1.0000E+09; COUPLING
DC;DESKEW 0.0000E+00;OFFSET
0.0000E+00;POSITION 0.0000E+00
;SCALE 5.0000E-01;TERMINATION
1.0000E+06;PROBCAL INIT;
PROBE:GAIN 1.0000E+00;RESISTANCE
1.0000E+06;UNITS "V"
;ID:TYPE "1X";SERNUMBER "N/A";:CH1:PROBE-
FUNC:EXTATTEN
1.0000E+00;EXTUNITS "None";:CH1:LABEL:NAME
"";XPOS 5
;YPOS 5, showing the vertical parameters
for channel 1.
```

CH<x>:BANdwidth

Description

This command sets or queries the selectable low-pass bandwidth limit filter of the specified channel. This is equivalent to selecting Bandwidth from the Vertical menu.

Note: For TDS7254B and TDS7404B instruments, this command always sets the channel to its maximum attainable bandwidth. The query form of this command always returns the maximum bandwidth of the channel.

Group

Vertical

Syntax 1

```
CH<x>:BANdwidth {TWEnty|TWOfifty|FUL1|<NR3>}
```

Syntax 2

```
CH<x>:BANdwidth?
```

Arguments

- FUL1

This disables any optional bandwidth limiting. The specified channel operates at its maximum attainable bandwidth.

- <NR3>

This represents a double-precision ASCII string, which is then rounded to available bandwidths using geometric rounding and sets the upper bandwidth.

Example

CH2:BAWdwidth?

For TDS7254B instruments, this query might return

:CH2:BAWdWIDTH 2.5000E+09

as this command always returns the maximum bandwidth of the channel.

For TDS7404B instruments, this query might return

:CH2:BAWdWIDTH 4.0000E+09

as this command always returns the maximum bandwidth of the channel.

CH<x>:COUPling

Description

This command sets or queries the input attenuator coupling setting for the specified channel. This command is equivalent to selecting Coupling from the Vertical menu.

Note: For TDS7254B and TDS7404B instruments, the AC argument is not available.

Group

Vertical

Syntax 1

```
CH<x>:COUPling {AC|DC|GND}
```

Syntax 2

```
CH<x>:COUPling?
```

Arguments

- AC

This sets the specified channel to AC coupling. This argument is not available for TDS7254B and TDS7404B instruments.

- DC

This sets the specified channel to DC coupling.

- GND

This sets the specified channel to ground.

Example 1

```
CH2:COUPling GND
```

This command sets channel 2 to ground.

Example 2

```
CH3:COUPling?
```

This query might return

```
:CH3:COUPling DC
```

indicating that channel 3 is set to DC coupling.

CH<x>:DESKew

Description

This command sets or queries the deskew time for the specified channel. The channel is specified by x, which can range from 1 through 4. This command is equivalent to selecting Deskew from the Vertical Setup menu.

You can adjust the deskew time to add an independent, channel-based delay time to the delay (set by the horizontal position control and common to all channels) from the common trigger point to the first sample taken for each channel. This lets you compensate individual channels for different delays introduced by their individual input hook ups.

Note: Deskew does not work with Fast Acquisition XY and XYZ formats.

Group

Vertical

Syntax 1

```
CH<x>:DESKew <NR3>
```

Syntax 2

```
CH<x>:DESKew?
```

Argument

- <NR3>

This is the deskew time for this channel, ranging from -75 ns to +75 ns with a resolution of 1 ps. Out-of-range values are clipped.

Example 1

```
CH4:DESKew 5.0E-9
```

This command sets the deskew time for channel 4 to 5 ns.

Example 2

```
CH2:DESKew?
```

This query might return

```
:CH2:DESKEW 2.0000E-09
```

indicating that the deskew time for channel 2 is set to 2 ns.

CH<x>:LABEL:NAME

Description

This command sets or queries the label attached to the displayed waveform for the specified channel. This command is equivalent to selecting Label from the Vertical menu.

Group

Vertical

Syntax 1

```
CH<x>:LABEL:NAME <str>
```

Syntax 2

```
CH<x>:LABEL:NAME?
```

Argument

- <str>

This is an alphanumeric character string, ranging from 1 through 32 characters in length.

Example 1

```
CH2:LABEL:NAME "Pressure"
```

This command changes the waveform label for the CH2 waveform to "Pressure".

Example 2

```
CH3:LABEL:NAME?
```

This query might return

```
:CH3:LABEL:NAME "CH3"
```

indicating that the waveform label for the CH 3 waveform is "CH3".

CH<x>:LABEL:XPOS

Description

This command sets or queries the X screen offset at which the label (attached to the displayed waveform of the specified channel) is displayed, relative to the left edge of the screen. Channels are specified by x, which ranges from 1 through 4. This command is equivalent to selecting Label from the Vertical menu and either viewing or setting X Pos.

Group

Vertical

Syntax 1

```
CH<x>:LABEL:XPOS <NR3>
```

Syntax 2

```
CH<x>:LABEL:XPOS?
```

Argument

- <NR3>

This is the location (in divisions) where the waveform label for the selected channel is displayed, relative to the left edge of the screen.

Example 1

```
CH3:LABEL:XPOS 2.5
```

This command moves the waveform label for the CH3 waveform, so that it begins 2.5 divisions to the right of the left edge of the screen.

Example 2

```
CH2:LABEL:XPOS?
```

This query might return

```
:CH2:LABEL:XPOS 0.0
```

indicating that the waveform label for the CH2 waveform is currently on the far left edge of the screen.

CH<x>:LABEL:YPOS

Description

This command sets or queries the Y screen offset at which the label (attached to the displayed waveform of the specified channel) is displayed, relative to the vertical position of the associated waveform. Channels are specified by x, which ranges from 1 through 4. This command is equivalent to selecting Label from the Vertical menu and either viewing or setting Y Pos.

Group

Vertical

Syntax 1

```
CH<x>:LABEL:YPOS <NR3>
```

Syntax 2

```
CH<x>:LABEL:YPOS?
```

Argument

- <NR3>

This is the location (in divisions) where the waveform label for the selected channel is displayed, relative to the vertical position of the associated waveform.

Example 1

```
CH3:LABEL:YPOS -2.5
```

This command moves the waveform label for the CH3 waveform to beneath (2.5 divisions below) the associated waveform.

Example 2

```
CH2:LABEL:YPOS?
```

This query might return

```
:CH2:LABEL:YPOS 0.0000
```

indicating that the waveform label for the CH2 waveform is currently located at the same Y position as the waveform.

CH<x>:OFFSet

Description

This command sets or queries the vertical offset for the specified channel. The channel is specified by x, which can range from 1 through 4. This command is equivalent to selecting Offset from the Vertical menu.

This command offsets the vertical acquisition window (moves the level at the vertical center of the acquisition window) for the selected channel. Visualize offset as scrolling the acquisition window towards the top of a large signal for increased offset values, and scrolling towards the bottom for decreased offset values. The resolution of the vertical window sets the offset increment for this control.

Offset adjusts only the vertical center of the acquisition window for channel waveforms to help determine what data is acquired. The instrument always displays the input signal minus the offset value. The channel reference marker will move to the vertical graticule position given by the negative of the offset value divided by the scale factor, unless that position is off-screen. If the computed coordinate for the reference mark is off-screen, the mark moves to the nearest screen limit and changes from a right-pointing arrow (→) to an arrow pointing in the appropriate off-screen direction.

The settable range of a channel offset is either ± 100 V, ± 10 V or ± 1.0 V, depending on the vertical scale factor.

Vertical Scale Adjust Factors			
When Internal scale adjust gain is:	From 0.001 to 0.1	From 0.101 to 01.0	From 1.01 to 100.0
Product of Offset range, probe gain, and transducer gain is:	± 1.0 V	± 10.0 V	± 100.0 V

Note: The above table describes oscilloscope behavior only when no probe is attached, and when the external attenuation factor is 1.0.

Group

Vertical

Related Commands

CH<x>POSition

Syntax 1

CH<x>:OFFSet <NR3>

Syntax 2

CH<x>:OFFSet?

Argument

- <NR3>

This is the offset value for the specified channel, ranging from -1.6V through 1.6V.

Example 1

CH3:OFFSet 2.0E-3

This command sets the offset for channel 3 to 2mV.

Example 2

CH2:OFFSet?

This query might return

:CH4:OFFSET 1.0000E-03

indicating that the offset for channel 4 is set to 1 mV.

CH<x>:POSition

Description

This command sets or queries the vertical position of the specified channel. The channel is specified by x, which ranges from 1 through 4. The position value is applied to the signal before it is digitized. This command is equivalent to selecting Position/Scale from the Vertical menu and either viewing or setting Position.

Increasing the position value of a waveform causes the waveform to move up, and decreasing the position value causes the waveform to move down. Position adjusts only the display position of a waveform, whether channel, math, or reference waveform. The position value determines the vertical graticule coordinate at which input signal values, equal the present offset setting for that channel, are displayed. For example, if the position for Channel 3 is set to 2.0 and the offset is set to 3.0, then input signals equal to 3.0 units are displayed 2.0 divisions above the center of the screen.

Group

Vertical

Related Commands

CH<x>:OFFSet, MATH<x>:POSition

Syntax 1

```
CH<x>:POSition <NR3>
```

Syntax 2

```
CH<x>:POSition?
```

Argument

- <NR3>

This is the position value, in divisions from the center graticule, ranging from 8 to -8 divisions.

Example 1

```
CH2:POSition 1.3E+00
```

This command positions the Channel 2 input signal 1.3 divisions above the center graticule.

Example 2

CH1:POSition?

This query might return

:CH1:POSITION -1.3000E+00

indicating that the current position of Channel 1 is 1.3 divisions below the center graticule.

CH<x>:PROBE?

Description

This query-only command returns all information concerning the probe that is attached to the specified channel. The channel is specified by x, which ranges from 1 through 4.

Group

Vertical

Related Commands

CALibrate:PROBEstate:CH<x>?, CH<x>:PROBECal?

Syntax

CH<x>:PROBE?

Example

CH2:PROBE?

This query might return

```
:CH2:PROBE:GAIN 1.0000E-01; RESISTANCE  
1.0000E+07;UNITS "V";ID:TYPE "10X" 'SERNUM-  
BER "N/A"
```

for a 10x probe, indicating that the attenuation factor for the probe attached to channel 2 is 100.0 mV (assuming that probe units are set to volts).

CH<x>:PROBECal?

Description

This query-only command returns the probe calibration state for the selected channel. The channel is specified by x, which ranges from 1 through 4. This command is equivalent to selecting Probe Cal from the Vertical menu.

Group

Vertical

Related Commands

CALibrate:PROBEstate:CH<x>?

Syntax

CH<x>:PROBECal?

Outputs

- FAIL

This signifies that the probe calibration has failed for the selected channel.

- INIT

This signifies that the probe calibration has not yet been run for the selected channel.

- PASS

This signifies the probe calibration has passed for the selected channel.

Example

CH2:PROBECAL?

This query might return

:CH2:PROBECAL PASS

indicating that the probe calibration has passed for Channel 2.

CH<x>:PRObe:GAIN?

Description

This query-only command returns the gain factor of the probe that is attached to the specified channel. The channel is specified by x, which ranges from 1 through 4. The "gain" of a probe is the output divided by the input transfer ratio. For example, a common 10x probe has a gain of 0.1.

Group

Vertical

Related Commands

CH<x>:SCALE

Syntax

CH<x>:PRObe:GAIN?

Example

CH2:PROBE:GAIN?

This query might return

```
:CH2:PROBE:GAIN 0.1000E+00
```

indicating that the attached 10x probe delivers 0.1 V to the channel 2 BNC for every 10 V applied to the probe input.

CH<x>:PRObe:ID?

Description

This query-only command returns the type and serial of the probe that is attached to the specified channel. The channel is specified by x, which ranges from 1 through 4.

Group

Vertical

Syntax

```
CH<x>:PRObe:ID?
```

Example

```
CH2:PROBE:ID?
```

This query might return

```
:CH2:PROBE:ID:TYPE "10X";SERNUMBER "N/A"
```

indicating that a passive 10x probe of unknown serial number is attached to channel 2.

CH<x>:PRObe:ID:SERnumber?

Description

This query-only command returns the serial number of the probe that is attached to the specified channel. The channel is specified by x, which ranges from 1 through 4.

Note: For Level 0 and 1 probes, the serial number will be "N/A".

Group

Vertical

Syntax

CH<x>:PRObe:ID:SERnumber?

Example

CH1:PROBE:ID:SERnumber?

This query might return

:CH1:PROBE:ID:SERNUMBER "B010289"

indicating that the serial number of the probe attached to channel 1 is B010289.

CH<x>:PRObe:ID:TYPe?

Description

This query-only command returns the type of probe that is attached to the specified channel. The channel is specified by x, which ranges from 1 through 4. Level 2 (or higher) probes supply their exact product nomenclature; for Level 0 or 1 probes, a generic 'type string' of "nnX" is returned.

Group

Vertical

Syntax

```
CH<x>:PRObe:ID:TYPe?
```

Example

```
CH1:PROBE:ID:TYPE?
```

This query might return

```
:CH1:PROBE:ID:TYPE "P6203"
```

indicating that P6203-type probe is attached to channel 1.

CH<x>:PRObe:RESistance?

Description

This query-only command returns the resistance factor of the probe that is attached to the specified channel. The channel is specified by x, which ranges from 1 through 4.

Group

Vertical

Syntax

```
CH<x>:PRObe:RESistance?
```

Example

```
CH2:PROBE:RESISTANCE?
```

This query might return

```
:CH2:PROBE:RESISTANCE 10.0000E+06
```

indicating that the input resistance of the probe attached to Channel 2 is 10.0 megOhms.

CH<x>:PRObe:SET

Description

This command sets or queries the internal signal path for the P7260 probe. One path provides a +/- 3V dynamic range with 6 GHz bandwidth and 25X attenuation. The other provides a +/- 0.75 dynamic range with 6 GHz bandwidth and 5X attenuation. Both paths will need their own set of probe cal constants. The user can select between the two paths using either this GPIB command or the push switch on the P7260 probe. The instrument will tell the probe to enable the relevant path and adjust the set of available vertical settings based on the characteristics of the path in use. The probe's current signal path selection is not kept in persistent storage. The probe will lose the selection if the user reboots the instrument or removes the probe. Also, the instrument does not store the selection in the save/recall setup operation.

Group

Vertical

Syntax 1

CH<x>:PRObe:SET?

Syntax 2

CH<x>:PRObe:SET? "ATTENUation"

Syntax 3

CH<x>:PRObe:SET {"ATTENUation 5X"|"ATTENUation 25X"}

Argument

- ATTENUATION 25X

Selects the P7260 probe's setting to +/- 3V dynamic range with 6 GHz bandwidth and 25X attenuation.

- ATTENUATION 5X

Selects the P7260 probe's setting to +/- 0.75V dynamic range with 6 GHz bandwidth and 5X attenuation.

Example 1

CH2:PROBE:SET?

This query might return

:CH2:PROBE:SET "ATTENUATION 5X"

Indicating that the P7260 probe is set to the selection for +/- 0.75 dynamic range with 6 GHz bandwidth and 5X attenuation.

Example 2

```
CH2:PROBE:SET? "ATTEN"
```

This query might return

```
:CH2:PROBE:SET "ATTENUATION 25X"
```

Indicating that the P7260 probe is set to the selection for +/- 3V dynamic range with 3 GHz bandwidth and 25X attenuation.

Example 3

```
CH2:PROBE:SET "ATTEN 25X"
```

This sets the P7260 probe to the selection for +/- 3V dynamic range with 3 GHz bandwidth and 25X attenuation.

CH<x>:PRObe:UNIts?

Description

This query-only command returns a string describing the units of measure for the probe attached to the specified channel. The channel is specified by x, which ranges from 1 through 4.

Group

Vertical

Related Commands

CH<x>:PROBEFunc:EXTUnits

Syntax

CH<x>:PRObe:UNIts?

Example

CH4:PROBE:UNITS?

This query might return

:CH4:PROBE:UNITS "V"

indicating that the units of measure for the probe attached to channel 4 are volts.

CH<x>:PROBEFunc:EXTatten

Description

This command instructs the oscilloscope when to make vertical settings (offset and scale) for the specified channel effective. The vertical settings become effective at the input of the external network or transducer (of a specified input+output transfer ratio) when the output is connected to the input of the probe attached to the channel.

The channel is specified by x, which ranges from 1 through 4. There is also a corresponding query, which returns the user-specified attenuation. Note that, as the name implies, this command deals with an attenuation factor, not a gain factor, unlike CH<x>:PROBE? (Note that this command returns a value independent of the External Attenuation). For example, if you specify a 20x attenuation factor for channel 1, then the commands return the following values (assuming that a 1x probe is presently attached, since the external attenuation is used in combination with the probe attenuation):

```
CH1:PROBE:EXTA? 20.00E+0
```

```
CH1:PROBE? 1.0E+0
```

This command is equivalent to selecting Attenuation from the Vertical command and then viewing or setting Ext Atten.

Group

Vertical

Related Commands

```
CH<x>:PROBEFunc:EXTDBatten
```

Syntax 1

```
CH<x>:PROBEFunc:EXTatten <NR3>
```

Syntax 2

```
CH<x>:PROBEFunc:EXTatten?
```

Argument

- <NR3>

This is the attenuation value, which is specified as a multiplier in the range from 1.00E-10 to 1.00E+10.

Example 1

```
CH1:PROBEFunc:EXTatten 167.00E-3
```

This command specifies an external 'gain of six' network, which is connected between the user's input signal and the input of the probe attached to channel 1.

Example 2

```
CH2:PROBEFunc:EXTatten?
```

This query might return

```
:CH2:PROBEFUNC:EXTATTEN 1.0000E+00
```

indicating that the probe attached to channel 2 is connected directly to the user's signal.

CH<x>:PROBEFunc:EXTDBatten

Description

This command instructs the oscilloscope to make vertical scale and offset settings for a specified channel effective at the input of an external network or transducer (with a specified input ÷ output transfer ratio in decibels). It is assumed that the input of the probe for the specified channel is connected to the output of this network or transducer. The channel is specified by x, which ranges from 1 through 4.

There is also a corresponding query, which returns the user-specified attenuation, in decibels. Note that 1X = 0 dB, 10X = 20 dB, 100X = 40 dB etc.

This command is equivalent to selecting Attenuation from the Vertical menu and then either viewing or setting Ext Att(dB).

Group

Vertical

Related Commands

CH<x>:PROBEFunc:EXTatten

Syntax 1

CH<x>:PROBEFunc:EXTDBatten <NR3>

Syntax 2

CH<x>:PROBEFunc:EXTDBatten?

Argument

- <NR3>

This is the attenuation value, which is specified in the range from -200.00 to 200.00.dB.

Example 1

CH3:PROBEFunc:EXTDBatten 2.5

This command specifies an external 2.5 dB attenuator on channel 3.

Example 2

CH1:PROBEFunc:EXTDBatten?

This query might return

:CH1:PROBEFUNC:EXTDBATTEEN 2.5000E+00

indicating that the attenuation for channel 1 is 2.5 dB.

CH<x>:PROBEFunc:EXTUnits

Description

This command sets the unit of measurement for the external attenuator of the specified channel. The channel is specified by x, which ranges from 1 through 4. There is also a corresponding query that returns the user-specified unit of measurement for the external attenuator. Unless these units are set to the factory default string value of "None", they become the attenuated units of measure for that channel. It is assumed that the probe connected to the specified channel is of the correct type to receive the output of the user's external transducer or network.

Group

Vertical

Related Commands

CH<x>:PROBe:UNITS

Syntax 1

CH<x>:PROBEFunc:EXTUnits <str>

Syntax 2

CH<x>:PROBEFunc:EXTUnits?

Argument

- <str>

This can contain a string of up to eight characters to indicate the attenuation unit of measurement for the specified channel. However, most oscilloscope attenuators only display the first two characters.

Example 1

```
CH4:PROBEFunc:EXTUnits "Pa"
```

This command sets the unit of measurement for the Channel 4 external attenuator.

Example 2

```
CH2:PROBEFunc:EXTUnits?
```

This query might return

```
:CH2:PROBEFUNC:EXTUNITS "Pa"
```

indicating that the Channel 2 external attenuator units of measurement are pascals.

CH<x>:SCALE

Description

This command sets or queries the vertical scale of the specified channel. The channel is specified by x, which can range from 1 through 4. Sending this command is equivalent to selecting Vertical Setup from the Vertical menu and then viewing or setting the Scale.

Each waveform has a vertical scale parameter. For a signal with constant amplitude, increasing the Scale causes the waveform to be displayed smaller. Decreasing the scale causes the waveform to be displayed larger.

Scale affects all waveforms, but affects channel waveforms differently from other waveforms:

- For channel waveforms, this setting controls the vertical size of the acquisition window as well as the display scale. The range and resolution of scale values depends on the probe attached and any other external factors you have specified.
- For reference and math waveforms, this setting controls the display only, graphically scaling these waveforms and having no affect on the acquisition hardware.

Group

Vertical

Related Commands

CH<x>OFFSet, MATH<x>:SCALE

Syntax 1

CH<x>:SCALE <NR3>

Syntax 2

CH<x>:SCALE?

Argument

- <NR3>

This is the vertical channel scale in units per division.

Example 1

CH4:SCALE 100E-03

This command sets the channel 4 scale to 100 mV per division.

Example 2

CH2:SCALE?

This query might return

:CH2:SCALE 1.0000E+00

indicating that the current scale setting of channel 2 is 1 volt per division.

CH<x>:TERmination

Description

This command sets the connected/disconnected status of a 50 Ω resistor, which may be connected between the specified channel coupled input and oscilloscope ground. The channel is specified by x, which ranges from 1 through 4. There is also a corresponding query that requests the termination parameter and translates this enumeration into one of the two float values. This command is equivalent to selecting Termination from the Vertical menu or toggling between termination values from the VERTICAL area of the front panel.

Note: The TDS7254B and TDS7404B instruments are limited to setting a 50 Ω termination value.

Group

Vertical

Syntax 1

```
CH<x>:TERmination <NR3>
```

Syntax 2

```
CH<x>:TERmination?
```

Argument

- <NR3>

This specifies the channel input resistance, which can be specified as 50 Ω .

Example 1

```
CH4:TERmination 50.0E+0
```

This command establishes 50 Ω impedance on channel 1.

Example 2

```
CH2:TERmination?
```

This query might return

```
:CH2:TERMINATION 50.0E+0
```

indicating that channel 2 is set to 50 Ω impedance.

CLEARMenu

Description

This command (no query form) clears all half-screen or lower-right control windows currently displayed on the instrument user interface.

Group

Miscellaneous

Syntax

CLEARMenu

CMDBatch

Description

This command sets or queries the state of command batching. By batching commands, database transactions can be optimized, increasing command throughput. Also, batching allows for ALL commands in an individual batch to be order independent and accomplish the same result as if the commands were coupled.

The Batch state is persistent and will be saved across power cycles, but will not be saved and recalled as part of a setup. In a setup scenario, the factory initial value is enabled.

Group

Miscellaneous

Syntax 1

```
CMDBatch {OFF|ON|<NR1>}
```

Syntax 2

```
CMDBatch?
```

Arguments

- OFF

This turns command batching off.

- ON

This turns command batching on.

- <NR1>

A 0 turns command batching off; any other value turns command batching on.

Example 1

```
CMDBatch OFF
```

This command disables command batching.

Example 2

```
CMDBatch?
```

This query might return

```
:CMDBATCH 1
```

indicating that command batching is turned on.

CURSor?

Description

This query-only command returns all of the current cursor settings.

Group

Cursor

Syntax

CURSor?

Example: CURSOR?

This query might return the following as the current cursor settings:

```
:CURSOR:STATE OFF;FUNCTION VBARS;MODE IN-  
DEPENDENT  
;SOURCE CH1;VBARS:UNITS SECONDS;POSITION1  
-8.0000E-07  
;POSITION2 8.0000E-07;:CURSOR:HBARS:POSI-  
TION1 3.0000E-01  
;POSITION2 -3.0000E-01;UNITS BASE;:CUR-  
SOR:PAIRED  
:POSITION1 -8.0000E-07;POSITION2  
8.0000E-07;UNITS BASE;  
:CURSOR:SPLIT:POSITION1 -8.0000E-07;POSI-  
TION2 8.0000E-07  
;SOURCE2 CH1;UNITS BASE
```

CURSor:FUNcTION

Description

This command selects or queries the cursor type. Sending this command is equivalent to selecting Cursor Type from the Cursor menu and then choosing from the drop-down list.

Group

Cursor

Related Commands

CURSor:STAtE, CURSor:VBArS:POSITION, CURSor:HBArS:POSITION, CURSor:SOUrce

Syntax 1

```
CURSor:FUNcTION  
{HBArS | OFF | VBArS | PAIred | SPLit | SCREEN}
```

Syntax 2

```
CURSor:FUNcTION?
```

Arguments

■ HBArS

Specifies horizontal bar cursors, which measure in vertical units.

■ OFF

Removes the cursors from the display but does not change the cursor type.

■ VBArS

Specifies vertical bar cursors, which measure in horizontal units.

■ PAIred

Specifies paired cursors that measure in both horizontal and vertical units.

■ SPLit

Specifies split cursors, which measure horizontal and vertical units across to waveforms.

■ SCREEN

Specifies both horizontal and vertical bar cursors, which measure in horizontal and vertical units specified by the Cursor 1 and Cursor 2 Sources.

Example 1

`CURSOR:FUNCTION VBARS`

This command selects vertical bar type cursors.

Example 2

`CURSOR:FUNCTION?`

This query might return `:CURSOR:FUNCTION HBARS`, indicating that the horizontal bar cursors are currently selected.

CURSor:HBArS?

Description

This query-only command returns the current settings for the horizontal bar cursors.

Group

Cursor

Syntax

```
CURSor:HBArS?
```

Example

```
CURSor:HBArS?
```

```
This command might return :CURSOR:HBARS:POSITION1 320.0000E-03;POSITION2-320.0000E-03;UNITS BASE
```

CURSor:HBARs:DELTA?

Description

This query-only command returns the vertical difference between the two horizontal bar cursors.

Group

Cursor

Syntax

```
CURSor:HBARs:DELTA?
```

Returns

<NR3>

Example

```
CURSOR:HBARS:DELTA?
```

This command might return `:CURSOR:HBARS:DELTA 5.0800E+00`, indicating that the voltage difference between the two cursors is 5.08 V.

CURSOR:HBARS:POSITION<x>

Description

This command specifies or queries the horizontal bar cursor position relative to ground, which is expressed in vertical units (usually volts). This command is the equivalent of selecting Cursor Position from the Cursors menu, selecting the H Bars Cursor Type and then viewing or editing the desired cursor position.

Group

Cursor

Syntax 1

```
CURSOR:HBARS:POSITION<x> <NRf>
```

Syntax 2

```
CURSOR:HBARS:POSITION<x>?
```

Argument

- <NR3>

Specifies the cursor position relative to ground.

Example 1

```
CURSOR:HBARS:POSITION1 25.0E-3
```

Positions one of the horizontal cursors at 25 mV.

Example 2

```
CURSOR:HBARS:POSITION2?
```

This query might return :CURSOR:HBARS:POSITION2
-64.0000E-03, indicating that one of the horizontal bar
cursors is at -64 mV.

CURSor:HBArS:UNIts?

Description

This query-only command returns the units for the horizontal bar cursors.

Group

Cursor

Syntax

```
CURSor:HBArS:UNIts?
```

Example

```
CURSor:HBArS:UNIts?
```

This query might return `:CURSOR:HBARS:UNITS BASE` indicating that the units for the horizontal bar cursors are base.

CURSor:MODe

Description

This command sets or queries whether the two cursors move together in unison or separately. This command is the equivalent of selecting Cursor Mode from the Cursor menu and then choosing from the drop-down list.

Group

Cursor

Syntax 1

```
CURSor:MODe {TRACk|INDepeNdeNt}
```

Syntax 2

```
CURSor:MODe?
```

Arguments

- TRACk

This ties the navigational functionality of these two cursors together.

For cursor 1 adjustments, this ties the movement of the two cursors together. However, cursor 2 continues to move independently of cursor 1.

- INDepeNdeNt

This frees the two cursors to be independently adjusted.

Example 1

```
CURSor:MODe TRACk
```

Specifies that the cursor positions move in unison.

Example 2

```
CURSor:MODe?
```

This query might return `:CURSOR:MODE TRACK` indicating that the two cursors move in unison.

CURSOR:SCREEN:STYLE

Description

This command sets and queries the Cursor Type in Screen mode.

Group

Cursor

Syntax 1

```
CURSOR:SCREEN:STYLE {LINE_X|LINES|X}
```

Syntax 2

```
CURSOR:SCREEN:STYLE?
```

Arguments

- LINE_X

Specifies the cursor style to be a line with superimposed X.

- LINES

Specifies the cursor style to be a line.

- X

Specifies the cursor style to be an X.

Example

```
CURSOR:SCREEN:STYLE X
```

CURSor:SOURce

Description

This command sets and queries the source for horizontal and vertical bar cursors. This command is equivalent to selecting Cursor Setup from the Cursor menu and then choosing the desired cursor source.

Group

Cursor

Syntax 1

```
CURSor:SOURce {CH<x> | MATH<x> | REF<x>}
```

Syntax 2

```
CURSor:SOURce?
```

Arguments

- CH<x>

This is an input channel waveform. The valid channel waveform range is from 1 through 4.

- MATH<x>

This is a math waveform. The valid math waveform range is from 1 through 4.

- REF<x>

This is a reference waveform. The valid reference waveform range is from 1 through 4.

Returns

Any valid waveform.

Example 1

```
CURSor:SOURce CH2
```

This command sets the cursor source to channel2.

Example 2

```
CURSor:SOURce?
```

This query might return :CURSOR:SOURCE CH2, indicating that the cursor source is channel 2.

CURSor:STATE

Description

This command sets or returns the state of the cursor. The cursor can either be turned on or off. Note that setting the cursor state does not modify the cursor type. This command is equivalent to pressing the **CURSOR** button on the front panel.

Cursor

Related Commands

CURSor:FUNcTION OFF

Syntax 1

```
CURSor:STATE {ON|OFF|<NR1>}
```

Syntax 2

```
CURSor:STATE?
```

Arguments

- ON

This activates the display cursor.

- OFF

This removes the display cursor.

- <NR1>

A 0 disables the display cursor; any other value enables the display cursor.

Example 1

```
CURSor:STATE ON
```

This command displays the cursor.

Example 2

```
CURSor:STATE?
```

This query might return :CURSOR:STATE ON, indicating that the cursor is active on the display.

CURSor:VBArS

Description

This command positions the vertical bar cursor to the position defined by DATA:START or DATA:STOP . This query returns the current vertical bar settings for horizontal position and units.

Group

Cursor

Related Commands

DATA:START, DATA:STOP

Syntax 1

```
CURSor:VBArS?
```

Syntax 2

```
CURSor:VBArS SNAP
```

Argument

- SNAP

This positions the vertical bar cursors at DATA:START and DATA:STOP.

Example 1

```
CURSor:VBArS?
```

This query might return

```
:CURSOR:VBARS:UNITS SECONDS;POSITION1  
1.0000E-06;POSITION2 9.0000E-06
```

Example 2

```
CURSor:VBArS SNAP
```

This command positions the vertical bar cursor to the position defined by DATA:START or DATA:STOP.

CURSor:VBArS:DELTA?

Description

This query-only command returns the difference between the two vertical bar cursors. The units are specified by the CURSor:VBArS:UNIts command. The position can appear in units of Base or 1/Base.

Group

Cursor

Related Commands

CURSor:VBArS:UNIts

Syntax

CURSor:VBArS:DELTA?

Returns

A floating point value with an exponent.

Example

CURSor:VBArS:DELTA?

This command might return :CURSOR:VBARS:DELTA
1.0640E+00, indicating that the time between the vertical
bar cursors is 1.064 s.

CURSor:VBArS:POSITION<x>

Description

This command sets or queries the horizontal position for both vertical bar and paired cursors. Values are with respect to trigger position or the zero reference point for the designated waveform (if horizontal units are not set to Time). Use the CURSor:VBArS:UNIts command to specify units. The position can appear in units of base or 1/base.

This command is the equivalent of selecting Cursor Setup from the Cursors menu, selecting the V Bars Cursor Type and then viewing or editing the desired cursor position.

Group

Cursor

Related Commands

CURSor:VBArS:UNIts

Syntax 1

```
CURSor:VBArS:POSITION<x> <NRf>
```

Syntax 2

```
CURSor:VBArS:POSITION<x>?
```

Argument

<NRf>

Specifies the cursor position.

Returns

A floating point value with an exponent.

Example 1

```
CURSor:VBArS:POSITION2 9.00E-6
```

Positions the cursor2 vertical bar cursors at 9 ms.

Example 2

```
CURSor:VBArS:POSITION1?
```

This command might return :CURSOR:VBARS:POSITION1 1.0000E-06, indicating the cursor1 vertical bar is at 1 ms.

CURSor:VBArS:UNIts

Description

This command sets or queries the units for the vertical bar cursors.

Group

Cursor

Syntax 1

```
CURSor:VBArS:UNIts {SECOnds|HERtz}
```

Syntax 2

```
CURSor:VBArS:UNIts?
```

Arguments

- SECOnds

Fix the units for the VBArS for time domain.

- HERtz

Fix the units for the VBArS for frequency domain.

Returns

A string for SECOnds or HERtz, depending upon current vertical bar cursor units.

Example 1

```
CURSor:VBArS:UNIts HERtz
```

Sets the units for the VBArS cursors to 1/seconds.

Example 2

```
CURSor:VBArS:UNIts?
```

This command might return :CURSOR:VBARS:UNITS SECONDS, indicating that the units for the vertical bar cursor are currently set to seconds.

CURSor:WAVEform:STYle

Description

This command sets and queries the Cursors Type in Waveform mode.

Group

Cursor

Syntax 1

```
CURSor:WAVEform:STYle {LINE_X|LINES|X}
```

Syntax 2

```
CURSor:WAVEform:STYle?
```

Arguments

- LINE_X

Specifies the cursor style to be a line with superimposed X.

- LINES

Specifies the cursor style to be a line.

- X

Specifies the cursor style to be an X.

Example

```
CURSor:WAVEform:STYle X
```

CURVe

Description

This command transfers waveform data to and from the oscilloscope. Each transferred waveform has an associated waveform preamble that contains information such as data format and scale.

The CURVe? query transfers data from the instrument. The data source is specified by the DATA:SOURce command. For conventional non-Digital Phosphorus Oscilloscope WFMDB waveforms, the first and last data points are specified by the DATA:START and DATA:STOP commands. The first and last data frames are specified by the DATA:FRAMESTART and DATA:FRAMESTOP commands. For additional information on Fast Acquisition and WFMDB waveforms, refer to the WFMOupre:PT_OR? command.

If more than one source is specified, a concatenated list of data blocks is returned, for example:

```
<binary> :CURVe #3500<500bytes of block data>#3500<500 bytes of block data>
```

```
<ascii> :CURVe x1,x2,....,x500,y1,y2,....,y500
```

When the CURVe command transfers waveform data, the data is stored in the reference memory location specified by DATA:DESTination, starting with the data point specified by DATA:START. Only one waveform can be transferred at a time; the waveform will be transferred only if it is displayed.

For FastAcq (DPO) and WFMDB waveforms, the CURVe? query data represents a pixel map of the waveform, which is described under the WFMOupre:PT_OR? query description.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the time out is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that

the query was interrupted by *ESR?. In such cases, increase the time out value and send the query again.

Group

Waveform Transfer

Related Commands

DATA:DESTination, DATA:SOURce, DATA:START, DATA:STOP, DATA:FRAMESTART, DATA:FRAMESTOP, SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMInpre, WFMInpre:BYT_Nr, WFMOutpre, WFMInpre:NR_FR?, WFMOutpre:NR_FR?

Syntax 1

```
CURVe {<Block> | <asc curve>}
```

Syntax 2

CURVe?

Arguments

■ <Block>

This is the waveform data in binary format. The waveform is formatted as: #<x><yyy><data><newline>, where

<x> is the number of y bytes. For example, if <yyy>=500, then <x>=3).

<yyy> is the number of bytes to transfer. If width is 1, then all bytes on the bus are single data points. If width is 2, then all bytes on the bus are 2-byte pairs. If width is 4, then all bytes on the bus are 4-byte pairs. Use the WFMInpre:BYT_Nr command to set the width.

<data> is the curve data.

<newline> is a single byte new line character at the end of the data.

■ <asc curve>

This is the waveform data in ASCII format. The format for ASCII data is <NR1>[,<NR1>...], where each <NR1> represents a data point.

Example 1

CURVe?

This query with ASCII encoding, start and stop of 1 and 10 respectively, and a width set to 1 might return

```
:CURVE 61,62,61,60,60,-59,-59,-58,-58,-59
```

Example 2

CURVe <Block>

This command sets the format of the waveform data, transferred to and from the oscilloscope, to binary format.

CURVEStream

Description

This command continuously transfers waveform data from the oscilloscope as it is acquired.

This command places the oscilloscope into a talk-only mode, allowing the controller to receive waveform records immediately as they are acquired.

You must use the DATA:SOURce command to specify the waveform sources; the set form of this command does the same thing as the CURVe command.

- User interface control of the oscilloscope, or other clients, during the streaming mode is not possible.
- The GPIB Controller must exit its continuous talking mode in order to terminate the query. This enables the other input sources to resume communication with the oscilloscope.
- You can use the following options to transition out of the streaming curve mode:
 - Send a device clear over the bus.
 - Send another query (a MEPE Query Interrupted error will occur, but the oscilloscope will return to normal talk/listen operation).
- While in streaming mode, two extreme conditions can occur:
 - If the oscilloscope is acquiring waveform records at a slow rate (high resolution mode), you must configure the controller for long timeout thresholds. Data will not be transferred until a complete record is acquired.
 - If the oscilloscope is acquiring waveform records at a rapid rate (low resolution mode), and the controller is not reading data from the bus at a fast enough rate, the trigger rate will be lowered to allow each waveform to be transferred sequentially.

Group

Waveform Transfer

Syntax 1

```
CURVEStream {<Block>|<asc curve>}
```

Syntax 2

CURVEStream?

Arguments

- <Block>

This is the waveform data in binary format. The waveform is formatted as: #<x><yyy><data><newline>, where

<x> is the number of y bytes. For example, if <yyy>=500, then <x>=3).

<yyy> is the number of bytes to transfer. If width is 1, then all bytes on the bus are single data points. If width is 2, then all bytes on the bus are 2-byte pairs. If width is 4, then all bytes on the bus are 4-byte pairs. Use the WFMInpre:BYT_Nr command to set the width.

<data> is the curve data.

<newline> is a single byte new line character at the end of the data.

- <asc curve>

This is the waveform data in ASCII format. The format for ASCII data is <NR1>[,<NR1>...], where each <NR1> represents a data point.

Example 1

CURVEStream?

This query might return the following for a single data source, 1 channel with a record length of 500:

```
:CURVESTREAM #41000<binary data>;<newline>
:CURVESTREAM #41000<binary data>;<newline>
```

Example 2

CURVEStream?

This query might return the following for multiple data sources, for example, 2 channels with a record length of 500:

```
:CURVESTREAM #41000<ch1 binary
data>#41000<ch2 binary data>;<newline>
```

```
:CURVESTREAM #41000<ch1 binary
data>#41000<ch2 binary data>;<newline>
```

DATA

Description

This command sets or queries the format and location of the waveform data that is transferred with the CURVe command.

Group

Waveform Transfer

Related Commands

CURVe, DATA:FRAMESTART, DATA:FRAMESTOP, DATA:START, DATA:STOP, DATA:ENCdg SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMinpre:NR_FR?, WFMinpre:NR_Pt, WFMOuppre:NR_FR?, WFMOuppre:NR_Pt?

Syntax 1

```
DATA {INIT|SNAP}
```

Syntax 2

```
DATA?
```

Argument

- INIT

This initializes the waveform data parameters to their factory defaults.

- SNAP

This sets DATA:START and DATA:STOP to match the current V Bar/Paired/Split cursor positions.

Example 1

```
DATA?
```

This query might return

```
:DATA:DESTINATION REF1:ENCDG RIBINARY;SOURCE CH1;START 1;STOP 500
```

Example 2

```
DATA INIT
```

This command initializes the waveform data parameters to their factory defaults.

DATA:DESTINATION

Description

This command sets or queries the reference memory location for storing waveform data that is transferred into the oscilloscope by the CURVe command.

Group

Waveform Transfer

Related Commands

CURVe

Syntax 1

```
DATA:DESTINATION REF<x>
```

Syntax 2

```
DATA:DESTINATION?
```

Argument

- REF<x>

This is the reference where the waveform will be stored. The reference number is specified by x, which ranges from 1 through 4.

Example 1

```
DATA:DESTINATION?
```

This query might return

```
:DATA:DESTINATION REF3
```

indicating that reference 3 is the currently selected reference memory location for waveform data.

Example 2

```
DATA:DESTINATION REF1
```

This command requests that incoming waveform data be stored in reference 1.

DATA:ENCdg

Description

This command sets or queries the format of outgoing waveform data. This command is equivalent to setting WFMOutpre:ENCdg, WFMOutpre:BN_Fmt, and WFMOutpre:BYT_Or. Setting the DATA:ENCdg value causes the corresponding WFMOutpre values to be updated and visa versa.

Note 1: Values are constrained (for outbound data) to the data format specified by DATA:SOURce.

Note 2: For Math waveforms, or reference waveforms saved from Math waveforms, the data format is floating point. The only permitted encodings for these waveforms are FPBinary or SFPbinary.

Group

Waveform Transfer

Related Commands

WFMOutpre:ENCdg, WFMOutpre:BN_Fmt,
WFMOutpre:BYT_Or

Syntax 1

```
DATA:ENCdg {ASCIi|FAStest|RIBinary|RPBinary  
|FPBinary|SRIBinary  
|SRPbinary|SFPbinary}
```

Syntax 2

```
DATA:ENCdg?
```

Argument

■ ASCIi

This specifies the ASCII representation of signed INT, FLOAT. If ASCII is the value, then :BN_Fmt and :BYT_Or are ignored.

■ FAStest

This requests that the data be sent in the fastest possible manner consistent with maintaining accuracy and is interpreted with respect to the first waveform specified in the DATA:SOURce list.

:ENCdg will always be BIN, :BYT_Or will always be MSB, but :BN_Fmt and :BYT_Nr will depend on the first DATA:SOURce waveform. :BN_Fmt will be RI unless the

waveform is internally stored as a floating point number, in which case FP format will be used.

■ RIBinary

This specifies signed integer data point representation with the most significant byte transferred first.

When :BYT_Nr is 1, the range is -128 through 127.
When :BYT_Nr is 2, the range is -32,768 through 32,767.
When :BYT_Nr is 8, then the waveform being queried has been set to Fast Acquisition mode. Center screen is 0.
The upper limit is one division above the top of the screen and the lower limit is one division below the bottom of the screen. This is the default argument.

■ RPBinary

This specifies the positive integer data-point representation, with the most significant byte transferred first.

When :BYT_Nr is 1, the range from 0 through 255. When :BYT_Nr is 2, the range is from 0 to 65,535. When :BYT_Nr is 8, then the waveform being queried has been set to Fast Acquisition mode. Center screen is 127. The upper limit is one division above the top of the screen and the lower limit is one division below the bottom of the screen.

■ FPBinary

This specifies the floating point (width = 4) data.

The range is from -3.4×10^{38} to 3.4×10^{38} . Center screen is 0. The upper limit is one division above the top of the screen and the lower limit is one division below the bottom of the screen.

The FPBinary argument is only applicable to math waveforms or ref waveforms saved from math waveforms.

■ SRIBinary

This is the same as RIBinary except that the byte order is swapped, meaning that the least significant byte is transferred first. This format is useful when transferring data to IBM compatible PCs.

■ SRPbinary

This is the same as RPBinary except that the byte order is swapped, meaning that the least significant byte is transferred first. This format is useful when transferring data to IBM compatible PCs.

- SFPbinary

This specifies floating point data in IBM format.

The SFPbinary argument only works on math waveforms or ref waveforms saved from math waveforms.

DATA and WFMOUTPRE	Parameter Settings			
	WFMOUTPRE	WFMOUTPRE	WFMOUTPRE	WFMOUTPRE
DATA:ENCdg Sett	:ENCdg	:BN Fmt	:BYT Or	:BYT NR
AScii	ASC	N/A	N/A	1, 2, 4, 8
FASText	BIN	RI/FP	MSB	1, 2, 4
RIBinary	BIN	RI	MSB	1, 2, 8
RPBinary	BIN	RP	MSB	1, 2, 8
FPBinary	BIN	FP	MSB	4
SRIBinary	BIN	RI	LSB	1, 2, 8
SRPBinary	BIN	RP	LSB	1, 2, 8
SFPbinary	BIN	FP	LSB	4

Example 1

DATA:ENCdg?

This query might return

```
:DATA:ENCDG SRPBINARY
```

for the format of the outgoing waveform data.

Example 2

```
DATA:ENGdg RPBinary
```

This command sets the data encoding format to be a positive integer where the most significant byte is transferred first.

DATA:FRAMESTART

Description

This command sets or queries the starting data frame for waveform transfer. This command allows for the transfer of partial frames of the waveform to and from the oscilloscope.

Group

Waveform Transfer

Related Commands

CURVe, DATA:FRAMESTOP, DATA:START, DATA:STOP, SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMinpre:NR_Pt, WFMinpre:NR_FR?, WFMOupre:NR_FR?, WFMOupre:NR_Pt?

Syntax 1

```
DATA:FRAMESTART <NR1>
```

Syntax 2

```
DATA:FRAMESTART?
```

Argument

- <NR1>

This is the first data frame that will be transferred, which ranges from 1 to the number of acquired frames. Data will be transferred from <NR1> to FRAMEDATA:STOP or the number of acquired frames, whichever is less. If <NR1> is greater than the number of acquired frames, then you will receive the last acquired frame. If no frames were acquired, then you will receive the first frame which contains all nulls.

DATA:FRAMESTART and DATA:FRAMESTOP are order independent. When DATA:FRAMESTOP is less than DATA:FRAMESTART, the values will be swapped internally for the CURVE? and SAVE:WAVEform queries.

Example 1

```
DATA:FRAMESTART?
```

This query might return

```
:DATA:FRAMESTART 214
```

indicating that this is the first waveform frame that will be transferred.

Example 2

`DATA:FRAMESTART 10`

This command specifies that the waveform transfer will begin with frame 10.

DATA:FRAMESTOP

Description

This command sets or queries the last data frame that will be transferred when using the CURVe? query. This command allows for the transfer of partial frames of the waveform to the controller.

Note: When using the CURVe command, DATA:FRAMESTOP is ignored and the first frame is used.

Group

Waveform Transfer

Related Commands

CURVe, DATA:FRAMESTART, DATA:START, DATA:STOP, SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMinpre:NR_Pt, WFMOuppre:NR_FR?, WFMOuppre:NR_Pt? WFMinpre:NR_FR?

Syntax 1

DATA:FRAMESTOP <NR1>

Syntax 2

DATA:FRAMESTOP?

Argument

- <NR1>

This is the last data frame that will be transferred, which ranges from 1 to the number of acquired frames. If <NR1> is greater than the number of acquired frames, then data will be transferred up to the number of acquired frames. If both DATA:FRAMESTART and DATA:FRAMESTOP are greater than the number of acquired frames, then you will receive the last acquired frame. If no frames were acquired, then you will receive the first frame which contains all nulls.

DATA:FRAMESTART and DATA:FRAMESTOP are order independent. When DATA:FRAMESTOP is less than DATA:FRAMESTART, the values will be swapped internally for the CURVe? and SAVE:WAVEform queries.

If you always want to transfer complete waveforms, simply set DATA:FRAMESTART to 1 and DATA:FRAMESTOP to the maximum number of acquired frames, or larger.

Example 1

`DATA:FRAMESTOP?`

This query might return

`:DATA:FRAMESTOP 14900`

indicating that this is the last waveform data frame that will be transferred.

Example 2

`DATA:FRAMESTOP 1500`

This command specifies that the waveform transfer will stop at data frame 1500.

DATA:SOURce

Description

This command sets or queries the location of waveform data transferred from the oscilloscope by the CURVe? query.

The source data is always transferred in a predefined order, regardless of the order specified in the command. The predefined order is CH1 through CH4, MATH1 through MATH4, and REF1 through REF4, respectively. Note that the data returned by the :WFMOutpre? query returns the preamble of the first waveform in the data source list.

Group

Waveform Transfer

Related Commands

CURVe?

Syntax 1

```
DATA:SOURce <wfm> [ ,<wfm> ]...
```

Syntax 2

```
DATA:SOURce?
```

Argument

- <wfm>

This is the location of the waveform data that will be transferred from the oscilloscope to the controller.

Example 1

```
DATA:SOURce?
```

This query might return

```
:DATA:SOURCE REF3
```

indicating the source for the waveform data that is transferred using a CURVe? query.

Example 2

```
DATA:SOURce CH1
```

This command specifies that the CH1 waveforms will be transferred in the next CURVe? query.

Example 3

`DATA:SOURce REF2, CH2, MATH1, CH1`

This command specifies that four waveforms will be transferred in the next CURVe? query. The data will transfer in the following order: CH1, CH2, MATH1, REF2.

DATA:START

Description

This command sets or queries the starting data point for waveform transfer. This command allows for the transfer of partial waveforms to and from the oscilloscope.

Group

Waveform Transfer

Related Commands

CURVE, DATA, DATA:STOP, DATA:FRAMESTART, DATA:FRAMESTOP, SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMinpre:NR_Pt, WFMOupre:NR_Pt, WFMinpre:NR_FR?, WFMOupre:NR_FR?

Syntax 1

```
DATA:START <NR1>
```

Syntax 2

```
DATA:START?
```

Argument

- <NR1>

This is the first data point that will be transferred, which ranges from 1 to the record length. Data will be transferred from <NR1> to DATA:STOP or the record length, whichever is less. If <NR1> is greater than the record length, the last data point in the record is transferred.

DATA:START and DATA:STOP are order independent. When DATA:STOP is greater than DATA:START, the values will be swapped internally for the CURVE? query.

Example 1

```
DATA:START?
```

This query might return

```
:DATA:START 214
```

indicating that this is the first waveform data point that will be transferred.

Example 2

```
DATA:START 10
```

This command specifies that the waveform transfer will begin with data point 10.

DATA:STOP

Description

This command sets or queries the last data point that will be transferred when using the CURVE? query. This command allows for the transfer of partial waveforms to the controller.

Changes to the record length value are not automatically reflected in the data:stop value. As record length is varied, the data:stop value must be explicitly changed to ensure the entire record is transmitted. In other words, curve results will not automatically and correctly reflect increases in record length if the distance from data:start to data:stop stays smaller than the increased record length.

Note: When using the CURVE command, DATA:STOP is ignored and WFMinpre:NR_Pt is used.

Group

Waveform Transfer

Related Commands

CURVE, DATA, DATA:START, DATA:FRAMESTART, DATA:FRAMESTOP, SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMinpre:NR_Pt, WFMOutpre:NR_Pt, WFMinpre:NR_FR?, WFMOutpre:NR_FR?

Syntax 1

DATA:STOP <NR1>

Syntax 2

DATA:STOP?

Argument

- <NR1>

This is the last data point that will be transferred, which ranges from 1 to the record length. If <NR1> is greater than the record length, then data will be transferred up to the record length. If both DATA:START and DATA:STOP are greater than the record length, the last data point in the record is returned.

DATA:START and DATA:STOP are order independent. When DATA:STOP is less than DATA:START, the values will be swapped internally for the CURVE? query.

If you always want to transfer complete waveforms, just set DATA:START to 1 and DATA:STOP to the maximum record length, or larger.

Example 1

`DATA:STOP?`

This query might return

`:DATA:STOP 14900`

indicating that this is the last waveform data point that will be transferred.

Example 2

`DATA:STOP 15000`

This command specifies that the waveform transfer will stop at data point 15000.

DATE

Description

This command sets or queries the date that the oscilloscope can display. This command is equivalent to selecting Set Date & Time from the Utilities menu and then setting the fields in the Date group box.

Group

Miscellaneous

Related Commands

TIME

Syntax 1

DATE <Qstring>

Syntax 2

DATE?

Argument

- <QString>

This is a date in the form "yyyy-mm-dd" where yyyy refers to a four-digit year number, mm refers to a two-digit month number from 01 to 12, and dd refers to a two-digit day number in the month.

Example 1

```
DATE "2000-01-24"
```

This command specifies that the date is set to January 24, 2000.

Example 2

```
DATE?
```

This query might return

```
:DATE 2000-01-24
```

indicating the current date is set to January 24, 2000.

DELEte:SETUp

Description

This command (no query form) changes the setup to reference the factory setup instead of the specific user setup slot. The content of the setup slot is unchanged but the data will no longer be accessible to you. This command is equivalent to selecting Instrument Setup from the File menu, choosing the Delete tab and then clicking the Delete button.

Group

Save and Recall

Related Commands

*RCL?, RECall:SETUp, *RST, *SAV, SAVE:SETUp, *SDS

Syntax

```
DELEte:SETUp {ALL|<NR1>}
```

Arguments

- ALL

This specifies to delete all the stored setups.

- <NR1>

This specifies a setup storage location to delete. The setup in that position, which ranges from 1 through 10, will now refer to the Factory setup when recalled. Using an out-of-range value causes an error.

Example

```
DELEte:SETUp ALL
```

This command removes all stored setups. All ten storage locations are initialized to the factory default setup.

DELEte:WAVEform

Description

This command (no query form) deletes one or all stored reference waveforms from memory. This command is equivalent to selecting Reference Waveforms from the File menu and choosing Reference Setup from the drop-down list (when deleting individual reference waveforms) or selecting Delete All Refs from the File menu (when deleting all reference waveforms).

Group

Save and Recall

Related Commands

RECAI:WAVEform, Save:WAVEform

Syntax

```
DELEte:WAVEform {ALL | REF<x>}
```

Arguments

- ALL

This specifies to delete all the stored reference waveforms.

- REF<x>

This specifies to delete one of the reference memory locations. Reference memory location values range from 1 through 4.

Example 1

```
DELEte:WAVEform ALL
```

This command removes all waveforms stored in reference memory.

Example 2

```
DELEte:WAVEform REF2
```

This command removes the waveform stored at REF2.

DESE

Description

This command sets and queries the bits in the Device Event Status Enable Register (DESER). The DESER is the mask that determines whether events are reported to the Standard Event Status Register (SESR), and entered into the Event Queue. For a more detailed discussion of the use of these registers, see Registers.

Group

Status and Error

Related Commands

*CLS, *ESE, *ESR?, *EVENT?, EVMsg?, *SRE, *STB?

Syntax 1

```
DESE <NR1>
```

Syntax 2

```
DESE?
```

Argument

- <NR1>

The binary bits of the DESER are set according to this value, which ranges from 1 through 255. For example, DESE 209 sets the DESER to the binary value 11010001 (that is, the most significant bit in the register is set to 1, the next most significant bit to 1, the next bit to 0, etc.).

The power-on default for DESER is all bits set if *PSC is 1. If *PSC is 0, the DESER maintains the previous power cycle value through the current power cycle.

Note: Setting the DESER and ESER to the same value allows only those codes to be entered into the Event Queue and summarized on the ESB bit (bit 5) of the Status Byte Register. Use the *ESE command to set the ESER.

Example 1

```
DESE 209
```

This sets the DESER to binary 11010001, which enables the PON, URQ, EXE and OPC bits.

Example 2

DESE?

This query might return

:DESE 186,

showing that the DESER contains the binary value
10111010.

DIAG:CONTROL:HALT

Description

This command (no query form) determines whether the next execution of diagnostics will stop on the first diagnostic failure that occurs or will execute the selected set of diagnostic functions. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then enabling Halt on Fail.

Group

Diagnostics

Related Commands

DIAG:CONTROL:LOOP

Syntax

```
DIAG:CONTROL:HALT (ON|OFF|<NR1>}
```

Arguments

- ON

This enables the halt function, causing the execution of diagnostics to halt at the first diagnostic failure that occurs.

- OFF

This disables the halt function, allowing the oscilloscope to execute the entire set of diagnostics before halting, even if diagnostic failure occurs.

- <NR1>

A 0 enables the halt function; any other value disables the halt function.

Example

```
DIAG:CONTROL:HALT ON
```

This command enables the halt function, causing the execution of diagnostics to halt at the first diagnostic failure that occurs.

DIAG:CONTROL:LOOP

Description

This command (no query form) determines whether the next execution of diagnostics executes once or continuously loops on the selected set of diagnostics (assuming the halt control is set to off using the DIAG:CONTROL:HALT command or that the halt control is set to ON but no failures occur). This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then enabling Loop Control.

Group

Diagnostics

Related Commands

DIAG:CONTROL:HALT, DIAG:STOP

Syntax

```
DIAG:CONTROL:LOOP {ON|OFF|<NR1>}
```

Arguments

- ON

This enables the loop function, causing the execution of diagnostics to continuously loop.

- OFF

This disables the loop function, causing the oscilloscope to execute the entire set of diagnostics once and then halt.

- <NR1>

A 0 enables the loop function; any other value disables the loop function.

Example

```
DIAG:CONTROL:LOOP ON
```

This command enables the loop function.

DIAG:EXECUTE

Description

This command (no query form) starts the execution of the currently selected set of diagnostics. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then pressing Run.

Group

Diagnostics

Related Commands

DIAG:STATE

Syntax

DIAG:EXECUTE

Example

DIAG:EXECUTE

This command starts the execution of the entire set of diagnostics.

DIAG:ITEM?

Description

This query-only command returns all the data associated with a particular item from the RSI menu (level of test hierarchy). This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the diagnostics settings.

Group

Diagnostics

Related Commands

DIAG:ITEM:FAILURES?, DIAG:ITEM:NAME?,
DIAG:ITEM:RESULT?, DIAG:ITEM:SUBITEMS?

Syntax

DIAG:ITEM?

Arguments

- <NR1>

This sets the index item about which data will be returned, which ranges from 0 through 14.

Example

DIAG:ITEM? 2

This query might return :DIAG:ITEM "TRIGGER", "FAIL", 2, 7

DIAG:ITEM:FAILURES?

Description

This query-only command returns the total number of failures. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Related Commands

DIAG:ITEM?, DIAG:ITEM:NAME?, DIAG:ITEM:RESULT?,
DIAG:ITEM:SUBITEMS?

Syntax

DIAG:ITEM:FAILURES?

Example

DIAG:ITEM:FAILURES?

This query might return

```
:DIAG:ITEM:FAILURES 2
```

indicating the number of failures.

DIAG:ITEM:NAME?

Description

This query-only command returns the name of the selected menu item. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Subsystem, Area and Test settings.

Group

Diagnostics

Related Commands

DIAG:ITEM?, DIAG:ITEM:FAILURES?, DIAG:ITEM:RESULT?, DIAG:ITEM:SUBITEMS?

Syntax

DIAG:ITEM:NAME?

Example

DIAG:ITEM:NAME?

This query might return

```
:DIAG:ITEM:NAME "TRIGGER"
```

indicating that the selected menu item is Trigger.

DIAG:ITEM:RESULT?

Description

This query-only command returns the result from the last execution of the item. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Related Commands

DIAG:ITEM?, DIAG:ITEM:NAME?, DIAG:ITEM:FAILURES?, DIAG:ITEM:SUBITEMS?

Syntax

```
DIAG:ITEM:RESULT?
```

Example 1

```
DIAG:ITEM:RESULT?
```

This query might return `:DIAG:ITEM:RESULT "PASS"`, indicating that the item passed during the last execution.

Example 2

```
DIAG:ITEM:RESULT?
```

This query might return `:DIAG:ITEM:RESULT "FAIL"`, indicating that the item failed during the last execution.

Example 3

```
DIAG:ITEM:RESULT?
```

This query might return

```
:DIAG:ITEM:RESULT "*****"
```

indicating that the item was not run.

DIAG:ITEM:SUBITEMS?

Description

This query-only command returns the number of sub-items associated with the item. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu, choosing the Subsystem, Area or Test setting and then reviewing the resulting sub-items.

Group

Diagnostics

Related Commands

DIAG:ITEM?, DIAG:ITEM:NAME?, DIAG:ITEM:FAILURES?,
DIAG:ITEM:RESULT?

Syntax

```
DIAG:ITEM:SUBITEMS?
```

Example

```
DIAG:ITEM:SUBITEMS?
```

This query might return

```
:DIAG:ITEM:SUBITEMS 15
```

indicating that there are 15 sub-items associated with the selected item.

DIAG:LEVEL

Description

This command sets or returns the currently selected level of diagnostic test hierarchy. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Syntax 1

```
DIAG:LEVEL (SUBSYS|AREA|TEST}
```

Syntax 2

```
DIAG:LEVEL?
```

Arguments

- SUBSYS

This sets diagnostic testing to the subsystem level.

- AREA

This sets diagnostic testing to the area level.

- TEST

This sets diagnostic testing to the test level.

Example 1

```
DIAG:LEVEL AREA
```

This command sets the level of diagnostic test hierarchy to Area.

Example 2

```
DIAG:LEVEL?
```

This query might return

```
:DIAG:LEVEL:SUBSYS
```

indicating that the current level of diagnostic test hierarchy is Subsys.

DIAG:LOOPS?

Description

This query-only command returns the number of times that the selected diagnostics set was completed during the last diagnostic execution. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Loop Count.

Group

Diagnostics

Syntax

DIAG:LOOPS?

Example

DIAG:LOOPS?

This query might return

:DIAG:LOOPS 5

indicating that the selected set of diagnostics was completed five times during the last diagnostic execution.

DIAG:NAME?

Description

This query-only command returns the names of the subsystem, area, and test of the current diagnostic test.

Group

Diagnostics

Related Commands

DIAG:NAME:AREA?, DIAG:NAME:SUBSYS?,
DIAG:NAME:TEST?

Syntax

DIAG:NAME?

Example

DIAG:NAME?

This query might return

```
:DIAG:NAME:SUBSYS "Acquisition";AREA  
"Memory";TEST"diag Data Format"
```

indicating the subsystem name, area name, and test name of the currently selected diagnostic test.

DIAG:NAME:AREA?

Description

This query-only command returns the selected area of the current diagnostic test. There are three levels of diagnostic test hierarchy: subsystem, area and test. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Related Commands

DIAG:NAME?, DIAG:NAME:SUBSYS?, DIAG:NAME:TEST?

Syntax

DIAG:NAME:AREA?

Example

DIAG:NAME:AREA?

This query might return

```
:DIAG:NAME:AREA "Memory"
```

indicating the selected area name of the current diagnostic test.

DIAG:NAME:SUBSYS?

Description

This query-only command returns the subsystem of the current diagnostic test. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Related Commands

DIAG:NAME?, DIAG:NAME:AREA?, DIAG:NAME:TEST?

Syntax

DIAG:NAME:SUBSYS?

Example

DIAG:NAME:SUBSYS?

This query might return

```
:DIAG:NAME:SUBSYS "Acquisition"
```

indicating the subsystem name of the current diagnostic test.

DIAG:NAME:TEST?

Description

This query-only command returns the name of the current diagnostic test. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Related Commands

DIAG:NAME?, DIAG:NAME:AREA?, DIAG:NAME:SUBSYS?

Syntax

DIAG:NAME:TEST?

Example

DIAG:NAME:TEST?

This query might return

```
:DIAG:NAME:TEST "diagDataFormatConf"
```

indicating the test name of the current diagnostic test.

DIAG:NUMITEMS?

Description

This query-only command returns the number of items on the currently selected level of test hierarchy, which ranges from 0 through 15. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Syntax

```
DIAG:NUMITEMS?
```

Example

```
DIAG:NUMITEMS?
```

This query might return

```
:DIAG:NUMITEMS 7
```

indicating the number of items on the currently selected level of test hierarchy.

DIAG:RESULts?

Description

This query-only command returns an abbreviated status about the results of the last diagnostic (or test) execution. For a more explanatory status message, use the DIAG:RESULts:VERBose? query. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Related Commands

DIAG:RESULts:VERBose?

Syntax

DIAG:RESULts?

Example 1

DIAG:RESULts?

This query might return

```
:DIAG:RESULTS "135"
```

indicating the specific failure code of the first test failure that occurred.

Example 2

DIAG:RESULts?

This query might return

```
:DIAG:RESULTS "Pass"
```

indicating that the last diagnostic test passed.

Example 3

DIAG:RESULts?

This query might return

```
:DIAG:RESULTS "*****"
```

indicating that the diagnostic test was not run.

DIAG:RESULts:VERBoSe?

Description

This query-only command returns a more explanatory message about the results of the last diagnostic (or test) execution than the DIAG:RESULts? query. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Related Commands

DIAG:RESULts?

Syntax

DIAG:RESULts:VERBoSe?

Example

DIAG:RESULts:VERBoSe?

This query might return

```
:DIAG:RESULTS:VERBOSE "DIAG ABORTED, ERROR  
15 in A2D-Dmux 1 interconnects"
```

DIAG:SElect:ALL

Description

This command (no query form) selects all available diagnostics. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then choosing ALL from the Subsystem, Area and Test pull-down lists.

Group

Diagnostics

Related Commands

DIAG:NAME:AREA?, DIAG:NAME:SUBSYS?,
DIAG:NAME:TEST?, DIAG:EXECUTE

Syntax

```
DIAG:SElect:ALL ALL
```

Argument

- ALL

Selects all diagnostic options.

Example

```
DIAG:SElect:ALL ALL
```

This command selects all available diagnostics.

DIAG:SElect:AREA

Description

This command (no query form) selects one of the available areas. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then selecting an Area from the pull-down list.

Note: This command should be executed only if DIAG:LEVEL is currently set to AREA.

Group

Diagnostics

Related Commands

DIAG:SElect:SUBSYS, DIAG:SElect:TEST, DIAG:SElect:LAST

Syntax

```
DIAG:SElect:AREA <NR1>
```

Argument

- <NR1>

This selects an area by number, which can range from 1 through 15 (as specified by DIAG:NUMITEMS).

Example

```
DIAG:SElect:AREA 1
```

This command selects Area 1 as the area to be diagnosed.

DIAG:SElect:LAST

Description

This command (no query form) selects one or more diagnostic menu items to be executed via the DIAG:EXECUTE command. If you specify DIAG:LEVEL SUBSYS, then menu items come from this diagnostic level and are limited to the value returned by the DIAG:NUMITEMS? query. For example, specifying DIAG:SElect:SUBSYS 2 and DIAG:NUMITEMS 5 indicates that diagnostics will start from subsystem 2 and that you can specify a range from 2 through 5 for DIAG:SElect:LAST.

If you enter:

DIAG:SElect:LAST 2, only subsystem 2 will be executed.

DIAG:SElect:LAST 4, subsystems 2 *through* 4 will be executed.

Group

Diagnostics

Related Commands

DIAG:EXECUTE, DIAG:LEVEL, DIAG:NAME:AREA?, DIAG:NAME:SUBSYS?, DIAG:NAME:TEST?, DIAG:NUMITEMS?, DIAG:SElect:SUBSYS

Syntax

```
DIAG:SElect:LAST <NR1>
```

Arguments

- <NR1>

This selects an integer that identifies the number of the last item that will be executed when the DIAG:EXECUTE command is run.

Example

```
DIAG:SElect:LAST 2
```

This command specifies that (based on the previous example) only subsystem 2 will be executed.

DIAG:SElect:SUBSYS

Description

This command (no query form) selects one of the available subsystems. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then choosing a Subsystem from the drop-down list.

Note: This command should be executed only if DIAG:LEVEL is currently set to SUBSYS.

Group

Diagnostics

Related Commands

DIAG:SElect:AREA, DIAG:SElect:TEST, DIAG:SElect:LAST

Syntax

```
DIAG:SElect:SUBSYS <NR1>
```

Argument

- <NR1>

This selects a subsystem by number, which can range from 1 through 15 (as specified by DIAG:NUMITEMS).

Example

```
DIAG:SElect:SUBSYS
```

This command selects Subsystem 1 as the subsystem to be diagnosed.

DIAG:SElect:TEST

Description

This command (no query form) selects one of the available tests. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then choosing a Test from the drop-down list.

Note: This command should be executed only if DIAG:LEVEL is currently set to TEST.

Group

Diagnostics

Related Commands

DIAG:SElect:AREA, DIAG:SElect:SUBSYS, DIAG:SElect:LAST

Syntax

```
DIAG:SElect:TEST <NR1>
```

Arguments

- <NR1>

This selects a test by number, which can range from 1 through 15 (as specified by DIAG:NUMITEMS).

Example

```
DIAG:SElect:TEST 1
```

This command selects Test 1 as the test to be executed.

DIAG:STATE

Description

This command (no query form) changes the oscilloscope operating state. Depending on the argument, diagnostics capabilities are either turned on or off. This command is equivalent to opening the DIAG:STATE dialog (ON) or closing it (OFF).

Group

Diagnostics

Related Commands

TEST:STOP

Syntax

```
DIAG:STATE {ON|OFF}
```

Arguments

- ON

This puts the instrument into the state in which diagnostics can be run. This argument is thrown automatically if either the DIAG:EXECUTE or DIAG:TEST commands are executed.

- OFF

This disables diagnostics capabilities and returns the oscilloscope to a normal operating state.

Example

```
DIAG:STATE OFF
```

This command turns off diagnostics capabilities and returns the instrument to a normal operating state.

DIAG:STOP

Description

This command (no query form) causes diagnostics (or test) execution to terminate at the end of the next low-level test. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then clicking Abort.

Group

Diagnostics

Related Commands

TEST:STOP

Syntax

DIAG:STOP

Example

DIAG:STOP

This command terminates diagnostics (or test) execution at the end of the next low-level test.

DISplay?

Description

This query-only command returns the current display settings. This command is equivalent to selecting Display Set-up from the Display menu and then viewing the contents of each tab.

Group

Display Control

Syntax

DISplay?

Example

DISPLAY?

This query might return

```
:DISPLAY:CLOCK 1
;COLOR:PALETTE NORMAL;MATHCOLOR DEFAULT
;REFCOLOR DEFAULT;:DISPLAY:FILTER
SINX;FORMAT YT
;GRATICULE FULL;INTENSITY:WAVEFORM
6.0000E+01
;AUTOBRIGHT 1;SCRENSAVER 1;SCRENSAVERDE-
LAY 28800,
:DISPLAY:PERSISTENCE OFF;STYLE VEC-
TORS;TRIGBAR SHORT
;TRIGT 0;VARPERSIST 5.0000E-01
```

DISplay:CLOCK

Description

This command sets or queries the display of the date and time. This is equivalent to selecting Display Date & Time from the Display menu. The query form of this command returns an ON (1) or an OFF (0).

Group

Display Control

Syntax 1

```
DISplay:CLOCK {ON|OFF|<NR1>}
```

Syntax 2

```
DISplay:CLOCK?
```

Arguments

- ON

This enables display of date and time.

- OFF

This disables display of date and time.

- <NR1>

A 0 disables display of date and time; any other value enables display of date and time.

Example 1

```
DISPLAY:CLOCK ON
```

This command enables display of date and time.

Example 2

```
DISPLAY:CLOCK?
```

This query might return

```
:DISPLAY:CLOCK 1
```

indicating that the display of date and time is currently enabled.

DISplay:COLOr?

Description

This query-only command returns the settings from the PALETTE, MATHCOLOR and REFCOLOR commands. This is equivalent to selecting Colors from the Display menu.

Group

Display Control

Syntax

DISplay:COLOr?

Arguments

None.

Example

DISPLAY:COLOr?

This query might return

```
:DISPLAY:COLOR:PALETTE NORMAL;MATHCOLOR  
DEFAULT;REFCOLOR INHERIT
```

DISplay:COLOr:MATHCOLOr

Description

This command sets or queries the color to be used for math traces, either in the palette nominal Math color, or according to the color of the source waveform. This command is equivalent to selecting Display Setup from the Display menu and then choosing the Colors tab.

Group

Display Control

Related Commands

DISplay:COLOr:PALETTE

Syntax 1

```
DISplay:COLOr:MATHCOLOr {DEFAULT|INHERIT}
```

Syntax 2

```
DISplay:COLOr:MATHCOLOr?
```

Arguments

- DEFAULT

This sets color math traces in nominal palette math color, which is red.

- INHERIT

This sets color math traces in the source waveform color. Math waveforms are drawn in the same color as their primary source waveform.

Example 1

```
DISPLAY:COLOr:MATHCOLOr DEFAULT
```

This command sets the color math traces in the nominal palette math color.

Example 2

```
DISPLAY:COLOr:MATHCOLOr?
```

This query might return

```
:DISPLAY:COLOR:MATHCOLOR INHERIT
```

indicating that the primary source waveform color is used for each math trace.

DISplay:COLOr:PALEtte

Description

This command sets or queries the palette to be used for trace display. This is equivalent to selecting Colors from the Display menu and choosing from the displayed color palette options.

This command is equivalent to setting both DISplay:COLOr:PALEtte:IMAGEView and DISplay:COLOr:PALEtte:RECORDView to the given palette type. For example, executing DISplay:COLOr:PALEtte:NORMal is equivalent to executing DISplay:COLOr:PALEtte:IMAGEView NORMal and DISplay:COLOr:PALEtte:RECORDView NORMal.

Group

Display Control

Related Commands

DISplay:COLOr, DISplay:COLOr:MATHCOLOr, DISplay:COLOr:PALEtte:IMAGEView, DISplay:COLOr:PALEtte:RECORDView, DISplay:COLOr:REFCOLOr

Syntax 1

```
DISplay:COLOr:PALEtte {NORMAL|MONO-  
GREEN|MONOGRAY|TEMPERature|SPECTral|USER}
```

Syntax 2

```
DISplay:COLOr:PALEtte?
```

Arguments

- NORMal

This colors traces according to their channel. This is the default color palette.

- MONOGREEN

This colors traces green, emulating a traditional oscilloscope color palette.

- MONOGRAY

This colors traces gray, emulating a monochrome TDS instrument.

- TEMPERature

All traces share a multi-colored palette, where "intensity" is represented by hue; blue for least frequently hit, red for most frequently hit.

- SPECTral

All traces share a multi-colored palette, where "intensity" is represented by hue; red for least frequently hit, blue for most frequently hit.

- USER

All traces share a user-defined palette.

Example 1

```
DISPLAY:COLOR:PALETTE MONOGRAY
```

This command sets the palette for all display types.

Example 2

```
DISPLAY:COLOR:PALETTE?
```

This query might return

```
:DISPLAY:COLOR:PALETTE:IMAGEVIEW TEMPERA-  
TURE;RECORDVIEW NORMAL;USER:GRATICULE  
165,50,15;CH1 180,50,100;CH2  
300,50,100;CH3 60,50,100;CH4  
240,50,100;REF1 0,90,0;REF2 0,90,100;REF3  
60,90,100;REF4 240,90,100;MATH1  
160,50,100;MATH2 40,60,100;MATH3  
120,60,100;MATH4 195,50,100;HISTOGRAM  
320,50,100;CARET 150,50,100;MASK  
0,25,75;MASKHIGHLIGHT 140,50,100
```

indicating the various color palette selections.

DISplay:COLOr:PALEtte:IMAGEView

Description

This command sets or queries the color palette for all image view (DPO and WfmDB) traces.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:IMAGEView {NORMAL|MONOGREEN|MONOGRAY|TEMPERATURE|SPECTRAL|USER}
```

Syntax 2

```
DISplay:COLOr:PALEtte:IMAGEView?
```

Arguments

- NORMAL

This colors traces according to their channel. This is the default color palette.

- MONOGREEN

This colors traces green, emulating a traditional oscilloscope color palette.

- MONOGRAY

This colors traces gray, emulating a monochrome TDS instrument.

- TEMPERature

All traces share a multi-colored palette, where "intensity" is represented by hue; blue for least frequently hit, red for most frequently hit.

- SPECTral

All traces share a multi-colored palette, where "intensity" is represented by hue; red for least frequently hit, blue for most frequently hit.

- USER

All traces share a user-defined palette.

Example 1

```
DISPLAY:COLOR:PALETTE:IMAGEVIEW MONOGRAY
```

This command sets the palette for all display types.

Example 2

DISPLAY:COLOR:PALETTE:IMAGEVIEW?

This query might return

:DISPLAY:COLOR:PALETTE:IMAGEVIEW NORMAL

indicating that the Normal color palette is currently selected, which colors traces according to their channel.

DISplay:COLOr:PALEtte:RECORDView

Description

This command sets or queries the color palette for all record view (non image view) waveforms.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:RECORDView {NORMAL|MONOGREEN|MONOGRAY|TEMPERATURE|SPECTRAL|USER}
```

Syntax 2

```
DISplay:COLOr:PALEtte:RECORDView?
```

Arguments

- NORMAL

This colors traces according to their channel. This is the default color palette.

- MONOGREEN

This colors traces green, emulating a traditional oscilloscope color palette.

- MONOGRAY

This colors traces gray, emulating a monochrome TDS instrument.

- TEMPERATURE

All traces share a multi-colored palette, where "intensity" is represented by hue; blue for least frequently hit, red for most frequently hit.

- SPECTRAL

All traces share a multi-colored palette, where "intensity" is represented by hue; red for least frequently hit, blue for most frequently hit.

- USER

All traces share a user-defined palette.

Example 1

```
DISPLAY:COLOR:PALETTE:RECORDVIEW MONOGRAY
```

This command sets the palette for all display types.

Example 2

DISPLAY:COLOR:PALETTE:RECORDVIEW?

This query might return

:DISPLAY:COLOR:PALETTE:RECORDVIEW NORMAL

indicating that the Normal color palette is currently selected, which colors traces according to their channel.

DISplay:COLOR:PALEtte:USER?

Description

This command queries the color palette for group settings. It outputs settings from the DISPlay CARET, CH<x>, GRATICULE, HISTOGRAM, MASK, MASKHIGHLIGHT, MATH<x>, and REF<x> commands. .

Group

Display Control

Syntax

```
DISplay:COLOr:PALEtte:USER?
```

Example 1

```
DISplay:COLOR:PALETTE:USER?
```

This query might return

```
:DISPLAY:COLOR:PALETTE:USER:GRATICULE  
165,50,15;CH1 180,50,100;CH2 300  
50,100;CH3 60,50,100;CH4 240,50,100;REF1  
0,90,0;REF2 0,90,100;REF3 60,90,100;REF4  
240,90,100;MATH1 160,50,100;MATH2  
40,60,100;MATH3 120,60,100;MATH4  
195,50,100;HISTOGRAM 320,50,100;CARET  
150,50,100;MASK 0,25,75;MASKHIGHLIGHT  
140,50,100
```

DISplay:COLOR:PALEtte:USER:CARet

Description

This command sets or queries the user palette caret color. It assigns the HUE, Light, Saturation (HLS) triplet to be used for the caret color for the user palette.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:USER:CARet  
<NR1>,<NR1>,<NR1>
```

Syntax 2

```
DISplay:COLOr:PALEtte:USER:CARet?
```

Arguments

- <NR1>

Hue. Range of 0 to 360.

- <NR1>

Lightness. Range of 0 to 100.

- <NR1>

Saturation. Range of 0 to 100.

Example 1

```
DISPLAY:COLOR:PALETTE:USER:CARET 330, 75,  
100
```

This command sets the caret color hue to 330, the lightness to 75, and the saturation to 100.

Example 2

```
DISPLAY:COLOR:PALETTE:USER:CARET?
```

DISplay:COLOr:PALEtte:USER:CH<x>

Description

This command sets or queries the hue-lightness-saturation (HLS) triplet to be used for the specified channel color for the user palette.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:USER:CH {1|2|3|4}  
<NR1>,<NR1>,<NR1>
```

Syntax 2

```
DISplay:COLOr:PALEtte:USER:CH {1|2|3|4}?
```

Arguments

- <NR1>

Hue. Range of 0 to 360.

- <NR1>

Lightness. Range of 0 to 100.

- <NR1>

Saturation. Range of 0 to 100.

Example 1

```
DISPLAY:COLOR:PALETTE:USER:CH1 330, 75,  
100
```

This command sets the specified channel color for the user palette to 330, 75, 100.

Example 2

```
DISPLAY:COLOR:PALETTE:USER:CH1?
```

DISplay:COLOr:PALEtte:USER:GRAticule

Description

This command sets or queries the user palette graticule color. It assigns the hue–lightness–saturation triplet to be used for the histogram color for the user palette.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:USER:GRAticule  
<NR1>,<NR1>,<NR1>
```

Syntax 2

```
DISplay:COLOr:PALEtte:USER:GRAticule?
```

Arguments

- <NR1>

Hue. Range of 0 to 360.

- <NR1>

Lightness. Range of 0 to 100.

- <NR1>

Saturation. Range of 0 to 100.

Example 1

```
DISPLAY:COLOR:PALETTE:USER:GRATICULE 330,  
75, 100
```

This command sets the graticule color for the user palette.

Example 2

```
DISPLAY:COLOR:PALETTE:USER:GRATICULE?
```

DISplay:COLOr:PALEtte:USER:HIStoqram

Description

This command sets or queries the user palette histogram color. It assigns the hue-saturation-lightness (HLS) triplet to be used for the histogram color for the user palette.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:USER:HIStoqram  
<NR1>,<NR1>,<NR1>
```

Syntax 2

```
DISplay:COLOr:PALEtte:USER:HIStoqram?
```

Arguments

- <NR1>

Hue. Range of 0 to 360.

- <NR1>

Lightness. Range of 0 to 100.

- <NR1>

Saturation. Range of 0 to 100.

Example 1

```
DISPLAY:COLOR:PALETTE:USER:HISTOGRAM 330,  
75, 100
```

This command sets the histogram color for the user palette.

Example 2

```
DISPLAY:COLOR:PALETTE:USER:HISTOGRAM?
```

DISplay:COLOr:PALEtte:USER:MASK

Description

This command sets or queries the user palette mask color. It assigns the hue–lightness–saturation (HLS) color for the mask color for the user palette.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:USER:MASK  
<NR1>,<NR1>,<NR1>
```

Syntax 2

```
DISplay:COLOr:PALEtte:USER:MASK?
```

Arguments

- <NR1>

Hue. Range of 0 to 360.

- <NR1>

Lightness. Range of 0 to 100.

- <NR1>

Saturation. Range of 0 to 100.

Example 1

```
DISPLAY:COLOR:PALETTE:USER:MASK 330, 75,  
100
```

This command sets the mask color for the user palette.

Example 2

```
DISPLAY:COLOR:PALETTE:USER:MASK?
```

DISplay:COLOR:PALETTE:USER:MASKHighlight

Description

This command sets or queries the user palette mask hits color. It assigns the hue–lightness–saturation (HLS) triplet to be used for the mask highlight (mask hits) color for the user palette.

Group

Display Control

Syntax 1

```
DISplay:COLOR:PALETTE:USER:MASKHighlight  
<NR1>,<NR1>,<NR1>
```

Syntax 2

```
DISplay:COLOR:PALETTE:USER:MASK:Highlight?
```

Arguments

- <NR1>

Hue. Range of 0 to 360.

- <NR1>

Lightness. Range of 0 to 100.

- <NR1>

Saturation. Range of 0 to 100.

Example 1

```
DISPLAY:COLOR:PALETTE:USER:MASK:HIGHLIGHT  
330, 75, 100
```

This command sets the mask highlight (mask hits) color for the user palette.

Example 2

```
DISPLAY:COLOR:PALETTE:USER:MASK:HIGHLIGHT?
```

DISplay:COLOr:PALEtte:USER:MATH<x>

Description

This command sets or queries the user palette math colors. It assigns the hue–lightness–saturation (HLS) triplet to be used for the specified math color for the user palette.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:USER:MASK  
<NR1>,<NR1>,<NR1>
```

Syntax 2

```
DISplay:COLOr:PALEtte:USER:MASK?
```

Arguments

- <NR1>

Hue. Range of 0 to 360.

- <NR1>

Lightness. Range of 0 to 100.

- <NR1>

Saturation. Range of 0 to 100.

Example 1

```
DISPLAY:COLOR:PALETTE:USER:MASK 330, 75,  
100
```

This command sets the mask color for the user palette.

Example 2

```
DISPLAY:COLOR:PALETTE:USER:MASK?
```

DISplay:COLOr:PALEtte:USER:REF<x>

Description

This command sets or queries the user palette reference colors. It assigns the hue–lightness–saturation (HLS) triplet to be used for the specified reference color for the user palette.

Group

Display Control

Syntax 1

```
DISplay:COLOr:PALEtte:USER:REF {1|2|3|4}  
<NR1>,<NR1>,<NR1>
```

Syntax 2

```
DISplay:COLOr:PALEtte:USER:REF?
```

Arguments

- <NR1>

Hue. Range of 0 to 360.

- <NR1>

Lightness. Range of 0 to 100.

- <NR1>

Saturation. Range of 0 to 100.

Example 1

```
DISPLAY:COLOR:PALETTE:USER:REF1 330, 75,  
100
```

This command sets the mask color for the user palette.

Example 2

```
DISPLAY:COLOR:PALETTE:USER:REF1?
```

DISplay:COLOr:REFCOLOr

Description

This command sets or queries the color to be used for reference traces, either in the palette nominal REF color or according to the color of the source waveform. This command is equivalent to selecting Display Setup from the Display menu and then choosing the Colors tab.

Group

Display Control

Related Commands

DISplay:COLOr:PALETTE

Syntax 1

```
DISplay:COLOr:REFCOLOr {DEFAUlt|INHERIT}
```

Syntax 2

```
DISplay:COLOr:REFCOLOr?
```

Arguments

- DEFAUlt

This assigns color reference traces in nominal palette reference color, which is off-white.

- INHERIT

This assigns color reference traces in the source waveform color.

Example 1

```
DISPLAY:COLOr:REFCOLOr DEFAULT
```

This command assigns color reference traces in nominal palette reference color, which is off-white.

Example 2

```
DISPLAY:COLOr:REFCOLOr?
```

This query might return

```
:DISPLAY:COLOr:REFCOLOr INHERIT
```

indicating that source waveform color is used for reference traces.

DISplay:DESKew

Description

This command sets or queries the deskew feature.

Group

Display Control

Related Commands

CH<x>:DESKew

Syntax 1

DISplay:DESKew {OFF | ON | <NR1>}

Syntax 2

DISplay:DESKew?

Arguments

- OFF

Enables the sub-sample deskew feature; provides the most accurate measurement with some data rate loss.

- ON

Disables the sub-sample deskew feature; provides the fastest data rate with occasional incorrect acquisition.

- <NR1>

Zero enables the deskew feature; a nonzero disables the deskew feature.

Example 1

```
DISPLAY:DESKew ON
```

This command disables the deskew feature.

Example 2

```
DISPLAY:DESKew?
```

This query might return

```
:DISPLAY:DESKEW 1
```

indicating that the deskew feature is disabled.

DISplay:FILTer

Description

This command sets or queries the type of interpolation to use for the display. Filtering only applies to normal-mode acquisition. Neither DPO nor WFMDB ever interpolates. The DISplay:FILTer command also provides selection for acquisition interpolation type. This command is equivalent to selecting Waveform Interpolation from the Display menu.

Group

Display Control

Syntax 1

```
DISplay:FILTer {LINEAr|SINX}
```

Syntax 2

```
DISplay:FILTer?
```

Arguments

- LINEAr

This specifies linear interpolation where acquired points are connected with straight lines.

- SINX

This specifies $\sin(x)/x$ interpolation where acquired points are fit to a curve.

Example 1

```
DISplay:FILTer SINX
```

This command specifies sine-curve interpolation when magnifying waveforms.

Example 2

```
DISplay:FILTer?
```

This query might return

```
:DISPLAY:FILTER LINEAR
```

indicating that straight-line interpolation is specified for magnifying waveforms.

DISplay:FORMat

Description

This command sets or queries the display format. This command is equivalent to setting Format from the Display menu.

Group

Display Control

Syntax 1

```
DISplay:FORMat {YT|XY|XYZ}
```

Syntax 2

```
DISplay:FORMat?
```

Arguments

- YT

This sets the display to a voltage versus time format and is the normal mode.

- XY

This displays one waveform against another. The source pairs that make up an XY trace are predefined and are listed in the following table. Selecting one source causes its corresponding source to be implicitly selected, producing a single trace from the two input waveforms.

- XYZ

This combines channel 1 and channel 2 for X and Y coordinates and uses channel 3 to provide the intensity value for the sample. XYZ groups channels 1, 2 and 3 to form a single trace. Other channel, math, and reference waveforms are turned off. Selecting XYZ forces a switch to FASTAcq mode.

XY and XYZ Format Pairs				
Display Mode	Format	X-Axis Source	Y-Axis Source	Intensity Source
Normal	XY	Ch 1 Ch 3 Ref 1 Ref 3	Ch 2 Ch 4 Ref 2 Ref 4	
Fast Acquisition	XY	Ch 1 Ch 3*	Ch 2 Ch 4*	
	XYZ	Ch 1	Ch 2	CH 3

* Available on TDS7054B and TDS7104B oscilloscopes operating in Fast Acquisition mode.

Example 1

DISPLAY:FORMAT YT

This command selects a voltage versus time format for the display.

Example 2

DISPLAY:FORMAT?

This query might return

:DISPLAY:FORMAT XYZ

for the display format.

DISplay:GRATicule

Description

This command selects or queries the type of graticule that is displayed. This command is equivalent to selecting Graticule Style from the Display menu.

Group

Display Control

Syntax 1

```
DISplay:GRAticule {CROSS-  
Hair | FRame | FULL | GRId}
```

Syntax 2

```
DISplay:GRAticule?
```

Arguments

- CROSSHair

This specifies a frame and cross hairs.

- FRame

This specifies just a frame.

- FULL

This specifies a frame, a grid and cross hairs.

- GRId

This specifies a frame and grid only.

Example 1

```
DISPLAY:GRATICULE FRAME
```

This command sets the graticule type to display the frame only.

Example 2

```
DISPLAY:GRATICULE?
```

This query might return

```
:DISPLAY:GRATICULE FULL
```

indicating that all graticule elements are selected.

DISplay:INTENSITy?

Description

This query-only command returns the waveform saturation level, autobright state and screensaver settings. This command is equivalent to selecting Display Setup from the Display menu and then choosing the Appearance tab.

Group

Display Control

Related Commands

DISplay:INTENSITy:AUTOBright, DISplay:INTENSITy:WAVEform, DISplay:INTENSITy:SCRENSAVER, DISplay:INTENSITy:SCRENSAVERDELAY

Syntax

```
DISplay:INTENSITy?
```

Arguments

None

Example

```
DISPLAY:INTENSITY?
```

This query might return

```
:DISPLAY:INTENSITY:WAVEFORM 7.5000E+01;  
AUTOBRIGHT 1; SCRENSAVER 1; SCRENSAVER-  
DELAY 120
```

DISplay:INTENSITy:AUTOBright

Description

This command enables automatic, ongoing adjustment of the hi-res to 4-bit-display mapping transform for converting DPO, WFMDb, or RecordVu images to display images. The query form returns an ON (1) or OFF (0). This command is equivalent to selecting Display Setup from the Display menu and then choosing the Appearance tab.

Group

Display Control

Related Commands

DISplay:INTENSITy:WAVEform

Syntax 1

```
DISplay:INTENSITy:AUTOBright  
{OFF | ON | <NR1>}
```

Syntax 2

```
DISplay:INTENSITy:AUTOBright?
```

Arguments

- OFF

This allows the system to use the manually set waveform intensity value against an absolute scale.

- ON

This allows the system to adjust settings with the user-defined waveform intensity value in concert with the maximum and minimum non-zero acquired pixel intensity.

- <NR1>

A 0 allows the system to use the manually set waveform intensity value against an absolute scale; any other value allows the system to adjust settings.

Example 1

```
DISPLAY:INTENSITY:AUTOBRIGHT ON
```

This command allows the system to adjust settings using the waveform intensity value and the pixel intensity found in the acquired waveform.

Example 2

This query might return

```
:DISPLAY:INTENSITY:AUTOBRIGHT 1
```

indicating that system adjustment of waveform intensity is enabled.

DISplay:INTENSITY:SCRENSAVER

Description

This command sets and queries the screen protection features of the underlying MS Windows operating system. When enabled, a delay timer (set in seconds by the DISplay:INTENSITY:SCRENSAVERDELAY command) begins counting down. When this screensaver delay times out, the screen low-power mode engages. This causes the LCD backlight to switch off and clears both waveform and text displays. Any control (front panel, mouse or keyboard) or touch screen activity resets the delay timer and restores normal scope display. This command is equivalent to selecting Screen Saver Enabled from the Display menu.

Group

Display Control

Related Commands

DISplay:INTENSITY:SCRENSAVERDELAY

Syntax 1

```
DISplay:INTENSITY:SCRENSAVER  
{OFF|ON|<NR1>}
```

Syntax 2

```
DISplay:INTENSITY:SCRENSAVER?
```

Arguments

- OFF

This disables the screen saver feature.

- ON

This enables the protection after the specified screen saver delay seconds of control activity have passed.

- <NR1>

A 0 disables the screensaver feature; a 1 enables the screen saver protection features.

Example 1

```
DISPLAY:INTENSITY:SCRENSAVER OFF
```

This command disables the screen saver feature.

Example 2

DISPLAY:INTENSITY:SCRENSAVER?

This query might return

:DISPLAY:INTENSITY:SCRENSAVER 0

indicating that the screensaver feature is disabled.

DISplay:INTENSITY:SCRENSAVERDELAY

Description

This command sets or queries the timeout of the screen protection features of the display system. When enabled (after the specified screen saver delay seconds of control activity and when the screensaver feature is enabled) the instrument activates the screensaver feature. Normal scope displays are restored and the delay timer is reset upon any control activity. The oscilloscope continues to acquire and process data normally while in screensaver mode; only the display is disabled. This command is equivalent to selecting Display Setup from the Display menu and then choosing the Objects tab.

Group

Display Control

Related Commands

DISplay:INTENSITY:SCRENSAVER

Syntax 1

```
DISplay:INTENSITY:SCRENSAVERDELAY {<NR1>}
```

Syntax 2

```
DISplay:INTENSITY:SCRENSAVERDELAY?
```

Arguments

- <NR1>

This sets the screensaver timeout, which ranges from 30 through 28800 seconds.

Example 1

```
DISPLAY:INTENSITY:SCRENSAVERDELAY 120
```

This command sets the screen saver delay feature to activate after 120 seconds of control inactivity.

Example 2

```
DISPLAY:INTENSITY:SCRENSAVERDELAY?
```

This query might return

```
:DISPLAY:INTENSITY:SCRENSAVERDELAY 300
```

indicating that the screen saver delay feature is set to 300 seconds of control inactivity.

DISplay:INTENSITy:WAVEform

Description

This command sets or queries the intensity saturation of the waveforms. This command is equivalent adjusting the **INTENSITY** knob on the front panel.

This command is also equivalent to setting both DISplay:INTENSITy:WAVEform:IMAGEView and DISplay:INTENSITy:WAVEform:RECORDView to the same given intensity. For example, executing DISplay:INTENSITy:WAVEform 100 is equivalent to executing DISplay:INTENSITy:WAVEform:IMAGEView 100 and DISplay:INTENSITy:WAVEform:RECORDView 100.

Group

Display Control

Related Commands

DISplay:INTENSITy:AUTOBright, DISplay:INTENSITy:WAVEform:IMAGEView, DISplay:INTENSITy:WAVEform:RECORDView

Syntax 1

```
DISplay:INTENSITy:WAVEform <NR1>
```

Syntax 2

```
DISplay:INTENSITy:WAVEform?
```

Arguments

- <NR1>

Waveform intensity ranges from 10 to 100 percent.

Example 1

```
DISPLAY:INTENSITy:WAVEFORM 100
```

This command sets the waveform intensity to 100 percent.

Example 2

```
DISPLAY:INTENSITy:WAVEFORM?
```

This query might return

```
:DISPLAY:INTENSITy:WAVEFORM:IMAGEVIEW  
75.0000;RECORDVIEW 75.0000
```

indicating the waveform intensities.

DISplay:INTENSITY:WAVEform:IMAGEView

Description

This command sets or queries the saturation level for image view waveforms.

Group

Display Control

Related Commands

DISplay:INTENSITY:WAVEform:RECORDView, DIS-
play:INTENSITY:AUTOBright

Syntax 1

```
DISplay:INTENSITY:WAVEform:IMAGEView <NR1>
```

Syntax 2

```
DISplay:INTENSITY:WAVEform:IMAGEView?
```

Arguments

- <NR1>

This is the waveform saturation and ranges from 10 to 100 percent.

Example 1

```
DISPLAY:INTENSITY:WAVEFORM:IMAGEVIEW 100
```

This command sets the image view waveform saturation to 100 percent.

Example 2

```
DISPLAY:INTENSITY:WAVEFORM:IMAGEVIEW?
```

This query might return

```
:DISPLAY:INTENSITY:WAVEFORM:IMAGEVIEW  
60.0000
```

indicating that the saturation level of the image view waveforms is currently set to 60 percent.

DISplay:INTENSITY:WAVEform:RECORDView

Description

This command sets or queries the saturation level for record view waveforms.

Group

Display Control

Related Commands

DISplay:INTENSITY:WAVEform:IMAGEView, DISplay:INTENSITY:AUTOBright

Syntax 1

```
DISplay:INTENSITY:WAVEform:RECORDView  
<NR1>
```

Syntax 2

```
DISplay:INTENSITY:WAVEform:RECORDView?
```

Arguments

- <NR1>

This is the waveform saturation and ranges from 10 to 100 percent.

Example 1

```
DISPLAY:INTENSITY:WAVEFORM:RECORDVIEW 100
```

This command sets the record view waveform saturation to 100 percent.

Example 2

```
DISPLAY:INTENSITY:WAVEFORM:RECORDVIEW?
```

This query might return

```
:DISPLAY:INTENSITY:WAVEFORM:RECORDVIEW  
60.0000
```

indicating that the saturation of the record view waveforms is currently set to 60 percent.

DISplay:PERsistence

Description

This command sets or queries the persistence aspect of the display. This affects the display only and is equivalent to selecting Display Persistence from the Display menu.

Persistence can be affected by the Autofit feature. Autofit comes on instruments that support mask testing (the TDS7000 Series with option SM or the CSA7000 Series). When enabled, Autofit moves the waveform vertically and/or horizontally to reduce the number of hits within a given mask. If infinite or variable persistence is enabled, these movements will clear any persisted data. If Autofit makes frequent adjustments, there can be very little or even no persisted data displayed.

Group

Display Control

Related Commands

DISplay:VARPersist

Syntax 1

```
DISplay:PERsistence {OFF|INFPersist|VAR-  
Persist}
```

Syntax 2

```
DISplay:PERsistence?
```

Arguments

- OFF

This disables the persistence aspect of the display.

- INFPersist

This sets a display mode where any pixels, once touched by samples, remain set until cleared by a mode change.

- VARPersist

This sets a display mode where set pixels are gradually dimmed.

Example 1

```
DISplay:PERsistence VARPersist
```

This command sets the persistence aspect of the display to fade set pixels according to the time set in the DISplay:VARPersist command.

Example 2

DISPLAY:PERSistence?

This query might return

```
:DISPLAY:PERSISTENCE OFF
```

indicating that the persistence aspect of the display is disabled.

DISplay:PERSiStence:RESEt

Description

This command controls the clearing of persistence data.

Group

Display Control

Related Commands

DISplay:PERSiStence, DISplay:VARPersist

Syntax

```
DISplay:PERSiStence:RESEt
```

Example 1

```
DISPLAY: PERSISTENCE: RESET
```

This command clears the display of persistence data.

DISplay:SCREENTExt?

Description

This command returns all screen text settings.

Group

Display Control

Related Commands

DISplay:SCREENTExt:STATE, DISplay:SCREENTExt:LABel<x>?, DISplay:SCREENTExt:LABel<x>:NAME, DISplay:SCREENTExt:LABel<x>:XPOS, DISplay:SCREENTExt:LABel<x>:YPOS

Syntax

```
DISplay:SCREENTExt?
```

Example 1

```
DISPLAY:SCREENTEXT?
```

This command might reply:

```
:DISPLAY:SCREENTEXT:STATE 0;LABEL1:NAME  
"";XPOS 100;YPOS 5;:DISPLAY:SCREENTEXT:LA-  
BEL2:NAME "";XPOS 100;YPOS 20;:DIS-  
PLAY:SCREENTEXT:LABEL3:NAME "";;XPOS  
100;YPOS 35;:DISPLAY:SCREENTEXT:LA-  
BEL4:NAME "";XPOS 100;YPOS 50;:DIS-  
PLAY:SCREENTEXT:LABEL5:NAME "";XPOS  
100;YPOS 65;:DISPLAY:SCREENTEXT:LA-  
BEL6:NAME "";XPOS 100;YPOS 80;:DIS-  
PLAY:SCREENTEXT:LABEL7:NAME "";XPOS  
100;YPOS 95;:DISPLAY:SCREENTEXT:LA-  
BEL8:NAME "";XPOS 100;YPOS 110
```

DISplay:SCREENTExt:LABel<x>?

Description

This command returns the screen text setting for a given label. Specifically, it provides the name, horizontal position (XPOS), and vertical position (YPOS).

Group

Display Control

Related Commands

DISplay:SCREENTExt?, DISplay:SCREENTExt:STATE, DISplay:SCREENTExt:LABel<x>:NAME, DISplay:SCREENTExt:LABel<x>:XPOS, DISplay:SCREENTExt:LABel<x>:YPOS

Syntax

```
DISplay:SCREENTExt:LABel<x>?
```

Example

```
DISPLAY:SCREENTEXT:LABEL3?
```

Might return:

```
:DISPLAY:SCREENTEXT:LABEL3:NAME "THIS IS  
SCREEN TEXT"; :XPOS 100; :YPOS 35
```

DISplay:SCREExt:LABel<x>:NAME

Description

This command sets the text to be displayed for a given label.

Group

Display Control

Related Commands

DISplay:SCREExt?, DISplay:SCREExt:STATE, DISplay:SCREExt:LABel<x>?, DISplay:SCREExt:LABel<x>:XPOS, DISplay:SCREExt:LABel<x>:YPOS

Syntax

```
DISplay:SCREExt:LABel<x>:NAME <String>
```

Arguments

- <String>

The text to be displayed for a given label.

Example

```
DISPLAY:SCRETEXT:LABEL1:NAME "THIS IS  
SCREEN TEXT"
```

DISplay:SCREExt:LABel<x>:XPOS

Description

This command sets the horizontal position of a given label.

Group

Display Control

Related Commands

DISplay:SCREExt?, DISplay:SCREExt:STATE, DISplay:SCREExt:LABel<x>?, DISplay:SCREExt:LABel<x>:NAME, DISplay:SCREExt:LABel<x>:YPOS

Syntax

```
DISplay:SCREExt:LABel<x>:XPOS <NR1>
```

Arguments

- <NR1>

The horizontal position of the label.

Example

```
DISPLAY:SCRETEXT:LABEL3:XPOS 100
```

DISplay:SCREExt:LABel<x>:YPOS

Description

This command sets the vertical position of a given label.

Group

Display Control

Related Commands

DISplay:SCREExt?, DISplay:SCREExt:STATE,
DISplay:SCREExt:LABel<x>?,
DISplay:SCREExt:LABel<x>:NAME, DIS-
play:SCREExt:LABel<x>:XPOS

Syntax

```
DISplay:SCREExt:LABel<x>:YPOS <NR1>
```

Arguments

- <NR1>

The vertical position of the label.

Example

```
DISPLAY:SCRETEXT:LABEL3:YPOS 100
```

DISplay:SCREExt:STATE

Description

This command controls the display of screen text.

Group

Display Control

Related Commands

DISplay:SCREExt?, DISplay:SCREExt:LABel<x>?, DISplay:SCREExt:LABel<x>:NAME, DISplay:SCREExt:LABel<x>:XPOS, DISplay:SCREExt:LABel<x>:YPOS

Syntax

```
DISplay:SCREExt:STATE {ON|OFF|<NR1>}
```

Arguments

- ON

This turns on the display of screen text.

- OFF

This turns off the display of screen text.

- <NR1>

A 0 disables screen text; any other value enables screen text.

Example 1

```
DISPLAY:SCRETEXT:STATE ON
```

DISplay:STYLE

Description

This command sets or queries how the data is displayed for normal and FastAcq modes. This command is equivalent to setting Display Style from the Display menu.

Group

Display Control

Syntax 1

```
DISplay:STYLE {VECTors | DOTs | INTENSIFied}
```

Syntax 2

```
DISplay:STYLE?
```

Arguments

- VECTors

This connects adjacent data points. New points immediately replace old ones.

- DOTs

This displays individual data points. New points immediately replace old ones.

- INTENSIFied

This causes the display to show interpolated samples with dark spaces (Only the "real" samples are displayed).

When FastAcq mode is enabled, intensified samples display as dots. However, turning off FastAcq mode causes the display style to snap back to its previous setting.

Example 1

```
DISPLAY:STYLE VECTors
```

This command sets the display to connect adjacent data points.

Example 2

```
DISPLAY:STYLE?
```

This query might return

```
:DISPLAY:STYLE INTENSIFIED
```

indicating that interpolated samples are not displayed.

DISplay:TRIGBar

Description

This command controls or queries the display of the trigger-level indicator bar/s. Indicator bars show where the trigger voltage level is set.

The oscilloscope will only display the bar if the associated trigger source is also displayed. If both a main and a delayed trigger are displayed, then two bars will appear. One will accompany each source. If a logic trigger is selected, then multiple bars may appear. One will show the upper threshold and one will show the lower threshold. This command is equivalent to selecting Display Setup from the Display menu and then choosing the Objects tab.

Group

Display Control

Syntax 1

```
DISplay:TRIGBar {OFF|SHORT|LONG}
```

Syntax 2

```
DISplay:TRIGBar?
```

Arguments

- OFF

This removes the trigger indicator bar from the display.

- SHORT

This displays, as the indicator, a short arrow at the right side of the graticule for each displayed trigger signal.

- LONG

This displays, as the indicator, a horizontal line across the width of the graticule for each displayed trigger signal.

Example 1

```
DISplay:TRIGBar LONG
```

This command sets the display to show a long trigger indicator bar (or bars).

Example 2

DISplay:TRIGBar?

This query might return

:DISPLAY:TRIGBAR OFF

indicating that the indicator bar is removed from the display.

DISplay:TRIGT

Description

This command controls or queries the display of the trigger point indicator. This is equivalent to selecting Objects from the Display menu. The query form returns an ON (1) or an OFF (0).

Group

Display Control

Syntax 1

```
DISplay:TRIGT {ON|OFF|<NR1>}
```

Syntax 2

```
DISplay:TRIGT?
```

Arguments

- ON

This displays a trigger indicator on each of the displayed waveforms. The trigger indicator is in reverse video for the selected waveform.

- OFF

This removes the trigger indicator from the display.

- <NR1>

A 0 removes the trigger indicator from the display; a 1 displays a trigger indicator on each of the displayed waveforms.

Example 1

```
DISPLAY:TRIGT ON
```

This command sets the display to show a trigger indicators.

Example 2

```
DISPLAY:TRIGT?
```

This query might return

```
:DISPLAY:TRIGT 0
```

indicating that the trigger indicator is removed from the display.

DISplay:VARPersist

Description

This command sets or queries persistence decay time, which is the approximate decay time for a freshly struck persistence sample. This command is equivalent to selecting Display Setup from the Display menu, selecting the Appearance tab and then entering the desired Persist Time.

Persistence can be affected by the Autofit feature. Autofit comes on instruments that support mask testing (the TDS7000 Series with option SM or the CSA7000 Series). When enabled, Autofit moves the waveform vertically and/or horizontally to reduce the number of hits within a given mask. If infinite or variable persistence is enabled, these movements will clear any persisted data. If Autofit makes frequent adjustments, there can be very little or even no persisted data displayed.

Group

Display Control

Related Commands

DISplay:PERSistence

Syntax 1

```
DISplay:VARPersist <NR3>
```

Syntax 2

```
DISplay:VARPersist?
```

Arguments

- <NR3>

This indicates the persistence decay time setting in numeric seconds. The persistence time ranges from 32 ms to 10 s.

Example 1

```
DISPLAY:VARPersist 5E-1
```

This command sets the persistence decay time to 500 ms.

Example 2

DISPLAY:VARPersist?

This query might return

:DISPLAY:VARPERSIST 5.0000E-01

indicating that persistence decay time is currently set to 500 ms.

EVENT?

Description

This query-only command returns an event code from the Event Queue that provides information about the results of the last *ESR? read. EVENT? also removes the returned value from the Event Queue.

Group

Status and Error

Related Commands

ALLEv?, *CLS, DESE, *ESE, *ESR?, EVMsg?, *SRE, *STB?

Syntax

EVENT?

Example

EVENT?

This query might return

```
:EVENT 110
```

showing that there was an error in a command header.

EVMsg?

Description

This query-only command removes a single event code from the Event Queue that is associated with the results of the last *ESR? read and returns the event code along with an explanatory message. For more information, see Event Handling.

Group

Status and Error

Related Commands

ALLEv?

*CLS, DESE, *ESE, *ESR?, EVENT?, *SRE, *STB?

Syntax

EVMsg?

Returns

The event code and message in the following format:

```
<Event Code><Comma><Qstring>[<Event Code><Comma><Qstring>...]  
<Qstring>::= <Message>;[<Command>]
```

where <Command> is the command that caused the error and may be returned when a command error is detected by the oscilloscope. As much of the command will be returned as possible without exceeding the 60 character limit of the <Message> and <Command> string combined. The command string is right-justified.

Example

EVMsg?

This query might return

```
:EVMMSG 110, "Command header error"
```

EVQty?

Description

This query-only command returns the number of event codes that are in the Event Queue. This is useful when using the ALLEv? query, since it lets you know exactly how many events will be returned.

Group

Status and Error

Related Commands

ALLEv?, EVENT?, EVMsg?

Syntax

EVQty?

Example

EVQty?

This query might return

```
:EVQTY 3
```

indicating the number of event codes in the Event Queue.

EXPort

Description

This command sends a copy of the waveform to the file path specified by EXPort:FILENAME. The EXPort query returns image format and file information.

Group

Export

Syntax 1

```
EXPort START
```

Syntax 2

```
Export?
```

Arguments

- START

Initiates export.

Example 1

```
EXPort START
```

This command initiates export to the file specified file by EXPort:FILENAME.

Example 2

```
EXPort?
```

This query returns the export formatting and file path. The query might return:

```
:EXPORT:FILENAME "C:\TekScope\Images\hcFILE.BMP"
```

EXPort:FILENAME

Description

This command sets or returns the file/path that will be sent export data on the next EXPort command.

Group

Export

Related Commands

HARDCopy

Syntax 1

```
EXPort:FILENAME <file path>
```

Syntax 2

```
EXPort:FILENAME?
```

Argument

- <file path>

This specifies that the hard copy is sent to the named file. <file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/ <filename>.<drive> and one or more <dir>s are optional. <filename> stands for a filename of up to 128 characters. If you don't specify the path with <drive> and one or more <dir>s, the default location is "C:\TekScope\Images". While filename extensions are not required, they are highly recommended.

File separators may be either forward slashes ("/") or back slashes ("\").

Example 1

```
EXPort:FILENAME "TEK.IBM"
```

This command sets TEK.IBM as the selected file name. The next export will be sent to this file in the default hard copy directory.

Example 2

```
EXPort:FILENAME?
```

This query might return

```
EXPORT:FILENAME "C:\TekScope\Images\TEK.IBM" as the selected file.
```

EXPort:FORMat

Description

This command sets or returns the image format for exporting waveforms to a file (File > Export Setup > Images > Data Format).

Group

Export

Syntax 1

```
EXPort:FORMat {BMP|JPEG|PNG}
```

Syntax 2

```
EXPort:FORMat?
```

Arguments

- BMP

Specifies BMP image format.

- JPEG

Specifies JPEG image format.

- PNG

Specifies PNG image format.

Example 1

```
EXPort:FORMat BMP
```

This command sets the export format to BMP.

EXPort:IMAGe

Description

This command sets or returns the inksaver mode for exporting waveforms (File > Export Setup > Images > Image).

Group

Export

Syntax 1

```
EXPort:IMAGe {NORMal | INKSaver | ENHANCEDwfm}
```

Syntax 2

```
EXPort:IMAGe?
```

Arguments

- NORMal

Preserves the screen background when printing waveforms.

- INKSaver

Changes the screen background to white. This saves ink when printing waveforms.

- ENHANCEDwfm

Prints (color) screen images with colors designed to print on a white background.

Example 1

```
EXPort:IMAGe NORMal
```

This command sets the export mode to Normal.

EXPort:PALEtte

Description

This command sets or returns the export color palette (File > Export Setup > Images > Palette).

Group

Export

Syntax 1

```
EXPort:PALEtte {COLOr|GRAYscale|BLACKAND-  
White}
```

Syntax 2

```
EXPort:PALEtte?
```

Arguments

- COLOr

Hardcopy output is color.

- GRAYscale

Hardcopy output is gray.

- BLACKANDwhite

Hardcopy output is black and white.

Example 1

```
EXPort:PALEtte
```

This command sets the export color palette to Color.

EXPort:READOuts

Description

This command sets or returns the area on the screen where the readouts appear for export.

Group

Export

Syntax 1

```
EXPort:READOuts {OFFGRaticule|ONGRaticule}
```

Syntax 2

```
EXPort:READOuts?
```

Arguments

- OFFGRaticule

Displays the readouts for export below the graticule area.

- ONGRaticule

Displays the readouts for export as they appear on the instrument screen.

Example 1

```
EXPort:READOuts OFFGRaticule
```

This command sets the readouts to appear below the graticule area at the time of export.

EXPort:VIEW

Description

This command sets or returns the area of the screen to be exported (File > Export Setup > Images > View).

Group

Export

Syntax 1

```
EXPort:VIEW {FULLSCREEN|GRaticule}
```

Syntax 2

```
EXPort:VIEW?
```

Arguments

- FULLSCREEN

Displays both the graticule and menu areas of the screen

- CRaticule

Displays only the graticule area of the screen.

Example 1

```
EXPort:VIEW FULLSCREEN
```

This command sets the exported view area to Full Screen.

FACTory

Description

This command (no query form) resets the oscilloscope to its factory default settings. This command is equivalent to pressing the **DEFAULT SETUP** button located on the front panel.

Group

Save and Recall

Related Commands

*PSC, *RCL?, RECall:SETUp, *RST, *SAV, SAVE:SETUp

Syntax

FACTory

Arguments

None

Example

FACTory

This command resets the oscilloscope settings to factory defaults:

- Clears the Event Status Enable Register.
- Clears the Service request Enable Register.
- Sets the Device Event Status Enable Register to 2555.
- Sets the Power On Status Clear Flag to TRUE
- Purges all defined aliases.
- Enables all Command Headers.
- Sets the macro defined by *DDT to a "zero-length field."
- Clears the pending operation flag and associated operations.

This command does not reset the following:

- The state of the GPIB (IEEE Std 488.1-1987) interface.
- The selected GPIB address.

-
- Calibration data that affects device specifications.
 - Protected user data.
 - Stored settings.
 - The current password (if implemented).

FASTAcq?

Description

This query-only command returns the state of Fast Acquisition. This command is equivalent to the **FASTACQ** button on the front panel of a Digital Phosphorus Oscilloscope (DPO).

Group

Acquisition

Syntax

FASTAcq?

Example

FASTAcq?

This query might return the following string for the current acquisition parameters:

```
:FASTACQ:STATE 0
```

FASTAcq:STATE

Description

This command sets or queries the state of Fast Acquisition. It is equivalent to pressing the **FASTACQ** button on the front panel of a Digital Phosphorus Oscilloscope (DPO).

In this mode, the data is a continuous stream that cannot be triggered (there are no waveform records generated). Intensity information for XY and XYZ displays is supported.

For Record View XY displays, the data can be triggered and individual X and Y waveform records are available. This mode is particularly useful for studying phase relationships.

Note: FastACQ mode does not support histograms when using XY and XYZ display formats.

Group

Acquisition

Syntax 1

```
FASTAcq:STATE {ON|OFF|<NR1>}
```

Syntax 2

```
FASTAcq:STATE?
```

Arguments

- ON

This enables Fast Acquisition mode.

- OFF

This disables Fast Acquisition mode.

- <NR1>

A 0 disables Fast Acquisition mode; any other value enables Fast Acquisition mode.

Example 1

```
FASTAcq:STATE ON
```

This command enables the FastAcq mode.

Example 2

This query might return

```
:FASTACQ:STATE 1
```

Indicating that the Fast Acquisition state is currently active.

FILESystem?

Description

This query-only command returns the directory listing of the current working directory. It works the same as the FILESystem:DIR query.

Group

File System

Related Commands

FILESystem:COpy, FILESystem:DELEte, FILESystem:DIR?, FILESystem:PRInt, FILESystem:READFile, FILESystem:REName, FILESystem:WRITEFile

Syntax

```
FILESystem?
```

Arguments

None.

Example

```
FILESYSTEM?
```

This query might return

```
:FILESYSTEM:DIR, ".","..",myFile.txt","my-Waveform.wfm".
```

FILESystem:COpy

Description

This command (no query form) copies a named file or files to a new file. The new file may be in a totally separate directory than the old file. You can only copy one file at a time using this command. Wild card characters are not allowed.

Group

File System

Related Commands

FILESystem:DELEte

Syntax

```
FILESystem:COpy {<source file path>,<destination file path>}
```

Arguments

- <file path>

This is a quoted string that defines the file name and path. If the file path is within the current working directory, you need only specify the file name.

Input the file path using the form

<drive>/<dir>/<filename>.<drive> and one or more <dir>s are optional. <filename> stands for a filename of up to 128 characters. While filename extensions are not required, they are highly recommended. You can also use the inclusive filename *.* in the source file path to copy all files.

File separators may be either forward slashes ("/") or back slashes ("\").

- <source file path>

This is a quoted string that defines the source file path. If the file path is within the current working directory, you need only specify the file name.

Input the directory using the form

<drive>/<dir>/<directory name>.<drive> and one or more <dir>s are optional. <directory name> stands for a directory name of up to 128 characters. While filename extensions are not required, they are highly recommended.

File separators may be either forward slashes ("/") or back slashes ("\").

Example

```
FILESYSTEM: COPY "C:/MYDIR/  
TEK00001.SET", "C:/AnotherDir/copyOf-  
TEK00001.SET"
```

This command copies the file named `TEK00001.SET` in the `MYDIR` directory on the C drive to the file named `copyOfTEK00001.SET` in the `AnotherDir` directory on the C drive.

FILESystem:DELEte

Description

This command (no query form) deletes a named file. It does not delete directories. Use the FILESystem:RMDir command to remove a directory.

Group

File System

Related Commands

FILESystem:COpy

Syntax

```
FILESystem:DELEte <file path>
```

Arguments

- <file path>

This is a quoted string that defines the file name and path. If the file path is within the current working directory, you need only specify the file name.

Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. If you do not specify them, the oscilloscope will delete the file in the current directory. <filename> stands for a filename of up to 128 characters. While filename extensions are not required, they are highly recommended. You can also use the inclusive filename *.* to delete all files.

File separators may be either forward slashes ("/") or back slashes ("\").

You will not be able to delete files which have read-only attributes. Refer to relevant operating system documentation for information on removing read-only attributes.

Example 1

```
FILESYSTEM:DELETE "NOT_MINE.SET"
```

This command deletes the file named NOT_MINE.SET on the default drive and directory.

FILESystem:DIR?

Description

This query-only command returns a list of quoted strings. Each string contains the name of a file or directory in the current working directory.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the timeout is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that the query was interrupted by *ESR?. In such cases, increase the timeout value and send the query again.

Group

File System

Related Commands

FILESystem:MKDir

Syntax

FILESystem:DIR?

Arguments

None.

Example

```
FILESYSTEM:DIR?
```

This query returns a list of files and directories in the current working directory.

FILESystem:MKDir

Description

This command (no query form) creates a new directory.

Group

File System

Related Commands

FILESystem:DIR?

Syntax

```
FILESystem:MKDir <directory path>
```

Arguments

- <directory path>

This is a quoted string that defines the directory to create. Input the directory using the form <drive>/<dir>/<directory name>.<drive> and one or more <dir>s are optional. <directory name> stands for a directory name of up to 128 characters. While filename extensions are not required, they are highly recommended.

File separators may be either forward slashes ("/") or back slashes ("\").

Example

```
FILESYSTEM:MKDIR "C:/NEATPICS"
```

This command creates the directory named NEATPICS at the root of the C drive.

FILESystem:PRInt

Description

This command (no query form) prints a named file to a named port. The only port allowed for this command is the GPIB port. It works the same as the FILESystem:READFile command.

This command is not IEEE 488.2 compliant for the following reasons:

- It produces output on the IEEE 488.2 bus. However, queries, not commands, are the only message units that should produce output on the 488.2 bus. Errors may not be recognized if they occur while data is being output in response to this command. This will be evident if a command is sent to the instrument before all of the data generated by this command has been read. In such cases, you should send a device clear message to the instrument.
- It does not use a standard IEEE 488.2 output format. The closest defined format is the indefinite format of ARBITRARY BLOCK PROGRAM DATA encoding. However, this command output omits the initial '#0' characters needed for that format.
- It allows newlines, ASCII value 10, in the output data. However, output on the bus can be terminated by newlines. The output of this command is only terminated by <EOI>.

Group

File System

Related Commands

FILESystem:READFile

Syntax

```
FILESystem:PRInt <filepath>, GPIb
```

Arguments

- <file path>

This is a quoted string that defines the file name and path. If the file path is within the current working directory, you need only specify the file name.

Input the file path using the form
<drive>/<dir>/<filename>.<drive> and one or

more <dir>s are optional. <filename> stands for a filename of up to 128 characters. While filename extensions are not required, they are highly recommended. You can also use the inclusive filename *.* to print all files.

File separators may be either forward slashes ("/") or back slashes ("\").

■ GPIb

This specifies that the file is sent out the GPIB port. GPIb is the only port allowed for this command.

Example

```
FILESYSTEM:PRINT "C:/TEK00000.IBM",GPIb
```

This command sends the file named TEK00000.IBM out the GPIB port.

FILESystem:READFile

Description

This command (no query form) prints a named file to the named port. It works the same as the FILESystem:PRInt command.

This command is not IEEE 488.2 compliant for the following reasons:

- It produces output on the IEEE 488.2 bus. However, queries, not commands, are the only message units that should produce output on the 488.2 bus. Errors may not be recognized if they occur while data is being output in response to this command. This will be evident if a command is sent to the instrument before all of the data generated by this command has been read. In such cases, you should send a device clear message to the instrument.
- It does not use a standard IEEE 488.2 output format. The closest defined format is the indefinite format of ARBITRARY BLOCK PROGRAM DATA encoding. However, this command output omits the initial '#0' characters needed for that format.
- It allows newlines, ASCII value 10, in the output data. However, output on the bus can be terminated by newlines. The output of this command is only terminated by <EOI>.

Group

File System

Related Commands

FILESystem:PRInt

Syntax

```
FILESystem:READFile <filepath>, GPIb
```

Arguments

- <file path>

This is a quoted string that defines the file name and path. If the file path is within the current working directory, you need only specify the file name.

Input the file path using the form <drive>/<dir>/<filename>.<drive> and one or more <dir>s are optional. <filename> stands for a file-

name of up to 128 characters. While filename extensions are not required, they are highly recommended. You can also use the inclusive filename *.* to read all files.

File separators may be either forward slashes ("/") or back slashes ("\").

- GPIb

This specifies that the file is sent out the GPIB port. GPIb is the only port allowed for this command.

Example

```
FILESYSTEM:READFILE "C:/ABC.DEF"
```

This command reads a file named "ABC.DEF" on the C drive and sends it to the GPIB port.

FILESystem:REName

Description

This command (no query form) assigns a new name to an existing file.

Group

File System

Syntax

```
FILESystem:REName <old file path>,<new  
file path>
```

Arguments

- <old file path>

This is a quoted string that defines the file to rename. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. <filename> stands for a filename of up to 128 characters. While filename extensions are not required, they are highly recommended.

File separators may be either forward slashes ("/") or back slashes ("\").

- <new file path>

This is a quoted string that defines the new name of the file. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. <filename> stands for a filename of up to 128 characters. While filename extensions are not required, they are highly recommended.

File separators may be either forward slashes ("/") or back slashes ("\").

Example

```
FILESYSTEM:RENAME  
"C:/TEK00000.SET", "C:/MYSETTING.SET"
```

This command gives the file named `TEK00000.SET` the new name of `MYSETTING.SET`. The file remains in the root directory on the C drive.

FILESystem:RMDir

Description

This command (no query form) deletes a named directory.

The directory cannot contain any files. If there are files in the directory, you must delete them before deleting the directory. Also, you cannot delete a directory if it is set up as read-only.

Group

File System

Syntax

```
FILESystem:RMDir <directory path>
```

Arguments

- <directory path>

This is a quoted string that defines the directory. If the file path is within the current working directory, you need only specify the file name.

Input the directory using the form <drive>/<dir>/<directory name>.<drive> and one or more <dir>s are optional. <directory name> stands for a directory name of up to 128 characters. While filename extensions are not required, they are highly recommended.

File separators may be either forward slashes ("/") or back slashes ("\").

Example

```
FILESYSTEM:RMDIR "C:/NOT-MINE"
```

This command deletes the directory named NOT-MINE on the C drive.

FILESystem:WRITEFile

Description

This command (no query form) copies block data from the GPIB port to a named file.

Group

File System

Syntax

```
FILESystem:WRITEFile <file path>,<data>
```

Arguments

- <file path>

This is a quoted string that defines the file name and path. If the file path is within the current working directory, you need only specify the file name.

Input the file path using the form <drive>/<dir>/<filename>.<drive> and one or more <dir>s are optional. <filename> stands for a filename of up to 128 characters. While filename extensions are not required, they are highly recommended.

File separators may be either forward slashes ("/") or back slashes ("\").

- <data>

This is sent in the indefinite format of ARBITRARY BLOCK PROGRAM DATA encoding as described in IEEE 488.2.

Example

```
FILESYSTEM:WRITEFILE "C:/SomeNew-  
File.txt",#221 1.23 2.54 8.22  
-1.22<NL+EOI>
```

This command creates a file named "SomeNew-File.txt" on the root directory of the C drive, containing a string of 21 characters " 1.23 2.54 8.22 -1.22".

HARDCopy

Description

This command sends a copy of the screen to the file specified by HARDCopy:FILEName. This command is equivalent to pressing the **PRINT** button on the front panel. When printing to a file, the file format is Windows Bitmap Image (BMP).

The HARDCopy query returns format, layout, and file information.

Group

Hardcopy

Related Commands

*WAI, *CLS?

Syntax 1

```
HARDCopy START
```

Syntax 2

```
HARDCopy?
```

Arguments

- START

Initiates copying.

Note: Use the *WAI command between HARDCopy START commands to ensure that the first hard copy is complete before starting another.

Example 1

```
HARDCopy START
```

This command initiates the screen copy.

Example 2

```
HARDCopy?
```

This query returns the hardcopy file path. The query might return:

```
:HARDCOPY:FILENAME "C:\TekScope\Images\hcFILE.BMP"
```

HARDCopy:FILENAME

Description

This command sets or returns the file/path that will be sent hard copy data on the next HARDCopy command.

Group

Hardcopy

Related Commands

HARDCopy

Syntax 1

```
HARDCopy:FILENAME <file path>
```

Syntax 2

```
HARDCopy:FILENAME?
```

Argument

- <file path>

This specifies that the hard copy is sent to the named file. <file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/ <filename>.<drive> and one or more <dir>s are optional. <filename> stands for a filename of up to 128 characters. If you don't specify the path with <drive> and one or more <dir>s, the default location is "C:\TekScope\Images". While filename extensions are not required, they are highly recommended.

Example 1

```
HARDCopy:FILENAME "TEK.IBM"
```

This command sets TEK.IBM as the selected file name. The next hard copy will be sent to this file in the default hard copy directory.

Example 2

```
HARDCopy:FILENAME?
```

This query might return

```
HARDCOPY:FILENAME "C:\TekScope\Images\TEK.IBM"
```

as the selected file.

HARDCopy:IMAGe

Description

This command sets or returns the inksaver mode for hard copy printing (File > Export Setup > Images > Image).

Group

Hardcopy

Syntax 1

```
HARDCopy:IMAGe {NORMal | INKSaver | ENHAN-  
cedwfm}
```

Syntax 2

```
HARDCopy:IMAGe?
```

Arguments

- NORMal

Preserves the screen background when printing waveforms.

- INKSaver

Changes the screen background to white. This saves ink when printing waveforms.

- ENHANcedwfm

Prints (color) screen images with colors designed to print on a white background.

Example 1

```
HARDCopy:IMAGe NORMal
```

This command sets the hardcopy mode to Normal.

HARDCopy:LAYout

Description

This command sets or returns the page orientation for hard copy.

Group

Hardcopy

Syntax 1

```
HARDCopy:LAYout {PORTRait | LANDscape}
```

Syntax 2

```
HARDCopy:LAYout?
```

Arguments

- PORTRait

Orients the screen image vertically on the printed page.

- LANDscape

Orients the screen image horizontally on the printed page.

Example 1

```
HARDCopy:LAYout LANDscape
```

This command sets the hard copy page orientation to Landscape.

HARDCopy:PALEtte

Description

This command sets or returns the hard copy color palette (File > Export Setup > Images > Palette).

Group

Hardcopy

Syntax 1

```
HARDCopy:PALEtte {COLOr|GRAYscale|BLACK-ANDWhite}
```

Syntax 2

```
HARDCopy:PALEtte?
```

Arguments

- COLOr

Hard copy output is color.

- GRAYscale

Hard copy output is gray.

- BLACKANDwhite

Hard copy output is black and white.

Example 1

```
HARDCopy:PALEtte
```

This command sets the hardcopy color palette to Color.

HARDCopy:READOuts

Description

This command sets or returns the area on the screen where the readouts appear for hardcopy.

Group

Hardcopy

Syntax 1

```
HARDCopy:READOuts {OFFGRaticule|ONGRaticule}
```

Syntax 2

```
HARDCopy:READOuts?
```

Arguments

- OFFGRaticule

Displays the readouts for hardcopy below the graticule area.

- ONGRaticule

Displays the readouts for hardcopy as they appear on the instrument screen.

Example 1

```
HARDCopy:READOuts OFFGRaticule
```

This command sets the readouts to appear below the graticule area for a hardcopy.

HARDCopy:VIEW

Description

This command sets or returns the area of the screen to be hard copied (File > Export Setup > Images > View).

Group

Hardcopy

Syntax 1

```
HARDCopy:VIEW {FULLSCREEN|GRaticule}
```

Syntax 2

```
HARDCopy:VIEW?
```

Arguments

- FULLSCREEN

Displays both the graticule and menu areas of the screen

- CRaticule

Displays only the graticule area of the screen.

Example 1

```
HARDCopy:VIEW FULLSCREEN
```

This command sets the hard copy view to Full Screen.

HDR

Description

This command is identical to the HEADer query and is included for backward compatibility purposes.

Group

Miscellaneous

Related Commands

HEADer, VERBose

Syntax 1

```
HDR {OFF|ON|<NR1>}
```

Syntax 2

```
HDR?
```

Arguments

■ OFF

This sets the Response Header Enable State to false. This causes the oscilloscope to omit headers on query responses, so that only the argument is returned.

■ ON

This sets the Response Header Enable State to true. This causes the oscilloscope to include headers on applicable query responses. You can then use the query response as a command.

■ <NR1>

A 0 sets the Response Header Enable State to false; any other value sets this state to true, which causes the oscilloscope to omit headers on query responses.

Example 1

```
HDR OFF
```

This command specifies that the oscilloscope omits headers on query responses, so that only the argument is returned.

Example 2

HDR?

This query might return

```
:HEADER 1
```

indicating that the oscilloscope is including headers on applicable query responses.

HEADer

Description

This command sets or queries the Response Header Enable State that causes the oscilloscope to either include or omit headers on query responses.

Note: This command does not affect IEEE Std 488.2-1987 Common Commands (those starting with an asterisk); they never return headers.

Group

Miscellaneous

Related Commands

HDR, VERBose

Syntax 1

```
HEADer {OFF | ON | <NR1>}
```

Syntax 2

```
HEADer?
```

Arguments

■ OFF

This sets the Response Header Enable State to false. This causes the oscilloscope to omit headers on query responses, so that only the argument is returned.

■ ON

This sets the Response Header Enable State to true. This causes the oscilloscope to include headers on applicable query responses. You can then use the query response as a command.

■ <NR1>

A 0 sets the Response Header Enable State to false; any other value sets this state to true, which causes the oscilloscope to omit headers on query responses.

Example 1

```
HEADer OFF
```

This command specifies that the oscilloscope omits headers on query responses, so that only the argument is returned.

Example 2

HEADer?

This query might return

```
:HEADER 1
```

indicating that the oscilloscope is including headers on applicable query responses.

HIStogram?

Description

This query-only command returns all histogram parameters; it queries the state of all histogram parameters that the user can set. This command is equivalent to selecting Waveform Histograms from the Measure menu.

Group

Histogram

Related Commands

MEASUrement:MEtHod

Syntax 1

HIStogram?

Arguments

None

Example 1

HIStogram?

This query might return

```
:HISTOGRAM:BOXPCNT 3.0000E+01,  
2.5100E+01,7.0000E+01,7.5200E+01;DISPLAY  
LINEAR  
;STATE 0;FUNCTION HORIZONTAL;SIZE  
2.0000E+00  
;SOURCE CH1
```

HIStogram:Box

Description

This command defines or returns the left, top, right, and bottom boundaries of the histogram box, in source waveform coordinates. This command is equivalent to selecting Waveform Histograms from the Measure menu and then setting Limits for Left, Right, Top, and Bottom in the appropriate boxes.

For example, If you selected Vertical Histogram and set the box left corner to the intersection of the center graticule, the top to one division up from center, the right to one division right of center, and the bottom to the ground graticule line, the coordinates returned would be 0,1,1, 0, corresponding to left, top, right, and bottom.

This command is similar to the HIStogram:BOXPcnt command except that command uses percentage coordinates to define the boundaries of the histogram box. This command is equivalent to selecting Waveform Histograms from the Measure menu and reviewing the histogram Left, Right, Top, and Bottom Limits.

Note: This command implicitly clears the histogram count and statistics for the histogram and histogram source data when HIStogram:STATE is turned on.

Group

Histogram

Related Commands

HIStogram:BOXPcnt

Syntax 1

```
HIStogram:Box <NR3>,<NR3>,<NR3>,<NR3>
```

Syntax 2

```
HIStogram:Box?
```

Arguments

- <NR3>

This specifies the left position of the histogram box in source waveform coordinates.

- <NR3>

This specifies the top position of the histogram box in source waveform coordinates.

- <NR3>

This specifies the right position of the histogram box in source waveform coordinates.

- <NR3>

This specifies the bottom position of the histogram box in source waveform coordinates.

Example 1

```
HISTogram:Box 1E-9, 0.250, 2E-9, 0.500
```

This command defines the coordinates of the histogram box in source waveform coordinates.

Example 2

```
HISTogram:Box?
```

This query might return

```
HISTOGRAM:BOX 1.0000E-09, 0.2500,  
2.0000E-09, 0.5000
```

which identifies the coordinates of the histogram box in source waveform coordinates.

HISistogram:BOXPcnt

Description

This command defines or queries the left, top, right, and bottom positions of the histogram box, in percentage coordinates. The upper left has the value 0,0 and the lower right has the value 100, 100. Any value outside the range of 0 to 100 is clipped. This command is similar to the HISistogram:Box command except that this command uses percentage coordinates to define the boundaries of the histogram box.

Note: This command implicitly clears the histogram count and statistics for the histogram and histogram source data when HISistogram:STATE is turned on.

Group

HISistogram

Related Commands

HISistogram:Box

Syntax 1

```
HISistogram:BOXPcnt <NR3>, <NR3>, <NR3>, <NR3>
```

Syntax 2

```
HISistogram:BOXPcnt?
```

Arguments

- <NR3>

This specifies the left position of the histogram box in percentage coordinates. The default value is 30.0.

- <NR3>

This specifies the top position of the histogram box in percentage coordinates. The default value is 25.1.

- <NR3>

This specifies the right position of the histogram box in percentage coordinates. The default value is 70.0.

- <NR3>

This specifies the bottom position of the histogram box in percentage coordinates. The default value is 75.2.

Example 1

`HISTogram:BOXPct 30.0,25.1,70.0,75.2`

This command sets the hardcopy page-layout format to default values.

Example 2

`HISTogram:BOXPct?`

This query might return

```
:HISTOGRAM:BOXPCT
3.0000E+01,25.1000E+01,7.0000E+01,7.5200E+
01.
```

HIStogram:COUNT

Description

This command (no query form) clears the count and statistics for the histogram and the histogram source data. If the histogram is on, then counting restarts. This command is equivalent selecting Waveform Histograms from the Measure menu and then clicking the Reset button.

Group

HIStogram

Related Commands

HIStogram:Box, HIStogram:BOXPcnt

Syntax

```
HIStogram:COUNT RESET
```

- RESET

This zeros the histogram count.

Example

```
HIStogram:COUNT RESET
```

This command clears the count and statistics for the histogram and the histogram source data.

HIStogram:DATa?

Description

This query-only command returns an ASCII comma-separated list of histogram data values (CSV). These values are compatible with many spreadsheet applications.

This command returns values for each bin comprising the histogram; 200 bin count values for vertical histograms and 500 bin count values for horizontal histograms.

Note: If there is no active histogram, there will be no response to the query and the instrument will report the following error: "Source waveform is not active."

Group

HIStogram

Syntax

HIStogram:DATa?

Arguments

None

Example

HIStogram:DATa?

This query might return

```
:HISTOGRAM:DATA  
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,  
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,  
0,0,0,0,0,  
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,  
0,0,0,0,0,  
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,  
0,0,0,0,0,  
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,  
0,0,0,0,0,  
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,  
0,0,0,0,0,  
0,0,2008,2008,2008,2008,2008,2008,2008,2008,200  
8,2008,  
2008,2008,2008,2008,2008,2008,2008,2008,20  
08,2008,  
2008,2008,2008,2008,2008,2008,2008,2008,20  
08,2008,  
2008,2008,2008,2008,2008,2008,2008,2008,20  
08,2008,
```

2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,2008,2008,2008,2008,2008,2008,20
08,2008,
2008,2008,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,
0,
0,0,0,0,
0,
0,0,0,0,
0,
0,0,0,0,
0,
0,0,0,0,
0,
0,0,0,0,
0,0,0,0,0

HIStogram:DISplay

Description

This command sets the format of the histogram displayed or whether the histogram is displayed. The query form of this command either returns the histogram display format or that histogram display is disabled. This command is equivalent to selecting Waveform Histogram from the Measure menu and then ensuring that Display is set to On.

Group

HIStogram

Related Commands

HIStogram:MODE, HIStogram:STATE

Syntax 1

```
HIStogram:DISplay {OFF|LOG|LINEAr}
```

Syntax 2

```
HIStogram:DISplay?
```

Arguments

- <OFF>

This disables the histogram display but allows the histogram data to accumulate.

- <LOG>

This specifies that the histogram display type is set to Log format.

- <LINEAr>

This sets the histogram display type to Linear format.

Example 1

```
HIStogram:DISplay OFF
```

This command disables the histogram display.

Example 2

```
HIStogram:DISplay?
```

This query might return

```
:HISTOGRAM:DISPLAY LOG
```

indicating that the current histogram type being displayed is in Log format.

HIStogram:FUNction

Description

This command either selects the type of histogram to create or returns the current histogram type. This command is equivalent to selecting Waveform Histogram from the Measure menu and then choosing either **Horizontal** or **Vertical** from the Histogram Mode group box.

Group

HIStogram

Related Commands

HIStogram:MODE, HIStogram:STATE

Syntax 1

```
HIStogram:FUNction {HORizontal|VERTical}
```

Syntax 2

```
HIStogram:FUNction?
```

Arguments

- **HORizontal**

This displays a horizontally positioned histogram that shows time distribution.

- **VERTical**

This displays a vertically positioned histogram that shows a voltage distribution (or another distribution such as amperes).

Example 1

```
HIStogram:FUNction HORizontal
```

This command sets the histogram mode to the horizontal display mode.

Example 2

```
HIStogram:FUNction?
```

This query might return

```
:HISTOGRAM:FUNCTION VERTICAL
```

indicating that the vertical display mode is selected for the histogram.

HIStogram:MODE

Description

This command selects the type of histogram to create or disables the histogram display. The query form of this command returns the current histogram type or that histogram display functionality is disabled. This command is equivalent to selecting Waveform Histogram from the Measure menu and then choosing from the Histogram Mode group box.

HIStogram:MODE is functionally equivalent to the HIStogram:FUNCTION and HIStogram:STATE commands and is included for compatibility purposes.

Group

HIStogram

Related Commands

HIStogram:FUNCTION, HIStogram:STATE, HIStogram:DISplay

Syntax 1

```
HIStogram:MODE {HORizontal|VERTical|OFF}
```

Syntax 2

```
HIStogram:MODE?
```

Arguments

- HORizontal

This enables a horizontally positioned histogram that shows time distribution.

- VERTical

This enables a vertically positioned histogram that shows a voltage distribution (or another distribution such as amperes).

- OFF

This disables collection of the histogram measurement.

Example 1

```
HIStogram:MODE HORizontal
```

This command sets the type of histogram created to horizontal, which displays a horizontally positioned histogram that shows time distribution.

Example 2

HISTogram:MODE?

This query might return

:HISTOGRAM:MODE OFF

indicating that histogram display is disabled.

HIStogram:SIze

Description

This command sets or queries the width or height of the histogram on the screen. This is equivalent to selecting Waveform Histograms from the Measure menu and then entering a value in the Histo Size box.

Group

HIStogram

Related Commands

HIStogram:DISplay

Syntax 1

```
HIStogram:SIze <NR3>
```

Syntax 2

```
HIStogram:SIze?
```

Arguments

- <NR3>

This specifies the histogram size. The value can vary from 0.1 to 8.0 divisions in HORizontal mode and from 0.1 to 10.0 divisions in VERTical mode. Resolution is to the nearest pixel.

Example 1

```
HIStogram:SIze 2.0
```

This command sets the size of the histogram to 2.0 divisions.

Example 2

```
HIStogram:SIze?
```

This query might return

```
:HISTOGRAM:SIze 2.0000E+01
```

indicating the histogram size is set to 2.0 divisions.

HIStogram:SOUrce

Description

This command sets or queries which source will be compared against the histogram box when the histogram testing is enabled. This is equivalent to selecting Waveform Histograms from the Measure menu and then choosing the desired waveform source. The waveform need not be displayed for histograms to run. You might want the channel displays disabled so you can see a full-screen histogram and not have waveform data confuse the display.

Group

HIStogram

Related Commands

HIStogram:DISplay,

Syntax 1

```
HIStogram:SOUrce {CH<x> | MATH<x> | REF<x>}
```

Syntax 2

```
HIStogram:SOUrce?
```

Arguments

■ CH<x>

This selects a channel waveform as the source for the histogram. The channel waveform range is from 1 through 4.

■ MATH<x>

This selects a math waveform as the source for the histogram. The valid math waveform range is from 1 through 4.

■ REF<x>

This selects a reference waveform as the source for the histogram. The valid reference waveform range is from 1 through 4.

Example 1

```
HIStogram:SOUrce CH1
```

This command enables the channel 1 waveform to be compared against the histogram box. The default time-base, Main, is used for the histogram display.

Example 2

HISTogram:SOUrce?

This query might return

:HISTOGRAM:SOURCE CH1

indicating that the waveform for channel 1 is the source for the histogram.

HIStogram:STATE

Description

This command sets or queries whether the histogram calculations are enabled. This is equivalent to selecting Waveform Histograms from the Measure menu.

Group

HIStogram

Related Commands

HIStogram:MODE, MEASUREMENT:METHOD

Syntax 1

```
HIStogram:STATE {ON|OFF|NR1}
```

Syntax 2

```
HIStogram:STATE?
```

Arguments

- <ON>

This enables histogram calculations.

- <OFF>

This disables the histogram calculations.

- <NR1>

A 0 disables histogram calculations; any other value enables histogram calculations.

Example 1

```
HIStogram:STATE ON
```

This command enables histogram calculations.

Example 2

```
HIStogram:STATE?
```

This query might return

```
:HISTOGRAM:STATE 0
```

indicating that histogram calculations are currently disabled.

HORizontal?

Description

This query-only command returns all learnable settings for the horizontal commands.

Group

HORizontal

Syntax

HORizontal?

Example

HORizontal?

This query might return

```
:HORIZONTAL:DELAY:MODE 1
;POSITION 5.0000E+01;TIME 0.0000E+00;
:HORIZONTAL:MAIN:SCALE 2.0000E-07;POSITION
5.0000E+01
;UNITS:VALUE 1.0000E+00;STRING "s";OFFSET
0.0000E+00;
:HORIZONTAL:RECORDLENGTH 5000;ROLL
AUTO;FASTFRAME:STATE 0
;LENGTH 5000;COUNT 2;SELECTED:CH1 2;CH2
2;CH3 2;CH4 2
;MATH1 2;MATH2 2;MATH3 2;MATH4 2;REF1
2;REF2 2;REF3 2
;REF4 2;:HORIZONTAL:FASTFRAME:REF:SOURCE
CH1;FRAME 1;
:HORIZONTAL:FASTFRAME:TRACK LIVE
```

HORizontal:DIVisions?

Description

This query-only command returns the number of graticule divisions over which the waveform is displayed.

Group

Horizontal

Syntax

```
HORizontal:DIVisions?
```

Arguments

None.

Example

```
HORizontal:DIVisions?
```

This query might return

```
:HORIZONTAL:DIVISIONS 1.0000E-01
```

indicating that the waveform is displayed across one division.

HORizontal:FASTframe:COUNT

Description

This command sets or queries FastFrame frame count. This is equivalent to selecting FastFrame Setup from the Horiz/Acq menu and entering a value in the Frame Count box.

FastFrame, also known as memory segmentation, captures a series of triggered acquisitions with minimal intervening time. This command is equivalent to selecting Fast Frame Setup from the Horiz/Acq menu and entering a value in the Frame Count box.

Group

Horizontal

Related Commands

HORizontal:RECOrdlength

Syntax 1

```
HORizontal:FASTframe:COUNT <NR1>
```

Syntax 2

```
HORizontal:FASTframe:COUNT?
```

Arguments

- <NR1>

This indicates the number of frames to acquire.

Example 1

```
HORIZONTAL:FASTFRAME:COUNT 2
```

This command sets up FastFrame mode to acquire two frames (segments) of data.

Example 2

```
HORIZONTAL:FASTFRAME:COUNT?
```

This query might return

```
:HORIZONTAL:FASTFRAME:COUNT 4
```

indicating that FastFrame mode is currently set to acquire 4 frames (segments) of data.

HORizontal:FASTframe:REF:FRame

Description

This command sets or queries the FastFrame reference frame number. The Reference Frame number is used to calculate time differences for each frame and displays those differences in the graticule. This command is equivalent to selecting FastFrame Setup in the Horiz/Acq menu and entering a value in the Time Stamps/Frame box.

Group

Horizontal

Related Commands

HORizontal:FASTframe:REF:SOUrce, HORizontal:FASTframe:TIMEstamp:DELTA:<wfm>?, HORizontal:FASTframe:TIMEstamp:REF?, HORizontal:FASTframe:TRACK

Syntax 1

```
HORizontal:FASTframe:REF:FRame <NR1>
```

Syntax 2

```
HORizontal:FASTframe:REF:FRame?
```

Argument

- <NR1>

This specifies the FastFrame reference frame.

Example 1

```
HORIZONTAL:FASTFRAME:REF:FRAME 2
```

This command sets the FastFrame reference frame to 2.

Example 2

```
HORIZONTAL:FASTFRAME:REF:FRAME?
```

This query might return

```
:HORIZONTAL:FASTFRAME:REF:FRAME 3
```

indicating that the FastFrame reference frame is currently set to 3.

HORizontal:FASTframe:LENgth

Description

This command sets or queries the horizontal record length to the number of sample points in each frame. This command is equivalent to selecting FastFrame Setup from the Horiz/Acq menu and entering a value in the Rec Length box. FastFrame, also known as memory segmentation, captures a series of triggered acquisitions with minimal intervening time between them.

Group

Horizontal

Syntax 1

```
HORizontal:FASTframe:LENgth <NR1>
```

Syntax 2

```
HORizontal:FASTframe:LENgth?
```

Argument

- <NR1>

This represents the supported values for horizontal record lengths, which can range from 500 through 400000. For more information about valid data point ranges, select Specifications from the Help menu and choose the Horizontal & Acquisition tab.

Example 1

```
HORIZONTAL:FASTFRAME:LENGTH 5000
```

This command sets the horizontal record length to 5000 sample points in each frame.

Example 2

```
HORizontal:FASTFRAME:LENGTH?
```

This query might return

```
:HORIZONTAL:FASTFRAME:LENGTH 25000
```

indicating that the FastFrame record length is set to 25000 sample points in each frame.

HORizontal:FASTframe:MULTipleframes:MODE

Description

This command sets or queries the FastFrame multiple frames mode. This feature displays multiple overlaid frames.

Group

Horizontal

Related Commands

HORizontal:FASTframe:COUNT
HORizontal:FASTframe:MULTipleframes:FRAMES-
Tart:CH<x>, MATH<x>, REF<x>
HORizontal:FASTframe:MULTipleframes:NUM-
FRames:CH<x>, MATH<x>, REF<x>

Syntax 1

```
HORizontal:FASTframe:MULTipleframes:MODE  
{OFF|OVERlay}
```

Syntax 2

```
HORizontal:FASTframe:MULTipleframes:MODE?
```

Arguments

- OFF

Disables the multiple-frames mode.

- OVERlay

Sets the multiple-frames mode to overlay.

Example 1

```
HORizontal:FASTframe:MULTipleframes:MODE  
OFF
```

This command disables the FastFrame multiple frames feature.

Example 2

```
HORizontal:FASTframe:MULTipleframes:MODE?
```

This query might return

```
:HORIZONTAL:FASTFRAME:MULTIPLEFRAMES:MODE  
OFF,
```

indicating that the FastFrame multiple frames feature is disabled.

HORizontal:FASTframe:MULtipleframes:FRAMEStart: CH<x>

Description

This command sets or queries the start frame number of the waveform source for the FastFrame multiple frames feature. This feature displays multiple overlaid frames.

Group

Horizontal

Related Commands

HORizontal:FASTframe:COUNT,
HORizontal:FASTframe:MULtipleframes:MODE,
HORizontal:FASTframe:MULtipleframes:NUM-
FRames:CH<x>, MATH<x>, REF<x>

Syntax 1

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart:CH<x> <NR1>
```

Syntax 2

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart:CH<x>?
```

Arguments

- CH<x>

<x> specifies the waveform source as an integer from 1 through 4 for four-channel oscilloscopes.

- <NRX>

Specifies the start frame number.

Example 1

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart CH1 3
```

This command sets the start number to frame 3 on Channel 1.

Example 2

`HORizontal:FASTframe:MULtipleframes:FRA-
MESTart:CH1?`

This query might return `:HORIZONTAL:FASTFRAME:MUL-
TIPLEFRAMES:FRAMEStart CH1 3,`
indicating that the frame number starts on frame 3 of
Channel 1.

HORizontal:FASTframe:MULtipleframes:FRAMEStart: MATH<x>

Description

This command sets or queries the start frame number of the waveform source for the FastFrame multiple frames feature. This feature displays multiple overlaid frames.

Group

Horizontal

Related Commands

HORizontal:FASTframe:COUNT,
HORizontal:FASTframe:MULtipleframes:MODE,
HORizontal:FASTframe:MULtipleframes:NUM-
FRames:CH<x>, MATH<x>, REF<x>

Syntax 1

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart:MATH<x> <NR1>
```

Syntax 2

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart:MATH<x>?
```

Arguments

- MATH<x>

<x> specifies the waveform source as an integer from 1 through 4 for four-channel oscilloscopes.

- <NRX>

Specifies the start frame number.

Example 1

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart MATH1 3
```

This command sets the start number to frame 3 on the Math 1 waveform.

Example 2

`HORizontal:FASTframe:MULtipleframes:FRA-
MESTart:MATH1?`

This query might return `:HORIZONTAL:FASTFRAME:MUL-
TIPLEFRAMES:FRAMEStart MATH1 3,`
indicating that the frame number starts on frame 3 of Math
1 waveform.

HORizontal:FASTframe:MULtipleframes:FRAMEStart: REF<x>

Description

This command sets or queries the start frame number of the waveform source for the FastFrame multiple frames feature. This feature displays multiple overlaid frames.

Group

Horizontal

Related Commands

HORizontal:FASTframe:COUNT,
HORizontal:FASTframe:MULtipleframes:MODE,
HORizontal:FASTframe:MULtipleframes:NUM-
FRames:CH<x>, MATH<x>, REF<x>

Syntax 1

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart:REF<x> <NR1>
```

Syntax 2

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart:REF<x>?
```

Arguments

- REF<x>

<x> specifies the waveform source as an integer from 1 through 4 for four-channel oscilloscopes.

- <NRX>

Specifies the start frame number.

Example 1

```
HORizontal:FASTframe:MULtipleframes:FRA-  
MEStart REF1 3
```

This command sets the start number to frame 3 on the Reference 1 waveform.

Example 2

`HORizontal:FASTframe:MULtipleframes:FRA-
MESTart:REF1?`

This query might return `:HORIZONTAL:FASTFRAME:MUL-
TIPLEFRAMES:FRAMEStart REF1 3,`
indicating that the frame number starts on frame 3 of Ref-
erence 1 waveform.

HORizontal:FASTframe:MULtipleframes:NUMFRames: CH<x>

Description

This command sets or queries the number of frames on the specified waveform for the FastFrame multiple frames feature. This feature displays multiple overlaid frames.

Group

Horizontal

Related Commands

HORizontal:FASTframe:COUNT,
HORizontal:FASTframe:MULtipleframes:MODE,
HORizontal:FASTframe:MULtipleframes:FRAMES-
Tart:CH<x>, MATH<x>, REF<x>

Syntax 1

```
HORizontal:FASTframe:MULtipleframes:NUM-  
FRames:CH<x> <NR1>
```

Syntax 2

```
HORizontal:FASTframe:MULtipleframes:NUM-  
FRames:CH<x>?
```

Arguments

- CH<x>

<x> specifies the waveform source as an integer from 1 through 4 for four-channel oscilloscopes.

- <NRX>

Represents the number of frames on the specified waveform.

Example 1

```
HORizontal:FASTframe:MULtipleframes:NUM-  
FRames CH1 3
```

This command sets the number of frames to 3 on Channel 1.

Example 2

```
HORizontal:FASTframe:MULtipleframes:NUM-  
FRames:CH1?
```

HORizontal:FASTframe:MULtipleframes:NUMFRames: MATH<x>

Description

This command sets or queries the number of frames on the specified waveform for the FastFrame multiple frames feature. This feature displays multiple overlaid frames.

Group

Horizontal

Related Commands

HORizontal:FASTframe:COUNT,
HORizontal:FASTframe:MULtipleframes:MODE,
HORizontal:FASTframe:MULtipleframes:FRAMES-
Tart:CH<x>, MATH<x>, REF<x>

Syntax 1

```
HORizontal:FASTframe:MULtipleframes:NUM-  
FRames:MATH<x> <NR1>
```

Syntax 2

```
HORizontal:FASTframe:MULtipleframes:NUM-  
FRames:MATH<x>?
```

Arguments

- MATH<x>

<x> specifies the waveform source as an integer from 1 through 4 for four-channel oscilloscopes.

- <NRX>

Represents the number of frames on the specified waveform.

Example 1

```
HORizontal:FASTframe:MULtipleframes:NUM-  
FRames MATH1 3
```

This command sets the number of frames to 3 for the Math 1 waveform.

Example 2

`HORizontal:FASTframe:MULtipleframes:NUM-
FRames:MATH1?`

This query might return `:HORIZONTAL:FASTFRAME:MUL-
TIPLEFRAMES:NUMFRAMES MATH1 3`,
indicating that the number of frames for the Math 1 wave-
form is 3.

HORizontal:FASTframe:MULTipleframes:NUMFRames: REF<x>

Description

This command sets or queries the number of frames on the specified waveform for the FastFrame multiple frames feature. This feature displays multiple overlaid frames.

Group

Horizontal

Related Commands

HORizontal:FASTframe:COUNT,
HORizontal:FASTframe:MULTipleframes:MODE,
HORizontal:FASTframe:MULTipleframes:FRAMES-
Tart:CH<x>, MATH<x>, REF<x>

Syntax 1

```
HORizontal:FASTframe:MULTipleframes:NUM-  
FRames:REF<x> <NR1>
```

Syntax 2

```
HORizontal:FASTframe:MULTipleframes:NUM-  
FRames:REF<x>?
```

Arguments

- REF<x>

<x> specifies the waveform source as an integer from 1 through 4 for four-channel oscilloscopes.

- <NRX>

Represents the number of frames on the specified waveform.

Example 1

```
HORizontal:FASTframe:MULTipleframes:NUM-  
FRames REF1 3
```

This command sets the number of frames to 3 for the Reference 1 waveform.

Example 2

`HORizontal:FASTframe:MULtipleframes:NUM-
FRames:REF1?`

This query might return `:HORIZONTAL:FASTFRAME:MUL-
TIPLEFRAMES:NUMFRAMES REF1 3,`
indicating that the number of frames for the Reference 1
waveform is 3.

HORizontal:FASTframe:REF:SOURce

Description

This command sets or queries FastFrame Reference waveform source. This is equivalent to selecting FastFrame Setup from the Horiz/Acq menu and choosing the reference source.

Group

Horizontal

Related Commands

HORizontal:FASTframe:REF:FRAME, HORizontal:FASTframe:TIMEstamp:DELTA:<wfm>?, HORizontal:FASTframe:TIMEstamp:REF?, HORizontal:FASTframe:TRACK

Syntax 1

```
HORizontal:FASTframe:REF:SOURce <wfm>
```

Syntax 2

```
HORizontal:FASTframe:REF:SOURce?
```

Argument

- <wfm>

This specifies the FastFrame Reference waveform source. Valid waveforms include CH<x>, MATH<x>, and REF<x>; the x variable can be expressed as an integer ranging from 1 through 4.

Example 1

```
HORIZONTAL:FASTFRAME:REF:SOURCE CH2
```

This command sets the horizontal FastFrame waveform reference source to CH2.

Example 2

```
HORIZONTAL:FASTFRAME:REF:SOURCE?
```

This query might return

```
:HORIZONTAL:FASTFRAME:REF:SOURCE CH2
```

indicating that the FastFrame waveform source is currently set to CH2.

HORizontal:FASTframe:SELECTED:<wfm>

Description

This command sets or returns the FastFrame Selected frame number on the specified waveform. This is equivalent to selecting FastFrame Setup from the Horiz/Acq menu and then choosing the waveform source and frame. Valid waveforms include CH<x>, MATH<x> and REF<x>; the x variable can be expressed as an integer ranging from 1 through 4.

Group

Horizontal

Related Commands

HORizontal:FASTframe:TRACk, HORizontal:FASTframe:TIMEstamp:DELTA:<wfm>?, HORizontal:FASTframe:TIMEstamp:REF?

Syntax 1

```
HORizontal:FASTframe:SELECTED:<wfm> <NR1>
```

Syntax 2

```
HORizontal:FASTframe:SELECTED?
```

Argument

- <NR1>

This specifies the selected frame number on the specified waveform.

Example 1

```
HORIZONTAL:FASTFRAME:SELECTED:CH2 33
```

This command sets the FastFrame Selected frame number on channel 2 to 33.

Example 2

```
HORIZONTAL:FASTFRAME:SELECTED:CH4?
```

This query might return :HORIZONTAL:FASTFRAME:SELECTED:CH4 25, indicating that the FastFrame frame number on channel 4 is set to 25.

Example 3

HORIZONTAL:FASTFRAME:SELECTED?

This query might return

```
:HORIZONTAL:FASTFRAME:SELECTED:CH1 2; CH2  
2; CH3 2; CH4 2; MATH1 2; MATH2 2; MATH3  
2; MATH4 2; REF1 8; REF2 8; REF3 8; REF4 8
```

indicating all waveforms' Fastframe selected frame numbers.

HORizontal:FASTframe:STATE

Description

This command sets or returns the state of FastFrame acquisition. This is equivalent to setting FastFrame On in the FastFrame Setup menu. FastFrame, also known as memory segmentation, lets users capture a series of triggered acquisitions with minimal time between them.

The digitizing oscilloscope in FastFrame mode is ready to accept a continuous burst of triggers 400 ms after the controller sends the ACQUIRE:STATE RUN command.

Group

Horizontal

Related Commands

ACQUIRE:STATE RUN

Syntax 1

```
HORizontal:FASTframe:STATE {ON|OFF|<NR1>}
```

Syntax 2

```
HORizontal:FASTframe:STATE?
```

Arguments

- <ON>

This turns on horizontal FastFrame.

- <OFF>

This turns off horizontal FastFrame.

- <NR1>

A 0 turns off horizontal FastFrame; any other value turns on horizontal FastFrame.

Example 1

```
HORIZONTAL:FASTFRAME:STATE ON
```

This command turns on horizontal FastFrame.

Example 2

`HORIZONTAL:FASTFRAME:STATE?`

This query might return

`:HORIZONTAL:FASTFRAME:STATE 0`

indicating that the current state of FastFrame is off.

HORizontal:FASTframe:TIMEStamp:ALL:<wfm>?

Description

This query-only command returns the frame number and time stamp for each frame between requested frames, inclusive, within the specified waveform. Argument order is unimportant. Valid waveforms include CH<x>, MATH<x> and REF<x>; the x variable can be expressed as an integer ranging from 1 through 4.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the timeout is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that the query was interrupted by *ESR?. In such cases, increase the timeout value and send the query again.

Group

Horizontal

Syntax

```
HORizontal:FASTframe:TIMEStamp:ALL:<wfm>?  
<NRF>, <NRF>
```

Arguments

- <NRF>

This specifies the selected "from" frame number within the specified waveform.

- <NRF>

This specifies the selected "to" frame number within the specified waveform.

Example

```
HORIZONTAL:FASTFRAME:TIMESTAMP:ALL:CH2?  
4,1
```

This query might return

```
:HORIZONTAL:FASTFRAME:TIMESTAMP:ALL:CH1  
4,1,"02 Mar 2000 20:10:54.542 037 272 620"
```

which is the list of time stamps from frame 1 through frame 4 on channel 1.

HORizontal:FASTframe:TIMEStamp:BEtween:<wfm>?

Description

This query-only command returns the relative trigger for the delta time between the specified frames, within the specified waveform. Valid waveforms include CH<x>, MATH<x> and REF<x>; the x variable can be expressed as an integer ranging from 1 through 4.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the timeout is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that the query was interrupted by *ESR?. In such cases, increase the timeout value and send the query again.

Group

Horizontal

Syntax

```
HORizontal:FASTframe:TIMEStamp:BE-  
TWEEN:<wfm>? <NRF>, <NRF>
```

Arguments

- <NRF>

This specifies the selected "from" frame number within the specified waveform.

- <NRF>

This specifies the selected "to" frame number within the specified waveform.

Example

```
HORIZONTAL:FASTFRAME:TIMESTAMP:BE-  
TWEEN:CH1? 5,2
```

This query might return

```
:HORIZONTAL:FASTFRAME:TIMESTAMP:BE-  
TWEEN:CH1 5,2,"00:00:00.010 000 540 624",
```

which is the delta of channel 1's Time Stamp frame 5 -
Time Stamp frame 2 (TS[5]-TS[2]).

HORizontal:FASTframe:TIMESta:DELTA:<wfm>?

Description

This query-only command returns the relative time between the triggers of the FastFrame Selected and the FastFrame Reference, within the specified waveform. Valid waveforms include CH<x>, MATH<x> and REF<x>; the x variable can be expressed as an integer ranging from 1 through 4.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the timeout is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that the query was interrupted by *ESR?. In such cases, increase the timeout value and send the query again.

Group

Horizontal

Syntax

```
HORizontal:FASTframe:TIMESta:DEL-  
Ta:<wfm>?
```

Example

```
HORIZONTAL:FASTFRAME:TIMESTAMP:DELTA:CH1?
```

This query might return

```
:HORIZONTAL:FASTFRAME:TIMESTAMP:DELTA:CH1  
"00:00:00.006 000 306 556"
```

which is the delta time stamp of FastFrame Selected Frame on channel 1 - FastFrame Reference Frame (TS[C1 Selected]-TS[Reference]).

HORizontal:FASTframe:TIMEStamp:FRame:<wfm>?

Description

This query-only command returns the absolute trigger date and time for the specified frame and waveform. Valid waveforms include CH<x>, MATH<x> and REF<x>; the x variable can be expressed as an integer ranging from 1 through 4.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the timeout is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that the query was interrupted by *ESR?. In such cases, increase the timeout value and send the query again.

Group

Horizontal

Syntax

```
HORizontal:FASTframe:TIME-  
Stamp:FRame:<wfm>? <NR1>
```

Argument

- <NR1>

The frame number for which the timestamp will be returned on the specified waveform.

Example

```
HORIZONTAL:FASTFRAME:TIMESTAMP:FRAME:CH1?  
10
```

This query might return

```
:HORIZONTAL:FASTFRAME:TIMESTAMP:FRAME:CH1  
10,"02 Mar 2000 20:10:54.536 036 928 432"
```

which is the time stamp of channel 1, frame 10.

HORizontal:FASTframe:TIMEStamp:REF?

Description

This query-only command returns the absolute trigger date and time for FastFrame reference.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the timeout is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that the query was interrupted by *ESR?. In such cases, increase the timeout value and send the query again.

Group

Horizontal

Related Commands

HORizontal:FASTframe:REF:FRame, HORizontal:FASTframe:REF:SOURce

Syntax

```
HORizontal:FASTframe:TIMEStamp:REF?
```

Example

```
HORIZONTAL:FASTFRAME:TIMESTAMP:REF?
```

This query might return

```
:HORIZONTAL:FASTFRAME:TIMESTAMP:REF "02  
Mar 2000 20:10:54.533 036 838 784", which  
is the time stamp of FastFrame Reference.
```

HORizontal:FASTframe:TIMEStamp:SE- LECTED:<wfm>?

Description

This query-only command returns the absolute trigger date and time for the FastFrame Selected, within the specified waveform. Valid waveforms include CH<x>, MATH<x> and REF<x>; the x variable can be expressed as an integer ranging from 1 through 4.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the timeout is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that the query was interrupted by *ESR?. In such cases, increase the timeout value and send the query again.

Group

Horizontal

Related Commands

HORizontal:FASTframe:SELECTED:<wfm>?

Syntax

```
HORizontal:FASTframe:TIMEStamp:SE-  
LECTED:<wfm>?
```

Example

```
HORIZONTAL:FASTFRAME:TIMESTAMP:SE-  
LECTED:REF1?
```

This query might return

```
:HORIZONTAL:FASTFRAME:TIMESTAMP:SE-  
LECTED:REF1 "02 MAR 2000 20:10:54:539 037  
145 340"
```

which is the time stamp of FastFrame Selected on reference 1.

HORizontal:FASTframe:TRACk

Description

This command sets up or returns the state of FastFrame tracking feature. This command is equivalent to selecting FastFrame Setup from the Horiz/Acq menu and then clicking the desired Frame Tracking state.

When FastFrame Track is set to "live", the channel and math waveforms are locked together. Adjusting a channel waveform also adjusts a related math waveform. All reference waveforms are also locked together but they are separate from channel and math waveforms.

For example, when you set the Selected Frame Source Ch1 to Frame 3, then Selected Frame Ch2, Ch3, Ch4, Math1, Math2, Math3 and Math4 are also set to Frame 3.

When you set the Selected Frame Source Ref1 to Frame 2, then Selected Frame Ref2, Ref3 and Ref4 are also set to Frame 2. If the Frame Tracking is set to Live, changing Selected Frame Ch1 will not affect the Selected Frame Ref1 frame of the Reference Frame setting.

When FastFrame Track is set to "all", the channel, math and reference waveforms are locked together. Adjusting a channel waveform also adjusts the related math and reference waveforms.

For example, when you set the Selected Frame Source Ch1 to Frame 3, then Selected Frame Ch2, Ch3, Ch4, Math1, Math2, Math3, Math4, Ref1, Ref2, Ref3 and Ref4 are also set to Frame 3.

Group

Horizontal

Related Commands

HORizontal:FASTframe:REF:FRAMe, HORizontal:FASTframe:REF:SOURce, HORizontal:FASTframe:SELECTED:<wfm>?

Syntax 1

```
HORizontal:FASTframe:TRACk {LIVE|ALL}
```

Syntax 2

```
HORizontal:FASTframe:TRACk?
```

Arguments

- LIVE

This sets FastFrame Track to Live.

- ALL

This sets FastFrame Track to All.

Example 1

```
HORIZONTAL:FASTFRAME:TRACK LIVE
```

This command sets FastFrame Track to live.

Example 2

```
HORIZONTAL:FASTFRAME:TRACK?
```

This query might return

```
:HORIZONTAL:FASTFRAME:TRACK ALL
```

indicating that all FastFrame Selected Frames and the FastFrame Reference will be locked together.

HORizontal:MAIn?

Description

This query-only command returns the time per division of the time base. This command is equivalent to selecting Position/Scale from the Horiz/Acq menu.

Group

Horizontal

Related Commands

HORizontal:SCAlE

Syntax

HORizontal:MAIn?

Example

HORizontal:MAIn?

This query might return

```
:HORIZONTAL:MAIN:SCALE 1.0000E-04;POSITION  
5.0000E+01;UNITS:VALUE 1.0000E+00;STRING  
"s";OFFSET 0.0000E+01
```

HORizontal[:MAIn]:DELay:POSition

Description

This command sets or queries the time base position when Horizontal Delay Mode is turned on. This command is equivalent to selecting Horizontal/Acquisition Setup from the Horiz/Acq menu and then entering a Ref Point value.

Group

Horizontal

Related Commands

HORizontal[:MAIn]:DELay:TIME

Syntax 1

```
HORizontal[:MAIn]:DELay:POSition <NR3>
```

Syntax 2

```
HORizontal[:MAIn]:DELay:POSition?
```

Arguments

- <NR3>

This is from 0 to ≈ 100 and is the percent of the waveform that is displayed left of the center graticule.

Note: The upper limit of the waveform position is slightly limited by a value that is determined from the record length (upper limit = $100 - 1/\text{record length}$).

Example 1

```
HORizontal:DELay:POSition 5E+1
```

This command sets the time base trigger delay position to 50 percent.

Example 2

```
HORizontal:DELay:POSition?
```

This query might return

```
:HORIZONTAL:MAIN:DELAY:POSITION 5.0000E+01
```

indicating that the time base trigger delay position is currently set to 50 percent

HORizontal[:MAIn]:DELay:MODE

Description

This command sets or queries the time base trigger delay mode. This command is equivalent to choosing Delay Mode On from the Horiz/Acq menu.

Group

Horizontal

Related Commands

HORizontal[:MAIn]:DELay:TIME

Syntax 1

```
HORizontal[:MAIn]:DELay:MODE  
[ON|OFF|<NR1>]
```

Syntax 2

```
HORizontal[:MAIn]:DELay:MODE?
```

Arguments

- ON

This enables the time base trigger delay mode.

- OFF

This disables the time base trigger delay mode.

- <NR1>

A 0 disables the time base trigger delay mode, any other value enables the time base trigger delay mode.

Example 1

```
HORizontal:DELay:MODE ON
```

This command enables the time base trigger delay mode.

Example 2

```
HORizontal:DELay:MODE?
```

This query might return

```
:HORIZONTAL:DELAY:MODE 1
```

indicating that the time base trigger delay mode is currently enabled.

HORizontal[:MAIn]:DELay:TIME

Description

This command sets or queries the time base trigger delay time. This command is equivalent to selecting Position/Scale from the Horiz/Acq menu and choosing a value for Horiz Delay.

Group

Horizontal

Related Commands

HORizontal[:MAIn]:DELay:MODE, HORizontal[:MAIn]:DELay:POSition

Syntax 1

```
HORizontal[:MAIn]:DELay:TIME <NR3>
```

Syntax 2

```
HORizontal[:MAIn]:DELay:TIME?
```

Argument

- <NR3>

This specifies the time base trigger delay time setting, typically represented in seconds.

Example 1

```
HORizontal:DELay:TIME 5.0E-3
```

This command sets the time base trigger delay time to 5 ms.

Example 2

```
HORizontal:DELay:TIME?
```

This query might return

```
:HORIZONTAL:MAIN:DELAY 5.0000E-05
```

indicating that the time delay setting is 5 μ s.

HORizontal:MAIn:INTERPRatio?

Description

This query-only command returns the Horizontal Main time base interpolation ratio. The ratio is useful for calculating the maximum sampling rate when users request sampling rates that are faster than the maximum rate specified by the acquisition hardware.

The query HORizontal:MAIn:SAMPLERate? returns the sampling rate (real and interpolated). The real maximum sampling rate, as specified by the acquisition hardware, is:

Max sampling rate =

`HORizontal:MAIn:SAMPLERate?/HORizontal:MAIn:INTERPRatio?`

The default value is 1.

An interpolation ratio other than 1 means that displayed and retrieved waveforms contain additional points that are derived through interpolation. At such times, the retrieved waveform data is actually 16-bit data. You should set the data width accordingly.

Group

Horizontal

Syntax

`HORizontal:MAIn:INTERPRatio?`

Example

`HORIZONTAL:MAIN:INTERPRATIO?`

This query might return

`:HORIZONTAL:MAIN:INTERPRATIO 10.0`

HORizontal[:MAIn]:POSition

This command either positions the waveform horizontally on the display or returns the horizontal position on the display and is identical to the HORizontal:POSition command. When Horizontal Delay mode is turned off, this command is equivalent to adjusting the **HORIZONTAL POSITION** knob on the front-panel. When Horizontal Delay mode is turned on, this command is equivalent to selecting Horizontal/Acquisition from the Horiz/Acq menu and then choosing a Horizontal Ref Point value.

Group

Horizontal

Related Commands

HORizontal[:MAIn]:DELay:MODE

Syntax 1

```
HORizontal:MAIn:POSition <NR3>
```

Syntax 2

```
HORizontal:MAIn:POSition?
```

Arguments

- <NR3>

This is from 0 to ≈ 100 and is the percent of the waveform that is displayed left of the center graticule.

Note: The upper limit of the waveform position is slightly limited by a value that is determined from the record length (upper limit = $100 - 1/\text{record length}$).

Example 1

```
HORizontal:MAIn:POSition 5.000E+01
```

This command sets the horizontal position of the waveform such that 10 percent of the waveform is to the left of screen center.

Example 2

```
HORizontal:MAIn:POSition?
```

This query might return

```
:HORIZONTAL:MAIN:POSITION 5.0000E+01
```

indicating that the horizontal position of the waveform on the screen is currently set to 50 percent.

HORizontal:MAIn:SAMPLERate

Description

This command sets the horizontal sample rate to the desired number of samples per second. The record length is automatically adjusted at the same time to maintain a constant number of data points in each frame. The query form of this command returns the current horizontal sample rate. This command is equivalent to selecting Horizontal/Acquisition Setup from the Horiz/Acq menu and then entering the desired Sample Rate.

Group

Horizontal

Related Commands

HORizontal:RESOLution

Syntax 1

```
HORizontal:MAIn:SAMPLERate <NR3>
```

Syntax 2

```
HORizontal:MAIn:SAMPLERate?
```

Arguments

- <NR3>

This represents the size of the sample rate, which ranges from 5 S/s to 250 GS/s.

Example 1

```
HORizontal:MAIn:SAMPLERate 125E6
```

This command sets the sample rate to the sample rate increment that is closest to 125 MS/s.

Example 2

```
HORizontal:MAIn:SAMPLERate?
```

This query might return

```
:HORIZONTAL:MAIN:SAMPLERATE 2.5000E+09
```

indicating that the sample rate is currently set to 2.5 GS/s.

HORizontal[:MAIn]:SCAlE

Description

This command sets the time per division for the time base or returns its horizontal scale on the display and is identical to the HORizontal:SCAlE command. The specified scale value is rounded to a valid scale setting. This command is equivalent to selecting Position/Scale from the Horiz/Acq menu and then choosing a Scale value.

Group

Horizontal

Related Commands

HORizontal:SCAlE

Syntax 1

```
HORizontal:MAIn:SCAlE <NR3>
```

Syntax 2

```
HORizontal:MAIn:SCAlE?
```

Argument

- <NR3>

This is the time per division. The range is from 200 ps through 40 s.

Example 1

```
HORizontal:MAIn:SCAlE 2E-6
```

This command sets the main scale to 2 μ s per division.

Example 2

```
HORizontal:MAIn:SCAlE?
```

This query might return

```
:HORIZONTAL:MAIN:SCALE 2.0000E-06
```

indicating that the main scale is currently set to 2 μ s per division.

HORizontal:MAIn:UNIts

Description

This command sets or returns the units for the horizontal time base. It is equivalent to separately setting the HORizontal:MAIn:UNIts:STRing command.

Group

Horizontal

Related Commands

HORizontal:MAIn:UNIts:STRing

Syntax 1

```
HORizontal:MAIn:UNIts <string>
```

Syntax 2

```
HORizontal:MAIn:UNIts?
```

Arguments

- <string>

This is the time base units string.

Example 1

```
HORizontal:MAIn:UNIts "lum"
```

This command sets the units to "lum", indicating that the units' string for the time base is set to lumens.

Example 2

```
HORizontal:MAIn:UNIts?
```

This query might return

```
:HORIZONTAL:MAIN:UNITS:STRING "lum"
```

HORizontal:MAIn:UNIts:STRing

Description

This command sets or returns the units string for the horizontal time base trigger delay.

Group

Horizontal

Related Commands

HORizontal:MAIn:UNIts

Syntax 1

```
HORizontal:MAIn:UNIts:STRing <string>
```

Syntax 2

```
HORizontal:MAIn:UNIts:STRing?
```

Argument

- <string>

This is the time base units string.

Example 1

```
HORizontal:MAIn:UNIts:STRing lum
```

This command sets the units string for the time base trigger delay to lumens.

Example 2

```
HORizontal:MAIn:UNIts:STRing?
```

This query might return

```
:HORIZONTAL:MAIN:UNITS:STRING "lum"
```

indicating that the units string for the time base trigger delay is set to lumens.

HORizontal:POSition

Description

This command either positions the waveform horizontally on the display or returns its horizontal position on the display. When Horizontal Delay Mode is turned off, this command is equivalent to adjusting the **HORIZONTAL POSITION** knob on the front panel.

Group

Horizontal

Related Commands

HORizontal[:MAIn]:DELay:MODE

Syntax 1

HORizontal:POSition <NR3>

Syntax 2

HORizontal:POSition?

Argument

- <NR3>

This is from 0 to ≈ 100 and is the percent of the waveform that is displayed left of the center graticule.

Note: The upper limit of the waveform position is slightly limited by a value that is determined from the record length (upper limit = $100 - 1/\text{record length}$).

Example 1

HORizontal:POSition 10

This command sets the horizontal position of the waveform such that 10% of the waveform is to the left of screen center.

Example 2

HORizontal:POSition?

This query might return

:HORIZONTAL:POSITION 5.0000E+01

indicating the horizontal position of the waveform on the screen.

HORizontal:RECOrdlength

Description

This command sets the horizontal record length to the number of data points in each frame. The query form of this command returns the current horizontal record length. This command is equivalent to selecting Resolution from the Horiz/Acq menu and then entering the desired Rec Length.

Group

Horizontal

Syntax 1

```
HORizontal:RECOrdlength <NR1>
```

Syntax 2

```
HORizontal:RECOrdlength?
```

Argument

- <NR1>

This represents the supported values for horizontal record lengths, which ranges from 500 through the maximum record length. For more information about valid data point ranges, check your instrument specifications.

Example 1

```
HORizontal:RECOrdlength 50000
```

This command specifies that 50000 data points will be acquired for each record.

Example 2

```
HORizontal:RECOrdlength?
```

This query might return

```
:HORIZONTAL:RECOrdlength 5000
```

indicating that the horizontal record length is equal to 5000 data points.

HORizontal:RESOLution

Description

This command sets the horizontal record length to the number of data points in each frame. The sample rate is automatically adjusted at the same time to maintain a constant time per division. The query form of this command returns the current horizontal record length. This command is equivalent to adjusting the RESOLUTION knob on the front panel.

Group

Horizontal

Syntax 1

```
HORizontal:RESOLution <NR1>
```

Syntax 2

```
HORizontal:RESOLution?
```

Argument

- <NR1>

This represents the supported values for horizontal record lengths, which range from 500 through 400K data points. For more information about valid data point ranges, select Specifications from the Help menu and choose the Horizontal & Acquisition tab.

Example 1

```
HORizontal:RESOLution 50000
```

This command specifies that 50000 data points will be acquired for each record.

Example 2

```
HORizontal:RESOLution?
```

This query might return

```
:HORIZONTAL:RESOLUTION 5000
```

indicating that the horizontal record length is equal to 5000 data points.

HORizontal:ROLL

Description

This command sets or queries the Roll Mode status. Use Roll Mode when you want to view data at very slow sweep speeds. It is useful for observing data samples on the screen as they occur. This command is equivalent to selecting Horizontal/Acquisition Setup from the Horiz/Acq menu, selecting the Acquisition tab and then choosing the desired Roll Mode.

Group

Horizontal

Syntax 1

```
HORizontal:ROLL {AUTO|OFF|ON}
```

Syntax 2

```
HORizontal:ROLL?
```

Arguments

- AUTO

This enables Roll Mode, if the time/division is set appropriately.

- OFF

This disables Roll Mode.

- ON

This enables Roll Mode, if the time/division is set appropriately.

Example 1

```
HORizontal:ROLL ON
```

This command enables Roll Mode.

Example 2

```
HORizontal:RECOrdlength?
```

This query might return

```
:HORIZONTAL:ROLL OFF
```

indicating that the roll mode is disabled.

HORizontal:SCAlE

Description

This command sets or queries the time per division for the time base and is identical to the HORizontal:MAIn:SCAlE command. It is provided to maintain program compatibility with some older models of Tektronix oscilloscopes. This command is equivalent to selecting Position/Scale from the Horiz/Acq menu and then entering the desired Scale value.

Group

Horizontal

Related Commands

HORizontal:MAIn:SCAlE

Syntax 1

```
HORizontal:SCAlE <NR3>
```

Syntax 2

```
HORizontal:SCAlE?
```

Argument

- <NR3>

This is the time per division. The range is from 200 ps to 40 s.

Example 1

```
HORizontal:SCAlE 2E-6
```

This command sets the main scale to 2 μ s per division.

Example 2

```
HORizontal:SCAlE?
```

This query might return

```
:HORIZONTAL:SCALE 2.0000E-06
```

indicating that the time per division is currently set to 2 μ s,

HORizontal:TRIGger:POSition

Description

This command sets or queries the position of the trigger and is identical to the HORizontal:MAIn:POSition command. It is provided to maintain program compatibility with some older models of Tektronix oscilloscopes. This command is equivalent to selecting Position/Scale from the Horiz/Acq menu and then entering the desired Position.

Group

Horizontal

Related Commands

HORizontal:MAIn:POSition

Syntax 1

```
HORizontal:TRIGger:POSition <NR1>
```

Syntax 2

```
HORizontal:TRIGger:POSition?
```

Argument

- <NR1>

This is the amount of pre-trigger information in the waveform. The range is from 0 to 100%.

Example 1

```
HORizontal:TRIGger:POSition 50
```

This command sets the pre-trigger information in the waveform to 50%.

Example 2

```
HORizontal:TRIGger:POSition?
```

This query might return

```
:HORIZONTAL:TRIGGER:POSITION 50
```

HORizontal:TIMESTamp:CHX?

Description

This query-only command returns the absolute trigger date and time for the first frame on the selected channel (CH) waveform.

It is equivalent to: HORizontal:FASTFRAME:TIME-STamp:FRame:CHX? 1.

Group

Horizontal

Syntax

HORizontal:FASTframe:TIMEStamp:CH<wfm>?

Example

HORIZONTAL:FASTFRAME:TIMESTAMP:CH1?

This query might return

```
:HORIZONTAL:FASTFRAME:TIMESTAMP:CH1,"12  
Apr 2002 15:28:11.862 834 627 282"
```

which is the time stamp of channel 1, frame 1.

HORizontal:TIMESTamp:REFX?

Description

This query-only command returns the absolute trigger date and time for the first frame on the selected REF waveform.

It is equivalent to: HORizontal:FASTFRAME:TIMESTamp:FRame:REFX? 1.

Group

Horizontal

Syntax

```
HORizontal:FASTframe:TIMESTamp:REF<wfm>?
```

Example

```
HORIZONTAL:FASTFRAME:TIMESTAMP:FRAME:REF1?
```

This query might return

```
:HORIZONTAL:FASTFRAME:TIMESTAMP:FRAME:REF1,"02 Mar 2000 20:10:54.536  
036 928 432"
```

which is the time stamp of reference waveform 1, frame 1.

ID?

Description

This query-only command returns identifying information about the oscilloscope and related firmware.

Group

Miscellaneous

Related Commands

*IDN?

Syntax

ID?

Example

ID?

This query might return

```
:TEK/TDS7404,CF:91.1CT,FV:01.00.912
```

indicating the oscilloscope model number, configured format, and firmware version number.

LOCK

Description

This command enables or disables all front panel buttons and knobs, including the touch screen. There is no front panel equivalent.

Group

Miscellaneous

Related Commands

UNLock

Syntax 1

LOCK {ALL | NONE}

Syntax 2

LOCK?

Arguments

- ALL

This disables all front panel controls.

- NONE

This enables all front panel controls. This is equivalent to the UNLock ALL command.

Note: If the oscilloscope is in the Remote With Lockout State (RWLS), the LOCK NONE command has no effect. For more information, see the ANSI/IEEE Std 488.1-1987 Standard Digital Interface for Programmable Instrumentation, section 2.8.3 on RL State Descriptions.

Example 1

```
LOCK ALL
```

This command locks the front panel controls.

Example 2

```
LOCK?
```

This query might return

```
:LOCK NONE
```

indicating that the front panel controls are enabled by this command.

MASK?

Description

This query-only command returns the states of all settable mask parameters

Group

Mask

Syntax

MASK?

Example

MASK?

MASK:AUTOAdjust

Description

This command optimizes signal position within the mask to minimize hits. It sets a mode so that the MASK:SOURCE waveform is compared against the mask and is shifted up, down, left, or right to minimize the hits.

Group

Mask

Syntax

```
MASK:AUTOAdjust {ON|OFF|<NR1>}
```

Arguments

- ON

This enables the autoadjust function.

- OFF

This disables the autoadjust function

- <NR1>

A 0 disables the autoadjust function; any other value enables it.

Example

```
MASK:AUTOAdjust ON
```

Enables the autoadjust function

MASK:AUTOAdjust:HDELTA

Description

This command sets or returns how far autofit searches horizontally.

Autofit moves the waveform vertically and/or horizontally to reduce the number of hits within a given mask. If infinite or variable persistence is enabled, these movements will clear any persisted data. If Autofit makes frequent adjustments, there can be very little or even no persisted data displayed.

Group

Mask

Syntax

```
MASK:AUTOAdjust:HDELTA {<NR3>}
```

Argument

- <NR3>

Is a floating point number which represents a percent of a division.

Example

```
MASK:AUTOAdjust:HDELTA 0.6
```

Sets the horizontal autofit search to 0.6 division.

MASK:AUTOAdjust:VDELTA

Description

This command sets or returns how far autofit searches vertically.

Autofit moves the waveform vertically and/or horizontally to reduce the number of hits within a given mask. If infinite or variable persistence is enabled, these movements will clear any persisted data. If Autofit makes frequent adjustments, there can be very little or even no persisted data displayed.

Group

Mask

Syntax 1

```
MASK:AUTOAdjust:VDELTA {<NR3>}
```

Syntax 2

```
MASK:AUTOAdjust:VDELTA?
```

Argument

- <NR3>

Is a floating point number which represents a percent of a division.

Example

```
MASK:AUTOAdjust:VDELTA 0.6
```

Sets the vertical autofit search to 0.6 of a division.

MASK:AUTOSet:AUTOAdjust

Description

This command sets or returns whether the instrument settings are automatically matched to signal characteristics and specific mask requirements. It controls what happens at the end of a mask autoset. If ON, the horz/vert auto adjustment is run ONCE to potentially improve the result of mask autoset.

This command, like all the MASK:AUTOSET commands, affects only an autoset on a standard mask, not the general oscilloscope autoset function.

Group

Mask

Syntax 1

```
MASK:AUTOSet:AUTOAdjust {ON|OFF|<NR1>}
```

Syntax 2

```
MASK:AUTOSet:AUTOAdjust?
```

Arguments

- ON

This enables the autoset:autoadjust function.

- OFF

This disables the autoset:autoadjust function.

- <NR1>

A 0 disables the autoset:autoadjust function; any other value enables it.

Example

```
MASK:AUTOSet:AUTOAdjust ON
```

Enables the autoset autoadjust function

MASK:AUTOSet:HPOS

Description

This command controls whether the autosegment algorithm will attempt to change the horizontal position of the signal while attempting to autosegment. The default is ON.

This command, like all the MASK:AUTOSET commands, affects only an autosegment on a standard mask, not the general oscilloscope autosegment function.

Group

Mask

Syntax

```
MASK:AUTOSet:HPOS {ON|OFF|<NR1>}
```

Arguments

- ON

This enables the autosegment:hpos function, causing ... to occur.

- OFF

This disables the autosegment:hpos function, causing ... to occur.

- <NR1>

A 0 disables the autosegment:hpos function; any other value disables enables it.

Example

```
MASK:AUTOSet:HPOS ON
```

Enables the autosegment:hpos function

MASK:AUTOSet:HSCALE

Description

This command controls whether the autosegment algorithm will attempt to change the horizontal scale while attempting to autosegment.

This command, like all the MASK:AUTOSET commands, affects only an autosegment on a standard mask, not the general oscilloscope autosegment function.

Group

Mask

Syntax

```
MASK:AUTOSet:HSCALE {ON|OFF|<NR1>}
```

Arguments

- ON

This enables the autosegment:hscal function.

- OFF

This disables the autosegment:hscal function.

- <NR1>

A 0 disables the autosegment:hscal function; any other value enables it.

Example

```
MASK:AUTOSet:HSCALE ON
```

Enables the autosegment:hscal function.

MASK:AUTOSet:MODE

Description

This command controls whether or not an autoselect will be automatically done after a standard mask is selected. The autoselect will never happen if the standard mask is selected from GPIB, since a "SET?" sent back to the scope should not do a mask autoselect.

The default value is: manual.

This command, like all the MASK:AUTOSET commands, affects only an autoselect on a standard mask, not the general oscilloscope autoselect function.

Group

Mask

Syntax

```
MASK:AUTOSet:MODE {MANual|AUTO}
```

Argument

- MANual

This disables autoselect from occurring automatically after a standard mask is selected.

- AUTO

This enables autoselect to automatically occur after a standard mask is selected.

Example

```
MASK:AUTOSet:MODE AUTO
```

Enables autoselect to occur automatically after a standard mask is selected.

MASK:AUTOSet:OFFSETAdj

Description

This command sets autosec to not enforce the rule that, for pulse standards, require 0V to be in a certain place in the mask. Instead, autosec will attempt to measure the DC offset in the signal and use scope offset or position controls to compensate for the DC offset in the signal.

This command, like all the MASK:AUTOSET commands, affects only an autosec on a standard mask, not the general oscilloscope autosec function.

Group

Mask

Syntax 1

```
MASK:AUTOSet:OFFSETAdj {ON|OFF|<NR1>}
```

Syntax 2

```
MASK:AUTOSet:OFFSETAdj?
```

Arguments

- ON

This enables the autosec:offsetadj function.

- OFF

This disables the autosec:offsetadj function.

- <NR1>

A 0 disables the autosec:offsetadj function; any other value enables it.

Example

```
MASK:AUTOSET:OFFSETADJ ON
```

Enables the autosec:offsetadj function

MASK:AUTOSet:STANdard

(TDS7000 Series option SM & CSA7000 Series)

Description

This command selects the standard mask in autoselect.

This command, like all the MASK:AUTOSET commands, affects only an autoselect on a standard mask, not the general oscilloscope autoselect function.

Group

Mask

Syntax 1

```
MASK:AUTOSet:STANdard {G703D1
|E1Symmetrical|E1Coax
|DS2RATESymmetrical|DS2RATECoax|E2
|RATE32Mbit|E3|G703DS3|RATE97Mbit|E4_0
|E4_1|STM1E_0|STM1E_1|DS1|DS1A|DS1C
|DS2|DS3|DS4NA|DS4NA_Max|STS1Pulse
|STS1Eye|STS3|STS3_Max|ENET1000FX
|ENET100STP|ENET100UTP|ENET1250
|FC133|FC266|FC531|FC1063|FC1063Draft
|FC2125|FC133E|FC266E|FC531E|FC1063E
|FC1063E_NBT|FC1063E_NDT|FC1063E_NGT
|FC1063E_ABT|FC1063E_ADT|FC1063E_AGT
|FC1063E_ABR|FC1063E_ADR|FC1063E_AGR
|FC2125E_NBT|FC2125E_NDT|FC2125E_NGT
|FC2125E_ABT|FC2125E_ADT|FC2125E_AGT
|FC2125E_ABR|FC2125E_ADR|FC2125E_AGR
|OC1|OC3|OC12|OC48|OC48_FEC|INF2_5G
|INF2_5GE|ATATXG1|ATATXG2|ATATXG3|ATARXG1
|ATARXG2|ATARXG3|FST1|FST2|FST3|FST4|FST5
|FST6|HST1|HST2|HST3|HST4|HST5|HST6
|FW1394BS400BT1|FW1394BS400BT2
|FW1394BS800BT1|FW1394BS800BT2
|FW1394BS1600BT1|FW1394BS1600BT2
|FW1394BS400B|FW1394BS800B|FW1394BS1600B
|USERMask|ENET1000BCX_NTP2
|ENET1000BCX_ATP2|ENET1000BCX_ATP3
|ENETXAUI_Near|ENETXAUI_Far|RIO_DRV500M
|RIO_DRV750M|RIO_DRV1G|RIO_DRV1_5G
|RIO_DRV2G|RIO_EDRV500M|RIO_EDRV750M
|RIO_EDRV1G|RIO_EDRV1_5G|RIO_EDRV2G
|RIO_RCV500M|RIO_RCV750M|RIO_RCV1G
|RIO_RCV1_5G|RIO_RCV2G
|RIO_SERIAL_1G|RIO_SERIAL_2G|RIO_SERIAL_3G
```

```

|SFI5_XMITADATA2|SFI5_XMITCDATA2
|SFI5_XMITACLK2|SFI5_XMITCCLK2
|SFI5_RCVBDATA2|SFI5_RCVDDATA2
|SFI5_RCVBCLK2|SFI5_RCVDCCLK2
|SFI5_XMITADATA3|SFI5_XMITCDATA3
|SFI5_XMITACLK3|SFI5_XMITCCLK3
|SFI5_RCVBDATA3|SFI5_RCVDDATA3
|SFI5_RCVBCLK3|SFI5_RCVDCCLK3|VSROC192
|TFI5_2|TFI5_3|PCIEXPRESS_Xmit
|PCIEXPRESS_Rcv}

```

Syntax 2

MASK:AUTOSet:STANdard?

Arguments

All of the masks may not be available on every instrument.

- ATATXG1: Serial ATA, G1 Tx, 1.5 Gb/s
- ATATXG2: Serial ATA, G2 Tx, 3.0 Gb/s
- ATATXG3: Serial ATA, G3 Tx
- ATARXG1: Serial ATA, G1 Rx, 1.5 Gb/s
- ATARXG2: Serial ATA, G2, Rx, 3.0 Gb/s
- ATARXG3: Serial ATA, G3, Rx
- DS1: ANSI T1.102-1993 (R1999), DS1, 1.544 Mb/s
- DS1A: ANSI T1.102-1993 (R1999), DS1A, 2.048 Mb/s
- DS1C: ANSI T1.102-1993 (R1999), DS1C, 3.152 Mb/s
- DS2: ANSI T1.102-1993 (R1999), DS2, 6.312 Mb/s
- DS2RATECoax: ITU-T, G703 (10/98), D2 Rate Coax, 6.312 Mb/s
- DS2RATESymmetrical: ITU-T, G703 (10/98), D2 Rate Sym, 6.312 Mb/s
- DS3: ANSI T1.102-1993 (R1999), DS3, 44.736 Mb/s
- DS4NA: ANSI T1.102-1993 (R1999), DS4NA, 139.26 Mb/s
- DS4NA_Max: ANSI T1.102-1993 (R1999), DSNA Max Output, 139.26 Mb/s
- ElCoax: ITU-T, G703 (10/98), E1 Coax Pair, 2.048 Mb/s
- ElSymmetrical: ITU-T, G703 (10/98), E1 Sym Pair, 2.048 Mb/s

-
- E2: ITU-T, G703 (10/98), E2, 8.448 Mb/s
 - E3: ITU-T, G703 (10/98), E3, 34.368 Mb/s
 - E4_0: ITU-T, G703 (10/98), E4 Binary 0
 - E4_1: ITU-T, G703 (10/98), E4 Binary 1
 - ENET100FX: IEEE Std 802.3, X fiber over PMD, 100 Mb/s
 - ENET100STP: IEEE Std 802.3 and ANSI X3.263-1995, 100 Base-Tx, STP, 125 Mb/s
 - ENET100UTP: IEEE Std 802.3 and ANSI X3.263-1995, 100 Base-Tx, UTP, 125 Mb/s
 - ENET1000BCX_ATP2: 1000B-CX Abs, TP2, 1.25 Gb/s
 - ENET1000BCX_ATP3: 1000B-CX Abs, TP3, 1.25 Gb/s
 - ENET1000BCX_NTP2: 1000B-CX Norm, TP2, 1.25 Gb/s
 - ENET1250: IEEE Std 802.3 and ANSI X3.263-1995, GB Ethernet, 1.25 Gb/s
 - ENETXAUI_Far: 10 Gigabit Attachment Unit Interface (XAUI), Far, 3.125 Gb/s
 - ENETXAUI_Near: 10 Gigabit Attachment Unit Interface (XAUI), Near, 3,125 Gb/s
 - FC133: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 132.8 Mb/s
 - FC133E: ANSI X3.230-1999 NCITS 1235D/Rev 11, Electrical 132.8 Mb/s
 - FC266: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 265.6 Mb/s
 - FC266E: ANSI X3.230-1999 NCITS 1235D/Rev 11, Electrical, 132.8 Mb/s
 - FC531: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 531.2 Mb/s
 - FC531E: ANSI X3.230-1999 NCITS 1235D/Rev 11, Electrical, 531.2 Mb/s
 - FC1063: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 1.065 Gb/s
 - FC1063E: ANSI X3.230-1999 NCITS 1235D/Rev 11, Electrical, 1.0625 Gb/s
 - FC1063Draft: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, Draft Rev 11
 - FC1063E_ABR: ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Rev

-
- FC1063E_ABT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Transm
 - FC1063E_ADR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Delta, Recv
 - FC1063E_ADT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Transm
 - FC1063E_AGR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Recv
 - FC1063E_AGT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Transm
 - FC1063E_NBT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Beta, Transm
 - FC1063E_NDT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Delta, Transm
 - FC1063E_NGT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Gamma, Transm
 - FC2125: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 2.125 Gb/s
 - FC2125E_ABR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Recv
 - FC2125E_ABT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Transm
 - FC2125E_ADR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Delta, Recv
 - FC2125E_ADT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Delta, Transm
 - FC2125E_AGR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Recv
 - FC2125E_AGT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Transm
 - FC2125E_NBT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Beta, Transm
 - FC2125E_NDT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Delta, Transm
 - FC2125E_NGT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Gamma, Transm
 - FC4250E_ABR:
 - FC4250E_ABT:
 - FC4250E_ADR:
 - FC4250E_ADT:
 - FC4250E_AGR:

-
- FC4250E_AGT:
 - FC4250E_NBT:
 - FC4250E_NDT:
 - FC4250E_NGT:
 - FST1: USB, FS:T1, 12 Mb/s
 - FST2: USB, FS:T2, 12 Mb/s
 - FST3: USB, FS:T3, 12 Mb/s
 - FST4: USB, FS:T4: 12 Mb/s
 - FST5: USB, FS:T5, 12 Mb/s
 - FST6: USB, FS:T6, 12 Mb/s
 - FW1394BS400B: IEEE 1394b, S400 Optical, 491.5 Mb/s
 - FW1394BS400BT1: IEEE 1394b, S400b T1, 491.5 Mb/s
 - FW1394BS400BT2: IEEE 1394b, S400b T2, 491.5 Mb/s
 - FW1394BS800B: IEEE 1394b, S800 Optical, 988.0 Mb/s
 - FW1394BS800BT1: IEEE 1394b, S800b T1, 983.0 Mb/s
 - FW1394BS800BT2: IEEE 1394b, S800b T2, 983.0 Mb/s
 - FW1394BS1600B: IEEE 1394b, S1600 Optical, 1.966 Gb/s
 - FW1394BS1600BT1: IEEE 1394b, S1600b T1, 1.966 Gb/s
 - FW1394BS1600BT2: IEEE 1394b, S1600b T2, 1.966 Gb/s
 - G703D1: ITU-T, G703 (10/98), DS1 Rate, 1.544 Mb/s
 - G703DS3: ITU-T, G703 (10/98)
 - HST1: USB, HS:T1, 480 Mb/s
 - HST2: USB, HS:T2, 480 Mb/s
 - HST3: USB, HS:T3, 480 Mb/s
 - HST4: USB, HS:T4, 480 Mb/s
 - HST5: USB, HS:T5, 480 Mb/s
 - HST6: USB, HS:T6, 480 Mb/s
 - INF2_5G: InfiniBand, IBTA Spec 1.0a, 2.5 Optical, 2.5 Gb/s

-
- INF2_5GE: InfiniBand, IBTA Spec 1.0a, 2.5 Electrical, 2.5 Gb/s
 - OC1: GR 253-CORE Issue 3 9/21/2000 OC1/STM0, 51.84 Mb/s
 - OC3: GR 253-CORE Issue 3 9/21/2000 OC1/STM1, 155.52, Mb/s
 - OC12: GR 253-CORE Issue 3 9/21/2000 OC1/STM4, 622.08 Mb/s
 - OC48: GR 253-CORE Issue 3 9/21/2000 OC1/STM16, 2.4883 Gb/s
 - OC48_FEC: Forward Error Correction - CSA8000 mask, 2.666 Gb/s
 - PCIEXPRESS_Rcv: PCI Express Receiver, 2.5 Gb/s
 - PCIEXPRESS_Xmit: PCI Express Transmitter, 2.5 Gb/s
 - RATE32Mbit: ITU-T, G703 (10/98), 32.064 Mb/s
 - RATE97Mbit: ITU-T, G703 (10/98), 97 Mbit, 97.728 Mb/s
 - RIO_DRV1G: Rapid IO Driver, 1 Gb/s
 - RIO_DRV1_5G: Rapid IO Driver, 5 Gb/s
 - RIO_DRV2G: Rapid IO Driver, 2 Gb/s
 - RIO_DRV500M: Rapid IO Driver, 500 Mb/s
 - RIO_DRV750M: Rapid IO Driver, 750 Mb/s
 - RIO_EDRV1G: Rapid IO Extended Driver, 1 Gb/s
 - RIO_EDRV1_5G: Rapid IO Extended Driver, 1.5 Gb/s
 - RIO_EDRV2G: Rapid IO Extended Driver, 2 Gb/s
 - RIO_EDRV500M: Rapid IO Extended Driver, 500 Mb/s
 - RIO_EDRV750M: Rapid IO Extended Driver, 750 Mb/s
 - RIO_RCV1G: Rapid IO Receiver, 1 Gb/s
 - RIO_RCV1_5G: Rapid IO Receiver, 1.5 Gb/s
 - RIO_RCV2G: Rapid IO Receiver, 2 Gb/s
 - RIO_RCV500M: Rapid IO Receiver, 500 Mb/s
 - RIO_RCV750M: Rapid IO Receiver, 750 Mb/s

- RIO_SERIAL_1G: Rapid IO Serial, 1.25 Gb/s
- RIO_SERIAL_2G: Rapid IO Serial, 2.5 Gb/s
- RIO_SERIAL_3G: Rapid IO Serial, 3.25 Gb/s
- SFI5_RCVBCLK2: SFI15 Receive - Test Point B Clock Signal 2, 2.488 Gb/s
- SFI5_RCVBCLK3: SFI15 Receive - Test Point B Clock Signal 3, 3.125 Gb/s
- SFI5_RCVBDATA2: SFI15 Receive - Test Point B Data Signal 2, 2.488 Gb/s
- SFI5_RCVBDATA3: SFI15 Receive - Test Point B Data Signal 3, 3.125 Gb/s
- SFI5_RCVDCLK2: SFI15 Receive - Test Point D Clock Signal 2, 2.488 Gb/s
- SFI5_RCVDCLK3: SFI15 Receive - Test Point D Clock Signal 3, 3.125 Gb/s
- SFI5_RCVDDATA2: SFI15 Receive - Test Point D Data Signal 2, 2.488 Gb/s
- SFI5_RCVDDATA3: SFI15 Receive - Test Point D Data Signal 3, 3.125 Gb/s
- SFI5_XMITACLK2: SFI15 Transmit - Test Point A Clock Signal 2, 2.488 Gb/s
- SFI5_XMITACLK3: SFI15 Transmit - Test Point A Clock Signal 3, 3.125 Gb/s
- SFI5_XMITADATA2: SFI15 Transmit - Test Point A Data Signal 2, 2.488 Gb/s
- SFI5_XMITADATA3: SFI15 Transmit - Test Point A Data Signal 3, 3.125 Gb/s
- SFI5_XMITCCLK2: SFI15 Transmit - Test Point C Clock Signal 2, 2.488 Gb/s
- SFI5_XMITCCLK3: SFI15 Transmit - Test Point C Clock Signal 3, 3.125 Gb/s
- SFI5_XMITCDATA2: SFI15 Transmit - Test Point C Data Signal 2, 2.488 Gb/s
- SFI5_XMITCDATA3: SFI15 Transmit - Test Point C Data Signal 3, 3.125 Gb/s
- STM1E_0: ITU-T, G703 (10/98), STM1E Binary 0
- STM1E_1: ITU-T, G703 (10/98), STM1E Binary 1
- STS1Pulse: ANSI T1.102-1993 (R1999), STS-1 Pulse, 51.84 Mb/s

-
- STS1Eye: ANSI T1.102-1993 (R1999), STS-1 Eye, 51.84 Mb/s
 - STS3: ANSI T1.102-1993 (R1999), STS-3, 155.52 Mb/s
 - STS3_Max: ANSI T1.102-1993 (R1999), STS-3 Max Output, 155.52 Mb/s
 - TFI15_2: TFI-5, 2.488 Gb/s
 - TFI5_3: TFI-5, 3.1104 Gb/s
 - USERMask
 - VSROC192: VSR OC192/STM64, 1.24416 Gb/s

Example 1

`MASK:AUTOSET:STANDARD E1COAX`

Selects the E1 Coax standard mask.

Example 2

`MASK:AUTOSET:STANDARD?`

Might return `:MASK:AUTOSET:STANDARD E1COAX`

Indicating that the E1 Coax mask is selected as the current mask standard.

MASK:AUTOSet:TRIGger

Description

This command controls if a MASK:AUTOSet affects the trigger level. Other trigger settings such as type of trigger are not changed by MASK:AUTOSet. The default is ON.

This command, like all the MASK:AUTOSET commands, affects only an autoselect on a standard mask, not the general oscilloscope autoselect function.

Group

Mask

Syntax

```
MASK:AUTOSet:TRIGger {ON|OFF|<NR1>}
```

Arguments

- ON

This enables the autoselect:hpos function.

- OFF

This disables the autoselect:hpos function.

- <NR1>

A 0 disables the autoselect:trigger function; any other value enables it.

Example

```
MASK:AUTOSet:HPOS ON
```

Enables the autoselect:hpos function

MASK:AUTOSet:VPOS

Description

This command controls whether the autosegment algorithm will affect the vertical position (or offset) of the signal while attempting to autosegment. The default is ON.

This command, like all the MASK:AUTOSET commands, affects only an autosegment on a standard mask, not the general oscilloscope autosegment function.

Group

Mask

Syntax

```
MASK:AUTOSet:VPOS {ON|OFF|<NR1>}
```

Arguments

- ON

This enables the autosegment:vpos function.

- OFF

This disables the autosegment:vpos function.

- <NR1>

A 0 disables the autosegment:vpos function; any other value enables it.

Example

```
MASK:AUTOSet:VPOS ON
```

Enables the autosegment:vpos function

MASK:AUTOSet:VSCALE

Description

This command controls whether the autosegment algorithm will affect the vertical scale while attempting to autosegment.

This command, like all the MASK:AUTOSET commands, affects only an autosegment on a standard mask, not the general oscilloscope autosegment function.

Group

Mask

Syntax

```
MASK:AUTOSet:VSCALE {ON|OFF|<NR1>}
```

Arguments

- ON

This enables the autosegment:vscale function, causing ... to occur.

- OFF

This disables the autosegment:vscale function, causing ... to occur.

- <NR1>

A 0 disables the autosegment:vscale function; any other value enables it.

Example

```
MASK:AUTOSet:VSCALE ON
```

Enables the autosegment:vscale function.

MASK:COpy:USER

Description

This command copies the currently displayed mask to the User mask.

Group

Mask

Syntax

`MASK:COpy:USER`

Example

`MASK:COpy:USER`

Copies the current mask to the user mask.

MASK:COUNT:FAILURES?

Description

This command returns the number of pass/fail tests that have failed. The default is 0. To work pass/fail testing must be enabled.

Group

Mask

Syntax

```
MASK:COUNT:FAILURES?
```

Returns

```
<NR1>
```

Example

```
MASK:COUNT:FAILURES?
```

Might return 0 showing no tests have failed.

MASK:COUNT:HITS?

Description

This command returns the number of hits on all mask segments. To work, hit counting must be turned on.

Group

Mask

Syntax

```
MASK:COUNT:HITS?
```

Returns

```
<NR1>
```

Example

```
MASK:COUNT:HITS?
```

Might return: 0

Showing there were no hits from any source on all mask segments.

MASK:COUNT:RESET

Description

This command resets to zero the number of hits for all mask segments on the specified source.

Group

Mask

Syntax

```
MASK:COUNT:RESET
```

Example

```
MASK:COUNT:RESET
```

Resets to zero the number of hits for all mask segments on the specified source.

MASK:COUNT:SEG<m>:HITS?

Description

This command returns the number of hits in mask segment <m>. To work, hit counting must be turned on.

Group

Mask

Syntax

```
MASK:COUNT:SEG<m>:HITS?
```

Argument

- <m>

is the mask segment number.

Returns

```
<NR1>
```

Example

```
MASK:COUNT:SEG1:HITS?
```

Might return 0

showing there have been no hits.

MASK:COUNT:STATE

Description

Sets or returns the mask hits count state. In other words, it controls whether or not mask counting is being done.

MASK:DISplay must be ON to enable MASK:COUNT:STATE to count mask violations.

Group

Mask

Syntax

```
MASK:COUNT:STATE {OFF | ON | <NR1>}
```

Arguments

- ON

Turns on mask counting.

- OFF

Turns off mask counting. This is the default state.

- <NR1>

A 0 turns off mask hit counting, and other values turn on mask hit counting.

Example

```
MASK:COUNT:STATE ON
```

Turns on mask hit counting

MASK:COUNT:TESTS?

Description

Returns the number of pass/fail tests that have occurred.

Group

Mask

Syntax

```
MASK:COUNT:TESTS?
```

Returns

```
<NR1>
```

Example

```
MASK:COUNT:TESTS?
```

Might return: 50

showing 50 tests have occurred.

MASK:COUNT:TOTAl?

Description

Returns the sum of all hits in all mask segments. This command is the same as MASK:COUNT:HITS? And is kept for compatibility with other Tektronix oscilloscopes.

Group

Mask

Syntax

```
MASK:COUNT:TOTAl?
```

Returns

<NR1>

Example

```
MASK:COUNT:TOTAL?
```

Might return 50

Showing 50 hits have occurred.

MASK:COUNT:VIOLATIONS?

Description

This command returns the number of test violations that have occurred in the current pass/fail test. A test violation occurs when any part of a waveform falls within any mask segment. The default is 0.

Group

Mask

Related Commands

MASK:COUNT:TESTS?, MASK:COUNT:TOTAL?,
MASK:COUNT:WAVEFORMS?, MASK:TEST:REPEAT

Syntax

MASK:COUNT:VIOLATIONS?

Returns

<NR1>

Example

MASK:COUNT:VIOLATIONS?

Might return: 50

Showing 50 test violations have occurred.

MASK:COUNT:WAVEFORMS?

Description

This command returns the number of waveforms that have been acquired and processed during pass/fail testing.

Group

Mask

Related Commands

MASK:COUNT:TESTS?, MASK:COUNT:TOTAL?,
MASK:COUNT:VIOLATIONS?

Syntax

MASK : COUNT : WAVEFORMS ?

Returns

<NR1>

Example

MASK : COUNT : WAVEFORMS ?

Might return: 1000

Showing 1,000 waveforms have been acquired and processed.

MASK:DISplay

Description

Sets or returns whether or not defined masks are displayed on the screen. This is useful for temporarily turning off user defined masks without deleting them. It is also useful for removing a standard mask from the screen, but leaving it as the selected standard. Mask counting, mask testing, and mask autoselect are not available if the mask display is Off. The default is On.

Group

Mask

Syntax 1

```
MASK:DISplay {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:DISplay?
```

Arguments

- ON

Shows the masks on the display. This is the default value.

- OFF

Removes the masks from the display

- <NR1>

A 0 removes the masks from the display; any other value shows the masks on the display.

Example

```
MASK:DISPLAY ON
```

Sets the display to show the defined masks

Example 2

```
MASK:DISPLAY?
```

Might return 1, indicating that the display shows masks

MASK:FILTER

Description

This command sets or returns whether the selected digital filter will be run on the waveform data. The filter simulates optical hardware. That is, it simulates different hardware for each of several different optical standards.

The digital filter runs on OC1, OC3, OC12, OC48, FC133, FC266, FC531, FC1063, FC2125Draft, Gigabit Ethernet, Infiniband 2.5Gb, 1394b, 393Mb, 786.4 3Mb, 1.572 Gb

Group

Mask

Related Commands

ACQUIRE:MODE

Syntax

MASK:FILTER {OFF|ON|<NR1>}

Arguments

- ON

This enables the digital filter.

- OFF

This disables the digital filter.

- <NR1>

A 0 disables the digital filter; any other value enables it.

Example

MASK:FILTER ON

Enables the digital filter.

MASK:HIGHLIGHTHits

Description

This command sets or returns whether or not hits in a mask are highlighted in different colors than other waveform data. The default is On.

Group

Mask

Syntax 1

```
MASK:HIGHLIGHTHits {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:HIGHLIGHTHits?
```

Arguments

- ON

This enables the mask:highlighthits function.

- OFF

This disables the mask:highlighthits function.

- <NR1>

A 0 disables the mask:highlighthits function; any other value enables it.

Example

```
MASK:HIGHLIGHTHITS ON
```

Enables the mask highlighthits function.

MASK:INVert

Description

This command controls whether the mask is drawn inverted. It has no effect if this mask cannot be inverted. The default is Off (Positive).

Group

Mask

Syntax

```
MASK:INVert {OFF|ON|<NR1>}
```

Arguments

- ON

Negative

- OFF

Positive

- <NR1>

A 0 disables the mask:invert function; any other value enables it.

Example

```
MASK:INVERT ON
```

Enables the mask invert function.

MASK:LOCK

Description

This command sets or returns the mask lock to waveform mode. It locks the mask to the waveform such that changes to the horizontal and/or vertical scale settings re-draws the mask segments in proportion. This feature allows for expanding horizontal and/or vertical settings to zoom in on waveforms and masks to visually examine violation areas in more detail.

Group

Mask

Syntax 1

```
MASK:LOCK {OFF | ON | <NR1>}
```

Syntax 2

```
MASK:LOCK?
```

Arguments

- ON

turns on mask lock to waveform.

- OFF

turns off mask lock to waveform.

- <NR1>

A 0 turns off mask lock to waveform and any other value turns on mask lock to waveform.

Example

```
MASK:LOCK ON
```

Turns on mask lock to waveform

MASK:MARgin:PERCent

Description

This command sets or queries the mask margin percentage. If you set a positive mask margin percentage, the polygon expands and the mask test will be more difficult to pass due to the expanded test area. If you set a negative mask margin percentage, the polygon shrinks and the mask test will be easier to pass due to the smaller area.

Group

Mask

Syntax 1

```
MASK:MARgin:PERCent <NR3>
```

Syntax 2

```
MASK:MARgin:PERCent?
```

Argument

- <NR3>

This values range from -50.0 to +50.0. The default is 5.

Example 1

```
MASK:MARGIN:PERCENT 25
```

Sets the mask margin percentage to 25.

Example 2

```
MASK:MARGIN:PERCENT?
```

```
Might return :MASK:MARGIN:PERCENT  
1.00000000000E+001,
```

indicating that the mask margin is set to 10%

MASK:MARgin:STATE

Description

This command sets or returns the state of the mask margins.

Group

Mask

Syntax

```
MASK:MARgin:STATE {OFF | ON | <NR1>}
```

Arguments

- ON

Turns on mask margins.

- OFF

Turns off mask margins. The currently displayed margined mask is erased and the original mask is displayed.

- <NR1>

A 0 turns off mask margins; any other integer turns on the selected mask margins.

Example 1

```
MASK:MARGIN:STATE ON
```

Turns on mask margins

Example 2

```
MASK:MARGIN:STATE?
```

Might return :MASK:MARGIN:STATE 1, indicating that mask margins are on.

MASK:MASKPRE:AMPlitude

Description

This command sets or returns the current mask's nominal pulse amplitude in volts. The query form of this command returns the nominal pulse amplitude of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:VSCALE

Syntax 1

MASK:MASKPRE:AMPlitude <NR3>

Syntax 2

MASK:MASKPRE:AMPlitude?

Argument

- <NR3>

This is a floating number that sets the nominal pulse amplitude in volts

Example

```
MASK:MASKPRE:AMPLITUDE 2.4
```

Sets the current mask nominal pulse amplitude to 2.4 volts.

MASK:MASKPRE:HSCALE

Description

This command sets or returns the nominal timing resolution used to draw the mask, in time/division. The query form of this command returns the nominal timing resolution of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:HTRIGPOS

Syntax 1

MASK:MASKPRE:HSCALE <NR3>

Syntax 2

MASK:MASKPRE:HSCALE?

Argument

- <NR3>

This is a floating point number that sets the mask drawing timing resolution.

Example

MASK:MASKPRE:HSCALE 40e-9

Sets the nominal timing resolution used to draw the mask pulse shape to 40 nanoseconds per division.

MASK:MASKPRE:HTRIGPOS

Description

This command sets or returns the nominal trigger position.(pulse leading edge) used to draw the mask as a fraction of the display width. The query form of this command returns the nominal trigger position of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:HSCALE?

Syntax 1

MASK:MASKPRE:HTRIGPOS <NR3>

Syntax 2

MASK:MASKPRE:HTRIGPOS?

Argument

- <NR3>

Is a floating point number in the range of 0.0 to 1.0 that sets the trigger points as a fraction of the display width, referenced from the left edge of the graticule.

0.0 represents the left edge.

Example

MASK:MASKPRE:HTRIGPOS 0.6

Sets the current mask nominal pulse leading edge position to the 6th horizontal graticule division.

MASK:MASKPRE:PATTERNBITS

Description

This command sets or returns the number of bits used for serial trigger for the User mask standard. For example, DS1 requires six bits, four leading zeros, a one, and a trailing zero. The query form of this command returns the serial bit value of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:PRESAMPBITS

Syntax 1

MASK:MASKPRE:PATTERNBITS <NR1>

Syntax 2

MASK:MASKPRE:PATTERNBITS?

Argument

- <NR1>

Is an integer that sets the number of bits.

Example

MASK:MASKPRE:PATTERNBITS 6

Sets the current mask number of serial pattern bits to 6.

MASK:MASKPRE:PRESAMPBITS

Description

This command sets or returns the number of bits before the (isolated one) pulse leading edge in the serial trigger pass/fail testing. For example, DS1 has four leading zeros. The query form of this command returns the presample bit value of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:PATTERNBITS

Syntax 1

MASK:MASKPRE:PRESAMPBITS <NR1>

Syntax 2

MASK:MASKPRE:PRESAMPBITS?

Argument

- <NR1>

Is an integer that sets the number of bits before the trigger pulse.

Example

MASK:MASKPRE:PRESAMPBITS 4

Sets the current mask number of bits before the trigger pulse to 4.

MASK:MASKPRE:RECOrdlength

Description

This command sets or returns the nominal record length for pulse mask testing. The query form of this command returns the record length value of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Syntax

MASK:MASKPRE:RECOrdlength <NR1>

Syntax

MASK:MASKPRE:RECOrdlength?

Argument

- <NR1>

Is an integer number that sets the record length.

Example

MASK:MASKPRE:RECORDLENGTH 500

Sets the current mask record length to 500 points.

MASK:MASKPRE:SERIALTRIG

Description

This command sets or returns the type of triggering used in pass/fail testing of the current mask.

Group

Mask

Related Commands

MASK:MASKPRE:PATTERNBITS,
MASK:MASKPRE:PRESAMPBITS

Syntax 1

```
MASK:MASKPRE:SERIALTRIG {AMI | HDB3 | B3ZS |  
B6ZS | B8ZS | CMI | NRZ | MLT3 | EDGE}
```

Syntax 2

```
MASK:MASKPRE:SERIALTRIG?
```

Argument

- AMI

Alternate Mark Inversion

- HDB3

High-Density Bipolar Three-Bit substitution

- B3ZS

Bipolar 3 Zero Substitution

- B6ZS

Bipolar 6 Zero Substitution

- B8ZS

Bipolar 8 Zero Substitution

- CMI

Coded Mark Inversion

- NRZ

Non-Return to Zero

- MLT3

Multi-Level Transition

- EDGE

Example

`MASK:MASKPRE:SERIALTRIG AMI`

Sets AMI as the type of triggering used in testing the current mask.

MASK:MASKPRE:TRIGTOSAMP

Description

This command sets or returns the time, in seconds, from the (leading edge) trigger position to the pulse bit sampling position. The query form of this command returns the time value of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:WIDTh

Syntax 1

MASK:MASKPRE:TRIGTOSAMP <NR3>

Syntax 2

MASK:MASKPRE:TRIGTOSAMP?

Argument

- <NR3>

Is the floating point number that sets the time to the pulse bit sampling position.

Example

MASK:MASKPRE:TRIGTOSAMP 162e-9

Sets the current mask time from the (leading edge) trigger position to the pulse bit sampling position to 162 nanoseconds.

MASK:MASKPRE:VOFFSet

Description

This command sets or returns the nominal vertical offset, in volts, used to vertically offset the input channels. The query form of this command returns the offset value of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:VPOS

MASK:MASKPRE:VSCALE

Syntax 1

MASK:MASKPRE:VOFFSet <NR3>

Syntax 2

MASK:MASKPRE:VOFFSet?

Argument

- <NR3>

Is a floating point number that sets the nominal vertical offset in volts.

Example

MASK:MASKPRE:VOFFSET 0.0

Sets the current mask nominal vertical offset to 0.0 volts.

MASK:MASKPRE:VPOS

Description

This command sets or returns the nominal vertical position, in divisions, used to vertically position the input channels. The query form of this command returns the vertical position value of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:VOFFSet

MASK:MASKPRE:VSCALE

Syntax 1

MASK:MASKPRE:VPOS <NR3>

Syntax 2

MASK:MASKPRE:VPOS?

Argument

- <NR3>

Is a floating point number that sets the nominal vertical position in divisions.

Example

MASK:MASKPRE:VPOS -0.8

Sets the current mask nominal vertical position to -0.8 divisions.

MASK:MASKPRE:VSCALE

Description

This command sets or returns the nominal vertical scale, in volts per division, used to vertically scale the input channels. The query form of this command returns the vertical scale value of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:VOFFSet

MASK:MASKPRE:VPOS

Syntax 1

MASK:MASKPRE:VSCALE <NR3>

Syntax 2

MASK:MASKPRE:VSCALE?

Argument

- <NR3>

Is a floating point number that sets the nominal bit width in seconds.

Example

MASK:MASKPRE:VSCALE 200e-3

Sets the current mask nominal vertical scale to 200 millivolts per division

MASK:MASKPRE:WIDTH

Description

This command sets or returns the nominal bit width in seconds. The query form of this command returns the bit width value of the displayed mask. The set form of this command only affects the current (displayed) mask.

Group

Mask

Related Commands

MASK:MASKPRE:TRIGTOSAMP

Syntax 1

MASK:MASKPRE:WIDTH <NR3>

Syntax 2

MASK:MASKPRE:WIDTH?

Argument

- <NR3>

Is a floating point number that sets the nominal bit width in seconds.

This is the time of one bit of data where $\text{bit width} = 1 / (\text{data rate of the signal})$.

Example

MASK:MASKPRE:WIDTH 648e-9

Sets the current mask nominal bit width to 648 nanoseconds (1.544 Mbits/second).

MASK:POLarity

Description

This command sets or returns the input waveform polarity for the pass/fail test. It controls whether to test positive pulse, negative pulse, or both during pass/fail testing. This command only applies when MASK:TEST:STATE is on.

Group

Mask

Related Commands

MASK:TEST:STATE

Syntax 1

MASK:POLarity {BOTH|NEGAtive|POSITIVe}

Syntax 2

MASK:POLarity?

Arguments

- BOTH

This enables testing for both positive and negative pulses. The oscilloscope tests positive pulses on the mask:source waveform until $\frac{1}{2}$ of the waveform has been tested. Then the oscilloscope inverts the mask and performs the remaining tests.

- NEGAtive

This enables testing on negative pulses

- POSITIVe

This enables testing on positive pulses. This is the default.

Example

```
MASK:POLARITY BOTH
```

Sets pass/fail to test both positive (normal) and negative (inverted) waveforms.

MASK:SEG<m> DELEte

Description

This command deletes the specified mask segment from the Current mask.

Group

Mask

Syntax

```
MASK:SEG<m> DELEte
```

Arguments

- <m>

Is an integer that specifies the mask segment number to delete from the Current mask.

Example

```
MASK:SEG3 DELETE
```

Deletes mask segment three from the Current mask.

MASK:SEG<m>:NR_Pt?

Description

This command returns the number of points that make up the specified mask segment of the Current mask. Each mask point consists of an X-Y pair of coordinates.

Group

Mask

Syntax

`MASK:SEG<m>:NR_Pt?`

Arguments

- <m>

Is an integer number that specifies a mask segment number of the Current mask.

Returns

<NR1> is 0 to 50.

Example

`MASK:SEG2:NR_PT?`

Might return `MASK:SEG2:NR_PT 5`

MASK:SEG<m>:POINTS

Description

This command sets or returns the X-Y user coordinates of all points in the Current mask segment. The set form defines new points in the Current mask, replacing any existing points in the Current mask segment. In other words, it sets or returns the vertices for a particular segment in the Current mask.

Group

Mask

Syntax 1

```
MASK:SEG<m>:POINTS <NR3>, <NR3> [, <NR3>,  
<NR3>]
```

Syntax 2

```
MASK:SEG<m>:POINTS?
```

Arguments

- <m>

Is an integer that specifies the Current mask segment number.

- <NR3>

Refers to the coordinates of one of the vertices in the Current mask. Each pair of numbers is the horizontal and vertical coordinates of a mask segment vertex. The pairs must be listed in a counterclockwise order. If the vertical or horizontal scale or position is changed after this command and then the query form of this command is issued, then the value returned from the oscilloscope will not be the same. If just one pair is input then they are ignored and the Current mask segment is marked as undefined. The default is no points in the Current mask segment.

Example

```
MASK:SEG2:POINTS -2.3e-9, 44e-3, -2.5e-9,  
47e-3, 1.2e-9, 44e-3
```

Defines the points in Current mask segment 2.

MASK:SOURce

Description

This command sets or reports which source will be compared against the mask(s) when counting is turned on. In other words, it controls which trace to use in mask counting. It also affects mask autoselect and how triggering is set up when you select the mask.

Group

Mask

Syntax 1

```
MASK:SOURce {CH[ 1-4 ] | MATH[ 1-4 ] | REF[ 1-4 ] }
```

Syntax 2

```
MASK:SOURce?
```

Arguments

- CH<1-4>

Selects a channel waveform to be compared against the specified mask. The range for is 1 through 4.

- MATH<1-4>

Selects a math waveform to be compared against the specified mask. The range for is 1 through 4.

- REF<1-4>

Selects a reference waveform to be compared against the specified mask. The range is 1 through 4.

Example 1

```
MASK:SOURCE CH1
```

Enables CH1 to be compared against the mask. Since a timebase is not specified, the default, Main, is used.

Example 2

```
MASK:SOURCE?
```

Might return :MASK:SOURCE CH1,MAIN, indicated that the Channel 1 waveform will be compared against the mask using the Main timebase.

MASK:STANdard

(TDS7000 Series option SM & CSA7000 Series)

Description

This command deletes the existing mask and sets the selected standard mask. If MASK:COUNT:STATE is ON, then mask counting starts. The query form of this command returns the current mask standard.

Group

Mask

Related Commands

ACQuire:MODE

Syntax 1

```
MASK:STANdard {G703D1|E1Symmetrical|E1Coax
|DS2RATESymmetrical|DS2RATCoax|E2
|RATE32Mbit|E3|G703DS3|RATE97Mbit|E4_0
|E4_1|STM1E_0|STM1E_1|DS1|DS1A|DS1C|DS2
|DS3|DS4NA|DS4NA_Max|STS1Pulse
|STS1Eye|STS3|STS3_Max|ENET1000FX
|ENET100STP|ENET100UTP|ENET1250|FC133
|FC266|FC531|FC1063|FC1063Draft|FC2125
|FC133E|FC266E|FC531E|FC1063E|FC1063E_NBT
|FC1063E_NDT|FC1063E_NGT|FC1063E_ABT
|FC1063E_ADT|FC1063E_AGT|FC1063E_ABR
|FC1063E_ADR|FC1063E_AGR|FC2125E_NBT
|FC2125E_NDT|FC2125E_NGT|FC2125E_ABT
|FC2125E_ADT|FC2125E_AGT|FC2125E_ABR
|FC2125E_ADR|FC2125E_AGR|OC1|OC3|OC12
|OC48|OC48_FEC|INF2_5G|INF2_5GE|ATATXG1
|ATATXG2|ATATXG3|ATARXG1|ATARXG2|ATARXG3
|FST1|FST2|FST3|FST4|FST5|FST6|HST1|HST2
|HST3|HST4|HST5|HST6|FW1394BS400BT1
|FW1394BS400BT2|FW1394BS800BT1
|FW1394BS800BT2|FW1394BS1600BT1
|FW1394BS1600BT2|FW1394BS400B|FW1394BS800B
|FW1394BS1600B|USERMask|ENET1000BCX_NTP2
|ENET1000BCX_ATP2|ENET1000BCX_ATP3
|ENETXAUI_Near|ENETXAUI_Far|RIO_DRV500M
|RIO_DRV750M|RIO_DRV1G|RIO_DRV1_5G
|RIO_DRV2G|RIO_EDRV500M|RIO_EDRV750M
|RIO_EDRV1G|RIO_EDRV1_5G|RIO_EDRV2G
|RIO_RCV500M|RIO_RCV750M|RIO_RCV1G
|RIO_RCV1_5G|RIO_RCV2G|RIO_SERIAL_1G
|RIO_SERIAL_2G|RIO_SERIAL_3G
```

```

|SFI5_XMITADATA2|SFI5_XMITCDATA2
|SFI5_XMITACLK2|SFI5_XMITCCLK2
|SFI5_RCVBDATA2|SFI5_RCVDDATA2
|SFI5_RCVBCLK2|SFI5_RCVDCLK2
|SFI5_XMITADATA3|SFI5_XMITCDATA3
|SFI5_XMITACLK3|SFI5_XMITCCLK3
|SFI5_RCVBDATA3|SFI5_RCVDDATA3
|SFI5_RCVBCLK3|SFI5_RCVDCLK3|VSROC192
|TFI5_2|TFI5_3|PCIEXPRESS_Xmit
|PCIEXPRESS_Rcv}

```

Syntax 2

MASK:STANdard?

Arguments

All of the masks may not be available on every instrument.

- ATATXG1: Serial ATA, G1 Tx, 1.5 Gb/s
- ATATXG2: Serial ATA, G2 Tx, 3.0 Gb/s
- ATATXG3: Serial ATA, G3 Tx
- ATARXG1: Serial ATA, G1 Rx 1.5 Gb/s
- ATARXG2: Serial ATA, G2, Rx, 3.0 Gb/s
- ATARXG3: Serial ATA, G3, Rx
- DS1: ANSI T1.102-1993 (R1999), DS1, 1.544 Mb/s
- DS1A: ANSI T1.102-1993 (R1999), DS1A, 2.048 Mb/s
- DS1C: ANSI T1.102-1993 (R1999), DS1C, 3.152 Mb/s
- DS2: ANSI T1.102-1993 (R1999), DS2, 6.312 Mb/s
- DS2RATECoax: ITU-T, G703 (10/98), D2 Rate, Coax, 6.312 Mb/s
- DS2RATESymmetrical: ITU-T, G703 (10/98), D2 Rate Sym, 6.312 Mb/s
- DS3: ANSI T1.102-1993 (R1999), DS3, 44.736 Mb/s
- DS4NA: ANSI T1.102-1993 (R1999), DS4NA, 139.26 Mb/s
- DS4NA_Max: ANSI T1.102-1993 (R1999), DSNA Max Output, 139.26 Mb/s
- ElCoax: ITU-T, G703 (10/98), E1 Coax Pair, 2.048 Mb/s
- ElSymmetrical: ITU-T, G703 (10/98), E1 Sym Pair, 2.048 Mb/s

-
- E2: ITU-T, G703 (10/98), E2, 8.448 Mb/s
 - E3: ITU-T, G703 (10/98), E3, 34.368 Mb/s
 - E4_0: ITU-T, G703 (10/98), E4 Binary 0
 - E4_1: ITU-T, G703 (10/98), E4 Binary 1
 - ENET100FX: IEEE Std 802.3, X fiber over PMD, 100 Mb/s
 - ENET100STP: IEEE Std 802.3 and ANSI X3.263-1995, 100 Base-Tx, STP, 125 Mb/s
 - ENET100UTP: IEEE Std 802.3 and ANSI X3.263-1995, 100 Base-Tx, UTP, 125 Mb/s
 - ENET1000BCX_ATP2: 1000B-CX Abs, TP2, 1.25 Gb/s
 - ENET1000BCX_ATP3: 1000B-CX Abs, TP3, 1.25 Gb/s
 - ENET1000BCX_NTP2: 1000B-CX Norm, TP2, 1.25 Gb/s
 - ENET1250: IEEE Std 802.3 and ANSI X3.263-1995, GB Ethernet, 1.25 Gb/s
 - ENETXAUI_Far: 10 Gigabit Attachment Unit Interface (XAUI), Far, 3.125 Gb/s
 - ENETXAUI_Near: 10 Gigabit Attachment Unit Interface (XAUI), Near, 3,125 Gb/s
 - FC133: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 132.8 Mb/s
 - FC133E: ANSI X3.230-1999 NCITS 1235D/Rev 11, Electrical 132.8 Mb/s
 - FC266: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 265.6 Mb/s
 - FC266E: ANSI X3.230-1999 NCITS 1235D/Rev 11, Electrical, 132.8 Mb/s
 - FC531: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 531.2 Mb/s
 - FC531E: ANSI X3.230-1999 NCITS 1235D/Rev 11, Electrical, 531.2 Mb/s
 - FC1063: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 1.065 Gb/s
 - FC1063E: ANSI X3.230-1999 NCITS 1235D/Rev 11, Electrical, 1.0625 Gb/s
 - FC1063Draft: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, Draft Rev 11
 - FC1063E_ABR: ANSI X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Recv

-
- FC1063E_ABT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Transm
 - FC1063E_ADR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Delta, Recv
 - FC1063E_ADT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Transm
 - FC1063E_AGR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Recv
 - FC1063E_AGT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Transm
 - FC1063E_NBT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Beta, Transm
 - FC1063E_NDT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Delta, Transm
 - FC1063E_NGT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Gamma, Transm
 - FC2125: ANSI X3.230-1999 NCITS 1235D/Rev 11, Optical, 2.125 Gb/s
 - FC2125E_ABR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Recv
 - FC2125E_ABT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Beta, Transm
 - FC2125E_ADR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Delta, Recv
 - FC2125E_ADT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Delta, Transm
 - FC2125E_AGR: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Recv
 - FC2125E_AGT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Abs, Gamma, Transm
 - FC2125E_NBT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Beta, Transm
 - FC2125E_NDT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Delta, Transm
 - FC2125E_NGT: ANS1 X3.230-1999 NCITS 1235D/Rev 11, Norm, Gamma, Transm
 - FC4250E_ABR:
 - FC4250E_ABT:
 - FC4250E_ADR:
 - FC4250E_ADT:
 - FC4250E_AGR:

-
- FC4250E_AGT:
 - FC4250E_NBT:
 - FC4250E_NDT:
 - FC4250E_NGT:
 - FST1: USB, FS:T1, 12 Mb/s
 - FST2: USB, FS:T2, 12 Mb/s
 - FST3: USB, FS:T3, 12 Mb/s
 - FST4: USB, FS:T4: 12 Mb/s
 - FST5: USB, FS:T5, 12 Mb/s
 - FST6: USB, FS:T6, 12 Mb/s
 - FW1394BS400B: IEEE 1394b, S400 Optical, 491.5 Mb/s
 - FW1394BS400BT1: IEEE 1394b, S400b T1, 491.5 Mb/s
 - FW1394BS400BT2: IEEE 1394b, S400b T2, 491.5 Mb/s
 - FW1394BS800B: IEEE 1394b, S800 Optical, 988.0 Mb/s
 - FW1394BS800BT1: IEEE 1394b, S800b T1, 983.0 Mb/s
 - FW1394BS800BT2: IEEE 1394b, S800b T2, 983.0 Mb/s
 - FW1394BS1600B: IEEE 1394b, S1600 Optical, 1.966 Gb/s
 - FW1394BS1600BT1: IEEE 1394b, S1600b T1, 1.966 Gb/s
 - FW1394BS1600BT2: IEEE 1394b, S1600b T2, 1.966 Gb/s
 - G703D1: ITU-T, G703 (10/98), DS1 Rate, 1.544 Mb/s
 - G703DS3: ITU-T, G703 (10/98)
 - HST1: USB, HS:T1, 480 Mb/s
 - HST2: USB, HS:T2, 480 Mb/s
 - HST3: USB, HS:T3, 480 Mb/s
 - HST4: USB, HS:T4, 480 Mb/s
 - HST5: USB, HS:T5, 480 Mb/s
 - HST6: USB, HS:T6, 480 Mb/s
 - INF2_5G: InfiniBand, IBTA Spec 1.0a, 2.5 Optical, 2.5 Gb/s

-
- INF2_5GE: InfiniBand, IBTA Spec 1.0a, 2.5 Electrical, 2.5 Gb/s
 - OC1: GR 253-CORE Issue 3 9/21/2000 OC1/STM0, 51.84 Mb/s
 - OC3: GR 253-CORE Issue 3 9/21/2000 OC1/STM1, 155.52, Mb/s
 - OC12: GR 253-CORE Issue 3 9/21/2000 OC1/STM4, 622.08 Mb/s
 - OC48: GR 253-CORE Issue 3 9/21/2000 OC1/STM16, 2.4883 Gb/s
 - OC48_FEC: Forward Error Correction - CSA8000 mask, 2.666 Gb/s
 - PCIEXPRESS_Rcv: PCI Express Receiver, 2.5 Gb/s
 - PCIEXPRESS_Xmit: PCI Express Transmitter, 2.5 Gb/s
 - RATE32Mbit: ITU-T, G703 (10/98), 32.064 Mb/s
 - RATE97Mbit: ITU-T, G703 (10/98), 97 Mbit, 97.728 Mb/s
 - RIO_DRV1G: Rapid IO Driver, 1 Gb/s
 - RIO_DRV1_5G: Rapid IO Driver, 5 Gb/s
 - RIO_DRV2G: Rapid IO Driver, 2 Gb/s
 - RIO_DRV500M: Rapid IO Driver, 500 Mb/s
 - RIO_DRV750M: Rapid IO Driver, 750 Mb/s
 - RIO_EDRV1G: Rapid IO Extended Driver, 1 Gb/s
 - RIO_EDRV1_5G: Rapid IO Extended Driver, 1.5 Gb/s
 - RIO_EDRV2G: Rapid IO Extended Driver, 2 Gb/s
 - RIO_EDRV500M: Rapid IO Extended Driver, 500 Mb/s
 - RIO_EDRV750M: Rapid IO Extended Driver, 750 Mb/s
 - RIO_RCV1G: Rapid IO Receiver, 1 Gb/s
 - RIO_RCV1_5G: Rapid IO Receiver, 1.5 Gb/s
 - RIO_RCV2G: Rapid IO Receiver, 2 Gb/s
 - RIO_RCV500M: Rapid IO Receiver, 500 Mb/s
 - RIO_RCV750M: Rapid IO Receiver, 750 Mb/s

-
- RIO_SERIAL_1G: Rapid IO Serial, 1.25 Gb/s
 - RIO_SERIAL_2G: Rapid IO Serial, 2.5 Gb/s
 - RIO_SERIAL_3G: Rapid IO Serial, 3.25 Gb/s
 - SFI5_RCVBCLK2: SFI15 Receive - Test Point B Clock Signal 2, 2.488 Gb/s
 - SFI5_RCVBCLK3: SFI15 Receive - Test Point B Clock Signal 3, 3.125 Gb/s
 - SFI5_RCVBDATA2: SFI15 Receive - Test Point B Data Signal 2, 2.488 Gb/s
 - SFI5_RCVBDATA3: SFI15 Receive - Test Point B Data Signal 3, 3.125 Gb/s
 - SFI5_RCVDCLK2: SFI15 Receive - Test Point D Clock Signal 2, 2.488 Gb/s
 - SFI5_RCVDCLK3: SFI15 Receive - Test Point D Clock Signal 3, 3.125 Gb/s
 - SFI5_RCVDDATA2: SFI15 Receive - Test Point D Data Signal 2, 2.488 Gb/s
 - SFI5_RCVDDATA3: SFI15 Receive - Test Point D Data Signal 3, 3.125 Gb/s
 - SFI5_XMITACLK2: SFI15 Transmit - Test Point A Clock Signal 2, 2.488 Gb/s
 - SFI5_XMITACLK3: SFI15 Transmit - Test Point A Clock Signal 3, 3.125 Gb/s
 - SFI5_XMITADATA2: SFI15 Transmit - Test Point A Data Signal 2, 2.488 Gb/s
 - SFI5_XMITADATA3: SFI15 Transmit - Test Point A Data Signal 3, 3.125 Gb/s
 - SFI5_XMITCCLK2: SFI15 Transmit - Test Point C Clock Signal 2, 2.488 Gb/s
 - SFI5_XMITCCLK3: SFI15 Transmit - Test Point C Clock Signal 3, 3.125 Gb/s
 - SFI5_XMITCDATA2: SFI15 Transmit - Test Point C Data Signal 2, 2.488 Gb/s
 - SFI5_XMITCDATA3: SFI15 Transmit - Test Point C Data Signal 3, 3.125 Gb/s
 - STM1E_0: ITU-T, G703 (10/98), STM1E Binary 0
 - STM1E_1: ITU-T, G703 (10/98), STM1E Binary 1
 - STS1Pulse: ANSI T1.102-1993 (R1999), STS-1 Pulse, 51.84 Mb/s

-
- STS1Eye: ANSI T1.102-1993 (R1999), STS-1 Eye, 51.84 Mb/s
 - STS3: ANSI T1.102-1993 (R1999), STS-3, 155.52 Mb/s
 - STS3_Max: ANSI T1.102-1993 (R1999), STS-3 Max Output, 155.52 Mb/s
 - TFI15_2: TFI-5, 2.488 Gb/s
 - TFI5_3: TFI-5, 3.1104 Gb/s
 - USERMask
 - VSROC192: VSR OC192/STM64, 1.24416 Gb/s

Example 1

`MASK:STANDARD E1COAX`

Selects the E1 Coax standard mask.

Example 2

`MASK:STANDARD?`

Might return `:MASK:STANDARD E1COAX`

Indicating that the E1 Coax mask is selected as the current mask standard.

MASK:STOPOnviolation

Description

This command sets or returns the stop on violation setting. When enabled, this command stops waveform acquisitions on the first occurrence of a waveform violation. MASK:DISPLAY must be ON for the oscilloscope to detect violations. In other words, it controls whether the pass/fail test stops running when a violation is detected.

Group

Mask

Syntax 1

```
MASK:STOPOnviolation {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:STOPOnviolation?
```

Arguments

- ON

Stops waveform acquisition on the first occurrence of a mask violation.

- OFF

Turns off mask test stop on violation.

- <NR1>

A 0 is an integer, which turns off stop on violation, and any other value turns on stop on violation.

Example

```
MASK:STOPONVIOLATION ON
```

Sets the oscilloscope to stop waveform acquisition on the first occurrence of a waveform violation.

MASK:TEST:AUX:COMPLetion

Description

This command sets or returns the test:aux on pass/fail test completion mode. When enabled, this command causes the oscilloscope to send a TTL signal out the Aux:Out port when the mask pass/fail status changes to 'Failing'. In other words, this command controls whether or not a TTL signal is sent out the AuxOut port on completion of the pass/fail test.

Group

Mask

Syntax 1

```
MASK:TEST:AUX:COMPLetion {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:TEST:AUX:COMPLetion?
```

Arguments

- ON

Turns on the pass/fail mask:test:aux on completion.

- OFF

Turns off the pass/fail mask:test:aux on completion.

- <NR1>

A 0 disables the mask:test:aux:completion function; any other value enables it.

Example

```
MASK:TEST:AUX:COMPLETION ON
```

Turns on the pass/fail test on completion

MASK:TEST:AUX:FAILure

Description

This command sets or returns the test:aux status on pass/fail test failure mode. When enabled, this command causes the oscilloscope to send a TTL signal out the Aux-Out port when the pass/fail status changes to 'Failing'. In other words, it sends the signal when a failure occurs during the mask pass/fail test.

Group

Mask

Syntax

```
MASK:TEST:AUX:FAILure {OFF|ON|<NR1>}
```

Syntax

```
MASK:TEST:AUX:FAILure?
```

Arguments

- ON

Turns on the pass/fail mask:test:aux on failure.

- OFF

Turns off the pass/fail mask:test:aux on failure. This is the default.

- <NR1>

A 0 disables the mask:test:aux:failure function; any other value enables it.

Example

```
MASK:TEST:AUX:FAILURE ON
```

Turns on the pass/fail test on failure

MASK:TEST:BEEP:COMPLetion

Description

This command sets or returns the beep on pass/fail test completion mode. When enabled, this command causes the oscilloscope to emit a tone when the mask pass/fail test completes.

Group

Mask

Syntax 1

```
MASK:TEST:BEEP:COMPLetion {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:TEST:BEEP:COMPLetion?
```

Arguments

- ON

Turns on the pass/fail beep on completion.

- OFF

Turns off the pass/fail beep on completion.

- <NR1>

A 0 turns off the pass/fail beep on completion, any other integer turns on the pass/fail beep on completion.

Example

```
MASK:TEST:BEEP:COMPLETION ON
```

Turns on pas/fail beep on completion

MASK:TEST:BEEP:FAILure

Description

This command sets or returns the beep status on pass/fail test failure mode. When enabled, this command causes the oscilloscope to emit a tone when the pass/fail status changes to 'Failed'. In other words, it sounds the bell when a failure occurs during the mask pass/fail test.

Group

Mask

Syntax 1

```
MASK:TEST:BEEP:FAILure {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:TEST:BEEP:FAILure?
```

Arguments

- ON

Turns on the pass/fail beep on failure.

- OFF

Turns off the pass/fail beep on failure. This is the default.

- <NR1>

A 0 turns off the pass/fail beep on failure, and any other integer turns on the pass/fail beep on failure.

Example

```
MASK:TEST:BEEP:FAILURE OFF
```

Turns off the pass/fail beep on failure

MASK:TEST:DELAY

Description

This command sets or returns the amount of time, in seconds, to wait after starting pass/fail testing to evaluate waveforms. This is useful if the test system requires 'settling' time prior to evaluating waveforms.

Group

Mask

Syntax

```
MASK:TEST:DELAY <NR3>
```

Syntax

```
MASK:TEST:DELAY?
```

Argument

- <NR3>

Is a floating point number that specifies the amount of time, in seconds, to delay the start of the pass/fail test.

Example

```
MASK:TEST:DELAY 1
```

Delays waveform evaluation by 1 second after the start of pass/fail testing.

MASK:TEST:HARDCopy

Description

This command sets or returns the hard copy in pass/fail failure mode. When enabled, this command causes the oscilloscope to generate a screen hard copy, using the current oscilloscope hard copy settings, as soon as the pass/fail status changes to 'Failing'. In other words, it prints the screen to the default printer.

Group

Mask

Syntax 1

```
MASK:TEST:HARDCOPY {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:TEST:HARDCOPY?
```

Arguments

- ON

Turns on generate hard copy on failure.

- OFF

Turns off generate hard copy on failure.

- <NR1>

A 0 turns off generate hard copy on failure, and any other value turns on generate hard copy on failure.

Example

```
MASK:TEST:HARDCOPY ON
```

Turns on generate hard copy on failure.

MASK:TEST:LOG:FAILure

Description

This command sets or returns the log status on pass/fail test failure mode. When enabled, this command causes the oscilloscope to log the current date and time to a file when the pass/fail status changes to 'Failing'. The file name is determined from the current date and time. The user cannot change this name.

Group

Mask

Related Commands

MASK:TEST:SAVEWFM:FILENAME

Syntax 1

MASK:TEST:LOG:FAILure {OFF|ON|<NR1>}

Syntax 2

MASK:TEST:LOG:FAILure?

Arguments

- ON

Turns on the pass/fail log on failure.

- OFF

Turns off the pass/fail log on failure. This is the default.

- <NR1>

A 0 turns off the pass/fail log on failure, and any other integer turns on the pass/fail log on failure.

Example

MASK:TEST:LOG:FAILURE OFF

Turns off the pass/fail log on failure.

MASK:TEST:REPeat

Description

This command sets or returns the repeat pass/fail test on completion mode. When enabled, this command repeats the pass/fail test cycle at the completion of the previous test cycle.

Group

Mask

Related Commands

MASK:COUNT:VIOLATIONS?

Syntax 1

MASK:TEST:REPeat {OFF|ON|<NR1>}

Syntax 2

MASK:TEST:REPeat?

Arguments

- ON

Turns on repeat on completion.

- OFF

Turns off repeat on completion

- <NR1>

A 0 turns off repeat on completion, and any other integer turns on repeat on completion.

Example

MASK:TEST:REPEAT ON

Causes the oscilloscope to continuously repeat the pass/fail test.

MASK:TEST:SAmpLe

Description

This command sets or returns the minimum number of waveform database (WfmDB) points the oscilloscope can acquire before it stops a single sequence acquisition or stops running a mask test.

Hint: a bigger sample size may allow a greater throughput (more waveforms to get acquired at a time). However, a very large sample size may slow down the display update. So if you find yourself wanting to speed up the display rate, consider reducing the sample size.

This command works the same as the ACQUIRE:NUMSAMPLES command.

Group

Mask

Syntax 1

```
MASK:TEST:SAmpLe {<NR1>}
```

Syntax 2

```
MASK:TEST:SAmpLe?
```

Arguments

- <NR1>

The number of points to sample.

Example

```
MASK:TEST:SAmpLe 100000
```

Sets the oscilloscope to acquire 100,000 WfmDB points.

MASK:TESt:SAMple:THReshold

Description

This command sets or returns the minimum number of hits in mask regions needed to cause the pass/fail status to change from PASSING to FAILING. This affects the mask test when the oscilloscope is acquiring in waveform data-base (WfmDB) mode.

Group

Mask

Syntax 1

```
MASK:TESt:SAMple:THReshold {<NR1>}
```

Syntax 2

```
MASK:TESt:SAMple:THReshold?
```

Arguments

- <NR1>

The number of hits that can happen when the test fails.

Example

```
MASK:TESt:SAMPLE:THRESHOLD 100
```

The test will fail when 100 or more hits happen.

MASK:TEST:SAVEWFM

Description

This command sets or returns the save waveform to file on failure setting. When enabled, this command copies waveform data from active channels to files when the pass/fail test status changes to 'Failing'.

Note: This oscilloscope saves files only on the first violation that causes the pass/fail test to fail. To generate a waveform file for every violation occurrence, set MASK:TEST:REPEAT to ON and MASK:TEST:WAVEFORM to 1. Use this technique only when the expected number of violations is small and the violation occurs intermittently.

Group

Mask

Syntax 1

```
MASK:TEST:SAVEWFM {OFF | ON | <NR1>}
```

Syntax 2

```
MASK:TEST:SAVEWFM?
```

Arguments

- ON

Turns on copying all active channel waveforms to files.

- OFF

Turns off copying all active channel waveforms to files.

- <NR1>

A 0 turns off copying all active channel waveforms to files, and any other integer turns on copying all active channel waveforms to files.

Example

```
MASK:TEST:SAVEWFM ON
```

Sets the oscilloscope to save the failing waveform data to files.

MASK:TEST:SAVEWFM:FILENAME

Description

This command sets or returns the name of the directory and file to use with the MASK:TEST:SAVEWFM command. It defines the directory the files will be put in (name comes from date and time).

Group

Mask

Related Commands

MASK:TEST:SAVEWFM, MASK:TEST:LOG:FAILURE

Syntax 1

MASK:TEST:SAVEWFM:FILENAME <Qstring>

Syntax 2

MASK:TEST:SAVEWFM:FILENAME?

Argument

- <Qstring>

Is a string representing the name of the file to save waveform data to

Example

```
MASK:TEST:SAVEWFM:FILENAME "TEST"
```

Sets the file "TEST" as the file to save waveform data to on failure.

MASK:TEST:SRQ:COMPLetion

Description

This command sets or returns an SRQ on pass/fail test completion mode. When enabled, this command causes the oscilloscope to send an GPIB SRQ command when the mask pass/fail test completes.

Group

Mask

Syntax 1

```
MASK:TEST:SRQ:COMPLetion {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:TEST:SRQ:COMPLetion?
```

Arguments

- ON

Turns on the pass/fail SRQ on completion.

- OFF

Turns off the pass/fail SRQ on completion.

- <NR1>

A 0 turns off the pass/fail SR on completion, any other integer turns on the pass/fail SRQ on completion.

Example

```
MASK:TEST:SRQ:COMPLETION ON
```

Turns on pass/fail SRQ on completion

MASK:TEST:SRQ:FAILure

Description

This command sets or returns the SRQ status on pass/fail test failure mode. When enabled, this command causes the oscilloscope to send a GPIB SRQ command when the pass/fail status changes to 'Failing'. In other words, it sends an SRQ when a failure occurs during the mask pass/fail test.

Group

Mask

Syntax 1

```
MASK:TEST:SRQ:FAILure {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:TEST:SRQ:FAILure?
```

Arguments

- ON

Turns on the pass/fail SRQ on failure.

- OFF

Turns off the pass/fail SRQ on failure. This is the default.

- <NR1>

A 0 turns off the pass/fail SRQ on failure, and any other integer turns on the pass/fail SRQ on failure.

Example

```
MASK:TEST:SRQ:FAILURE OFF
```

Turns off the pass/fail SRQ on failure

MASK:TEST:STATE

Description

This command sets or returns the state of mask pass/fail testing. When enabled, this command resets the pass/fail status information and starts pass/fail mask testing. This command also generates an 'Operation Complete' message when pass/fail testing completes. Refer to BUSY? And *OPC commands for information on synchronizing a controller with pass/fail testing.

Group

Mask

Syntax 1

```
MASK:TEST:STATE {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:TEST:STATE?
```

Arguments

- ON

Turns on mask pass/fail testing.

- OFF

Turns off mask pass/fail testing.

- <NR1>

A 0 turns off mask pass/fail testing; any other integer turns on mask pass/fail testing.

Example

```
MASK:TEST:STATE ON
```

Turns on mask pass/fail testing.

MASK:TEST:STATUS?

Description

This command returns the pass/fail test status. This command returns one of: OFF, DELAY, PASSING, FAILING, PASSED, FAILED, and VIOLATION. In other words, it indicates the result of the pass/fail test. When the violation count exceeds the violation threshold, the status changes from Passing to Failed.

Group

Mask

Syntax

```
MASK:TEST:STATUS?
```

Example

```
MASK:TEST:STATUS?
```

Might return: MASK:TEST:STATUS PASSING

MASK:TEST:STOP:FAILure

Description

This command sets or returns the stop status on pass/fail test failure mode. When enabled, this command causes the oscilloscope to stop acquiring data when the pass/fail status changes to 'Failing'. In other words, it stops when a failure occurs during the mask pass/fail test. Repeat-on-completion mode has no effect.

Group

Mask

Syntax 1

```
MASK:TEST:STOP:FAILure {OFF|ON|<NR1>}
```

Syntax 2

```
MASK:TEST:STOP:FAILure?
```

Arguments

- ON

Turns on the pass/fail stop on failure.

- OFF

Turns off the pass/fail stop on failure. This is the default.

- <NR1>

A 0 turns off the pass/fail stop on failure, and any other integer turns on the pass/fail SRQ on failure.

Example

```
MASK:TEST:STOP:FAILURE OFF
```

Turns off the pass/fail stop on failure

MASK:TEST:THReshold

Description

This command sets or returns the number of failed tested waveforms needed in each pass/fail test to cause the pass/fail status to change from PASSING to FAILING.

Group

Mask

Syntax 1

```
MASK:TEST:THReshold <NR1>
```

Syntax 2

```
MASK:TEST:THReshold?
```

Argument

- <NR1>

Is an integer that represents the number of tested waveform violations occurring in each pass/fail test needed to change the pass/fail status to FAILING.

This means that if the threshold is set to the default of 1 then 1 failure would cause a test failure. If the threshold is set to 2 then 2 failures would cause a test failure.

Example

```
MASK:TEST:THRESHOLD?
```

Might return: MASK:TEST:THRESHOLD 32

MASK:TEST:WAVEform

Description

This command sets or returns the number of waveforms to test during pass/fail testing. If Averaging is on, this is the number of averaged waveforms to test during pass/fail testing. In other words, it specifies the number of waveforms to be acquired in counting the mask hits in the pass/fail test. The default is 20.

Group

Mask

Syntax 1

```
MASK:TEST:WAVEform <NR1>
```

Syntax 2

```
MASK:TEST:WAVEform?
```

Argument

- <NR1>

Is an integer that sets the number of tested (sampled or averaged) waveforms to test for each pass/fail test.

Example

```
MASK:TEST:WAVEFORM?
```

Might return: 500

MASK:USER:AMPLitude

Description

This command sets or returns the current mask's nominal pulse amplitude in volts. The query form of this command returns the nominal pulse amplitude of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:VSCALE

Syntax 1

MASK:USER:AMPLitude <NR3>

Syntax 2

MASK:USER:AMPLitude?

Argument

- <NR3>

This is a floating number that sets the nominal pulse amplitude in volts

Example

```
MASK:USER:AMPLITUDE 2.4
```

Sets the User mask nominal pulse amplitude to 2.4 volts.

MASK:USER:BITRate

Description

This command sets or returns the bit rate for the user mask.

Group

Mask

Syntax 1

```
MASK:USER:BITRate <NR1>
```

Syntax 2

```
MASK:USER:BITRate?
```

Argument

- <NR1>

This is a number that sets the bit rate of the user mask in bits per second.

Example 1

```
MASK:USER:BITRATE 156e6
```

Sets the bit rate.

Example 2

```
MASK:USER:BITRATE?
```

Might return:

```
1560000
```

MASK:USER:HSCALE

Description

This command sets or returns the nominal timing resolution used to draw the mask, in time/division. The query form of this command returns the nominal timing resolution of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:HTRIGPOS

Syntax 1

MASK:USER:HSCALE <NR3>

Syntax 2

MASK:USER:HSCALE?

Argument

- <NR3>

This is a floating point number that sets the mask drawing timing resolution.

Example

MASK:USER:HSCALE 40e-9

Sets the nominal timing resolution used to draw the mask pulse shape to 40 nanoseconds per division.

MASK:USER:HTRIGPOS

Description

This command sets or returns the nominal trigger position.(pulse leading edge) used to draw the mask as a fraction of the display width. The query form of this command returns the nominal trigger position of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:HSCALE?

Syntax 1

MASK:USER:HTRIGPOS <NR3>

Syntax 2

MASK:USER:HTRIGPOS?

Argument

- <NR3>

Is a floating point number in the range of 0.0 to 1.0 that sets the trigger points as a fraction of the display width, referenced from the left edge of the graticule.

0.0 represents the left edge.

Example

MASK:USER:HTRIGPOS 0.6

Sets the User mask nominal pulse leading edge position to the 6th horizontal graticule division.

MASK:USER:PATTERNBITS

Description

This command sets or returns the number of bits used for serial trigger for the User mask standard. For example, DS1 requires six bits, four leading zeros, a one, and a trailing zero. The query form of this command returns the serial bit value of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:PRESAMPBITS

Syntax 1

MASK:USER:PATTERNBITS <NR1>

Syntax 2

MASK:USER:PATTERNBITS?

Argument

- <NR1>

Is an integer that sets the number of bits.

Example

MASK:USER:PATTERNBITS 6

Sets the User mask number of serial pattern bits to 6.

MASK:USER:PRESAMPBITS

Description

This command sets or returns the number of bits before the (isolated one) pulse leading edge in the serial trigger pass/fail testing. For example, DS1 has four leading zeros. The query form of this command returns the presample bit value of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:PATTERNBITS

Syntax 1

MASK:USER:PRESAMPBITS <NR1>

Syntax 2

MASK:USER:PRESAMPBITS?

Argument

- <NR1>

Is an integer that sets the number of bits before the trigger pulse.

Example

MASK:USER:PRESAMPBITS 4

Sets the User mask number of bits before the trigger pulse to 4.

MASK:USER:RECOrdlength

Description

This command sets or returns the nominal record length for pulse mask testing. The query form of this command returns the record length value of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Syntax 1

```
MASK:USER:RECOrdlength <NR1>
```

Syntax 2

```
MASK:USER:RECOrdlength?
```

Argument

- <NR1>

Is an integer number that sets the record length.

Example

```
MASK:USER:RECORDLENGTH 500
```

Sets the User mask record length to 500 points.

MASK:USER:SEG<m> DELEte

Description

This command deletes the specified mask segment from the User mask, whether or not the User mask is the current mask.

Group

Mask

Syntax

```
MASK:USER:SEG<m> DELEte
```

Arguments

- <m>

Is an integer that specifies the User mask segment number to delete from the User mask.

Example

```
MASK:USER:SEG3 DELETE
```

Deletes mask segment three from the User mask.

MASK:USER:SEG<m>:NR_PT?

Description

This command returns the number of points that make up the specified User mask segment. Each mask point consists of an X-Y pair of coordinates.

Group

Mask

Syntax

```
MASK:USER:SEG<m>:NR_PT?
```

Arguments

- <m>

Is an integer number that specifies a User mask segment number.

Returns

<NR1> is 0 to 50.

Example

```
MASK:USER:SEG<m>:NR_PT?
```

Might return MASK:USER:SEG2:NR_PT 5

MASK:USER:SEG<m>:POINTS

Description

This command sets or returns the X-Y user coordinates of all points in the specified User mask segment. The set form defines new points in the User mask, replacing any existing points in the specified User mask segment. In other words, it sets or returns the vertices for a particular segment in the selected mask.

Group

Mask

Syntax 1

```
MASK:USER:SEG<m>:POINTS <NR3>, <NR3> [ ,  
<NR3>, <NR3> ]
```

Syntax 2

```
MASK:USER:SEG<m>:POINTS?
```

Arguments

- <m>

Is an integer that specifies the User mask segment number.

- <NR3>

Refers to the coordinates of one of the vertices in the User mask. Each pair of numbers is the horizontal and vertical coordinates of a mask segment vertex. The pairs must be listed in a counterclockwise order. If the vertical or horizontal scale or position is changed after this command and then the query form of this command is issued, then the value returned from the oscilloscope will not be the same. If just one pair is input then they are ignored and the User mask segment is marked as undefined. The default is not points in the User mask segment.

Example

```
MASK:USER:SEG2:POINTS -2.3e-9, 44e-3,  
-2.5e-9, 47e-3, 1.2e-9, 44e-3
```

Defines the points in User mask segment 2.

MASK:USER:SERIALTRIG

Description

This command sets or returns the type of triggering used in pass/fail testing of the User mask.

Group

Mask

Related Commands

MASK:USER:PATTERNBITS, MASK:USER:PRESAMP-BITS

Syntax 1

MASK:USER:SERIALTRIG {AMI | HDB3 | B3ZS | B6ZS | B8ZS | CMI | NRZ | MLT3 | EDGE}

Syntax 2

MASK:USER:SERIALTRIG?

Argument

- AMI
Alternate Mark Inversion
- HDB3
High-Density Bipolar Three-Bit substitution
- B3ZS
Bipolar 3 Zero Substitution
- B6ZS
Bipolar 6 Zero Substitution
- B8ZS
Bipolar 8 Zero Substitution
- CMI
Coded Mark Inversion
- NRZ
Non-Return to Zero
- MLT3
Multi-Level Transition
- EDGE

Example

`MASK:USER:SERIALTRIG AMI`

Sets AMI as the type of triggering used in testing the User mask.

MASK:USER:TRIGTOSAMP

Description

This command sets or returns the time, in seconds, from the (leading edge) trigger position to the pulse bit sampling position. The query form of this command returns the time value of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:WIDTH

Syntax 1

MASK:USER:TRIGTOSAMP <NR3>

Syntax 2

MASK:USER:TRIGTOSAMP?

Argument

- <NR3>

Is the floating point number that sets the time to the pulse bit sampling position.

Example

MASK:USER:TRIGTOSAMP 162e-9

Sets the User mask time from the (leading edge) trigger position to the pulse bit sampling position to 162 nanoseconds.

MASK:USER:VOFFSet

Description

This command sets or returns the nominal vertical offset, in volts, used to vertically offset the input channels. The query form of this command returns the offset value of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:VPOS

MASK:USER:VScale

Syntax 1

MASK:USER:VOFFSet <NR3>

Syntax 2

MASK:USER:VOFFSet?

Argument

- <NR3>

Is a floating point number that sets the nominal vertical offset in volts.

Example

MASK:USER:VOFFSET 0.0

Sets the User mask nominal vertical offset to 0.0 volts.

MASK:USER:VPOS

Description

This command sets or returns the nominal vertical position, in divisions, used to vertically position the input channels. The query form of this command returns the vertical position value of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:VOFFSet

MASK:USER:VSCALE

Syntax 1

MASK:USER:VPOS <NR3>

Syntax 2

MASK:USER:VPOS?

Argument

- <NR3>

Is a floating point number that sets the nominal vertical position in divisions.

Example

MASK:USER:VPOS -0.8

Sets the User mask nominal vertical position to -0.8 divisions.

MASK:USER:VSCALE

Description

This command sets or returns the nominal vertical scale, in volts per division, used to vertically scale the input channels. The query form of this command returns the vertical scale value of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:VOFFSet

MASK:USER:VPOS

Syntax 1

MASK:USER:VSCALE <NR3>

Syntax 2

MASK:USER:VSCALE?

Argument

- <NR3>

Is a floating point number that sets the nominal bit width in seconds.

Example

MASK:USER:VSCALE 200e-3

Sets the User mask nominal vertical scale to 200 millivolts per division

MASK:USER:WIDTH

Description

This command sets or returns the nominal bit width in seconds. The query form of this command returns the bit width value of the displayed mask. The set form of this command only affects the User mask, regardless of the current (displayed) mask.

Group

Mask

Related Commands

MASK:USER:TRIGTOSAMP

Syntax 1

MASK:USER:WIDTH <NR3>

Syntax 2

MASK:USER:WIDTH?

Argument

- <NR3>

Is a floating point number that sets the nominal bit width in seconds.

This is the time of one bit of data where
 $\text{bit width} = 1 / (\text{data rate of the signal})$.

Example

MASK:USER:WIDTH 648e-9

Sets the User mask nominal bit width to 648 nanoseconds (1.544 Mbits/second).

MATH<x>

Description

This query-only command returns the definition for the math waveform specified by x, which ranges from 1 through 4.

Group

Math

Related Commands

SElect:<wfm>

Syntax

MATH<x>?

Example

MATH1?

This query might return

```
:MATH1 DEFINE "PROBE POINT7"  
;NUMAVG 2;SCALE 1.0000E+00;POSITION  
0.0000E+00  
;LABEL:NAME "Math1";XPOS 5;YPOS  
65;;MATH1:SPECTRAL:MAG DB  
;PHASE DEGREES;GATEPOS 0.0000E+00;GATE-  
WIDTH 1.9996E-06  
;REFLEVEL 2.0000E+01;REFLEVELOFFSET  
2.2360E-01  
;SPAN 1.2500+09;CENTER 6.2500E+08;RESBW  
1.0002E+06  
;WINDOW GAUSSIAN;SUPPRESS -3.5000EE+01;UN-  
WRAP 0;LOCK 0
```

MATH<x>:DEFine

Description

This command allows you to define new waveforms using mathematical expressions. Sending this command is equivalent to selecting Math Setup from the Math menu, selecting a math waveform (Math 1 through Math 4) and then entering a math expression in the Math<x> box. The query form of this command returns the math definition for the specified math waveform.

You can specify a math expression from waveforms, measurements and scalar sources, functions, operands, and numerical constants. You can define and display up to four math waveforms simultaneously.

Math expressions can be simple, such as Ch1, which specifies that a waveform should show the signal source of channel 1 with no mathematical computation. Math expressions can also be complex, consisting of 100 plus characters and comprising many sources, functions, and operands. As an example, you can enter the expression $\text{Log}(\text{Ch1}+\text{Ch2})$, which specifies that the signals from channels 1 and 2 are to be algebraically added, and the base 10 log of the sum is to be shown as the final math waveform.

For information about constructing mathematical expressions, see *Creating and Using Math Waveforms* in the User's Guide for this oscilloscope.

Group

Math

Syntax 1

```
MATH<x>:DEFine <QString>
```

Syntax 2

```
MATH<x>:DEFine?
```

Argument

- <QString>

This quoted string argument is the mathematical expression that defines the waveform.

Example 1

```
MATH2:DEFINE "Ch1+Ch2"
```

This command adds the Channel 1 waveform and Channel 2 waveforms, storing the results in Math 2.

Example 2

```
MATH2:DEFINE?
```

This query might return

```
:MATH1:DEFINE "CH2*REF2"
```

As the expression that defines Math 1.

MATH<x>:LABEL:NAME

Description

This command sets or returns the label string, which is used for annotating the math waveform on the screen. This command is equivalent to selecting Math Setup from the Math menu and then entering a label in the Label box.

Group

Math

Syntax 1

```
MATH<x>:LABEL:NAME <string>
```

Syntax 2

```
MATH<x>:LABEL:NAME?
```

Argument

- <string>

This specifies the label to annotate the math waveform.

Example 1

```
MATH2:LABEL:NAME "Probe point7"
```

This command assigns Probe point7 as the label that annotates Math2.

Example 2

```
MATH2:LABEL:NAME?
```

This query might return

```
:MATH2:LABEL:NAME "Probe point7"
```

as the label that annotates Math 2.

MATH<x>:LABEL:XPOS

Description

This command sets or queries the X screen offset at which the label attached to the displayed math waveform is displayed, relative to the left edge of the screen. Channels are specified by x, which ranges from 1 through 4. This command is equivalent to selecting Math Label from the Math menu and entering a value in the X Position box.

Group

Math

Related Commands

MATH<x>:LABEL:YPOS

Syntax 1

MATH<x>:LABEL:XPOS <NR3>

Syntax 2

MATH<x>:LABEL:XPOS?

Argument

- <NR3>

This is the location (in divisions) where the waveform label for the selected math is displayed, relative to the left edge of the screen.

Example 1

MATH2:LABEL:XPOS 2.5

This command moves the waveform label for the MATH3 waveform, so that it begins 2.5 divisions to the right of the left edge of the screen.

Example 2

MATH2:LABEL:XPOS?

This query might return

:MATH2:LABEL:XPOS 0.0000

Indicating that the waveform label for the MATH2 waveform is currently on the far left edge of the screen.

MATH<x>:LABEL:YPOS

Description

This command sets or queries the Y screen offset at which the label attached to the displayed math waveform is displayed, relative to the vertical position of the associated waveform. Channels are specified by x, which ranges from 1 through 4. This command is equivalent to selecting Math Label from the Math menu and entering a value in the Y Position box.

Group

Math

Related Commands

MATH<x>:LABEL:XPOS

Syntax 1

MATH<x>:LABEL:YPOS <NR3>

Syntax 2

MATH<x>:LABEL:YPOS?

Argument

- <NR3>

This is the location (in divisions) where the waveform label for the selected math is displayed, relative to the vertical position of the associated waveform.

Example 1

MATH2:LABEL:YPOS -2.5

This command moves the waveform label for the MATH2 waveform to beneath (2.5 divisions below) the associated waveform.

Example 2

MATH2:LABEL:YPOS?

This query might return

:MATH2:LABEL:YPOS 0.000

Indicating that the waveform label for the MATH2 waveform is currently located on the same Y position as the associated waveform.

MATH<x>:NUMAVg

Description

This command sets or returns the acquisition number at which the averaging algorithm will begin *exponential averaging*. Prior to that acquisition number, the algorithm uses *stable averaging*. This has no affect unless the AVG() function is used in the specified math. If so, it affects all AVG() functions in this math. This command is equivalent to selecting Set Math Averages from the Math menu and then entering an averaging value for the math waveform.

Group

Math

Related Commands

ACQuire:NUMAVg

Syntax 1

MATH<x>:NUMAVg <NR1>

Syntax 2

MATH<x>:NUMAVg?

Argument

- <NR1>

This specifies the number of acquisitions over which exponential averaging is performed.

Example 1

```
MATH2:NUMAVg 10
```

This command successively averages Math 2 by 10 times.

Example 2

```
MATH2:NUMAVg?
```

This query might return

```
:MATH2:NUMAVG 10
```

Indicating 10 Math 2 waveforms are successively averaged before a single acquisition occurs.

MATH<x>:POsition

Description

This command sets or queries the vertical position of the specified Math waveform. The Math waveform is specified by *x*, which can range from 1 through 4. The position value is usually applied to the signal before it is digitized. The highest three units/div scale ranges of a given math are implemented by changing the way the acquired data is displayed. When the instrument is operating in any of these highest three scale ranges, the position control operates only on the signal after it is digitized. Note that if a signal that exceeds the range of the digitizer in one of these three scale ranges is repositioned, the displayed waveform will contain clipped values on-screen. This command is equivalent to selecting Position/Scale from the Math menu and then entering a Vert Pos value or adjusting the front-panel Vertical **POSITION** knob.

Increasing the position value of a waveform causes the waveform to move up, and decreasing the position value causes the waveform to move down. Position adjusts only the display position of a waveform, whether channel, math, or reference waveform. The position value determines the vertical graticule coordinate at which input signal values, equal the present offset setting for that reference, are displayed. For example, if the position for Math 3 is set to 2.0 and the offset is set to 3.0, then the input signals equal to 3.0 are displayed 2.0 divisions above the center of the screen.

Autoscaling occurs when a math waveform is first defined and enabled. It also occurs when a math string changes. After the math waveform is computed for the first time, the instrument determines the min + max of that waveform data. Then, the math position is set so that $(\text{min} + \text{max})/2$ is in the center of the screen. In addition, the instrument sets the math scale so that the range of the min and max covers 6 divisions. This autoscaling process can take up to $\frac{1}{2}$ second to complete and will override any vertical scale or position commands for that math waveform received during this time. You should insert an appropriate pause in your program after defining and enabling a math waveform before changing its position and scale.

Group

Math, Vertical

Related Commands

CH<x>POSition

Syntax 1

MATH<x>:POSition <NR3>

Syntax 2

MATH<x>:POSition?

Argument

- <NR3>

This is the desired position in divisions from the center graticule.

Example 1

```
MATH2:POSition 1.3E+00
```

This command positions the Math 2 input signal 1.3 divisions higher than a position of 0.

Example 2

```
MATH1:POSition?
```

This query might return

```
:MATH1:POSITION -1.3000E+00
```

Indicating that the current position of Math 1 is 1.3 divisions below the center graticule.

MATH<x>:SCALE

Description

This command sets or queries the vertical scale of the specified math waveform. The math waveform is specified by *x*, which can range from 1 through 4. This command is equivalent to selecting Position/Scale from the Math menu and then entering a Vert Scale value or adjusting the front-panel Vertical **SCALE** knob.

Each waveform has its own vertical scale parameter. For a signal with constant amplitude, increasing the Scale causes the waveform to be displayed smaller. Decreasing the scale causes the waveform to be displayed larger.

Scale affects all waveforms. For reference and math waveforms, the scale setting controls the display only, graphically scaling these waveforms and having no effect on the acquisition hardware.

Autoscaling occurs when a math waveform is first defined and enabled. It also occurs when a math string changes. After the math waveform is computed for the first time, the instrument determines the min + max of that waveform data. Then, the math position is set so that $(\text{min} + \text{max})/2$ is in the center of the screen. In addition, the instrument sets the math scale so that the range of the min and max covers 6 divisions. This autoscaling process can take up to $\frac{1}{2}$ second to complete and will override any vertical scale or position commands for that math waveform received during this time. You should insert an appropriate pause in your program after defining and enabling a math waveform before changing its position and scale.

Group

Math, Vertical

Related Commands

CH<x>:SCALE

Syntax 1

MATH<x>:SCALE <NR3>

Syntax 2

MATH<x>:SCALE?

Argument

- <NR3>

This is the scale, in volts, amps or watts per division. The range is from 100.0E-36 through 100.0E+36.

Example 1

```
MATH4:SCALE 100E-03
```

This command sets the Math 4 scale to 100 mV per division.

Example 2

```
CH2:SCALE?
```

This query might return

```
:MATH2:SCALE 1.0000E+00
```

Indicating that the current scale setting of Math 2 is 1 volt per division.

MATH<x>:SPECTral?

Description

This query-only command returns the current spectral setups for the specified math waveform. This command is equivalent to selecting Spectral Setup from the Math menu and viewing the current spectral setup values.

Group

Math

Syntax

```
MATH<x>:SPECTral?
```

Example

```
MATH1:SPECTral?
```

This query might return

```
:MATH1:SPECTRAL:MAG DB;PHASE DEGREES  
;GATEPOS 0.0000E+00;GATEWIDTH  
1.9996E-06;REFLEVEL 4.4587+01  
;REFLEVELOFFSET 2.2360E-01;SPAN  
1.2500E+09;CENTER 6.2500E+08  
;RESBW 1.0002E=06;WINDOW GAUSSIAN;SUPPRESS  
-3.5000E+01  
;UNWRAP 0;LOCK 0
```

MATH<x>:SPECTral:CENTER

Description

This command specifies or returns the center frequency of the spectral analyzer output data span. This command is equivalent to selecting Spectral Setup from the Math menu and then entering a Center Freq value.

Group

Math

Related Commands

MATH<x>:SPECTral:SPAN

Syntax 1

MATH<x>:SPECTral:CENTER <NR3>

Syntax 2

MATH<x>:SPECTral:CENTER?

Argument

- <NR3>

This is the desired frequency of the spectral analyzer output data span. Units are represented in hertz.

Example 1

```
MATH3:SPECTral:CENTER 10.09E6
```

This command sets the center frequency to the closest value it can attain to 10.09 MHz.

Example 2

```
MATH2:SPECTral:CENTER?
```

This query might return

```
:MATH2:SPECTral:CENTER 10.0900E+06
```

Indicating that the center frequency is currently set at 10.09 MHz.

MATH<x>:SPECTral:GATEPOS

Description

This command sets or returns the position of the center of the gate, which is used as the data input to the spectral analyzer. This command is equivalent to selecting Spectral Setup from the Math menu and then entering a Gate Pos value.

Group

Math

Related Commands

MATH<x>:SPECTral:GATEWIDTH

Syntax 1

```
MATH<x>:SPECTral:GATEPOS <NRf>
```

Syntax 2

```
MATH<x>:SPECTral:GATEPOS?
```

Argument

- <NRf>

This is the gate position. Units are represented in seconds, with respect to trigger position.

Example 1

```
MATH1:SPECTral:GATEPOS 0
```

This command specifies the position of the center edge of the gate used as the data input to the spectral analyzer.

Example 2

```
MATH2:SPECTral:GATEPOS?
```

This query might return

```
:MATH2:SPECTRAL:GATEPOS 0.0000E+00
```

MATH<x>:SPECTral:GATEWIDTH

Description

This command sets or returns the gate width input, in seconds, to the spectral analyzer. This command is equivalent to selecting Spectral Setup from the Math menu and then entering a value in the Duration box. This command is equivalent to selecting Spectral Setup from the Math menu and then entering a Gate Dur value.

Group

Math

Related Commands

MATH<x>:SPECTral:GATEPOS

Syntax 1

MATH<x>:SPECTral:GATEWIDTH <NR3>

Syntax 2

MATH<x>:SPECTral:GATEWIDTH?

Argument

- <NR3>

This is the time across the 10-division screen. Units are represented in seconds.

Example 1

```
MATH1:SPECTral:GATEWIDTH 1.0E-3
```

This command sets the gate width input to the spectral analyzer at 1 ms.

Example 2

```
MATH3:SPECTral:GATEWIDTH?
```

This query might return

```
:MATH3:SPECTRAL:GATEWIDTH 1.0000E-03
```

indicating that the gate width to the spectral analyzer is set to 1 ms.

MATH<x>:SPECTral:LOCK

Description

This command locks menus for Math<x> and Math<y> together as a group. The query form of this command returns an ON (1) or OFF (0), indicating whether spectral locking is turned on. This command is equal to selecting Spectral Setup from the Math menu, choosing the Control tab and then clicking the Time/Track Frequency Domain Controls button associated with the math waveforms that you want to lock. However, applying spectral locking functionality from the interface is limited to locking Math1 and Math2 and/or Math 3 and Math4.

MATH<x> Lock Combinations			
MATH1	MATH2	MATH3	Locked Math Waveform
Off	Off	Off	None
Off	Off	On	Math3 and Math4 locked
Off	On	Off	Math2 and Math3 locked
Off	On	On	Math2, Math3 and Math4 locked
On	Off	Off	Math1 and Math2 locked
On	Off	On	Math1 and Math 2 locked, Math3 and Math4 locked
On	On	Off	Math1, Math2 and Math3 locked
On	On	On	Math1, Math2, Math3 and Math4 locked

Note: Executing MATH4:SPECTral:LOCK via the GPIB interface has no affect since there is no Math5 to which it can be locked.

Group

Math

Syntax 1

MATH<x>:SPECTral:LOCK {ON|OFF}

Syntax 2

MATH<x>:SPECTral:LOCK?

Arguments

- ON

This turns on the parameter lock for the specified math waveform.

- OFF

This turns off the parameter lock for the specified math waveform.

- <NR1>

A 0 disables the parameter lock for the specified math waveform; any other value enables the parameter lock.

Example 1

```
MATH1:SPECTral:LOCK ON
```

This command turns on the parameter Lock1, which causes the parameter for Math1 and Math2 to be locked together.

Example 2

```
MATH1:SPECTral:LOCK?
```

This query might return

```
:MATH1:SPECTRAL:LOCK 0
```

indicating that the parameter Lock1 is turned off.

MATH<x>:SPECTral:MAG

Description

This command controls or returns the units of a Spectral-Mag function in the math string. This command is equal to selecting Spectral Setup from the Math menu, choosing the Mag tab and then clicking the desired Scale button.

Group

Math

Related Commands

MATH<x>:SPECTral:PHASE

Syntax 1

MATH<x>:SPECTral:MAG {LINEAR|DB|DBM}

Syntax 2

MATH<x>:SPECTral:MAG?

Arguments

- LINEAR

This sets the SpectralMag units to linear.

- DB

This sets the SpectralMag units to decibels.

- DBM

This sets the SpectralMag units to decibels. It also sets the Ref Level Offset to a value that is the equivalent of 1mW into 50Ω.

Example 1

```
MATH2:SPECTral:MAG DB
```

This command sets the SpectralMag units for Math2 to decibels.

Example 2

```
MATH2:SPECTral:MAG?
```

This query might return

```
:MATH2:SPECTRAL:MAG DB
```

indicating that the SpectralMag units for Math2 are set to decibels.

MATH<x>:SPECTral:PHASE

Description

This command controls or returns the units of a Spectral-Phase function in the math string. This command is equal to selecting Spectral Setup from the Math menu, choosing the Phase tab and then clicking the desired Scale button.

Group

Math

Related Commands

MATH<x>:SPECTral:MAG

Syntax 1

MATH<x>:SPECTral:PHASE {DEGrees|RADians|GROUPDelay}

Syntax 2

MATH<x>:SPECTral:PHASE?

Arguments

- DEGrees

This sets the SpectralPhase units to degrees.

- RADians

This sets the SpectralPhase units to radians.

- GROUPDelay

This sets the SpectralPhase units to groupdelay, which commutes the derivative of unwrapped phase spectrum. Units are expressed in seconds.

Example 1

```
MATH2:SPECTral:PHASE DEGREES
```

This command sets the SpectralPhase units for Math2 to degrees.

Example 2

```
MATH2:SPECTral:PHASE?
```

This query might return

```
:MATH2:SPECTRAL:PHASE RADIANS
```

indicating that the SpectralPhase units for Math2 are set to radians.

MATH<x>:SPECTral:REFLEVEL

Description

This command specifies the vertical position of the output data from the spectral analyzer on the display screen. The numerical value represents the position at the top of the display graticule. This command is equal to selecting Spectral Setup from the Math menu, choosing the Mag tab and then entering a value in the Reference Level box.

Group

Math

Related Commands

MATH<x>:SPECTral:REFLEVELOffset

Syntax 1

MATH<x>:SPECTral:REFLEVEL <NR3>

Syntax 2

MATH<x>:SPECTral:REFLEVEL?

Argument

- <NR3>

This is the position that represents the top of the display screen graticule. The range depends on the units and both the MATH<x>:SCALE and MATH<x>:POSITION settings.

Example 1

```
MATH1:SPECTRAL:REFLEVEL -10
```

This sets the top of the display screen to be a reference level of -10 dB. If the vertical scale is LINEAR, then the vertical units will be the same as the input waveform.

Example 2

```
MATH1:SPECTRAL:REFLEVEL?
```

This query might return

```
:MATH1:SPECTRAL:REFLEVEL 2.0000E+01
```

indicating that the top of the display screen is set to a reference level of 20 dB.

MATH<x>:SPECTral:REFLEVELOffset

Description

This command sets or returns the spectral level offset used for calculating the dB value. Changing the reference level offset causes the spectral waveform to move vertically, with respect to zero dB. This command is equal to selecting Spectral Setup from the Math menu, choosing the Mag tab and then entering a value in the Reference Level Offset box.

Group

Math

Related Commands

MATH<x>:SPECTral:REFLEVEL

Syntax 1

```
MATH<x>:SPECTral:REFLEVELOffset  
{DBM|<NR3>}
```

Syntax 2

```
MATH<x>:SPECTral:REFLEVELOffset?
```

Arguments

- DBM

This specifies the reference level used for calculation to be equivalent to 1 mW into 50 Ω (Zero dB will occur at this level).

- <NR3>

This specifies the reference level used for calculation of the decibel value when the output units are Log.

Example 1

```
MATH1:SPECTral:REFLEVELOffset 0.5
```

This sets the reference level for the Log calculation for decibel to be 0.5. $\text{dB} = A \times \text{Log}(y/\text{<NR3>})$ where A is 10 if the input units are watts and A is 20 if the input units are otherwise.

Example 2

```
MATH1:SPECTral:REFLEVELOffset DBM
```

This sets the decibel reference to be equivalent to 1 mW into 50 Ω . The reference level numerical value will depend on the input units. If the units are volts, the value is set to 0.2236 V; if the units are amperes, the value is set to 40 μ A; if the units are watts, the value is set to 0.001 W.

Example 3

```
MATH1:SPECTral:REFLEVELOffset?
```

This query might return

```
:MATH1:SPECTRAL:REFLEVELOFFSET 2.23360E-01
```

indicating that the spectral reference level offset is 223.6 mV.

MATH<x>:SPECTral:RESBw

Description

This command sets or returns the resolution bandwidth of the spectral analyzer. This command is equivalent to selecting Spectral Setup from the Math menu and then entering a value in the Res BW box.

Group

Math

Related Commands

MATH<x>:SPECTral:GATEWIDTH, MATH<x>:SPECTral:CENTER, MATH<x>:SPECTral:SPAN, MATH<x>:SPECTral:WINDow

Syntax 1

```
MATH<x>:SPECTral:RESBw <NR3>
```

Syntax 2

```
MATH<x>:SPECTral:RESBw?
```

Argument

- <NR3>

This is the desired resolution bandwidth value. Units are represented in hertz.

Example 1

```
MATH1:SPECTral:RESBw 250E3
```

This command sets the resolution bandwidth to the attainable value that is close to 250 KHz.

Example 2

```
MATH1:SPECTral:RESBw?
```

This query might return

```
:MATH1:SPECTRAL:RESBW 1.0002E+06
```

indicating the actual resolution bandwidth value obtained from the spectral analyzer.

MATH<x>:SPECTral:SPAN

Description

This command sets the ceiling of the span to a value that is closest to the specified value. The query form of this command returns the current span value for specified math waveform. This command is equal to selecting Spectral Setup from the Math menu and then entering a value in the Freq Span box.

Group

Math

Related Commands

MATH<x>:SPECTral:CENTER

Syntax 1

MATH<x>:SPECTral:SPAN {<NR3>|FULL}

Syntax 2

MATH<x>:SPECTral:SPAN?

Arguments

- <NR3>

This specifies the frequency span of the output data vector from the spectral analyzer.

- FULL

This sets the top of the span to 1/2 the sample rate and sets the Center frequency to 1/2 the span.

Example 1

```
MATH1:SPECTral:SPAN FULL
```

This command sets the top of the span to 1/2 the sample rate and sets the Center frequency to 1/2 the span.

Example 2

```
MATH1:SPECTral:SPAN 2.56E6
```

This command sets the top of the span to a value that is closest to 2.56 MHz.

Example 3

`MATH1:SPECTral:SPAN?`

This query might return

`:MATH1:SPECTRAL:SPAN 1.2500E+09`

indicating the actual span value obtained by the spectral analyzer.

MATH<x>:SPECTral:SUPPress

Description

This command sets or returns the phase suppression threshold. This command is equal to selecting Spectral Setup from the Math menu, choosing the Phase tab and then entering a value in the Suppression Threshold box.

Group

Math

Related Commands

MATH<x>:SPECTral:UNWRap

Syntax 1

```
MATH<x>:SPECTral:SUPPress <NR3>
```

Syntax 2

```
MATH<x>:SPECTral:SUPPress?
```

Argument

- <NR3>

This is the level under which all data with magnitudes are displayed as zero phase.

Example 1

```
MATH1:SPECTral:SUPPress -62
```

This command specifies that any magnitude values less than -62 dB will have their phase output set to zero.

Example 2

```
MATH1:SPECTral:SUPPress?
```

This query might return

```
:MATH1:SPECTRAL:SUPPRESS -3.5000E+01
```

indicating that the phase suppression threshold is currently set to -35 dB.

MATH<x>:SPECTral:UNWRap

Description

This command specifies or returns whether or not phase unwrap of the spectral analyzer output data is enabled. This command is equal to selecting Spectral Setup from the Math menu, choosing the Phase tab and then clicking the Unwrap button.

Group

Math

Related Commands

MATH<x>:SPECTral:SUPPress

Syntax 1

MATH<x>:SPECTral:UNWRap <ON|OFF>

Syntax 2

MATH<x>:SPECTral:UNWRap?

Arguments

- ON

This enables phase unwrap.

- OFF

This disables phase wrap.

- <NR1>

A 0 disables phase wrap; any other value enables phase wrap.

Example 1

```
MATH1:SPECTral:UNWRap ON
```

This command enables phase wrap of the spectral analyzer output data.

Example 2

```
MATH1:SPECTral:UNWRap?
```

This query might return

```
:MATH1:SPECTRAL:UNWRAP 0
```

indicating that phase unwrap of the spectral analyzer output data is disabled.

MATH<x>:SPECTral:WINDow

Description

This command sets or returns the window function used to multiply the spectral analyzer input data. A spectral window determines what the filter shape of the spectral analyzer will be in the frequency domain. It can be described by a mathematical function that is multiplied point-by-point times the input data to the spectral analyzer. This command is equal to selecting Spectral Setup from the Math menu, and then choosing from the Window Type drop-down list.

Following is a list of arguments that specify the window function used to multiply the spectral analyzer input data. The windows are listed in the order of their ability to resolve frequencies (resolution bandwidth). For additional information about spectral windows, see *Selecting a Spectral Window* in the online help for this oscilloscope.

Group

Math

Related Commands

MATH<x>:SPECTral:RESBw

Syntax 1

```
MATH<x>:SPECTral:WINDow {RECTANGular|HAM-  
Ming|HANNing  
|KAISERBessel|BLACKMANHarris|FLAT-  
TOP2|GAUSSian|TEKEXPonential}
```

Syntax 2

```
MATH<x>:SPECTral:WINDow?
```

Arguments

- RECTANGular

This type of window function is equivalent to multiplying all gate data by one.

- HAMMing

This type of window function is based on cosine series.

- HANNing

This type of window function is based on cosine series.

- KAISERBessel

This type of window function is based on cosine series.

- BLACKMANHarris

This type of window function is based on cosine series.

- GAUSSian

This type of window function has the best localization characteristics in the joint time/frequency plane.

- TEKEXPonential

This type of window has an exponential non-symmetrical shape in the time domain and a triangular shape in the frequency domain.

- FLATTOP2

This type of window function is a cosine series window with a flattened frequency response lobe.

Example 1

```
MATH2:SPECTral:WINDow HANNING
```

This command specifies to apply a Hanning window to the spectral analyzer input data.

Example 2

```
MATH2:SPECTral:WINDow?
```

This query might return

```
:MATH2:SPECTRAL:WINDOW TEKEXPONENTIAL
```

indicating the window function used to multiply the spectral analyzer input data.

MATHVAR:VAR<n>

Description

This command sets or returns the value of VAR<x>, where <n> can be 1.8.

Group

Math

Syntax 1

```
MATHVAR:VAR<n> <nr3>
```

Syntax 2

```
MATHVAR:VAR<n>?
```

Argument

- VAR<n> value

Example

```
MATHVAR:VAR8 3.14
```

MEASUrement?

Description

This query-only command returns all measurement parameters in the following order: MEAS1, MEAS2, MEAS3, MEAS4, MEAS5, MEAS6, MEAS7, MEAS8, IMMED.

Group

Measurement

Syntax

MEASUrement?

Example

MEASUrement?

This query might return

```
:MEASUREMENT:GATING OFF
;METHOD HISTOGRAM;IMMED:TYPE UNDE-
FINED;UNITS "V"
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1
RISE;EDGE2 RISE
;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS1:STATE 0
;TYPE UNDEFINED;UNITS "V";SOURCE1
CH1;SOURCE2 CH1
;DELAY:EDGE1 RISE;EDGE2 RISE;DIRECTION
FORWARDS;
:MEASUREMENT:MEAS2:STATE 0;TYPE UNDE-
FINED;UNITS "V"
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1
RISE;EDGE2 RISE
;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS3:STATE 0
;TYPE UNDEFINED;UNITS "V";SOURCE1
CH1;SOURCE2 CH1;DELAY
:EDGE1 RISE;EDGE2 RISE;DIRECTION FORWARDS;
:MEASUREMENT:MEAS4:STATE 0;TYPE UNDE-
FINED;UNITS "V"
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1
RISE;EDGE2 RISE
;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS5:STATE 0
;TYPE UNDEFINED;UNITS "V";SOURCE1
CH1;SOURCE2 CH1
;DELAY:EDGE1 RISE;EDGE2 RISE;DIRECTION
FORWARDS;
:MEASUREMENT:MEAS6:STATE 0;TYPE UNDE-
```

```
FINED;UNITS "V"  
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1  
RISE;EDGE2 RISE  
;DIRECTION FORWARDS;:MEA-  
SUREMENT:MEAS7:STATE 0  
;TYPE UNDEFINED;UNITS "V";SOURCE1  
CH1;SOURCE2 CH1;DELAY  
:EDGE1 RISE;EDGE2 RISE;DIRECTION FORWARDS;  
:MEASUREMENT:MEAS8:STATE 0;TYPE UNDE-  
FINED;UNITS "V"  
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1  
RISE;EDGE2 RISE  
;DIRECTION FORWARDS;:MEASUREMENT:REFLE-  
VEL:METHOD PERCENT  
;ABSOLUTE:HIGH 0.0000;LOW 0.0000;MID1  
0.0000;MID2 0.0000;  
:MEASUREMENT:REFLEVEL:PERCENT:HIGH  
90.0000;LOW 10.0000  
;MID1 50.0000;MID2 50.0000;:MEA-  
SUREMENT:STATISTICS:MODE  
OFF;WEIGHTING 32
```

MEASUrement:ANNOtation:IMMEDStAte

Description

This command assigns measurement annotation to the immediate measurement, allowing the GPIB user to obtain annotated measurement values.

Note: The instrument will not display markers on screen if measurement annotation is assigned to the immediate measurement. Setting this command to ON will set the MEASUrement:ANNOtation:StAte to OFF; setting MEASUrement:ANNOtation:StAte to values other than OFF, will set MEASUrement:ANNOtation:IMMEDStAte to OFF.

Group

Measurement

Syntax 1

```
MEASUrement:ANNOtation:IMMEDStAte  
{ON|OFF|<NR1>}
```

Arguments

- ON

Turns on measurement annotation for the immediate measurement.

- OFF

Turns off measurement annotation for the immediate measurement.

- <NR1>

A 0 disables measurement annotation for the immediate measurement; any other value enables measurement annotation for the immediate measurement.

Example 1

```
MEASUrement:ANNOtation:IMMEDStAte ON
```

This command assigns measurement annotation to the immediate measurement and turns MEASUrement:ANNOtation:TYPe to OFF.

MEASUrement:ANNOtation:STATE

Description

This command assigns measurement annotation to the specified measurement.

Note: Unlike operation of the instrument user interface, measurement annotation does not follow a "last selected" measurement model if implemented from the GPIB. Changing or adding a measurement does not change the MEASUrement:ANNOtation:STATE.

Setting MEASUrement:ANNOtation to MEAS<x> will change MEASUrement:ANNOtation:IMMEDState to OFF.

Group

Measurement

Syntax 1

```
MEASUrement:ANNOtation:STATE  
{OFF | MEAS1...MEAS8}
```

Arguments

- OFF

Turns off measurement annotation for the specified measurement.

- MEAS1...MEAS8

Turns on measurement annotation for the specified measurement.

Example 1

```
MEASUrement:ANNOtation:STATE MEAS1
```

This command specifies that measurement annotation is for measurement 1, and turns on the appropriate markers.

MEASUREMENT:ANNOTATION:TYPE

Description

This command sets the number of annotation markers displayed on screen. The number of annotation values can be returned by sending the MEASUREMENT:ANNOTATION:X<x>? and MEASUREMENT:ANNOTATION:Y<x>? commands, regardless of the DETAILED state.

Group

Measurement

Syntax 1

```
MEASUREMENT:ANNOTATION:TYPE {STANDARD|DETAILED}
```

Arguments

■ STANDARD

Uses standard annotation markers specified in the instrument Utilities > User Preferences > Measurement > Standard tab.

■ DETAILED

Uses additional annotation markers specified in the instrument Utilities > User Preferences > Measurement > Detailed tab.

Example 1

```
MEASUREMENT:ANNOTATION:TYPE DETAILED
```

This command specifies standard annotation markers plus additional "detailed" markers.

MEASUrement:ANNOtation:NUMX?

Description

This query-only command returns the maximum number of horizontally-related annotation values defined for the annotated measurement.

Group

Measurement

Syntax 1

```
MEASUrement:ANNOtation:NUMX?
```

Example 1

```
MEASUrement:ANNOtation:NUMX?
```

This command might return `:MEASUREMENT:ANNOtation:NUMX2` indicating that the number of measurement annotations is 2.

MEASUrement:ANNOtation:NUMY?

Description

This query-only command returns the maximum number of vertically-related annotation values defined for the annotated measurement.

Group

Measurement

Syntax 1

```
MEASUrement:ANNOtation:NUMY?
```

Example 1

```
MEASUrement:ANNOtation:NUMY?
```

This command might return `:MEASUREMENT:ANNOtation:NUMY2` indicating that the number of measurement annotations is 2.

MEASUrement:ANNOtation:X<x>?

Description

This query-only command returns the value of the Xth horizontally-related measurement annotation.

Group

Measurement

Syntax 1

```
MEASUrement:ANNOtation:X<x>?
```

Example 1

```
MEASUrement:ANNOtation:X2?
```

This command might return :MEAS:ANNOTATION:X2
9.9E-6.

MEASUrement:ANNOtation:Y<x>?

Description

This query-only command returns the value of the Xth vertically-related measurement annotation.

Group

Measurement

Syntax 1

```
MEASUrement:ANNOtation:Y<x>?
```

Example 1

```
MEASUrement:ANNOtation:Y2?
```

This command might return :MEAS:ANNOTATION:Y2
9.9E-2.

MEASUrement:GATing

Description

This command specifies or returns the measurement gating setting. This command is equivalent to selecting Gating from the Measure menu and then clicking the desired Measurement Gating setting.

Group

Measurement

Syntax 1

```
MEASUrement:GATing  
{ZOOM | ZOOM1 | ZOOM2 | ZOOM3 | ZOOM4 | CUR-  
Sor | ON | OFF | <NRf>}
```

Syntax 2

```
MEASUrement:GATing?
```

Arguments

■ ZOOM

This turns on gating, using the left and right edges of the zoom box.

■ ZOOM1 | ZOOM2 | ZOOM3 | ZOOM4

This turns on gating for the channel specified (1 through 4), using the left and right edges of the zoom box.

■ CURSOR

This limits measurements to the portion of the waveform between the vertical bar cursors, even if they are off screen.

■ ON

This turns on measurement gating.

■ OFF

This turns off measurement gating.

■ <NRf>

A 0 turns off measurement gating; any other value turns on measurement gating.

Example 1

```
MEASUrement:GATing ON
```

This command turns on measurement gating.

Example 2

MEASUREMENT:GATING?

This query might return

:MEASUREMENT:GATING CURSOR

indicating that measurements are limited to the portion of the waveform between the vertical bar cursors.

MEASUrement:IMMed?

Description

This query-only command returns all immediate measurement setup parameters.

Group

Measurement

Syntax

```
MEASUrement:IMMed?
```

Example

```
MEASUrement:IMMed?
```

This query might return

```
:MEASUREMENT:IMMED:TYPE PERIOD;UNITS  
"s";SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1  
RISE;EDGE2 RISE;DIRECTION FORWARDS
```

MEASUrement:IMMed:DELaY?

Description

This query-only command returns information about the immediate delay measurement. This command is equivalent to selecting Measurement Setup from the Measure menu, choosing the Time tab and then clicking the Delay button.

Group

Measurement

Syntax

```
MEASUrement:IMMed:DELaY?
```

Example

```
MEASUrement:IMMed:DELaY?
```

This query might return

```
:MEASUREMENT:IMMED:DELAY:EDGE1 RISE;EDGE2  
RISE; DIRECTION FORWARDS
```

MEASUrement:IMMed:DELAy:DIREction

Description

This command sets or returns the starting point and direction that determines the delay "to" edge when taking an immediate delay measurement. Use the MEASUrement:IMMed:SOURCE2 command to specify the delay "to" waveform. This command is equivalent to selecting Measurement Setup from the Measure menu, choosing the Time tab, clicking the Delay button to display the delay settings and then clicking the desired Search Direction setting.

Group

Measurement

Related Commands

MEASUrement:IMMed:SOURCE2

Syntax 1

```
MEASUrement:IMMed:DELAy:DIREction {BACK-  
Wards|FORWards}
```

Syntax 2

```
MEASUrement:IMMed:DELAy:DIREction?
```

Arguments

■ BACKWards

This starts the search at the right end of the waveform and looks for the last rising or falling edge in the waveform.

■ FORWards

This starts the search at the beginning (or left end) of the waveform and looks for the first rising or falling edge in the waveform.

Example 1

```
MEASUrement:IMMed:DELAy:DIREction FORWards
```

This command starts searching from the beginning (or left end) of the waveform record and looks for the first rising or falling edge.

Example 2

MEASUREMENT:IMMED:DELAY:DIRECTION?

This query might return

:MEASUREMENT:IMMED:DELAY:DIRECTION BACK-
WARDS

indicating that searching begins at the right end of the waveform record and looks for the last rising or falling edge.

MEASUREMENT:IMMED:DELAY:EDGE[1]

Description

This command sets or queries the slope of the edge that is used for the delay "from" waveform when taking an immediate delay measurement. Use the MEASUREMENT:IMMED:SOURCE1 command to specify the waveform. This command is equivalent to selecting Measurement Setup from the Measure menu, choosing the Time tab, clicking the Delay button to display the delay settings and then clicking the desired Delay Edge1 setting.

Group

Measurement

Related Commands

MEASUREMENT:IMMED:SOURCE1

Syntax 1

```
MEASUREMENT:IMMED:DELAY:EDGE[1]  
{FALL|RISe}
```

Syntax 2

```
MEASUREMENT:IMMED:DELAY:EDGE[1]?
```

Arguments

- FALL

This specifies the falling edge.

- RISe

This specifies the rising edge.

Example 1

```
MEASUREMENT:IMMED:DELAY:EDGE[1] RISe
```

This command specifies that the rising edge be used for the immediate delay measurement .

Example 2

```
MEASUREMENT:IMMED:DELAY:EDGE[1]?
```

This query might return

```
MEASUREMENT:IMMED:DELAY:EDGE1 FALL
```

indicating that the falling edge or negative edge of the waveform is used for the immediate delay measurement.

MEASUREMENT:IMMED:DELAY:EDGE2

Description

This command sets or queries the slope of the edge that is used for the delay "to" waveform when taking an immediate delay measurement. Use the MEASUREMENT:IMMED:SOURCE2 command to specify the waveform. This command is equivalent to selecting Measurement Setup from the Measure menu, choosing the Time tab, clicking the Delay button to display the delay settings and then clicking the desired Delay Edge2 setting.

Group

Measurement

Related Commands

MEASUREMENT:IMMED:SOURCE2

Syntax 1

```
MEASUREMENT:IMMED:DELAY:EDGE2 {FALL|RISe}
```

Syntax 2

```
MEASUREMENT:IMMED:DELAY:EDGE2?
```

Arguments

- FALL

This specifies the falling edge.

- RISe

This specifies the rising edge.

Example 1

```
MEASUREMENT:IMMED:DELAY:EDGE2 RISe
```

This command specifies that the rising edge be used for the immediate delay measurement .

Example 2

```
MEASUREMENT:IMMED:DELAY:EDGE2?
```

This query might return

```
MEASUREMENT:IMMED:DELAY:EDGE2 FALL
```

indicating that the falling edge or negative edge of the waveform is used for the immediate delay measurement.

MEASUREMENT:IMMED:NOISE

Description

This command sets or queries whether the noise measurement is made on the high or low level of the waveform. Sending this command is equivalent to selecting Ref Levs > Eye > Top Level or Base Level in the Comm tab of the Measurement Setup dialog box. The Eye section is displayed only if you have an eye-pattern or optical measurement defined.

Group

Measurement

Syntax 1

```
MEASUREMENT:IMMED:NOISE {HIGH|LOW}
```

Syntax 2

```
MEASUREMENT:IMMED:NOISE?
```

Arguments

- HIGH

This argument causes the measurement for noise to be taken at the high level of the waveform.

- LOW

This argument causes the measurement for noise to be taken at the low level of the waveform.

Example 1

```
MEASUREMENT:IMMED:NOISE HIGH
```

This command causes the noise measurement to be made at the high level of the waveform.

Example 2

```
MEASUREMENT:IMMED:NOISE?
```

This query might return

```
:MEASUREMENT:IMMED:NOISE LOW
```

indicating that the noise measurement will be made at the low level of the waveform.

MEASUREMENT:IMMED:REFLEVEL?

Description

This query-only command returns the reference level settings for the immediate measurement. It returns them in the following order: ABSOLUTE and then PERCENT for individual user measurements.

Group

Measurement

Syntax

```
MEASUREMENT:IMMED:REFLEVEL?
```

Example

```
MEASUREMENT:IMMED:REFLEVEL?
```

MEASUrement:IMMed:REFLevel:ABSolute:HIGH

Description

This command sets or queries the high reference level, and is the upper reference level when MEASUrement:IMMed:REFLevel:METHOD is set to Absolute. Note that this command affects the results of rise and fall measurements.

Group

Measurement

Related Commands

MEASUrement:IMMed:REFLevel:METHOD

Syntax 1

```
MEASUrement:IMMed:REFLevel:ABSolute:HIGH  
<NRf>
```

Syntax 2

```
MEASUrement:IMMed:REFLevel:ABSolute:HIGH?
```

Argument

- <NRf>

This is the high reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:IMMed:REFLEVEL:ABSOLUTE:HIGH  
1.71
```

This command sets the high reference level to 1.71 V.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:HIGH?
```

This query might return

```
:MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:HIGH  
1.7100E+00
```

Indicating that the absolute high reference level is set to 1.71 V.

MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:LOW

Description

This command sets or queries the low reference level, and is the zero percent level when MEASUREMENT:IMMED:REFLEVEL:METHOD is set to Absolute. Note that this command affects the results of rise and fall measurements.

Group

Measurement

Related Commands

MEASUREMENT:IMMED:REFLEVEL:METHOD

Syntax 1

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:LOW  
<NRf>
```

Syntax 2

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:LOW?
```

Argument

- <NRf>

This is the low reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:LOW  
0.0 V.
```

This command sets the low reference level to 0.0 V.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:LOW?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:LOW  
0.0000E+00
```

Indicating that the absolute low reference level is set to 0.0 V.

MEASUREMENT:IMMED:REFLevel:ABSolute:MID[1]

Description

This command sets or queries the mid reference level, and is the 50% reference level when MEASUREMENT:IMMED:REFLevel:METHod is set to Absolute. Note that this command affects the results of period, frequency, delay, and all cyclic measurements.

Group

Measurement

Related Commands

MEASUREMENT:IMMED:REFLevel:METHod

Syntax 1

```
MEASUREMENT:IMMED:REFLevel:ABSolute:MID[1]
<NRf>
```

Syntax 2

```
MEASUREMENT:IMMED:REFLevel:ABSolute:MID[1]?
```

Argument

- <NRf>

This is the mid reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:MID[1]
.71
```

This command sets the mid reference level to .71 V.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:MID[1]?
```

This query might return

```
:MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:MID
0.7100E+00
```

Indicating that the absolute mid1 reference level is set to .71 V.

MEASUREMENT:IMMED:REFLevel:ABSolute:MID2

Description

This command sets or queries the mid reference level for the "to" waveform when taking a delay measurement, and is the 50% reference level when MEASUREMENT:IMMED:REFLevel:METHod is set to Absolute. Note that this command affects the results of delay measurements.

Group

Measurement

Related Commands

MEASUREMENT:IMMED:REFLevel:METHod

Syntax 1

```
MEASUREMENT:IMMED:REFLevel:ABSolute:MID2
<NRf>
```

Syntax 2

```
MEASUREMENT:IMMED:REFLevel:ABSolute:MID2?
```

Argument

- <NRf>

This is the mid2 reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:MID2
0.5
```

This command sets the mid reference level for the delay waveform to 0.5 V.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:MID2?
```

This query might return

```
:MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE:MID2
0.5000E+00
```

Indicating that the absolute mid2 reference level is set to 0.5 V.

MEASUREMENT:IMMED:REFLEVEL:METHOD

Description

This command specifies or queries the reference level units used for measurement calculations. This command is equivalent to selecting Reference Levels from the Measure menu and then choosing the desired reference level from the Units group box.

Group

Measurement

Syntax 1

```
MEASUREMENT:IMMED:REFLEVEL:METHOD {ABSOLUTE|PERCENT}
```

Syntax 2

```
MEASUREMENT:IMMED:REFLEVEL:METHOD?
```

Arguments

■ ABSOLUTE

This specifies that the reference levels are set explicitly using the MEASUREMENT:IMMED:REFLEVEL:ABSOLUTE commands. This method is useful when precise values are required. For instance, when designing to published interface specifications, such as RS-232-C.

■ PERCENT

This specifies that the reference levels are calculated as a percent relative to HIGH and LOW. The percentages are defined using the MEASUREMENT:IMMED:REFLEVEL:PERCENT commands.

Example 1

```
MEASUREMENT:IMMED:REFLEVEL:METHOD ABSOLUTE
```

This command specifies that explicit user-defined values are used for the reference levels.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:METHOD?
```

This query might return

```
:MEASUREMENT:IMMED:REFLEVEL:METHOD PERCENT
```

Indicating that the reference level units used are calculated as a percent relative to HIGH and LOW.

MEASUrement:IMMed:REFLevel:PERCent:HIGH

Description

This command sets or queries the percentage (where 100% is equal to HIGH) used to calculate the high reference level when MEASUrement:IMMed:REFLevel:METHOD is set to Percent. Note that this command affects the results of rise and fall measurements.

Group

Measurement

Related Commands

MEASUrement:IMMed:REFLevel:METHOD

Syntax 1

```
MEASUrement:IMMed:REFLevel:PERCent:HIGH  
<NRf>
```

Syntax 2

```
MEASUrement:IMMed:REFLevel:PERCent:HIGH?
```

Argument

- <NRf>

This is the high reference level, ranging from 0 to 100%. The default high reference level is 90%.

Example 1

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:HIGH 95
```

This command sets the high reference level to 95% of HIGH.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:HIGH?
```

This query might return

```
:MEASUREMENT:IMMED:REFLEVEL:PERCENT:HIGH  
90
```

Indicating that the percentage high reference level is set to 90% of HIGH.

MEASUREMENT:IMMED:REFLEVEL:PERCENT:LOW

Description

This command sets or queries the percentage (where 100% is equal to HIGH) used to calculate the low reference level when MEASUREMENT:IMMED:REFLEVEL:METHOd is set to Percent. Note that this command affects the results of rise and fall measurements. This command is equivalent to selecting Reference Levels from the Measure menu and then entering the Percentage Low Ref value.

Group

Measurement

Related Commands

MEASUREMENT:IMMED:REFLEVEL:METHOd

Syntax 1

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:LOW  
<NRf>
```

Syntax 2

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:LOW?
```

Argument

- <NRf>

This is the low reference level, ranging from 0 to 100%. The default low reference level is 10%.

Example 1

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:LOW 15
```

This command sets the low reference level to 15% of HIGH.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:LOW?
```

This query might return

```
:MEASUREMENT:IMMED:REFLEVEL:PERCENT:LOW 10
```

Indicating that the percentage low reference level is set to 10% of HIGH.

MEASUrement:IMMed:REFLevel:PERCent:MID[1]

Description

This command sets or queries the percentage (where 100% is equal to HIGH) used to calculate the mid reference level when MEASUrement:IMMed:REFLevel:METHod is set to Percent. Note that this command affects the results of period, frequency, delay, and all cyclic measurements.

Group

Measurement

Related Commands

MEASUrement:IMMed:REFLevel:METHod

Syntax 1

```
MEASUrement:IMMed:REFLevel:PERCent:MID[1]
<NRf>
```

Syntax 2

```
MEASUrement:IMMed:REFLevel:PERCent:MID[1]?
```

Argument

- <NRf>

This is the mid reference level, ranging from 0 to 100%. The default mid reference level is 50%.

Example 1

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:MID[1]
60
```

This command sets the mid reference level to 60% of HIGH.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:MID[1]?
```

This query might return

```
:MEASUREMENT:IMMED:REFLEVEL:PERCENT:MID 65
```

Indicating that the percentage mid reference level is set to 65% of HIGH.

MEASUrement:IMMed:REFLevel:PERCent:MID2

Description

This command sets or queries the percentage (where 100% is equal to HIGH) used to calculate the mid reference level for the second waveform specified when MEASUrement:IMMed:REFLevel:METHOD is set to Percent. Note that this command affects the results of delay measurements.

Group

Measurement

Related Commands

MEASUrement:IMMed:REFLevel:METHOD

Syntax 1

```
MEASUrement:IMMed:REFLevel:PERCent:MID2
<NRf>
```

Syntax 2

```
MEASUrement:IMMed:REFLevel:PERCent:MID2?
```

Argument

- <NRf>

This is the mid reference level, ranging from 0 to 100%. The default mid reference level is 50%.

Example 1

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:MID2 40
```

This command sets the mid2 reference level to 40% of HIGH.

Example 2

```
MEASUREMENT:IMMED:REFLEVEL:PERCENT:MID2?
```

This query might return

```
:MEASUREMENT:IMMED:REFLEVEL:PERCENT:MID2
45
```

Indicating that the percentage mid2 reference level is set to 45% of HIGH.

MEASUrement:IMMed:SOURCE[1]

Description

This command sets or queries the source for all single channel immediate measurements and specifies the source to measure "from" when taking an immediate delay measurement or phase measurement. This command is equivalent to selecting Measurement Setup from the Measure menu, choosing the Time tab, clicking the Delay button to display the delay settings and then clicking the desired Source1 (From) setting.

Group

Measurement

Related Commands

MEASUrement:IMMed:SOURCE2

Syntax 1

```
MEASUrement:IMMed:SOURCE[1]  
{CH<x>|MATH<y>|REF<x>|HISTogram}
```

Syntax 2

```
MEASUrement:IMMed:SOURCE[1]?
```

Arguments

- CH<x>

This is an input channel. The valid channel waveform range is from 1 through 4.

- MATH<y>

This is a math waveform. The valid math waveform range is from 1 through 4.

- REF<x>

This is a reference waveform. The valid reference waveform range is from 1 through 4.

- HISTogram

This indicates histogram as the object to be measured.

Example 1

```
MEASUrement:IMMed:SOURCE[1] MATH1
```

This command specifies Math1 as the immediate measurement source .

Example 2

MEASUREMENT:IMMED:SOURCE{1}?

This query might return

:MEASUREMENT:IMMED:SOURCE1 CH3

indicating that channel 3 is the immediate measurement source.

MEASUrement:IMMed:SOURCE2

Description

This command sets or queries the source to measure "to" for phase or delay immediate measurements. This command is equivalent to selecting Measurement Setup from the Measure menu, choosing the Time tab, clicking the Delay button to display the delay settings and then clicking the desired Source2 (To) setting.

Tip: Source2 measurements only apply to phase and delay measurement types, which require both a target (Source1) and reference (Source2) source.

Group

Measurement

Related Commands

MEASUrement:IMMed:SOURCE[1]

Syntax 1

```
MEASUrement:IMMed:SOURCE2  
{CH<x> | MATH<y> | REF<x>}
```

Syntax 2

```
MEASUrement:IMMed:SOURCE2?
```

Arguments

- CH<x>

This is an input channel. The valid channel waveform range is from 1 through 4.

- MATH<y>

This is a math waveform. The valid math waveform range is from 1 through 4.

- REF<x>

This is a reference waveform. The valid reference waveform range is from 1 through 4.

Example 1

```
MEASUrement:IMMed:SOURCE2 REF3
```

This command sets the waveform in reference memory location 3 as the phase or delay "to" second source when making phase or delay measurements.

Example 2

MEASUREMENT:IMMED:SOURCE2?

This query might return

:MEASUREMENT:IMMED:SOURCE2 MATH1

indicating that Math1 is the immediate measurement second source.

MEASUREMENT:IMMED:TYPE

Description

This command sets or queries the immediate measurement type.

Group

Measurement

Syntax 1

```
MEASUREMENT:IMMED:TYPE {AMPLitude|AREa|BURst|CARea|CMEan|CRMs|DELay|DISTDuty|EXTINCTDB|EXTINCTPCT|EXTINCTRATIO|EYEHeight|EYEwidth|FALL|FREQuency|HIGH|HITS|LOW|MAXimum|MEAN|MEDian|MINimum|NCROSS|NDuty|NOVershoot|Nwidth|PBASE|PCROSS|PCTCROSS|PDuty|PEAKHits|PERIOD|PHase|PK2Pk|PKPKJitter|PKPKNoise|POVerhoot|PTOP|Pwidth|QFACTOR|RISe|RMS|RMSJitter|RMSnoise|SIGMA1|SIGMA2|SIGMA3|SIXSigmajit|SNRatio|STDdev|UNDEFINED|WAVEFORMS}
```

Syntax 2

```
MEASUREMENT:IMMED:TYPE?
```

Arguments

■ AMPLitude

This measures the amplitude of the selected waveform. In other words, it measures the high value less the low value measured over the entire waveform or gated region.

Amplitude = High - Low

■ AREa

This measures the voltage over time. The area is over the entire waveform or gated region and is measured in volt-seconds. The area measured above the ground is positive, while the area below ground is negative.

■ BURst

The burst width measures the duration of a burst. The measurement is made over the entire waveform or gated region.

■ CARea

The cycle area measures the voltage over time. In other words, it measures, in volt-seconds, the area over the

first cycle in the waveform or the first cycle in the gated region. The area measured above the common reference point is positive, while the area below the common reference point is negative.

■ **CMEan**

This measures the arithmetic mean over the first cycle in the waveform or the first cycle in the gated region.

■ **CRMs**

This measures the true Root Mean Square voltage over the first cycle in the waveform or the first cycle in the gated region.

■ **DELay**

This measures the time between the middle reference (default = 50%) amplitude point of the source waveform and the destination waveform.

■ **DISTDUty**

The duty cycle distortion measures the time between the falling edge and the rising edge of the eye pattern at the mid reference level.

It is the peak-to-peak time variation of the first eye crossing measured at the mid-reference as a percent of the eye period.

■ **EXTINCTDB**

This (optical waveform (eye diagram) measurement) measures its extinction ratio. Extinction Ratio (dB) measures the ratio of the average power levels for the logic High to the logic Low of an optical waveform and expresses the result in dB.

This measurement only works for fast acquisition signals or a reference waveform saved in fast acquisition mode.

$$\text{Extinction dB} = 10 \times (\log_{10} (\text{High} / \text{Low}))$$

■ **EXTINCTPCT**

This (optical measurement) measures the extinction ratio of the selected optical waveform. Extinction Ratio (%) measures the ratio of the average power levels for the logic Low (off) to the logic (High) (on) of an optical waveform and expresses the result in percent.

This measurement only works for fast acquisition signals or a reference waveform saved in fast acquisition mode.

$$\text{Extinction \%} = 100.0 \times (\text{Low} / \text{High})$$

■ **EXTINCTRATIO**

This (optical measurement) measures the extinction ratio of the selected optical waveform. Extinction Ratio mea-

sures the ratio of the average power levels for the logic High to the logic Low of an optical waveform and expresses the result without units.

This measurement only works for fast acquisition signals or a reference waveform saved in fast acquisition mode.

Extinction Ratio = (High / Low), ratios greater than 100 or less than 1 generate errors; low must be greater than or equal to 1 microwatt.

■ **EYEHeight**

This measures the vertical opening of an eye diagram, in volts.

■ **EYEWidth**

This measures the width of an eye diagram, in seconds.

■ **FALL**

This measures the time taken for the falling edge of the first pulse in the waveform or gated region to fall from a high reference value (default is 90%) to a low reference value (default is 10%).

■ **FREQuency**

This measures the first cycle in the waveform or gated region. Frequency is the reciprocal of the period and is measured in hertz (Hz), where 1 Hz = 1 cycle per second.

■ **HIGH**

This measures the High reference (100% level, sometimes called Topline) of a waveform.

You can also limit the High measurement (normally taken over the entire waveform record) to a gated region on the waveform.

■ **HITS**

The hits-in-box histogram measurement measures the number of points in or on the histogram box.

■ **LOW**

This measures the Low reference (0% level, sometimes called Baseline) of a waveform.

■ **MAXimum**

This finds the maximum amplitude. This value is the most positive peak voltage found. It is measured over the entire waveform or gated region.

When histogram is selected with the measurement:method command, the maximum measurement measures the

voltage of the highest nonzero bin in vertical histograms or the time of the rightmost bin in horizontal histograms.

■ **MEAN**

This amplitude measurement finds the arithmetic mean over the entire waveform or gated region.

When histogram is selected with the measurement:method command, the mean measurement measures the average of all acquired points within or on the histogram.

■ **MEDian**

This (histogram measurement) measures the middle point of the histogram box. Half of all acquired points within or on the histogram box are less than this value and half are greater than this value.

■ **MINImum**

This amplitude measurement finds the minimum amplitude. This value is typically the most negative peak voltage. It is measured over the entire waveform or gated region.

When histogram is selected with the measurement:method command, the minimum measurement measures the lowest nonzero bin in vertical histograms or the time of the leftmost nonzero bin in the horizontal histograms.

■ **NCROss**

This (timing measurement) measures the time from the trigger point to the first falling edge of the waveform or gated region. The distance (time) is measured at the middle reference amplitude point of the signal.

■ **NDUty**

The negative duty cycle is the ratio of the negative pulse width to the signal period, expressed as a percentage. The duty cycle is measured on the first cycle in the waveform or gated region.

*Negative Duty Cycle = (Negative Width) / Period * 100%*

■ **NOVershoot**

The negative overshoot amplitude measurement finds the negative overshoot value over the entire waveform or gated region.

*Negative Overshoot = (Low – Minimum) / Amplitude * 100%*

■ **NWIdth**

The negative width measurement is the distance (time) between the middle reference (default = 50%) amplitude

points of a negative pulse. The measurement is made on the first pulse in the waveform or gated region.

■ **PBAsE**

This measures the base value used in extinction ratio measurements.

■ **PCROss**

This (timing measurement) measures the time from the trigger point to the first positive edge of the waveform or gated region. The distance (time) is measured at the middle reference amplitude point of the signal.

■ **PCTCROss**

The crossing percent (optical measurement) measures the location of the eye crossing point expressed as a percentage of EYEHeight.

*Crossing percent = 100 * [(eye-crossing-point – PBAsE)/(PTOP – PBAsE)]*

■ **PDUty**

The positive duty cycle is the ratio of the positive pulse width to the signal period, expressed as a percentage. It is measured on the first cycle in the waveform or gated region.

*Positive Duty Cycle = (Positive Width)/Period * 100%*

■ **PEAKHits**

This (histogram measurement) measures the number of points in the largest bin of the histogram.

■ **PERIOd**

This timing measurement is the time required to complete the first cycle in a waveform or gated region. Period is the reciprocal of frequency and is measured in seconds.

■ **PHAsE**

This measures the phase difference (amount of time a waveform leads or lags the reference waveform) between two waveforms. The measurement is made between the middle reference points of the two waveforms and is expressed in degrees, where 360° represents one waveform cycle.

■ **PK2Pk**

This peak-to-peak amplitude measurement finds the absolute difference between the maximum and minimum amplitude in the entire waveform or gated region.

When histogram is selected with the measurement:method command, the PK2Pk measurement measures the histogram's peak to peak difference.

- **PKPKJitter**

This measures the variance (minimum and maximum values) in the time locations of the cross point.

- **PKPKNoise**

This measures the peak-to-peak noise on a waveform at the mid reference level.

- **POvershoot**

The positive overshoot amplitude measurement finds the positive overshoot value over the entire waveform or gated region.

*Positive Overshoot = (Maximum – High) / Amplitude * 100%*

- **PTOP**

This measures the top value used in extinction ratio measurements.

- **PWidth**

The positive width is the distance (time) between the middle reference (default = 50%) amplitude points of a positive pulse. The measurement is made on the first pulse in the waveform or gated region.

- **QFACTOR**

This measures the quality factor. The Q factor is a figure of merit for an eye diagram, which indicates the vertical eye opening relative to the noise at the low and high logic levels. It is the ratio of the eye size to noise.

- **RISe**

This timing measurement finds the rise time of the waveform. The rise time is the time it takes for the leading edge of the first pulse encountered to rise from a low reference value (default is 10%) to a high reference value (default is 90%).

Rise time = TimeRef High - TimeRef Low

You can also limit the measurement (normally taken over the entire waveform record) to a gated region on the waveform.

- **RMS**

This amplitude measurement finds the true Root Mean Square voltage in the entire waveform or gated region.

- **RMSJitter**

This measures the variance in the time locations of the cross point. The RMS jitter is defined as one standard deviation at the cross point.

- **RMSNoise**

This measures the Root Mean Square noise amplitude on a waveform at the mid reference level.

- **SIGMA1**

This (histogram measurement) measures the percentage of points in the histogram that are within one standard deviation of the histogram mean.

- **SIGMA2**

This (histogram measurement) measures the percentage of points in the histogram that are within two standard deviations of the histogram mean.

- **SIGMA3**

This (histogram measurement) measures the percentage of points in the histogram that are within three standard deviations of the histogram mean.

- **SIXSigmajit**

This (histogram measurement) is six * RMSJitter.

- **SNRatio**

This measures the signal-to-noise ratio. The signal-to-noise ratio is the amplitude of a noise rejection band centered on the mid level.

- **STDdev**

This measures the standard deviation (Root Mean Square (RMS) deviation) of all acquired points within or on the histogram box.

- **UNDEFINED**

This is the default measurement type, which indicates that no measurement type is specified. Once a measurement type is chosen, it can be cleared using this argument.

- **WAVEFORMS**

The waveform count (histogram measurement) measures the number of waveforms used to calculate the histogram.

Example 1

`MEASUREMENT:IMMED:TYPE FREQUENCY`

This defines the immediate measurement to be a frequency measurement.

Example 2

`MEASUREMENT:IMMED:TYPE?`

This query might return

`:MEASUREMENT:IMMED:TYPE RMS`

indicating that the immediate measurement is the true Root Mean Square voltage.

MEASUREMENT:IMMED:UNITs?

Description

This query-only command returns the units of the immediate measurement.

Group

Measurement

Syntax

MEASUREMENT:IMMED:UNITs?

Returns (partial list)

- dB

Decibels. Occurs with communications-extinction ratio measurement.

- hits

Hits. Occurs with histogram measurements, such as hits in box.

- Hz

Hertz

- S

Seconds

- V

Volts

- Vs

Volt seconds. Occurs in area measurements.

- VV

Possible in a case like math1="ch1*ch2"

- W

Watts

- Wfms

Waveforms. Occurs with the histogram-waveform count measurement.

- %

Percent

- "\b0"

This displays as the degrees' symbol. It occurs with phase measurements.

■ ?

Unknown. Possible in a case like $\text{math1}=\text{ch1}+\text{ch2}$ if you try adding a volts measurement with a watts measurement

Example

`MEASUREMENT:IMMED:UNITs?`

This query might return

`:MEASUREMENT:IMMED:UNITs s`

indicating that units for the immediate measurement are represented in seconds.

MEASUrement:IMMed:VALue?

Description

This query-only command returns the value of the measurement specified by the MEASUrement:IMMed:TYPE command. The measurement is immediately taken on the source(s) specified by a MEASUrement:IMMed:SOURCE command.

Note: A change to HORizontal:MAIn:SCALE or CH<x>:SCALE will not necessarily have taken affect if followed by this command.

Group

Measurement

Related Commands

MEASUrement:IMMed:TYPE, MEASUrement:IMMed:SOURCE, *ESR, ALLEV?

Syntax

```
MEASUrement:IMMed:VALue?
```

Example

```
MEASUrement:IMMed:VALue?
```

This query might return

```
:MEASUREMENT:IMMED:VALUE 9.9000E+37
```

If the measurement has an error or warning associated with it, then an item is added to the error queue. The error can be checked for with the *ESR? and ALLEV? commands.

MEASUrement:MEAS<x>?

Description

This query-only command returns all measurement parameters for the displayed measurement specified by x, which can range from 1 through 8. This query command is equivalent to selecting Measurement Setup from the Measure menu and viewing the Measurements table; then choosing the Time tab, clicking the Delay button and viewing the Delay Edge and Search Direction settings.

Group

Measurement

Syntax

```
MEASUrement:MEAS<x>?
```

Example

```
MEASUrement:MEAS3?
```

This query might return

```
:MEASUREMENT:MEAS3:STATE 0;TYPE PE-  
RIOD;UNITS "s";SOURCE1 CH1;DELAY:EDGE1  
RISE;EDGE2 RISE;DIRECTION FORWARDS
```

MEASUrement:MEAS<x>:COUNT?

Description

This query-only command returns the number of values accumulated for this measurement since the last statistical reset. Some values may have been ignored because they generated an error. Measurements are specified by x, which ranges from 1 through 8.

Group

Measurement

Syntax

```
MEASUrement:MEAS<x>:COUNT?
```

Example

```
MEASUrement:MEAS3:COUNT?
```

This query might return

```
:MEASUREMENT:MEAS3:COUNT 3.247000B000E+03
```

MEASUrement:MEAS<x>:DELay?

Description

This query-only command returns the delay measurement parameters for the measurement specified by x, which ranges from 1 through 8. Measurement parameters are presented in the following order: Edge1, Edge2 and Direction. This query command is equivalent to selecting Time from the Measure menu, choosing Delay and then viewing the Delay Edge and Search Direction settings.

Group

Measurement

Syntax 1

```
MEASUrement:MEAS<x>:DELay?
```

Example

```
MEASUrement:MEAS3?
```

This query might return

```
:MEASUREMENT:MEAS3:DELAY:EDGE1 RISE;EDGE2  
RISE;DIRECTION FORWARDS
```

MEASUrement:MEAS<x>:DELay:DIREction

Description

This command sets or queries the starting point and direction that determines the delay "to" edge when taking a delay measurement. Use the MEASUrement:MEAS<x>:SOURCE2 command to specify the waveform. This command is equivalent to selecting Time from the Measure menu, choosing Delay from the drop-down list and then clicking the desired Search Direction setting.

Group

Measurement

Related Commands

MEASUrement:MEAS<x>:SOURCE2

Syntax 1

```
MEASUrement:MEAS<x>:DELay:DIREction {BACK-  
Wards | FORWards }
```

Syntax 2

```
MEASUrement:MEAS<x>:DELay:DIREction?
```

Arguments

- BACKWards

This means that the search starts at the end of the waveform and looks for the last rising or falling edge in the waveform. Use the MEASUrement:MEAS<x>:DELay:EDGE2 command to specify the slope of the edge.

- FORWards

This means that the search starts at the beginning of the waveform and looks for the first rising or falling edge in the waveform. Use the MEASUrement:MEAS<x>:DELay:EDGE2 command to specify the slope of the edge.

Example 1

```
MEASUrement:MEAS3:DELay:DIREction BACK-  
Wards
```

This command starts searching from the end of the waveform record.

Example 2

MEASUrement:MEAS3DELay:DIREction?

This query might return

:MEASUREMENT:MEAS3:DELAY:DIRECTION BACK-
WARDS

indicating the current search direction.

MEASUREMENT:MEAS<x>:DELAY:EDGE[1]

Description

This command sets or queries the slope of the edge that is used for the delay "from" waveform when taking a delay measurement. Use the MEASUREMENT:MEAS<x>:SOURCE[1] command to specify the waveform. This command is equivalent to selecting Time from the Measure menu, choosing Delay from the drop-down list and then clicking the desired Delay Edge1 setting.

Group

Measurement

Related Commands

MEASUREMENT:MEAS<x>:SOURCE[1]

Syntax 1

```
MEASUREMENT:MEAS<x>:DELAY:EDGE[1]  
{FALL|RISe}
```

Syntax 2

```
MEASUREMENT:MEAS<x>:DELAY:EDGE[1]?
```

Arguments

- FALL

This specifies the falling edge of the waveform.

- RISe

This specifies the rising edge of the waveform.

Example 1

```
MEASUREMENT:MEAS3:DELAY:EDGE[1] RISe
```

This command specifies that the rising edge of the waveform be used for measurement 3.

Example 2

```
MEASUREMENT:MEAS1:DELAY:EDGE[1]?
```

This query might return

```
:MEASUREMENT:MEAS1:DELAY:EDGE[1] FALL
```

indicating that the falling edge of the waveform is being used for measurement 1.

MEASUREMENT:MEAS<x>:DELAY:EDGE2

Description

This command sets or queries the slope of the edge that is used for the delay "to" waveform when taking a delay measurement. Use the MEASUREMENT:MEAS<x>:SOURCE2 command to specify the waveform. This command is equivalent to selecting Time from the Measure menu, choosing Delay from the drop-down list and then clicking the desired Delay Edge2 setting.

Group

Measurement

Related Commands

MEASUREMENT:MEAS<x>:SOURCE2

Syntax 1

```
MEASUREMENT:MEAS<x>:DELAY:EDGE2  
{FALL|RISe}
```

Syntax 2

```
MEASUREMENT:MEAS<x>:DELAY:EDGE2?
```

Arguments

- FALL

This specifies the falling edge of the waveform.

- RISe

This specifies the rising edge of the waveform.

Example 1

```
MEASUREMENT:MEAS3:DELAY:EDGE2 RISe
```

This command specifies that the rising edge of the waveform be used for measurement 3.

Example 2

```
MEASUREMENT:MEAS1:DELAY:EDGE2?
```

This query might return

```
:MEASUREMENT:MEAS1:DELAY:EDGE2 FALL
```

indicating that the falling edge of the waveform is being used for measurement 1.

MEASUrement:MEAS<x>:MAXimum?

Description

This query-only command returns the maximum value found for this measurement since the last statistical reset. Measurements are specified by x, which ranges from 1 through 8.

Group

Measurement

Syntax

```
MEASUrement:MEAS<x>:MAXimum?
```

Example

```
MEASUrement:MEAS3:MAXimum?
```

This query might return

```
:MEASUREMENT:MEAS3:MAXIMUM 4.18E-9
```

MEASUrement:MEAS<x>:MEAN?

Description

This query-only command returns the mean value accumulated for this measurement since the last statistical reset. Measurements are specified by x, which ranges from 1 through 8.

Group

Measurement

Syntax

```
MEASUrement:MEAS<x>:MEAN?
```

Example

```
MEASUrement:MEAS1:MEAN?
```

This query might return

```
:MEASUREMENT:MEAS1:MEAN 514.71E-09
```

MEASUREMENT:MEAS<x>:METHOD

Description

This command specifies or queries the method used to calculate the 0% and 100% reference level.

Group

Measurement

Syntax 1

```
MEASUREMENT:MEAS<x>:METHOD {HISTogram|MIN-  
Max}
```

Syntax 2

```
MEASUREMENT:MEAS<x>:METHOD?
```

Arguments

- HISTogram

This sets the high and low waveform levels statistically using a histogram algorithm.

- MINMax

This sets the high and low waveform levels to MAX and MIN, respectively.

Example 1

```
MEASUREMENT:MEAS1:METHOD HISTogram
```

This command specifies that the high and low reference levels are set statistically.

Example 2

```
MEASUREMENT:MEAS1:METHOD?
```

This query might return

```
:MEASUREMENT:MEAS1:METHOD MINMAX
```

indicating that the reference levels are set to MIN and MAX.

MEASUREMENT:MEAS<x>:NOISE

Description

This command sets or queries whether the noise measurement is made on the high or low level of the waveform. Sending this command is equivalent to selecting Ref Levs > Eye > Top Level or Base Level in the Comm tab of the Measurement Setup dialog box. The Eye section is displayed only if you have an eye-pattern or optical measurement defined.

Group

Measurement

Syntax 1

```
MEASUREMENT:MEAS<x>:NOISE {HIGH|LOW}
```

Syntax 2

```
MEASUREMENT:MEAS<x>:NOISE?
```

Arguments

- HIGH

This argument causes the measurement for noise to be taken at the high level of the waveform.

- LOW

This argument causes the measurement for noise to be taken at the low level of the waveform.

Example 1

```
MEASUREMENT:MEAS1:NOISE HIGH
```

This command causes the noise measurement to be made at the high level of the waveform for Measurement 1.

Example 2

```
MEASUREMENT:MEAS2:NOISE?
```

This query might return

```
:MEASUREMENT:MEAS2:NOISE LOW
```

indicating that Measurement 2 will measure noise at the low level of the waveform.

MEASUrement:MEAS<x>:MINImum?

Description

This query-only command returns the minimum value found for this measurement since the last statistical reset. Measurements are specified by x, which ranges from 1 through 8.

Group

Measurement

Syntax

```
MEASUrement:MEAS<x>:MINImum?
```

Example

```
MEASUrement:MEAS1:MINImum?
```

This query might return

```
:MEASUREMENT:MEAS1:MINIMUM 1.75E-09
```

MEASUrement:MEAS<x>:REFLevel?

Description

This query-only command returns the current reference level parameters. It returns them in the following order: ABSOLUTE and then PERCENT for individual user measurements.

Group

Measurement

Syntax

```
MEASUrement:MEAS<x>:REFLevel?
```

Example

```
MEASUREMENT:MEAS1:REFLEVEL?
```

MEASUrement:MEAS<x>:REFLevel:ABSolute:HIGH

Description

This command sets or queries the high reference level, and is the upper reference level when MEASUrement:MEAS<x>:REFLevel:METHOD is set to Absolute. Note that this command affects the results of rise and fall measurements.

Group

Measurement

Related Commands

MEASUrement:MEAS<x>:REFLevel:METHOD

Syntax 1

```
MEASUrement:MEAS<x>:REFLevel:ABSolute:HIGH  
<NR3>
```

Syntax 2

```
MEASUrement:MEAS<x>:REFLevel:ABSolute:HIGH?  
lute:HIGH?
```

Argument

- <NR3>

This is the high reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:HIGH  
1.71
```

This command sets the high reference level to 1.71 V.

Example 2

```
MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:HIGH?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:HIGH  
1.7100E+00
```

indicating that the absolute high reference level is set to 1.71 V.

MEASUrement:MEAS<x>:REFLevel:ABSolute:LOW

Description

This command sets or queries the low reference level, and is the lower reference level when MEASUrement:MEAS<x>:REFLevel:METHOD is set to Absolute. Note that this command affects the results of rise and fall measurements.

Group

Measurement

Related Commands

MEASUrement:MEAS<x>:REFLevel:METHOD

Syntax 1

```
MEASUrement:MEAS<x>:REFLevel:ABSolute:LOW  
<NR3>
```

Syntax 2

```
MEASUrement:MEAS<x>:REFLevel:ABSolute:LOW?
```

Argument

- <NR3>

This is the low reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUrement:REFLEVEL:ABSolute:LOW 0.0 V.
```

This command sets the low reference level to 0.0 V.

Example 2

```
MEASUREMENT:MEAS1:REFLEVEL:ABSolute:LOW?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:LOW  
0.0000E+00
```

indicating that the absolute low reference level is set to 0.0 V.

MEASUREMENT:MEAS<x>:REFLEVEL:ABSOLUTE:MID[1]

Description

This command sets or queries the mid reference level, and is the 50% reference level when MEASUREMENT:MEAS<x>:REFLEVEL:METHOd is set to Absolute. Note that this command affects the results of period, frequency, delay, and all cyclic measurements.

Group

Measurement

Related Commands

MEASUREMENT:MEAS<x>REFLEVEL::METHOd

Syntax 1

```
MEASUREMENT:MEAS<x>:REFLEVEL:ABSOLUTE:MID[1] <NR3>
```

Syntax 2

```
MEASUREMENT:MEAS<x>:REFLEVEL:ABSOLUTE:MID[1]?
```

Argument

- <NR3>

This is the mid reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:MID[1].71
```

This command sets the mid reference level to .71 V.

Example 2

```
MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:MID[1]?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:MID0.7100E+00
```

indicating that the absolute mid1 reference level is set to .71 V.

MEASUREMENT:MEAS<x>:REFLevel:ABSolute:MID2

Description

This command sets or queries the mid reference level for the "to" waveform when taking a delay measurement, and is the 50% reference level when MEASUREMENT:MEAS<x>:REFLevel:METHod is set to Absolute. Note that this command affects the results of delay measurements.

Group

Measurement

Related Commands

MEASUREMENT:MEAS<x>:REFLevel:METHod

Syntax 1

```
MEASUREMENT:MEAS<x>:REFLevel:ABSolute:MID2
<NR3>
```

Syntax 2

```
MEASUREMENT:MEAS<x>:REFLevel:ABSolute:MID2?

```

Argument

- <NR3>

This is the mid reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:MID2
0.5
```

This command sets the mid reference level for the delay waveform to 0.5 V.

Example 2

```
MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:MID2?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:ABSOLUTE:MID2
0.5000E+00
```

indicating that the absolute mid2 reference level is set to 0.5 V.

MEASUREMENT:MEAS<x>:REFLEVEL:METHOd

Description

This command specifies or queries the reference level units used for measurement calculations. This command is equivalent to selecting Reference Levels from the Measure menu and then choosing the desired reference level from the Units group box.

Group

Measurement

Syntax 1

```
MEASUREMENT:MEAS<x>:REFLEVEL:METHOd {ABSolute|PERCent}
```

Syntax 2

```
MEASUREMENT:MEAS<x>:REFLEVEL:METHOd?
```

Arguments

■ ABSolute

This specifies that the reference levels are set explicitly using the MEASUREMENT:MEAS<x>:REFLEVEL:ABSolute commands. This method is useful when precise values are required. For instance, when designing to published interface specifications, such as RS-232-C.

■ PERCent

This specifies that the reference levels are calculated as a percent relative to HIGH and LOW. The percentages are defined using the MEASUREMENT:MEAS<x>:REFLEVEL:PERCent commands.

Example 1

```
MEASUREMENT:MEAS1:REFLEVEL:METHOd ABSOLUTE
```

This command specifies that explicit user-defined values are used for the reference levels.

Example 2

```
MEASUREMENT:MEAS1:REFLEVEL:METHOd?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:METHOd PERCENT
```

indicating that the reference level units used are calculated as a percent relative to HIGH and LOW.

MEASUrement:MEAS<x>:REFLevel:PERCent:HIGH

Description

This command sets or queries the percent (where 100% is equal to HIGH) that is used to calculate the high reference level when MEASUrement:MEAS<x>:REFLevel:METHOD is set to Percent. Note that this command affects the results of rise and fall measurements.

Group

Measurement

Related Commands

MEASUrement:MEAS<x>:REFLevel:METHOD

Syntax 1

```
MEASUrement:MEAS<x>:REFLevel:PERCent:HIGH  
<NR3>
```

Syntax 2

```
MEASUrement:MEAS<x>:REFLevel:PERCent:HIGH?
```

Argument

- <NR3>

This is the high reference level, ranging from 0 to 100%. The default high reference level is 90%.

Example 1

```
MEASUREMENT:MEAS1:REFLEVEL:PERCENT:HIGH 95
```

This command sets the high reference level to 95% of HIGH.

Example 2

```
MEASUREMENT:MEAS1:REFLEVEL:PERCENT:HIGH?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:PERCENT:HIGH  
90
```

indicating that the percentage high reference level is set to 90% of HIGH.

MEASUREMENT:MEAS<x>:REFLEVEL:PERCENT:LOW

Description

This command sets or queries the percent (where 100% is equal to HIGH) that is used to calculate the low reference level when MEASUREMENT:MEAS<x>:REFLEVEL:METHOD is set to Percent. Note that this command affects the results of rise and fall measurements. This command is equivalent to selecting Reference Levels from the Measure menu and then entering the Percentage Low Ref value.

Group

Measurement

Related Commands

MEASUREMENT:MEAS<x>:REFLEVEL:METHOD

Syntax 1

```
MEASUREMENT:MEAS<x>:REFLEVEL:PERCENT:LOW  
<NR3>
```

Syntax 2

```
MEASUREMENT:MEAS<x>:REFLEVEL:PERCENT:LOW?
```

Argument

- <NR3>

This is the low reference level, ranging from 0 to 100%. The default low reference level is 10%.

Example 1

```
MEASUREMENT:MEAS<x>:REFLEVEL:PERCENT:LOW  
15
```

This command sets the high reference level to 15% of HIGH.

Example 2

```
MEASUREMENT:MEAS<x>:REFLEVEL:PERCENT:LOW?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:PERCENT:LOW 10
```

indicating that the percentage high reference level is set to 10% of HIGH.

MEASUREMENT:MEAS<x>:REFLEVEL:PERCENT:MID[1]

Description

This command sets or queries the percent (where 100% is equal to HIGH) that is used to calculate the mid reference level when MEASUREMENT:MEAS<x>:REFLEVEL:METHOd is set to Percent. Note that this command affects the results of period, frequency, delay, and all cyclic measurements.

Group

Measurement

Related Commands

MEASUREMENT:MEAS<x>:REFLEVEL:METHOd

Syntax 1

```
MEASUREMENT:MEAS<x>:REFLEVEL:PERCENT:MID[1] <NR3>
```

Syntax 2

```
MEASUREMENT:MEAS<x>:REFLEVEL:PERCENT:MID[1]?
```

Argument

- <NR3>

This is the mid reference level, ranging from 0 to 100%. The default mid reference level is 50%.

Example 1

```
MEASUREMENT:MEAS1:REFLEVEL:PERCENT:MID[1] 60
```

This command sets the mid reference level to 60% of HIGH.

Example 2

```
MEASUREMENT:MEAS1:REFLEVEL:PERCENT:MID[1]?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:PERCENT:MID 65
```

indicating that the percentage mid reference level is set to 65% of HIGH.

MEASUrement:MEAS<x>:REFLevel:PERCent:MiD2

Description

This command sets or queries the percent (where 100% is equal to HIGH) that is used to calculate the mid reference level for the second waveform specified when MEASUrement:MEAS<x>:REFLevel:MEthod is set to Percent. Note that this command affects the results of delay measurements.

Group

Measurement

Related Commands

MEASUrement:MEAS<x>:REFLevel:MEthod

Syntax 1

```
MEASUrement:MEAS<x>:REFLevel:PERCent:MiD2
<NR3>
```

Syntax 2

```
MEASUrement:MEAS<x>:REFLevel:PERCent:MiD2?
```

Argument

- <NR3>

This is the mid reference level, ranging from 0 to 100%. The default mid reference level is 50%.

Example 1

```
MEASUREMENT:MEAS1:REFLEVEL:PERCENT:MiD2 40
```

This command sets the mid2 reference level to 40% of HIGH.

Example 2

```
MEASUREMENT:MEAS1:REFLEVEL:PERCENT:MiD2?
```

This query might return

```
:MEASUREMENT:MEAS1:REFLEVEL:PERCENT:MiD2
45
```

indicating that the percentage mid2 reference level is set to 45% of HIGH.

MEASUrement:MEAS<x>:SOURCE[1]

Description

This command sets or queries the source for all single channel measurements and specifies the source to measure "from" when taking a delay measurement or phase measurement. Measurements are specified by x, which ranges from 1 through 8. This command is equivalent to selecting Measurement Setup from the Measure menu and then choosing the desired measurement source.

Group

Measurement

Syntax 1

```
MEASUrement:MEAS<x>:SOURCE[1]  
{CH<x>|MATH<y>|REF<x>|HISTogram}
```

Syntax 2

```
MEASUrement:MEAS<x>:SOURCE[1]?
```

Arguments

- CH<x>

This is an input channel. The valid channel waveform range is from 1 through 4.

- MATH<y>

This is a math waveform. The valid math waveform range is from 1 through 4.

- REF<x>

This is a reference waveform. The valid reference waveform range is from 1 through 4.

- HISTogram

This is a histogram.

MEASUrement:MEAS<x>:SOURCE2

Description

This command sets or queries the source to measure "to" for phase or delay measurements. Measurements are specified by x, which ranges from 1 through 8. This command is equivalent to selecting Measurement Setup from the Measure menu, selecting a measurement type of either Phase or Delay, and then choosing the desired measurement source.

Tip: Source2 measurements only apply to phase and delay measurement types, which require both a target (Source1) and reference (Source2) source.

Group

Measurement

Related Commands

MEASUrement:MEAS<x>:TYPE

Syntax 1

```
MEASUrement:MEAS<x>:SOURCE2  
{CH<x>|MATH<y>|REF<x>}
```

Syntax 2

```
MEASUrement:MEAS<x>:SOURCE2?
```

Arguments

- CH<x>

This is an input channel. The valid channel waveform range is from 1 through 4.

- MATH<y>

This is a math waveform. The valid math waveform range is from 1 through 4.

- REF<x>

This is a reference waveform. The valid reference waveform range is from 1 through 4.

Example 1

```
MEASUrement:MEAS4:SOURCE2 CH1
```

This command specifies CH1 as the phase or delay "to" source when making a phase or delay measurement.

Example 2

`MEASUREMENT:MEAS2:SOURCE2?`

This query might return

`:MEASUREMENT:MEAS2:SOURCE2 MATH1`

indicating that Math 1 is the measurement 2 source.

MEASUrement:MEAS<x>:STATE

Description

This command sets or queries whether the specified measurement slot is computed and displayed. The measurement slot is specified by x, which ranges from 1 through 8. This command is equivalent to selecting Measurement Setup from the Measure menu and then clicking the Display button.

For a measurement to display, you must have selected a source waveform and defined the measurement you want to take and display. You select the measurement using the MEASUrement:MEAS<x>:SOURCE[1] command. You define the measurement type using the MEASUrement:MEAS<x>:TYPE command.

Group

Measurement

Related Commands

MEASUrement:MEAS<x>:SOURCE[1], MEASUrement:MEAS<x>:TYPE

Syntax 1

```
MEASUrement:MEAS<x>:STATE {OFF|ON|<NR1>}
```

Syntax 2

```
MEASUrement:MEAS<x>:STATE?
```

Arguments

- OFF

This disables calculation and display of the specified measurement slot.

- ON

This enables calculation and display of the specified measurement slot.

- <NR1>

A 0 disables calculation and display of the specified measurement slot; any other value enables calculation and display of the specified measurement slot.

Example 1

`MEASUrement:MEAS2:STATE ON`

This command computes and displays the measurement defined as measurement 2.

Example 2

`MEASUrement:MEAS1:STATE?`

This query might return

`:MEASUREMENT:MEAS1:STATE 0`

indicating that measurement defined for measurement slot 1 is disabled.

MEASUrement:MEAS<x>:STDdev?

Description

This query-only command returns the standard deviation of values accumulated for this measurement since the last statistical reset. Measurements are specified by x, which ranges from 1 through 8.

Group

Measurement

Syntax

```
MEASUrement:MEAS<x>:STDdev?
```

Example

```
MEASUrement:MEAS1:STDdev?
```

This query might return

```
:MEASUREMENT:MEAS1:STDDEV 21.0E-12
```

MEASUREMENT:MEAS<x>:TYPE

Description

This command sets or queries the type of measurement that is defined for the specified measurement slot. The measurement slot is specified by x, which ranges from 1 through 8. This command is equivalent to selecting Measurement Setup from the Measure menu and then choosing the desired measurement type.

Group

Measurement

Syntax 1

```
MEASUREMENT:MEAS<x>:TYPE {AMPLITUDE|AREa|BURst|CA-  
Rea|CMEan|CRMs|DELay|DISTDUty|EX-  
TINCTDB|EXTINCTPCT|  
EXTINCTRATIO|EYEHEight|EYEWIdth|FALL|FRE-  
Quency|HIGH|HITS|LOW|MAXimum|MEAN| ME-  
Dian|MINimum|NCROSS|NDUty|NOVer-  
shoot|NwIdth|PBASE|PCROSS|PCTCROSS|PDUty|  
PEAKHits|PERIOD|PHase|PK2Pk|PKPKJit-  
ter|PKPKNoise|POVershoot|PTOP|PWidth|  
QFACTOR|RISe|RMS|RMSJitter|RMSNoise|SIGA-  
MA1|SIGMA2|SIGMA3|SIXSignmajit| SNRa-  
tio|STDdev|UNDEFINED|WAVEFORMS}
```

Syntax 2

```
MEASUREMENT:MEAS<x>:TYPE?
```

Arguments

■ AMPLitude

This measures the amplitude of the selected waveform. In other words, it measures the high value less the low value measured over the entire waveform or gated region.

Amplitude = High - Low

■ AREa

This measures the voltage over time. The area is over the entire waveform or gated region and is measured in volt-seconds. The area measured above the ground is positive, while the area below ground is negative.

■ BURst

The burst width measures the duration of a burst. The measurement is made over the entire waveform or gated region.

- **CARea**

The cycle area measures the voltage over time. In other words, it measures, in volt-seconds, the area over the first cycle in the waveform or the first cycle in the gated region. The area measured above the common reference point is positive, while the area below the common reference point is negative.

- **CMEan**

This measures the arithmetic mean over the first cycle in the waveform or the first cycle in the gated region.

- **CRMs**

This measures the true Root Mean Square voltage over the first cycle in the waveform or the first cycle in the gated region.

- **DELAy**

This measures the time between the middle reference (default = 50%) amplitude point of the source waveform and the destination waveform.

- **DISTDUty**

The duty cycle distortion measures the time between the falling edge and the rising edge of the eye pattern at the mid reference level.

It is the peak-to-peak time variation of the first eye crossing measured at the mid-reference as a percent of the eye period.

- **EXTINCTDB**

This (optical waveform (eye diagram) measurement) measures its extinction ratio. Extinction Ratio (dB) measures the ratio of the average power levels for the logic High to the logic Low of an optical waveform and expresses the result in dB.

This measurement only works for fast acquisition signals or a reference waveform saved in fast acquisition mode.

Extinction dB = 10 X (log 10 (High / Low))

- **EXTINCTPCT**

This (optical measurement) measures the extinction ratio of the selected optical waveform. Extinction Ratio (%) measures the ratio of the average power levels for the logic Low (off) to the logic (High) (on) of an optical waveform and expresses the result in percent.

This measurement only works for fast acquisition signals or a reference waveform saved in fast acquisition mode.

Extinction % = 100.0 x (Low / High)

■ **EXTINCTRATIO**

This (optical measurement) measures the extinction ratio of the selected optical waveform. Extinction Ratio measures the ratio of the average power levels for the logic High to the logic Low of an optical waveform and expresses the result without units.

This measurement only works for fast acquisition signals or a reference waveform saved in fast acquisition mode.

Extinction Ratio = (High / Low), ratios greater than 100 or less than 1 generate errors; low must be greater than or equal to 1 microwatt.

■ **EYEHeight**

This measures the vertical opening of an eye diagram, in volts.

■ **EYEWidth**

This measures the width of an eye diagram, in seconds.

■ **FALL**

This measures the time taken for the falling edge of the first pulse in the waveform or gated region to fall from a high reference value (default is 90%) to a low reference value (default is 10%).

■ **FREQuency**

This measures the first cycle in the waveform or gated region. Frequency is the reciprocal of the period and is measured in hertz (Hz), where 1 Hz = 1 cycle per second.

■ **HIGH**

This measures the High reference (100% level, sometimes called Topline) of a waveform.

You can also limit the High measurement (normally taken over the entire waveform record) to a gated region on the waveform.

■ **HITS**

The hits-in-box histogram measurement measures the number of points in or on the histogram box.

■ **LOW**

This measures the Low reference (0% level, sometimes called Baseline) of a waveform.

■ **MAXimum**

This finds the maximum amplitude. This value is the most positive peak voltage found. It is measured over the entire waveform or gated region.

When histogram is selected with the measurement:method command, the maximum measurement measures the voltage of the highest nonzero bin in vertical histograms or the time of the rightmost bin in horizontal histograms.

■ **MEAN**

This amplitude measurement finds the arithmetic mean over the entire waveform or gated region.

When histogram is selected with the measurement:method command, the mean measurement measures the average of all acquired points within or on the histogram.

■ **MEDian**

This (histogram measurement) measures the middle point of the histogram box. Half of all acquired points within or on the histogram box are less than this value and half are greater than this value.

■ **MINImum**

This amplitude measurement finds the minimum amplitude. This value is typically the most negative peak voltage. It is measured over the entire waveform or gated region.

When histogram is selected with the measurement:method command, the minimum measurement measures the lowest nonzero bin in vertical histograms or the time of the leftmost nonzero bin in the horizontal histograms.

■ **NCROss**

This (timing measurement) measures the time from the trigger point to the first falling edge of the waveform or gated region. The distance (time) is measured at the middle reference amplitude point of the signal.

■ **NDUty**

The negative duty cycle is the ratio of the negative pulse width to the signal period, expressed as a percentage. The duty cycle is measured on the first cycle in the waveform or gated region.

$$\text{Negative Duty Cycle} = (\text{Negative Width}) / \text{Period} * 100\%$$

■ **NOVershoot**

The negative overshoot amplitude measurement finds the negative overshoot value over the entire waveform or gated region.

$$\text{Negative Overshoot} = (\text{Low} - \text{Minimum}) / \text{Amplitude} * 100\%$$

- **NWidth**

The negative width measurement is the distance (time) between the middle reference (default = 50%) amplitude points of a negative pulse. The measurement is made on the first pulse in the waveform or gated region.

- **PBAsE**

This measures the base value used in extinction ratio measurements.

- **PCROss**

This (timing measurement) measures the time from the trigger point to the first positive edge of the waveform or gated region. The distance (time) is measured at the middle reference amplitude point of the signal.

- **PCTCROss**

The crossing percent (optical measurement) measures the location of the eye crossing point expressed as a percentage of EYEHeight.

*Crossing percent = 100 * [(eye-crossing-point – PBAsE)/(PTOP – PBAsE)]*

- **PDUty**

The positive duty cycle is the ratio of the positive pulse width to the signal period, expressed as a percentage. It is measured on the first cycle in the waveform or gated region.

*Positive Duty Cycle = (Positive Width)/Period * 100%*

- **PEAKHits**

This (histogram measurement) measures the number of points in the largest bin of the histogram.

- **PERIod**

This timing measurement is the time required to complete the first cycle in a waveform or gated region. Period is the reciprocal of frequency and is measured in seconds.

- **PHAsE**

This measures the phase difference (amount of time a waveform leads or lags the reference waveform) between two waveforms. The measurement is made between the middle reference points of the two waveforms and is expressed in degrees, where 360° represents one waveform cycle.

- **PK2Pk**

This peak-to-peak amplitude measurement finds the absolute difference between the maximum and minimum amplitude in the entire waveform or gated region.

When histogram is selected with the measurement:method command, the PK2Pk measurement measures the histogram's peak to peak difference.

- **PKPKJitter**

This measures the variance (minimum and maximum values) in the time locations of the cross point.

- **PKPKNoise**

This measures the peak-to-peak noise on a waveform at the mid reference level.

- **POvershoot**

The positive overshoot amplitude measurement finds the positive overshoot value over the entire waveform or gated region.

*Positive Overshoot = (Maximum – High) / Amplitude * 100%*

- **PTOP**

This measures the top value used in extinction ratio measurements.

- **PWidth**

The positive width is the distance (time) between the middle reference (default = 50%) amplitude points of a positive pulse. The measurement is made on the first pulse in the waveform or gated region.

- **QFACTOR**

This measures the quality factor. The Q factor is a figure of merit for an eye diagram, which indicates the vertical eye opening relative to the noise at the low and high logic levels. It is the ratio of the eye size to noise.

- **RISe**

This timing measurement finds the rise time of the waveform. The rise time is the time it takes for the leading edge of the first pulse encountered to rise from a low reference value (default is 10%) to a high reference value (default is 90%).

Rise time = TimeRef High - TimeRef Low

You can also limit the measurement (normally taken over the entire waveform record) to a gated region on the waveform.

- **RMS**

This amplitude measurement finds the true Root Mean Square voltage in the entire waveform or gated region.

- **RMSJitter**

This measures the variance in the time locations of the cross point. The RMS jitter is defined as one standard deviation at the cross point.

- **RMSNoise**

This measures the Root Mean Square noise amplitude on a waveform at the mid reference level.

- **SIGMA1**

This (histogram measurement) measures the percentage of points in the histogram that are within one standard deviation of the histogram mean.

- **SIGMA2**

This (histogram measurement) measures the percentage of points in the histogram that are within two standard deviations of the histogram mean.

- **SIGMA3**

This (histogram measurement) measures the percentage of points in the histogram that are within three standard deviations of the histogram mean.

- **SIXSigmajit**

This (histogram measurement) is six * RMSJitter.

- **SNRatio**

This measures the signal-to-noise ratio. The signal-to-noise ratio is the amplitude of a noise rejection band centered on the mid level.

- **STDdev**

This measures the standard deviation (Root Mean Square (RMS) deviation) of all acquired points within or on the histogram box.

- **UNDEFINED**

This is the default measurement type, which indicates that no measurement type is specified. Once a measurement type is chosen, it can be cleared using this argument.

- **WAVEFORMS**

The waveform count (histogram measurement) measures the number of waveforms used to calculate the histogram.

Example 1

`MEASUREMENT:MEAS2:TYPE FREQUENCY`

This command defines measurement 2 as a measurement of the frequency of a waveform.

Example 2

`MEASUREMENT:MEAS1:TYPE?`

This query might return

`:MEASUREMENT:MEAS1:TYPE RMS`

indicating that measurement 1 is defined to measure the RMS value of a waveform.

MEASUREMENT:MEAS<x>:UNITS?

Description

This query-only command returns the units associated with the specified measurement.

Group

Measurement

Related Commands

MEASUREMENT:MEAS<x>:TYPE

Syntax

MEASUREMENT:MEAS<x>:UNITS?

Returns (partial list)

- dB

Decibels. Occurs with the communications-extinction ratio measurement.

- hits

Hits. Occurs with histogram measurements, such as hits in box.

- Hz

Hertz

- S

Seconds

- V

Volts

- Vs

Volt seconds. Occurs in area measurements.

- VV

Possible in a case like $math1="ch1*ch2"$

- W

Watts

- wfms

Waveforms. Occurs with the histogram-waveform count measurement.

- %

Percent

- "\b0"

This displays as the degrees' symbol. It occurs with phase measurements.

- ?

Unknown. Possible in a case like $math1="ch1+ch2"$ if you try adding a volts measurement with a watts measurement

Example

```
MEASUREMENT:MEAS1:UNITs?
```

This query might return

```
:MEASUREMENT:MEAS1:UNITs %
```

indicating units for measurement 1 are set to percent.

MEASUrement:MEAS<x>:VALue?

Description

This query-only command returns the value that has been calculated for the measurement specified by x, which ranges from 1 through 8. This command is equivalent to selecting Display Statistics from the Measure menu and then choosing Value from the drop-down list to display *all* measurement values on-screen.

Note: This is the displayed value in the onscreen display. If measurement statistics are enabled, a new value is calculated with every waveform. In addition, this value is updated about every 1/3 second. If you are acquiring a long acquisition record, the oscilloscope may take longer to update.

Group

Measurement

Related Commands

MEASUrement:MEAS<x>:UNIts?, *ESR?, ALLEV?

Syntax

MEASUrement:MEAS<x>:VALue?

Example

MEASUrement:MEAS1:VALue?

This query might return

```
:MEASUREMENT:MEAS1:VALue 2.8740E-06
```

If the measurement has an error or warning associated with it, then an item is added to the error queue. The error can be checked for with the *ESR? and ALLEV? commands.

MEASUrement:METhod

Description

This command sets or queries the method used to calculate the 0% and 100% reference level. With per-measurement attributes, this is an obsolete command. The command form changes the IMMED and all MEAS<x> versions. The query returns the current values of the IMMED version only.

Group

Measurement

Related Commands

MEASUrement:REFLevel:PERCent:HIGH, MEASUrement:REFLevel:PERCent:LOW, MEASUrement:REFLevel:PERCent:MID, MEASUrement:REFLevel:PERCent:MID2

Syntax 1

MEASUrement:METhod {HISTogram|MEAN|MINMax}

Syntax 2

MEASUrement:METhod?

Arguments

- HISTogram

This sets the high and low reference levels to the most common values either above or below the mid point, depending on whether the high reference point or the low reference point is being defined. Because the statistical approach ignores short-term aberrations, such as overshoot or ringing, the histogram method is the best setting for examining pulses.

- MEAN

This sets the high and low reference levels to the mean values using all values either above or below the mid-point, depending of whether it is defining the high or low reference level. The selection is best used for examining eye patterns.

- MINMax

This uses the highest and lowest values of the waveform record. This selection is best for examining waveforms with no large, flat portions of a common value, such as sine waves and triangle waves.

Example 1

`MEASUREMENT:METHOD HISTogram`

This command specifies that the high and low reference levels are set statistically.

Example 2

`MEASUREMENT:METHOD?`

This query might return

`:MEASUREMENT:METHOD MINMAX`

Indicating that the reference levels are set to MIN and MAX.

MEASUREMENT:NOISE

Description

This command sets or queries whether the noise measurement is made on the high or low level of the waveform. Sending this command is equivalent to selecting Ref Levs > Eye > Top Level or Base Level in the Comm tab of the Measurement Setup dialog box. The Eye section is displayed only if you have an eye-pattern or optical measurement defined.

Group

Measurement

Syntax 1

```
MEASUREMENT:NOISE {HIGH|LOW}
```

Syntax 2

```
MEASUREMENT:NOISE?
```

Arguments

- HIGH

This argument causes the measurement for noise to be taken at the high level of the waveform.

- LOW

This argument causes the measurement for noise to be taken at the low level of the waveform.

Example 1

```
MEASUREMENT:NOISE HIGH
```

This command causes the noise measurement to be made at the high level of the waveform.

Example 2

```
MEASUREMENT:NOISE?
```

This query might return

```
:MEASUREMENT:NOISE LOW
```

indicating that the noise measurement will be made at the low level of the waveform.

MEASUREMENT:REFLEVEL:ABSOLUTE:HIGH

Description

This command sets or queries the high reference level, and is the upper reference level when MEASUREMENT:REFLEVEL:METHOD is set to Absolute. Note that this command affects the results of rise and fall measurements. With per-measurement attributes, this is an obsolete command. The command form changes the IMMEDIATE and all MEAS<x> versions. The query returns the current values of the IMMEDIATE version only.

Group

Measurement

Related Commands

MEASUREMENT:REFLEVEL:METHOD, MEASUREMENT:IMMEDIATE:TYPE, MEASUREMENT:MEAS<x>:TYPE

Syntax 1

```
MEASUREMENT:REFLEVEL:ABSOLUTE:HIGH <NR3>
```

Syntax 2

```
MEASUREMENT:REFLEVEL:ABSOLUTE:HIGH?
```

Argument

- <NR3>

This is the high reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:REFLEVEL:ABSOLUTE:HIGH 1.71
```

This command sets the high reference level to 1.71 V.

Example 2

```
MEASUREMENT:REFLEVEL:ABSOLUTE:HIGH?
```

This query might return

```
:MEASUREMENT:REFLEVEL:ABSOLUTE:HIGH  
1.7100E+00
```

Indicating that the absolute high reference level is set to 1.71 V.

MEASUREMENT:REFLEVEL:ABSOLUTE:LOW

Description

This command sets or queries the low reference level, and is the lower reference level when MEASUREMENT:REFLEVEL:METHOD is set to Absolute. Note that this command affects the results of rise and fall measurements. With per-measurement attributes, this is an obsolete command. The command form changes the IMMEDIATE and all MEAS<x> versions. The query returns the current values of the IMMEDIATE version only.

Group

Measurement

Related Commands

MEASUREMENT:REFLEVEL:METHOD, MEASUREMENT:IMMEDIATE:TYPE, MEASUREMENT:MEAS<x>:TYPE

Syntax 1

```
MEASUREMENT:REFLEVEL:ABSOLUTE:LOW <NR3>
```

Syntax 2

```
MEASUREMENT:REFLEVEL:ABSOLUTE:LOW?
```

Argument

- <NR3>

This is the low reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:REFLEVEL:ABSOLUTE:LOW 0.0 V.
```

This command sets the low reference level to 0.0 V.

Example 2

```
MEASUREMENT:REFLEVEL:ABSOLUTE:LOW?
```

This query might return

```
:MEASUREMENT:REFLEVEL:ABSOLUTE:LOW  
0.0000E+00
```

Indicating that the absolute low reference level is set to 0.0 V.

MEASUrement:REFLevel:ABSolute:MID

Description

This command sets or queries the mid reference level, and is the 50% reference level when MEASUrement:REFLevel:METHOD is set to Absolute. Note that this command affects the results of period, frequency, delay, and all cyclic measurements. With per-measurement attributes, this is an obsolete command. The command form changes the IMMED and all MEAS<x> versions. The query returns the current values of the IMMED version only.

Group

Measurement

Related Commands

MEASUrement:REFLevel:METHOD

Syntax 1

```
MEASUrement:REFLevel:ABSolute:MID[1] <NR3>
```

Syntax 2

```
MEASUrement:REFLevel:ABSolute:MID[1]?
```

Argument

- <NR3>

This is the mid reference level, in volts. The default is 0.0 V.

Example 1

```
MEASUREMENT:REFLEVEL:ABSOLUTE:MID[1] .71
```

This command sets the mid reference level to .71 V.

Example 2

```
MEASUREMENT:REFLEVEL:ABSOLUTE:MID[1]?
```

This query might return

```
:MEASUREMENT:REFLEVEL:ABSOLUTE:MID  
0.7100E+00
```

indicating that the absolute mid1 reference level is set to .71 V.

MEASUrement:REFLevel:ABSolute:MID2

Description

This command sets or queries the mid reference level for the "to" waveform when taking a delay measurement, and is the 50% reference level when MEASUrement:REFLevel:METHOD is set to Absolute. Note that this command affects the results of delay measurements. With per-measurement attributes, this is an obsolete command. The command form changes the IMMED and all MEAS<x> versions. The query returns the current values of the IMMED version only.

Group

Measurement

Related Commands

MEASUrement:REFLevel:METHOD

Syntax 1

```
MEASUrement:REFLevel:ABSolute:MID2 <NR3>
```

Syntax 2

```
MEASUrement:REFLevel:ABSolute:MID2?
```

Argument

- <NR3>

This is the mid reference level. The default is 0.0 V.

Example 1

```
MEASUREMENT:REFLEVEL:ABSOLUTE:MID2 0.5
```

This command sets the mid reference level for the delay waveform to 0.5 V.

Example 2

```
MEASUREMENT:REFLEVEL:ABSOLUTE:MID2?
```

This query might return

```
:MEASUREMENT:REFLEVEL:ABSOLUTE:MID2  
0.5000E+00
```

indicating that the absolute mid2 reference level is set to 0.5 V.

MEASUREMENT:REFLEVEL:METHOD

Description

This command specifies or queries the reference level units used for measurement calculations. With per-measurement attributes, this is an obsolete command. The command form changes the IMMEDIATE and all MEASUREMENT versions. The query returns the current values of the IMMEDIATE version only.

Group

Measurement

Syntax 1

```
MEASUREMENT:REFLEVEL:METHOD {ABSOLUTE|PERCENT}
```

Syntax 2

```
MEASUREMENT:REFLEVEL:METHOD?
```

Arguments

■ ABSOLUTE

This specifies that the reference levels are set explicitly using the MEASUREMENT:REFLEVEL:ABSOLUTE commands. This method is useful when precise values are required. For instance, when designing to published interface specifications, such as RS-232-C.

■ PERCENT

This specifies that the reference levels are calculated as a percent relative to HIGH and LOW. The percentages are defined using the MEASUREMENT:REFLEVEL:PERCENT commands.

Example 1

```
MEASUREMENT:REFLEVEL:METHOD ABSOLUTE
```

This command specifies that explicit user-defined values are used for the reference levels.

Example 2

```
MEASUREMENT:REFLEVEL:METHOD?
```

This query might return

```
:MEASUREMENT:REFLEVEL:METHOD PERCENT
```

indicating that the reference level units used are calculated as a percent relative to HIGH and LOW.

MEASUrement:REFLevel:PERCent:HIGH

Description

This command sets or queries the percent (where 100% is equal to HIGH) that is used to calculate the high reference level when MEASUrement:REFLevel:METHOD is set to Percent. Note that this command affects the results of rise and fall measurements. With per-measurement attributes, this is an obsolete command. The command form changes the IMMED and all MEAS<x> versions. The query returns the current values of the IMMED version only.

Group

Measurement

Related Commands

MEASUrement:REFLevel:METHOD, MEASUrement:IMMed:TYPE, MEASUrement:MEAS<x>:TYPE

Syntax 1

```
MEASUrement:REFLevel:PERCent:HIGH <NR3>
```

Syntax 2

```
MEASUrement:REFLevel:PERCent:HIGH?
```

Argument

- <NR3>

This is the high reference level, ranging from 0 to 100%. The default high reference level is 90%.

Example 1

```
MEASUREMENT:REFLEVEL:PERCENT:HIGH 95
```

This command sets the high reference level to 95% of HIGH.

Example 2

```
MEASUREMENT:REFLEVEL:PERCENT:HIGH?
```

This query might return

```
:MEASUREMENT:REFLEVEL:PERCENT:HIGH 90
```

indicating that the percentage high reference level is set to 90% of HIGH.

MEASUrement:REFLevel:PERCent:LOW

Description

This command sets or queries the percent (where 100% is equal to HIGH) that is used to calculate the low reference level when MEASUrement:REFLevel:METHOD is set to Percent. Note that this command affects the results of rise and fall measurements. With per-measurement attributes, this is an obsolete command. The command form changes the IMMED and all MEAS<x> versions. The query returns the current values of the IMMED version only.

Group

Measurement

Related Commands

MEASUrement:REFLevel:METHOD, MEASUrement:IMMed:TYPE, MEASUrement:MEAS<x>:TYPE

Syntax 1

```
MEASUrement:REFLevel:PERCent:LOW <NR3>
```

Syntax 2

```
MEASUrement:REFLevel:PERCent:LOW?
```

Argument

- <NR3>

This is the low reference level, ranging from 0 to 100%. The default low reference level is 10%.

Example 1

```
MEASUREMENT:REFLEVEL:PERCENT:LOW 15
```

This command sets the high reference level to 15% of HIGH.

Example 2

```
MEASUREMENT:REFLEVEL:PERCENT:LOW?
```

This query might return

```
:MEASUREMENT:REFLEVEL:PERCENT:LOW 10
```

indicating that the percentage high reference level is set to 10% of HIGH.

MEASUrement:REFLevel:PERCent:MID[1]

Description

This command sets or queries the percent (where 100% is equal to HIGH) that is used to calculate the mid reference level when MEASUrement:REFLevel:METHOD is set to Percent. Note that this command affects the results of period, frequency, delay, and all cyclic measurements. With per-measurement attributes, this is an obsolete command. The command form changes the IMMED and all MEAS<x> versions. The query returns the current values of the IMMED version only.

Group

Measurement

Related Commands

MEASUrement:REFLevel:METHOD

Syntax 1

```
MEASUrement:REFLevel:PERCent:MID[1] <NR3>
```

Syntax 2

```
MEASUrement:REFLevel:PERCent:MID[1]?
```

Argument

- <NR3>

This is the mid reference level, ranging from 0 to 100%. The default mid reference level is 50%.

Example 1

```
MEASUREMENT:REFLEVEL:PERCENT:MID[1] 60
```

This command sets the mid reference level to 60% of HIGH.

Example 2

```
MEASUREMENT:REFLEVEL:PERCENT:MID[1]?
```

This query might return

```
:MEASUREMENT:REFLEVEL:PERCENT:MID 65
```

indicating that the percentage mid reference level is set to 65% of HIGH.

MEASUrement:REFLevel:PERCent:MiD2

Description

This command sets or queries the percent (where 100% is equal to HIGH) that is used to calculate the mid reference level for the second waveform specified when MEASUrement:REFLevel:METhod is set to Percent. Note that this command affects the results of delay measurements. With per-measurement attributes, this is an obsolete command. The command form changes the IMMED and all MEAS<x> versions. The query returns the current values of the IMMED version only.

Group

Measurement

Related Commands

MEASUrement:REFLevel:METhod

Syntax 1

```
MEASUrement:REFLevel:PERCent:MiD2 <NR3>
```

Syntax 2

```
MEASUrement:REFLevel:PERCent:MiD2?
```

Argument

- <NR3>

This is the mid reference level, ranging from 0 to 100%. The default mid reference level is 50%.

Example 1

```
MEASUREMENT:REFLEVEL:PERCENT:MiD2 40
```

This command sets the mid2 reference level to 40% of HIGH.

Example 2

```
MEASUREMENT:REFLEVEL:PERCENT:MiD2?
```

This query might return

```
:MEASUREMENT:REFLEVEL:PERCENT:MiD2 45
```

indicating that the percentage mid2 reference level is set to 45% of HIGH.

MEASUrement:STATIstics:COUNT

Description

This command (no query form) clears existing measurement statistics from memory. This command is equivalent to selecting Reset Statistics from the Measure menu.

Group

Measurement

Related Commands

MEASUrement:STATIstics:MODE

Syntax

```
MEASUrement:STATIstics:COUNT {RESET}
```

Argument

- RESET

This clears existing measurement statistics from memory.

MEASUREMENT:STATISTICS:MODE

Description

This command controls the operation and display of management statistics. This command is equivalent to selecting Measurement Setup from the Measure menu, clicking the Statistics button and then choosing the desired Measurement Format.

Group

Measurement

Related Commands

MEASUREMENT:STATISTICS:WEIGHTING

Syntax 1

```
MEASUREMENT:STATISTICS:MODE {OFF|ALL|VALUEMean}
```

Syntax 2

```
MEASUREMENT:STATISTICS:MODE?
```

Arguments

- OFF

This turns measurements off. This is the default value.

- ALL

This turns on statistics and displays all statistics for each measurement.

- VALUEMean

This turns on statistics and displays the value and the mean (μ) of each measurement.

Example 1

```
MEASUREMENT:STATISTICS:MODE OFF
```

This command turns statistic measurements off.

Example 2

```
MEASUREMENT:STATISTICS:MODE?
```

This query might return

```
:MEASUREMENT:STATISTICS:MODE ALL
```

indicating that measurement statistics are turned on and all statistics are being displayed for each measurement.

MEASUREMENT:STATISTICS:WEIGHTING

Description

This command sets or queries the 'time constant' for mean and standard deviation statistical accumulations. This command is equivalent to selecting Measurement Setup from the Measure menu, clicking the Statistics button and then entering the desired Weight n= value.

Group

Measurement

Related Commands

MEASUREMENT:STATISTICS:MODE

Syntax 1

```
MEASUREMENT:STATISTICS:WEIGHTING <NR1>
```

Syntax 2

```
MEASUREMENT:STATISTICS:WEIGHTING?
```

Argument

- <NR1>

This is the time constant for the mean and standard deviation statistical accumulations.

Example 1

```
MEASUREMENT:STATISTICS:WEIGHTING 4
```

This command sets statistical weighting to 4.

Example 2

```
MEASUREMENT:STATISTICS:WEIGHTING?
```

This query might return

```
:MEASUREMENT:STATISTICS:WEIGHTING 4
```

indicating that measurement statistics weighting is currently set to 4.

NEWpass

Description

This command (no query form) changes the password that enables access to password protected data. The PASS-Word command must be successfully executed before using this command or an execution error will be generated.

Group

Miscellaneous

Related Commands

PASSWord, *PUD

Syntax

```
NEWpass <QString>
```

Argument

- <Qstring>

This is the new password, which can contain up to 10 characters.

Example

```
NEWpass "mypassword"
```

This command creates a new password for accessing your protected data.

PASSWord

Description

This command (no query form) enables the *PUD and NEWpass set commands. Sending PASSWord without any arguments disables these same commands. Once the password is successfully entered, the *PUD and NEWpass commands are enabled until the oscilloscope is powered off, or until the FACTory command, the PASSWord command with no arguments, or the *RST command is issued.

To change the password, you must first enter the valid password with the PASSWord command and then change to your new password with the NEWpass command. Remember that the password is case sensitive.

Group

Miscellaneous

Related Commands

NEWpass, *PUD

Syntax

```
PASSWord <QString>
```

Argument

- <QString>

This is the password, which can contain up to 10 characters. The factory default password is "XYZZY" and is always valid.

Example 1

```
PASSWord "XYZZY"
```

This command enables the *PUD and NEWPass set commands.

Example 2

```
PASSWord
```

This command disables the *PUD and NEWPass set commands. You can still use the query version of *PUD.

RECALL:MASK

Description

This command (no query form) recalls a stored mask as the User mask.

Group

Mask

Related Commands

SAVE:MASK

Syntax

```
RECALL:MASK <Qstring>
```

Arguments

- <Qstring>

Is a quoted string that defines the file name and path from where the mask will be recalled. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. If you do not specify them, the instrument will read the mask from the default directory. <filename> stands for a filename of up to 128 characters (use of wildcard characters in filenames is not supported). Filename extensions are not required but are highly recommended.

The file is stored in ASCII format and is user-editable.

Example

```
RECALL:MASK "TEST"
```

This command recalls the mask stored in the file named TEST as the User mask.

RECALL:SETUp

Description

This command (no query form) restores a stored or factory front-panel setup of the oscilloscope from a copy of the settings stored in memory. If 'factory' is referenced (by specifying '0'), the factory default values will be restored. This command is equivalent to *RCL and performs the same function as selecting Instrument Setup from the File menu and then choosing the Recall Setups tab.

Group

Save and Recall

Related Commands

FACTory, *RCL, *RST, *SAV, SAVE:SETUp

Syntax

```
RECALL:SETUp {FACTory|<NR1>|<file path>}
```

Arguments

- FACTory

This specifies to restore the factory setup.

- <NR1>

This specifies a setup storage location to restore. Setup storage location values range from 0 through 10; Using an out of range value causes an error (222, "Data out of range").

- <file path>

This is the location from where the setup will be recalled.

<file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. If you do not specify them, the instrument will read the file from the default directory. <file-name> stands for a filename of up to 128 characters (use of wildcard characters in filenames is not supported). Filename extensions are not required but are highly recommended.

Example 1

```
RECALL:SETUp FACTory
```

This command recalls (and makes current) the front-panel setup to its factory defaults.

Example 2

```
RECALL:SETUp 2
```

This command recalls the front panel setup from setup 2.

Example 3

```
RECALL:SETUp "TEK00000.SET"
```

This command recalls the front panel setup from the file `TEK00000.SET` in the default directory and on the default drive.

RECALL:WAVEform

Description

This command (no query form) recalls a stored waveform to a reference location. This command is equivalent to selecting Reference Waveforms from the File menu and then choosing Recall Wfm.

Group

Save and Recall

Related Commands

DELEte:WAVEform, SAVE_WAVEform

Syntax

```
RECALL:WAVEform {<file path>|REF<x>}
```

Arguments

- REF<x>

This specifies the location in internal reference memory from where the waveform is recalled. Reference memory location values range from 1 through 4.

- <file path>

This is the location from where the waveform will be recalled.

<file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. If you do not specify them, the instrument will read the waveform from the default directory. <filename> stands for a filename of up to 128 characters (use of wildcard characters in filenames is not supported). Filename extensions are not required but are highly recommended.

Example

```
RECALL:WAVEform "TEK00000.WFM",REF1
```

This command recalls the waveform stored in the file named TEK00000.WFM to reference location 1.

REF<x>:HORizontal:POSition

Description

This command sets or queries the horizontal position of the reference waveform. The reference waveform is specified by x, which ranges from 1 through 4. This command is equivalent to selecting Reference Waveforms from the File menu, choosing Reference Setup from the drop-down list, selecting a reference waveform, and then entering the Horz Pos value.

Group

Vertical

Syntax 1

```
REF<x>:HORizontal:POSition <NR3>
```

Syntax 2

```
REF<x>:HORizontal:POSition?
```

Argument

- <NR3>

Specifies the horizontal position of the reference waveform <1 through 4> in percentage of the waveform, displayed to the right of the center vertical graticule. The range is 0 through 100.

Example 1

```
REF2:HORizontal:POSition 0.0
```

This command positions the start of the Reference 2 waveform at the center graticule.

Example 2

```
REF1:HORizontal:POSition?
```

This query might return :HORIZONTAL:REF1:POSITION 50, indicating that the Reference 1 waveform is centered on the horizontal graticule.

REF<x>:LABel?

Description

This query-only command returns a branch query containing the waveform label name and the coordinates at which the label (attached to the displayed waveform of the specified reference) is displayed. References are specified by x, which ranges from 1 through 4. This command is equivalent to selecting Reference Waveforms from the File menu and then choosing Label from the drop-down list.

Group

Vertical

Related Commands

REF<x>:LABel:NAME?, REF<x>:LABel:XPOS?,
REF<x>:LABel:YPOS?

Syntax

REF<x>:LABel?

Argument

None.

Example

REF1:LABel?

This query might return

```
:REF1:LABel:NAME "Myname";XPOS 2.5000;YPOS  
-2.5000.
```

REF<x>:LABel:NAME

Description

This command sets or queries the label of the designated waveform. References are specified by x, which ranges from 1 through 4. This command is equivalent to selecting Reference Waveforms from the File menu and then choosing Reference Setup from the drop-down list. Then select the tab associated with the reference for which you want to create a label and then enter a label in the Label box.

Group

Vertical

Related Commands

REF<x>:LABel

Syntax 1

REF<x>:LABel:NAME <String>

Syntax 2

REF<x>:LABel:NAME?

Argument

- <String>

This is the character string that will be used for the reference waveform label name.

Example 1

REF3:LABel:NAME?

This query might return

```
:REF3:LABEL:NAME "Signal2"
```

indicating that the label name for Reference 3 is currently set to "Signal2".

Example 2

```
REF4:LABel:NAME "My Reference"
```

This command sets the label name of Reference 4 to "My Reference".

REF<x>:LABel:XPOS

Description

This command sets or queries the X screen offset at which the label (attached to the displayed waveform of the specified reference) is displayed, relative to the left edge of the screen. References are specified by x, which ranges from 1 through 4. This command is equivalent to selecting Reference Waveforms from the File menu and then choosing Reference Setup from the drop-down list. Then either view or set X Position.

Group

Vertical

Related Commands

REF<x>:LABel

Syntax 1

```
REF<x>:LABel:XPOS <NR3>
```

Syntax 2

```
REF<x>:LABel:XPOS?
```

Argument

- <NR3>

This is the location (in divisions) where the waveform label for the selected reference is displayed, relative to the left edge of the screen.

Example 1

```
REF4:LABel:XPOS 2.5
```

This command moves the waveform label for the REF3 waveform, so that it begins 2.5 divisions to the right of the left edge of the screen.

Example 2

```
REF2:LABel:XPOS?
```

This query might return

```
:REF2:LABEL:XPOS 0.0000
```

indicating that the x-axis for the REF 2 waveform is currently on the far left edge of the screen.

REF<x>:LABel:YPOS

Description

This command sets or queries the Y screen offset at which the label (attached to the displayed waveform of the specified reference) is displayed, relative to the vertical position of the associated waveform. References are specified by x, which ranges from 1 through 4. This command is equivalent to selecting Reference Waveforms from the File menu and then choosing Reference Setup from the drop-down list. Then either view or set Y Position.

Group

Vertical

Related Commands

REF<x>:LABel

Syntax 1

```
REF<x>:LABel:YPOS <NR3>
```

Syntax 2

```
REF<x>:LABel:YPOS?
```

Argument

This is the location (in divisions) where the waveform label for the selected reference is displayed, relative to the vertical position of the associated waveform.

Example 1

```
REF3:LABel:YPOS -2.5
```

This command moves the waveform label for the REF3 waveform to beneath (2.5 divisions below) the vertical position of the associated waveform.

Example 2

```
REF2:LABel:YPOS?
```

This query might return

```
:REF2:LABEL:YPOS 0.0000
```

indicating that the waveform label for the REF2 waveform is currently located at the same Y position as the associated waveform.

REF<x>:VERTical:POSition

Description

This command sets or queries the vertical position of the specified reference waveform. The reference waveform is specified by x, which ranges from 1 through 4. This command is equivalent to selecting Reference Waveforms from the File menu, choosing Reference Setup from the drop-down list, selecting a reference waveform, and then entering the Position value.

Increasing the position value of a waveform causes the waveform to move up, and decreasing the position value causes the waveform to move down. Position adjusts only the display position of a waveform. The position value determines the vertical graticule coordinate at which signal values are displayed. For example, if the position for Reference 3 is set to 2.0, the signal represented by that reference would be displayed at 2.0 divisions above the center of the screen.

Note: References are static. All position and scale changes are applied in a post-processing mode.

Group

Vertical

Related Commands

CH<x>:POSition, MATH<x>:POSition

Syntax 1

```
REF<x>:VERTical:POSition <NR3>
```

Syntax 2

```
REF<x>:VERTical:POSition?
```

Argument

- <NR3>

This is the desired position, in divisions from the center horizontal graticule. The range is from -5 to 5 divisions.

Example 1

```
REF2:VERTical:POSition 1.3E+00
```

This command positions the Reference 2 input signal 1.3 divisions above the center horizontal graticule.

Example 2

REF1:VERTical:POSition?

This query might return :REF1:VERTical:POSITION
-1.3000E+00, indicating that the current position of Reference 1 is 1.3 divisions below the center horizontal graticule.

REF<x>:VERTical:SCAle

Description

This command sets or queries the vertical scale of the specified reference waveform. The reference is specified by x, which can range from 1 through 4. This command is equivalent to selecting Reference Waveforms from the File menu, choosing Reference Setup from the drop-down list, selecting a reference waveform, and then entering the Scale value

Each waveform has a vertical scale parameter. For a signal with constant amplitude, increasing the Scale causes the waveform to be displayed smaller. Decreasing the scale causes the waveform to be displayed larger.

Scale affects all waveforms, but affects reference and math waveforms differently from channel waveforms:

- For reference and math waveforms, this setting controls the display only, graphically scaling these waveforms and having no affect on the acquisition hardware.
- For channel waveforms, this setting controls the vertical size of the acquisition window as well as the display scale. The range and resolution of scale values depends on the probe attached and any other external factors you have specified.

Group

Vertical

Related Commands

CH<x>:SCAle, MATH<x>:SCAle

Syntax 1

```
REF<x>:VERTical:SCAle <NR3>
```

Syntax 2

```
REF<x>:VERTical:SCAle?
```

Argument

- <NR3>

This is the gain, in user units per division.

Example 1

```
REF4:VERTical:SCAle 100E-03
```

This command sets the Reference 4 scale to 100 mV per division.

Example 2

```
CH2:VERTical:SCAle?
```

This query might return `:REF2:SCALE 1.0000E+00`, indicating that the current scale setting of Reference 2 is 1 volt per division.

REM

Description

This command (no query form) embeds a comment within GPIB programs as a means of internally documenting the programs. The oscilloscope ignores these embedded comment lines.

Group

Miscellaneous

Related Commands

NEWpass, *PUD

Syntax

```
REM <QString>
```

Argument

- <QString>

This is a string that can contain a maximum of 80 characters.

Example

```
REM "This is a comment"
```

The oscilloscope ignores this comment string.

ROSc:SOURce

Description

This command sets or queries the source for the 10 MHz timebase reference. Depending on the command argument that you specify, you can use an external reference or use the internal crystal oscillator as the timebase reference.

This command is also useful for synchronizing multiple instruments.

Group

Miscellaneous

Related Commands

ROSc:StAtE

Syntax 1

```
ROSc:SOURce {INTERNAL|EXTERNAL}
```

Syntax 2

```
ROSc:SOURce?
```

Arguments

- INTERNAL

This specifies the internal 10 MHz crystal oscillator as the timebase reference.

- EXTERNAL

This specifies the user-supplied external signal as the timebase reference.

Example 1

```
ROSc:SOURce INTERNAL
```

This command specifies the internal 10 MHz crystal oscillator as the timebase reference.

Example 2

ROSc:SOURce?

This query might return:

ROSC:SOURCE INTERNAL

indicating that the 10 MHz crystal oscillator is being used as the timebase reference.

ROSc:STate?

Description

This query-only command returns the state of the 10 MHz timebase reference.

Group

Miscellaneous

Related Commands

ROSc:SOUrce

Syntax

ROSc:STate?

Example

ROSc:STate?

This query might return:

ROSC:STATE LOCKED

indicating that the state of the 10 MHz timebase reference is locked.

SAVE:MASK

Description

This command (no query form) saves the User mask to a mass storage file.

Group

Mask

Related Commands

RECALL:MASK

Syntax

SAVE:MASK <Qstring>

Arguments

- <Qstring>

This is the location to where the mask is stored. <file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional.

The file is stored in ASCII format and is user-editable.

Example

```
SAVE:MASK, "TEST"
```

This command saves the User mask to the file TEST in the default directory and on the default drive.

SAVE:SETUp

Description

This command (no query form) saves the current front-panel setup into the specified memory location or file. This is equivalent to selecting Instrument Setup from the File menu and then choosing the Save Setups tab.

Group

Save and Recall

Related Commands

*RCL, RECALL:SETUp, *SAV

Syntax

```
SAVE:SETUp {<file path>|<NR1>}
```

Arguments

- <file path>

This is the location from where the waveform will be recalled.

<file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. If you do not specify them, the instrument will read the waveform from the default directory. <filename> stands for a filename of up to 125 characters, followed by a period (".") and the three-character extension "SET". The oscilloscope will generate an error if you use any other extension for saving a setup.

- <NR1>

This specifies a location for saving the current front-panel setup. The front-panel setup value ranges from 1 to 10. Using an out-of-range value causes an execution error. Any settings that have been stored previously at this location will be overwritten.

Example 1

```
SAVE:SETUP 5
```

This command saves the current front-panel setup in memory location 5.

Example 2

```
SAVE:SETUP "TEK00000.SET"
```

This command saves the current front-panel setup in the file `TEK00000.SET` in the default directory and on the default drive.

SAVE:WAVEform

Description

This command (no query form) saves a waveform to one of four reference memory locations or a mass storage file. This command is equivalent to selecting Reference Waveforms from the File menu and then choosing Save Wfm from the drop-down list.

To save the full record length, you may need to properly set values with the DATA:START and DATA:STOP commands before running the SAVE:WAVEform command.

Group

Save and Recall

Related Commands

DATA:START, DATA:STOP,

DELEte:WAVEform, RECALL:WAVEform

Syntax

```
SAVE:WAVEform <wfm>, {<file path> | REF<x>}
```

Arguments

- <wfm>

This is the waveform that will be saved. Valid waveforms include CH<x>, MATH<y>, and REF<x>.

- REF<x>

This specifies the location in internal reference memory to where the waveform is stored.

- <file path>

This is the location to where the waveform is stored.

<file path> is a quoted string that defines the file name and path. Input the file path using the form <drive>/<dir>/<filename>. <drive> and one or more <dir>s are optional. If you do not specify them, the instrument will read the waveform from the default directory. <filename> stands for a filename of up to 125 characters, followed by a period (".") and the three-character extension. Internal format waveforms use the .wfm extension for spreadsheet format files or the .dat extension for MathCad format files. The oscilloscope will generate an error if you use any other extension for saving a waveform.

Example 1

```
SAVE:WAVEFORM MATH2,REF1
```

This command saves the Math2 waveform in reference memory location2.

Example 2

```
SAVE:WAVEFORM MATH1,"TEK0000.WFM"
```

This command saves the Math1 waveform to the file TEK00000.WFM in the default directory and on the default drive.

SAVE:WAVEform:FILEFormat

Description

This command specifies or returns the file format for saved waveforms. Waveform header and timing information is included in the resulting file of non-internal formats. The instrument saves DPO and WFMDB waveforms as a 500x200 matrix, with the first row corresponding to the most recently acquired data. The values specified by DATA:START, DATA:STOP, DATA:FRAMESTART and DATA:FRAMESTOP determine the range of waveform data to output. In the event that DATA:STOP value is greater than the current record length, the current record length determines the last output value.

This command is equivalent to selecting Export Setup from the File menu, clicking the Waveforms tab and then choosing the desired waveform file format from the Data destination drop-down list. Note that you choose the waveform file format after first selecting Select for Export from the File menu and then choosing Waveform (data) from the resulting drop-down list.

Group

Save and Recall

Related Commands

CURVe , DATA, DATA:START, DATA:STOP, DATA:FRAMESTART, DATA:FRAMESTOP, SAVE:WAVEform, WFMInpre:NR_FR?, WFMInpre:NR_Pt, WFMOutpre:NR_FR?, WFMOutpre:NR_Pt?

Syntax 1

```
SAVE:WAVEform:FILEFormat {INTERNAL | MATH-Cad | MATLAB | SPREADSHEETCsv | SPREADSHEETTxt}
```

Syntax 2

```
SAVE:WAVEform:FILEFormat?
```

Arguments

- INTERNAL

This specifies that waveforms are saved in internal format, using a .wfm filename extension. These files can be recalled as reference waveforms. When this argument is specified, the settings specified via the DATA:START, DATA:STOP, DATA:FRAMESTART, and DATA:FRAMESTOP commands have no meaning as the entire waveform is saved.

- **MATHCad**

This specifies that waveforms are saved in MathCad format, using a .dat filename extension. When saving in this format, waveform values are delimited with new lines. MathCad format enables export of waveform data to MathCad or Matlab.

For FastAcq or WFMDB waveforms, data is exported as a matrix. For these formats, waveform header and timing information is saved in a separate file. MathCad format header files use a _hdr.dat extension.

- **MATLab**

This specifies that waveforms are saved in Matlab format, using a .dat filename extension. When saving in this format, waveform values are delimited with new lines. Matlab format enables easy import of waveform data into MathCad or Matlab.

For FastAcq or WFMDB waveforms, data is exported as a matrix. For these formats, waveform header and timing information is saved in a separate file. Matlab format header files use a _hdr.dat extension.

- **SPREADSHEETC**s**v**

This specifies that waveform data is saved in a format that contains comma delimited values. These waveform data files are named using the .csv filename extension. Saving waveforms in CSV format enables spreadsheet programs to import the data.

- **SPREADSHEETT**x**t**

This specifies that waveform data is saved in a format that contains tab delimited values. These waveform data files are named using the .txt filename extension. Saving waveforms in this format enables spreadsheet programs to import the data.

Example 1

```
SAVE:WAVEFORM:FILEFORMAT INTERNAL
```

This command specifies that the internal file format is the format used for saving waveforms.

Example 2

SAVE:WAVEFORM:FILEFORMAT?

This query might return:

SAVE:WAVEFORM:FILEFORMAT INTERNAL

indicating that waveforms are saved using the internal format.

SET?

Description

This query-only command returns the commands that list the oscilloscope settings, except for configuration information for the calibration values. You can use these commands to return the oscilloscope to the state it was in when you made the SET? query. This command is identical to the *LRN? command.

Group

Miscellaneous

Related Commands

HEADer, *LRN?, VERBose

Syntax

SET?

Note: The SET? query always returns command headers, regardless of the setting of the HEADer command. This is because the returned commands are intended to be sent back to the oscilloscope as a command string. The VERBose command can still be used to specify whether the returned headers should be abbreviated or full-length.

Example

SET?

This query might return the following response:

```
:ACQUIRE:STOPAFTER RUNSTOP;STATE 1;MODE
SAMPLE;NUMENV
10;NUMAVG 16;REPET 1;:FASTACQ:STATE 1;:AP-
PLICATION
:GPKNOB1:ACTIVE 0;:APPLICATION:GPKNOB2:AC-
TIVE 0;
:APPLICATION:WINDOW:HEIGHT 0;WIDTH 0;:AP-
PLICATION
:SCOPEAPP:STATE NOTRUNNING;WINDOW FULL-
SCREEN;
:APPLICATION:EXTAPP:STATE NOTRUNNING;:AUX-
OUT:SOURCE
ATRIGGER;EDGE FALLING;:CMDBATCH 1;:HEADER
0;:LOCK
NONE;:VERBOSE 1;:ALIAS:STATE 0;:DIS-
PLAY:CLOCK 1
;COLOR:PALETTE NORMAL;MATHCOLOR DE-
```

```

FAULT;REFCOLOR
DEFAULT;:DISPLAY:FILTER SINX;FORMAT
YT;GRATICULE
FULL;INTENSITY:WAVEFORM 60.0000;AUTOBRIGHT
1
;SCREENSAVER 1;SCREENSAVERDELAY
28800;:DISPLAY
:PERSISTENCE OFF;STYLE VECTORS;TRIGBAR
SHORT;TRIGT 1
;VARPERSIST 500.0000E-3;:HARDCOPY:FILENAME
"";FORMAT
BMP;LAYOUT PORTRAIT;PALETTE HARDCOPY;PORT
FILE;
:HORIZONTAL:DELAY:MODE 1;POSITION
50.0000;TIME
0.0000;:HORIZONTAL:MAIN:SCALE
200.0000E-9;POSITION
50.0000;:HORIZONTAL:RECORDLENGTH 2500;ROLL
AUTO
;FASTFRAME:STATE 0;LENGTH 2500;COUNT 2;SE-
LECTED:CH1 2
;CH2 2;CH3 2;CH4 2;MATH1 2;MATH2 2;MATH3
2;MATH4 2
;REF1 2;REF2 2;REF3 2;REF4 2;:HORIZON-
TAL:FASTFRAME
:REF:SOURCE CH1;FRAME 1;:HORIZONTAL:FAST-
FRAME:TRACK
LIVE;:DIAG:LEVEL SUBSYS;:TRIGGER:A:MODE
AUTO;TYPE
EDGE;LEVEL 0.0000;HOLDOFF:BY DEFAULT;TIME
250.0000E-9;
:TRIGGER:A:EDGE:SOURCE CH1;COUPLING
DC;SLOPE RISE;
:TRIGGER:A:LOGIC:CLASS PATTERN;FUNCTION
AND;WHEN TRUE
;THRESHOLD:CH1 1.4000;CH2 1.4000;CH3
1.4000;CH4 1.4000;
:TRIGGER:A:LOGIC:INPUT:CH1 HIGH;CH2 X;CH3
X;:TRIGGER:A
:LOGIC:PATTERN:INPUT:CH4 X;:TRIGGER:A:LOG-
IC:PATTERN
:WHEN TRUE;WHEN:LESSLIMIT 5.0000E-9;MORE-
LIMIT 5.0000E-9;
:TRIGGER:A:LOGIC:SETHOLD:CLOCK:EDGE
RISE;THRESHOLD 1.4000
;SOURCE CH2;:TRIGGER:A:LOGIC:SE-
THOLD:DATA:THRESHOLD
1.4000;SOURCE CH1;:TRIGGER:A:LOGIC:SE-
THOLD:HOLDTIME
2.0000E-9;SETTIME 3.0000E-9;:TRIG-
GER:A:LOGIC:STATE:INPUT

```

```

:CH4 RISE;:TRIGGER:A:LOGIC:STATE:WHEN
TRUE;:TRIGGER:A
:PULSE:CLASS GLITCH;SOURCE
CH1;GLITCH:WIDTH 2.0000E-9
;TRIGIF ACCEPT;POLARITY POSITIVE;:TRIG-
GER:A:PULSE:RUNT
:POLARITY POSITIVE;THRESHOLD:HIGH
1.2000;LOW 800.0000E-3;
:TRIGGER:A:PULSE:RUNT:WHEN OCCURS;WIDTH
2.0000E-9;
:TRIGGER:A:PULSE:TRANSITION:DELTATIME
2.0000E-9;POLARITY
POSITIVE;THRESHOLD:HIGH 1.2000;LOW
800.0000E-3;:TRIGGER
:A:PULSE:TRANSITION:WHEN SLOWERTHAN;:TRIG-
GER:A:PULSE
:WIDTH:LOWLIMIT 2.0000E-9;HIGHLIMIT
2.0000E-9;WHEN WITHIN
;POLARITYPOSITIVE;:TRIGGER:A:PULSE:TIME-
OUT:POLARITY
STAYSHIGH;TIME 2.0000E-9;:TRIGGER:B:STATE
0;TYPE EDGE
;LEVEL 0.0000;BY EVENTS;EDGE:SOURCE
CH1;SLOPE RISE
;COUPLING DC;:TRIGGER:B:TIME
16.0000E-9;EVENTS:COUNT 2;
:MATH1:DEFINE " ";NUMAVG 2;SCALE
1.0000;POSITION 0.0000
;LABEL:NAME "";XPOS 5;YPOS 65;:MATH1:SPEC-
TRAL:MAG DB
;PHASE DEGREES;GATEPOS -800.0000E-12;GATE-
WIDTH 1.9992E-6
;REFLEVEL 20.0000;REFLEVELOFFSET
223.6000E-3;SPAN
600.0000E+6;CENTER 325.0000E+6;RESBW
1.0004E+6;WINDOW
GAUSSIAN;SUPPRESS -35000.0000E-3;UNWRAP
0;LOCK 0;
:MATH2:DEFINE " ";NUMAVG 2;SCALE
1.0000;POSITION 0.0000
;LABEL:NAME "";XPOS 5;YPOS 80;:MATH2:SPEC-
TRAL:MAG DB
;PHASE DEGREES;GATEPOS -800.0000E-12;GATE-
WIDTH 1.9992E-6
;REFLEVEL 20.0000;REFLEVELOFFSET
223.6000E-3;SPAN
600.0000E+6;CENTER 325.0000E+6;RESBW
1.0004E+6;WINDOW
GAUSSIAN;SUPPRESS -35000.0000E-3;UNWRAP
0;LOCK 0;
:MATH3:DEFINE " ";NUMAVG 2;SCALE

```

```

1.0000;POSITION 0.0000
;LABEL:NAME "";XPOS 5;YPOS 95;:MATH3:SPEC-
TRAL:MAG DB
;PHASE DEGREES;GATEPOS -800.0000E-12;GATE-
WIDTH 1.9992E-6
;REFLEVEL 20.0000;REFLEVELOFFSET
223.6000E-3;SPAN
600.0000E+6;CENTER 325.0000E+6;RESBW
1.0004E+6;WINDOW
GAUSSIAN;SUPPRESS -35000.0000E-3;UNWRAP
0;LOCK 0;
:MATH4:DEFINE " ";NUMAVG 2;SCALE
1.0000;POSITION 0.0000
;LABEL:NAME "";XPOS 5;YPOS
110;:MATH4:SPECTRAL:MAG DB
;PHASE DEGREES;GATEPOS -800.0000E-12;GATE-
WIDTH 1.9992E-6
;REFLEVEL 20.0000;REFLEVELOFFSET
223.6000E-3
;SPAN 600.0000E+6;CENTER 325.0000E+6;RESBW
1.0004E+6
;WINDOW GAUSSIAN;SUPPRESS
-35000.0000E-3;UNWRAP 0;LOCK 0
;:HISTOGRAM:BOXPCNT
30.0000,25.1000,70.0000,75.2000
;DISPLAY LINEAR;STATE 0;FUNCTION HORIZON-
TAL;SIZE 2.0000
;SOURCE CH1;:CH1:BANDWIDTH 1.0000E+9;COU-
PLING DC;DESKEW
0.0000;OFFSET 0.0000;POSITION 0.0000;SCALE
100.0000E-3
;TERMINATION 1.0000E+6;PROBEFUNC:EXTATTEN
1.0000
;EXTUNITS "None";:CH1:LABEL:NAME "";XPOS
5;YPOS 5;
:CH2:BANDWIDTH 1.0000E+9;COUPLING DC;DES-
KEW 0.0000
;OFFSET 0.0000;POSITION 0.0000;SCALE
100.0000E-3
;TERMINATION 1.0000E+6;PROBEFUNC:EXTATTEN
1.0000
;EXTUNITS "None";:CH2:LABEL:NAME "";XPOS
5;YPOS 20;
:CH3:BANDWIDTH 1.0000E+9;COUPLING DC;DES-
KEW 0.0000
;OFFSET 0.0000;POSITION 0.0000;SCALE
100.0000E-3
;TERMINATION 1.0000E+6;PROBEFUNC:EXTATTEN
1.0000
;EXTUNITS "None";:CH3:LABEL:NAME "";XPOS
5;YPOS 35;

```

```

:CH4:BANDWIDTH 1.0000E+9;COUPLING DC;DES-
KEW 0.0000
;OFFSET 0.0000;POSITION 0.0000;SCALE
100.0000E-3
;TERMINATION 1.0000E+6;PROBEFUNC:EXTATTEN
1.0000
;EXTUNITS "None";:CH4:LABEL:NAME "";XPOS
5;YPOS 50;
:SELECT:CH1 1;CH2 0;CH3 0;CH4 0;MATH1
0;MATH2 0
;MATH3 0;MATH4 0;REF1 0;REF2 0;REF3 0;REF4
0;CONTROL
CH1;:CURSOR:STATE OFF;FUNCTION VBARS;MODE
INDEPENDENT
;SOURCE CH1;VBARS:UNITS SECONDS;POSITION1
-800.0000E-9
;POSITION2 800.0000E-9;:CURSOR:HBARS:POSI-
TION1
300.0000E-3;POSITION2 -300.0000E-3;:CUR-
SOR:PAIRED
:POSITION1 -800.0000E-9;POSITION2
800.0000E-9;:CURSOR
:SPLIT:POSITION1 -800.0000E-9;POSITION2
800.0000E-9
;SOURCE2 CH1;UNITS BASE;MEASUREMENT:GATING
OFF;METHOD
HISTOGRAM;IMMED:TYPE UNDEFINED;SOURCE1
CH1;SOURCE2 CH1
;DELAY:EDGE1 RISE;EDGE2 RISE;DIRECTION
FORWARDS;
:MEASUREMENT:MEAS1:STATE 0;TYPE UNDE-
FINED;SOURCE1 CH1
;SOURCE2 CH1;DELAY:EDGE1 RISE;EDGE2
RISE;DIRECTION
FORWARDS;:MEASUREMENT:MEAS2:STATE 0;TYPE
UNDEFINED
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1
RISE;EDGE2 RISE
;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS3:STATE 0
;TYPE UNDEFINED;SOURCE1 CH1;SOURCE2
CH1;DELAY:EDGE1
RISE;EDGE2 RISE;DIRECTION FORWARDS;:MEA-
SUREMENT:MEAS4
:STATE 0;TYPE UNDEFINED;SOURCE1
CH1;SOURCE2 CH1
;DELAY:EDGE1 RISE;EDGE2 RISE;DIRECTION
FORWARDS;
:MEASUREMENT:MEAS5:STATE 0;TYPE UNDE-
FINED;SOURCE1 CH1
;SOURCE2 CH1;DELAY:EDGE1 RISE;EDGE2

```



```

RISE;DIRECTION
FORWARDS;;MEASUREMENT:MEAS6:STATE 0;TYPE
UNDEFINED
;SOURCE1 CH1;SOURCE2 CH1;DELAY:EDGE1
RISE;EDGE2 RISE
;DIRECTION FORWARDS;;MEA-
SUREMENT:MEAS7:STATE 0;TYPE
UNDEFINED;SOURCE1 CH1;SOURCE2
CH1;DELAY:EDGE1 RISE
;EDGE2 RISE;DIRECTION FORWARDS;;MEA-
SUREMENT:MEAS8
:STATE 0;TYPE UNDEFINED;SOURCE1
CH1;SOURCE2 CH1;DELAY
:EDGE1 RISE;EDGE2 RISE;DIRECTION FOR-
WARDS;;MEASUREMENT
:REFLEVEL:METHOD PERCENT;ABSOLUTE:HIGH
0.0000;LOW 0.0000
;MID1 0.0000;MID2 0.0000;;MEASUREMENT:RE-
FLEVEL:PERCENT
:HIGH 90.0000;LOW 10.0000;MID1
50.0000;MID2 50.0000;
:MEASUREMENT:STATISTICS:MODE OFF;WEIGHTING
32;;ZOOM
:MODE 0;GRATICULE:SIZE 50;;ZOOM:HORIZON-
TAL:LOCK ALL;
:ZOOM:CH1:HORIZONTAL:POSITION
50.0000;SCALE 2;;ZOOM
:CH1:VERTICAL:POSITION 0.0000;SCALE
1.0000;;ZOOM:CH2
:HORIZONTAL:POSITION 50.0000;SCALE
2;;ZOOM:CH2:VERTICAL
:POSITION 0.0000;SCALE
1.0000;;ZOOM:CH3:HORIZONTAL
:POSITION 50.0000;SCALE 2;;ZOOM:CH3:VER-
TICAL:POSITION
0.0000;SCALE 1.0000;;ZOOM:CH4:HORIZON-
TAL:POSITION
50.0000;SCALE 2;;ZOOM:CH4:VERTICAL:POSI-
TION 0.0000
;SCALE 1.0000;;ZOOM:MATH1:HORIZONTAL:POSI-
TION 50.0000
;SCALE 2;;ZOOM:MATH1:VERTICAL:POSITION
0.0000;SCALE 1.0000;;ZOOM:MATH2:HORIZON-
TAL:POSITION 50.0000;SCALE 2;
:ZOOM:MATH2:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:MATH3:HORIZONTAL:POSITION
50.0000;SCALE 2;;ZOOM
:MATH3:VERTICAL:POSITION 0.0000;SCALE
1.0000;;ZOOM
:MATH4:HORIZONTAL:POSITION 50.0000;SCALE

```

```

2;:ZOOM:MATH4
:VERTICAL:POSITION 0.0000;SCALE
1.0000;:ZOOM:REF1
:HORIZONTAL:POSITION 50.0000;SCALE
2;:ZOOM:REF1:VERTICAL
:POSITION 0.0000;SCALE
1.0000;:ZOOM:REF2:HORIZONTAL
:POSITION 50.0000;SCALE 2;:ZOOM:REF2:VER-
TICAL:POSITION
0.0000;SCALE 1.0000;:ZOOM:REF3:HORIZON-
TAL:POSITION
50.0000;SCALE 2;:ZOOM:REF3:VERTICAL:POSI-
TION 0.0000
;SCALE 1.0000;:ZOOM:REF4:HORIZONTAL:POSI-
TION 50.0000
;SCALE 2;:ZOOM:REF4:VERTICAL:POSITION
0.0000;SCALE
1.0000;:REF1:LABEL:NAME "";XPOS 5;YPOS
125;:REF1:SCALE
1.0000;POSITION 0.0000;:REF2:LABEL:NAME
"";XPOS 5;YPOS
140;:REF2:SCALE 1.0000;POSITION
0.0000;:REF3:LABEL:NAME
"";XPOS 5;YPOS 155;:REF3:SCALE
1.0000;POSITION 0.0000;
:REF4:LABEL:NAME "";XPOS 5;YPOS
170;:REF4:SCALE 1.0000
;POSITION 0.0000;:DATA:DESTINATION
REF1;ENCDG RIBINARY
;SOURCE CH1;START 1;STOP 500;WIDTH 8

```

SElect?

Description

This query-only command returns the selected waveform that is affected by the front-panel controls and the display status (on or off) of all waveforms. This query command is equivalent to selecting Measurement Setup from the Measure menu and viewing the Source waveform setting.

Group

Vertical

Syntax

SElect?

Example

SElect?

This query might return

```
:SELECT:CH1 1;CH2 0;CH3 0;CH4 0;MATH1  
0;MATH2 0;MATH3 0;MATH4 0;REF1 0;REF2  
0;REF3 0;REF4 0;CONTROL CH1
```

SElect:<wfm>

Description

This command turns on a specified waveform. This action also resets the acquisition. The query returns whether the channel is on or off but does not indicate whether it is the selected waveform.

Group

Vertical

Syntax 1

```
SElect:<wfm> {ON|OFF|<NR1>}
```

Syntax 2

```
SElect:<wfm>?
```

Arguments

- ON

This turns on the display of the specified waveform. This waveform also becomes the selected waveform.

- OFF

This turns off the display of the specified waveform.

- <NR1>

A 0 turns off the display of the specified waveform; any other value turns on the display of the specified waveform.

Example

```
SElect:CH2 ON
```

This command turns the channel 2 display on and selects channel 2.

```
SElect:REF1?
```

This query might return

```
:SELECT:REF1
```

indicating that REF1 is the selected waveform.

SElect:CONTRol<wfm>

Description

This command sets or queries the waveform that is selected as the 'implied' recipient of channel-related commands that support legacy style programs. The command form also performs the equivalent of a SELECT:<wfm> ON command. This command is equivalent to selecting Measurement Setup from the Measure menu and either viewing or setting the Source waveform.

Group

Vertical

Syntax 1

```
SElect:CONTRol (CH<x> | MATH<x> | REF<x>)
```

Syntax 2

```
SElect:CONTRol?
```

Arguments

- CH<x>

This selects the specified channel waveform as the waveform that is affected by the front-panel controls. The valid channel waveform range is from 1 through 4.

- MATH<x>

This selects the specified math waveform as the waveform that is affected by the front-panel controls. The valid math waveform range is from 1 through 4.

- REF<x>

This selects the specified reference waveform as the waveform that is affected by the front-panel controls. The valid reference waveform range is from 1 through 4.

Example 1

```
SElect:CONTRol CH2
```

This command resets acquisition, displays channel 2, and causes the selected waveform to be the implied object of waveform-manipulating commands.

Example 2

SElect:CONTROL?

This query might return

:SELECT:CONTROL MATH3

indicating that math 3 is the implied object of waveform-manipulating commands.

SETUp:NAME

Description

This command sets and queries a Setup Label that you define. This command is equivalent to selecting Instrument Setup from the File menu, choosing the Save Setups tab and then adding a meaningful label to the selected setup.

Group

Save and Recall

Related Commands

SAVE:SETUp

Syntax 1

```
SETUp:NAME <NR1>,<QString>
```

Syntax 2

```
SETUp:NAME? <NR1>
```

Arguments

- <NR1>

This specifies a location in which the Setup Label is stored. Location values range from 1 through 10.

- <QString>

This is a string containing the Setup Label.

Example 1

```
SETUp:NAME 1 "My Setup"
```

This command changes the Setup Label for internal setup slot 1 to "My Setup".

Example 2

```
SETUp:NAME? 2
```

This query might return

```
:SETUP:NAME 2,"My Setup"
```

indicating the Setup Label that you defined for internal set-up slot 2.

TEKSecure

Description

This command initializes both waveform and setup memories, overwriting any previously stored data.

TEKSecure deletes all four waveform reference memory slots on the hard drive, if they exist, and puts all setups in the factory-initialized state. External setups that are stored on the hard drive are not affected.

Group

Miscellaneous

Syntax

```
TEKSecure
```

Example

```
TEKSecure
```

This command initializes both waveform and setup memories.

TEST

Description

This command (no query form) provides the ability to select and execute an item at any level of the test hierarchy (Test, Area or Subsystem). This command is equivalent to selecting Instrument Diagnostics from the Utilities menu, choosing a test and then pressing Run.

Note: Turning off both DIAg:CONTROL:HALT and DIAg:CONTROL:LOOP before executing the TEST command is recommended.

Group

Diagnostics

Related Commands

TEST:RESults?, TEST:RESults:VERBose?, TEST:STOP

Syntax

TEST

Argument

- <NR3>

This sets the test ID, which ranges from 0 through 3 characters. If no test ID is specified, all available diagnostics are executed.

Example 1

```
TEST "1"
```

This command executes all Acquisition tests (Subsystem:Acquisition).

Example 2

```
TEST "11"
```

This command executes all Acquisition Memory tests (Subsystem:Acquisition, Area:Memory)

Example 3

```
TEST "113"
```

This command executes a specific Acquisition Memory test (Subsystem:Acquisition, Area:Memory, Test:Address Lines).

TEST:RESults?

Description

This query-only command returns an abbreviated status about the results of the last TEST execution. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then reviewing the Diagnostic Status.

Group

Diagnostics

Related Commands

TEST, TEST:RESults:VERBose?

Syntax

TEST:RESults?

Example 1

TEST:RESULTS?

This query might return

:TEST:RESULTS "135"

indicating the specific failure code of the first test failure.

Example 2

TEST:RESULTS?

This query might return

:TEST:RESULTS "PASS"

indicating that the previously executed test passed.

Example 3

TEST:RESULTS?

This query might return

:TEST:RESULTS "*****"

indicating that the test was not run.

TEST:RESults:VERBose?

Description

This query-only command returns a more explanatory message about the results of the last TEST execution than the TEST:RESults query. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then clicking the Error Log button.

Group

Diagnostics

Related Commands

TEST, TEST:RESults?

Syntax

TEST:RESults:VERBose?

Example

TEST:RESULTS:VERBOSE?

This query might return

```
:TEST:RESULTS:VERBOSE DIAG ABORTED, ERROR  
15 in A2D-Dmux 1 interconnects"
```

TEST:STOP

Description

This command (no query form) causes test (or diagnostics) execution to terminate at the end of the next low-level test. This command is equivalent to selecting Instrument Diagnostics from the Utilities menu and then clicking Abort.

Group

Diagnostics

Related Commands

DIAG:STOP

Syntax

TEST:STOP

Example

TEST:STOP

This command terminates test (or diagnostics) execution at the end of the next low-level test.

TIME

Description

This command sets or queries the time that the oscilloscope can display. This command is equivalent to selecting Set Time & Date from the Utilities menu and then setting the fields in the Time group box

Group

Miscellaneous

Related Commands

DATE

Syntax 1

```
TIME <Qstring>
```

Syntax 2

```
TIME?
```

Argument

- <QString>

This is a time in the form "hh:mm:ss" where hh refers to a two-digit hour number, mm refers to a two-digit minute number from 01 to 60, and ss refers to a two-digit second number from 01 to 60.

Example 1

```
TIME "14:00:00"
```

This command specifies that the time is set to exactly 2:00 p.m..

Example 2

```
TIME?
```

This query might return

```
:TIME "14:05:17"
```

indicating the current time is set to 2:05 p.m..

TRIGger

Description

This command forces a trigger event to occur and the query returns current trigger parameters for the oscilloscope.

Group

Trigger

Syntax 1

```
TRIGger FORCe
```

Syntax 2

```
TRIGger?
```

Argument

- FORCe

This creates a trigger event. If TRIGger:STATE is set to READY, the acquisition will complete. Otherwise, this command will be ignored. This is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then clicking Force Trigger.

Example 1

```
TRIGger FORCe
```

This command forces a trigger event to occur.

Example 2

```
TRIGger?
```

This query might return

```
:TRIGGER:A:MODE AUTO;TYPE EDGE
;LEVEL 0.0000;HOLDOFF:BY DEFAULT;TIME
250.0000E-9;
:TRIGGER:A:EDGE:SOURCE CH1;COUPLING
DC;SLOPE RISE;
:TRIGGER:A:LOGIC:CLASS PATTERN;FUNCTION
AND;WHEN TRUE
;THRESHOLD:CH1 1.4000;CH2 1.4000;CH3
1.4000;CH4 1.4000;
:TRIGGER:A:LOGIC:INPUT:CH1 HIGH;CH2 X;CH3
X;
:TRIGGER:A:LOGIC:PATTERN:INPUT:CH4
X;:TRIGGER:A:LOGIC
```

```

:PATTERN:WHEN TRUE;WHEN:LESSLIMIT
5.0000E-9;MORELIMIT
5.0000E-9;:TRIGGER:A:LOGIC:SE-
THOLD:CLOCK:EDGE RISE
;THRESHOLD 1.4000;SOURCE CH2;:TRIG-
GER:A:LOGIC:SETHOLD
:DATA:THRESHOLD 1.4000;SOURCE CH1;:TRIG-
GER:A:LOGIC
:SETHOLD:HOLDTIME 2.0000E-9;SETTIME
3.0000E-9;:TRIGGER
:A:LOGIC:STATE:INPUT:CH4 RISE;:TRIG-
GER:A:LOGIC:STATE
:WHEN TRUE;:TRIGGER:A:PULSE:CLASS
GLITCH;SOURCE CH1
;GLITCH:WIDTH 2.0000E-9;TRIGIF ACCEPT;PO-
LARITY POSITIVE;
:TRIGGER:A:PULSE:RUNT:POLARITY POS-
ITIVE;THRESHOLD:HIGH
1.2000;LOW 800.0000E-3;:TRIG-
GER:A:PULSE:RUNT:WHEN OCCURS
;WIDTH 2.0000E-9;:TRIGGER:A:PULSE:TRAN-
SITION:DELTATIME
2.0000E-9;POLARITY POSITIVE;THRESHOLD:HIGH
1.2000;LOW
800.0000E-3;:TRIGGER:A:PULSE:TRAN-
SITION:WHEN SLOWERTHAN;
:TRIGGER:A:PULSE:WIDTH:LOWLIMIT
2.0000E-9;HIGHLIMIT
2.0000E-9;WHEN WITHIN;POLARITY POS-
ITIVE;:TRIGGER:A
:PULSE:TIMEOUT:POLARITY STAYSHIGH;TIME
2.0000E-9;
:TRIGGER:B:STATE 0;TYPE EDGE;LEVEL
0.0000;BY EVENTS
;EDGE:SOURCE CH1;SLOPE RISE;COUPLING
DC;:TRIGGER:B
:TIME 16.0000E-9;EVENTS:COUNT 2

```

TRIGger:A

Description

This command sets the A trigger level automatically to 50% of the minimum/maximum value of the trigger input signal. The query returns current A trigger parameters. TRIGger:A? is a composite of TRIGger:A:EDGE?, TRIGger:A:LOGic?, and TRIGger:A:PULSE?. The trigger level is the voltage threshold through which the trigger source signal must pass to generate a trigger event. This command is equivalent to pushing the **LEVEL** knob on the front-panel.

Group

Trigger

Related Commands

TRIGger:A:EDGE?, TRIGger:A:LOGic?, TRIGger:A:PULSE?

Syntax 1

```
TRIGger:A SETLevel
```

Syntax 2

```
TRIGger:A?
```

Argument

- SETLevel

This sets the A trigger level to 50% of minimum and maximum.

Example 1

```
TRIGger:A SETLevel
```

This command sets the A trigger level to 50% of minimum and maximum.

Example 2

```
TRIGger:A?
```

This query might return

```
:TRIGGER:A:MODE AUTO;TYPE EDGE  
;LEVEL 0.0000;HOLDOFF:BY DEFAULT;TIME  
250.0000E-9;  
:TRIGGER:A:EDGE:SOURCE CH1;COUPLING  
DC;SLOPE RISE;
```



```

:TRIGGER:A:LOGIC:CLASS PATTERN;FUNCTION
AND;WHEN TRUE
;THRESHOLD:CH1 1.4000;CH2 1.4000
;CH3 1.4000;CH4 1.4000;:TRIGGER:A:LOG-
IC:INPUT:CH1 HIGH
;CH2 X;CH3 X;:TRIGGER:A:LOGIC:PATTERN:IN-
PUT:CH4 X;
:TRIGGER:A:LOGIC:PATTERN:WHEN
TRUE;WHEN:LESSLIMIT
5.0000E-9;MORELIMIT 5.0000E-9;:TRIG-
GER:A:LOGIC:SETHOLD
:CLOCK:EDGE RISE;THRESHOLD 1.4000;SOURCE
CH2;:TRIGGER
:A:LOGIC:SETHOLD:DATA:THRESHOLD
1.4000;SOURCE CH1;
:TRIGGER:A:LOGIC:SETHOLD:HOLDTIME
2.0000E-9;SETTIME
3.0000E-9;:TRIGGER:A:LOGIC:STATE:INPUT:CH4
RISE;
:TRIGGER:A:LOGIC:STATE:WHEN TRUE;:TRIG-
GER:A:PULSE:CLASS
GLITCH;SOURCE CH1;GLITCH:WIDTH
2.0000E-9;TRIGIF ACCEPT
;POLARITY POSITIVE;:TRIG-
GER:A:PULSE:RUNT:POLARITY
POSITIVE;THRESHOLD:HIGH 1.2000;LOW
800.0000E-3;:TRIGGER
:A:PULSE:RUNT:WHEN OCCURS;WIDTH
2.0000E-9;:TRIGGER:A
:PULSE:TRANSITION:DELTATIME 2.0000E-9;PO-
LARITY POSITIVE
;THRESHOLD:HIGH 1.2000;LOW
800.0000E-3;:TRIGGER:A:PULSE
:TRANSITION:WHEN SLOWERTHAN;:TRIG-
GER:A:PULSE:WIDTH
:LOWLIMIT 2.0000E-9;HIGHLIMIT
2.0000E-9;WHEN WITHIN
;POLARITY POSITIVE;:TRIGGER:A:PULSE:TIME-
OUT:POLARITY
STAYSHIGH;TIME 2.0000E-9

```

TRIGger:A:COMMunication:AMI...PULSEForm

(TDS7000 Series option SM & CSA7000 Series)

Description

This entry covers five separate commands, one each for AMI and AMI subtypes. They set or query the AMI pulse form to one of three possibilities.

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunica-  
tion:(AMI|HDB3|B3ZS|B6ZS|B8ZS):PULSEForm:  
(PLUSOne|MINUSOne|EYEDiagram}
```

Syntax 2

```
TRIGger:A:COMMunica-  
tion:(AMI|HDB3|B3ZS|B6ZS|B8ZS):PULSEForm?
```

Arguments

- PLUSOne

Corresponds to the Isolated +1 on the front panel menu.

- MINUSOne

Corresponds to the Isolated -1 on the front panel menu.

- EYEDiagram

Sets the trigger to construct an Eye Diagram.

Example 1

```
TRIGger:A:COMMUNICATION:AMI:PULSEFORM PLUSONE
```

This commands sets the pulse form to PLUSONE (Isolated +1 on the front menu).

Example 2

```
TRIGger:A:COMMUNICATION:AMI:PULSEFORM?
```

This query might return

```
PLUSONE
```

TRIGger:A:COMMunication:AMI...THReshold:HIGH

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or queries the threshold high level.

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunica-  
tion:(AMI|HDB3|B3ZS|B6ZS|B8ZS):THResh-  
old:HIGH <NR3>
```

Syntax 2

```
TRIGger:A:COMMunica-  
tion:(AMI|HDB3|B3ZS|B6ZS|B8ZS):THResh-  
old:HIGH?
```

Arguments

- <NR3>

The high threshold parameter expressed in volts.

Example 1

```
TRIGger:A:COMMUNICATION:AMI:THRESHOLD:HIGH  
2.8 E-2
```

This command sets the threshold high level to 2.8 E-2.

Example 2

```
TRIGger:A:COMMUNICATION:AMI:THRESH-  
OLD:HIGH?
```

This query might return

```
2.8 E-2
```

indicating that the threshold high level in volts.

TRIGger:A:COMMunication:AMI...THReshold:LOW

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or queries the threshold low level.

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunica-  
tion:(AMI|HDB3|B3ZS|B6ZS|B8ZS):THResh-  
old:LOW <NR3>
```

Syntax 2

```
TRIGger:A:COMMunica-  
tion:(AMI|HDB3|B3ZS|B6ZS|B8ZS):THResh-  
old:LOW?
```

Arguments

- <NR3>

The high threshold parameter expressed in volts.

Example 1

```
TRIGger:A:COMMUNICATION:AMI:THRESHOLD:LOW  
2.8 E-2
```

This command sets the threshold low level to 2.8 E-2.

Example 2

```
TRIGger:A:COMMUNICATION:AMI:THRESHOLD:LOW?
```

This query might return

```
2.8 E-2
```

indicating that the threshold low level in volts.

TRIGger:A:COMMunication:BITRate

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or queries the bit rate.

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunication:BITRate <NR3>
```

Syntax 2

```
TRIGger:A:COMMunication:BITRate?
```

Arguments

- <NR3>

A non-negative number greater than one and expressed as bits per second.

Example 1

```
TRIGger:A:COMMUNICATION:BITRATE 1.544e6
```

This command sets the bit rate to 1.544e6 bits per second.

Example 2

```
TRIGger:A:COMMUNICATION:BITRATE?
```

This query might return

```
1.544e6
```

indicating a bit rate of 1.544e6 bits per second.

TRIGger:A:COMMunication:CMI:PULSEForm

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or queries the CMI pulse form.

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunication:CMI:PULSEForm  
(PLUSOne | MINUSOne | ZERO | EYEDiagram)
```

Syntax 2

```
TRIGger:A:COMMunication:CMI:PULSEForm?
```

Arguments

- PLUSOne

Triggering on a positive mark.

- MINUSOne

Triggering on a negative mark.

- ZERO

Triggering on the bit representing zero.

- EYEDiagram

Example 1

```
TRIGger:A:COMMUNICATION:CMI:PULSEFORM PLUSONE
```

This command sets triggering to occur on a positive mark.

Example 2

```
TRIGger:A:COMMUNICATION:CMI:PULSEFORM?
```

This query might return

```
PLUSONE
```

indicating that the pulse form is PLUSONE.

TRIGger:A:COMMunication:CODe

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or queries the signal code the communications trigger should expect on the incoming signal.

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunication:CODe  
{AMI | HDB3 | B3ZS | B6ZS | B8ZS | CMI | NRZ | MLT3}
```

Syntax 2

```
TRIGger:A:COMMunication:CODe?
```

Arguments

- AMI
- HDB3
- B3ZS
- B6ZS
- B8ZS
- CMI
- NRZ
- MLT3

Example 1

```
TRIGger:A:COMMUNICATION:CODE?
```

This query might return

```
AMI
```

indicating that the code is AMI.

Example 2

```
TRIGger:A:COMMUNICATION:CODE AMI
```

This command selects AMI as the line code the communications trigger expects on the incoming signal.

TRIGger:A:COMMunication:CLOCK:POLarity

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or queries the communication clock polarity

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunication:CLOCK:POLARITY  
(RISE|FALL)
```

Syntax 2

```
TRIGger:A:COMMunication:CLOCK:POLARITY?
```

Arguments

- RISE

Specifies to trigger on the rising or positive edge of a signal.

- FALL

Specifies to trigger on the falling or negative edge of a signal.

Example 1

```
TRIGGER:A:COMMUNICATION:CLOCK:POLARITY?
```

This query might return

```
RISE
```

indicating that the polarity is RISE.

Example 2

```
TRIGGER:A:COMMUNICATION:CLOCK:POLARITY  
RISE
```

This command sets RISE as the clock polarity.

TRIGger:A:COMMunication:SOURce

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or queries the source channel.

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunication:SOURce  
{CH1 | CH2 | CH3 | CH4}
```

Syntax 2

```
TRIGger:A:COMMunication:SOURce?
```

Arguments

- CH1
- CH2
- CH3
- CH4

Example 1

```
TRIGger:A:COMMUNICATION:SOURCE?
```

This query might return

```
CH1
```

indicating that the source is channel 1.

Example 2

```
TRIGger:A:COMM:SOURCE CH1
```

This command sets channel 1 as the source channel.

TRIGger:A:COMMunication:SOURce:TYPe

(TDS7000 Series option SM & CSA7000 Series)

Description

This command sets or queries the source type. This command works only when the Eye Diagram pulseform is selected.

TRIGger:MAIN:COMMunication:SOURce:TYPe works in an identical fashion to this command and provides compatibility with previous Tektronix oscilloscopes.

Group

Trigger

Syntax 1

```
TRIGger:A:COMMunication:SOURce:TYPe  
{DATA | CLOCK | RECOVERED}
```

Syntax 2

```
TRIGger:A:COMMunication:SOURce:TYPe?
```

Arguments

■ DATA

The oscilloscope triggers and shifts five unit intervals to form the expected eye pattern.

■ CLOCK

The oscilloscope triggers but no shift occurs. The clock type causes random triggers with respect to the data channel which must be one of the other three channels.

■ RECOVERED

The oscilloscope recovers a clock from the data signal on the source using a phase locked loop. Triggers based on the recovered clock are random with respect to the data so the waveform is not shifted.

Example 1

```
TRIGger:A:COMMUNICATION:SOURCE:TYPe?
```

This query might return

```
DATA
```

indicating that the source type is data.

TRIGger:A:COMMunication:STANdard

(TDS7000 Series option SM & CSA7000B Series)

Description

This command sets or queries the standard that identifies the code and bit rate. The bit rate is used to compute the Unit Interval, which is the inverse of the bit rate. The Unit Interval influences time skew in an Eye Diagram, where you do post processing on AMI isolated pulses, and pulse width settings if CMI.

Group

Trigger

Related Commands

`MASK:STANdard`

Syntax 1

```
TRIGger:A:COMMunication:STANdard
{DS1 | DS1A | DS1C | DS2 | DS2RATESymmetrical
 | DS2RATECoax | RATE32Mbit | RATE97Mbit
 | DS3 | DS4NA | E1 | E2 | E3 | E4 | STS1 | STS3 | STM1E
 | GB_Ethernet | FC133 | FC266 | FC531 | FC1063
 | FC2125 | OC1 | OC3 | OC12 | OC48 | OC48_FEC | STM0
 | STM1 | STM4 | STM16 | INF2_5G | 100Base_Tx
 | ENET100 | ENET100FX | ENET1250 | INF2_5G | ATAG1
 | ATAG2 | ATAG3 | FST | HST | FW1394BS400B
 | FW1394BS800B | FW1394BS1600B | ENETXAUI
 | Custom | RIO_500M | RIO_750M | RIO_1G | RIO_1_5G
 | RIO_2G | RIO_SERIAL_1G | RIO_SERIAL_2G
 | RIO_SERIAL_3G | SFI5_2 | SFI5_3 | VSROC192
 | TFI5_2 | TFI5_3 | PCIExpress}
```

Syntax 2

```
TRIGger:A:COMMunication:STANdard?
```

Arguments

All of the standards may not be available on every instrument.

- 2.5 IBand: NRZ, 2.5 Gb/s
- 100Base-Tx: MLT3, 100 Mb/s, Ethernet
IEEE Std 802.3, ANSI X3.263 1995
- ATAG1: NRZ, 1.5 Gb/s, Serial ATA
- ATAG2: NRZ, 3.0Gb/s, Serial ATA

-
- ATAG3: NRZ, 6.0 Gb/s, Serial ATA
 - Custom: AMI, B3ZS, B6ZS, B8ZS, CMI, HDB3, MLT3, NRZ, 1.544 Mb/s
 - DS1: AMI, B8ZS, 1.544 Mb/s, ANSI T1.102 1993 (R1999)
 - DS1A: AMI, HDB3, 2.048 Mb/s, ANSI T1.102 1993 (R1999)
 - DS1C: AMI, B8ZS, 3.152 Mb/s, ANSI T2.102 1993 (R1999)
 - DS2: AMI, B6ZS, 6.312 Mb/s, ANSI T2.102 1993 (R1999)
 - DS2RATE Coax: AMI, B8ZS, 6.321 Mb/s, ITU-T G.703 (10/98)
 - DS2RATE Symmetrical: AMI, B6ZS, 6.321 Mb/s, ITU-T G.703 (10/98)
 - DS3: AMI, B3ZS, 44.736 Mb/s, ANSI T1.102 1993 (R1999)
 - DS4NA: CMI, 139.3 Mb/s, ANSI T1.102 1993 (R1999)
 - E1: AMI, HDB3, 2.048 Mb/s, ITU-T G.703 (10/98)
 - E2: AMI, HDB3, 8.448 Mb/s, ITU-T G.703 (10/98)
 - E3: AMI, HDB3, 34.368 Mb/s, ITU-T G.703 (10/98)
 - E4: CMI, 139.26 Mb/s, ITU-T G.703 (10/98)
 - EFM: NRZ, 1.25 Gb/s
 - ENET100: MLT3, 100 Mb/s, Ethernet IEEE Std 802.3, ANSI X3.263 1995
 - ENET100FX: MLT3, 100 Mb/s, Ethernet IEEE Std 802.3
 - ENET1250: NRZ, 1.25 Gb/s, Ethernet IEEE Std 802.3, ANSI X3.263 1995
 - ENETXAUI: NRZ, 3.125 Gb/s, 10 Gigabit Ethernet
 - FC133: NRZ, 132.8 Mb/s, Fibre Channel
 - FC266: NRZ, 265.6 Mb/s, Fibre Channel
 - FC531: NRZ, 531.2 Mb/s, Fibre Channel
 - FC1063: NRZ, 1.062.5 Gb/s, Fibre Channel
 - FC2125: NRZ, 2.125 Gb/s, Fibre Channel

-
- FC4250: NRZ, 4.256 Gb/s
 - FST: NRZ, 12.0 Mb/s, USB 1.1 and 2.0
 - FS USB: NRZ, 12.0 Mb/s
 - FW1394 BS400B: NRZ, 491.5 Mb/s, IEEE1394b
 - FW1394 BS800B: NRZ, 983.0 Mb/s, IEEE1394b
 - FW1394 BS1600B: NRZ, 1.966 Gb/s, IEEE1394b
 - G1ATA: NRZ, 1.5 Gb/s
 - G2ATA: NRZ, 3.0 Gb/s
 - G3ATA: NRZ, 6.0 Gb/s
 - GB Ethernet: NRZ, 1.25 Gb/s, Ethernet IEEE Std 802.3
 - HST: NRZ, 480.0 Mb/s, USB 1.1 and 2.0
 - HS USB: NRZ, 480 Mb/s
 - INF2_5G: NRZ, 2.5 Gb/s, Infiniband
 - OC1: NRZ, 51.84 Mb/s, SONET/SDH GR 253 CORE Issue 3 (9/21/02)
 - OC3: NRZ, 155.52 Mb/s, SONET/SDH GR 253 CORE Issue 3 (9/21/02)
 - OC12: NRZ, 622.08 Mb/s, SONET/SDH GR 253 CORE Issue 3 (9/21/02)
 - OC48_FEC: NRZ, 2.666 Gb/s
 - PCI-Express: NRZ, 2.5 Gb/s
 - RATE32 Mbit: AMI, 32.06 Mb/s, ITU-T G.703 (10/98)
 - RATE97 Mbit: AMI, 97.73 Mb/s, ITU-T G.703 (10/98)
 - RIO_500M: NRZ, 500 Mb/s
 - RIO_750M: NRZ, 750 Mb/s
 - RIO_1.0G: NRZ, 1.0 Gb/s
 - RIO_1.5G: NRZ, 1.56 Gb/s
 - RIO_2.0G: NRZ, 2.0 Gb/s
 - RIO_SERIAL 1G: NRZ, 1.25 Gb/s
 - RIO_SERIAL 2.0G: NRZ, 2.5 Gb/s
 - RIO_SERIAL 3.0G: NRZ, 3.125 Gb/s
 - S400B: NRZ, 491.5 Mb/s
 - S800B: NRZ, 983.0 Mb/s

-
- S1600B: NRZ, 1.966 Gb/s
 - SFI/SP15 2.5G: NRZ, 2.488 Gb/s
 - SFI/SP15 3.1G: NRZ, 3.125 Gb/s
 - STM-0: CMI, HDB3, NRZ, 51.84 Mb/s, SONET/SDH GR 253 CORE Issue 3 (9/21/02)
 - STM-1: NRZ, 155.52 Mb/s, SONET/SDH GR 253 CORE Issue 3 (9/21/02)
 - STM-4: NRZ, 622.08 Mb/s, SONET/SDH GR 253 CORE Issue 3 (9/21/02)
 - STM-16: NRZ, 2.488 Gb/s, SONET/SDH GR 253 CORE Issue 3 (9/21/02)
 - STM-1E: CMI, 155.52 Mb/s, ITU-T G.703 (10/98)
 - STS-1: AMI, B3ZS, 51.84 Mb/s, ANSI T1.102 1993 (R1999)
 - STS-3: CMI, 155.52 Mb/s, ANSI T1.102 1993 (R1999)
 - TFI-5 2.5G: NRZ, 2.488 Gb/s
 - TFI-5 3.1G: NRZ, 3.1104 Gb/s
 - VSROC192: NRZ, 1.2441 Gb/s

Example 1

TRIGger:A:COMMUNICATION:STANDARD E4

This command sets the standard to E4.

Example 2

TRIGger:A:COMMUNICATION:STANDARD?

This query might return

DS1.

TRIGger:A:EDGE?

Description

This query-only command returns the trigger source, coupling and slope for the A edge trigger. This command is equivalent to selecting Edge Setup from the Trig menu and then viewing the current setups.

Group

Trigger

Related Commands

TRIGger:A:PULse, TRIGger:A:LOGic

Syntax

TRIGger:A:EDGE?

Example

TRIGger:A:EDGE?

This query might return :TRIGGER:A:EDGE:SOURCE CH1;COUPLING DC;SLOPE RISE, indicating the trigger source, coupling and slope for the A edge trigger.

TRIGger:A:EDGE:COUPLing

Description

This command sets or queries the type of coupling for the A edge trigger. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then choosing from the Coupling drop-down list.

Group

Trigger

Related Commands

TRIGger:A:EDGE:SOUrce, TRIGger:A:EDGE:SLOpe

Syntax 1

```
TRIGger:A:EDGE:COUPLing  
{AC | DC | HFRej | LFrej | NOISErej}
```

Syntax 2

```
TRIGger:A:EDGE:COUPLing?
```

Arguments

- AC

This selects AC trigger coupling.

- DC

This selects DC trigger coupling.

- HFRej

This coupling removes the high frequency components of the DC signal.

- LFrej

This coupling removes the low frequency components of the AC signal.

- NOISErej

This selects DC low sensitivity. It requires added signal amplitude for more stability and less false triggering.

Example 1

```
TRIGger:A:EDGE:COUPLing DC
```

This command sets the A edge trigger coupling to DC.

Example 2

TRIGger:A:EDGE:COUPLing?

This query might return

:TRIGGER:A:EDGE:COUPLING DC

indicating that the A edge trigger coupling is set to DC.

TRIGger:A:EDGE:SLOpe

Description

This command sets or queries the slope for the A edge trigger. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then choosing the desired Slope.

Group

Trigger

Related Commands

TRIGger:A:EDGE:SOUrce, TRIGger:A:EDGE:COUPling

Syntax 1

```
TRIGger:A:EDGE:SLOpe {RISe|FALL}
```

Syntax 2

```
TRIGger:A:EDGE:SLOpe?
```

Arguments

- RISe

This specifies to trigger on the rising or positive edge of a signal.

- FALL

This specifies to trigger on the falling or negative edge of a signal.

Example 1

```
TRIGger:A:EDGE:SLOpe?
```

This query might return :TRIGGER:A:EDGE:SLOPE FALL, indicating that the A edge trigger slope is negative.

Example 2

```
TRIGger:A:EDGE:SLOpe RISe
```

This command sets the A edge trigger slope to positive, which triggers on the rising edge of the signal.

TRIGger:A:EDGE:SOURce

Description

This command sets or queries the source for the A edge trigger. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then choosing from the Source drop-down list.

Group

Trigger

Related Commands

TRIGger:A:EDGE:SLOpe, TRIGger:A:EDGE:COUPling

Syntax 1

```
TRIGger:A:EDGE:SOURce {AUXilia-  
ry|CH<x>|LINE}
```

Syntax 2

```
TRIGger:A:EDGE:SOURce?
```

Arguments

- AUXiliary

This specifies an external trigger using the Auxiliary Trigger Input connector located on the rear panel of the oscilloscope.

- CH<x>

This specifies one input channel as the A edge trigger source.

- LINE

This specifies AC line voltage.

Example 1

```
TRIGger:A:EDGE:SOURce?
```

This query might return :TRIGGER:A:EDGE:SOURCE CH1, indicating that channel 1 is the A edge trigger source.

Example 2

```
TRIGger:A:EDGE:SOURce CH1
```

This command specifies channel 1 as the A edge trigger source.

TRIGger:A:HOLDoff?

Description

This query-only command returns the A trigger holdoff parameters. These parameters specify the time period during which the trigger circuitry is not looking to generate a trigger event. This command is equivalent to selecting Holdoff from the Trig menu and then viewing the current settings.

Group

Trigger

Related Commands

TRIGger:A:HOLDoff:ACTUal?, TRIGger:A:HOLDoff:BY, TRIGger:A:HOLDoff:TIME

Syntax

```
TRIGger:A:HOLDoff?
```

Example

```
TRIGger:A:HOLDoff?
```

This query might return `:TRIGGER:A:HOLDOFF:TIME 900.0000E-09;BY DEFAULT`, indicating that the A edge trigger holdoff time is set 900 ns.

TRIGger:A:HOLDoff:ACTUal?

Description

This query-only command returns the holdoff time actually used (expressed in seconds) by the A trigger. This command is equivalent to selecting Holdoff from the Trig menu and then viewing the current Trig Holdoff value.

Group

Trigger

Related Commands

TRIGger:A:HOLDoff?, TRIGger:A:HOLDoff:BY, TRIGger:A:HOLDoff:TIME

Syntax

TRIGger:A:HOLDoff:ACTUal?

Example

TRIGger:A:HOLDoff:ACTUal?

This query might return :TRIGGER:A:HOLDOFF:ACTUAL
4.0000E-06, showing that the holdoff time is set to 4 μ s.

TRIGger:A:HOLDoff:BY

Description

This command sets or queries the type of holdoff for the A trigger. Holdoff types are expressed as either user-specified time (TIME) or by an internally calculated minimum time value (DEFAULT/AUTO). This command is equivalent to selecting Holdoff from the Trig menu and then setting the Holdoff type.

Note: Arguments listed for this command are not available for all TDS instruments. Review the argument descriptions carefully to ensure that you are specifying arguments that are valid for the instrument that you are using.

Group

Trigger

Related Commands

TRIGger:A:HOLDoff:TIME

Syntax 1

```
TRIGger:A:HOLDoff:BY {TIME | DEFAULT}
```

Syntax 2

```
TRIGger:A:HOLDoff:BY {TIME | RANDOM | AUTO}
```

Syntax 3

```
TRIGger:A:HOLDoff:BY?
```

Arguments

- TIME

This enables you to set the holdoff time via the TRIGger:A:HOLDoff:TIME command.

- DEFAULT

This works similarly to the AUTO argument described below.

- AUTO

This automatically calculates a holdoff time to use. This time is typically equivalent to the greater of 1/2 screen (5 divisions) of time or 250 ns. The maximum value is 12 s. For example, if the oscilloscope is set to 1 ms/division then the default holdoff will be 1 ms/division x 5 divisions = 5 ms.

- RANDOM

This generates a random holdoff time.

Example 1

```
TRIGger:A:HOLDOff:BY?
```

This query might return :TRIGGER:A:HOLDOFF:BY TIME, indicating that you will set the holdoff time.

Example 2

```
TRIGger:A:HOLDOff:BY TIME
```

This command sets the holdoff to the "by time" setting. This enables you to set the holdoff time.

TRIGger:A:HOLDoff:TIME

Description

This command sets or queries the A trigger holdoff time. This command is equivalent to selecting Holdoff from the Trig menu and then choosing the desired Trig Holdoff.

Group

Trigger

Related Commands

TRIGger:A:HOLDoff:BY

Syntax 1

```
TRIGger:A:HOLDoff:TIME <NR3>
```

Syntax 2

```
TRIGger:A:HOLDoff:TIME?
```

Argument

- <NR3>

This specifies the holdoff time in seconds. The range is from 250 ns through 12.0 s.

Example 1

```
TRIGger:A:HOLDoff:TIME?
```

This query might return `:TRIGGER:A:HOLDOFFTIME 1.2000E-06`, indicating that the A trigger holdoff time is set to 1.2 μ s.

Example 2

```
TRIGger:A:HOLDoff:TIME 10
```

This command sets the trigger A holdoff time to 10 s.

TRIGger:A:LEVel

Description

This command sets or queries the level for the A trigger. This command is equivalent to selecting Holdoff from the Trig menu and then viewing or setting the trigger Level.

Group

Trigger

Syntax 1

```
TRIGger:A:LEVel {ECL | TTL | <NR3>}
```

Syntax 2

```
TRIGger:A:LEVel?
```

Arguments

- ECL

This specifies the ECL high level.

- TTL

This specifies the TTL high level.

- <NR3>

This specifies the trigger level in user units (usually volts).

Example 1

```
TRIGger:A:LEVel?
```

This query might return :TRIGGER:A:LEVel
1.3000E+00, indicating that the A edge trigger is set to 1.3 V.

Example 2

```
TRIGger:A:LEVel TTL
```

This command sets the A edge trigger to TTL high level, which is 1.4 V.

TRIGger:A:LEVel:CH<x>

Description

This command sets or queries the CH<x> A trigger level for TRIGger:LVLsrcpreference SRCDependent mode.

The CH<x> range is 1 to 4.

Group

Trigger

Syntax 1

```
TRIGger:A:LEVel:CH<x> {<NR3>|ECL|TTL}
```

Syntax 2

```
TRIGger:A:LEVel:CH<x>?
```

Arguments

- <NR3>

This specifies the trigger level in user units (usually volts).

- ECL

This specifies the ECL high level.

- TTL

This specifies the TTL high level.

Example 1

```
TRIGger:A:LEVel:CH<x>?
```

This query might return

```
:TRIGGER:A:LEVEL:CH2 1.3000E+00
```

indicating that the A trigger level for channel 2 is set to 1.3 V

Example 2

```
TRIGger:A:LEVel:CH1 TTL
```

This command sets the A trigger level for channel 1 to TTL high level, which is 1.4 V.

TRIGger:A:LOGic?

Description

This query-only command returns all of the A logic trigger parameters.

Group

Trigger

Related Commands

TRIGger:A:LOGic:CLass

Syntax

TRIGger:A:LOGic?

Example

TRIGger:A:LOGic?

```
This query might return :TRIGGER:A:LOGIC:CLASS
PATTERN
;FUNCTION AND;WHEN TRUE;THRESHOLD:CH1
1.4000;CH2 1.4000
;CH3 1.4000;CH4 1.4000;:TRIGGER:A:LOGIC:IN-
PUT:CH1 HIGH
;CH2 X;CH3 X;:TRIGGER:A:LOGIC:PATTERN:IN-
PUT:CH4 X;
:TRIGGER:A:LOGIC:PATTERN:WHEN
TRUE;WHEN:LESSLIMIT
5.0000E-9;MORELIMIT 5.0000E-9;:TRIG-
GER:A:LOGIC:SETHOLD
:CLOCK:EDGE RISE;THRESHOLD 1.4000;SOURCE
CH2;:TRIGGER:A
:LOGIC:SETHOLD:DATA:THRESHOLD 1.4000;SOURCE
CH1;
:TRIGGER:A:LOGIC:SETHOLD:HOLDTIME
2.0000E-9;SETTIME
3.0000E-9;:TRIGGER:A:LOGIC:STATE:INPUT:CH4
RISE;:TRIGGER
:A:LOGIC:STATE:WHEN TRUE
```

TRIGger:A:LOGic:CLAss

Description

This command sets or queries the class of the Logic Trigger. Used in concert with TRIGger:A:TYPe, this command is equivalent to selecting Logic Pattern, Logic State or Setup/Hold Setup from the Trig menu.

Group

Trigger

Related Commands

TRIGger:A:TYPe, TRIGger:A:PULse:CLAss

Syntax 1

```
TRIGger:A:LOGic:CLAss {PATtern|STATE|SE-  
THold}
```

Syntax 2

```
TRIGger:A:LOGic:CLAss?
```

Arguments

■ PATtern

This means that the oscilloscope triggers when the specified logical combinations of channels 1, 2, 3 and 4 are met.

■ STATE

This means that the oscilloscope triggers when the specified conditions of channels 1, 2, and 3 are met after the channel 4 condition is met.

■ SETHold

This means that the oscilloscope will trigger on the setup and hold violations between a data source and a clock source.

Example 1

```
TRIGger:A:LOGic:CLAss?
```

This query might return

```
:TRIGGER:A:LOGIC:CLASS PATTERN
```

Example 2

```
TRIGger:A:LOGic:CLAss PATTERN
```

This command sets the trigger A logic class to PATtern, which causes the oscilloscope to trigger when the specified logical combinations of channels 1, 2, 3, and 4 are met.

TRIGger:A:LOGIc:FUNcTION

Description

This command sets or queries the logical combination of the input channels for the A pattern and A state logic triggers. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then setting or viewing the Pattern Type.

Group

Trigger

Related Commands

TRIGger:A:LOGIc:PATtern:WHEn, TRIGger:A:LOGIc:INPut:CH<x>, TRIGger:A:LOGIc:THReshold:CH<x>

Syntax 1

```
TRIGger:A:LOGIc:FUNcTION {AND|NANd|NOR|OR}
```

Syntax 2

```
TRIGger:A:LOGIc:FUNcTION?
```

Arguments

- AND

This specifies to trigger if all conditions are true.

- NANd

This specifies to trigger if any of the conditions are false.

- NOR

This specifies to trigger if all conditions are false.

- OR

This specifies to trigger if any of the conditions are true.

Example 1

```
TRIGger:A:LOGIc:FUNcTION?
```

This query might return :TRIGGER:A:LOGIC:FUNCTION
NAND

Example 2

```
TRIGger:A:LOGIc:FUNcTION AND
```

This command sets the logical combination of channels to be true when all conditions are true.

TRIGger:A:LOGic:INPut?

Description

This query-only command returns the A logic trigger input expected for channel 1, 2, and 3. Channel 4 is set or queried with the command TRIGger:A:LOGic:PAtern:INPut:CH4. This command is equivalent to selecting A Event (Main) Trigger Setup and then viewing or setting the Input Threshold for channel 1 through 3.

Group

Trigger

Related Commands

TRIGger:A:LOGic:PAtern:INPut:CH4

Syntax

TRIGger:A:LOGic:INPut?

Example

TRIGger:A:LOGic:INPut?

This query might return :TRIGGER:A:LOGIC:INPUT:CH1
HIGH;CH2 X;CH3 X

TRIGger:A:LOGic:INPut:CH<x>

Description

This command sets or queries the A logical input for the specified logic trigger channel, which ranges from CH1 through CH3. Note that CH4 cannot be set or queried with this command. For details about setting this channel, see TRIGger:A:LOGic:PATtern:INPut:CH4. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then choosing the desired logical input from the Ch<x> drop-down list, which is located in the Input Threshold group box.

Group

Trigger

Related Commands

TRIGger:A:LOGic:PATtern:INPut:CH4

Syntax 1

TRIGger:A:LOGic:INPut:CH<x> {HIGH|LOW|X}

Syntax 2

TRIGger:A:LOGic:INPut:CH<x>?

Arguments

- HIGH

This specifies the logic high.

- LOW

This specifies the logic low.

- X

This specifies a "do not care" state.

Example 1

TRIGger:A:LOGic:INPut:CH1?

This query might return :TRIGGER:LOGIC:INPUT:CH1 X, indicating that the setting for the A logic trigger input to channel 1 does not matter.

Example 2

TRIGger:A:LOGic:INPut:CH2 HIGH

This command sets the A logic trigger input to logic HIGH for channel 2.

TRIGger:A:LOGic:PATtern?

Description

This query-only command returns the conditions used for generating an A logic pattern trigger, with respect to the defined input pattern, and identifies the maximum and minimum time that the selected pattern may be true and still generate the trigger. This command is equivalent to selecting Logic Pattern from the Trig menu and then viewing the current setups.

Group

Trigger

Related Commands

TRIGger:A:LOGic:PATtern:INPut:CH4, TRIGger:A:LOGic:PATtern:WHEn, TRIGger:A:LOGic:PATtern:WHEn:LESSLimit, TRIGger:A:LOGic:PATtern:WHEn:MORELimit

Syntax

```
TRIGger:A:LOGic:PATtern?
```

Example

```
TRIGger:A:LOGic:PATtern?
```

This query might return :TRIGGER:A:LOGIC:PATTERN:INPUT:CH4 X;:TRIGGER:A:LOGIC:PATTERN:WHEN TRUE;WHEN:LESSLIMIT 5.0000E-9;MORELIMIT 5.0000E-9.

TRIGger:A:LOGIc:PATtern:INPut:CH4

Description

This command sets or queries the A logic trigger input for channel 4. This command specifies the logic value used when the pattern trigger detects the threshold level. This command is equivalent to selecting Logic Pattern from the Trig menu and then choosing the desired logical input from the Ch4 drop-down list, which is located in the Input Threshold group box.

Group

Trigger

Related Commands

TRIGger:A:LOGIc:FUNcTION, TRIGger:A:LOGIc:INPut:CH<x>, TRIGger:A:LOGIc:THReshold:CH<x>

Syntax 1

```
TRIGger:A:LOGIc:PATtern:INPut:CH4  
{HIGH|LOW|X}
```

Syntax 2

```
TRIGger:A:LOGIc:PATtern:INPut:CH4?
```

Arguments

- HIGH

This specifies the logic high.

- LOW

This specifies the logic low.

- X

This specifies a "do not care" state.

Example 1

```
TRIGger:A:LOGIc:PATtern:INPut:CH4?
```

This query might return `:TRIGGER:A:LOGIC:PATTERN:INPUT:CH4 HIGH`, indicating that the logic input for channel 4 is logic high.

Example 2

```
TRIGger:A:LOGic:PATtern:INPut:CH4 HIGH
```

This command sets the A logic trigger input to logic high for channel 4 when the logic class is set to PATtern. When the threshold level is detected, High places a 1 on the channel 4 input to the selected function.

TRIGger:A:LOGIc:PATtern:WHEn

Description

This command sets or queries the condition for generating an A logic pattern trigger with respect to the defined input pattern. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then choosing a trigger condition from the Pattern drop-down list, which is located in the Trigger When group box.

Group

Trigger

Related Commands

TRIGger:A:LOGIc:FUNcTION, TRIGger:A:LOGIc:INPut, TRIGger:A:LOGIc:THReshold

Syntax 1

```
TRIGger:A:LOGIc:PATtern:WHEn  
{TRUE | FALSE | LESSThan | MOREThan}
```

Syntax 2

```
TRIGger:A:LOGIc:PATtern:WHEn?
```

Arguments

- TRUE

This specifies to trigger when the pattern becomes true.

- FALSE

This specifies to trigger when the pattern becomes false.

- LESSThan

This specifies to trigger if the specific pattern is true less than the specified LESSLimit.

- MOREThan

This specifies to trigger if the specific pattern is true longer than the specified MORELimit.

Example 1

```
TRIGger:A:LOGIc:PATtern:WHEn?
```

This query might return `:TRIGGER:A:LOGIC:PATTERN:WHEN TRUE`, indicating that the A logic pattern will trigger when the pattern becomes true.

Example 2

`TRIGger:A:LOGic:PATtern:WHEn FALSE`

This command specifies to trigger the A logic pattern when the pattern becomes false.

TRIGger:A:LOGic:PATtern:WHEn:LESSLimit

Description

This command sets or queries the maximum time that the selected pattern may be true and still generate an A logic pattern trigger. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then entering the following Trigger When settings: choose Less Than from the Pattern drop-down list, then enter a maximum value for Time.

Group

Trigger

Related Commands

TRIGger:A:LOGic:PATtern:WHEn:MORELimit

Syntax 1

```
TRIGger:A:LOGic:PATtern:WHEn:LESSLimit  
<NR3>
```

Syntax 2

```
TRIGger:A:LOGic:PATtern:WHEn:LESSLimit?
```

Argument

- <NR3>

This specifies the amount of time to hold pattern true.

Example 1

```
TRIGger:A:LOGic:PATtern:WHEn:LESSLimit?
```

This query might return `:TRIGGER:A:LOGIC:PATTERN:WHEN:LESSLIMIT 5.0000E-09`, indicating that the selected pattern may hold true for up to 5 ns and still generate an A logic pattern trigger.

Example 2

```
TRIGger:A:LOGic:PATtern:WHEn:LESSLimit  
10.0E+00
```

This command sets the maximum time that the selected pattern may hold true (and generate an A logic pattern trigger) to 10 s.

TRIGger:A:LOGic:PATtern:WHEn:MORELimit

Description

This command sets or queries the minimum time that the selected pattern may be true and still generate an A logic pattern trigger. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then entering the following Trigger When settings: choose More Than from the Pattern drop-down list, then enter a minimum value for Time.

Group

Trigger

Related Commands

TRIGger:A:LOGic:PATtern:WHEn:LESSLimit

Syntax 1

```
TRIGger:A:LOGic:PATtern:WHEn:MORELimit  
<NR3>
```

Syntax 2

```
TRIGger:A:LOGic:PATtern:WHEn:MORELimit?
```

Argument

- <NR3>

This specifies the amount of time to hold pattern true.

Example 1

```
TRIGger:A:LOGic:PATtern:WHEn:MORELimit?
```

This query might return `:TRIGGER:A:LOGIC:PATTERN:WHEN:MORELIMIT 5.0000E-09`, indicating that the selected pattern must hold true for at least 5 ns to generate an A logic pattern trigger.

Example 2

```
TRIGger:A:LOGic:PATtern:WHEn:MORELimit  
10.0E+00
```

This command sets the minimum time that the selected pattern may hold true (and generate an A logic pattern trigger) to 10 s.

TRIGger:A:LOGic:SETHold?

Description

This query-only command returns the clock edge polarity, voltage threshold and source input; data voltage threshold and source; and both setup and hold times for setup and hold violation triggering. This command is equivalent to selecting Setup/Hold Setup from the Trig menu and then viewing the current setups.

Group

Trigger

Related Commands

TRIGger:A:LOGic:PATtern?, TRIGger:A:LOGic:STATE?

Syntax

```
TRIGger:A:LOGic:SETHold?
```

Example

```
TRIGger:A:LOGic:SETHold?
```

```
This query might return :TRIGGER:A:LOGIC:SE-  
THOLD:CLOCK:EDGE RISE  
;THRESHOLD 1.4000;SOURCE CH2;:TRIG-  
GER:A:LOGIC:SETHOLD:DATA  
:THRESHOLD 1.4000;SOURCE CH1;:TRIG-  
GER:A:LOGIC:SETHOLD  
:HOLDTIME 2.0000E-9;SETTIME 3.0000E-9
```

TRIGger:A:LOGic:SETHold:CLOCK?

Description

This query-only command returns the clock edge polarity, voltage threshold and source input for setup and hold triggering. This command is equivalent to selecting Setup/ Hold Setup from the Trig menu and then viewing the current clock setups.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:DATA?, TRIGger:A:LOGic:SETHold:CLOCK:EDGE, TRIGger:A:LOGic:SETHold:CLOCK:THReshold, TRIGger:A:LOGic:SETHold:CLOCK:SOURce

Syntax

```
TRIGger:A:LOGic:SETHold:CLOCK?
```

Example

```
TRIGger:A:LOGic:SETHold:CLOCK?
```

This query might return `:TRIGGER:A:LOGic:SETHold:CLOCK:EDGE RISE;THRESHOLD 1.4000;SOURCE CH2`, indicating the current clock settings for setup and hold triggering.

TRIGger:A:LOGic:SETHold:CLOCK:EDGE

Description

This command sets or queries the clock edge polarity for setup and hold triggering. This is equivalent to selecting Setup/Hold Setup from the Trig menu and then choosing the desired Clock Edge.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:CLOCK:SOURce, TRIGger:A:LOGic:SETHold:CLOCK:THReshold

Syntax 1

```
TRIGger:A:LOGic:SETHold:CLOCK:EDGE  
{FALL|RISe}
```

Syntax 2

```
TRIGger:A:LOGic:SETHold:CLOCK:EDGE?
```

Arguments

- FALL

This specifies polarity as the clock falling edge.

- RISe

This specifies polarity as the clock rising edge.

Example 1

```
TRIGger:A:LOGic:SETHold:CLOCK:EDGE?
```

This query might return `:TRIGGER:A:LOGIC:SETHOLD:CLOCK:EDGE RISE`, indicating that polarity is specified as the clock rising edge.

Example 2

```
TRIGger:A:LOGic:SETHold:CLOCK:EDGE RISE
```

This command specifies the polarity as the clock rising edge.

TRIGger:A:LOGic:SETHold:CLOCK:SOURce

Description

This command sets or queries the clock source for the A logic trigger setup and hold input. The channel is specified by x, which ranges from 1 through 4. This is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then choosing the desired channel from the Clock Source drop-down list.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:CLOCK:EDGE, TRIGger:A:LOGic:SETHold:CLOCK:THReshold

Syntax 1

```
TRIGger:A:LOGic:SETHold:CLOCK:SOURce CH<x>
```

Syntax 2

```
TRIGger:A:LOGic:SETHold:CLOCK:SOURce?
```

Argument

- CH<x>

This specifies the input channel, which ranges from 1 through 4.

Example 1

```
TRIGger:A:LOGic:SETHold:CLOCK:SOURce?
```

This query might return :TRIGGER:A:LOGIC:SETHOLD:CLOCK:SOURCE CH4, indicating that channel 4 is the clock source for the A logic trigger setup and hold input.

Example 2

```
TRIGger:A:LOGic:SETHold:CLOCK:SOURce CH1
```

This command specifies channel 1 as the A logic setup and hold input.

TRIGger:A:LOGic:SETHold:CLOCK:THReshold

Description

This command sets or queries the clock voltage threshold for setup and hold trigger. This command is equivalent to selecting Setup/Hold Setup from the Trig menu and then setting the desired Clock Level.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:CLOCK:EDGE, TRIGger:A:LOGic:SETHold:CLOCK:SOURce

Syntax 1

```
TRIGger:A:LOGic:SETHold:CLOCK:THReshold  
{ECL | TTL | <NR3>}
```

Syntax 2

```
TRIGger:A:LOGic:SETHold:CLOCK:THReshold?
```

Arguments

- ECL

This specifies a preset ECL high level of -1.3 V.

- TTL

This specifies a preset TTL high level of 1.4 V.

- <NR3>

This is the clock level, in volts.

Example 1

```
TRIGger:A:LOGic:SETHold:CLOCK:THReshold?
```

This query might return :TRIGGER:A:LOGIC:SETHOLD:CLOCK:THRESHOLD 1.2000E+00, indicating that the clock threshold for setup and hold trigger is 1.2 V.

Example 2

```
TRIGger:A:LOGic:SETHold:CLOCK:THReshold  
TTL
```

This command specifies the preset TTL value of 1.4 V as the clock threshold for setup and hold trigger.

TRIGger:A:LOGic:SETHold:DATA?

Description

This query-only command returns the voltage threshold and data source for setup and hold trigger. This command is equivalent to selecting Setup/Hold Setup from the Trig menu and then viewing the current data setups.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:CLOCK?

Syntax

TRIGger:A:LOGic:SETHold:DATA?

Example

TRIGger:A:LOGic:SETHold:DATA?

This query might return :TRIGGER:A:LOGIC:SETHOLD:DATA:THRESHOLD 1.4000;SOURCE CH1, indicating the current trigger data settings.

TRIGger:A:LOGic:SETHold:DATA:SOURce

Description

This command sets or queries the data source for setup and hold trigger. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then choosing the desired channel from the Data Source drop-down list.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:DATA:THReshold

Syntax 1

```
TRIGger:A:LOGic:SETHold:DATA:SOURce CH<x>
```

Syntax 2

```
TRIGger:A:LOGic:SETHold:DATA:SOURce?
```

Argument

- CH<x>

This specifies the input channel, which ranges from 1 through 4.

Example 1

```
TRIGger:A:LOGic:SETHold:DATA:SOURce?
```

This query might return

```
:TRIGGER:A:LOGIC:SETHOLD:DATA:SOURCE CH2
```

indicating that channel 2 is the current clock source for set-up and hold trigger.

Example 2

```
TRIGger:A:LOGic:SETHold:DATA:SOURce CH1
```

This command sets channel 1 as the clock source for set-up and hold trigger.

TRIGger:A:LOGic:SETHold:DATA:THReshold

Description

This command sets or queries the data voltage threshold for setup and hold trigger. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then setting the desired Data Level.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:DATA:SOUrce

Syntax 1

```
TRIGger:A:LOGic:SETHold:DATA:THReshold  
{ECL | TTL<NR3>}
```

Syntax 2

```
TRIGger:A:LOGic:SETHold:DATA:THReshold?
```

Arguments

- ECL

This specifies the preset ECL high level of -1.3 V.

- TTL

This specifies the preset TTL high level of 1.4 V.

- <NR3>

This is the setup and hold data level, in V.

Example 1

```
TRIGger:A:LOGic:SETHold:DATA:THReshold?
```

This query might return

```
:TRIGGER:A:LOGIC:SETHOLD:DATA:THRESHOLD  
1.2000E+00
```

indicating that 1.2 V is the current data voltage level for setup and hold trigger.

Example 2

```
TRIGger:A:LOGic:SETHold:DATA:THReshold TTL
```

This command specifies the preset ECL high level of 1.4 V as the current data voltage level for setup and hold trigger.

TRIGger:A:LOGic:SETHold:HOLDTime

Description

This command sets or queries the hold time for setup and hold violation triggering. This command is equivalent to selecting Setup/Hold Setup from the Trig menu and then setting the desired Hold Time.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:SETTime

Syntax 1

```
TRIGger:A:LOGic:SETHold:HOLDTime <NR3>
```

Syntax 2

```
TRIGger:A:LOGic:SETHold:HOLDTime?
```

Argument

- <NR3>

This specifies the hold time setting in seconds. Positive values for hold time occur after the clock edge. Negative values occur before the clock edge.

Example 1

```
TRIGger:A:LOGic:SETHold:HOLDTime?
```

This query might return

```
:TRIGGER:A:LOGIC:SETHOLD:HOLDTIME  
2.0000E-09
```

indicating that the current hold time for setup and hold trigger is 2 ns.

Example 2

```
TRIGger:A:LOGic:SETHold:HOLDTime 3.0E-3
```

This command sets the hold time for A logic setup and hold trigger to 3 ms.

TRIGger:A:LOGic:SETHold:QUALify

Description

This command sets or queries the CH<x> A trigger level for TRIGger:LVLSrcpreference SRCDependent mode.

The CH<x> range is 1 to 4.

Group

Trigger

Syntax 1

```
TRIGger:A:LEVEL:CH<x> {<NR3>|ECL|TTL}
```

Syntax 2

```
TRIGger:A:LEVEL:CH<x>?
```

Arguments

- <NR3>

This specifies the trigger level in user units (usually volts).

- ECL

This specifies the ECL high level.

- TTL

This specifies the TTL high level.

Example 1

```
TRIGger:A:LEVEL:CH<x>?
```

This query might return

```
:TRIGGER:A:LEVEL:CH2 1.3000E+00
```

indicating that the A trigger level for channel 2 is set to 1.3 V.

Example 2

```
TRIGger:A:LEVEL:CH1 TTL
```

This command sets the A trigger level for channel 1 to TTL high level, which is 1.4 V.

TRIGger:A:LOGic:SETHold:SETTime

Description

This command sets or queries the setup time for setup and hold violation triggering. This command is equivalent to selecting Setup/Hold Setup from the Trig menu and then setting the desired Hold Time.

Group

Trigger

Related Commands

TRIGger:A:LOGic:SETHold:HOLDTime

Syntax 1

```
TRIGger:A:LOGic:SETHold:SETTime <NR3>
```

Syntax 2

```
TRIGger:A:LOGic:SETHold:SETTime?
```

Argument

- <NR3>

This specifies the setup time for setup and hold violation triggering.

Example 1

```
TRIGger:A:LOGic:SETHold:SETTime?
```

This query might return

```
:TRIGGER:A:LOGIC:SETHOLD:SETTIME  
2.0000E-09
```

indicating that the current hold time for setup and hold trigger is 2 ns.

Example 2

```
TRIGger:A:LOGic:SETHold:SETTime 3.0E-6
```

This command specifies that the current hold time for setup and hold trigger is 3 μ s.

TRIGger:A:LOGic:STATE?

Description

This query-only command returns the data input and trigger criteria for the A logic trigger. This command is equivalent to selecting Logic State from the Trig menu and then viewing the current logic state settings.

Group

Trigger

Related Commands

TRIGger:A:LOGic:STATE:INPut:CH4, TRIGger:A:LOGic:STATE:WHEn

Syntax

TRIGger:A:LOGic:STATE?

Example

TRIGger:A:LOGic:STATE?

This query might return

```
:TRIGGER:A:LOGIC:STATE:INPUT:CH4 RISE;  
:TRIGGER:A:LOGIC:STATE:WHEN TRUE
```

TRIGger:A:LOGic:STATE:INPut:CH4

Description

This command sets or queries the slope for channel 4 when the logic class is set to STATE. This command is equivalent to selecting Logic State from the Trig menu and then choosing the desired channel input (NEG or POS) from the Ch4 drop-down list.

Group

Trigger

Related Commands

TRIGger:A:LOGic:STATE:WHEn

Syntax 1

```
TRIGger:A:LOGic:STATE:INPut:CH4  
{FALL|RISe}
```

Syntax 2

```
TRIGger:A:LOGic:STATE:INPut:CH4?
```

Arguments

- FALL

This specifies falling edge.

- RISe

This specifies rising edge.

Example 1

```
TRIGger:A:LOGic:STATE:INPut:CH4?
```

This query might return

```
:TRIGGER:A:LOGIC:STATE:INPUT:CH4 RISe
```

indicating that the A logic trigger input for channel 4 is the rising edge.

Example 2

```
TRIGger:A:LOGic:STATE:INPut:CH4 RISe
```

This command specifies that the A logic trigger input for channel 4 is the rising edge.

TRIGger:A:LOGic:STATE:WHEn

Description

This command sets or queries the condition for generating an A logic state trigger. This command is equivalent to selecting Logic State from the Trig menu and then choosing the desired condition from the Trigger When Pattern dropdown list.

Group

Trigger

Related Commands

TRIGger:A:LOGic:INPut:CH<x>, TRIGger:A:LOGic:STATE:INPut:CH4

Syntax 1

```
TRIGger:A:LOGic:STATE:WHEn {TRUE|FALSE}
```

Syntax 2

```
TRIGger:A:LOGic:STATE:WHEn?
```

Arguments

- TRUE

This specifies that the trigger occurs when the clock transition on channel 4 occurs and the pattern of channels 1-3 are at the desired logic input states.

- FALSE

This specifies that the trigger occurs when the desired clock transition on channel 4 occurs and the desired logic input states on channels 1-3 are not found.

Example 1

```
TRIGger:A:LOGic:STATE:WHEn?
```

This query might return

```
:TRIGGER:A:LOGIC:STATE:WHEN FALSE
```

indicating that the logic condition to trigger upon is false.

Example 2

```
TRIGger:A:LOGic:STATE:WHEn TRUE
```

This command specifies that the logic condition to trigger upon is true.

TRIGger:A:LOGic:THReshold?

Description

This query-only command returns the threshold voltage for all channels in A logic trigger. This command query is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then viewing the current Input Threshold voltage settings.

Group

Trigger

Related Commands

TRIGger:A:LOGic:THReshold:CH<x>, TRIGger:A:LOGic:INPut, TRIGger:A:LOGic:INPut:CH<x>

Syntax

```
TRIGger:A:LOGic:THReshold?
```

Example

```
TRIGger:A:LOGic:THReshold?
```

This query might return

```
:TRIGGER:A:LOGIC:THRESHOLD CH1  
24.0000E-03;CH2 1.2000E+00;CH3  
1.2000E+00;CH4 1.2000E+00
```

indicating the threshold voltages for the channels in A logic trigger are as follows: channel 1 = 24 mV; channel 2 = 1.2 V; channel 3 = 1.2 V; channel 4 = 1.2 V.

TRIGger:A:LOGic:THReshold:CH<x>

Description

This command sets or queries the A logic trigger threshold voltage for the channel, specified by x, which ranges from 1 through 4. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then setting the Input Threshold voltage for the desired channel.

Group

Trigger

Related Commands

TRIGger:A:LOGic:INPut?

Syntax 1

TRIGger:A:LOGic:THReshold:CH<x> <NR3>

Syntax 2

TRIGger:A:LOGic:THReshold:CH<x>?

Argument

- <NR3>

This specifies the threshold voltage.

Example 1

TRIGger:A:LOGic:THReshold:CH3?

This query might return

```
:TRIGGER:A:LOGIC:THRESHOLD:CH3 1.2000E+00
```

indicating that the A logic trigger threshold voltage for channel 3 is 1.2 V.

Example 2

TRIGger:A:LOGic:THReshold:CH2 3.0E-3

This command sets the A logic trigger threshold voltage for channel 2 to 3 mV.

TRIGger:A:LOWerthreshold:CH<x>

Description

This command sets or queries the CH<x> lower trigger level for TRIGger:LVLsrcpreference SRCDependent.

The CH<x> range is 1 to 4.

Group

Trigger

Related Commands

TRIGger:LVLsrcpreference, TRIGger:A:UPPerthreshold:CH<x>

Syntax 1

```
TRIGger:A:LOWerthreshold:CH<x>  
{<NR3> | ECL | TTL}
```

Syntax 2

```
TRIGger:A:LOWerthreshold:CH<x>?
```

Arguments

- <NR3>

This specifies the trigger level in user units (usually volts).

- ECL

This specifies the ECL high level.

- TTL

This specifies the TTL high level.

Example 1

```
TRIGger:A::CH<x>?
```

This query might return

```
:TRIGGER:A:LOWERTHRESHOLD:CH2 1.3000E+00
```

indicating that the A trigger level for channel 2 is set to 1.3 V.

Example 2

```
TRIGger:A:LOWerthreshold:CH1 TTL
```

This command sets the A trigger level for channel 1 to TTL low level.

TRIGger:A:MODE

Description

This command sets or queries the A trigger mode. This command is equivalent to selecting Mode from the Trig menu and then choosing the desired Trigger Mode.

Group

Trigger

Related Commands

TRIGger:A:LEVel

Syntax 1

TRIGger:A:MODE {AUTO|NORMAl}

Syntax 2

TRIGger:A:MODE?

Arguments

- AUTO

This generates a trigger if one is not detected within a specified time period.

- NORMAl

This waits for a valid trigger event.

Example 1

TRIGger:A:MODE?

This query might return

```
:TRIGGER:A:MODE NORMAL
```

indicating that a valid trigger event must occur before a trigger is generated.

Example 2

```
TRIGger:A:MODE NORMAL
```

This command specifies that a valid trigger event must occur before a trigger is generated.

TRIGger:A:PULse?

Description

This query-only command returns the A pulse trigger parameters.

Group

Trigger

Related Commands

TRIGger:A:EDGE?, TRIGger:A:LOGic?

Syntax

TRIGger:A:PULse?

Example

TRIGger:A:PULse?

This query might return

```
:TRIGGER:A:PULSE:CLASS GLITCH;SOURCE
CH1;GLITCH:WIDTH 2.0000E-9;TRIGIF AC-
CEPT;POLARITY
POSITIVE;;:TRIGGER:A:PULSE:RUNT:POLARITY
POSITIVE
;THRESHOLD:HIGH 1.2000;LOW
800.0000E-3;;:TRIGGER:A:PULSE
:RUNT:WHEN OCCURS;WIDTH 2.0000E-9;;:TRIG-
GER:A:PULSE:
TRANSITION:DELTATIME 2.0000E-9;POLARITY
POSITIVE
;THRESHOLD:HIGH 1.2000;LOW
800.0000E-3;;:TRIGGER:A
:PULSE:TRANSITION:WHEN SLOWERTHAN;;:TRIG-
GER:A:PULSE:WIDTH
:LOWLIMIT 2.0000E-9;HIGHLIMIT
2.0000E-9;WHEN WITHIN
;POLARITY POSITIVE;;:TRIGGER:A:PULSE:TIME-
OUT:POLARITY
STAYSHIGH;TIME 2.0000E-9
```

TRIGger:A:PULse:CLAss

Description

This command sets or queries the type of pulse on which to trigger. This command is equivalent to selecting A Event (Main) Trigger Setup from the Trig menu and then choosing the desired pulse Trigger Type.

Group

Trigger

Related Commands

TRIGger:A:PULse:GLItch, TRIGger:A:PULse:RUNT, TRIGger:A:PULse:WIDth, TRIGger:A:PULse:TIMEOut, TRIGger:A:PULse:TRANSition?, TRIGger:A:TYPe

Syntax 1

```
TRIGger:A:PULse:CLAss  
{GLItch|RUNT|WIDth|TRANSition|TIMEOut}
```

Syntax 2

```
TRIGger:A:PULse:CLAss?
```

Arguments

■ GLItch

This triggers when a pulse is found that is of the specified polarity and width. These are set with the commands TRIGger:A:PULse:GLITch:POLarity and TRIGger:A:PULse;GLItch:WIDth.

■ RUNT

This triggers when a pulse crosses the first preset voltage threshold but does not cross the second preset threshold before re-crossing the first. The thresholds are set with the TRIGger:A:PULse:RUNT:THReshold:LOW and TRIGger:A:PULse:RUNT:THReshold:HIGh commands.

■ WIDth

This triggers when a pulse is found that has the specified polarity and is either inside or outside the limits as specified by TRIGger:A;PULse:WIDth:LOWLIMIT and TRIGger:A:PULse:WIDth:HIGHLimit. The polarity is selected using the TRIGger:A:PULse:WIDth:POLarity command.

■ TRANSition

This triggers when a pulse crosses both thresholds in the same direction as the specified polarity and the transition

time between the two threshold crossings is greater or less than the specified time delta.

■ **TIMEOut**

This triggers when the pulse train stops in the selected state for longer than the specified time.

Example 1

```
TRIGger:A:PULse:CLAss?
```

This query might return

```
:TRIGGER:A:PULSE:CLASS GLITCH
```

indicating that a pulse was found that is of the specified polarity and width.

Example 2

```
TRIGger:A:PULse:CLAss WIDTHh
```

This command specifies a width pulse for the A trigger.

TRIGger:A:PULse:GLItch?

Description

This query-only command returns the current A glitch pulse trigger parameters. This command query is equivalent to selecting Glitch Setup from the Trig menu and then viewing the current pulse glitch settings.

Group

Trigger

Related Commands

TRIGger:A:PULse:GLItch:POLarity, TRIGger:A:PULse:GLItch:TRIGIF, TRIGger:A:PULse:GLItch:WIDth

Syntax

```
TRIGger:A:PULse:GLItch?
```

Example

```
TRIGger:A:PULse:GLItch?
```

This query might return

```
:TRIGGER:A:PULSE:GLITCH:WIDTH  
2.0000E-09;FILTER ACCEPT;POLARITY POSITIVE
```

TRIGger:A:PULse:GLItch:POLarity

Description

This command sets or queries the polarity for A pulse glitch trigger. This command is equivalent to selecting Glitch Setup from the Trig menu and then choosing the desired Polarity.

Group

Trigger

Related Commands

TRIGger:A:LEVel, TRIGger:A:PULse:GLItch:WIDth

Syntax 1

```
TRIGger:A:PULse:GLItch:POLarity {POSITIVE|NEGATIVE|EITHER}
```

Syntax 2

```
TRIGger:A:PULse:GLItch:POLarity?
```

Arguments

- POSITIVE

This specifies that the oscilloscope will only trigger when the polarity of the glitch is positive.

- NEGATIVE

This specifies that the oscilloscope will only trigger when the polarity of the glitch is negative.

- EITHER

This specifies that the oscilloscope will trigger when the polarity of the glitch is either positive or negative.

Example 1

```
TRIGger:A:PULse:GLItch:POLarity?
```

This query might return

```
:TRIGGER:A:PULSE:GLITCH:POLARITY POSITIVE
```

indicating that the polarity of the glitch must be positive for the trigger to occur.

Example 2

`TRIGger:A:PULse:GLItch:POLarity EITHER`

This command specifies that the polarity of the glitch can be either positive or negative for the trigger to occur.

TRIGger:A:PULse:GLItch:QUALify

Description

This command sets or queries the Glitch Trigger qualification. This is equivalent to selecting Glitch Setup from the Trig menu and selecting Occurs or Logic in the Trigger if Glitch drop-down list box.

Note: If you use the LOGIc argument, you must specify the channel logic conditions.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:GLItch:QUALify {OC-  
Curs|LOGIc}
```

Syntax 2

```
TRIGger:A:PULse:GLItch:QUALify?
```

Arguments

- OCCurs

This argument specifies a trigger if any detectable event occurs.

- LOGIc

This argument specifies a trigger if the individual channel qualifications meet the logic patterns and thresholds set by the TRIGger:A:LOGIc:INPut:CH<x> and TRIGger:A:LOGIc:THReshold:CH<x> commands.

Example 1

```
TRIGger:A:PULse:GLItch:QUALify?
```

indicating that a trigger will occur if the instrument detects any of the glitch qualifications.

Example 2

```
TRIGger:A:PULse:GLItch:QUALify OCCurs
```

This command sets the trigger qualifier to OCCurs, which initiates a trigger if the instrument detects any glitch trigger qualifications.

TRIGger:A:PULse:GLItch:TRIGIF

Description

This command sets or queries the acceptance/rejection of the glitch pulse trigger, based on width. This command is equivalent to selecting Glitch Setup from the Trig menu and then choosing the desired Trig if Width setting.

Group

Trigger

Related Commands

TRIGger:A:PULse:GLItch:WIDth

Syntax 1

```
TRIGger:A:PULse:GLItch:TRIGIF {ACCEpt|RE-  
Ject}
```

Syntax 2

```
TRIGger:A:PULse:GLItch:TRIGIF?
```

Arguments

- ACCEpt

This specifies that the oscilloscope will only trigger on pulses that are narrower than the specified width, when the trigger type is set to pulse glitch. The width is specified using the TRIGger:A:PULse:GLItch:WIDth command.

- REJect

This specifies that the oscilloscope will only trigger on pulses that are wider than the specified width, when the trigger type is set to pulse glitch. The width is specified using the TRIGger:A:PULse:GLItch:WIDth command.

Example 1

```
TRIGger:A:PULse:GLItch:TRIGIF?
```

This query might return

```
:TRIGGER:A:PULSE:GLITCH:TRIGIF ACCEPT
```

indicating that the oscilloscope is set to trigger on pulses that are narrower than the specified width.

Example 2

```
TRIGger:A:PULse:GLItch:TRIGIF REJect
```

This command specifies that the oscilloscope triggers on pulses that are wider than the specified width.

TRIGger:A:PULse:GLItch:WIDth

Description

This command sets or queries the width for the A pulse glitch trigger. This command is equivalent to selecting Glitch Setup from the Trig menu and then setting the desired Width.

For information about using the width value, refer to the TRIGger:A:PULse:GLItch:TRIGIF command.

Group

Trigger

Related Commands

TRIGger:A:LEVel, TRIGger:A:PULse:GLItch:TRIGIF

Syntax 1

```
TRIGger:A:PULse:GLItch:WIDth <NR3>
```

Syntax 2

```
TRIGger:A:PULse:GLItch:WIDth?
```

Argument

- <NR3>

This specifies the width of the glitch, in seconds.

Example 1

```
TRIGger:A:PULse:GLItch:WIDth?
```

This query might return

```
:TRIGGER:A:PULSE:GLITCH:WIDTH 2.0000E-09
```

indicating that the width of the glitch is currently set at 2 ns.

Example 2

```
TRIGger:A:PULse:GLItch:WIDth 15E-6
```

This command sets the width of the glitch to 15 μ s.

TRIGger:A:PULse:RUNT?

Description

This query-only command returns the current A runt pulse trigger parameters. This command query is equivalent to selecting Runt Setup from the Trig menu and then viewing the current settings.

Group

Trigger

Related Commands

TRIGger:A:PULse:GLItch?, TRIGger:A:PULse:TIMEOut?, TRIGger:A:PULse:TRANSition?, TRIGger:A:PULse:WIDth?

Syntax

TRIGger:A:PULse:RUNT?

Example

TRIGger:A:PULse:RUNT?

This query might return

```
:TRIGGER:A:PULSE:RUNT:POLARITY POS-  
ITIVE;THRESHOLD:HIGH 2.0000E+00;LOW  
8,0000E-01;:TRIGGER:A:PULSE:RUNT:WHEN OC-  
CURS;WIDTH 2.0000E-09
```

TRIGger:A:PULse:RUNT:POLarity

Description

This command sets or queries the polarity for the A pulse runt trigger. This command is equivalent to selecting Runt Setup from the Trig menu and then choosing the Polarity setting.

Group

Trigger

Related Commands

TRIGger:A:PULse:RUNT:THReshold?

Syntax 1

```
TRIGger:A:PULse:RUNT:POLarity {POSITIVE|NEGATIVE|EITHER}
```

Syntax 2

```
TRIGger:A:PULse:RUNT:POLarity?
```

Arguments

- POSitive

This indicates that the rising edge crosses the low threshold and the falling edge re-crosses the low threshold without either edge ever crossing the high threshold.

- NEGAtive

This indicates that the falling edge crosses the high threshold and the rising edge re-crosses the high threshold without either edge ever crossing the low threshold.

- EITher

This indicates either negative or positive polarity.

Example 1

```
TRIGger:A:PULse:RUNT:POLarity?
```

This query might return

```
:TRIGGER:A:PULSE:RUNT:POLARITY POSITIVE
```

indicating that the polarity of the A pulse runt trigger is positive.

Example 2

```
TRIGger:A:PULse:RUNT:POLarity NEGATIVE
```

This command specifies that the polarity of the A pulse runt trigger is negative.

TRIGger:A:PULse:RUNT:QUALify

Description

This command sets or queries the Runt Trigger qualification. This is equivalent to selecting Runt Setup from the Trig menu and selecting Occurs or Logic in the Trigger if Runt drop-down list box.

Note: If you use the LOGIc argument, you must specify the channel logic conditions.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:RUNT:QUALify {OCCurs | LOGIc}
```

Syntax 2

```
TRIGger:A:PULse:RUNT:QUALify?
```

Arguments

- OCCurs

This argument specifies a trigger if any detectable event occurs.

- LOGIc

This argument specifies a trigger if the individual channel qualifications meet the logic patterns and thresholds set by the TRIGger:A:LOGIc:INPut:CH<x> and TRIGger:A:LOGIc:THReshold:CH<x> commands.

Example 1

```
TRIGger:A:PULse:RUNT:QUALify?
```

This query might return

```
:TRIGGER:A:PULSE:RUNT:QUALIFY OCCURS
```

indicating that a trigger will occur if the instrument detects any of the runt qualifications.

Example 2

```
TRIGger:A:PULse:RUNT:QUALify OCCurs
```

TRIGger:A:PULse:RUNT:THReshold?

Description

This query-only command returns the upper and lower thresholds for the A pulse runt trigger. This command query is equivalent to selecting Runt Setup from the Trig menu and then viewing the runt trigger Upper Level and Lower Level settings.

Group

Trigger

Related Commands

TRIGger:A:PULse:RUNT:POLarity, TRIGger:A:PULse:RUNT:THReshold:BOTH, TRIGger:A:PULse:RUNT:THReshold:HIGh, TRIGger:A:PULse:RUNT:THReshold:LOW

Syntax

```
TRIGger:A:PULse:RUNT:THReshold?
```

Example

```
TRIGger:A:PULse:RUNT:THReshold?
```

This query might return

```
:TRIGGER:A:PULSE:THRESHOLD:HIGH  
2.0000E+00;LOW 8.0000E-01
```

indicating that the upper threshold is 2 V and that the lower threshold is 800 mV.

TRIGger:A:PULse:RUNT:THReshold:BOTh

Description

This command (no query form) sets the upper and lower switching thresholds for the A pulse runt trigger. This command is equivalent to selecting Runt Setup from the Trig menu and then setting the desired runt trigger Upper Level and Lower Level voltages.

Group

Trigger

Related Commands

TRIGger:A:PULse:RUNT:THReshold?

Syntax

```
TRIGger:A:PULse:RUNT:THReshold:BOTh  
{TTL|ECL}
```

Arguments

- TTL

This sets the upper threshold to 1.8 V and the lower threshold to 800 mV, which are the nominal TTL voltage levels.

- ECL

This sets the upper threshold to -1.1 V and the lower threshold to -1.5 V, which are the nominal ECL voltage levels.

Example

```
TRIGger:A:PULse:RUNT:THReshold:BOTh TTL
```

This command sets the threshold of the pulse runt trigger to the nominal TTL voltage levels.

TRIGger:A:PULse:RUNT:THReshold:HIGH

Description

This command sets or queries the upper limit for the A pulse runt trigger. This command is equivalent to selecting Runt Setup from the Trig menu and then setting the desired runt trigger Upper Level voltage.

Group

Trigger

Related Commands

TRIGger:A:PULse:RUNT:THReshold?

Syntax 1

TRIGger:A:PULse:RUNT:THReshold:HIGH <NR3>

Syntax 2

TRIGger:A:PULse:RUNT:THReshold:HIGH?

Argument

- <NR3>

This specifies the threshold value, in volts.

Example 1

TRIGger:A:PULse:RUNT:THReshold:HIGH?

This query might return

```
:TRIGGER:A:PULSE:RUNT:THRESHOLD:HIGH  
1.1000E+00
```

indicating that the upper limit of the pulse runt trigger is currently set to -1.1 V.

Example 2

TRIGger:A:PULse:RUNT:THReshold:HIGH 120E-3

This command sets the upper limit of the pulse runt trigger to 120 mV.

TRIGger:A:PULse:RUNT:THReshold:LOW

Description

This command sets or queries the lower limit for the A pulse runt trigger. This command is equivalent to selecting Runt Setup from the Trig menu and then setting the desired runt trigger Lower Level voltage.

Group

Trigger

Related Commands

TRIGger:A:PULse:RUNT:THReshold?

Syntax 1

TRIGger:A:PULse:RUNT:THReshold:LOW <NR3>

Syntax 2

TRIGger:A:PULse:RUNT:THReshold:LOW?

Argument

- <NR3>

This specifies the threshold value, in volts.

Example 1

TRIGger:A:PULse:RUNT:THReshold:LOW?

This query might return

```
:TRIGGER:A:PULSE:RUNT:THRESHOLD:LOW  
1.2000E-01
```

indicating that the lower limit of the pulse runt trigger is currently set to 120 mV.

Example 2

TRIGger:A:PULse:RUNT:THReshold:LOW 50E-3

This command sets the lower limit of the pulse runt trigger to 50 mV.

TRIGger:A:PULse:RUNT:WHEn

Description

This command sets or queries the type of pulse width the trigger checks for when it uncovers a runt. This is equivalent to selecting Runt Setup from the Trig menu and then choosing the desired Trigger When setting from the Runt drop-down list.

Group

Trigger

Related Commands

TRIGger:A:PULse:RUNT:WIDth

Syntax 1

```
TRIGger:A:PULse:RUNT:WHEn {OCCurs | WIDERthan}
```

Syntax 2

```
TRIGger:A:PULse:RUNT:WHEn?
```

Arguments

- OCCurs

This specifies a trigger if a runt of any detectable width occurs.

- WIDERthan

This specifies a trigger if a runt greater than the specified width occurs.

Example 1

```
TRIGger:A:PULse:RUNT:WHEn?
```

This query might return

```
:TRIGGER:A:PULSE:RUNT:WHEN OCCURS
```

indicating that a runt trigger will occur if the oscilloscope detects a runt of any detectable width.

Example 2

```
TRIGger:A:PULse:RUNT:WHEn WIDERthan
```

This command sets the runt trigger to occur when the oscilloscope detects a runt in a pulse wider than the specified width.

TRIGger:A:PULse:RUNT:WIDth

Description

This command sets or queries the minimum width for a valid A pulse runt trigger. This command is equivalent to selecting Runt Setup from the Trig menu and then setting the desired Width.

Group

Trigger

Related Commands

TRIGger:A:PULse:RUNT:WHEn

Syntax 1

TRIGger:A:PULse:RUNT:WIDth <NR3>

Syntax 2

TRIGger:A:PULse:RUNT:WIDth?

Argument

- <NR3>

This specifies the minimum width, in seconds.

Example 1

TRIGger:A:PULse:RUNT:WIDth?

This query might return :TRIG-
GER:A:PULSE:RUNT:WIDTH 2.0000E-09, indicating
that the minimum width of a pulse runt trigger is 2 ns.

Example 2

TRIGger:A:PULse:RUNT:WIDth 15E-6

This command sets the minimum width of the pulse runt
trigger to 15 μ s.

TRIGger:A:PULse:SOUrce

Description

This command sets or queries the source for the A pulse trigger. This source parameter applies to all classes of pulse trigger. This command is equivalent to selecting Transition Setup from the Trig menu and then choosing the desired channel from the Source pull-down list.

Group

Trigger

Related Commands

TRIGger:A:EDGE:SOUrce

Syntax 1

```
TRIGger:A:PULse:SOUrce CH<x>
```

Syntax 2

```
TRIGger:A:PULse:SOUrce?
```

Argument

- CH<x>

This specifies one of the input channels, which range from 1 through 4.

Example 1

```
TRIGger:A:PULse:SOUrce?
```

This query might return `:TRIGGER:A:PULSE:SOURCE CH2`, indicating that channel 2 is the source for the A pulse trigger.

Example 2

```
TRIGger:A:PULse:SOUrce CH4
```

This command sets channel 4 as the source for the A pulse trigger.

TRIGger:A:PULse:TIMEOut?

Description

This query-only command returns the polarity and time-out duration for the A pulse time-out trigger. This command is equivalent to selecting Transition Setup from the Trig menu and then viewing the current Polarity and Time settings.

Group

Trigger

Related Commands

TRIGger:A:PULse:TIMEOut:POLarity, TRIGger:A:PULse:TIMEOut:TIME

Syntax

TRIGger:A:PULse:TIMEOut?

Example

TRIGger:A:PULse:TIMEOut?

This query might return :TRIGGER:A:PULSE:TIME-OUT:POLARITY STAYSHIGH;TIME 2.0000E-9, indicating that the current polarity setting for the A pulse trigger is STAYSHIGH (positive) and the current time-out duration is 2 ns.

TRIGger:A:PULse:TIMEOut:POLarity

Description

This command sets or queries the polarity for the A pulse time-out trigger. This command is equivalent to selecting Transition Setup from the Trig menu and then setting the desired Polarity.

Group

Trigger

Related Commands

TRIGger:A:PULse:TIMEOut?

Syntax 1

```
TRIGger:A:PULse:TIMEOut:POLarity {STAYS-  
High|STAYSLow|EITher}
```

Syntax 2

```
TRIGger:A:PULse:TIMEOut:POLarity?
```

Arguments

- STAYSHigh

This indicates that a pulse edge must stay high (positive) for the required time period to permit time out triggering to occur. This is the default polarity.

- STAYSLow

This indicates that a pulse edge must stay low (negative) for the required time period to permit time out triggering to occur.

- EITher

This indicates that the polarity of the time out trigger can stay either high or low (positive or negative) for the required time period to permit time out triggering to occur.

Example 1

```
TRIGger:A:PULse:TIMEOut:POLarity?
```

This query might return `:TRIGGER:A:PULSE:TIME-OUT:POLARITY EITHER`, indicating that the polarity of the A pulse time out trigger can be either positive or negative.

Example 2

`TRIGger:A:PULse:TIMEOut:POLarity STAYSHigh`

This command sets the polarity of the A pulse time out trigger to positive.

TRIGger:A:PULse:TIMEOut:QUALify

Description

This command sets or queries the Timeout Trigger qualification. This is equivalent to selecting Timeout Setup from the Trig menu and selecting Occurs or Logic in the Trigger if Timeout drop-down list box.

Note: If you use the LOGic argument, you must specify the channel logic conditions.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:TIMEOut:QUALify {OC-  
Curs|LOGic}
```

Syntax 2

```
TRIGger:A:PULse:TIMEOut:QUALify?
```

Arguments

- OCCurs

This argument specifies a trigger if any detectable event occurs.

- LOGic

This argument specifies a trigger if the individual channel qualifications meet the logic patterns and thresholds set by the TRIGger:A:LOGic:INPut:CH<x> and TRIGger:A:LOGic:THReshold:CH<x> commands.

Example 1

```
TRIGger:A:PULse:TIMEOut:QUALify?
```

This query might return

```
:TRIGGER:A:PULSE:TIMEOUT:QUALIFY OCCURS
```

indicating that a trigger will occur if the instrument detects any of the timeout qualifications.

Example 2

```
TRIGger:A:PULse:TIMEOut:QUALify OCCurs
```

This command sets the trigger qualifier to OCCurs, which initiates a trigger if the instrument detects any timeout trigger qualifications.

TRIGger:A:PULse:TIMEOut:TIME

Description

This command sets or queries the pulse time-out trigger time, in seconds. This command is equivalent to selecting Transition Setup from the Trig menu and then setting the desired time-out Time.

Group

Trigger

Related Commands

TRIGger:A:PULse:TIMEOut?

Syntax 1

TRIGger:A:PULse:TIMEOut:TIME <NR3>

Syntax 2

TRIGger:A:PULse:TIMEOut:TIME?

Argument

- <NR3>

This specifies the time-out time period, in seconds.

Example 1

TRIGger:A:PULse:TIMEOut:TIME?

This query might return

:TRIGGER:A:PULSE:TIMEOUT:TIME 2.0000E-9

indicating that the time-out time is currently set to 2 ns.

Example 2

TRIGger:A:PULse:TIMEOut:TIME 3.134E-6

This command sets the time-out time to 3.134 μ s.

TRIGger:A:PULse:TRANsition?

Description

This query-only command returns delta time, polarity, and both upper and lower threshold limits for the transition time trigger. This command is equivalent to selecting Transition Setup from the Trig menu and then viewing the current transition settings.

Group

Trigger

Related Commands

TRIGger:A:TYPE, TRIGger:A:PULse:CLAss, TRIGger:A:PULse:GLtch, TRIGger:A:PULse:RUNT, TRIGger:A:PULse:TIMEOut, TRIGger:A:PULse:WIDth

Syntax

```
TRIGger:A:PULse:TRANsition?
```

Example

```
TRIGger:A:PULse:TRANsition?
```

This query might return `:TRIGGER:A:PULSE:TRANSITION:DELTATIME 2.0000E-9;POLARITY POSITIVE;THRESHOLD:HIGH 1.2000;LOW 800.0000E-3;:TRIGGER:A:PULSE:TRANSITION:WHEN SLOWERTHAN`, indicating the current transition time trigger settings.

TRIGger:A:PULse:TRANSition:DELTATime

Description

This command sets or queries the delta time used in calculating the transition value. This is equivalent to selecting Transition Setup from the Trig menu and then setting the desired Time.

Group

Trigger

Related Commands

TRIGger:A:PULse:TRANSition:POLarity, TRIGger:A:PULse:TRANSition:THReshold?

Syntax 1

```
TRIGger:A:PULse:TRANSition:DeltaTime <NR3>
```

Syntax 2

```
TRIGger:A:PULse:TRANSition:DeltaTime?
```

Argument

- <NR3>

This specifies the delta time, in seconds.

Example 1

```
TRIGger:A:PULse:TRANSition:DeltaTime?
```

This query might return `:TRIGGER:A:PULSE:TRANSITION:DELTATIME 2.0000E-09`, indicating that the transition trigger delta time is set to 2 ns.

Example 2

```
TRIGger:A:PULse:TRANSition:DeltaTime 15E-6
```

This command sets the delta time of the transition trigger to 15 μ s.

TRIGger:A:PULse:TRANSition:POLarity

Description

This command sets or queries the polarity for the A pulse transition trigger. This command is equivalent to selecting Transition Setup from the Trig menu and then choosing from the Polarity pull-down list.

Group

Trigger

Related Commands

TRIGger:A:PULse:TRANSition:DELTATime

Syntax 1

```
TRIGger:A:PULse:TRANSition:POLarity {POSITIVE|NEGATIVE|EITHER}
```

Syntax 2

```
TRIGger:A:PULse:TRANSition:POLarity?
```

Arguments

■ POSITIVE

This indicates that a pulse edge must traverse from the lower (most negative) to higher (post positive) level for transition triggering to occur.

■ NEGATIVE

This indicates that a pulse edge must traverse from the upper (most positive) to lower (most negative) level for transition triggering to occur.

■ EITHER

This indicates either positive or negative polarity.

Example 1

```
TRIGger:A:PULSE:TRANSITION:POLARITY?
```

This query might return `:TRIGGER:A:PULSE:TRANSITION:POLARITY EITHER` indicating that the transition can be either positive or negative.

Example 2

```
TRIGGER:A:PULSE:TRANSITION:DELTATIME NEGATIVE
```

This command sets the transition polarity to negative.

TRIGger:A:PULse:TRANSition:QUALify

Description

This command sets or queries the Transition Time Trigger qualification. This is equivalent to selecting Transition Set-up from the Trig menu and selecting Occurs or Logic in the Trigger if Violation drop-down list box.

Note: If you use the LOGic argument, you must specify the channel logic conditions.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:TRANSition:QUALify {OC-  
Curs|LOGic}
```

Syntax 2

```
TRIGger:A:PULse:TRANSition:QUALify?
```

Arguments

- OCCurs

This argument specifies a trigger if any violations occur.

- LOGic

This argument specifies a trigger if the individual channel qualifications meet the logic patterns and thresholds set by the TRIGger:A:LOGic:INPut:CH<x> and TRIGger:A:LOGic:THReshold:CH<x> commands.

Example 1

```
TRIGger:A:PULse:TRANSition:QUALify?
```

This query might return

```
:TRIGGER:A:PULSE:TRANSITION:QUALIFY OCCURS
```

indicating that a trigger will occur if the instrument detects any transition violations.

Example 2

```
TRIGger:A:PULse:TRANSition:QUALify OCCurs
```

This command sets the trigger qualifier to OCCurs, which initiates a trigger if the instrument detects any transition violations.

TRIGger:A:PULse:TRANSition:THReshold?

Description

This query-only command returns the upper and lower threshold limits for the transition time trigger. This command is equivalent to selecting Transition Setup from the Trig menu and then viewing the Upper Level and Lower Level voltage settings.

Group

Trigger

Related Commands

TRIGger:A:PULse:TRANSition:DELTATime, TRIGger:A:PULse:TRANSition:POLarity

Syntax

```
TRIGger:A:PULse:TRANSition:THReshold?
```

Example

```
TRIGger:A:PULse:TRANSition:THReshold?
```

This query might return `:TRIGGER:A:PULSE:TRANSITION:THRESHOLD:HIGH 1.2000;LOW 800.0000E-3`, indicating the upper and lower threshold limits for the transition time trigger.

TRIGger:A:PULse:TRANsition:THReshold:BOTh

Description

This command (no query form) sets the upper and lower thresholds for pulse transition trigger. This command is equivalent to selecting Transition Setup from the Trig menu and then setting the desired Upper Level and Lower Level voltages.

Group

Trigger

Related Commands

TRIGger:A:PULse:TRANsition:THReshold:HIGH, TRIGger:A:PULse:TRANsition:THReshold:LOW

Syntax 1

```
TRIGger:A:PULse:TRANsition:THReshold:BOTh  
{TTL|ECL}
```

Arguments

- TTL

This sets the upper threshold to 1.2 V and the lower threshold to 800 mV, which represent the nominal TTL voltage levels.

- ECL

This sets the upper threshold to -1.1 V and the lower threshold to -1.5 V, which represent the nominal ECL voltage levels.

Example 1

```
TRIGger:A:PULse:TRANsition:THReshold:BOTh  
TTL
```

This command sets the thresholds of the pulse transition trigger to the nominal TTL voltage levels.

TRIGger:A:PULse:TRANSition:THReshold:HIGH

Description

This command sets or queries the upper (most positive) transition trigger threshold. This command is equivalent to selecting Transition Setup from the Trig menu and then setting the desired Upper Level voltage.

Group

Trigger

Related Commands

TRIGger:A:PULse:TRANSition:THReshold:LOW

Syntax 1

```
TRIGger:A:PULse:TRANSition:THReshold:HIGH  
<NR3>
```

Syntax 2

```
TRIGger:A:PULse:TRANSition:THReshold:HIGH?
```

Argument

- <NR3>

This specifies the upper threshold, in volts.

Example 1

```
TRIGger:A:PULse:TRANSition:THReshold:HIGH?
```

This query might return `:TRIGGER:A:PULSE:TRANSITION:THRESHOLD:HIGH 2.0000E+00`, indicating that the upper limit of the pulse transition trigger is 2 V.

Example 2

```
TRIGger:A:PULse:TRANSition:THReshold:HIGH  
120E-3
```

This command sets the upper limit of the pulse transition trigger to 120 mV.

TRIGger:A:PULse:TRANSition:THReshold:LOW

Description

This command sets or queries the lower (most negative) transition trigger threshold. This command is equivalent to selecting Transition Setup from the Trig menu and then setting the desired Lower Level voltage.

Group

Trigger

Related Commands

TRIGger:A:PULse:TRANSition:THReshold:HIGH

Syntax 1

```
TRIGger:A:PULse:TRANSition:THReshold:LOW  
<NR3>
```

Syntax 2

```
TRIGger:A:PULse:TRANSition:THReshold:LOW?
```

Argument

- <NR3>

This specifies the lower threshold, in volts.

Example 1

```
TRIGger:A:PULse:TRANSition:THReshold:LOW?
```

This query might return `:TRIGGER:A:PULSE:TRANSITION:THRESHOLD:LOW 50.0000E-03`, indicating that the lower limit of the pulse transition trigger is 50 mV.

Example 2

```
TRIGger:A:PULse:TRANSition:THReshold:LOW  
20E-3
```

This command sets the lower limit of the pulse transition trigger to 20 mV.

TRIGger:A:PULse:TRANSition:WHEn

Description

This command sets or queries whether to check for a transitioning signal that is faster or slower than the specified delta time. This is equivalent to selecting Transition Setup from the Trig menu and then choosing the desired Trigger When Transition Time setting.

Group

Trigger

Related Commands

TRIGger:A:PULse:TRANSition:DELTAtime, TRIGger:A:PULse:TRANSition:POLarity, TRIGger:A:PULse:TRANSition:THReshold?

Syntax 1

```
TRIGger:A:PULse:TRANSition:WHEn {FASTERthan|SLOWERthan}
```

Syntax 2

```
TRIGger:A:PULse:TRANSition:WHEn?
```

Arguments

- FASTERthan

This sets the trigger to occur when the transitioning signal is faster than the set volts/second rate.

- SLOWERthan

This sets the trigger to occur when the transitioning signal is slower than the set volts/second rate.

Example 1

```
TRIGger:A:PULse:TRANSition:WHEn?
```

This query might return :TRIGGER:A:PULSE:TRANSITION:WHEN FASTERthan, indicating that the transition triggers when the transitioning signal is faster than the set volts/second rate.

Example 2

```
TRIGger:A:PULse:TRANSition:WHEn SLOWERthan
```

This command sets the transition trigger to work when the transitioning signal is slower than the set volts/second rate.

TRIGger:A:PULse:WIDth?

Description

This query-only command returns the width parameters for the A pulse width trigger. This command is equivalent to selecting Width Setup from the Trig menu and then viewing the current pulse width trigger Lower Limit, Upper Limit, Trig When and Polarity settings.

Group

Trigger

Related Commands

TRIGger:A:PULse:WIDth:HIGHLimit, TRIGger:A:PULse:WIDth:LOWLIMIT, TRIGger:A:PULse:WIDth:POLarity, TRIGger:A:PULse:WIDth:WHEn

Syntax 1

```
TRIGger:A:PULse:WIDth
```

Example

```
TRIGger:A:PULse:WIDth?
```

This query might return :TRIGger:A:PULSE:WIDTH:LOWLIMIT 2.0000E-9;HIGHLIMIT 2.0000E-9;WHEN WITHIN;POLARITY POSITIVE as the current A pulse trigger parameters.

TRIGger:A:PULse:WIDth:HIGHLimit

Description

This command sets or queries the upper limit for the A pulse width trigger. This command is equivalent to selecting Width Setup from the Trig menu and then setting the pulse width trigger Upper Limit.

Group

Trigger

Related Commands

TRIGger:A:PULse:WIDth?

Syntax 1

TRIGger:A:PULse:WIDth:HIGHLimit <NR3>

Syntax 2

TRIGger:A:PULse:WIDth:HIGHLimit?

Argument

- <NR3>

This specifies the A pulse width trigger upper limit, in seconds.

Example 1

TRIGger:A:PULse:WIDth:HIGHLimit?

This query might return :TRIGGER:A:PULSE:WIDTH:HIGHLIMIT 2.0000E-9, indicating that the A pulse upper limit is set to 2 ns.

Example 2

TRIGger:A:PULse:WIDth:HIGHLimit 5.0E-6

This command sets the A pulse upper limit to 5 μ s.

TRIGger:A:PULse:WIDth:LOWLIMIT

Description

This command sets or queries the lower limit for the A pulse width trigger. This command is equivalent to selecting Width Setup from the Trig menu and then setting the pulse width trigger Lower Limit.

Group

Trigger

Related Commands

TRIGger:A:PULse:WIDth?

Syntax 1

TRIGger:A:PULse:WIDth:LOWLIMIT <NR3>

Syntax 2

TRIGger:A:PULse:WIDth:LOWLIMIT?

Argument

- <NR3>

This specifies the A pulse width trigger lower limit, in seconds.

Example 1

TRIGger:A:PULse:WIDth:LOWLIMIT?

This query might return :TRIGger:A:PULSE:WIDTH:LOWLIMIT 1.0000E-9, indicating that the A pulse lower limit is set to 1 ns.

Example 2

TRIGger:A:PULse:WIDth:LOWLIMIT 2.0E-6

This command sets the A pulse lower limit to 2 μ s.

TRIGger:A:PULse:WIDth:POLarity

Description

This command sets or queries the polarity for the A pulse width trigger. This command is equivalent to selecting Width Setup from the Trig menu and then selecting the pulse width trigger Polarity.

Group

Trigger

Related Commands

TRIGger:A:PULse:WIDth?

Syntax 1

```
TRIGger:A:PULse:WIDth:POLarity {NEG-  
Ative|POSITIVE}
```

Syntax 2

```
TRIGger:A:PULse:WIDth:POLarity?
```

Arguments

- NEGAtive

This specifies a negative pulse.

- POSITIVE

This specifies a positive pulse.

Example 1

```
TRIGger:A:PULse:WIDth:POLarity?
```

This query might return :TRIGGER:A:PULSE:WIDTH:POLARITY POSITIVE, indicating a positive pulse.

Example 2

```
TRIGger:A:PULse:WIDth:POLarity NEGAtive
```

This command sets the pulse polarity to negative.

TRIGger:A:PULse:WIDth:QUALify

Description

This command sets or queries the Width Trigger qualification. This is equivalent to selecting Width Setup from the Trig menu and selecting Occurs or Logic in the Trigger if Width drop-down list box.

Note: If you use the LOGic argument, you must specify the channel logic conditions.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:WIDth:QUALify {OCCurs | LOGic}
```

Syntax 2

```
TRIGger:A:PULse:WIDth:QUALify?
```

Arguments

- OCCurs

This argument specifies a trigger if any detectable event occurs.

- LOGic

This argument specifies a trigger if the individual channel qualifications meet the logic patterns and thresholds set by the TRIGger:A:LOGic:INPut:CH<x> and TRIGger:A:LOGic:THReshold:CH<x> commands.

Example 1

```
TRIGger:A:PULse:WIDth:QUALify?
```

This query might return

```
:TRIGGER:A:PULSE:WIDTH:QUALIFY OCCURS
```

indicating that a trigger will occur if the instrument detects any of the width qualifications.

Example 2

```
TRIGger:A:PULse:WIDth:QUALify OCCurs
```

This command sets the trigger qualifier to OCCurs, which initiates a trigger if the instrument detects any width trigger qualifications.

TRIGger:A:PULse:WIDth:WHEn

Description

This command sets or queries whether to trigger on a pulse width that falls outside (or within) the specified range of limits. You can define or query trigger pulse width upper and lower limits using the TRIGger:A:PULse:WIDth:HIGHLimit and TRIGger:A:PULse:WIDth:LOWLIMIT commands.

This command is equivalent to selecting Width Setup from the Trig menu and then choosing from the Trig When drop-down list.

Group

Trigger

Related Commands

TRIGger:A:PULse:WIDth:HIGHLimit, TRIGger:A:PULse:WIDth:LOWLIMIT

Syntax 1

```
TRIGger:A:PULse:WIDth:WHEn {OUTside|WITHin}
```

Syntax 2

```
TRIGger:A:PULse:WIDth:WHEn?
```

Arguments

■ OUTside

This specifies a trigger when the duration of the pulse is greater than the high limit or less than the low limit specified. The high and low limits are specified with the TRIGger:A:PULse:WIDth:HIGHLimit and TRIGger:A:PULse:WIDth:LOWLIMIT commands respectively.

■ WITHin

This specifies a trigger when the duration of the pulse is within the high and low limits. The high and low limits are specified with the TRIGger:A:PULse:WIDth:HIGHLimit and TRIGger:A:PULse:WIDth:LOWLIMIT command respectively.

Example 1

```
TRIGger:A:PULse:WIDth:WHEn?
```

This query might return `:TRIGger:A:PULse:WIDth:WHEn OUTSIDE`, indicating the conditions for generating a pulse trigger.

Example 2

`TRIGger:A:PULse:WIDth:WHEn WITHin`

This command specifies that the duration of the A pulse will fall within defined high and low limits.

TRIGger:A:PULse:WINDow:QUALify

Description

This command sets or queries the Window Trigger qualification. This is equivalent to selecting Window Setup from the Trig menu and selecting Occurs or Logic in the Trigger if Window drop-down list box.

Note: If you use the LOGic argument, you must specify the channel logic conditions.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:WINDow:QUALify {OC-  
Curs|LOGic}
```

Syntax 2

```
TRIGger:A:PULse:WINDdow:QUALify?
```

Arguments

- OCCurs

This argument specifies a trigger if any detectable event occurs.

- LOGic

This argument specifies a trigger if the individual channel qualifications meet the logic patterns and thresholds set by the TRIGger:A:LOGic:INPut:CH<x> and TRIGger:A:LOGic:THReshold:CH<x> commands.

Example 1

```
TRIGger:A:PULse:WINDow:QUALify?
```

This query might return

```
:TRIGGER:A:PULSE:WINDOW:QUALIFY OCCURS
```

indicating that a trigger will occur if the instrument detects any of the window qualifications.

Example 2

```
TRIGger:A:PULse:WINDow:QUALify OCCurs
```

This command sets the trigger qualifier to OCCurs, which causes a trigger to occur if the instrument detects any of the window trigger qualifications.

TRIGger:A:PULse:WINDow:THReshold:BOTh

Description

This command (no query form) sets the upper and lower limits for the Window trigger. This command is equivalent to selecting Window Setup from the Trig menu and then setting the desired Upper Level and Lower Level voltage limits.

Group

Trigger

Syntax

```
TRIGger:A:PULse:WINDow:THReshold:BOTh  
{TTL|ECL}
```

Arguments

■ TTL

This sets the upper limit to 2.0 V and the lower limit to 800 mV, which are the nominal TTL voltage levels.

■ ECL

This sets the upper limit to -1.1 V and the lower limit to -1.5 V, which are the nominal ECL voltage levels.

Example

```
TRIGger:A:PULse:WINDow:THReshold:BOTh TTL
```

This command sets the limits of the Window trigger to the nominal TTL voltage levels.

TRIGger:A:PULse:WINDow:THReshold:HIGH

Description

This command sets or queries the upper limit for the Window trigger. This command is equivalent to selecting Window Setup from the Trig menu and then setting the Upper Level voltage limit.

Group

Trigger

Related Commands

TRIGger:A:PULse:WINDow:THReshold:BOTH

Syntax 1

```
TRIGger:A:PULse:WINDow:THReshold:HIGH  
<NR3>
```

Syntax 2

```
TRIGger:A:PULse:WINDow:THReshold:HIGH?
```

Arguments

- <NR3>

This specifies the threshold value in volts.

Example 1

```
TRIGger:A:PULse:WINDow:THReshold:HIGH?
```

This query might return

```
:TRIGGER:A:PULSE:WINDOW:THRESHOLD:HIGH  
1.1000E+00
```

indicating that the upper limit of the pulse Window trigger is currently set to 1.1 V.

Example 2

```
TRIGger:A:PULse:WINDow:THReshold:HIGH 2.0
```

This command sets the upper limit of the Window trigger to 2.0 V.

TRIGger:A:PULse:WINDow:THReshold:LOW

Description

This command sets or queries the lower limit for the Window trigger. This command is equivalent to selecting Window Setup from the Trig menu and then setting the Lower Level voltage limit.

Group

Trigger

Related Commands

TRIGger:A:PULse:WINDow:THReshold:BOTH

Syntax 1

TRIGger:A:PULse:WINDow:THReshold:LOW <NR3>

Syntax 2

TRIGger:A:PULse:WINDow:THReshold:LOW?

Arguments

- <NR3>

This specifies the threshold value in volts.

Example 1

TRIGger:A:PULse:WINDow:THReshold:LOW?

This query might return

```
:TRIGGER:A:PULSE:WINDOW:THRESHOLD:LOW  
1.2000E-01
```

indicating that the lower limit of the Window trigger is currently set to 120 mV.

Example 2

TRIGger:A:PULse:WINDow:THReshold:LOW 0.8

This command sets the lower threshold of the pulse Window trigger to 0.8 V.

TRIGger:A:PULse:WINDow:TYPe

Description

This command sets or queries the window type and is equivalent to selecting Window Setup from the Trig menu and selecting Exits Window or Enter Window in the Window Event drop-down list box.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:WINDow:TYPe {INSide|OUT-  
side}
```

Syntax 2

```
TRIGger:A:PULse:WINDow:TYPe?
```

Arguments

■ INSide

This argument causes a trigger event to occur when a pulse enters the window defined by the upper and lower levels.

■ OUTside

This argument causes a trigger event to occur when a pulse goes outside the window defined by the upper and lower levels.

Example 1

```
TRIGger:A:PULse:WINDow:TYPe?
```

This query might return

```
:TRIGGER:A:PULSE:WINDOW:TYPE INSIDE
```

indicating that the Window Event is set to Enter Window, which causes a trigger event when a pulse enters the window defined by the upper and lower levels.

Example 2

```
TRIGger:A:PULse:WINDow:TYPe OUTside
```

This command sets the window type to outside.

TRIGger:A:PULse:WINDow:WHEn

Description

This command sets or queries the Window Trigger qualification. This is equivalent to selecting Window Setup from the Trig menu and selecting Occurs or Logic in the Trigger if Window drop-down list box.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:WINDow:WHEn {OCCurs|WID-  
ERthan}
```

Syntax 2

```
TRIGger:A:PULse:WINDow:WHEn?
```

Arguments

- OCCurs

This argument specifies a trigger event if any detectable window qualifications occurs.

- WIDERthan

This specifies a trigger event if a window violation greater than the specified width occurs.

Example 1

```
TRIGger:A:PULse:WINDow:WHEn?
```

This query might return

```
:TRIGGER:A:PULSE:WINDOW:WHEN OCCURS
```

indicating that a trigger will occur if the instrument detects any window violation.

Example 2

```
TRIGger:A:PULse:WINDow:WHEn OCCurs
```

This command sets the widow violation qualifier to OCCurs, which causes a trigger to occur when the instrument detects any window violation.

TRIGger:A:PULse:WINDow:WIDth

Description

This command sets or queries the width for a window violation. This command is equivalent to selecting Window Setup from the Trig menu, selecting Inside > t or Outside > t in the Window Event drop-down list box, and setting the Time.

Group

Trigger

Syntax 1

```
TRIGger:A:PULse:WINDow:WIDth <NR3>
```

Syntax 2

```
TRIGger:A:PULse:WINDow:WIDth?
```

Arguments

- <NR3>

This specifies the minimum width in seconds.

Example 1

```
TRIGger:A:PULse:WINDow:WIDth?
```

This query might return

```
:TRIGGER:A:PULSE:WINDOW:WIDTH 2.0000E-09
```

indicating that the minimum width of a window violation is 2 ns.

Example 2

```
TRIGger:A:PULse:WINDow:WIDth 15E-9
```

This command sets the minimum width of a valid window violation to 15 ns.

TRIGger:A:SERIAL:BITRate

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries the clock/data bit rate. Changing the bit rate causes the standard to become "custom". It remains "custom" until another standard is chosen.

Group

Trigger

Syntax 1

```
TRIGger:A:SERIAL:BITRate (<NR3>}
```

Syntax 2

```
TRIGger:A:SERIAL:BITRate?
```

Arguments

- <NR3>

This is the series bit rate and is expressed in bits per second. The range is 1.5e6 to 1.25e9.

Example 1

```
TRIGger:A:SERIAL:BITRATE?
```

This query might return

```
51.8400E+6
```

indicating that a bit rate of 51.84 Mbits/second.

Example 2

```
TRIGger:A:SERIAL:BITRATE 51.8400E+6.
```

This command sets the bit rate to 51.84 Mbits/second.

TRIGger:A:SERIAL:CLOCK:LEVEL

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries the serial trigger clock level.

Group

Trigger

Syntax 1

```
TRIGger:A:SERIAL:CLOCK:LEVEL (<NR3>}
```

Syntax 2

```
TRIGger:A:SERIAL:CLOCK:LEVEL?
```

Arguments

- <NR3>

This is a value in the range of +/- 9.9e37 and is expressed in volts.

Example 1

```
TRIGger:A:SERIAL:CLOCK:LEVEL?
```

This query might return

```
32.0000E-3
```

indicating that the clock level is 32 mV.

Example 2

```
TRIGger:A:SERIAL:CLOCK:LEVEL 0.0
```

This command sets the clock level to 0 V.

TRIGger:A:SERIAL:CLOCK:POLARITY

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries the serial clock polarity.

Group

Trigger

Syntax 1

```
TRIGger:A:SERIAL:CLOCK:POLARITY  
(RISE | FALL)
```

Syntax 2

```
TRIGger:A:SERIAL:CLOCK:POLARITY?
```

Arguments

- RISE

Specifies to trigger on the rising or positive edge of a signal.

- FALL

Specifies to trigger on the falling or negative edge of a signal.

Example 1

```
TRIGger:A:SERIAL:CLOCK:POLARITY?
```

This query might return

```
RISE
```

indicating that the polarity is RISE.

Example 2

```
TRIGger:A:SERIAL:CLOCK:POLARITY RISE
```

This command sets RISE as the clock polarity.

TRIGger:A:SERIAL:CLOCK:SOURce

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries the serial data source channel.

Group

Trigger

Syntax 1

```
TRIGger:A:SERIAL:CLOCK:SOURce {RE-  
COVERed|CH1|CH2|CH3|CH4}
```

Syntax 2

```
TRIGger:A:SERIAL:CLOCK:SOURce?
```

Arguments

- CH1-4

Specifies one of the input channels.

- RECOVERed

Specifies clock polarity and level are extracted from the serial data stream and other clock parameters are ignored.

Example 1

```
TRIGger:A:SERIAL:CLOCK:SOURce?
```

This query might return

```
CH1
```

indicating that the source is channel 1.

Example 2

```
TRIGger:A:SERIAL:CLOCK:SOURce CH1
```

This command sets channel 1 as the source channel.

TRIGger:A:SERIAL:CODE

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries the signal code.

Group

Trigger

Syntax 1

```
TRIGger:A:SERIAL:CODE {NRZ}
```

Syntax 2

```
TRIGger:A:SERIAL:CODE?
```

Arguments

- NRZ

Example 1

```
TRIGger:A:SERIAL:CODE?
```

This query might return

```
NRZ
```

indicating that the signal code is NRZ.

Example 2

```
TRIGger:A:SERIAL:CODE NRZ
```

This command sets the signal code to NRZ.

TRIGger:A:SERIAL:DATA:FORMAT

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries how the Pattern string is formatted.

Group

Trigger

Related Commands

TRIGger:A:SERIAL:DATA:Pattern

Syntax 1

TRIGger:A:SERIAL:DATA:FORMAT (BINary|HEX)

Syntax 2

TRIGger:A:SERIAL:DATA:FORMAT?

Arguments

- BINary

Binary pattern symbols are: {'SPACE'|0|1|X}

'SPACE' is white space and can be included to make the 32 bit pattern easier to read when setting the pattern.

- HEX

Hex pattern symbols are used only with NRZ.

They are: {'|0|1|2|3|4|5|6|7|8|9|A|B|C|D|E|F|X|?'}

Hex 'X' represents the bit pattern 'XXXX'.

Hex '?' represents any other binary bit pattern which doesn't have a hex representation, such as '00X1'.

Example 1

TRIGger:A:SERIAL:DATA:FORMAT?

This query might return

BINARY

indicating that the Pattern string is in a binary format.

Example 2

`TRIGger:A:SERIAL:DATA:FORMAT BINARY.`

This command sets the Pattern string to a binary format.

TRIGger:A:SERIAL:DATA:PATtern

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries the data pattern to allow up to 32 bit serial patterns.

Group

Trigger

Related Commands

TRIGger:A:SERIAL:DATA:FORMat

Syntax 1

TRIGger:A:SERIAL:DATA:PATtern (<string>}

Syntax 2

TRIGger:A:SERIAL:DATA:PATtern?

Arguments

- <string>

Specifies the serial pattern to trigger on. The default value is "XX.. (30 X's)..XX01", such that there are 32 characters total.

Example 1

If one sets the data format to binary with a TRIGGER:A:SERIAL:DATA:FORMAT command, as follows:

```
TRIGger:A:SERIAL:DATA:FORMAT BIN
```

Then the following query:

```
TRIGger:A:SERIAL:DATA:PATTERN?
```

Might return the following 32 characters:

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX10
```

Showing, in this case, 30 X's followed by a 1 and a 0:

Example 2

If one next sets the data format to hexadecimal with the TRIGGER:A:SERIAL:DATA:FORMAT commands as follows:

```
TRIGger:A:SERIAL:DATA:FORMAT HEX
```

Then the following query:

```
TRIGger:A:SERIAL:DATA:PATTERN?
```

Might return the following 8 characters:

```
XXXXXXXX?
```

Showing, in this case, 7 "X"s followed by one "?". Each hex character "X" has replaced four binary "X"s. The hex "?" has replaced the binary "XX10". The "?" was used because "XX10" doesn't have an exact hex representation.

Example 3

If one sends:

```
TRIGger:A:SERIAL:DATA:PATTERN "XXXX XXXa"
```

This will set the pattern to eight hex digits. The space in the middle only serves to make the pattern easier to read.

Example 4

Now, if one sends the following query:

```
TRIGger:A:SERIAL:DATA:PATTERN?
```

This might return:

```
XXXXXXXXA
```

Showing the effect of the pattern sent in example 3. The cosmetic space from example 3 is not returned.

Example 5

Finally, one could change the data format back to binary with the TRIGGER:A:SERIAL:DATA:FORMAT command, as follows:

```
TRIGGER:A:SERIAL:DATA:FORMAT BIN
```

Then one could query the oscilloscope using the pattern command, as follows:

```
TRIGGER:A:SERIAL:DATA:PATTERN?
```

This could return:

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX1010

Showing the 32 character response due to the binary format. Four binary characters "1010" now replace the hex character "A".

TRIGger:A:SERIAL:SOURce

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries the serial data source channel.

Group

Trigger

Syntax 1

```
TRIGger:A:SERIAL:SOURce {CH1|CH2|CH3|CH4}
```

Syntax 2

```
TRIGger:A:SERIAL:SOURce?
```

Arguments

- CH1-4

Specifies one of the input channels.

Example 1

```
TRIGger:A:SERIAL:SOURce?
```

This query might return

```
CH1
```

indicating that the source is channel 1.

Example 2

```
TRIGger:A:SERIAL:SOURce CH1
```

TRIGger:A:SERial:STANdard

(TDS7000 Series option ST & CSA7000 Series)

Description

This command sets or queries the standard that identifies the code and bit rate.

Group

Trigger

Related Commands

TRIGger:A:COMMunication:STANdard,
MASK:STANdard

Syntax 1

```
TRIGger:A:SERIal:STANdard  
{CUSTOm|OC1|OC3|OC12|FC133|FC266|FC531  
|FC1063|FW1394BS400B|FW1394BS800B|  
ENET100FX|ENET1250|RIO_500M|RIO_750M|  
RIO_1G|RIO_SERIAL_1G|VSROC192}
```

Syntax 2

```
TRIGger:A:SERIal:STANdard?
```

Arguments

All of the standards may not be available on every instrument.

- CUSTOm: NRZ, 155.5 Mb/s
- ENET100FX: MLT3, 100 Mb/s
- ENET1250: NRZ, 1.25 Gb/s
- FC133: NRZ, 132.8 Mb/s
- FC266: NRZ, 265.6 Mb/s
- FC531: NRZ, 531.2 Mb/s
- FC1063: NRZ, 1.063 Gb/s
- FW1394BS400B: NRZ, 491.5 Mb/s
- FW1394BS800B: NRZ, 983.0 Mb/s
- OC1: NRZ, 51.84 Mb/s
- OC3: NRZ, 155.5 Mb/s
- OC12: NRZ, 622.1 Mb/s
- RIO_500M: NRZ, 500 Mb/s, triggers
RIO_DRV500, RIO_EDRV500M, and RIO_RCV500,
500Mb/s

-
- RIO_750M: NRZ, 750 Mb/s, triggers RIO_DRV750, RIO_EDRV750M, and RIO_RCV750, 750Mb/s
 - RIO_1G: NRZ, 1.0 Gb/s, triggers RIO_1G
 - RIO_SERIAL_1G: NRZ, 1.25 Gb/s, triggers RIO_Serial_1G
 - VSROC192: NRZ, 1.2441 Gb/s, triggers VSR OC192/STM64

Example 1

TRIGger:A:SERIAL:STANDARD?

This query might return

OC1

Indicating that the signal standard is OC1.

Example 2

TRIGger:A:SERIAL:STANDARD OC1

This command sets the standard to OC1.

TRIGger:A:TYPE

Description

This command sets or queries the type of A trigger. The three types of triggers consist of Edge, Logic, and Pulse. Both Logic and Pulse triggers contain classes. Logic triggers consist of State and Pattern classes; Pulse triggers consist of Glitch, Runt, Width, Transition, and Timeout classes. Once you have set the trigger type, you may also need to identify the associated trigger class. For details on selecting Logic and Pulse trigger classes, see TRIGger:A:LOGic:CLAss and TRIGger:A:PULse:CLAss respectively. This command is similar to selecting A Event (Main) Trigger Setup from the Trig menu and then selecting the desired Trigger Type.

Group

Trigger

Related Commands

TRIGger:A:EDGE, TRIGger:A:LOGic:CLAss, TRIGger:A:PULse:CLAss

Syntax 1

```
TRIGger:A:TYPE {EDGE | LOGic | PULse | COMM | SERIAL}
```

Syntax 2

```
TRIGger:A:TYPE?
```

Arguments

■ EDGE

This is a normal trigger. A trigger event occurs when a signal passes through a specified voltage level in a specified direction and is controlled by the TRIGger:A:EDGE commands.

■ LOGic

This specifies that a trigger occurs when specified conditions are met and is controlled by the TRIGger:A:LOGic commands.

■ PULse

This specifies that a trigger occurs when a specified pulse is found and is controlled by the TRIGger:A:PULse commands.

- COMM

This specifies that a trigger occurs when a communications signal is found. Supports AMI, HDB3, BnZS, CMI, MLT3, Manchester and NRZ encoded communications signals

- SERIAL

This specifies that a trigger occurs when NRZ-encoded data up to 1.25 Gbaud is found, providing a 32-bit serial word. This argument is available with TDS7000 Series Option ST and CSA7000 Series instruments

Example 1

```
TRIGger:A:TYPE?
```

This query might return

```
:TRIGGER:A:TYPE PULSE
```

indicating that the A trigger type is a pulse trigger.

Example 2

```
TRIGger:A:TYPE EDGE
```

This command sets the A trigger type to EDGE.

TRIGger:A:UPPerthreshold:CH<x>

Description

This command sets or queries the CH<x> upper trigger level for TRIGger:LVLSrcpreference SRCDependent.

The CH<x> range is 1 to 4.

Group

Trigger

Related Commands

TRIGger:LVLSrcpreference, TRIGger:A:LOWerthreshold:CH<x>

Syntax 1

```
TRIGger:A:UPPerthreshold:CH<x> {<NR3>|ECL|TTL}
```

Syntax 2

```
TRIGger:A:UPPerthreshold:CH<x>?
```

Arguments

- <NR3>

This specifies the trigger level in user units (usually volts).

- ECL

This specifies the ECL high level.

- TTL

This specifies the TTL high level.

Example 1

```
TRIGger:A:UPPerthreshold:CH<x>?
```

This query might return

```
:TRIGGER:A:UPPERTHRESHOLD:CH2 1.3000E+00
```

indicating that the A trigger level for channel 2 is set to 1.3 V.

Example 2

```
TRIGger:A:UPPerthreshold:CH1 TTL
```

This command sets the A trigger level for channel 1 to TTL high level.

TRIGger:AUXLevel

Description

This command sets or queries the auxiliary (Aux) level for the edge trigger. This command supercedes any other argument that sets the auxiliary trigger level.

Group

Trigger

Related Commands

TRIGger:A:EDGE:SOUrce, TRIGger:B:EDGE:SOUrce

Syntax 1

```
TRIGger:AUXlevel {<NR3> | ECL | TTL}
```

Syntax 2

```
TRIGger:AUXlevel?
```

Arguments

- <NR3>

This is the auxiliary level in volts.

- ECL

This specifies a preset ECL level of -1.3 V.

- TTL

This specifies a preset TTL level of 1.4 V.

Example 1

```
TRIGger:AUXlevel?
```

This query might return

```
:TRIGGER:AUXLEVEL TTL
```

indicating that the auxiliary level for the edge trigger is set to TTL.

Example 2

```
TRIGger:AUXlevel TTL
```

This command sets the auxiliary level for the edge trigger to TTL.

TRIGger:B

Description

This command sets the B trigger level to 50% of minimum and maximum. The query form of this command returns the B trigger parameters. This command is similar to selecting B Event (Delayed) Trigger Setup from the Trig menu and then viewing the current setups.

Group

Trigger

Related Commands

TRIGger:A

Syntax 1

```
TRIGger:B SETLevel
```

Syntax 2

```
TRIGger:B?
```

Argument

- SETLevel

This sets the B trigger level to 50% of MIN and MAX.

Example 1

```
TRIGger:B?
```

This query might return

```
:TRIGGER:B:STATE 0;TYPE EDGE;LEVEL  
-220.0000E-3;BY TIME;EDGE:SOURCE CH1;SLOPE  
RISE;COUPLING DC;:TRIGGER:B:TIME  
16.0000E-9;EVENTS:COUNT 2
```

Example 2

```
TRIGger:B SETLevel
```

This command sets the B trigger level to 50% of MIN and MAX.

TRIGger:B:BY

Description

This command selects or returns whether the B trigger occurs after a specified number of events or a specified period of time after the A trigger. This is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu, selecting the A→B Seq tab, and then choosing Trig After Time or Trig on nth event.

Note: The traditional Runs After functionality is now served by the Horizontal Delay function. For details, see the HORIZontal[:MAIN]:DELay:MODE and HORIZontal[:MAIN]:DELay:TIME commands.

Group

Trigger

Related Commands

TRIGger:B:EVENTS:COUNT, TRIGger:B:TIME, HORIZontal[:MAIN]:DELay:MODE, HORIZontal[:MAIN]:DELay:TIME

Syntax 1

```
TRIGger:B:BY {EVENTS | TIME}
```

Syntax 2

```
TRIGger:B:BY?
```

Arguments

■ EVENTS

This sets the B trigger to take place following a set number of trigger events after the A trigger occurs. The number of events is specified by TRIGger:B:EVENTS:COUNT.

■ TIME

This sets the B trigger to be ready to occur a set time after the A trigger event. The time period is specified by TRIGger:B:TIME.

Example 1

```
TRIGger:B:BY?
```

This query might return

```
:TRIGGER:B:BY EVENTS
```

indicating that the B trigger takes place following a set number of trigger events after the A trigger occurs.

Example 2

`TRIGger:B:BY TIME`

This command sets the B trigger to be ready to occur a set time after the A trigger event.

TRIGger:B:EDGE?

Description

This query-only command returns the source, slope, and coupling for the edge type of B trigger. This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu and then viewing the current Source, Slope, and Coupling settings.

Group

Trigger

Related Commands

TRIGger:B:EDGE:COUpling, TRIGger:B:EDGE:SLOpe,
TRIGger:B:EDGE:SOURce

Syntax

TRIGger:B:EDGE?

Example

TRIGger:B:EDGE?

This query might return

```
:TRIGGER:B:EDGE:SOURCE CH1;SLOPE RISE;COU-  
PLING DC
```

TRIGger:B:EDGE:COUPLing

Description

This command sets or queries the type of coupling for the B trigger. This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu and choosing the desired setting from the Coupling drop-down list.

Group

Trigger

Related Commands

TRIGger:B:EDGE?

Syntax 1

```
TRIGger:B:EDGE:COUPLing {DC|ATRIGger|  
NOISerej}
```

Syntax 2

```
TRIGger:B:EDGE:COUPLing?
```

Arguments

- ATRIGger

Sets the B trigger coupling to match the setting on the A trigger.

- DC

Selects DC trigger coupling. This command is not available for CSA74xx and TDS74xx series instruments.

- NOISerej

Selects DC low sensitivity. This command is not available for CSA74xx and TDS74xx series instruments.

Example 1

```
TRIGger:B:EDGE:COUPLing?
```

This query might return

```
:TRIGGER:B:EDGE:COUPLING ATRIGGER
```

for the B trigger coupling.

Example 2

```
TRIGger:B:EDGE:COUPLing DC
```

This command selects DC for the B trigger coupling.

TRIGger:B:EDGE:SLOpe

Description

This command sets or queries the slope for the B edge trigger. This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu and then choosing the desired Slope.

Group

Trigger

Related Commands

TRIGger:B:EDGE?

Syntax 1

```
TRIGger:B:EDGE:SLOpe {RISe|FALL}
```

Syntax 2

```
TRIGger:B:EDGE:SLOpe?
```

Arguments

- RISE

This specifies the trigger on the rising or positive edge of a signal.

- FALL

This specifies the trigger in the falling or negative edge of a signal.

Example 1

```
TRIGger:B:EDGE:SLOpe?
```

This query might return

```
:TRIGGER:B:EDGE:SLOPE RISE
```

indicating that the B edge trigger occurs on the rising slope.

Example 2

```
TRIGger:B:EDGE:SLOpe FALL
```

This command sets the B edge trigger to occur on the falling slope.

TRIGger:B:EDGE:SOURce

Description

This command sets or queries the source for the B edge trigger. This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu and then choosing the desired setting from the Source drop-down list.

Group

Trigger

Related Commands

TRIGger:B:EDGE?

Syntax 1

```
TRIGger:B:EDGE:SOURce {AUXiliary|CH<x>}
```

Syntax 2

```
TRIGger:B:EDGE:SOURce?
```

Arguments

- AUXiliary

This specifies an external trigger (using the Auxiliary Trigger Input connector, located on the rear panel of the oscilloscope) as the B trigger source.

- CH<x>

This specifies one of the input channels as the delay trigger source. Input channels are specified by x, which can range from 1 through 4.

Example 1

```
TRIGger:B:EDGE:SOURce?
```

This query might return

```
:TRIGGER:B:EDGE:SOURCE CH1
```

indicating that the current input source for the B trigger is channel 1.

Example 2

```
TRIGger:B:EDGE:SOURce CH4
```

This command sets channel 4 as the input source for the B trigger.

TRIGger:B:EVENTS?

Description

This query-only command returns the current B trigger events parameter. This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu, selecting the A→B Seq tab, choosing Trig on nth event, and then viewing the Trig Event setting.

Group

Trigger

Related Commands

TRIGger:B:EVENTS:COUNT

Syntax

TRIGger:B:EVENTS?

Example

TRIGger:B:EVENTS?

This query might return

```
:TRIGGER:B:EVENTS:COUNT 2
```

indicating that 2 events must occur before the B trigger occurs.

TRIGger:B:EVENTS:COUNT

Description

This command sets or queries the number of events that must occur before the B trigger (when TRIG:DELay:BY is set to EVENTS). This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu, selecting the A→B Seq tab, choosing Trig on nth event, and then setting the desired Trig Event value.

Group

Trigger

Related Commands

TRIGger:B:EVENTS?

Syntax 1

```
TRIGger:B:EVENTS:COUNT <NR1>
```

Syntax 2

```
TRIGger:B:EVENTS:COUNT?
```

Argument

- <NR1>

This is the number of B edge trigger events, which can range from 1 to 10,000,000.

Example 1

```
TRIGger:B:EVENTS:COUNT?
```

This query might return

```
:TRIGGER:B:EVENTS:COUNT 2
```

indicating that two events must occur after the A trigger before the B trigger can occur.

Example 2

```
TRIGger:B:EVENTS:COUNT 4
```

This command specifies that the B trigger will occur four trigger events after the A trigger.

TRIGger:B:LEVel

Description

This command sets or queries the level for the B trigger. This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu, selecting the A→B Seq tab and then setting the B Trig Level voltage.

Group

Trigger

Related Commands

TRIGger:A:LEVEL, TRIGger:B, TRIGger:B:EDGE:SOUrce

Syntax 1

```
TRIGger:B:LEVel {ECL|TTL|<NR3>}
```

Syntax 2

```
TRIGger:B:LEVel?
```

Arguments

- ECL

This specifies a preset ECL level of -1.3 V.

- TTL

This specifies a preset TTL level of 1.4 V.

- <NR3>

This is the B trigger level, in volts.

Example 1

```
TRIGger:B:LEVel?
```

This query might return

```
:TRIGGER:B:LEVEL 173.0000E-03
```

indicating that the B trigger level is currently set at 173 mV.

Example 2

```
TRIGger:B:LEVel ECL
```

This command sets the B trigger level to -1.3 V.

TRIGger:B:LEVel:CH<x>

Description

This command sets or queries the CH<x> B trigger level for TRIGger:LVLsrcpreference SRCDependent mode.

The CH<x> range is 1 to 4.

Group

Trigger

Syntax 1

```
TRIGger:B:LEVel:CH<x> {<NR3>|ECL|TTL}
```

Syntax 2

```
TRIGger:B:LEVel:CH<x>?
```

Arguments

- <NR3>

This specifies the trigger level in user units (usually volts).

- ECL

This specifies the ECL high level.

- TTL

This specifies the TTL high level.

Example 1

```
TRIGger:B:LEVel:CH<x>?
```

This query might return

```
:TRIGGER:B:LEVEL:CH2 1.3000E+00
```

indicating that the B trigger level for channel 2 is set to 1.3 V.

Example 2

```
TRIGger:B:LEVel:CH1 TTL
```

This command sets the B trigger level for channel 1 to TTL high level, which is 1.4 V.

TRIGger:B:RESET:SOURce

Description

This command sets or queries the trigger source for the A→B sequential trigger reset feature, except for the Time-out trigger type.

Group

Trigger

Related Commands

TRIGger:B:RESET:TYPE, TRIGger:B:RESET:THReshold

Syntax 1

```
TRIGger:B:RESET:SOURce {CH<x>|AUXiliary}
```

Syntax 2

```
TRIGger:B:RESET:SOURce?
```

Arguments

- CH<x>

This specifies one of the input channels as the reset source. Input channels are specified by x, which can range from 1 through 4

- AUXiliary

This specifies an external trigger (using the Auxiliary Trigger Input connector located on the rear panel) as the reset source

Example 1

```
TRIGger:B:RESET:SOURce?
```

This query might return

```
:TRIGGER:B:RESET:SOURCE CH1
```

indicating that the current input source for the trigger reset is channel 1.

Example 2

```
TRIGger:B:RESET:SOURce CH4
```

This command sets channel 4 as the input source for the trigger reset.

TRIGger:B:RESET:STATE

Description

This command sets or queries the trigger reset state for the A→B sequential State trigger reset feature.

Group

Trigger

Related Commands

TRIGger:B:RESET:TYPE, TRIGger:B:RESET:SOURce, TRIGger:B:RESET:THReshold

Syntax 1

```
TRIGger:B:RESET:STATE {HIGH|LOW}
```

Syntax 2

```
TRIGger:B:RESET:STATE?
```

Arguments

- HIGH

Sets the condition for the state sequential trigger reset to high.

- LOW

Sets the condition for the state sequential trigger reset to low.

Example 1

```
TRIGger:B:RESET:STATE?
```

This query might return

```
:TRIGGER:B:RESET:STATE HIGH
```

indicating that High is the condition for the state sequential trigger reset.

Example 2

```
TRIGger:B:RESET:STATE LOW
```

This command sets the condition for the state sequential trigger reset to Low.

TRIGger:B:RESET:TIMEOut

Description

This command sets or queries the reset timer for a sequential timeout trigger reset. For example, if the timeout is set to 1 μ s, this additional reset time begins following an A trigger event and runs sequentially with any B Trigger Delay. If a B trigger event is not found before it expires, the instrument begins looking for a new A event trigger sequence

Group

Trigger

Related Commands

TRIGger:B:RESET:TYPE

Syntax 1

TRIGger:B:RESET:TIMEOut {<NR3>}

Syntax 2

TRIGger:B:RESET:TIMEOut?

Arguments

- <NR3>

Specifies the reset time.

Example 1

TRIGger:B:RESET:TIMEOut?

This query might return

```
:TRIGGER:B:RESET:TIMEOUT 2.0000E-09
```

indicating that the trigger reset timeout is 2 ns.

Example 2

TRIGger:B:RESET:TIMEOut 1.0E-6

This command sets the trigger reset timeout to 1 μ s.

TRIGger:B:RESET:THReshold

Description

This command sets or queries the trigger threshold for the A→B sequential trigger reset, except for the Timeout reset type.

Group

Trigger

Related Commands

TRIGger:B:RESET:STATE

Syntax 1

TRIGger:B:RESET:THReshold {<NR3> | TTL | ECL}

Syntax 2

TRIGger:B:RESET:THReshold?

Arguments

- <NR3>

This sets the reset threshold in volts.

- ECL

This sets the reset threshold to ECL voltage levels.

- TTL

This sets the reset threshold to TTL voltage levels.

Example 1

TRIGger:B:RESET:THReshold?

This query might return

:TRIGGER:B:RESET:THRESHOLD TTL

indicating that the trigger reset threshold level is set to TTL.

Example 2

TRIGger:B:RESET:THReshold TTL

This command sets the trigger reset threshold level to TTL.

TRIGger:B:RESET:TRANSition

Description

This command sets or queries the type of threshold required for a Transition trigger reset.

Group

Trigger

Related Commands

TRIGger:B:RESET:SOUrce, TRIGger:B:RESET:THResh-old, TRIGger:B:RESET:TYPE

Syntax 1

```
TRIGger:B:RESET:TRANSition {HIGH|LOW}
```

Syntax 2

```
TRIGger:B:RESET:TRANSition?
```

Arguments

- RISE

Indicates that a reset occurs when the trigger exceeds the threshold level.

- FALL

Indicates that a reset occurs when the trigger falls below the threshold level.

Example 1

```
TRIGger:B:RESET:TRANSition?
```

This query might return

```
:TRIGGER:B:RESET:TRANSITION LOW
```

indicating that the reset will occur if the trigger falls below the threshold level.

Example 2

```
TRIGger:B:RESET:TRANSition HIGH
```

This command allows the reset to occur if the trigger exceeds the threshold level.

TRIGger:B:RESET:TYPE

Description

This command sets or queries the type of A→B sequential trigger reset. If the B trigger reset is active, the reset criteria are part of the B triggering sequence. If the reset conditions defined by the reset type are not met, the instrument must start over searching for a new occurrence of the A event.

You must identify a trigger Source and Threshold for each reset type, except for the Timeout trigger type.

Note: If a reset condition occurs, the reset criteria itself is reset and must start over.

Group

Trigger

Syntax 1

```
TRIGger:B:RESET:TYPE {NONE | TIMEOut | STATE |  
TRANSition}
```

Syntax 2

```
TRIGger:B:RESET:TYPE?
```

Arguments

- NONE

Defeats the trigger reset feature.

- TIMEOut

Initiates a reset if the timeout conditions specified by TRIGger:B:RESET:TIMEOut are met.

- STATE

Initiates a reset if the state conditions specified by TRIGger:B:RESET:SOURce, TRIGger:B:RESET:STATE, and TRIGger:B:RESET:THReshold are met.

- TRANSition

Initiates a reset if the transition conditions specified by TRIGger:B:RESET:SOURce, TRIGger:B:RESET:TRANSition, and TRIGger:B:RESET:THReshold are met.

Example 1

```
TRIGger:B:RESET:TYPE?
```

This query might return

`:TRIGGER:B:RESET:TYPE TIMEOUT`

indicating that the sequential trigger reset is active following a timeout.

Example 2

`TRIGger:B:RESET:TYPE NONE`

This command deactivates the sequential trigger reset.

TRIGger:B:STATE

Description

This command sets or queries the state of B trigger activity. If the trigger B state is on, the B trigger is part of the triggering sequence. If trigger B state is off, then only the A trigger causes the trigger event.

Group

Trigger

Related Commands

TRIGger:A:MODE

Syntax 1

```
TRIGger:B:STATE {ON|OFF|<NR1>}
```

Syntax 2

```
TRIGger:B:STATE?
```

Arguments

- ON

This indicates that the B trigger is active and in conjunction with the A trigger causes trigger events.

- OFF

This indicates that only the A trigger causes trigger events.

- <NR1>

A 0 turns off the B trigger; any other value activates the B trigger.

Example 1

```
TRIGger:B:STATE?
```

This query might return

```
:TRIGGER:B:STATE 0
```

indicating that the B trigger is inactive and that only the A trigger causes trigger events.

Example 2

```
TRIGger:B:STATE ON
```

This command sets the B trigger to active, making it capable of causing trigger events.

TRIGger:B:TIME

Description

This command sets or queries trigger B delay time. Trigger B time applies only if TRIGger:B:BY is set to TIME. This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu, choosing the A→B Seq tab, and then setting Trig Delay.

Group

Trigger

Related Commands

TRIGger:B:BY, TRIGger:B:EVENTS:COUNT

Syntax 1

```
TRIGger:B:TIME <NR3>
```

Syntax 2

```
TRIGger:B:TIME?
```

Argument

- <NR3>

This is the B trigger time, in seconds.

Example 1

```
TRIGger:B:TIME?
```

This query might return

```
:TRIGGER:B:TIME 16.0000E-9
```

indicating that trigger B time is currently set to 16 ns.

Example 2

```
TRIGger:B:TIME 4E-6
```

This command sets trigger B time to 4 μ s.

TRIGger:B:TYPe

Description

This command sets or queries the type of B trigger. This command is equivalent to selecting B Event (Delayed) Trigger Setup from the Trig menu and then choosing Edge.

Group

Trigger

Related Commands

TRIGger:A:TYPe

Syntax 1

```
TRIGger:B:TYPe EDGE
```

Syntax 2

```
TRIGger:B:TYPe?
```

Argument

- EDGE

This sets the B trigger type to edge.

Example 1

```
TRIGger:B:TYPe?
```

This query will return

```
:TRIGGER:B:TYPe EDGE
```

Example 2

```
TRIGger:B:TYPe EDGE
```

This command sets the B trigger type to edge.

TRIGger:LVLSrcpreference

Description

This command sets or queries the dependent source/level trigger feature. SRCIndependent is the default value.

Group

Trigger

Related Commands

TRIGger:A:LEVel:CH<x>, TRIGger:B:LEVel:CH<x>, TRIGger:A:UPPerthreshold:CH<x>, TRIGger:A:LOWerthreshold:CH<x>

Syntax 1

```
TRIGger:LVLSrcpreference {SRCDependent | SRCIndependent }
```

Syntax 2

```
TRIGger:LVLSrcpreference?
```

Arguments

- SRCDependent

Following the application of this argument, the level of all trigger sources changes to the value you currently select, regardless of the last value selected.

- SRCIndependent

Following the application of this argument, each trigger source changes to the level you are currently selecting.

Example 1

```
TRIGger:LVLSrcpreference?
```

This query might return

```
:TRIGGER:LVLsrcpreference SRCDEPENDENT
```

indicating that the level of all sources changes to the value you currently select.

Example 2

```
TRIGger:LVLSrcpreference SRCIndependent
```

This command allows you to set the level of each source to the value you are currently selecting.

TRIGger:STATE?

Description

This query-only command returns the current state of the triggering system. This command is equivalent to viewing the trigger status LEDs on the instrument front-panel.

Group

Trigger

Related Commands

TRIGger:A:MODE

Syntax

TRIGger:STATE?

Outputs

- ARMed

This indicates that the oscilloscope is acquiring pretrigger information. All triggers are ignored when TRIGger:STATE is arming.

- AUTO

This indicates that the oscilloscope is in the auto mode and acquires data even in the absence of a trigger.

- DPO

This indicates that the oscilloscope is in DPO mode.

- PARTial

This indicates that the A trigger has occurred and the oscilloscope is waiting for the B trigger to occur.

- REAdy

This indicates that all pre-trigger information has been acquired and that the oscilloscope is ready to accept a trigger.

- SAVe

This indicates that the oscilloscope is in save mode and is not acquiring data.

- TRIGger

This indicates that the oscilloscope has seen a trigger and is acquiring the post trigger information.

Example

TRIGger:STATE?

This query might return

:TRIGGER:STATE ARMED

indicating that the pretrigger data is being acquired.

UNLock

Description

This command (no query form) unlocks the front panel. The command is equivalent to LOCK NONE.

Note: If the oscilloscope is in the Remote With Lockout State (RWLS), the UNLock command has no effect. For more information, see the ANSI-IEEE Std 488.1-1987 Standard Digital Interface for Programmable Instrumentation, section 2.8.3 on RL State Descriptions.

Group

Miscellaneous

Related Commands

LOCK

Syntax

```
UNLock ALL
```

Argument

- ALL

This specifies that all front-panel buttons and knobs are unlocked.

Example

```
UNLock ALL
```

This command unlocks all front-panel buttons and knobs.

VERBoSe

Description

This command sets and queries the Verbose state that controls the length of keywords on query responses. Keywords can be both headers and arguments. This command does not affect IEEE Std 488.2-1987 Common Commands (those starting with an asterisk).

Group

Miscellaneous

Related Commands

HEADer, *LRN?, SET?

Syntax

```
VERBoSe {OFF | ON | <NR1>}
```

Arguments

- OFF

This sets the Verbose State to true, which returns full-length keywords for applicable setting queries.

- ON

This sets the Verbose State to false, which returns minimum-length keywords for applicable setting queries.

- <NR1>

A 0 returns minimum-length keywords for applicable setting queries; any other value returns full-length keywords.

Example 1

```
VERBoSe ON
```

This command enables the Verbose setting.

Example 2

```
VERBOSe?
```

This query might return

```
:VERBOSe 0
```

indicating that the Verbose state is disabled.

WAVFrm?

Description

This query-only command returns WFMOutpre? and CURVe? data for the waveform(s) specified by the DATA:SOURce command. This command is equivalent to sending both WFMOutpre? and CURVe?, with the additional provision that the response to WAVFrm? is guaranteed to provide a synchronized preamble and curve.

Note: This query uses an asynchronous method to obtain and write its output data, which may result in an arbitrarily large volume. Until all output data has been generated and read by your program, or a device clear message is sent, the instrument will not respond to its user controls.

It is recommend that you follow this query with a query of the event status register (*ESR?) to determine if the original query completed successfully.

If the timeout is set to less than the time required to generate and transfer all of the output data, the *ESR? query will indicate that the error bit is set. A subsequent query of the event queue (EVMsg? or ALLEV?) will indicate that the query was interrupted by *ESR?. In such cases, increase the timeout value and send the query again.

Group

Waveform Transfer

Related Commands

CURVe?, DATA:SOURce, WFMOutpre?

Syntax

WAVFrm?

Example

WAVFrm?

This query might return:

```
:WFMOUTPRE:BIT_NR 8;BN_FMT RI;BYT_NR
1;BYT_OR MSB;ENCDG
ASC;NR_PT 500;PT_FMT Y;PT_ORDER LIN-
EAR;PT_OFF 0
;XINCR 400.0000E-12;XZERO 0.0000;XUNIT
"s";YMULT
4.0000E-3;YOFF 0.0000;YZERO 0.0000;YUNIT
"V";WFID "Ch1,
DC coupling, 100.0mV/div, 200.0ns/div,
```

5000 points,
Sample mode";:CURVE
51,50,51,48,51,48,50,49,51,49,51,48,
51,48,51,49,50,49,50,48,49,49,52,49,49,50,
50,48,50,49,
49,49,49,49,50,47,49,47,50,48,49,48,50,48,
49,47,49,48,
51,48,49,47,50,48,50,47,51,47,49,48,48,48,
50,46,50,46,
48,45,48,47,49,47,49,48,49,48,49,45,49,47,
48,46,48,48,
49,45,49,45,47,46,46,46,49,47,48,46,48,46,
48,47,47,46,
47,47,48,46,48,45,48,46,47,46,47,47,46,45,
46,45,47,47,
47,46,46,44,47,45,45,45,46,43,46,45,46,44,
46,45,48,45,
45,44,46,45,46,45,45,45,45,43,45,44,46,44,
47,44,44,44,
45,43,44,44,45,44,46,43,46,43,45,44,45,44,
45,41,44,43,
45,42,44,43,44,43,44,42,43,42,44,41,44,41,
44,42,43,42,
43,42,45,42,43,41,42,41,43,42,44,42,43,42,
43,40,42,40,
42,40,42,41,43,41,42,41,42,39,41,41,42,42,
43,41,42,39,
41,39,41,40,41,40,41,39,41,38,41,39,41,38,
42,39,41,37,
40,38,40,38,39,38,39,38,41,39,39,38,40,37,
38,37,38,38,
39,38,40,36,39,37,39,37,39,37,39,36,40,38,
38,35,38,36,
38,37,39,37,38,37,38,37,37,36,38,35,38,36,
37,36,38,36,
38,35,39,34,36,36,36,35,38,35,36,34,37,34,
37,35,37,36,
37,35,37,35,36,34,36,33,36,35,36,34,36,33,
34,33,34,34,
34,34,36,33,36,34,36,33,34,33,34,32,34,32,
34,33,33,31,
33,32,34,32,33,32,32,31,33,31,31,32,33,31,
32,31,32,30,
32,30,32,31,31,30,31,30,31,29,31,29,32,31,
31,30,30,29,
30,29,30,30,31,29,30,28,30,29,31,28,30,28,
30,29,29,27,
29,28,29,27,30,27,30,26,28,28,29,27,28,27,
28,27,28,28,
28,27,28,27,28,26,28,26,28,26,26,25,27,25,
28,27,28,26,

25,25,26,25,25,25,25,25,26,25,27,25,25,24,
25,25,26,24,
26,24,24,23,25,24,24,22,25,23,24,23,24,22,
24,22,23,22,
25,24,24,21,23,22,22,21,24,22,22,22,22,21,
22,20,23,21,
21,20

WAVFRMStream?

Description

This query-only command returns WFMOutpre? and CURVEStream? waveform data specified by the DATA:SOURce command. This command is similar to sending both WFMOutpre? and CURVEStream?. Additionally, each CURVEStream? response to WAVFRMs? includes a WFMOutpre response, which guarantees a continuously synchronized preamble and curve.

Group

Waveform Transfer

Syntax

WAVFRMStream?

Example 1

WAVFRMStream?

This query might return the following for a single data source, 1 channel with a record length of 500.

```
:WFMOUTPRE <wfmoutpre branch query>;:CURVE #41000<binary data>;<newline>
```

```
:WFMOUTPRE <wfmoutpre branch query>;:CURVE #41000<binary data>;<newline>
```

Example 2

WAVFRMStream?

This query might return the following for multiple data sources with a record length of 500. Wf<x> denotes one of each waveform source specified by DATA:SOURce wf1, wf2, wf3.

```
:WFMOUTPRE <wf1 wfmoutpre branch query>;:CURVE #41000<wfm1 binary data>;:WFMOUTPRE <wf2 wfmoutpre branch query>;:CURVE #41000<wfm2 binary data>;:WFMOUTPRE <wf3 wfmoutpre branch query>;:CURVE #41000<wfm3 binary data>;<newline>
```

```
:WFMOUTPRE <wf1 wfmoutpre branch query>;:CURVE #41000<wfm1 binary data>;:WFMOUTPRE <wf2 wfmoutpre branch query>;:CURVE #41000<wfm2 binary data>;
```

```
:WFMOUTPRE <wf3 wfmoutpre branch  
query>;:CURVE #41000<wfm3 binary  
data>;<newline>
```

WFMinpre?

Description

This query-only command returns the waveform formatting specification to be applied to the next incoming CURVe command data.

Group

Waveform Transfer

Related Commands

WFMOutpre?

Syntax

WFMinpre?

Example

WFMinpre?

This query might return

```
:WFMINPRE:BIT_NR 8;BN_FMT RI;BYT_NR  
1;BYT_OR MSB;ENCDG BIN;NR_PT 500;PT_FMT  
Y;PT_OFF 0;XINCR 2.0000E-6;XZERO  
1.7536E-6;XUNIT "s";YMULT 1.0000E-3;YOFF  
0.0000;YZERO 0.0000;YUNIT "V"
```

WFMInpre:BIT_Nr

Description

This command sets or returns the number of bits per binary waveform point for the waveform, as specified by the DATA:DESTINATION command. This specification is only meaningful when WFMInpre:ENCdg is set to BIN.

Group

Waveform Transfer

Related Commands

DATA:DESTINATION, WFMInpre:ENCdg, WFMInpre:BYT_Nr, WFMOutpre:BIT_Nr

Syntax 1

```
WFMInpre:BIT_Nr <NR1>
```

Syntax 2

```
WFMInpre:BIT_Nr?
```

Argument

- <NR1>

This number of bytes per data point can be 8, 16 (RI, RP) or 32 (FP).

Example 1

```
WFMInpre:BIT_Nr?
```

This query might return

```
:WFMINPRE:BIT_NR 8
```

indicating that incoming RI or RP binary format data uses 8 bits per waveform point.

Example 2

```
WFMInpre:BIT_Nr 16
```

This command sets the number of bits per waveform point to 16, for incoming RI and RP binary format data.

WFMinpre:BN_Fmt

Description

This command sets or queries the format of binary data for incoming waveforms.

Group

Waveform Transfer

Related Commands

WFMOutpre:BN_Fmt

Syntax 1

```
WFMinpre:BN_Fmt {RI|RP|FP}
```

Syntax 2

```
WFMinput:BN_Fmt?
```

Arguments

- RI

This specifies signed integer data point representation.

- RP

This specifies positive integer data point representation.

- FP

This specifies single-precision binary floating point representation.

Example 1

```
WFMinpre:BN_Fmt?
```

This query might return

```
:WFMINPRE:BN_FMT RI
```

indicating that the incoming data is currently interpreted as signed integers.

Example 2

```
WFMinpre:BN_Fmt FP
```

This command specifies that incoming data will be interpreted as single-precision binary floating point.

WFMInpre:BYT_Nr

Description

This command sets or returns the binary field data width for the first ordered waveform, as specified by the DATA:DESTINATION command. This specification is only meaningful when WFMInpre:ENCdg is set to BIN and WFMInpre:BN_Fmt is set to either RI or RP.

Group

Waveform Transfer

Related Commands

DATA:DESTINATION, WFMInpre:BN_Fmt,
WFMInpre:ENCdg, WFMInpre:BIT_Nr,
WFMOutpre:BYT_Nr

Syntax 1

```
WFMInpre:BYT_Nr <NR1>
```

Syntax 2

```
WFMInpre:BYT_Nr?
```

Argument

- <NR1>

This is the number of bytes per data point and can be 1, 2 (RI, RP) or 4 (FP).

Example 1

```
WFMInpre:BYT_Nr?
```

This query might return

```
:WFMINPRE:BYT_NR 2
```

indicating that there are 2 bytes per incoming waveform data point.

Example 2

```
WFMInpre:BYT_Nr 1
```

This command sets the number of bytes per incoming waveform data point to 1, which is the default setting.

WFMInpre:BYT_Or

Description

This command sets or queries which byte of binary waveform data is transmitted first for incoming waveform data when data points require more than one byte. This specification only has meaning when WFMInpre:ENCdg is set to BIN and WFMInpre:BYT_Nr is greater than 1.

Group

Waveform Transfer

Related Commands

WFMInpre:ENCdg, WFMInpre:BYT_Nr,
WFMOutpre:BYT_Or

Syntax 1

```
WFMInpre:BYT_Or {LSB|MSB}
```

Syntax 2

```
WFMInpre:BYT_Or?
```

Arguments

- LSB

This specifies that the least significant byte will be transmitted first.

- MSB

This specifies that the most significant byte will be transmitted first.

Example 1

```
WFMInpre:BYT_Or?
```

This query might return

```
:WFMINPRE:BYT_OR LSB
```

indicating that the least significant incoming CURVe data byte will be transmitted first.

Example 2

```
WFMInpre:BYT_Or MSB
```

This command sets the most significant incoming byte of incoming waveform data to be transmitted first.

WFMInpre:ENCdg

Description

This command sets or queries the type of encoding for incoming waveform data.

Group

Waveform Transfer

Related Commands

WFMOutpre:ENCdg

Syntax 1

```
WFMInpre:ENCdg {ASC|BIN}
```

Syntax 2

```
WFMInpre:ENCdg?
```

Arguments

- ASC

This specifies that the incoming data is in ASCII format.

- BIN

This specifies that the incoming data is in a binary format whose further interpretation requires knowledge of BYT_NR, BIT_NR, BN_FMT, and BYT_OR.

Example 1

```
WFMInpre:ENCdg?
```

This query might return

```
:WFMINPRE:ENCDG BIN
```

indicating that the incoming waveform data will be in binary format.

Example 2

```
WFMInpre:ENCdg ASC
```

This command sets the format of incoming waveform data to ASCII format.

WFMinpre:NR_FR?

Description

This query-only command returns one frame, indicating the number of frames in the transmitted waveform record.

Group

Waveform Transfer

Related Commands

CURVe, DATa, DATa:START, DATa:STOP, DATa:FRAMES-TARt, DATa:FRAMESTOP, SAVe:WAVEform, SAVe:WAVEform:FILEFormat, WFMinpre:NR_Pt, WFMOupre:NR_FR?, WFMOupre:NR_Pt?

Syntax

```
WFMinpre:NR_FR?
```

Example

```
WFMinpre:NR_FR?
```

This query will return

```
:WFMINPRE:NR_FR 1
```

indicating that there is 1 data frame in the transmitted waveform record.

WFMInpre:NR_Pt

Description

This command sets or returns the number of points that are in the transmitted waveform record.

Group

Waveform Transfer

Related Commands

CURVe , DATA, DATA:START, DATA:STOP, DATA:FRAMES-TART, DATA:FRAMESTOP, SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMInpre:NR_FR?, WFMOutpre:NR_FR?, WFMOutpre:NR_Pt?,

Syntax 1

```
WFMInpre:NR_Pt <NR1>
```

Syntax 2

```
WFMInpre:NR_Pt?
```

Argument

- <NR1>

If WFMInpre:PT_Fmt is set to Y, this is the number of data points; If WFMInpre:PT_Fmt is set to ENV, this is the number of max-min pairs.

Example 1

```
WFMInpre:NR_Pt?
```

This query might return

```
:WFMINPRE:NR_PT 8000
```

indicating that there are 8000 data points in the transmitted waveform record.

Example 2

```
WFMInpre:NR_Pt 5000
```

This command specifies that 5000 data points will be expected.

WFMInpre:PT_Fmt

Description

This command sets or queries the point format of the incoming waveform data. Regardless of the argument used, the scale, offset, etc. are interpreted similarly. When ENV is used, waveform data interpreted over the min/max pair; when Y is used, it is interpreted over a single point.

Group

Waveform Transfer

Related Commands

WFMOutpre:PT_Fmt

Syntax 1

```
WFMInpre:PT_Fmt {ENV|Y}
```

Syntax 2

```
WFMInpre:PT_Fmt?
```

Arguments

■ ENV

This specifies that the waveform is transmitted as maximum and minimum point pairs. Only Y values are explicitly transmitted. Absolute coordinates are given by:

$$X_n = XZ_{Ero} + XIN_{cr} (n - PT_{_Off})$$

$$Y_{nmax} = YZ_{Ero} + YMUI_{t} (y_{nmax} - YOFF)$$

$$Y_{nmin} = YZ_{Ero} + YMUI_{t} (y_{nmin} - YOFF)$$

■ Y

This specifies a normal waveform where one ASCII or binary data point is transmitted for each point in the waveform record. Only Y values are explicitly transmitted. Absolute coordinates are given by:

$$X_n = XZ_{Ero} + XIN_{cr} (N - PT_{_Off})$$

$$Y_n = YZ_{Ero} + YMUI_{t} (Y_n - YOFF)$$

Example 1

```
WFMInpre:PT_Fmt ENV
```

This command sets the incoming waveform data point format to enveloped.

Example 2

WFMInpre:PT_Fmt?

This query might return

:WFMINPRE:PT_FMT ENV

indicating that the waveform is transmitted as maximum and minimum point pairs.

WFMInpre:PT_Off

Description

This command specifies or returns the trigger point, within the waveform record, for the reference waveform specified by the DATA:DESTINATION command.

Group

Waveform Transfer

Related Commands

DATA:DESTINATION, DATA:START, WFMOutpre:PT_Off

Syntax 1

```
WFMInpre:PT_Off <NR1>
```

Syntax 2

```
WFMInpre:PT_Off?
```

Argument

- <NR1>

This is (record length -1) to record length and is the position of the data point immediately following the actual trigger. <NR1> is expressed relative to DATA:START.

Example 1

```
WFMInpre:PT_Off 0
```

This command specifies that the trigger point is the first point in the waveform record, which is the default.

Example 2

```
WFMInpre:PT_Off?
```

This query might return

```
:WFMINPRE:PT_OFF 0
```

indicating the incoming waveform trigger point.

WFMinpre:WFId

Description

This command (no query form) accepts but ignores the argument. This command is provided only to allow a waveform extracted from the oscilloscope to be more easily imported.

Group

Waveform Transfer

Related Commands

DATA:DESTination, WFMinpre:BN_Fmt, WFMinpre:ENCdg, WFMaxpre:WFId

Syntax

```
WFMinpre:WFId <String>
```

Argument

- <String>

This must be a valid 488.2 string (but the contents are ignored).

Example

```
WFMinpre:WFId "Ch1, DC coupling,  
2.000V/div, 400.0ns/div, 500 points, Sam-  
ple mode"
```

This is a syntactically correct command.

WFMInpre:XINcr

Description

This command sets or queries the horizontal interval between incoming waveform points in units specified by WFMInpre:XUNit.

Group

Waveform Transfer

Related Commands

WFMInpre:XUNit, WFMOutpre:XINcr

Syntax 1

```
WFMInpre:XINcr <NR3>
```

Syntax 2

```
WFMInpre:XINcr?
```

Argument

- <NR3>

This is the horizontal interval representation.

Example 1

```
WFMInpre:XINcr?
```

This query might return

```
:WFMINPRE:XINCR 1.0000E-3
```

indicating that (assuming that WFMInpre:XUNit is set to "s") there is a 1 ms interval between incoming waveform points.

Example 2

```
WFMInpre:XINcr 3E-3
```

This command sets the interval between Incoming waveform points to 3 ms.

WFMinpre:XUNit

Description

This command sets or returns the horizontal units of the incoming waveform.

Group

Waveform Transfer

Related Commands

WFMOutpre:XUNit

Syntax 1

```
WFMinpre:XUNit <String>
```

Syntax 2

```
WFMinpre:XUNit?
```

Argument

- <String>

This contains a maximum of three alpha characters that represent the horizontal unit of measure for the incoming waveform.

Example 1

```
WFMinpre:XUNit?
```

This query might return

```
:WFMINPRE:XUNIT "s"
```

indicating that the horizontal units for the incoming waveform are expressed in seconds.

Example 2

```
WFMinpre:XUNit "Hz"
```

This command specifies that the horizontal units for the incoming waveform are hertz.

WFMInpre:XZEro

Description

This command sets or queries the (subsample) time between the trigger sample (designated by PT_OFF) and the occurrence of the actual trigger of the incoming waveform. This value is used to compute TTOFF for the incoming waveform, and is expressed in terms of WFMInpre:XUnit.

Group

Waveform Transfer

Related Commands

WFMInpre:PT_Off, WFMInpre:XINcr, WFMInpre:XUnit, WFMOutpre:XZEro

Syntax 1

```
WFMInpre:XZEro <NR3>
```

Syntax 2

```
WFMInpre:XZEro?
```

Argument

- <NR3>

This is value that ranges from -WFMInpre:XINcr to 0.

Example 1

```
WFMInpre:XZEro?
```

This query might return

```
:WFMINPRE:XZEro 7.5000E-6
```

indicating that the trigger occurs 7.5 μ s before the sample designated by WFMInpre:PT_Off.

Example 2

```
WFMInpre:XZEro 5.7E-6
```

This command specifies that the trigger actually occurred 5.7 μ s before the sample designated by WFMInpre:PT_Off

WFMInpre:YMUlt

Description

This command sets or queries the vertical scale factor (in units/digitizing level) for the reference waveform, specified by DATA:DESTination, upon a CURVe command.

Group

Waveform Transfer

Related Commands

DATA:DESTination, WFMInpre:BYT_Nr, WFMInpre:YUNit

Syntax 1

```
WFMInpre:YMUlt <NR3>
```

Syntax 2

```
WFMInpre:UMUlt?
```

Argument

- <NR3>

This is the vertical scale factor per digitizing level of the incoming waveform points.

Example 1

```
WFMInpre:YMUlt?
```

This query might return

```
:WFMINPRE:YMULT 40.0000E-3
```

indicating that the vertical scale is 40 mV/digitizing level (1V/div).

Example 2

```
WFMInpre:YMUlt 20E-3
```

This command specifies that (if WFMInpre:YUNit is "V" and WFMInpre:BYT_Nr is 1) the vertical scale is 20 mV/digitizing level (500 mV/div).

WFMInpre:YOFF

Description

This command sets or queries the vertical position of the incoming waveform in digitizing levels. Variations in this number are analogous to changing the vertical position of the waveform. For those formats in which WFMInpre:BYT_Nr is important (all nonfloating point formats), this command must take the location of the binary point implied by BYT_NR into consideration.

Group

Waveform Transfer

Related Commands

WFMInpre:BYT_Nr, WFMInpre:YMUIt, WFMOutpre:YOFF

Syntax 1

```
WFMInpre:YOFF <NR3>
```

Syntax 2

```
WFMInpre:YOFF?
```

Argument

- <NR3>

This is the vertical offset in digitizing levels.

Example 1

```
WFMInpre:YOFF?
```

This query might return

```
:WFMINPRE:YOFF 25
```

indicating the vertical position of the incoming waveform in digitizing levels.

Example 2

```
WFMInpre:YOFF 50
```

This command specifies that the zero reference for the incoming waveform is 50 digitizing levels (2 divisions) above the center of the data range.

WFMinpre:YUNit

Description

This command sets or queries the vertical units of the incoming waveform.

Group

Waveform Transfer

Related Commands

WFMOutpre:YUNit

Syntax 1

```
WFMinpre:YUNit <String>
```

Syntax 2

```
WFMinpre:YUNit?
```

Argument

- <String>

This contains a maximum of three alpha characters that represent the vertical unit of measure for the incoming waveform.

Example 1

```
WFMinpre:YUNit?
```

This query might return

```
:WFMINPRE:YUNIT "s"
```

indicating the vertical units for the incoming waveform are expressed in seconds.

Example 2

```
WFMinpre:YUNit "Pa"
```

This command specifies that the vertical units for the incoming waveform are pascal.

WFMInpre:YZEro

Description

This command sets or queries the offset of the incoming waveform in units specified by WFMInpre:YUNit. Variations in this number are analogous to changing the vertical offset of the waveform.

Group

Waveform Transfer

Related Commands

WFMInpre:YUNit, WFMOutpre:YZEro

Syntax 1

```
WFMInpre:YZEro <NR3>
```

Syntax 2

```
WFMInpre:YZEro?
```

Argument

- <NR3>

This is of the offset in YUNits,

Example 1

```
WFMInpre:YZEro?
```

This query might return

```
:WFMINPRE:YZEro 7.5000E-6
```

indicating that the zero reference for the incoming waveform is 7.5 μ V below the center of the data range (given that WFMInpre:YUNit is set to V).

Example 2

```
WFMInpre:YZEro 1.5E+0
```

This command specifies that the zero reference for the incoming waveform is 1.5 V below the center of the data range (given that WFMInpre:YUNit is set to V).

WFMOutpre?

Description

This query-only command returns the waveform formatting data for the waveform specified by the DATA:SOURce command. The preamble components are considered to be of two types; formatting and interpretation. The formatting components are: ENCDg, BN_Fmt, BYT_Or, BYT_Nr, BIT_Nr. The interpretation components are derived from the DATA:SOURce specified waveform.

Group

Waveform Transfer

Syntax

WFMOutpre?

Example

WFMOutpre:?

This query might return

```
:WFMOUTPRE:BIT_NR 8;BN_FMT RI
;BYT_NR 1;BYT_OR MSB;ENCDG BIN;NR_PT
500;PT_FMT Y
;PT_ORDER LINEAR;PT_OFF 0;XINCR 8.0000E-9
;XZERO 4.8794E-9;XUNIT "s";YMULT
-2000.0000E-3
;YOFF -4999.9995E-3;YZERO 0.0000;YUNIT "V"
;WFID "Ch1, DC coupling, 2.000V/div,
400.0ns/div,
500 points, Sample mode"
```

WFMOutpre:BIT_Nr

Description

This command sets and returns the number of bits per waveform point that outgoing waveforms contain, as specified by the DATA:SOURce command. Note that values will be constrained according to the underlying waveform data. This specification is only meaningful when WFMOutpre:ENCdg is set to BIN and WFMOutpre:BN_Fmt is set to either RI or RP.

Note: For Math waveforms, or reference waveforms saved from Math waveforms, the data format is floating point and the number of bits per data point is always 32.

Group

Waveform Transfer

Related Commands

DATA:SOURce, WFMOutpre:BN_Fmt, WFMOutpre:ENCdg

Syntax 1

```
WFMOutpre:BIT_Nr <NR1>
```

Syntax 2

```
WFMOutpre:BIT_Nr?
```

Argument

- <NR1>

This number of bits per data point can be 8, 16, 32 or 64.

Example 1

```
WFMOutpre:BIT_Nr?
```

This query might return

```
:WFMOUTPRE:BIT_NR 8
```

indicating that outgoing RI or RP binary format data uses 8 bits per waveform point.

Example 2

```
WFMOutpre:BIT_Nr 16
```

This command sets the number of bits per waveform point to 16, for incoming RI and RP binary format data.

WFMOutpre:BN_Fmt

Description

This command sets or queries the format of binary data for outgoing waveforms specified by the DATA:SOURce command.

Note: For Math waveforms, or reference waveforms saved from Math waveforms, the data format is floating point and the binary format is always FP.

Group

Waveform Transfer

Related Commands

DATA:SOURce

Syntax 1

```
WFMOutpre:BN_Fmt {RI|RP|FP}
```

Syntax 2

```
WFMOutput:BN_Fmt?
```

Arguments

- RI

This specifies signed integer datapoint representation.

- RP

This specifies positive integer datapoint representation.

- FP

This specifies single-precision binary floating point.

Example 1

```
WFMOutpre:BN_Fmt?
```

This query might return

```
:WFMOUTPRE:BN_FMT RI
```

indicating that the outgoing waveform data is currently in signed integer format.

Example 2

```
WFMOutpre:BN_Fmt FP
```

This command specifies that outgoing waveform data will be in single-precision binary floating point format.

WFMOutpre:BYT_Nr

Description

This command sets or returns the binary field data width for the waveform specified by the DATA:SOURCE command. Note that values will be constrained according to the underlying waveform data. This specification is only meaningful when WFMOutpre:ENCdg is set to BIN and WFMOutpre:BN_Fmt is set to either RI or RP.

Note: For Math waveforms, or reference waveforms saved from Math waveforms, the data format is floating point and the number of bytes per data point is always 4.

Group

Waveform Transfer

Related Commands

DATA:SOURCE, WFMOutpre:BN_Fmt, WFMOutpre:ENCdg

Syntax 1

```
WFMOutpre:BYT_Nr <NR1>
```

Syntax 2

```
WFMOutpre:BYT_Nr?
```

Argument

- <NR1>

This is the number of bytes per data point and can be 1, 2, 4 or 8. A value of 1 or 2 bytes per waveform point indicates channel data; 4 bytes per waveform point indicate math data; 8 bytes per waveform point indicate pixel map (DPO or WFMDB) data.

Example 1

```
WFMOutpre:BYT_Nr?
```

This query might return

```
:WFMOUTPRE:BYT_NR 2
```

indicating that there are 2 bytes per outgoing waveform data point.

Example 2

```
WFMOutpre:BYT_Nr 1
```

This command sets the number of bytes per outgoing waveform data point to 1, which is the default setting.

WFMOutpre:BYT_Or

Description

This command sets or queries which byte of binary waveform data is transmitted first, during a waveform data transfer, when data points require more than one byte. This specification only has meaning when WFMOutpre:ENCdg is set to BIN.

Group

Waveform Transfer

Related Commands

WFMOutpre:ENCdg

Syntax 1

```
WFMOutpre:BYT_Or {LSB|MSB}
```

Syntax 2

```
WFMOutpre:BYT_Or?
```

Arguments

- LSB

This specifies that the least significant byte will be transmitted first.

- MSB

This specifies that the most significant byte will be transmitted first.

Example 1

```
WFMOutpre:BYT_Or?
```

This query might return

```
:WFMOUTPRE:BYT_OR LSB
```

indicating that the least significant data byte will be transmitted first.

Example 2

```
WFMOutpre:BYT_Or MSB
```

This command sets the most significant outgoing byte of waveform data to be transmitted first.

WFMOutpre:ENCdg

Description

This command sets and queries the type of encoding for outgoing waveforms.

Note: For Math waveforms, or reference waveforms saved from Math waveforms, the data format is floating point. The only permitted encodings for these waveforms are FPBinary or SFPbinary.

Group

Waveform Transfer

Related Commands

DATA:ENCdg, WFMOutpre:BYT_Nr, WFMOutpre:BYT_Or, WFMOutpre:BIT_Nr, WFMOutpre:BN_Fmt

Syntax 1

```
WFMOutpre:ENCdg {ASC|BIN|FPBinary|SFPbinary}
```

Syntax 2

```
WFMOutpre:ENCdg?
```

Argument

- ASC

This specifies that the outgoing data is to be in ASCII format. Waveforms internally stored as integers will be sent as <NR1> numbers while those stored as floats will be sent as <NR3> numbers.

- BIN

This specifies that outgoing data is to be in a binary format whose further specification is determined by WFMOutpre:BYT_Nr, WFMOutpre:BIT_Nr, WFMOutpre:BN_Fmt and WFMOutpre:BYT_Or.

- FPBinary

This specifies the floating point (width = 4) data.

The range is from -3.4×10^{38} to 3.4×10^{38} . Center screen is 0. The upper limit is one division above the top of the screen and the lower limit is one division below the bottom of the screen.

The FPBinary argument is only applicable to math waveforms or ref waveforms saved from math waveforms.

- SFPbinary

This specifies floating point data in IBM format.

The SFPbinary argument only works on math waveforms or ref waveforms saved from math waveforms.

DATA and WFMOUTPRE		Parameter Settings		
DATA:ENCdg Sett	:ENCdg	:BN_Fmt	:BYT_Or	:BYT_NR
AScii	ASC	N/A	N/A	1, 2, 4, 8
FASText	BIN	RI/FP	MSB	1, 2, 4
RIBinary	BIN	RI	MSB	1, 2, 8
RPBinary	BIN	RP	MSB	1, 2, 8
FPBinary	BIN	FP	MSB	4
SRIbinary	BIN	RI	LSB	1, 2, 8
SRPbinary	BIN	RP	LSB	1, 2, 8
SFPbinary	BIN	FP	LSB	4

Example 1

WFMOutpre:ENCdg?

This query might return

```
:WFMOUTPRE:ENCDG BIN
```

indicating that outgoing waveform data will be sent in binary format.

Example 2

```
WFMOutpre:ENCdg ASC
```

This command specifies that the outgoing waveform data will be sent in ASCII format.

WFMOutpre:NR_FR?

Description

This query-only command returns the number of frames for the DATA:SOURce waveform transmitted in response to a CURVe? query.

Group

Waveform Transfer

Related Commands

CURVe , DATA, DATA:START, DATA:STOP, DATA:FRAMES-TARt, DATA:FRAMESTOP, SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMinpre:NR_FR?, WFMinpre:NR_Pt, WFMOutpre:NR_Pt?

Syntax

```
WFMOutpre:NR_FR?
```

Example

```
WFMOutpre:NR_FR?
```

This query might return

```
:WFMOUTPRE:NR_FR 500
```

indicating that there are 500 data frames to be sent.

WFMOutpre:NR_Pt?

Description

This query-only command returns the number of points for the DATA:SOURce waveform that will be transmitted in response to a CURVe? query.

Group

Waveform Transfer

Related Commands

CURVe , DATA, DATA:START, DATA:STOP, DATA:FRAMES-TARt, DATA:FRAMESTOP, SAVE:WAVEform, SAVE:WAVEform:FILEFormat, WFMinpre:NR_FR?, WFMinpre:NR_Pt, WFMOutpre:NR_FR?

Syntax

```
WFMOutpre:NR_Pt?
```

Example

```
WFMOutpre:NR_Pt?
```

This query might return

```
:WFMOUTPRE:NR_PT 5000
```

indicating that there are 5000 data points to be sent.

WFMOutpre:PT_Fmt?

Description

This query-only command returns the point format for the waveform specified by the DATA:SOURce command. The format specifies a set of equations describing how the scale factors in the preamble are used to give meaning to the CURVe data points.

An error is reported if the DATA:SOURce waveform does not exist.

Group

Waveform Transfer

Related Commands

CURVe?, DATA:SOURce

Syntax

```
WFMOutpre:PT_Fmt?
```

Example

```
WFMOutpre:PT_Fmt?
```

This query might return

```
:WFMOutpre:PT_Fmt ENV
```

indicating that the waveform data are a series of max-min pairs.

WFMOutpre:PT_Off?

Description

This query-only command returns the trigger point relative to DATA:START for the waveform specified by the DATA:SOURce command.

Note: This value is the point immediately following the actual trigger.

Group

Waveform Transfer

Related Commands

DATA:SOURce, DATA:START, WFMOutpre:XZEr0

Syntax

```
WFMOutpre:PT_Off?
```

Example

```
WFMOutpre:PT_Off?
```

This query might return

```
:WFMOUTPRE:PT_OFF 251
```

Specifying that the trigger actually occurred between points 250 and 251.

WFMOutpre:PT_OR?

Description

This query-only command specifies whether the point ordering for the source waveform is FastAcq (Digital Phosphorus Oscilloscopes), WFMDB, or neither.

A FastAcq or WFMDB waveform is stored as a 200 (vertical) by 500 (horizontal) point bitmap. Each point represents the display intensity for that screen location. Only CURVe? query functions are allowed on FastAcq or WFMDB waveforms.

Each data value in the CURVe command query is a 64-bit unsigned quantity representing a normalized intensity of that pixel, based on display persistence.

The DATA:START and DATA:STOP specifications for FastAcq or WFMDB waveforms specify the starting pixmap column and the ending pixmap column, respectively. Any DATA:STOP value in excess of 500 is not applicable to FastAcq or WFMDB waveforms, even though setting DATA:STOP beyond 500 will result in a DATA:STOP? query returning a value that could be in excess of 500. Note that the DATA:START and DATA:STOP values are always constrained by the HORIZONTAL:RECORDlength.

When the WFMOutpre:PT_OR query returns Column, this indicates that the source is a FastAcq or WFMDB waveform and that each of 500 possible horizontal columns being transmitted contains 200 vertical points.

When the WFMOutpre:PT_OR? query returns Linear, this indicates that the source is neither a FastAcq nor a WFMDB waveform and that each horizontal column being sent contains only one vertical point. Note that waveform points are transmitted in the following order: top-to-bottom, then left-to-right.

Group

Waveform Transfer

Related Commands

DATA:SOURce

Syntax

WFMOutpre:PT_OR?

Example 1

WFMOutpre:PT_OR?

This query might return

:WFMOUTPRE:PT_OR COL

indicating that the waveform designated by the DATA:SOURce command is a FastAcq or WFMDB waveform.

Example 2

WFMOutpre:PT_OR?

This query might return

:WFMOUTPRE:PT_OR LINEAR

indicating that the source pointed to is neither a FastAcq nor a WFMDB waveform.

WFMOutpre:WFId?

Description

This query-only command returns a string describing several aspects of the acquisition parameters for the waveform specified by the DATA:SOURce command.

An error is reported if the DATA:SOURce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATA:SOURce

Syntax

WFMOutpre:WFId?

Outputs

<string> comprises the following comma-separated fields:

Waveform Suffixes		
Field	Description	Examples
Source	Source identification string as it appears in the front panel scale factor readouts.	"Ch1-4" "Math1-3" "Ref1-4"
Coupling	A string describing the vertical coupling of the waveform (the Source 1 waveform in the case of Dual Waveform Math)	"AC coupling" "DC coupling" "GND coupling"
Vert Scale	A string giving the vertical scale factor of the unzoomed waveform. The numeric portion will always be 4 digits. The examples cover all known internal units.	"100.0mV/div" "20.00 dB/div" "45.00 deg/div" "785.4 mrad/div" "500.0 uVs/div" "10.00 kV/s/div" "200.0 mV/div" "50.00 unk/div"
Horiz Scale	A string giving the horizontal scale factor of the unzoomed waveform. The numeric portion will always be four 4 digits. The examples cover all known internal units.	"100.0 ms/div" "10.00 kHz/div" "50.00 c/div"
Record Length	A string stating the number of waveform points available in the entire record. The numeric portion is given as an integer.	"500 points" "500000 points"
Acquisition Mode	A string describing the mode used to acquire the waveform.	"Sample mode" "Pk Detect mode" "Hi Res mode" "Envelope mode" "Average mode" "WfmDB mode"
Primary Reference Offset	A string specifying the delta between the Primary Reference (typically, the A trigger) and the CURVe? zero reference location identified by a combination of PT_Off and XZErO in units of XUNits. For example, in Triggerable After Delay, this number would be the actual time between the A and B trigger. In the event that this number is not meaningful, the string will be exactly "0".	"57.2345 ms" "87.3 Hz" "0"

Example

WFMOutpre:WFID?

This query might return :WFMOUTPRE:WFID "Ch1, DC coupling, 100.0mVolts/div,500.0µs/div,500 points, Hi Res mode"

WFMOutpre:XINcr?

Description

This query-only command returns the horizontal point spacing in units of WFMOutpre:XUNit for the waveform specified by the DATA:SOURce command. This value corresponds to the sampling interval.

An error is reported if the DATA:SOURce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATA:SOURce, WFMOutpre:XUNit_x

Syntax

```
WFMOutpre:XINcr?
```

Example

```
WFMOutpre:XINcr?
```

This query might return

```
:WFMOUTPRE:XINCR 10.0000E-6
```

indicating that the horizontal sampling interval was 10 μ s/point (500 μ s/div).

WFMOutpre:XUNit?

Description

This query-only command returns the horizontal units for the waveform specified by the DATA:SOURce command.

An error is reported if the DATA:SOURce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATA:SOURce

Syntax

```
WFMOutpre:XUNit?
```

Example

```
WFMOutpre:XUNit?
```

This query might return

```
:WFMOUTPRE:XUNIT "Hz"
```

indicating that the horizontal units for the waveform are measured in hertz.

WFMOutpre:XZEro?

Description

This query-only command returns the (subsample) time between the trigger sample (designated by PT_OFF) and the occurrence of the actual trigger for the waveform specified by the DATA:SOURce command. This value is in units of WFMOutpre:XUNit.

An error is reported if the DATA:SOURce waveform does not exist.

Note: During *stopped state operation* (i.e., AC-Quire:STATE OFF), this is the only preamble that changes on each acquisition. If a query is run during *steady state operation* (i.e., all control changes have settled and triggers are arriving on a regular basis), the XZEro value of the *last* stopped state is returned.

Group

Waveform Transfer

Related Commands

DATA:SOURce, WFMOutpre:XUNit,

Syntax

```
WFMOutpre:XZEro?
```

Example

```
WFMOutpre:XZEro?
```

This query might return

```
:WFMOUTPRE:XZERO 5.6300E-9
```

indicating that the trigger actually occurred 5.63 ns before the trigger sample.

WFMOutpre:YMUlt?

Description

This query-only command returns the vertical scale factor per digitizing level in units specified by the WFMOutpre:YUNit command for the waveform specified by the DATA:SOURce command. For those formats in which WFMOutpre:BYT_Nr is important (all nonfloating point formats), WFMOutpre:YMUlt? must take the location of the binary point implied by BYT_NR into consideration.

An error is reported if the DATA:SOURce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATA:SOURce

Syntax

```
WFMOutpre:YMUlt?
```

Example

```
WFMOutpre:YMUlt?
```

This query might return

```
:WFMOUTPRE:YMULT 4.0000E-3
```

indicating that the vertical scale for the corresponding waveform was 100 mV/div.

WFMOutpre:YOff?

Description

This query-only command returns the vertical offset in digitized levels for the waveform specified by the DATA:SOURce command. For those formats in which BYT_NR is important (all non-floating point formats), this command must take the location of the binary point implied by WFMOutpre:BYT_Nr into consideration.

An error is reported if the DATA:SOURce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATA:SOURce, WFMOutpre:BYT_Nr

Syntax

```
WFMOutpre:YOff?
```

Example

```
WFMOutpre:YOff?
```

This query might return

```
:WFMOUTPRE:YOFF -50.0000E+0
```

indicating that the position indicator for the waveform was 50 digitizing levels (2 divisions) below center screen.

WFMOutpre:YUNIT?

Description

This query-only command returns the vertical units for the waveform specified by the DATA:SOURce command.

An error is reported if the DATA:SOURce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATA:SOURce

Syntax

```
WFMOutpre:YUNIT?
```

Example

```
WFMOutpre:YUNIT?
```

This query might return

```
:WFMOUTPRE:YUNIT "dB"
```

indicating that the vertical units for the waveform are measured in decibels.

WFMOutpre:YZero?

Description

This query-only command returns the vertical offset in units specified by WFMOutpre:YUNit for the waveform specified by the DATa:SOURce command.

An error is reported if the DATa:SOURce waveform does not exist.

Group

Waveform Transfer

Related Commands

DATa:SOURce, WFMOutpre:YUNit

Syntax

```
WFMOutpre:YZero?
```

Example

```
WFMOutpre:YUNit?
```

This query might return

```
:WFMOUTPRE:YZERO -100.0000E-3
```

indicating that vertical offset was set to -100 mV.

WLENgth:VALue

Description

This command sets or returns the wavelength to calibrate the instrument to. This allows the instrument to accurately convert optical power to internal volts using the gain factor associated with that wavelength. Accounting for this gain factor enables the instrument to display the most accurate vertical setting and readings.

This works with a CSA7000B Series instrument, when equipped with the O/E Electrical Out-to-Ch1 Input Interconnector and so working as a calibrated optical electrical receiver.

Group

Vertical

Syntax 1

```
WLENgth:VALue  
{WL1550NM|WL1310NM|WL850NM|WL780NM}
```

Syntax 2

```
WLENgth:VALue?
```

Arguments

- WL1550NM

Calibrate to a wavelength of 1550 NM

- WL1310NM

Calibrate to a wavelength of 1310 NM

- WL850NM

Calibrate to a wavelength of 850 NM

- WL780NM

Calibrate to a wavelength of 780 NM

Example

```
WLENGTH:VALUE?
```

This query might return

```
:WLENGTH:VALUE WL1550NM
```

indicating a calibration to 1550 NM

ZOOM

Description

This command resets the zoom transforms to default values for all traces or live traces. The ZOOM query returns the current vertical and horizontal positioning and scaling of the display.

Group

Zoom

Syntax 1

```
ZOOM {RESET|RESETLive}
```

Syntax 2

```
ZOOM?
```

Arguments

- RESET

This resets the zoom transforms to default values for all traces.

- RESETLive

This resets the zoom transforms to default values for live traces.

Example 1

```
ZOOM?
```

This query might return :

```
:ZOOM:MODE 0;GRATICULE:SIZE 50;:ZOOM:HORIZONTAL:LOCK ALL;
:ZOOM:CH1:HORIZONTAL:POSITION 50.0000;SCALE 2;
:ZOOM:CH1:VERTICAL:POSITION 0.0000;SCALE 1.0000;
:ZOOM:CH2:HORIZONTAL:POSITION 50.0000;SCALE 2;
:ZOOM:CH2:VERTICAL:POSITION 0.0000;SCALE 1.0000;
:ZOOM:CH3:HORIZONTAL:POSITION 50.0000;SCALE 2;
:ZOOM:CH3:VERTICAL:POSITION 0.0000;SCALE 1.0000;
:ZOOM:CH4:HORIZONTAL:POSITION 50.0000;SCALE 2;
```

```
:ZOOM:CH4:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:MATH1:HORIZONTAL:POSITION
50.0000;SCALE 2;
:ZOOM:MATH1:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:MATH2:HORIZONTAL:POSITION
50.0000;SCALE 2;
:ZOOM:MATH2:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:MATH3:HORIZONTAL:POSITION
50.0000;SCALE 2;
:ZOOM:MATH3:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:MATH4:HORIZONTAL:POSITION
50.0000;SCALE 2;
:ZOOM:MATH4:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:REF1:HORIZONTAL:POSITION
50.0000;SCALE 2;
:ZOOM:REF1:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:REF2:HORIZONTAL:POSITION
50.0000;SCALE 2;
:ZOOM:REF2:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:REF3:HORIZONTAL:POSITION
50.0000;SCALE 2;
:ZOOM:REF3:VERTICAL:POSITION 0.0000;SCALE
1.0000;
:ZOOM:REF4:HORIZONTAL:POSITION
50.0000;SCALE 2;
:ZOOM:REF4:VERTICAL:POSITION 0.0000;SCALE
1.0000
```

Example 2

```
ZOOM RESET
```

This command resets the zoom transforms to default values for all traces.

ZOOM:GRATICULE:SIZE

Description

This command sets or queries the size, in percent, of the Zoom (lower) graticule.

Group

Zoom

Syntax 1

```
ZOOM:GRATICULE:SIZE {50|80|100}
```

Syntax 2

```
ZOOM:GRATICULE:SIZE?
```

Arguments

- 50

Set the size of the Zoom graticule to 50 %.

- 80

Set the size of the Zoom graticule to 80 %.

- 100

Set the size of the Zoom graticule to 100 %.

Example 1

```
ZOOM:GRATICULE:SIZE?
```

This query might return

```
:ZOOM:GRATICULE:SIZE 100
```

Indicating that the zoom graticule size is set to 100%.

Example 2

```
ZOOM:GRATICULE:SIZE 100
```

Setting the zoom graticule size to 100%.

ZOOM:GRAticule:SPLIT

Description

This command sets or returns the sizes of the acquisition and zoom windows when Zoom is selected.

Note: For this command to have an effect on the display, you must have the zoom mode turned on. Use the ZOOM:MODE command to turn on the zoom mode. The ZOOM:GRAticule:SPLIT command is equivalent to selecting Zoom Setup from the Horiz/Acq menu and selecting a value for the Graticule Split in the control window.

Group

Zoom

Related Commands

ZOOM:MODE

Syntax 1

```
ZOOM:GRAticule:SPLit {FIFTYfifty|EIGHTyt-  
wenty|FuLL}
```

Syntax 2

```
ZOOM:GRAticule:SPLit?
```

Arguments

- FIFTYfifty

This argument sets half of the available display to the zoomed graticule and half of the available display to the acquisition graticule; this argument is the default value.

- EIGHTytwenty

This argument sets 80% of the available display to the zoomed graticule and 20% to the acquisition graticule.

- FuLL

This argument sets the entire display to the zoomed graticule.

Example 1

```
ZOOM:GRAticule:SPLit?
```

This query might return :ZOOM:GRATICULE:SPLIT FIFTYFIFTY, indicating that the display area is divided equally between the zoomed graticule and the acquisition graticule.

Example 2

```
ZOOm:GRAticule:SPLit FULL
```

This command sets the full display area to the zoomed graticule.

ZOOm:MODE

Description

This command turns Zoom mode on or off. The Zoom query returns the current state of Zoom mode. This command is equivalent to pressing the **ZOOM** button located on the front panel.

Group

Zoom

Syntax 1

```
ZOOm:MODE {ON|OFF|<NR1>}
```

Syntax 2

```
ZOOm:MODE
```

Arguments

- ON

This turns on Zoom mode.

- OFF

This turns off Zoom mode.

- <NR1>

A 0 turns off Zoom mode; any other value turns on Zoom mode.

Example 1

```
ZOOM:MODE?
```

This query might return

```
:WFMOUTPRE:ZOOM:MODE 1
```

indicating that Zoom mode is currently turned on.

Example 2

```
ZOOM:MODE OFF
```

This command turns off Zoom mode.

ZOOm:SCROLL:DIREction

Description

This command sets or returns the direction for automatic scrolling of zoomed waveforms.

Group

Zoom

Syntax 1

```
ZOOm:SCROLL:DIREction {FWD|FFWD|RE-  
Verse|FREVerse|STOP}
```

Syntax 2

```
ZOOm:SCROLL:DIREction?
```

Arguments

- FWD

Starts AutoScroll. The Zoom Box moves from left to right.

- FFWD

Starts AutoScroll. Increases the scrolling speed in the forward direction.

- REVerse

Starts AutoScroll. The Zoom Box moves from right to left.

- FREVerse

Starts AutoScroll. Increases the scrolling speed in the reverse direction.

- STOP

Halts AutoScroll.

Example 1

```
ZOOm:SCROLL:DIREction FWD
```

Starts AutoScroll. The Zoom Box moves from left to right.

Example 2

```
ZOOm:SCROLL:DIREction?
```

Returns FWD, FFWD, REVERSE, FREVERSE, or STOP.

ZOOm:SCROLL:LOCK

Description

This command sets or returns the state of Scroll Lock, which "locks" zoomed waveforms under common control.

Group

Zoom

Syntax 1

```
ZOOm:SCROLL:LOCK {ON|OFF}
```

Syntax 2

```
ZOOm:SCROLL:LOCK?
```

Arguments

- ON

Enables Scroll Lock.

- OFF

Disables Scroll Lock.

Example 1

```
ZOOm:SCROLL:LOCK ON
```

Enables Scroll Lock for zoomed waveforms identified as locked.

Example 2

```
ZOOm:SCROLL:LOCK?
```

Returns either ON or OFF.

ZOOm:SCROLL:SPEED

Description

This command sets or returns the speed of automatic scrolling.

Group

Zoom

Syntax 1

```
ZOOm:SCROLL:SPEED <NR1>
```

Syntax 2

```
ZOOm:SCROLL:SPEED?
```

Arguments

- <NR1>

This is a value from 1 to 10.

Example 1

```
ZOOm:SCROLL:SPEED 2
```

AutoScroll will move zoomed waveform(s) 2 divisions per update.

Example 2

```
ZOOm:SCROLL:LOCK?
```

Returns <NR1> 1-10.

ZOOm:ZOOM<x>

Description

This command resets the zoom transforms to default values for all traces of the specified zoom. The ZOOm query returns the current vertical and horizontal positioning and scaling of the display.

Group

Zoom

Syntax 1

ZOOm:ZOOM<x> {RESET|RESETLive}

Syntax 2

ZOOm:ZOOM<x>?

Arguments

- RESET

Resets the zoom transforms to default values for all traces of the specified zoom.

- RESETLive

Resets the zoom transforms to default values for live traces of the specified zoom.

ZOOM:ZOOM<x>:SCROLLLock

Description

This command sets or queries Scroll Lock for the specified zoom.

Group

Zoom

Syntax 1

```
ZOOM:ZOOM<x>:SCROLLLock {ON|OFF|1|0}
```

Syntax 2

```
ZOOM:ZOOM<x>:SCROLLLock?
```

Arguments

- ON {1}

Turns Zoom<1-4> on.

- OFF {0}

Turns Zoom<1-4> off.

Example 1

```
ZOOM:ZOOM1:SCROLLLock ON
```

This command turns Zoom1 scroll lock on.

Example 2

```
ZOOM:ZOOM1:SCROLLLock?
```

This query returns either ON|1 or OFF|0 for Zoom1.

ZOOM:ZOOM<x>:STATE

Description

This command sets or queries the specified zoom on or off. Default is Zoom1 on, Zoom2 - 4 off.

Group

Zoom

Syntax 1

```
ZOOM:ZOOM<x>:STATE {ON|OFF|<NR1>}
```

Syntax 2

```
ZOOM:ZOOM<x>:STATE?
```

Arguments

- ON {1}

Turns Zoom<1-4> on.

- OFF {0}

Turns Zoom<1-4> off.

- <NR1>

A 0 disables the specified zoom; any other value enables the specified zoom.

Example 1

```
ZOOM:ZOOM1:STATE ON
```

This command turns Zoom1 on.

Example 2

```
ZOOM:ZOOM1:STATE?
```

This query returns either ON | 1 or OFF | 0 for Zoom1.

ZOOm:ZOOM<x>:<wfm>:HORizontal:POSition

Description

This command sets or queries the horizontal position of the specified waveform for the specified zoom. The setting of the ZOOm:ZOOM<x>:SCROLLLOCK command determines the waveforms that are affected.

Group

Zoom

Syntax 1

```
ZOOm:ZOOM<x>:<wfm>:HORizontal:POSition  
<NR3>
```

Syntax 2

```
ZOOm:ZOOM<x>:<wfm>:HORizontal:POSition?
```

Argument

- <NR3>

This is a value from 0 to 100.00 and is the percent of the waveform that is to the left of screen center, when the zoom factor is 1x or greater.

Example 1

```
ZOOm:ZOOM1:CH1:HORizontal:POSition?
```

This query might return :ZOOM1:CH1:HORIZONTAL:POSITION 50.0000, indicating that the Zoom1 reference pointer for Channel 1 is currently set at 50% of acquired waveform.

Example 2

```
ZOOm:ZOOM1:CH1:HORizontal:POSition 50
```

This command sets the Zoom1 reference pointer at 50% of acquired waveform.

ZOOm:ZOOM<x>:<wfm>:HORizontal:SCAle

Description

This command sets or queries the zoom horizontal scale factor of the specified waveform for the specified zoom.

Group

Zoom

Syntax 1

```
ZOOm:ZOOM<x>:<wfm>:HORizontal:SCAle <NR3>
```

Syntax 2

```
ZOOm:ZOOM<x>:<wfm>:HORizontal:SCAle?
```

Argument

- <NR3>

This is the amount of expansion in the horizontal direction in 1-2 -5 increments.

Example 1

```
ZOOm:ZOOM2:CH2:HORizontal:SCAle?
```

This query might return :ZOOM2:CH2:HORIZONTAL:SCALE 1, indicating that Channel 2 has a horizontal scale factor of 1.

Example 2

```
ZOOm:ZOOM1:CH1:HORizontal:SCAle 5
```

This command sets the horizontal scale factor of Channel 1 to 5.

ZOOm:ZOOM<x>:<wfm>:VERTical:POSition

Description

This command sets or queries the vertical position of the specified waveform for the specified zoom. The setting of the ZOOm:ZOOM<x>:SCROLLLOCK command determines the waveforms that are affected.

Group

Zoom

Syntax 1

```
ZOOm:ZOOM<x>:<wfm>:VERTical:POSition <NR3>
```

Syntax 2

```
ZOOm:ZOOM<x>:<wfm>:VERTical:POSition?
```

Argument

- <NR3>

This is the vertical position, expressed in divisions.

Example 1

```
ZOOm:ZOOM1:CH1:VERTical:POSition?
```

This query might return :ZOOm1:CH1:VERTical:POSition 0.0000, indicating that the Zoom1 trace for Channel 1 is centered at division 0 of the acquired trace.

Example 2

```
ZOOm:ZOOM1:CH1:VERTical:POSition 2
```

This command sets the Zoom1 vertical position to 2, which centers the zoom trace at the second division of the acquired trace.

ZOOm:ZOOM<x>:<wfm>:VERTical:SCAle

Description

This command sets or queries the zoom vertical scale of the specified waveform for the specified zoom.

Group

Zoom

Syntax 1

```
ZOOm:ZOOM<x>:<wfm>:VERTical:SCAle <NR3>
```

Syntax 2

```
ZOOm:ZOOM<x>:<wfm>:VERTical:SCAle?
```

Argument

- <NR3>

This is the amount of vertical expansion or compression, which operates on a 1-2-5 sequence (for example, 1, 2, 5, 10, 20, 50, 100...). Based on the value entered, this command uses the nearest scale factor. Setting the vertical scale to 1 indicates unity (no zoom).

Example 1

```
ZOOm:ZOOM2:CH2:VERTical:SCAle?
```

This query might return :ZOOM2:CH2:VERTICAL:SCALE 2.0000, indicating that the vertical scale is 2x.

Example 2

```
ZOOm:ZOOM2:CH2:VERTical:SCAle 5
```

This command sets the vertical scale of Channel 2 to 5x.

Status and Events

Status and Events

The oscilloscope provides a status and event reporting system for the GPIB interfaces. This system informs you of certain significant events that occur within the oscilloscope.

The oscilloscope status handling system consists of five 8-bit registers and two queues for each interface. The remaining Status subtopics describe these registers and components. They also explain how the event handling system operates.

Registers

Overview

The registers in the event handling system fall into two functional groups:

- Status Registers contain information about the status of the oscilloscope. They include the Standard Event Status Register (SESR) and the Status Byte Register (SBR).
- Enable Registers determine whether selected types of events are reported to the Status Registers and the Event Queue. They include the Device Event Status Enable Register (DESER), the Event Status Enable Register (ESER), and the Service Request Enable Register (SRER).

Status

The Standard Event Status Register (SESR) and the Status Byte Register (SBR) record certain types of events that may occur while the oscilloscope is in use. IEEE Std 488.2-1987 defines these registers.

Each bit in a Status Register records a particular type of event, such as an execution error or message available. When an event of a given type occurs, the oscilloscope sets the bit that represents that type of event to a value of one. (You can disable bits so that they ignore events and remain at zero. See Enable Registers). Reading the status registers tells you what types of events have occurred.

The Standard Event Status Register (SESR) - The SESR records eight types of events that can occur within the oscilloscope. Use the *ESR? query to read the SESR register. Reading the register clears the bits of the register so that the register can accumulate information about new events.

Note: TekVISA applications use SESR bit 6 to respond to any of several events, including some front panel actions.

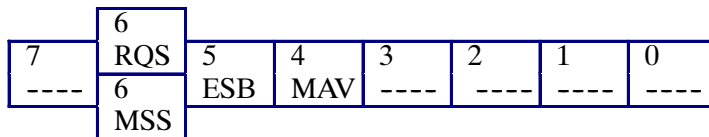
Standard Event Status Register (SESR)							
7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

SESR Bit Functions	
Bit	Function
7 (MSB)	PON Power On. Shows that the oscilloscope was powered on. On completion, the diagnostic tests also set this bit.
6	URQ User Request. Indicates that an application event has occurred. *See note.
5	CME Command Error. Shows that an error occurred while the oscilloscope was parsing a command or query.
4	EXE Execution Error. Shows that an error occurred while the oscilloscope was executing a command or query.
3	DDE Device Error. Shows that a device error occurred.
2	QYE Query Error. Either an attempt was made to read the Output Queue when no data was present or pending, or that data in the Output Queue was lost.
1	RQC Request Control. This is not used.
0 (LSB)	OPC Operation Complete. Shows that the operation is complete. This bit is set when all pending operations complete following an *OPC command.

The Status Byte Register (SBR) – Records whether output is available in the Output Queue, whether the oscilloscope requests service, and whether the SESR has recorded any events.

Use a Serial Poll or the *STB? query to read the contents of the SBR. The bits in the SBR are set and cleared depending on the contents of the SESR, the Event Status Enable Register (ESER), and the Output Queue. When you use a Serial Poll to obtain the SBR, bit 6 is the RQS bit. When you use the *STB? query to obtain the SBR, bit 6 is the MSS bit. Reading the SBR does not clear the bits.

Status Byte Register (SBR)



SBR Bit Functions	
Bit	Function
7 (MSB)	----- Not used.
6	RQS Request Service. Obtained from a serial poll. Shows that the oscilloscope requests service from the GPIB controller.
6	MSS Master Status Summary. Obtained from *STB? query. Summarizes the ESB and MAV bits in the SBR.
5	ESB Event Status Bit. Shows that status is enabled and present in the SESR.
4	MAV Message Available. Shows that output is available in the Output Queue.
3	----- Not used.
2	----- Not used.
1 – 0	----- Not used.

Enable

DESER, ESER, and SRER allow you to select which events are reported to the Status Registers and the Event Queue. Each Enable Register acts as a filter to a Status Register (the DESER also acts as a filter to the Event Queue) and can prevent information from being recorded in the register or queue.

Each bit in an Enable Register corresponds to a bit in the Status Register it controls. In order for an event to be reported to a bit in the Status Register, the corresponding bit in the Enable Register must be set to one. If the bit in the Enable Register is set to zero, the event is not recorded.

Various commands set the bits in the Enable Registers. The Enable Registers and the commands used to set them are described below.

The Device Event Status Enable Register (DESER) -
This register controls which types of events are reported to the SESR and the Event Queue. The bits in the DESER correspond to those in the SESR.

Use the DESE command to enable and disable the bits in the DESER. Use the DESE? query to read the DESER.

Device Event Status Enable Register (DESER)							
7 PON	6 URQ	5 CME	4 EXE	3 DDE	2 QYE	1 RQC	0 OPC

The Event Status Enable Register (ESER) – This register controls which types of events are summarized by the Event Status Bit (ESB) in the SBR. Use the *ESE command to set the bits in the ESER. Use the *ESE? query to read it.

Event Status Enable Register (ESER)							
7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

The Service Request Enable Register (SRER) – This register controls which bits in the SBR generate a Service Request and are summarized by the Master Status Summary (MSS) bit.

Use the *SRE command to set the SRER. Use the *SRE? query to read the register. The RQS bit remains set to one until either the Status Byte Register is read with a Serial Poll or the MSS bit changes back to a zero.

Service Request Enable Register (SRER)							
7	6	5	4	3	2	1	0
-----	-----	ESB	MAV	-----	-----	-----	-----

*PSC Command

The *PSC command controls the Enable Registers contents at power-on. Sending *PSC 1 sets the Enable Registers at power on as follows:

- DESER 255 (equivalent to a DESe 255 command)
- ESER 0 (equivalent to an *ESE 0 command)
- SRER 0 (equivalent to an *SRE 0 command)

Sending *PSC 0 lets the Enable Registers maintain their values in nonvolatile memory through a power cycle.

Note: To enable the PON (Power On) event to generate a Service Request, send *PSC 0, use the DESe and *ESE commands to enable PON in the DESER and ESER, and use the *SRE command to enable bit 5 in the SRER. Subsequent power-on cycles will generate a Service Request.

Queues

Overview

The oscilloscope status and event reporting system contains two queues: the Output Queue and the Event Queue.

Output Queue

The oscilloscope stores query responses in the Output Queue and empties this queue each time it receives a new command or query message after an <EOM>. The controller must read a query response before it sends the next command (or query) or it will lose responses to earlier queries.

Caution: When a controller sends a query, an <EOM>, and a second query, the oscilloscope normally clears the first response and outputs the second while reporting a Query Error (QYE bit in the ESER) to indicate the lost response. A fast controller, however, may receive a part or all of the first response as well. To avoid this situation, the controller should always read the response immediately after sending any terminated query message or send a DCL (Device Clear) before sending the second query.

Event Queue

The Event Queue stores detailed information on up to 129 events. If more than 128 events stack up in the Event Queue, the 128th event is replaced by event code 350, "Queue Overflow."

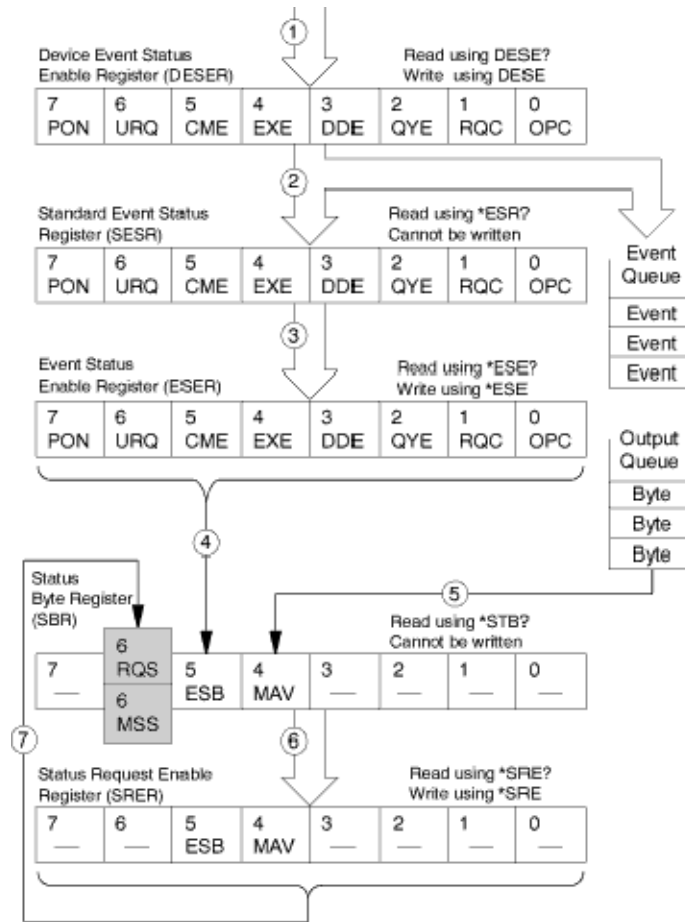
Read the Event Queue with the EVENT? query (which returns only the event number), with the EVMsg? query (which returns the event number and a text description of the event), or with the ALLEV? query (which returns all the event numbers along with a description of the event). Reading an event removes it from the queue.

Before reading an event from the Event Queue, you must use the *ESR? query to read the summary of the event from the SESR. This makes the events summarized by the *ESR? read available to the EVENT? and EVMsg? queries, and empties the SESR.

Reading the SESR erases any events that were summarized by previous *ESR? reads but not read from the Event Queue. Events that follow an *ESR? read are put in the Event Queue but are not available until *ESR? is used again.

Event Handling Sequence

The figure below shows how to use the status and event handling system. In the explanation that follows, numbers in parentheses refer to numbers in the figure.



When an event occurs, a signal is sent to the DESER (1). If that type of event is enabled in the DESER (that is, if the bit for that event type is set to 1), the appropriate bit in the SESR is set to one, and the event is recorded in the Event Queue (2). If the corresponding bit in the ESER is also enabled (3), then the ESB bit in the SBR is set to one (4).

When output is sent to the Output Queue, the MAV bit in the SBR is set to one (5).

When a bit in the SBR is set to one and the corresponding bit in the SRER is enabled (6), the MSS bit in the SBR is set to one and a service request is generated (7).

Synchronization Methods

Although most GPIB commands are completed almost immediately after being received by the oscilloscope, some commands start a process that requires more time. For example, once a single sequence acquisition command is executed, it may be a few seconds before the acquisition is complete. Rather than remain idle while the operation is in process, the oscilloscope will continue processing other commands. This means that some operations will not be completed in the order that they were sent.

Sometimes the result of an operation depends on the result of an earlier operation. A first operation must complete before the next one gets processed. The oscilloscope status and event reporting system provides ways to do this.

For example, a typical application might involve acquiring a single-sequence waveform and then taking a measurement on the acquired waveform. You could use the following command sequence to do this:

```
/** Set up conditional acquisition **/  
ACQUIRE:STATE OFF  
SELECT:CH1 ON  
HORIZONTAL:RECORDLENGTH 500  
ACQUIRE:MODE SAMPLE  
ACQUIRE:STOPAFTER SEQUENCE  
  
/** Acquire waveform data **/  
ACQUIRE:STATE ON  
  
/** Set up the measurement parameters **/  
MEASUREMENT:IMMED:TYPE AMPLITUDE  
MEASUREMENT:IMMED:SOURCE CH1  
  
/** Take amplitude measurement **/  
MEASUREMENT:MEAS1:VALUE?
```

The acquisition of the waveform requires extended processing time. It may not finish before the oscilloscope takes an amplitude measurement (see figure below). This can result in an incorrect amplitude value.

```
| ACQUIRE:STATE  
ON |
```

```
+-----+
```

```
| ACquiring |
```

```
| Waveform |
```

```
| Data      |
```

```
+-----+
```

```
| Take Measurement |
```

```
+-----+
```

```
|<-----Processing Time----->|
```

To ensure the oscilloscope completes waveform acquisition before taking the measurement on the acquired data, you can synchronize the program.

```
| ACQUIRE:STATE  
ON |
```

```
+-----+
```

```
| ACquiring |
```

```
| Waveform |
```

```
| Data      |
```

```
+-----+
```

```
| Take Measurement |
```

```
+-----+
```

```
|<-----Processing Time----->|
```

You can use four commands to synchronize the operation of the oscilloscope with your application program: *WAI, BUSY?, *OPC, and *OPC?

***WAI**

You can force commands to execute sequentially by using the *WAI command. This command forces completion of the previous commands that generate an OPC message before processing new ones. No commands after the *WAI are processed before the OPC message(s) are generated.

The same command sequence using the *WAI command for synchronization looks like this:

```
/* Set up conditional acquisition */
ACQUIRE:STATE OFF
SELECT:CH1 ON
HORIZONTAL:RECORDLENGTH 500
ACQUIRE:MODE SAMPLE
ACQUIRE:STOPAFTER SEQUENCE
/* Acquire waveform data */
ACQUIRE:STATE ON
/* Set up the measurement parameters */
MEASUREMENT:IMMED:TYPE AMPLITUDE
MEASUREMENT:IMMED:SOURCE CH1
/* Wait until the acquisition is complete before
   taking the measurement
*/
*WAI
/* Take amplitude measurement */
MEASUREMENT:IMMED:VALUE?
```

Though *WAI is one of the easiest ways to achieve synchronization, it is also the most costly. The processing time of the oscilloscope is slowed since it is processing a single command at a time. This time could be spent doing other tasks.

The controller can continue to write commands to the input buffer of the oscilloscope, but the commands will not be

processed by the oscilloscope until all (OPC related) operations in process are complete. If the input buffer becomes full, the controller will be unable to write more commands to the buffer. This can cause a time-out.

BUSY?

The BUSY? query allows you to find out whether the oscilloscope is busy processing a command that has an extended processing time such as single-sequence acquisition.

The same command sequence, using the BUSY? query for synchronization, looks like this:

```
/* Set up conditional acquisition */
ACQUIRE:STATE OFF
SELECT:CH1 ON
HORIZONTAL:RECORDLENGTH 500
ACQUIRE:MODE SAMPLE
ACQUIRE:STOPAFTER SEQUENCE

/* Acquire waveform data */
ACQUIRE:STATE ON

/* Set up the measurement parameters */
MEASUREMENT:IMMED:TYPE AMPLITUDE
MEASUREMENT:IMMED:SOURCE CH1

/* Wait until the acquisition is complete before
   taking the measurement
*/
While BUSY? keep looping

/* Take amplitude measurement */
MEASUREMENT:IMMED:VALUE?
```

This sequence lets you create your own wait loop rather than using the *WAI command. The BUSY? query helps you avoid time-outs caused by writing too many commands to the input buffer. The controller is still tied up, though, and the repeated BUSY? query will result in more bus traffic.

*OPC

If the corresponding status registers are enabled, the *OPC command sets the OPC bit in the Standard Event Status Register (SESR) when an operation is complete. You achieve synchronization by using this command with either a serial poll or service request handler.

Serial Poll Method: Enable the OPC bit in the Device Event Status Enable Register (DESER) and the Event Status Enable Register (ESER) using the DESE and *ESE commands.

When the operation is complete, the OPC bit in the Standard Event Status Register (SESR) will be enabled and the Event Status Bit (ESB) in the Status Byte Register will be enabled.

The same command sequence using the *OPC command for synchronization with serial polling looks like this:

```
/* Set up conditional acquisition */
ACQUIRE:STATE OFF
SELECT:CH1 ON
HORIZONTAL:RECORDLENGTH 500
ACQUIRE:MODE SAMPLE
ACQUIRE:STOPAFTER SEQUENCE
/* Enable the status registers */
DESE 1
*ESE 1
*SRE 0
/* Acquire waveform data */
ACQUIRE:STATE ON
/* Set up the measurement parameters */
MEASUREMENT:IMMED:TYPE AMPLITUDE
MEASUREMENT:IMMED:SOURCE CH1
/* Wait until the acquisition is complete before
taking the measurement.
*/
```

*OPC

While serial poll = 0, keep looping

/* Take amplitude measurement */

MEASUREMENT:IMMED:VALUE?

This technique requires less bus traffic than did looping on BUSY?.

Service Request Method: Enable the OPC bit in the Device Event Status Enable Register (DESER) and the Event Status Enable Register (ESER) using the DESE and *ESE commands.

You can also enable service requests by setting the ESB bit in the Service Request Enable Register (SRER) using the *SRE command. When the operation is complete, a Service Request will be generated.

The same command sequence using the *OPC command for synchronization looks like this:

/* Set up conditional acquisition */

ACQUIRE:STATE OFF

SELECT:CH1 ON

HORIZONTAL:RECORDLENGTH 500

ACQUIRE:MODE SAMPLE

ACQUIRE:STOPAFTER SEQUENCE

/* Enable the status registers */

DESE 1

*ESE 1

*SRE 32

/* Acquire waveform data */

ACQUIRE:STATE ON

/* Set up the measurement parameters */

MEASUREMENT:IMMED:TYPE AMPLITUDE

MEASUREMENT:IMMED:SOURCE CH1

/* Wait until the acquisition is complete before taking the measurement

*/

*OPC

Program can now do different tasks such as talk to other devices. The SRQ, when it comes, interrupts those tasks and returns control to this task.

/* Take amplitude measurement */

MEASUREMENT:IMMED:VALUE?

This technique is more efficient but requires more sophisticated programming.

***OPC?**

The *OPC? query places a 1 in the Output Queue once an operation that generates an OPC message is complete. A time out could occur if you try to read the output queue before there is any data in it.

The same command sequence using the *OPC? query for synchronization looks like this:

```
/* Set up conditional acquisition */  
ACQUIRE:STATE OFF  
SELECT:CH1 ON  
HORIZONTAL:RECORDLENGTH 500  
ACQUIRE:MODE SAMPLE  
ACQUIRE:STOPAFTER SEQUENCE  
/* Acquire waveform data */  
ACQUIRE:STATE ON
```

```
/* Set up the measurement parameters */  
MEASUREMENT:IMMED:TYPE AMPLITUDE  
MEASUREMENT:IMMED:SOURCE CH1  
/* Wait until the acquisition is complete before  
taking the measurement  
*/
```

```
*OPC?
```

Wait for read from Output Queue.

```
/* Take amplitude measurement */  
MEASUREMENT:IMMED:VALUE?
```

This is the simplest approach. It requires no status handling or loops. However, you must set the controller time-out for longer than the acquisition operation.

Messages

The information contained in the topic tabs above covers all the programming interface messages the oscilloscope generates in response to commands and queries.

For most messages, a secondary message from the oscilloscope gives more detail about the cause of the error or the meaning of the message. This message is part of the message string and is separated from the main message by a semicolon.

Each message is the result of an event. Each type of event sets a specific bit in the SESR and is controlled by the equivalent bit in the DESER. Thus, each message is associated with a specific SESR bit. In the message tables, the associated SESR bit is specified in the table title, with exceptions noted with the error message text.

No Event

The following table shows the messages when the system has no events or status to report. These have no associated SESR bit.

No Event Messages	
Code	Message
0	No events to report; queue empty
1	No events to report; new events pending *ESR?

Command Error

The following table shows the command error messages generated by improper syntax. Check that the command is properly formed and that it follows the rules in the section on command Syntax.

Command Error Messages (CME Bit 5)	
Code	Message
100	Command error
101	Invalid character
102	Syntax error
103	Invalid separator
104	Data type error
105	GET not allowed
108	Parameter not allowed
109	Missing parameter
110	Command header error
111	Header separator error
112	Program mnemonic too long
113	Undefined header
114	Header suffix out of range
120	Numeric data error
121	Invalid character in number
123	Exponent too large
124	Too many digits
128	Numeric data not allowed
130	Suffix error
131	Invalid suffix
134	Suffix too long
138	Suffix not allowed
140	Character data error
141	Invalid character data
144	Character data too long
148	Character data not allowed
150	String data error
151	Invalid string data
158	String data not allowed
160	Block data error
161	Invalid block data
168	Block data not allowed
170	Command expression error
171	Invalid expression
178	Expression data not allowed

Execution Error

The following table lists the execution errors that are detected during execution of a command.

Execution Error Messages (EXE Bit 4)	
Code	Message
200	Execution error
201	Invalid while in local
202	Settings lost due to RTL
210	Trigger error
211	Trigger ignored
212	Arm ignored
219	Trigger delay not available
220	Parameter error
221	Settings conflict
222	Data out of range
223	Too much data
224	Illegal parameter value
225	Out of memory
230	Data corrupt or stale
240	Hardware error
241	Hardware missing
244	Invalid parameter selected
250	Mass storage error
251	Missing mass storage
252	Missing media
253	Corrupt media
254	Media full
255	Directory full
256	File name not found
257	File name error
258	Media Protected
260	Execution expression error
261	Math error in expression
2200	Measurement error, Measurement system error
2201	Measurement error, Zero period
2202	Measurement error, No period found
2203	Measurement error, No period, second waveform
2204	Measurement error, Low signal amplitude
2205	Measurement error, Low amplitude, second waveform
2206	Measurement error, Invalid gate
2207	Measurement error, Measurement overflow
2208	Measurement error, Waveform does not cross Mid Ref

2209	Measurement error, No second Mid Ref crossing
2210	Measurement error, No Mid Ref crossing, second waveform
2211	Measurement error, No backwards Mid Ref crossing
2212	Measurement error, No negative crossing
2213	Measurement error, No positive crossing
2214	Measurement error, No crossing
2215	Measurement error, No crossing, second waveform
2216	Measurement error, No crossing, target waveform
2217	Measurement error, Constant waveform
2218	Measurement error, Unused
2219	Measurement error, No valid edge - No arm sample
2220	Measurement error, No valid edge - No arm cross
2221	Measurement error, No valid edge - No trigger cross
2222	Measurement error, No valid edge - No second cross
2223	Measurement error, Waveform mismatch
2225	Measurement error, No waveform to measure
2226	Measurement error, Null Waveform
2227	Measurement error, Positive and Negative Clipping
2228	Measurement error, Positive Clipping
2229	Measurement error, Negative Clipping
2230	Measurement error, High Ref < Low Ref
2231	Measurement error, no statistics available
2241	Waveform request is invalid
2243	This meas cannot be performed on this type of wfm
2244	Sourcewfm is not active
2248	This ref cannot be activated
2249	Reference deletion error, Waveform in use for math

2250	Reference error, Waveform reference file is invalid
2251	Reference error, Waveform preamble does not match actual
2252	Reference error, Source waveform is not valid
2253	Reference error, Too many points received
2254	Reference error, Too few points received
2259	File too big
2261	Calibration error, wait for the warmup interval to expire before invoking SPC
2400	Not enough memory available
2401	This channel cannot be activated
2402	MathMeasHisto have circular definition
2410	Empty math string
2411	Syntax error in math string
2412	Semantic error in math string
2413	Math expression is too complex
2420	Histogram cannot be performed on this type of waveform

Device Error

The following table lists the device errors that can occur during oscilloscope operation. These errors may indicate that the oscilloscope needs repair.

Device Error Messages (DDE Bit 3)	
Code	Message
300	Device-specific error
310	System error
311	Memory error
312	PUD memory lost
313	Calibration memory lost
314	Save/recall memory lost
315	Configuration memory lost
350	Queue overflow (does not set DDE bit)

System Event

The following table lists the system event messages. These messages are generated whenever certain system conditions occur.

System Event Messages	
Code	Message
400	Query event
401	Power on (PON bit 7 set)
402	Operation complete (OPC bit 0 set)
403	User request (URQ bit 6 set)
404	Power fail (DDE bit 3 set)
405	Request control
410	Query INTERRUPTED (QYE bit 2 set)
420	Query UNTERMINATED (QYE bit 2 set)
430	Query DEADLOCKED (QYE bit 2 set)
440	Query UNTERMINATED after indefinite response (QYE bit 2 set)

Execution Warning

The following table lists warning messages that do not interrupt the flow of command execution. These notify you that you may get unexpected results.

Execution Warning Messages (EXE Bit 4)	
Code	Message
540	Measurement warning
541	Measurement warning, Low signal amplitude
542	Measurement warning, Unstable histogram
543	Measurement warning, Low resolution
544	Measurement warning, Uncertain edge
545	Measurement warning, Invalid min/max
546	Measurement warning, Need 3 edges
547	Measurement warning, Clipping positive/negative
548	Measurement warning, Clipping positive
549	Measurement warning, Clipping negative
551	FASTAcq mode is active - deactivate to use math

Internal Warning

The following table shows internal errors that indicate an internal fault in the oscilloscope.

Internal Warning Messages	
Code	Message
600	Internal warning
630	Internal warning, 50% overload

Programming Examples

Overview of Programming Examples

The example programs illustrate methods you can use to control the oscilloscope from the GPIB interface. These programs are listed on your product CD and are auto-installed in an example directory for you.

The programs run on a PC compatible system equipped with a National Instruments GPIB board and associated drivers. For example, the GPIB programs work with a National Instruments AT-GPIB/TNT GPIB package.

Compiling the Example GPIB Programs

All the example GPIB programs assume that the GPIB system recognizes the oscilloscope as DEV1 and the PC (controller) as GPIB0. You can assign these names using different methods depending on your operating system.

In Windows NT and 2000, you can assign these names through the Start > Settings > Control Panel > GPIB > Configure Button.

In Windows 95 and 98, you can assign the GPIB name through the Start > Settings > Control Panel > System Icon > Device Manager > Display > AT-GPIB/TNT > GPIB.

The example GPIB software includes:

- MEAS: measures a parameter of an oscilloscope waveform.

- TL: a talker/listener program.

The example programs that come with your TDS or CSA 7000B Series oscilloscope contains programs written in Microsoft Visual C++ 6.0.

Visual C++ code is in V-CPLUS6. Within this directory, an executable version of the program is in the PROGRAMS subdirectory. Source versions are in the SOURCES subdirectory.

A README file in each directory has the latest documentation.

If you wish to develop code, you will need to use files that come with the GPIB system. Specifically, the Visual C++ programs use DECL_32.h and GPIB_32.obj.

Note: The GPIB programs you compile in the Sources directory work with the National Instruments AT-GPIB/TNT GPIB system. It may take extra steps or changes to get them to work with older Tektronix GURU and other GPIB systems.

Compiling and Linking Example Visual C++ Programs

To make an executable for any of the files (for example, AUTOSET project), perform the following steps:

1. Install Visual C++
2. Install the National Instruments AT-GPIB/TNT GPIB board and drivers. *Remember to identify the GPIB device as DEV1 (see GPIB Examples section for more information) and to reboot your PC to initialize the GPIB drivers.*
3. Set up a project for each example. The example below creates a new project for the AUTOSET example program.
 - a. Invoke Visual C++.
 - b. Select "New" from the File menu.
 - c. Choose "New Win32 Console Application".
 - d. Give the project a name, for example, GeTwfm.
 - e. Select "Empty Project".
 - f. Add the following files to the project from the Project menu:
 - getwfm.c
 - gpibwrite.c
 - gpiberr.c
 - gpibread.c
 - gpibwait.c
 - decl-32.h (provided by National Instruments)
 - gpib-32.obj (provided by National Instruments)
4. To compile and link your sample program, choose "BUILD ALL" from the Build menu.
5. To run the sample program, choose RUN from the "BUILD" menu.

Miscellaneous

Character Charts

Appendix A: ASCII & GPIB Code Chart

B7 B6 B5 BITS B4 B3 B2 B1	0 0 0		0 0 1		0 1 0		0 1 1		1 0 0		1 0 1		1 1 0		1 1 1	
	CONTROL				NUMBERS SYMBOLS				UPPER CASE				LOWER CASE			
0 0 0 0	0	NUL	20	DLE	40	LA0	60	LA16	100	TA0	120	TA16	140	SA0	160	SA16
0 0 0 1	1	SOH	21	DC1	41	LA1	61	LA17	101	TA1	121	TA17	141	SA1	161	SA17
0 0 1 0	2	STX	22	DC2	42	LA2	62	LA18	102	TA2	122	TA18	142	SA2	162	SA18
0 0 1 1	3	ETX	23	DC3	43	LA3	63	LA19	103	TA3	123	TA19	143	SA3	163	SA19
0 1 0 0	4	SDC EDT			44	LA4	64	LA20	104	TA4	124	TA20	144	SA4	164	SA20
0 1 0 1	5	PPC ENQ	25	PPU NAK	45	LA5	65	LA21	105	TA5	125	TA21	145	SA5	165	SA21
0 1 1 0	6	ACK	26	SYN	46	LA6	66	LA22	106	TA6	126	TA22	146	SA6	166	SA22
0 1 1 1	7	BEL	27	ETB	47	LA7	67	LA23	107	TA7	127	TA23	147	SA7	167	SA23
1 0 0 0	8	GET BS	30	SPE CAN	50	LA8	70	LA24	110	TA8	130	TA24	150	SA8	170	SA24
1 0 0 1	9	TCT HT	31	SPD EM	51	LA9	71	LA25	111	TA9	131	TA25	151	SA9	171	SA25
1 0 1 0	10	LF A	32	EUB 2A	52	LA10	72	LA26	112	TA10	132	TA26	152	SA10	172	SA26
1 0 1 1	11	VT B	33	ESC 1B	53	LA11	73	LA27	113	TA11	133	TA27	153	SA11	173	SA27
1 1 0 0	12	FF C	34	FB 1C	54	LA12	74	LA28	114	TA12	134	TA28	154	SA12	174	SA28
1 1 0 1	13	CR D	35	GS 1D	55	LA13	75	LA29	115	TA13	135	TA29	155	SA13	175	SA29
1 1 1 0	14	SO E	36	RS 1E	56	LA14	76	LA30	116	TA14	136	TA30	156	SA14	176	SA30
1 1 1 1	15	SI F	37	US 1F	57	LA15	77	UNL ?	117	TA15 D	137	UNT	157	SA15 o	177	
		ADDRESSED COMMANDS		UNIVERSAL COMMANDS		LISTEN ADDRESSES		TALK ADDRESSES		SECONDARY ADDRESSES OR COMMANDS						

KEY octal → 5 → PPC → GPIB code (with ATN asserted)
 hex → 5 → ENQ → ASCII character
 → 5 → decimal

Tektronix
 REF: ANSI STD X3.4-1977
 IEEE STD 488.1-1987
 ISO STD 646-2973

Reserved Words

This is a list of reserved words for the TDS7000 Series oscilloscopes. Capital letters identify the required minimum spelling. For the most robust code use the full spelling, since spelling rules may change over time and from instrument model to instrument model.

*CAL EXTAtten

*CAL	ATARXG1	B3ZS	DEGrees
*CLS	ATARXG2	B6ZS	DELay
*DDT	ATARXG3	B8ZS	DELEte
*ESE	ATATXG1	CALibrate	DELTA
*ESR	ATATXG2	CAREa	DELTAtime
*IDN	ATATXG3	CATalog	DESE
*LRN	AUTO	CENTER	DESKew
*OPC	AUTOAdjust	CLAss	DESTination
*OPT	AUTOBright	CLOCK	DIR
*PSC	AUTOSet	CMDBatch	DIREction
*PUD	AUX	CMEan	DISPlay
*RCL	AUXiliary	CMI	DISTDUty
*RST	AUXOut	COLOr	DIVisions
*SAV	AVERage	COMMunication	DOTs
*SDS	BACKWards	COMPLetion	DS2RATECoax
*SRE	BANdwidth	CONTRol	DS2RATESymmetrical
*STB	BEEP	COPY	ECL
*TRG	BELI	COUNT	EDGE
*WAI	BETWeen	COUPling	EITher
ABSolute	BIT_Nr	CRMs	ENCdg
AC	BLACKMANHarris	CROSSHair	ENET100STP
ACQuire	BN_Fmt	CURSor	ENET100UTP
ACTUal	BOTh	CURVe	ENET1250
ALias	BOX	DATA	ENVELOpe
ALL	BOXPcnt	DATE	EVENT
ALLev	BURst	DB	EVENTS
AMI	BUSY	DBM	EVMsg
AMPitude	BY	DC	EVQty
AND	BYT_Nr	DEFAult	EXECUTE
AREa	BYT_Or	DEFine	EXTAtten

EXTDBatten ITENSIFied

EXTDBatten	FC1063E_AGR	FUNcTion	HERtz
EXTINCTDB	FC1063E_AGT	FW1394BS1600B	HFRej
EXTINCTPCT	FC1063E_NBT	FW1394BS1600BT1	HIGH
EXTINCTRATIO	FC1063E_NDT	FW1394BS1600BT2	HIGHLIGHTHits
EXTUnits	FC1063_NGT	FW1394BS400B	HIGHLimit
EYEHeight	FC2125	FW1394BS400BT1	HIRes
EYEWidth	FC2125E_ABR	FW1394BS400BT2	HIStoqram
E1Coax	FC2125E_ABT	FW1394BS800B	HITS
E1Symmetrical	FC2125E_ADR	FW1394BS800BT1	HOLDoff
E4_0	FC2125E_ADT	FW1394BS800BT2	HOLDTime
E4_1	FC2125E_AGR	GAIN	HORizontal
FACTory	FC2125E_AGT	GATEPOS	HPOS<x>
FAILURES	FC2125E_NBT	GATEWIDTH	HSCale
FALL	FC2125E_NDT	GATing	HST1
FASTAcq	FC2125E_NGT	GAUSSian	HST2
FASTERthan	FILEName	GLItch	HST3
FASTframe	FILESystem	GPIb	HST4
FC133	FILTer	GRAticule	HST5
FC133E	FLATTOP2	GRId	HST6
FC266	FORMat	GROUPDelay	HTRIGPOS
FC266E	FORWards	G703DS3	ID
FC531	FRame	HALT	IMMed
FC531E	FREQuency	HAMMING	INDEpendent
FC1063	FST1	HANNing	INFInite
FC1063E	FST2	HARDCopy	INFPersist
FC1063Draft	FST3	HBArs	INF2_5G
FC1063E_ABR	FST4	HDB3	INF2_5GE
FC1063E_ABT	FST5	HDELTA	INHERIT
FC1063E_ADR	FST6	HDR	INPut
FC1063E_ADT	FULI	HEADer	INTENSIFied

INTENSITY RESISTANCE

INTENSITY	METHOD	OCCURS	POLARITY
INVERT	MID	OC1	PORT
KAISSERBessel	MID2	OC3	POSITION
LABEL	MINIMUM	OC12	POSITIVE
LAST	MINMAX	OC48	POVershoot
LENGTH	MKDir	OFF	PRESAMPBITS
LESSLimit	MLT3	OFFSet	PRINT
LESSThan	MODE	OFFSETAdj	PROBE
LEVEL	MONOGRAY	ON	PROBECal
LFRej	MONOGREEN	OR	PROBEFunc
LINEAr	MORELimit	OUTside	PTOP
LOCK	MOREThan	PAIred	PT_Fmt
LOG	NAME	PALETTE	PT_Off
LOGIc	NAND	PASSWord	PULse
LONG	NCROSS	PATtern	PWIdth
LOOP	NDUty	PATTERNBITS	QFACTOR
LOOPS	NEGAtive	PBASE	RADIans
LOW	NEWpass	PCROSS	RATE32Mbit
LOWLimit	NOISErej	PCTCROSS	RATE97Mbit
MAG	NONE	PDUty	RECAI
MAIn	NOR	PEAKdetect	RECOrdlength
MANual	NORMAL	PEAKHits	RECTANGular
MARgin	NOVershoot	PERCENT	REF
MASK	NRZ	PERIod	REFCOLOR
MASKPRE	NR_Pt	PERSistence	REFLevel
MATHCOLOR	NUMAcq	PHAsE	REFLEVELOffset
MAXimum	NUMAVg	PKPKJitter	REPeat
MEAN	NUMEnv	PKPKNoise	RESBw
MEASUrement	NUMSAMPles	PK2Pk	RESET
MEDIAn	NWIdth	POINTS	RESistance

REsults ZOOm

REsults	SIGMA2	TTL	XPOS
RMS	SIGMA3	TYPe	XUNit
RMSJitter	SIXSigmajit	UNDEFINED	XY
RMSNoise	SIZE	UNIts	XYZ
ROLL	SLOWERthan	UNLock	XZEro
ROSc	SNAp	UNWRap	YUNit
RUNSTop	SNRatio	USERMask	YOFF
RUNT	SRQ	VALue	YPOS
SAMPlE	STANdard	VARpersist	YT
SAMPLERate	STATE	VBArS	YUNit
SAVe	STAYSHigh	VDELTA	YZEro
SAVEWFM	STAYSLow	VERBoSe	ZOOm
SCALE	STM1E_0	VERTical	
SCOPEAPP	STM1E_1	VIOLATIONS	
SCREENSAVER	STOPAfter	VOFFSet	
SCREENSAVERDELAY	SUPPReSS	VPOS	
SECOnds	TEKEXPonential	VSCAle	
SEG	TEMPErature	WAVEFORM	
SELEct	TESt	WAVEFORMS	
SELECTED	THReShold	WEIghting	
SEQuence	TIMe	WFId	
SERIAL	TIMEOut	WFMDB	
SERIALTRIG	TIMEStamp	WFMInpre	
SERnumber	TRACk	WFMOuPre	
SET	TRANsition	WHEn	
SETHold	TRIGBar	WIDERthan	
SETTime	TRIGger	WIDth	
SETUp	TRIGIF	WINDOW	
SHORT	TRIGT	WIThin	
SIGMA1	TRUe	XINcr	

Factory Default Setup Values

Overview

This topic contains tables that list the default setup values by command group. These are the values that the instrument sets when you press the DEFAULT SETUP front-panel button or send the FACTORY command. Only those commands that have values set by the DEFAULT SETUP function are listed in these tables.

Acquisition

Default Setup Values	
Command	Default
ACQUIRE:MODE	SAMPLE
ACQUIRE:NUMAVG	16
ACQUIRE:NUMENV	10
ACQUIRE:SAMPLINGMODE	11
ACQUIRE:STATE	1 (ON)
ACQUIRE:STOPAFTER	RUNSTOP
FASTACQ:STATE	0 (OFF)

Cursor

Default Setup Values	
Command	Default
CURSOR:FUNCTION	VBARs
CURSOR:HBARs:POSITION<x>	$\pm 300.0000E-9$
CURSOR:MODE	INDEPENDENT
CURSOR:PAIRED:POSITION<x>	$\pm 800.0000E-9$
CURSOR:SOURCE	CH1
CURSOR:SPLIT:POSITION<x>	$\pm 800.0000E-9$
CURSOR:SPLIT:SOURCE2	CH1
CURSOR:SPLIT:UNITS	BASE
CURSOR:STATE	OFF
CURSOR:VBARs:POSITION<x>	$\pm 800.0000E-9$
CURSOR:VBARs:UNITS	SECONDS

Display

Default Setup Values	
Command	Default
DISplay:CLOCK	1
DISplay:COLOR:MATHCOLOR	DEFAULT
DISplay:COLOR:PALETTE	NORMAL
DISplay:COLOR:REFCOLOR	DEFAULT
DISplay:FILTer	SINX
DISplay:FORMat	YI
DISplay:INTENSIty:AUTOBright	1 (ON)
DISplay:INTENSIty:SCREENSAVER	1 (ON)
DISplay:INTENSIty:SCREENSAVERDELA	28800
DISplay:INTENSIty:WAVEform	75.0000
DISplay:PERSIstence	OFF
DISplay:STYle	VEctors
DISplay:TRIGBar	SHORT
DISplay:TRIGT	1 (ON)
DISplay:VARPersist	500.0000E-3

Hardcopy

Default Setup Values	
Command	Default
HARDCopy:FILENAME	""
HARDCopy:PORT	FILE

Histogram

Default Setup Values	
Command	Default
HIStogram:BOXPcnt	30.0000, 25.1000, 70.0000, 75.2000
HIStogram:DISplay	0 (OFF)
HIStogram:FUNCTION	HORizontal
HIStogram:SIze	2.0000
HIStogram:SOUrce	CH1
HIStogram:STATe	0 (OFF)

Horizontal

Default Setup Values	
Command	Default
HORizontal:FASTframe:COUNT	2
HORizontal:FASTframe:LENGth	5000
HORizontal:FASTframe:REF:FRAME	1
HORizontal:FASTframe:REF:SOURce	CH1
HORizontal:FASTframe:SELECTEDtra>	CH1-4 2 MATH1-4 2 REF1-4 2
HORizontal:FASTframe:STATE	0
HORizontal:FASTframe:TRACK	LIVE
HORizontal:RECOrdlength	5000
HORizontal:RESOLution	5000
HORizontal:ROLL	AUTO
HORizontal:SPECTral:RECOrdlength	5000
HORizontal:MAIN]DElay:MODE	0 (OFF)
HORizontal:MAIN]DElay:POSition	50.0000
HORizontal:MAIN]DElay:TIME	0.0000
HORizontal:MAIN]POSition	50.0000
HORizontal:MAIN]SAMPLERate	2.5000E+9
HORizontal:MAIN]SCALE	200.0000E-9

Math

Default Setup Values	
Command	Default
MATH<x>:DEFine	" "
MATH<x>:LABEL:NAME	" "
MATH<x>:LABEL:XPOS	5
MATH<x>:LABEL:YPOS	MATH1 65 MATH2 80 MATH3 95 MATH4 110
MATH<x>:NUMavg	2
MATH<x>:POSition	0.0000
MATH<x>:SCALE	1.0000
MATH<X>:SPECTral:CENTER	625.0000E+06
MATH<X>:SPECTral:GATEPOS	0.0000
MATH<X>:SPECTral:GATEWIDTH	2.0000E-06
MATH<X>:SPECTral:LOCK	0 (OFF)
MATH<X>:SPECTral:MAG	DB
MATH<X>:SPECTral:PHASE	DEGREES
MATH<X>:SPECTral:REFLEVEL	20.0000
MATH<X>:SPECTral:REFLEVELOffset	223.6000E-03
MATH<X>:SPECTral:RESBw	1.0000E+06
MATH<X>:SPECTral:SPAN	1.2500E+09
MATH<X>:SPECTral:SUPPress	-35.0000
MATH<X>:SPECTral:UNWRap	0 (OFF)
MATH<X>:SPECTral:WINDOW	GAUSSIAN

Measurement

Default Setup Values	
Command	Default
MEASUrement:GATING	0 (OFF)
MEASUrement:METhod	HIStogram
MEASUrement:IMMed:TYPe	undefined
MEASUrement:IMMed:SOURCE[1]	CH1
MEASUrement:IMMed:SOURCE2	CH1
MEASUrement:IMMed:DELAy:EDGE[1]	RISe
MEASUrement:IMMed:DELAy:EDGE2	RISe
MEASUrement:IMMed:DELAy:DIREction	FORWards
MEASUrement: MEAS<x>:STATE	0 (OFF)
MEASUrement:MEAS<x>:SOURCE[1]	CH1
MEASUrement:MEAS<x>:SOURCE2	CH1
MEASUrement:MEAS<x>:DELAy:EDGE[1]	RISe
MEASUrement:MEAS<x>:DELAy:EDGE2	RISe
MEASUrement:MEAS<x>:DELAy:DIREct	FORWards
MEASUrement:REFLevel:METhod	Percent
MEASUrement:REFLevel:ABSolute:HIGh	0.0000
MEASUrement:REFLevel:ABSolute:LOW	0.0000
MEASUrement:REFLevel:ABSolute:MID	0.0000
MEASUrement:REFLevel:ABSolute:MID2	0.0000
MEASUrement:REFLevel:PERCent:HIGh	90.0000
MEASUrement:REFLevel:PERCent:LOW	10.0000
MEASUrement:REFLevel:PERCent:MID	50.0000
MEASUrement:REFLevel:PERCent:MID2	50.0000
MEASUrement:STATIstics:MODE	OFF
MEASUrement:STATIstics:WEIghting	32

Miscellaneous

Default Setup Values	
Command	Default
AUXout:EDGE	FALIng
AUXout:SOURce	ATRIGger
CMDBatch	1 (ON)
HEADer	1 (ON)
LOCK	NONE
VERBoSe	1 (ON)

Trigger

Default Setup Values	
Command	Default
TRIGger:A:MODE	AUTO
TRIGger:A:TYPE	EDGE
TRIGger:A:LEVEL	0.0000
TRIGger:A:HOLDoff:BY	DEFAULT
TRIGger:A:HOLDoff:TIME	250.0000E-09
TRIGger:A:EDGE:SOURce	CH1
TRIGger:A:EDGE:COUPLing	DC
TRIGger:A:EDGE:SLOpe	RISe
TRIGger:A:LOGic:CLAss	PATtern
TRIGger:A:LOGic:FUNCTion	AND
TRIGger:A:LOGic:THReshold:CH<x>	CH1 1.2000 CH2 1.2000 CH3 1.2000 CH4 1.2000
TRIGger:A:LOGic:INPut:CH<x>	CH1 HIGH CH2 X CH3 X
TRIGger:A:LOGic:PAtern:INPut:CH4	X
TRIGger:A:LOGic:PAtern:WHEn	TRUe
TRIGger:A:LOGic:PAtern:WHEn:LESSLim	5.0000E-09
TRIGger:A:LOGic:PAtern:WHEn:MORELim	5.0000E-09
TRIGger:A:LOGic:SETHold:CLOCK:EDGE	RISe
TRIGger:A:LOGic:SETHold:CLOCK:THResh	1.2000
TRIGger:A:LOGic:SETHold:CLOCK:SOURce	CH2
TRIGger:A:LOGic:SETHold:DAta:THResho	1.2000
TRIGger:A:LOGic:SETHold:DAta:SOURce	CH1
TRIGger:A:LOGic:SETHold:HOLDTime	2.0000E-09
TRIGger:A:LOGic:SETHold:SETTime	3.0000E-09
TRIGger:A:LOGic:STATe:INPut:CH4	RISe
TRIGger:A:LOGic:STATe:WHEn	TRUe
TRIGger:A:PULse:CLAss	GLitch
TRIGger:A:PULse:SOURce	CH1
TRIGger:A:PULse:GLitch:WIDth	2.0000E-09
TRIGger:A:PULse:GLitch:TRIGIF	ACCEpt
TRIGger:A:PULse:GLitch:POLarity	POSITIVe
TRIGger:A:PULse:RUNI:POLarity	POSITIVe
TRIGger:A:PULse:RUNI:THReshold:HIGH	1.2000E-00
TRIGger:A:PULse:RUNI:THReshold:LOW	800.0000E-03
TRIGger:A:PULse:RUNI:WHEn	OCCurs
TRIGger:A:PULse:RUNI:WIDth	2.0000E-09
TRIGger:A:PULse:TRANSition:DELTAime	2.0000E-09
TRIGger:A:PULse:TRANSition:POLarity	POSITIVe
TRIGger:A:PULse:TRANSition:THReshold:F	1.2000E-00
TRIGger:A:PULse:TRANSition:THReshold:L	800.0000E-03
TRIGger:A:PULse:TRANSition:WHEn	FASTERthan
TRIGger:A:PULse:WIDth:LOWLimit	2.0000E-09
TRIGger:A:PULse:WIDth:HIGHLimit	2.0000E-09
TRIGger:A:PULse:WIDth:WHEn	WITHIN
TRIGger:A:PULse:WIDth:POLarity	POSITIVe
TRIGger:A:PULse:TIMEOut:POLarity	STAYSHigh
TRIGger:A:PULse:TIMEOut:TIME	2.0000E-09
TRIGger:B:STATe	0 (OFF)
TRIGger:B:TYPE	EDGE
TRIGger:B:LEVEL	0.0000
TRIGger:B:BY	EVENTS
TRIGger:B:EDGE:SOURce	CH1
TRIGger:B:EDGE:SLOpe	RISe
TRIGger:B:EDGE:COUPLing	DC
TRIGger:B:TIME	16.0000E-09
TRIGger:B:EVENTS:COUNt	2

Vertical

Default Setup Values	
Command	Default
CH<x>:BANdwidth	CH1-4 1.0000E+09
CH<x>:COUPling	CH1-4 DC
CH<x>:DESKew	CH1-4 0.0000
CH<x>:OFFSet	CH1-4 0.0000
CH<x>:POSition	CH1-4 0.0000
CH<x>:SCAlE	CH1-4 100.0000E-03
CH<x>:TERmination	CH1-4 1.0000E-06
CH<x>:PROBEFunc:EXIatten	CH1-4 1.0000
CH<x>:PROBEFunc:EXTUnits	CH1-4 "None"
CH<x>:LABEL:NAME	CH1-4 ""
CH<x>:LABEL:XPOS	CH1-4 5
CH<x>:LABEL:YPOS	CH1 5 CH2 20 CH3 35 CH4 50
CURVe	
DATA:DESTination	REF1
DATA:ENCdg	RIBinary
DATA:SOURce	CH1
DATA:STARt	1
DATA:STOP	500
REF<x>:LABel:NAME	""
REF<x>:LABel:XPOS	REF1-45
REF<x>:LABel:YPOS	REF1 125 REF2 140 REF3 155 REF4 170
REF<x>:SCAlE	1.0000
REF<x>:POSition	0.0000
SElect:CH<x>	CH1 1 CH2 0 CH3 0 CH4 0
SElect:CONTRol	CH1
SElect:MATH<x>	MATH1-40
SElect:REF<x>	REF1-4 0

Waveform

Default Setup Values	
Command	Default
WFId	"Ch1, DC coupling, 100.0mV/div, 2000.0ns/div, 5000 points, Sample mode"
WFMinpre:BIT_Nr	8
WFMinpre:BN_Fmt	HI
WFMinpre:BYT_Nr	1
WFMinpre:BYT_Or	MSB
WFMinpre:ENCdg	BIN
WFMinpre:NR_Pt	500
WFMinpre:PI_Fmt	Y
WFMinpre:PI_Off	0
WFMinpre:XINcr	2.0000E-06
WFMinpre:XUNit	"s"
WFMinpre:XZEro	0.000
WFMinpre:YMUlt	1.0000E-03
WFMinpre:YOff	0.0000
WFMinpre:YUNit	"V"
WFMinpre:YZEro	0.0000
WFMOupre:BIT_Nr	8
WFMOupre:BN_Fmt	HI
WFMOupre:BYT_Nr	1
WFMOupre:BYT_Or	MSB
WFMOupre:ENCdg	BIN
WFMOupre:NR_Pt	500
WFMOupre:PI_Fmt	Y
WFMOupre:PI_Off	2500
WFMOupre:XINcr	400.0000E-12
WFMOupre:XUNit	"s"
WFMOupre:XZEro	0.000
WFMOupre:YMUlt	4.0000E-03
WFMOupre:YOff	0.0000
WFMOupre:YUNit	"V"
WFMOupre:YZEro	0.0000

Zoom

Default Setup Values	
Command	Default
ZOOm:MODe	0 (OFF)
ZOOm:HORizontal:LOCK	ALL
ZOOm:wfM>HORizontal:POSition	CH1-4 50.0000 MATH1-4 50.0000 REF1-4 50.0000
ZOOm:wfM>HORizontal:SCAle	CH1-4 2 MATH1-4 2 REF1-4 2
ZOOm:wfM>VERtical:POSition	CH1-4 0.0000 MATH1-4 0.0000 REF1-4 0.0000
ZOOm:wfM>VERtical:SCAle	CH1-4 1.0000 MATH1-4 1.0000 REF1-4 1.0000

GPIB Interface Specifications

This topic describes details of the GPIB remote interface of the instrument. Normally, you will not need this information to use the instrument, but this information may be useful if you are connecting to controllers with unusual configurations.

Interface Messages

The following table shows the standard interface messages that the instrument supports.

Standard Interface Messages	
Message	GPIB
DCL	Yes
GET	Yes
GTL	Yes
LLO	Yes
PPC	No
PPD	No
PPE	No
PPU	No
SDC	Yes
SPD	Yes
SPE	Yes
TCT	No
UNL	Yes
UNT	Yes
Listen Addresses	Yes
Talk Addresses	Yes

GPIB Functions

The following table lists the GPIB interface functions and electrical functions subsets supported by this instrument with brief descriptions of each.

GPIB Functions		
Interface Function	Subset	Meaning
Acceptor Handshake	AH1	The instrument can receive multiline messages across the GPIB from other devices.
Controller	C0	No Controller capability; the instrument cannot control other devices.
Device Clear	DC1	The instrument can respond to both the DCL (Device Clear) interface message and to the Selected Device Clear (SDC) interface message when the instrument is listen-addressed.
Device Trigger	DT1	Device Trigger capability; the instrument does respond to the GET (Group Execute Trigger) interface message.
Electrical	E2	The instrument uses tri-state buffers, which are optimal for high-speed data transfer.
Listener	L4	The instrument becomes a listener when it detects the listen address being sent over the bus with the ATN line asserted. The instrument ceases to be a listener and becomes a talker when it detects the talk address being sent over the bus with the ATN line asserted.
Parallel Poll	PP0	No Parallel Poll capability; the instrument does not respond to PPC (Parallel Poll Configure), PPD (Parallel Poll Disable), PPE (Parallel Poll Enable), or PPU (Parallel Poll Unconfigure) interface messages, nor does it send a status message when the ATN and EOI lines are asserted simultaneously.
Remote/Local	RL1	The instrument can respond to both GTL (Go To Local) and LLO (Local Lock Out) interface messages.
Service Request	SR1	The instrument can assert the SRQ line to notify the controller in charge that it requires service.
Source Handshake	SH1	The instrument can initiate multiline messages to send across the GPIB to other devices.
Talker	T5	The instrument becomes a talker when it detects the talk address being sent over the bus with the ATN line asserted. The instrument ceases to be a talker and becomes a listener when it detects the listen address being sent over the bus with the ATN line asserted. The instrument also ceases to be a talker when it detects the talk address of another device being sent over the data lines with ATN asserted.