Agilent Technologies 8960 Series 10 E5515B,C Wireless Communications Test Set
Agilent Technologies E1961A AMPS/136 Mobile Test Application

GPIB Command Syntax

Test Application Revision A.04

© Copyright Agilent Technologies 2001
All Rights Reserved

http://www.agilent.com/find/8960support/

Agilent Technologies

Notice

Information contained in this document is subject to change without notice.

All Rights Reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

This material may be reproduced by or for the U.S. Government pursuant to the Copyright License under the clause at DFARS 52.227-7013 (APR 1988).

Agilent Technologies, Inc.
Learning Products Department
24001 E. Mission
Liberty Lake, WA 99019-9599
U.S.A.

Contents

Diagram Conventions 7
Description 8
ABORt 12
AFGenerator 13
CALibration 14
CALL:ACC 15
CALL:AVCTest 16
CALL:CCHannel 17
CALL:CONNected 18
CALL:CONTrol:DOWNlink:FREQuency:AUTO 19
CALL:DCCH 20
CALL:DTC:BURSt 21
CALL:END 22
CALL:DTC:IBERror 23
CALL:FM 24
CALL:HANDoff 25
CALL:MCCode 26
CALL:MS 27
CALL:OPERating 32
CALL:ORIGinate 33
CALL:PAGing:PNUMber 34
CALL:PAGing:REPeat 35

Contents

CALL:POWer 36
CALL:REGister 38
CALL:RFGenerator 39
CALL:SETup 40
CALL:SID 43
CALL:SOCode 44
CALL:STATus 45
CALL:TCHannel 46
DISPlay 47
FETCh:ACPower 48
FETCh:AFANalyzer 50
FETCh:ATXPower 54
FETCh:DPOWer 55
FETCh:DTXPower 57
FETCh:FM 58
FETCh:FSTability 61
FETCh:IQTuning 63
FETCh:LBERror 64
FETCh:MACCuracy 65
INITiate 71
READ 73
RFANalyzer 75

Contents

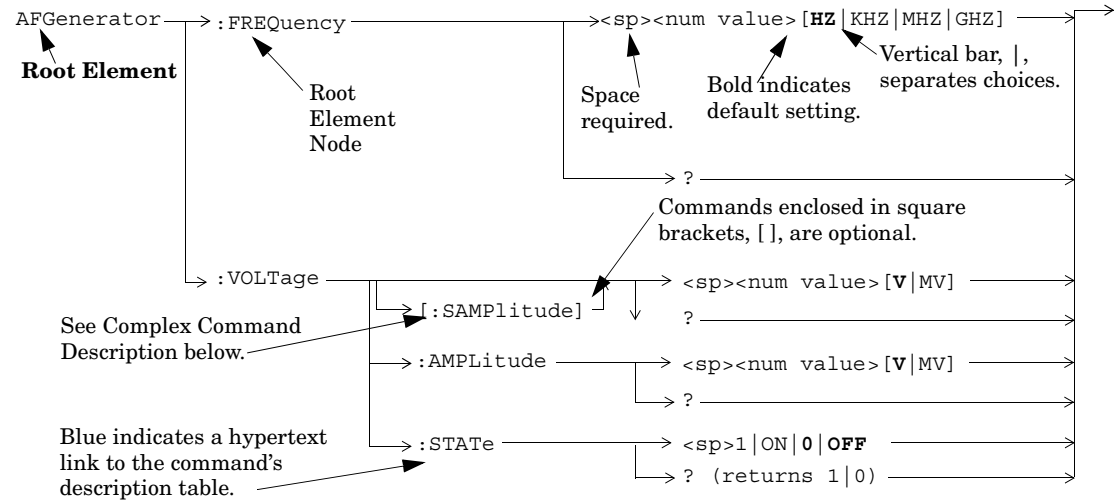
SETup[:ALL]:CONTinuous 78
SETup:ACPower 79
SETup:AFANalyzer 81
SETup:ATXPower 85
SETup:DPOWer 87
SETup:DTXPower 89
SETup:FM 91
SETup:FSTability 95
SETup:IQTuning 97
SETup:LBERror 101
SETup:MACCuracy 102
STATus:OPERation 104
STATus:PRESet 112
STATus:QUEStionable 113
Status Byte Register 126
Standard Event Status Register 129
SYSTem:APPLication 134
SYSTem:BEEPer 136
SYSTem:COMMunicate 137
SYSTem:CONFigure:INFormation:HARDware:VERBose? 138
SYSTem:CORRection 139
SYSTem:CURRent:TA 141

Contents

SYSTem:ERRor? 142
SYSTem:MEASurement:RESet 143
SYSTem:PRESet 144
SYSTem:ROSCillator 145
SYSTem:SYNChronized 146
IEEE 488.2 Common Commands 147
Description 147

Diagram Conventions

Description



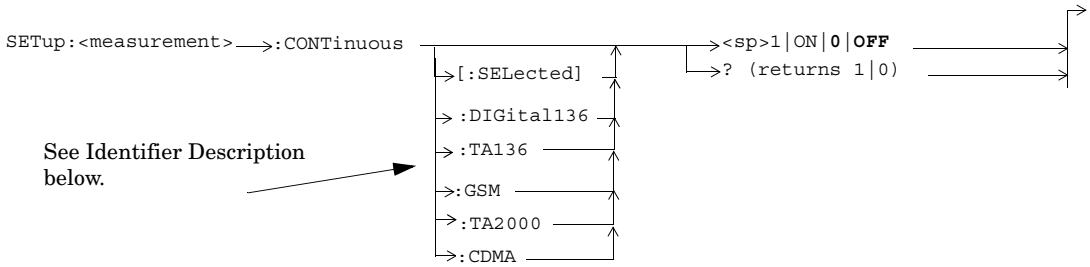

SETup:<measurement>
:CONTinuous
[:SELected]
:DIGital136
:TA136
:GSM
:TA2000
:CDMA
<sp>1|ON|0|OFF
? (returns 1|0)
See Identifier Description below.

Diagram Description

Statement elements are connected by lines. Each line can be followed in only one direction, as indicated by the arrow at the end of the line. Any combination of statement elements that can be generated by starting at the **Root Element** and following the line the **direction of the arrow** is syntactically correct. The drawings show the proper use of spaces. Where spaces are required they are indicated by **<sp>**, otherwise no spaces are allowed between statement elements.

Complex Command Description

A complex command sets the state of the parameter to ON, and is used to set a value for that parameter. These parameters; amplitude, frequency, gain, number, time, and value can be used as a complex command. Refer to the specific command for the parameter that applies.

Identifier Description

Some test applications are able to test more than one radio format. There may be commands/queries that are shared by more than one radio format in the some of these test applications. Identifiers are used to specify the radio format for the command/query. The command/query is sent to the active radio format if you don't use an identifier. An identifier must be used when sending commands to the inactive radio format.

Developing Code

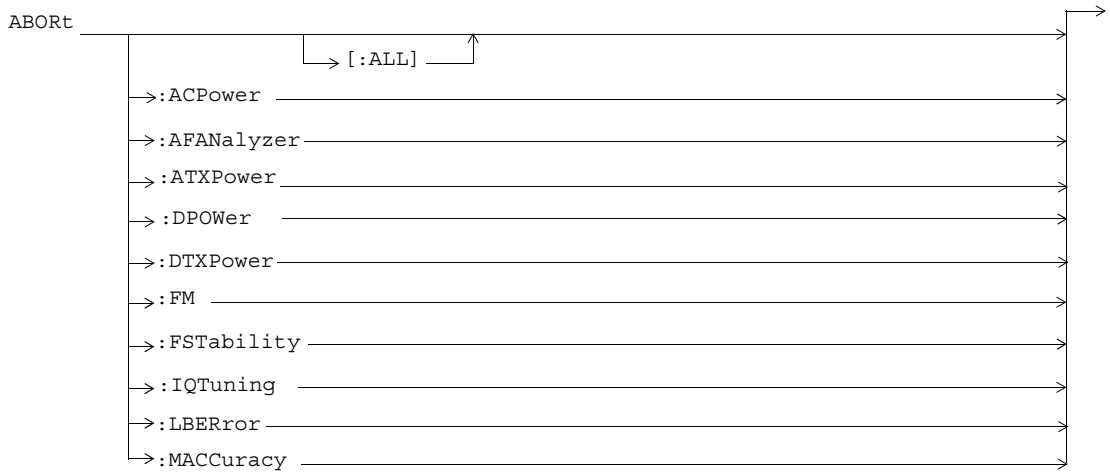
It is recommended that you set the Test Set's operating environment to debug. To set the Test Set debug mode to "ON" use the following syntax:

```
SYSTem:COMMunicate:GPIB:DEBug ON
```

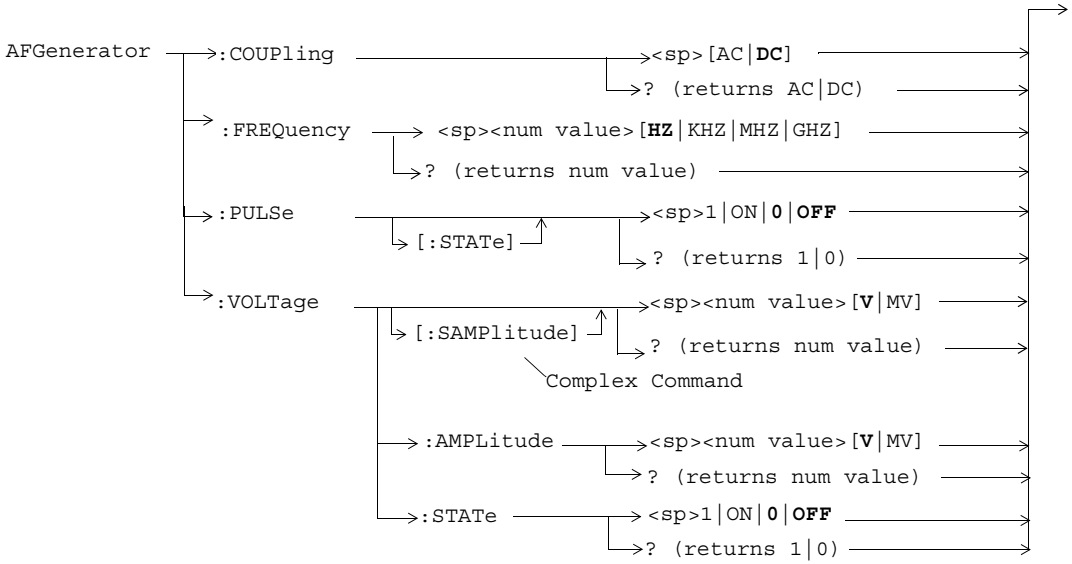
Units-of-Measure

Amplitude (linear)	V
Frequency	Hz
Power (logarithmic)	dBm
Time	s

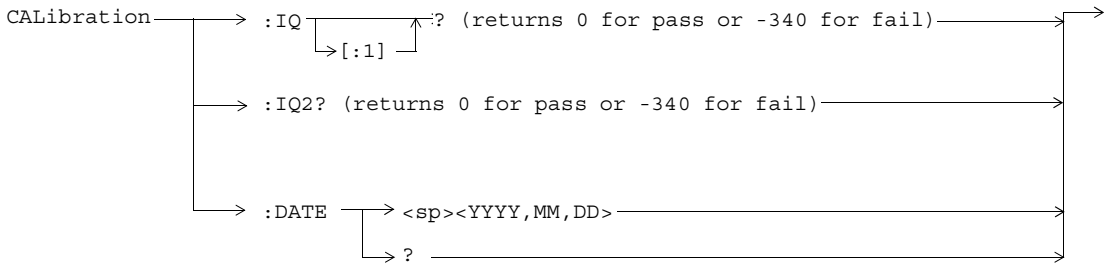
ABORt


ABORt
[:ALL]
:ACPower
:AFANalyzer
:ATXPower
:DPOWer
:DTXPower
:FM
:FSTability
:IQTuning
:LBERror
:MACCuracy

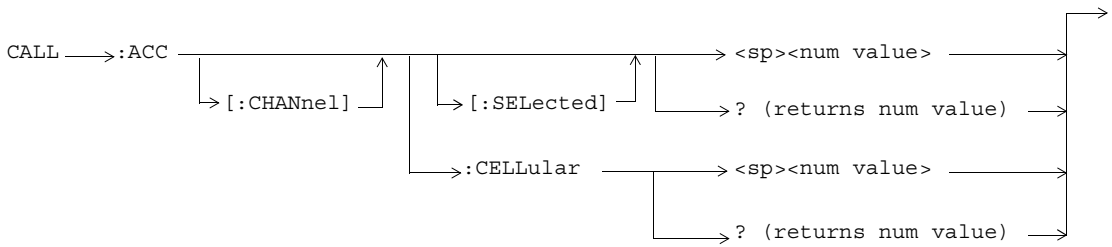
AFGenerator



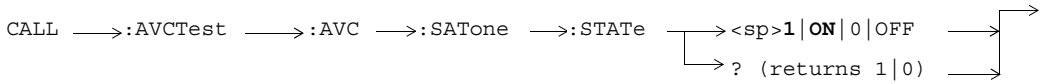
CALibration



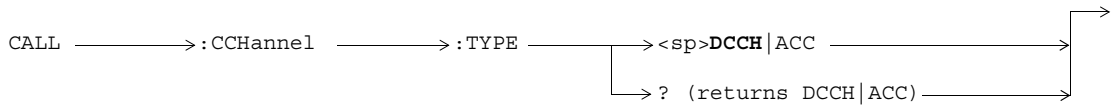
CALL:ACC


CALL
:ACC
[:CHANnel]
[:SELected]
<sp><num value>
? (returns num value)
:CELLular
<sp><num value>
? (returns num value)

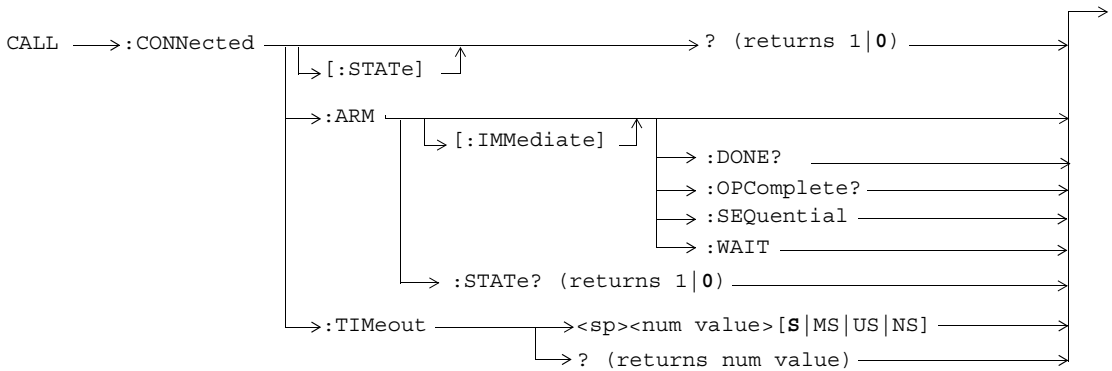
CALL:AVCTest



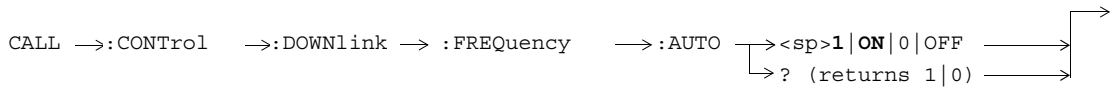
CALL:CCHannel



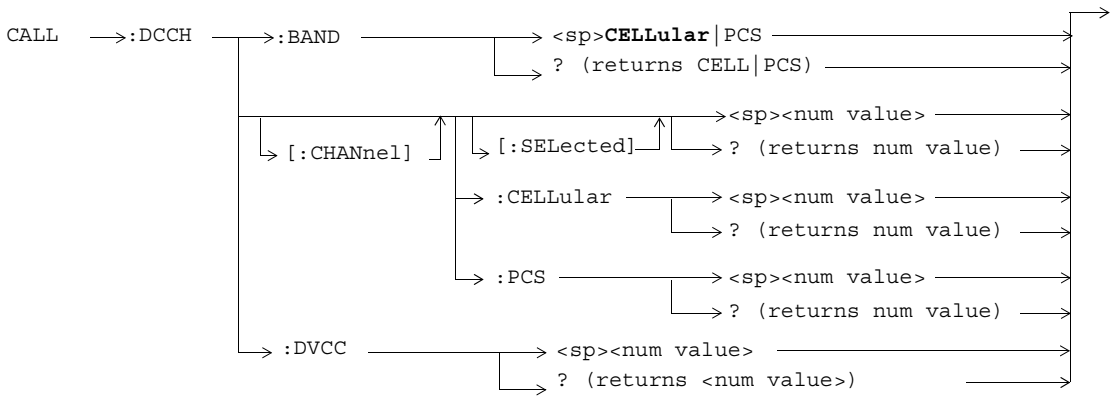
CALL:CONNected



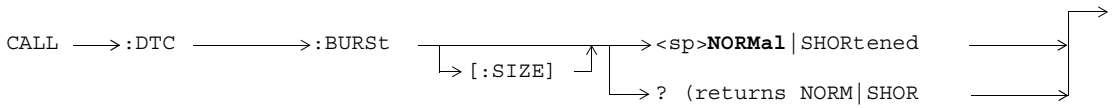
CALL:CONTrol:DOWNlink:FREQuency:AUTO



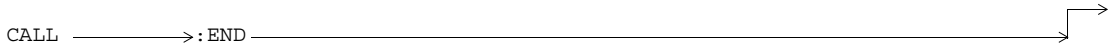
CALL:DCCH


CALL
:DCCH
:BAND
<sp>CELLular|PCS
? (returns CELL|PCS)
[:CHANnel]
[:SELected]
<sp><num value>
? (returns num value)
:CELLular
<sp><num value>
? (returns num value)
:PCS
<sp><num value>
? (returns num value)
:DVCC
<sp><num value>
? (returns <num value>)

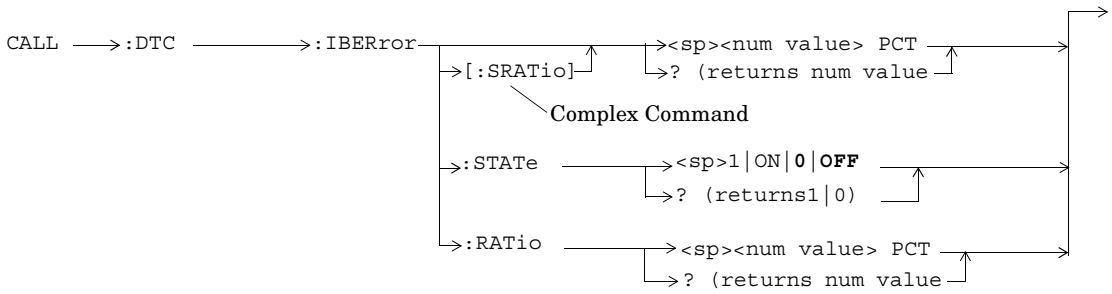
CALL:DTC:BURSt



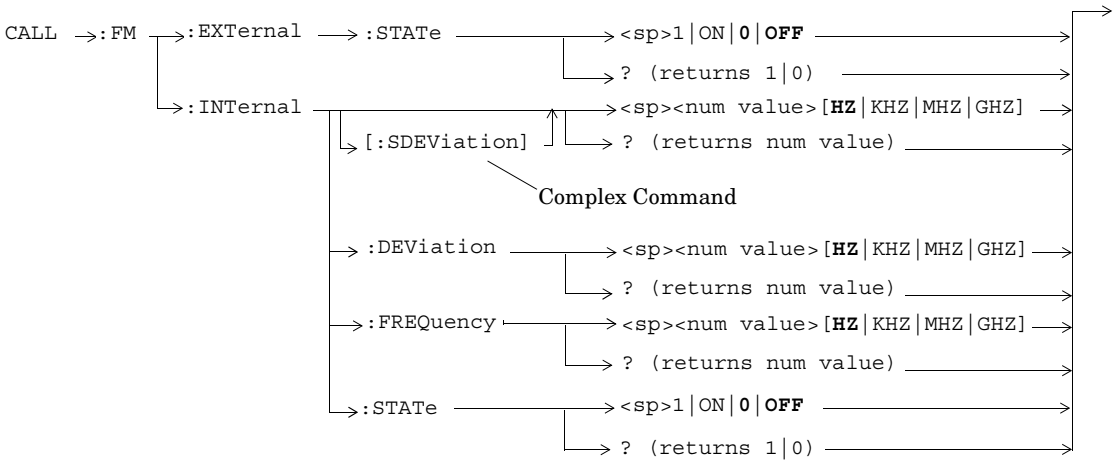
CALL:END



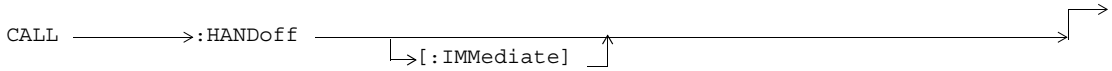
CALL:DTC:IBERror



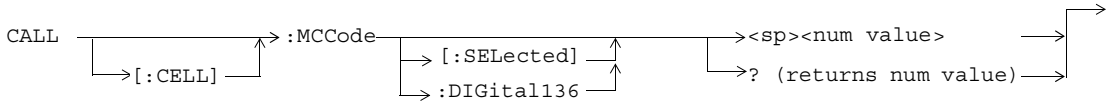
CALL:FM


CALL
:FM
:EXTernal
:STATe
<sp>1|ON|0|OFF
? (returns 1|0)
:INTernal
<sp><num value>[HZ|KHZ|MHZ|GHZ]
[:SDEViation]
? (returns num value)
Complex Command
:DEViation
<sp><num value>[HZ|KHZ|MHZ|GHZ]
? (returns num value)
:FREQuency
<sp><num value>[HZ|KHZ|MHZ|GHZ]
? (returns num value)
:STATe
<sp>1|ON|0|OFF
? (returns 1|0)

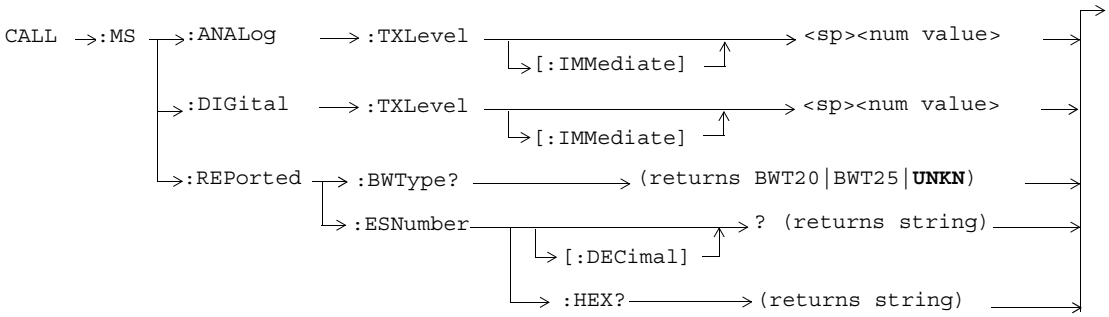
CALL:HANDoff

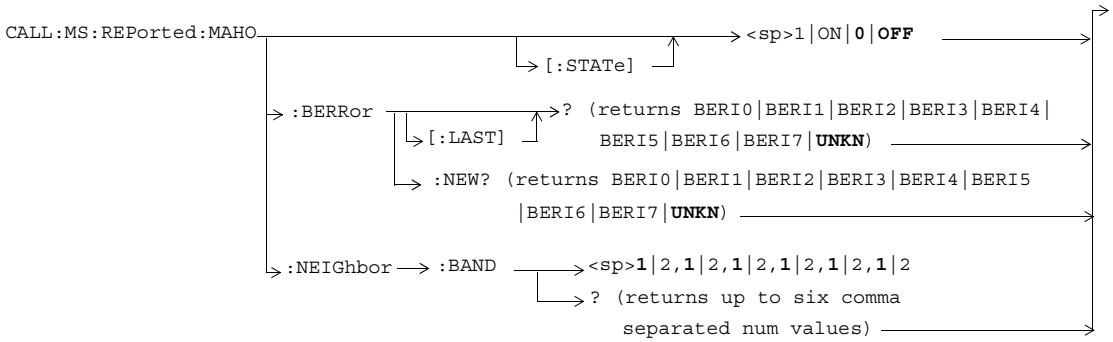


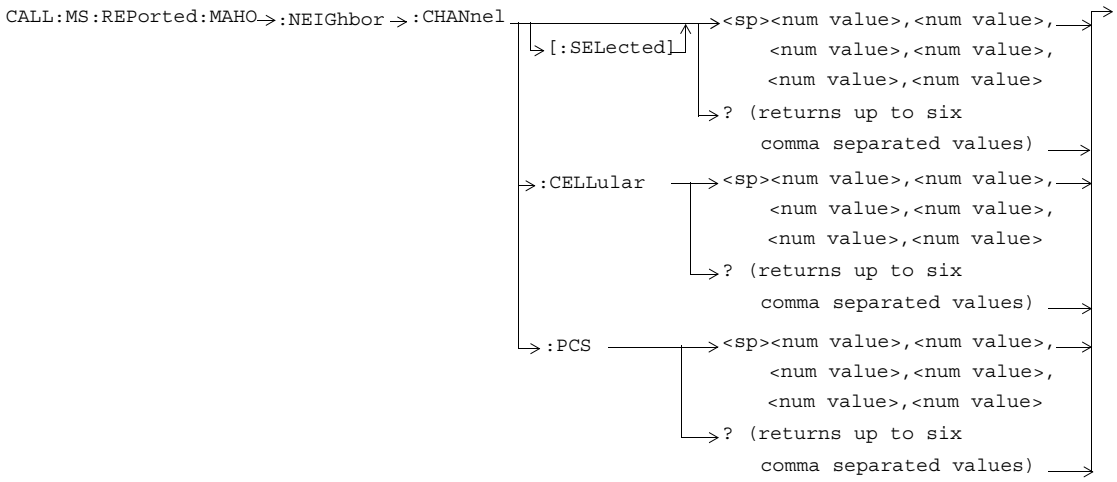
CALL:MCCode

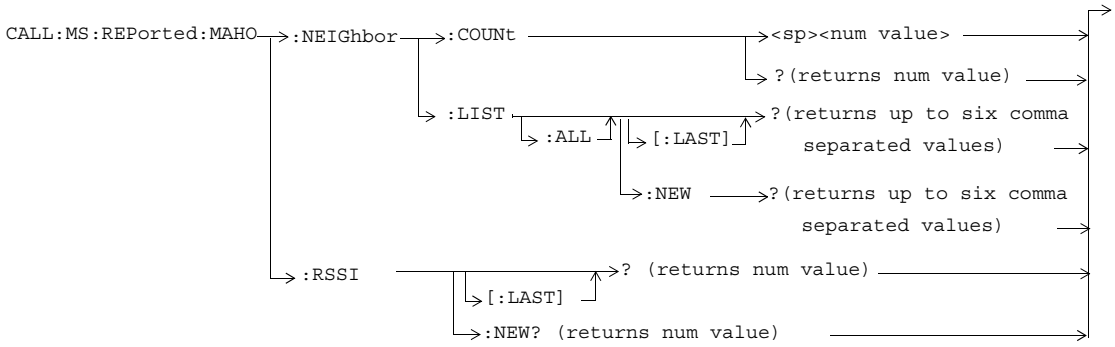

CALL
[:CELL]
:MCCode
[:SELected]
:DIGital136
<sp><num value>
? (returns num value)

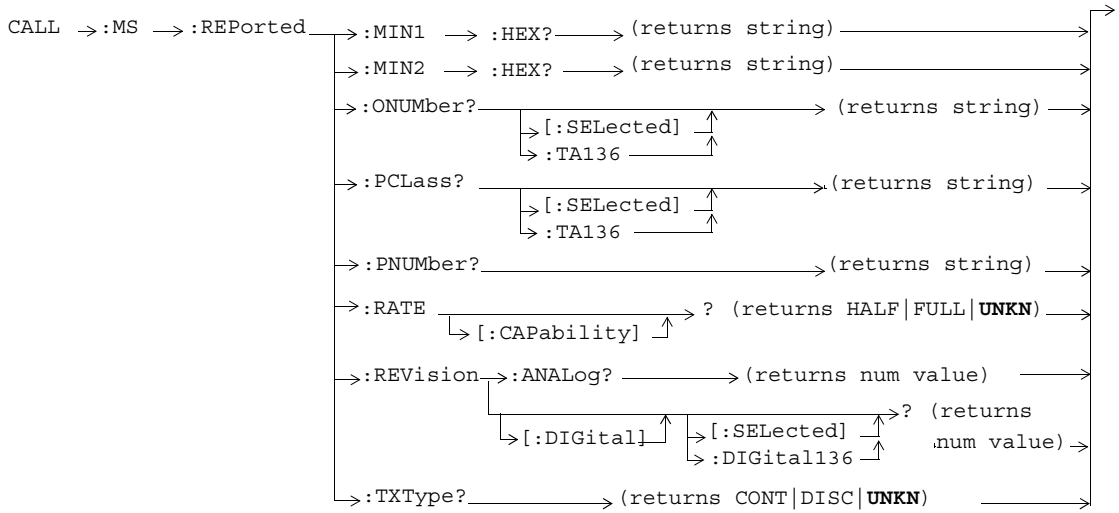
CALL:MS


CALL
:MS
:ANALog
:TXLevel
[:IMMediate]
<sp><num value>
:DIGital
:TXLevel
[:IMMediate]
<sp><num value>
:REPorted
:BWType?
(returns BWT20|BWT25|UNKN)
:ESNumber
? (returns string)
[:DECimal]
:HEX?
(returns string)


CALL:MS:REPorted:MAHO
[:STATe]
<sp>1|ON|0|OFF
:BERRor
[:LAST]
? (returns BERI0|BERI1|BERI2|BERI3|BERI4|
BERI5|BERI6|BERI7|UNKN)
:NEW? (returns BERI0|BERI1|BERI2|BERI3|BERI4|BERI5
|BERI6|BERI7|UNKN)
:NEIGhbor
:BAND
<sp>1|2,1|2,1|2,1|2,1|2,1|2
? (returns up to six comma
separated num values)

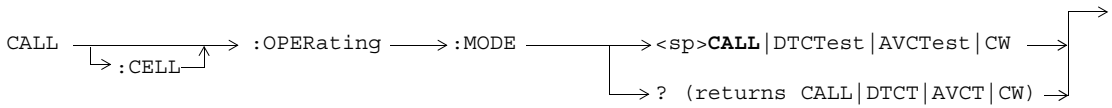

CALL:MS:REPorted:MAHO
:NEIGhbor
:CHANnel
[:SELected]
<sp><num value>,<num value>,
<num value>,<num value>,
<num value>,<num value>
? (returns up to six
comma separated values)
:CELLular
<sp><num value>,<num value>,
<num value>,<num value>,
<num value>,<num value>
? (returns up to six
comma separated values)
:PCS
<sp><num value>,<num value>,
<num value>,<num value>,
<num value>,<num value>
? (returns up to six
comma separated values)


CALL:MS:REPorted:MAHO
:NEIGhbor
:COUNt
<sp><num value>
?(returns num value)
:LIST
?(returns up to six comma separated values)
:ALL
[:LAST]
:NEW
?(returns up to six comma separated values)
:RSSI
? (returns num value)
[:LAST]
:NEW? (returns num value)

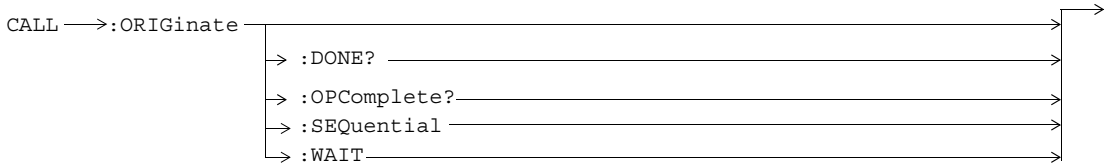


(Not supported in the first release of this mobile test application.)

CALL:OPERating

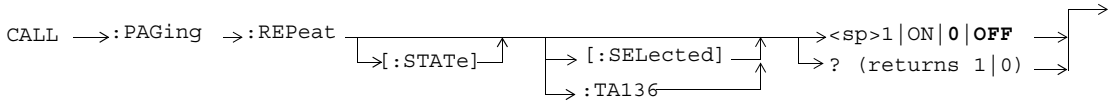


CALL:ORIGinate

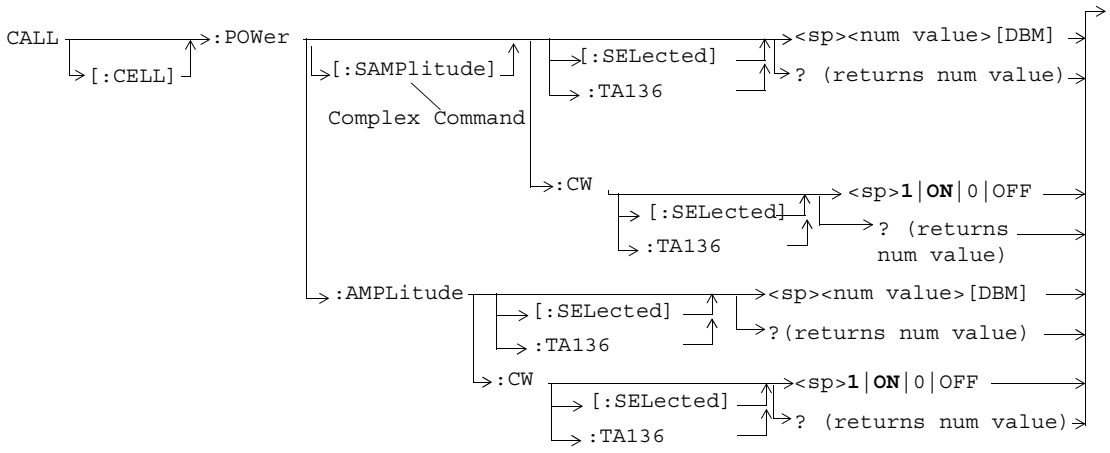


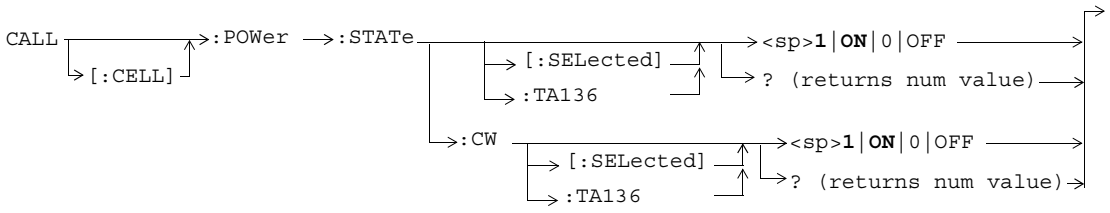
CALL:PAGing:PNUMber

CALL:PAGing:REPeat

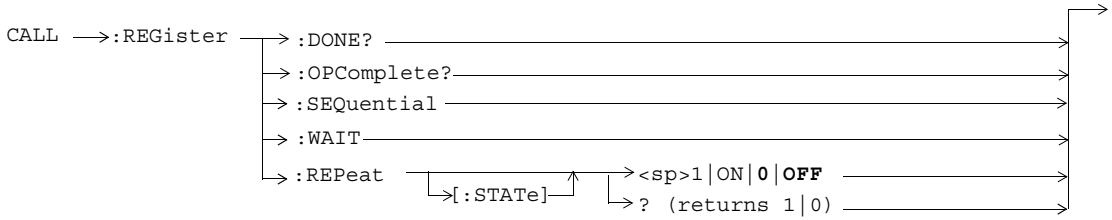


CALL:POWer

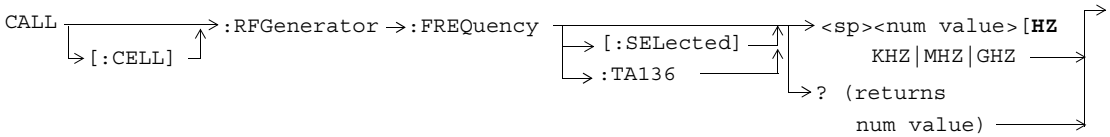

CALL
:POWer
[:CELL]
[:SAMPlitude]
Complex Command
[:SELected]
:TA136
<sp><num value>[DBM]
? (returns num value)
:CW
[:SELected]
:TA136
<sp>1|ON|0|OFF
? (returns num value)
:AMPLitude
[:SELected]
:TA136
<sp><num value>[DBM]
?(returns num value)
:CW
[:SELected]
:TA136
<sp>1|ON|0|OFF
? (returns num value)


CALL
[:CELL]
:POWer
:STATe
[:SELected]
:TA136
<sp>1|ON|0|OFF
? (returns num value)
:CW
[:SELected]
:TA136
<sp>1|ON|0|OFF
? (returns num value)

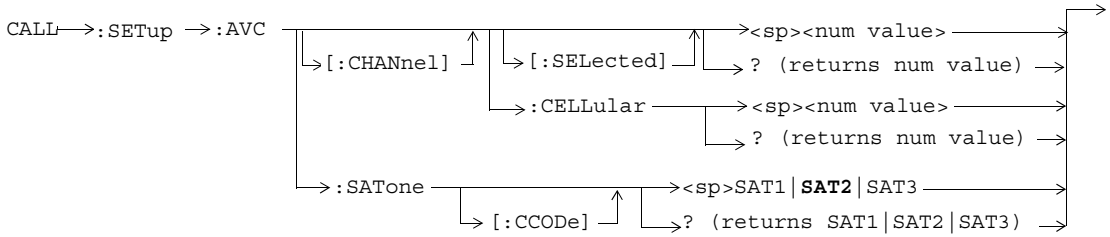
CALL:REGister

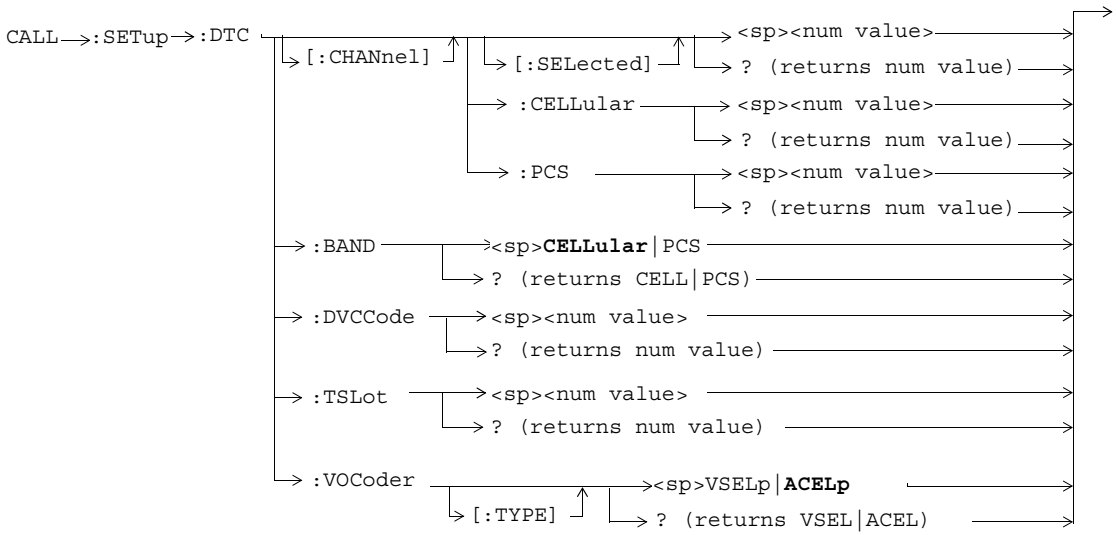


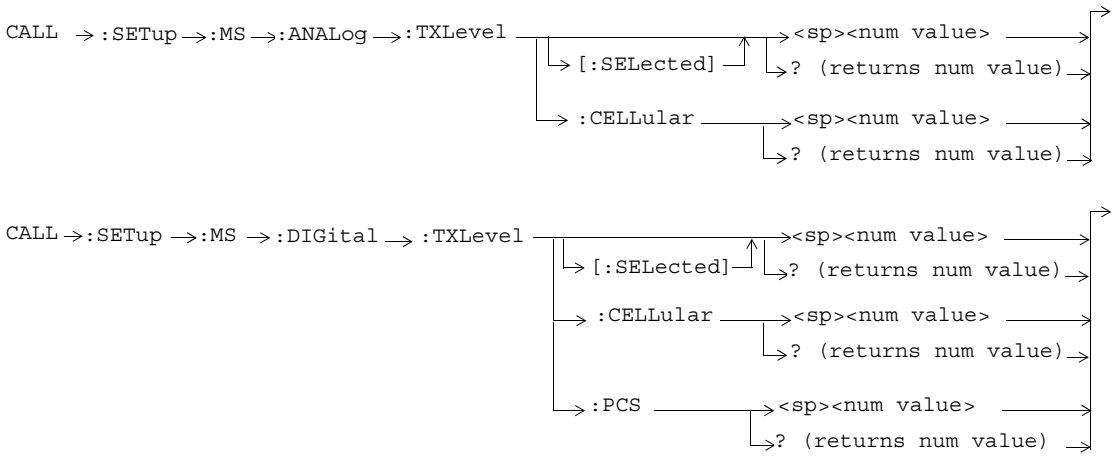
CALL:RFGenerator



CALL:SETup


CALL
:SETup
:AVC
[:CHANnel]
[:SELected]
<sp><num value>
? (returns num value)
:CELLular
<sp><num value>
? (returns num value)
:SATone
[:CCODe]
<sp>SAT1|SAT2|SAT3
? (returns SAT1|SAT2|SAT3)

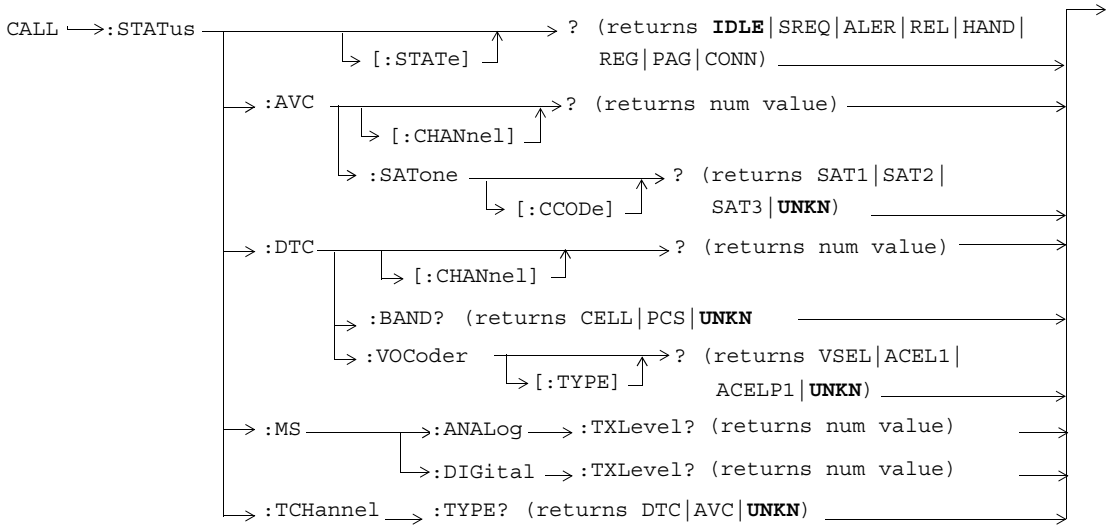

CALL
:SETup
:DTC
[:CHANnel]
[:SELected]
<sp><num value>
? (returns num value)
:CELLular
<sp><num value>
? (returns num value)
:PCS
<sp><num value>
? (returns num value)
:BAND
<sp>CELLular|PCS
? (returns CELL|PCS)
:DVCCode
<sp><num value>
? (returns num value)
:TSLot
<sp><num value>
? (returns num value)
:VOCoder
[:TYPE]
<sp>VSELp|ACELp
? (returns VSEL|ACEL)


CALL
:SETup
:MS
:ANALog
:TXLevel
[:SELected]
<sp><num value>
? (returns num value)
:CELLular
<sp><num value>
? (returns num value)
CALL
:SETup
:MS
:DIGital
:TXLevel
[:SELected]
<sp><num value>
? (returns num value)
:CELLular
<sp><num value>
? (returns num value)
:PCS
<sp><num value>
? (returns num value)

CALL:SID

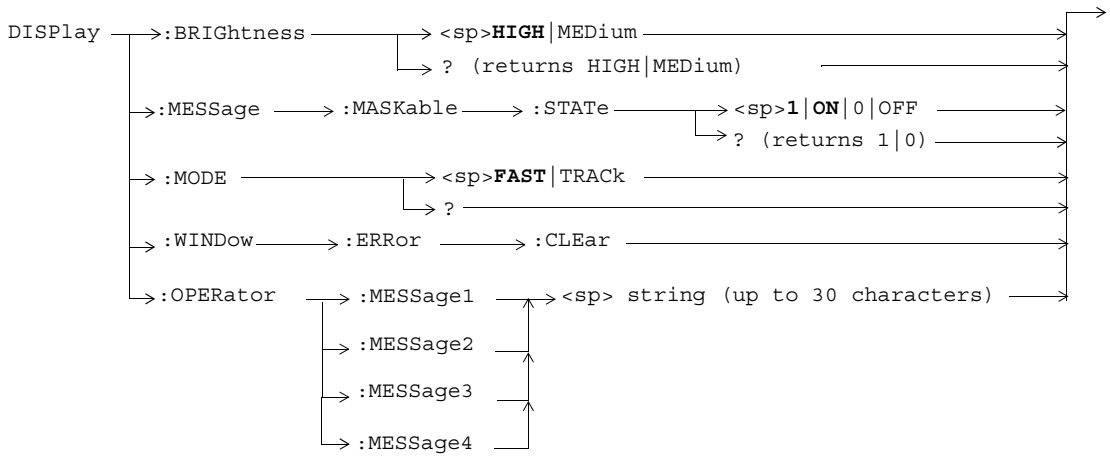
CALL:SOCode

CALL:STATus

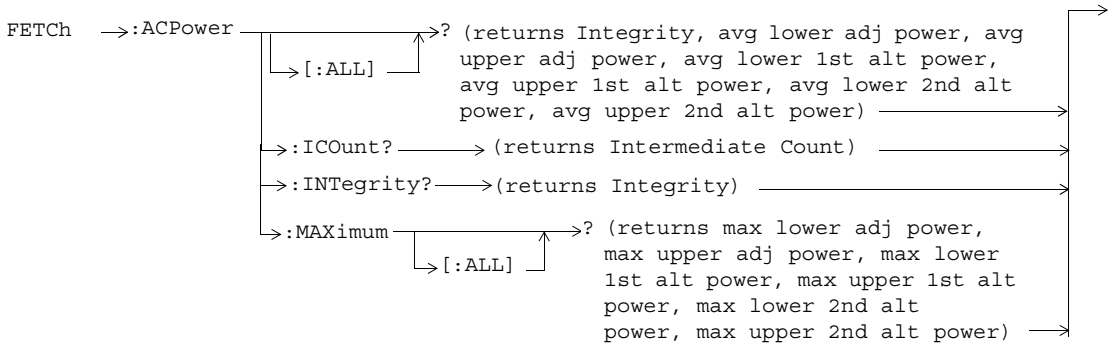

CALL
:STATus
[:STATe]
? (returns IDLE|SREQ|ALER|REL|HAND|
REG|PAG|CONN)
:AVC
[:CHANnel]
? (returns num value)
:SATone
[:CCODe]
? (returns SAT1|SAT2|
SAT3|UNKN)
:DTC
[:CHANnel]
? (returns num value)
:BAND? (returns CELL|PCS|UNKN
:VOCoder
[:TYPE]
? (returns VSEL|ACEL1|
ACELP1|UNKN)
:MS
:ANALog
:TXLevel? (returns num value)
:DIGital
:TXLevel? (returns num value)
:TCHannel
:TYPE? (returns DTC|AVC|UNKN)

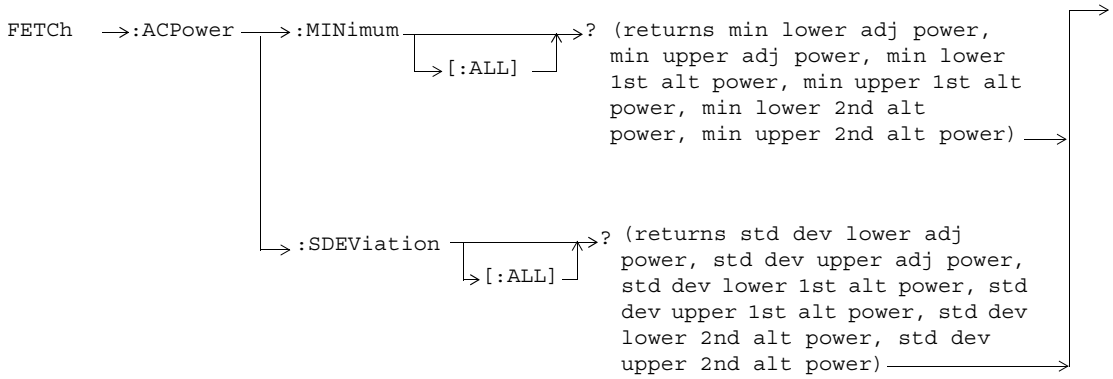
CALL:TCHannel

DISPlay

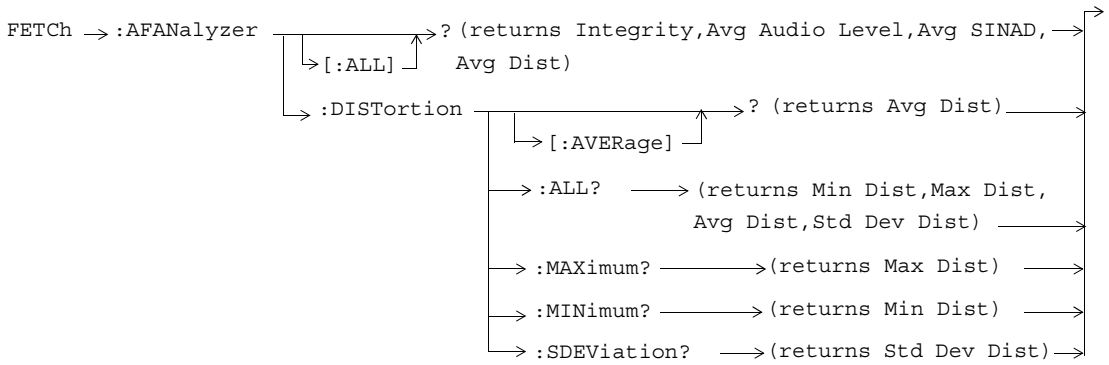

DISPlay
:BRIGhtness
<sp>HIGH|MEDium
? (returns HIGH|MEDium)
:MESSage
:MASKable
:STATe
<sp>1|ON|0|OFF
? (returns 1|0)
:MODE
<sp>FAST|TRACk
?
:WINDow
:ERRor
:CLEar
:OPERator
:MESSage1
:MESSage2
:MESSage3
:MESSage4
<sp> string (up to 30 characters)

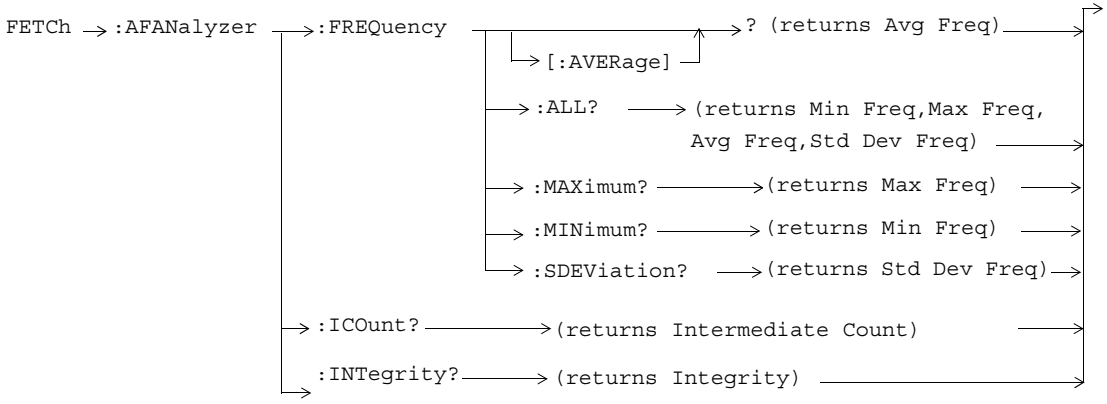
FETCh:ACPower

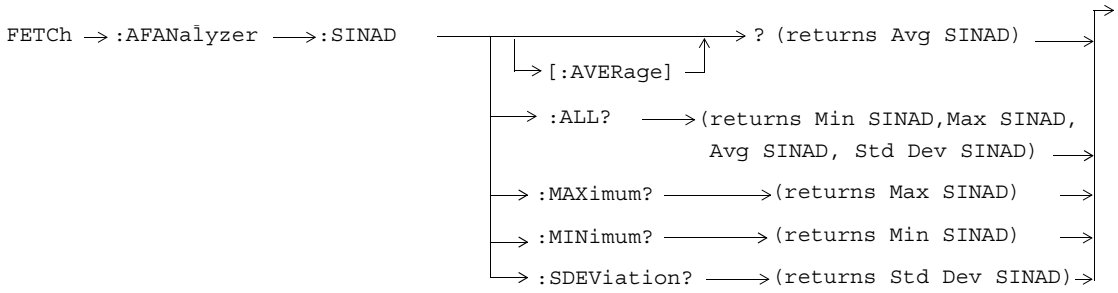


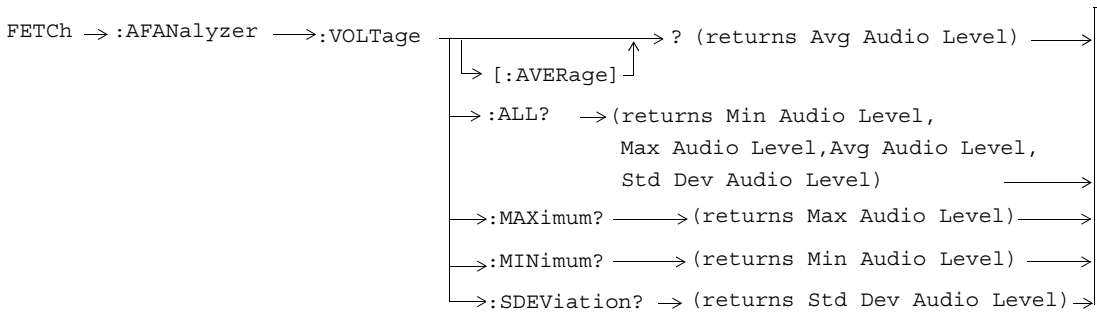

FETCh
:ACPower
:MINimum
[:ALL]
? (returns min lower adj power, min upper adj power, min lower 1st alt power, min upper 1st alt power, min lower 2nd alt power, min upper 2nd alt power)
:SDEViation
[:ALL]
? (returns std dev lower adj power, std dev upper adj power, std dev lower 1st alt power, std dev upper 1st alt power, std dev lower 2nd alt power, std dev upper 2nd alt power)

FETCh:AFANalyzer

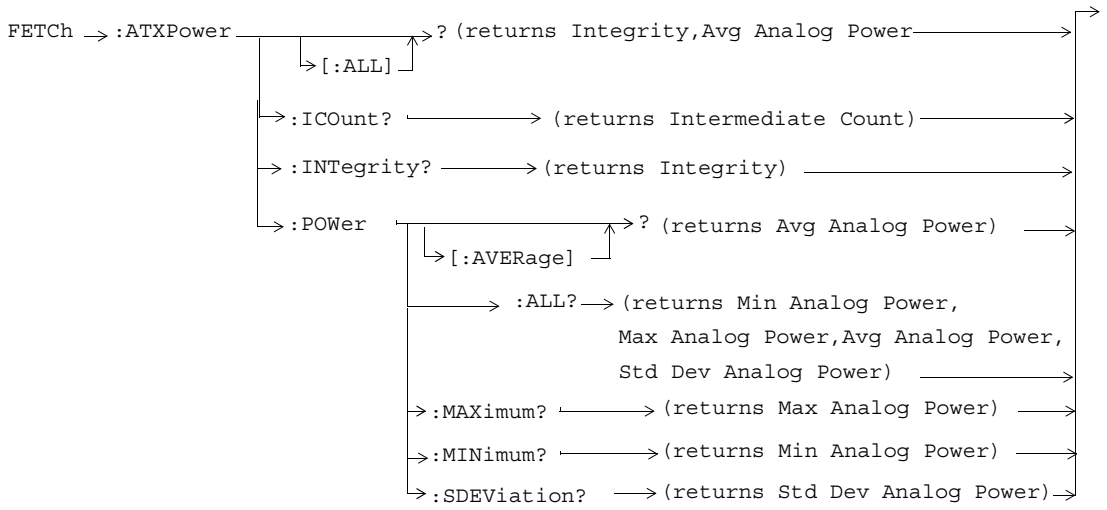



FETCh
:AFANalyzer
:FREQuency
[:AVERage]
? (returns Avg Freq)
:ALL?
(returns Min Freq,Max Freq,
Avg Freq,Std Dev Freq)
:MAXimum?
(returns Max Freq)
:MINimum?
(returns Min Freq)
:SDEViation?
(returns Std Dev Freq)
:ICOunt?
(returns Intermediate Count)
:INTegrity?
(returns Integrity)

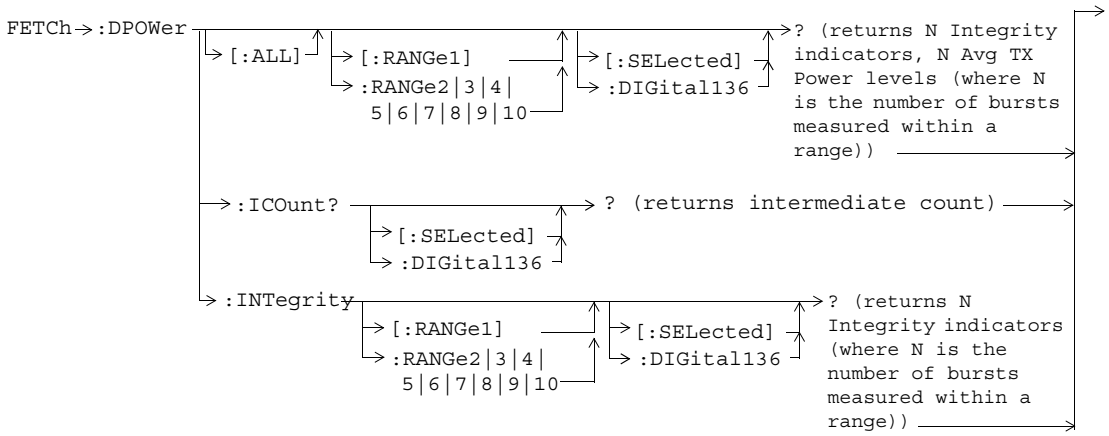

FETCh
:AFANalyzer
:SINAD
? (returns Avg SINAD)
[:AVERage]
:ALL?
(returns Min SINAD,Max SINAD,
Avg SINAD, Std Dev SINAD)
:MAXimum?
(returns Max SINAD)
:MINimum?
(returns Min SINAD)
:SDEViation?
(returns Std Dev SINAD)

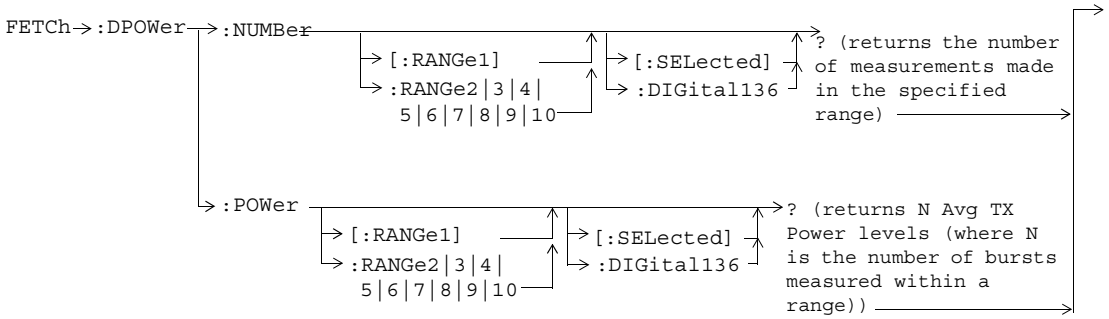

FETCh
:AFANalyzer
:VOLTage
? (returns Avg Audio Level)
[:AVERage]
:ALL?
(returns Min Audio Level,
Max Audio Level,Avg Audio Level,
Std Dev Audio Level)
:MAXimum?
(returns Max Audio Level)
:MINimum?
(returns Min Audio Level)
:SDEViation?
(returns Std Dev Audio Level)

FETCh:ATXPower

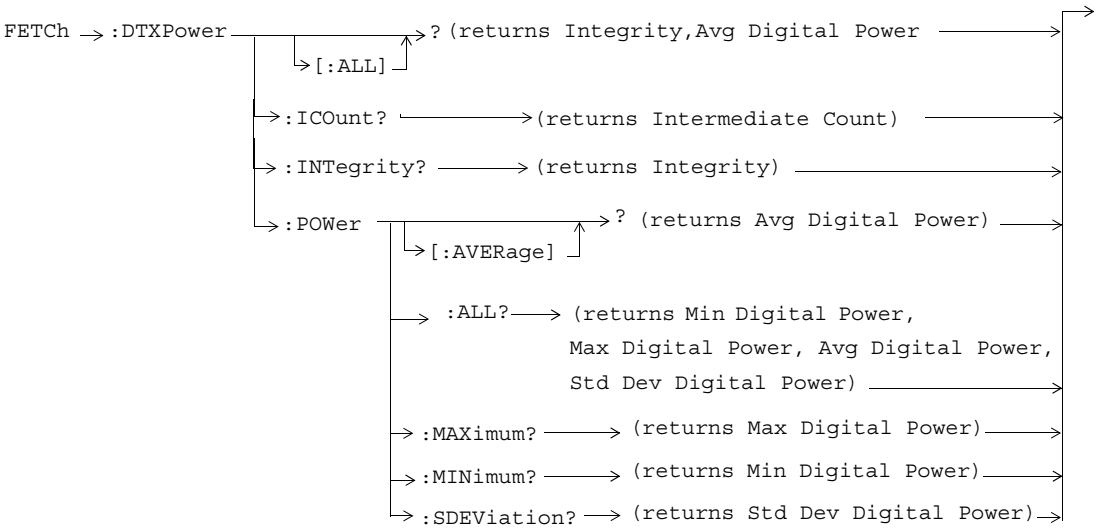


FETCh:DPOWer

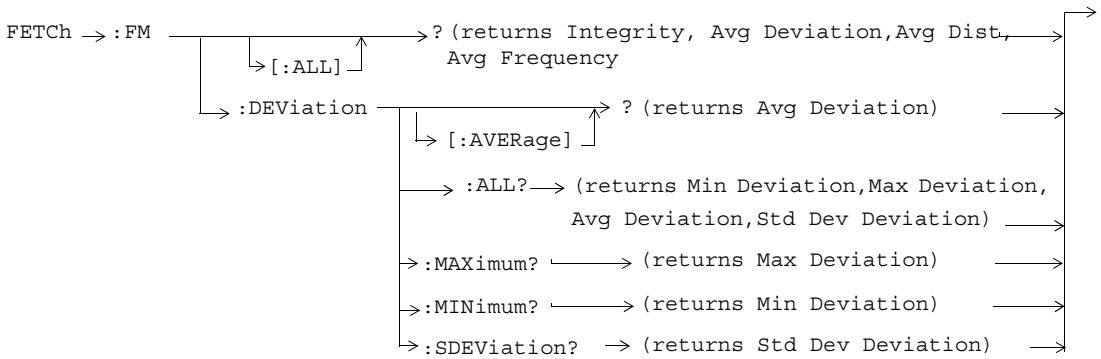


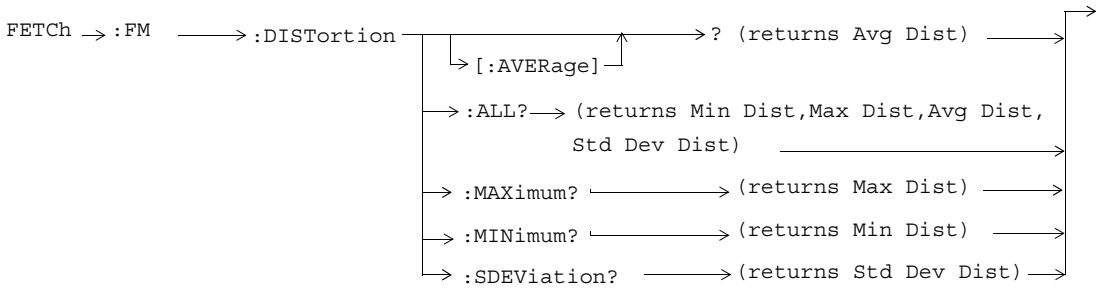

FETCh
:DPOWer
:NUMBer
[:RANGe1]
:RANGe2|3|4|
5|6|7|8|9|10
[:SELected]
:DIGital136
? (returns the number
of measurements made
in the specified
range)
:POWer
[:RANGe1]
:RANGe2|3|4|
5|6|7|8|9|10
[:SELected]
:DIGital136
? (returns N Avg TX
Power levels (where N
is the number of bursts
measured within a
range))

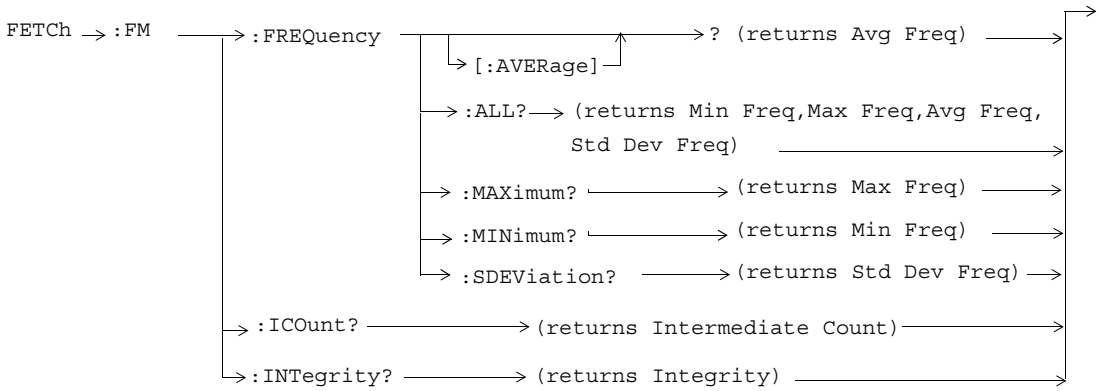
FETCh:DTXPower



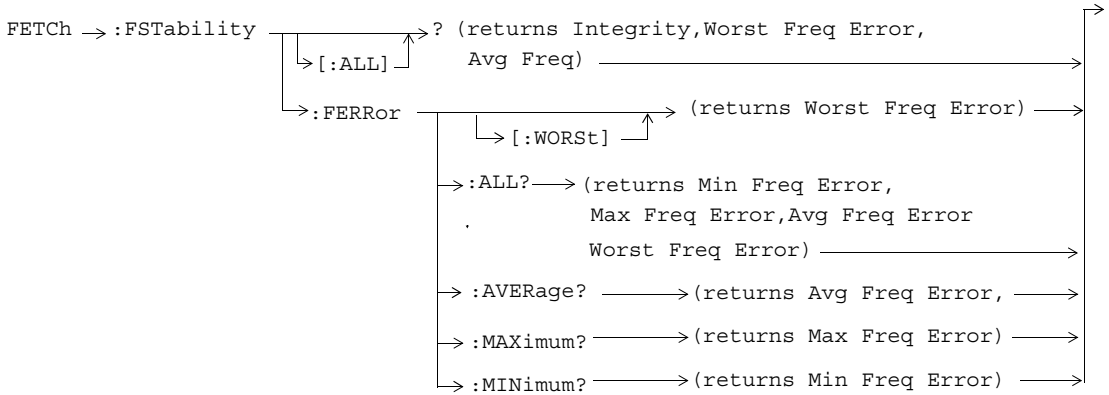
FETCh:FM

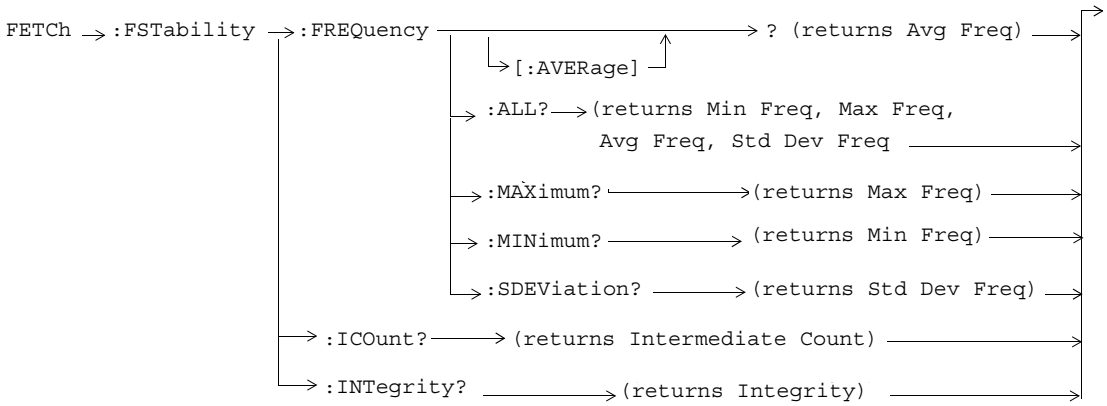

FETCh
:FM
[:ALL]
? (returns Integrity, Avg Deviation,Avg Dist,
Avg Frequency
:DEViation
[:AVERage]
? (returns Avg Deviation)
:ALL?
(returns Min Deviation,Max Deviation,
Avg Deviation,Std Dev Deviation)
:MAXimum?
(returns Max Deviation)
:MINimum?
(returns Min Deviation)
:SDEViation?
(returns Std Dev Deviation)


FETCh
:FM
:DISTortion
[:AVERage]
? (returns Avg Dist)
:ALL?
(returns Min Dist,Max Dist,Avg Dist,
Std Dev Dist)
:MAXimum?
(returns Max Dist)
:MINimum?
(returns Min Dist)
:SDEViation?
(returns Std Dev Dist)

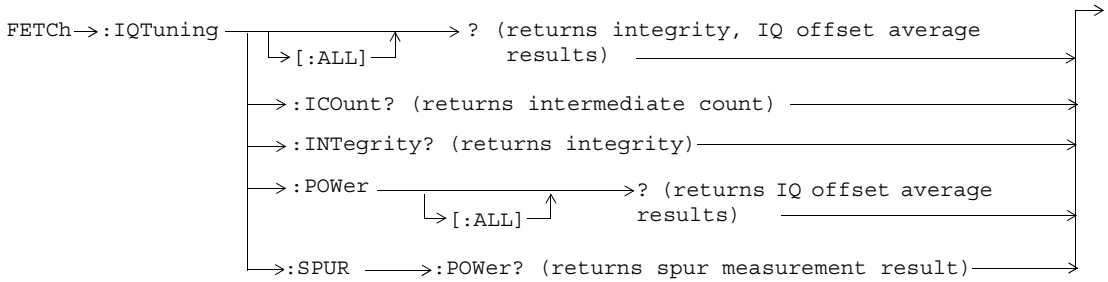

FETCh
:FM
:FREQuency
[:AVERage]
? (returns Avg Freq)
:ALL?
(returns Min Freq,Max Freq,Avg Freq,
Std Dev Freq)
:MAXimum?
(returns Max Freq)
:MINimum?
(returns Min Freq)
:SDEViation?
(returns Std Dev Freq)
:ICOunt?
(returns Intermediate Count)
:INTegrity?
(returns Integrity)

FETCh:FSTability

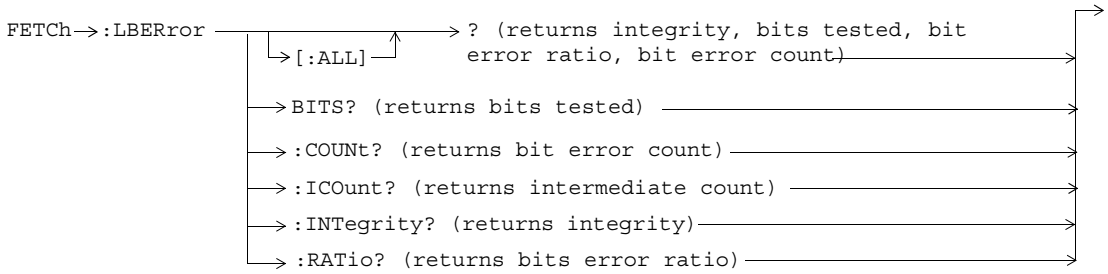



FETCh
:FSTability
:FREQuency
? (returns Avg Freq)
[:AVERage]
:ALL?
(returns Min Freq, Max Freq,
Avg Freq, Std Dev Freq
:MAXimum?
(returns Max Freq)
:MINimum?
(returns Min Freq)
:SDEViation?
(returns Std Dev Freq)
:ICOunt?
(returns Intermediate Count)
:INTegrity?
(returns Integrity)

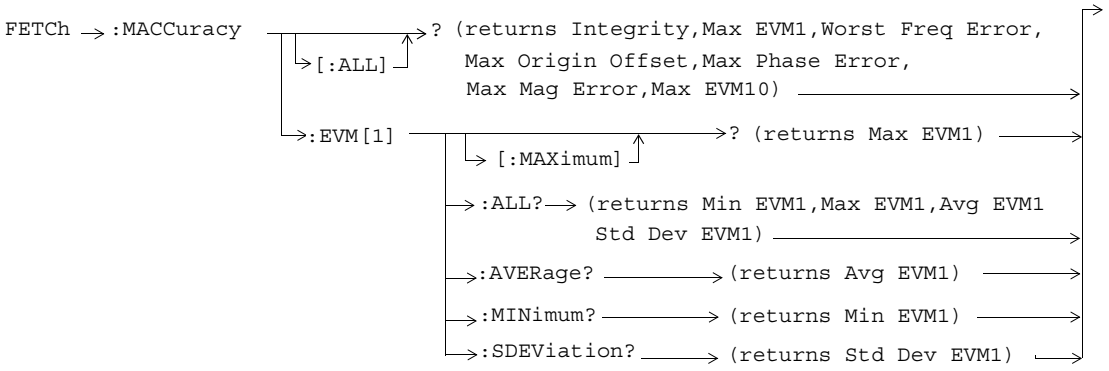
FETCh:IQTuning

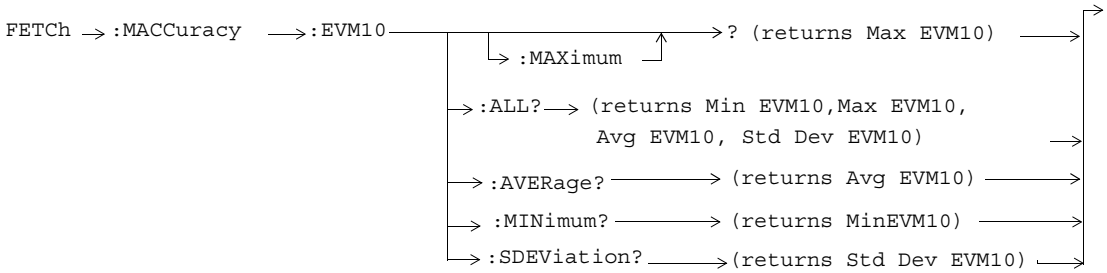


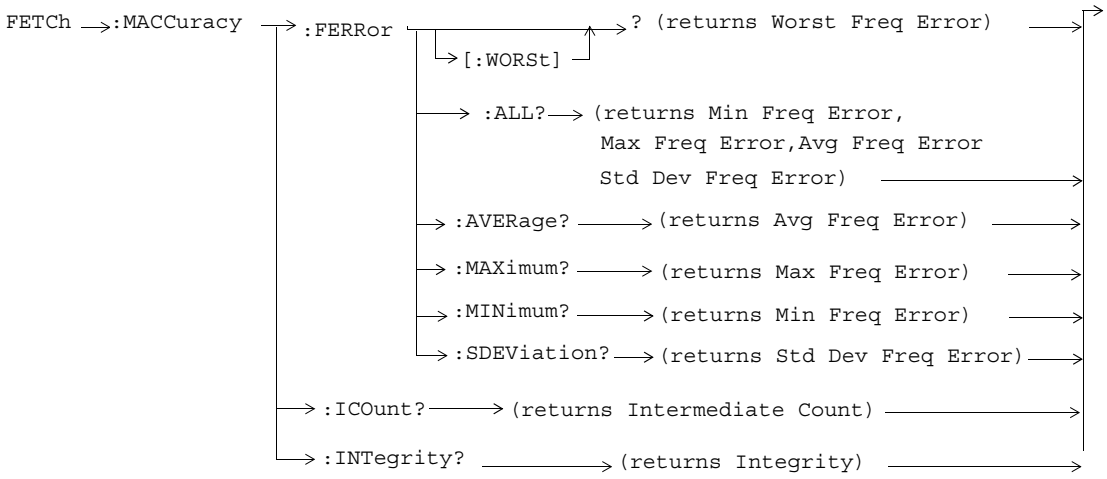
FETCh:LBERror

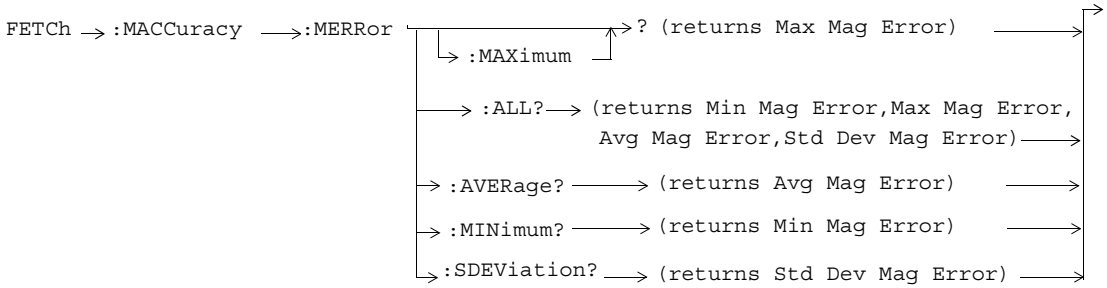


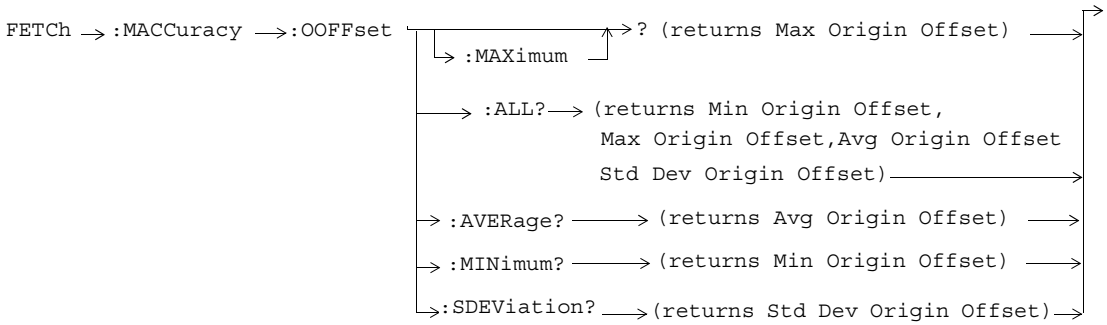
FETCh:MACCuracy

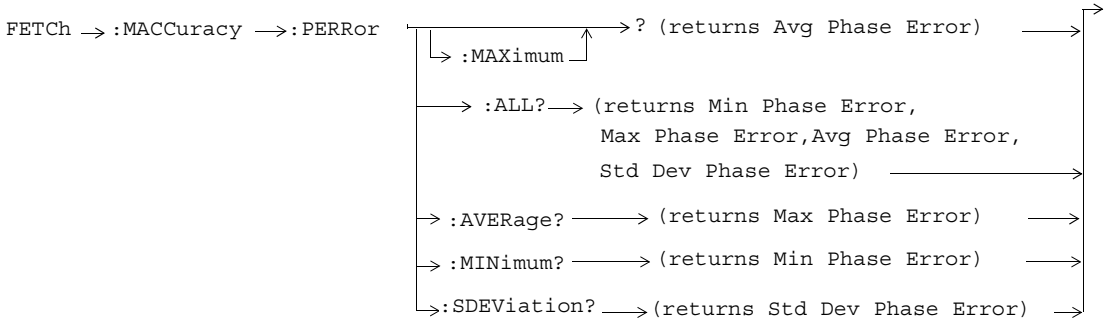



FETCh
:MACCuracy
:EVM10
? (returns Max EVM10)
:MAXimum
:ALL?
(returns Min EVM10,Max EVM10,
Avg EVM10, Std Dev EVM10)
:AVERage?
(returns Avg EVM10)
:MINimum?
(returns MinEVM10)
:SDEViation?
(returns Std Dev EVM10)

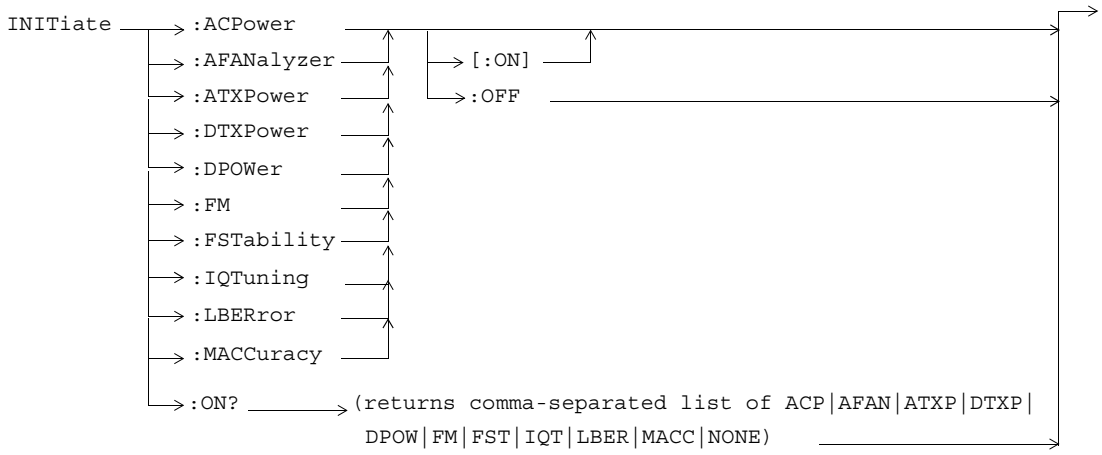

FETCh
:MACCuracy
:FERRor
? (returns Worst Freq Error)
[:WORSt]
:ALL?
(returns Min Freq Error,
Max Freq Error,Avg Freq Error
Std Dev Freq Error)
:AVERage?
(returns Avg Freq Error)
:MAXimum?
(returns Max Freq Error)
:MINimum?
(returns Min Freq Error)
:SDEViation?
(returns Std Dev Freq Error)
:ICOunt?
(returns Intermediate Count)
:INTegrity?
(returns Integrity)

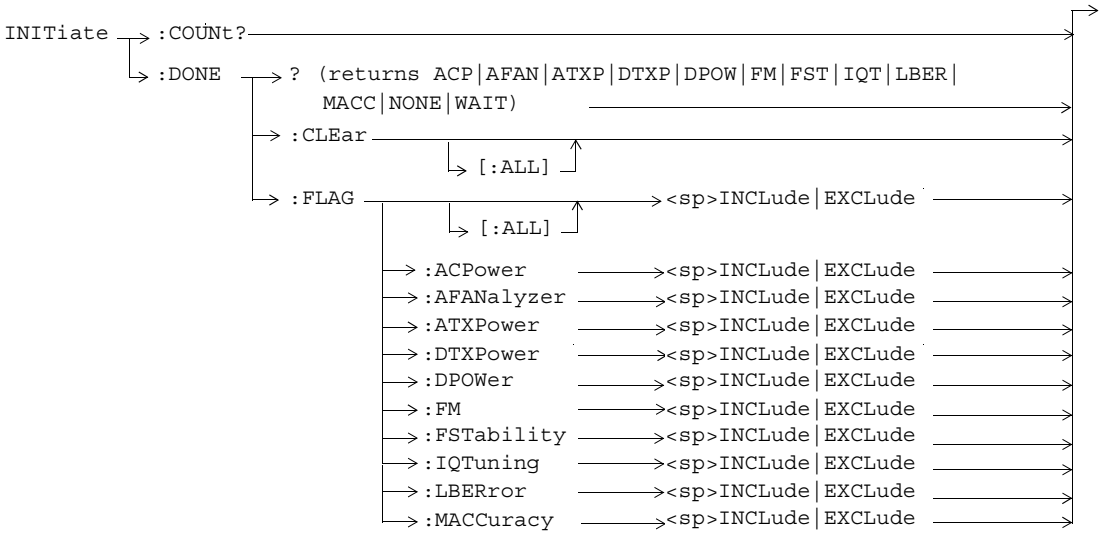

FETCh
:MACCuracy
:MERRor
? (returns Max Mag Error)
:MAXimum
:ALL?
(returns Min Mag Error,Max Mag Error,
Avg Mag Error,Std Dev Mag Error)
:AVERage?
(returns Avg Mag Error)
:MINimum?
(returns Min Mag Error)
:SDEViation?
(returns Std Dev Mag Error)


FETCh
:MACCuracy
:OOFFset
? (returns Max Origin Offset)
:MAXimum
:ALL?
(returns Min Origin Offset,
Max Origin Offset,Avg Origin Offset
Std Dev Origin Offset)
:AVERage?
(returns Avg Origin Offset)
:MINimum?
(returns Min Origin Offset)
:SDEViation?
(returns Std Dev Origin Offset)

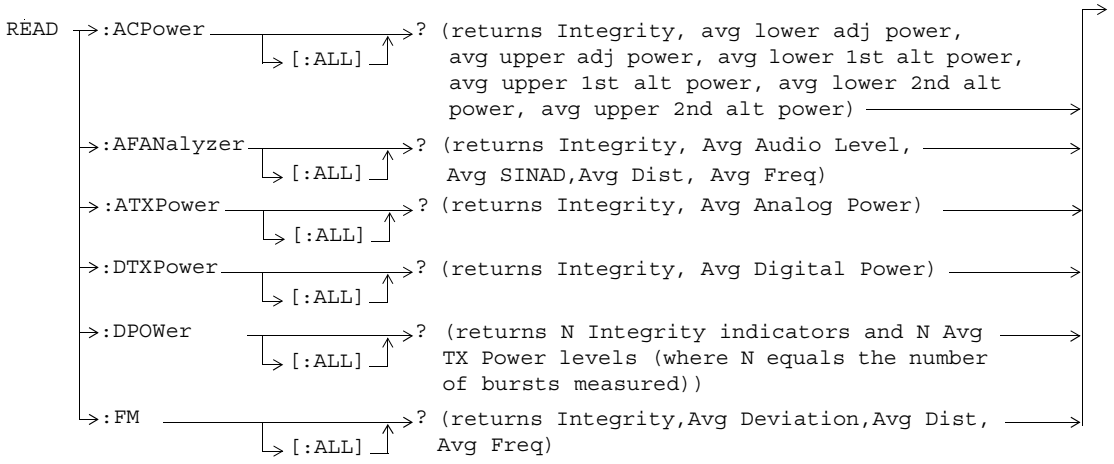

FETCh
:MACCuracy
:PERRor
? (returns Avg Phase Error)
:MAXimum
:ALL?
(returns Min Phase Error,
Max Phase Error,Avg Phase Error,
Std Dev Phase Error)
:AVERage?
(returns Max Phase Error)
:MINimum?
(returns Min Phase Error)
:SDEViation?
(returns Std Dev Phase Error)

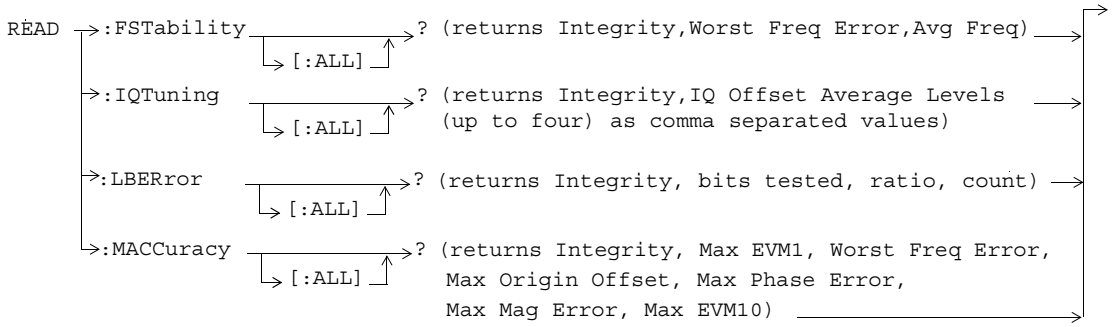
INITiate


INITiate
:ACPower
:AFANalyzer
:ATXPower
:DTXPower
:DPOWer
:FM
:FSTability
:IQTuning
:LBERror
:MACCuracy
:ON?
[:ON]
:OFF
(returns comma-separated list of ACP|AFAN|ATXP|DTXP|
DPOW|FM|FST|IQT|LBER|MACC|NONE)

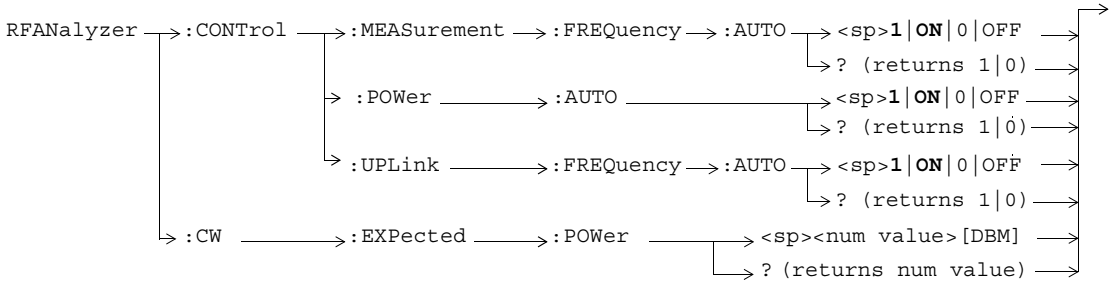

INITiate
:COUNt?
:DONE
? (returns ACP|AFAN|ATXP|DTXP|DPOW|FM|FST|IQT|LBER|
MACC|NONE|WAIT)
:CLEar
[:ALL]
:FLAG
[:ALL]
<sp>INCLude|EXCLude
:ACPower
:AFANalyzer
:ATXPower
:DTXPower
:DPOWer
:FM
:FSTability
:IQTuning
:LBERror
:MACCuracy

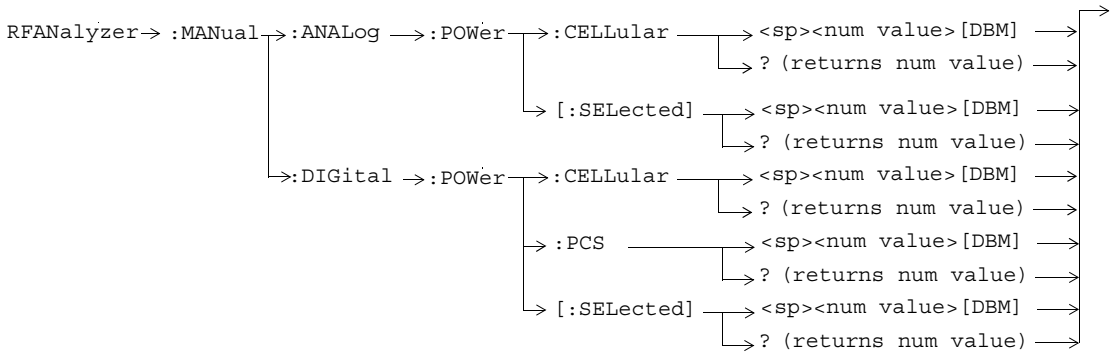
READ

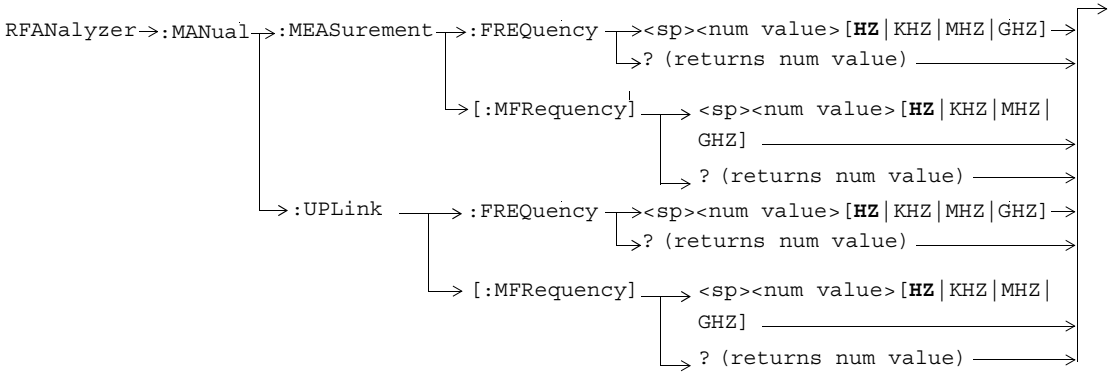

READ
:ACPower
[:ALL]
? (returns Integrity, avg lower adj power, avg upper adj power, avg lower 1st alt power, avg upper 1st alt power, avg lower 2nd alt power, avg upper 2nd alt power)
:AFANalyzer
[:ALL]
? (returns Integrity, Avg Audio Level, Avg SINAD,Avg Dist, Avg Freq)
:ATXPower
[:ALL]
? (returns Integrity, Avg Analog Power)
:DTXPower
[:ALL]
? (returns Integrity, Avg Digital Power)
:DPOWer
[:ALL]
? (returns N Integrity indicators and N Avg TX Power levels (where N equals the number of bursts measured))
:FM
[:ALL]
? (returns Integrity,Avg Deviation,Avg Dist, Avg Freq)


RËAD
:FSTability
[:ALL]
? (returns Integrity,Worst Freq Error,Avg Freq)
:IQTuning
[:ALL]
? (returns Integrity,IQ Offset Average Levels
(up to four) as comma separated values)
:LBERror
[:ALL]
? (returns Integrity, bits tested, ratio, count)
:MACCuracy
[:ALL]
? (returns Integrity, Max EVM1, Worst Freq Error,
Max Origin Offset, Max Phase Error,
Max Mag Error, Max EVM10)

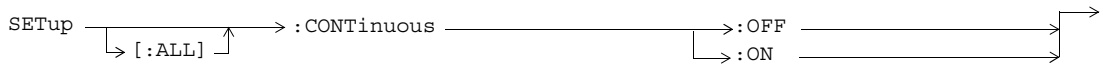
RFANalyzer


RFANalyzer
:CONTrol
:MEASurement
:FREQuency
:AUTO
<sp>1|ON|0|OFF
? (returns 1|0)
:POWer
:AUTO
<sp>1|ON|0|OFF
? (returns 1|0)
:UPLink
:FREQuency
:AUTO
<sp>1|ON|0|OFF
? (returns 1|0)
:CW
:EXPected
:POWer
<sp><num value>[DBM]
? (returns num value)

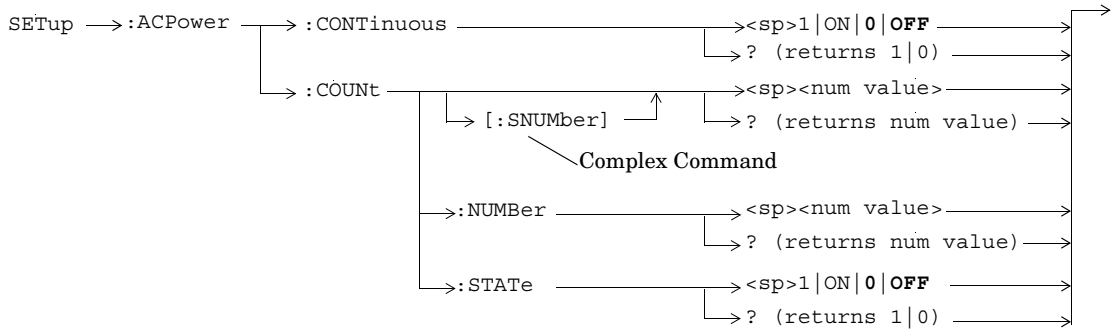

RFANalyzer
:MANual
:ANALog
:POWer
:CELLular
<sp><num value>[DBM]
? (returns num value)
[:SELected]
<sp><num value>[DBM]
? (returns num value)
:DIGital
:POWer
:CELLular
<sp><num value>[DBM]
? (returns num value)
:PCS
<sp><num value>[DBM]
? (returns num value)
[:SELected]
<sp><num value>[DBM]
? (returns num value)

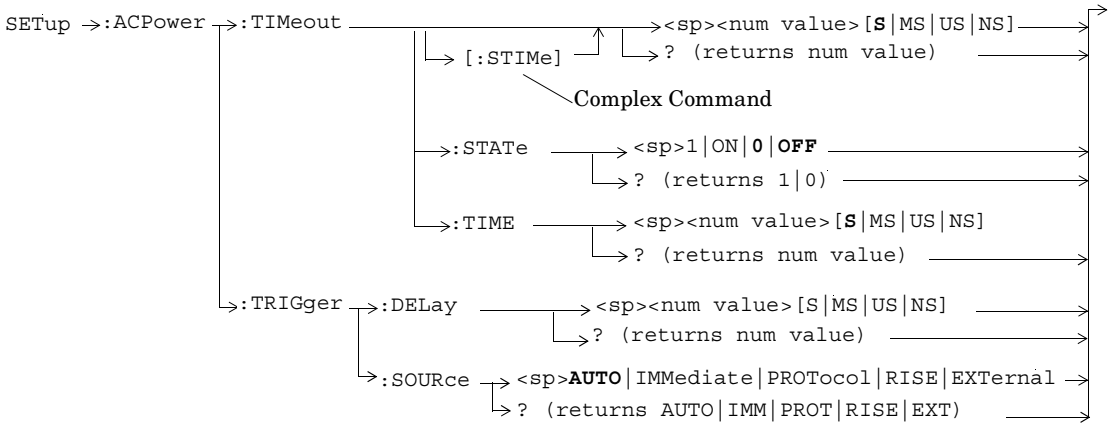

RFANalyzer
:MANual
:MEASurement
:FREQuency
<sp><num value>[HZ|KHZ|MHZ|GHZ]
? (returns num value)
[:MFRequency]
<sp><num value>[HZ|KHZ|MHZ|
GHZ]
? (returns num value)
:UPLink
:FREQuency
<sp><num value>[HZ|KHZ|MHZ|GHZ]
? (returns num value)
[:MFRequency]
<sp><num value>[HZ|KHZ|MHZ|
GHZ]
? (returns num value)

SETup[:ALL]:CONTinuous

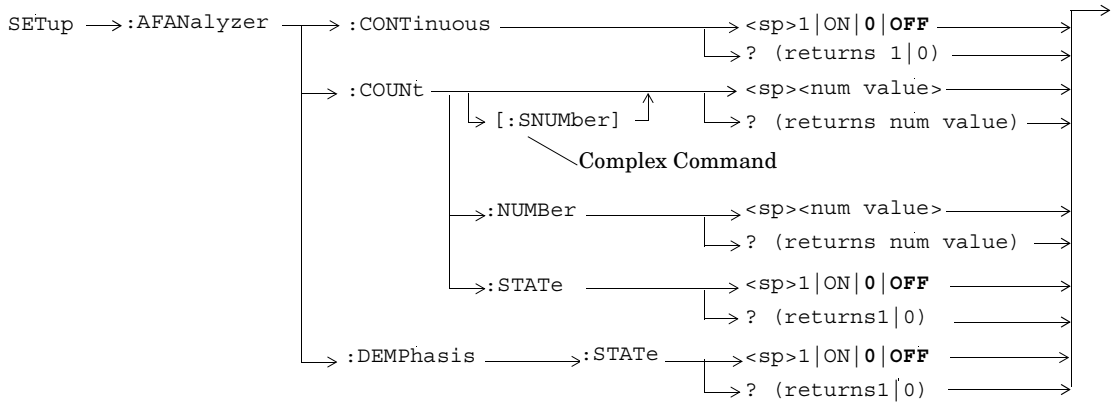


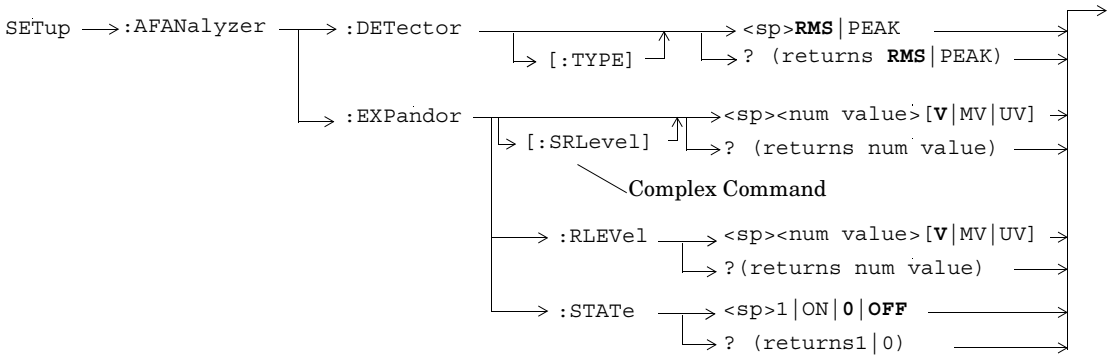
SETup:ACPower

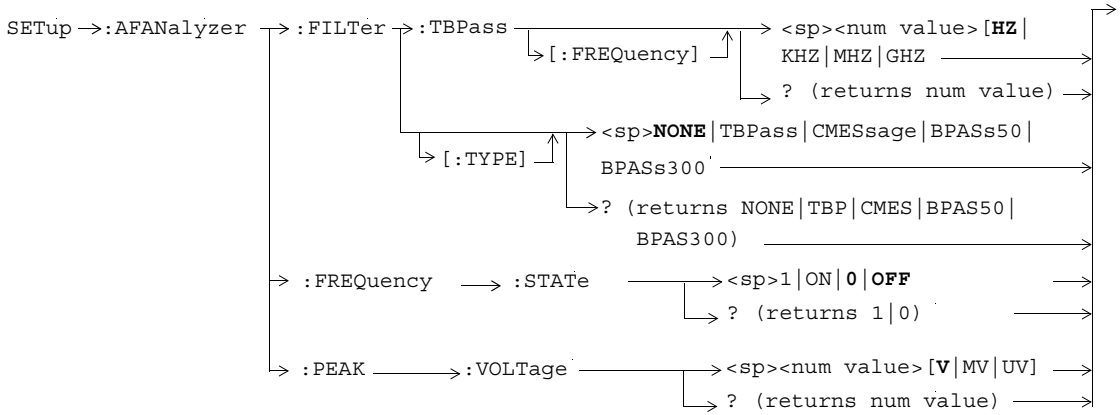


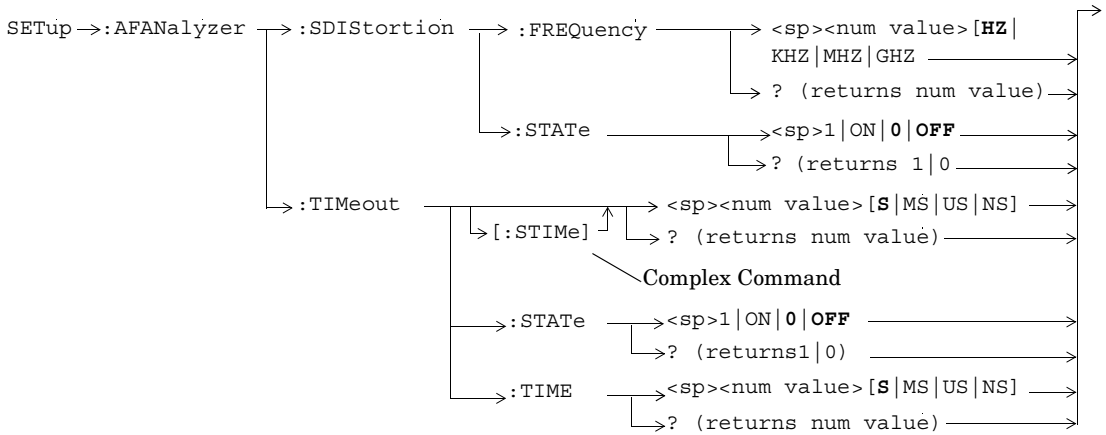

SETup
:ACPower
:TIMeout
<sp><num value>[S|MS|US|NS]
[:STIMe]
? (returns num value)
Complex Command
:STATe
<sp>1|ON|0|OFF
? (returns 1|0)
:TIME
<sp><num value>[S|MS|US|NS]
? (returns num value)
:TRIGger
:DELay
<sp><num value>[S|MS|US|NS]
? (returns num value)
:SOURce
<sp>AUTO|IMMediate|PROTocol|RISE|EXTernal
? (returns AUTO|IMM|PROT|RISE|EXT)

SETup:AFANalyzer

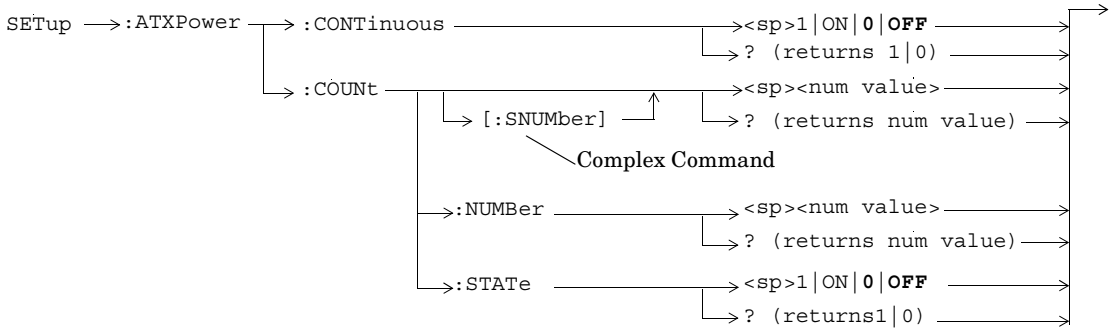


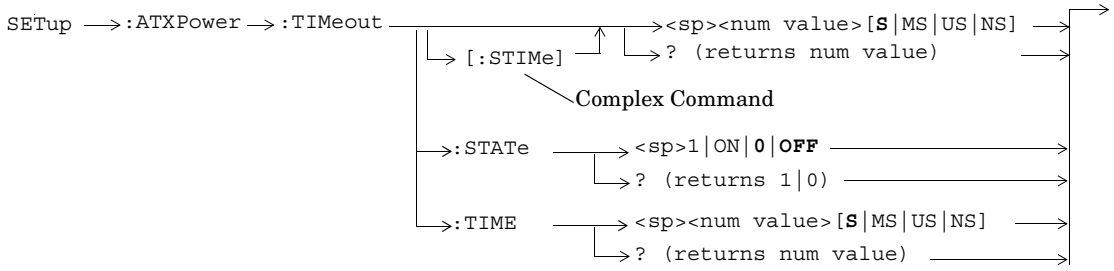

SETup
:AFANalyzer
:DETector
[:TYPE]
<sp>RMS|PEAK
? (returns RMS|PEAK)
:EXPandor
[:SRLevel]
<sp><num value>[V|MV|UV]
? (returns num value)
Complex Command
:RLEVel
<sp><num value>[V|MV|UV]
?(returns num value)
:STATe
<sp>1|ON|0|OFF
? (returns1|0)


SETup
:AFANalyzer
:FILTer
:TBPass
[:FREQuency]
<sp><num value>[HZ|
KHZ|MHZ|GHZ
? (returns num value)
[:TYPE]
<sp>NONE|TBPass|CMESsage|BPASs50|
BPASs300
? (returns NONE|TBP|CMES|BPAS50|
BPAS300)
:FREQuency
:STATe
<sp>1|ON|0|OFF
? (returns 1|0)
:PEAK
:VOLTage
<sp><num value>[V|MV|UV]
? (returns num value)

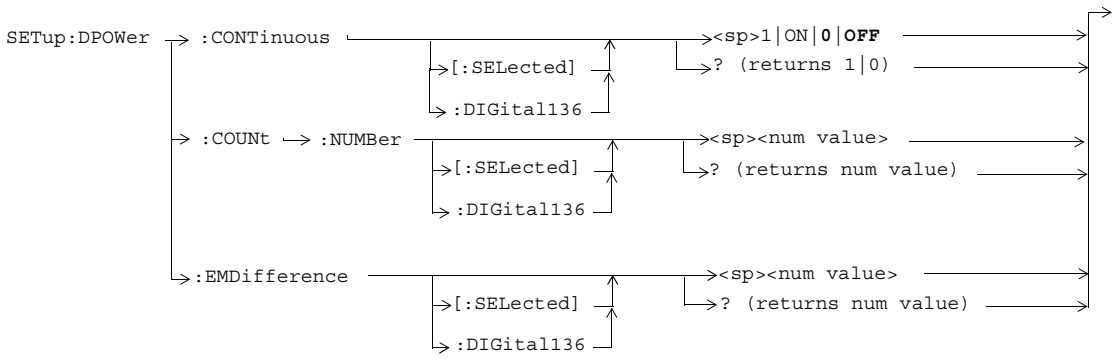

SETup
:AFANalyzer
:SDIStortion
:FREQuency
<sp><num value>[HZ|
KHZ|MHZ|GHZ
? (returns num value)
:STATe
<sp>1|ON|0|OFF
? (returns 1|0
:TIMeout
[:STIMe]
<sp><num value>[S|MS|US|NS]
? (returns num value)
Complex Command
:STATe
<sp>1|ON|0|OFF
? (returns1|0)
:TIME
<sp><num value>[S|MS|US|NS]
? (returns num value)

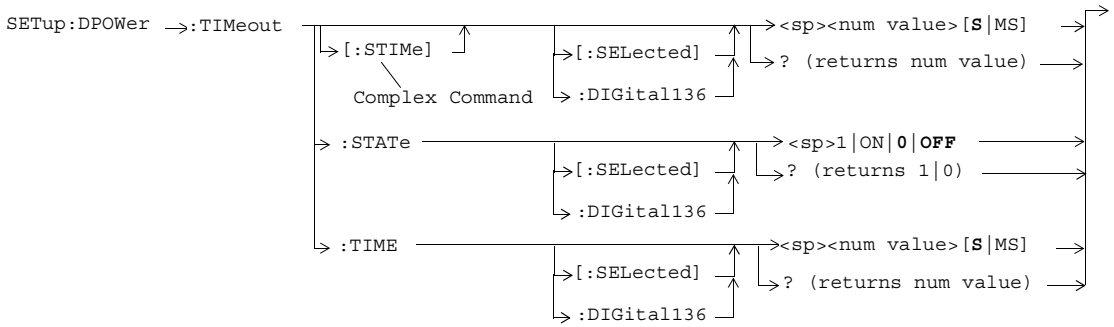
SETup:ATXPower



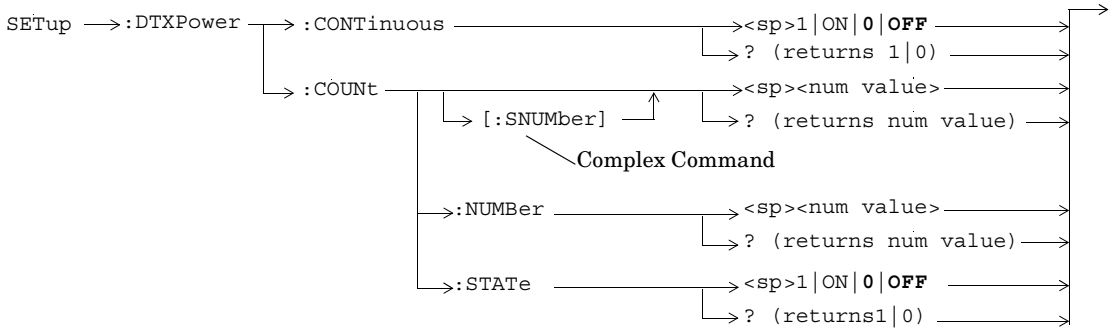

SETup
:ATXPower
:TIMeout
[:STIMe]
<sp><num value>[S|MS|US|NS]
? (returns num value)
Complex Command
:STATe
<sp>1|ON|0|OFF
? (returns 1|0)
:TIME
<sp><num value>[S|MS|US|NS]
? (returns num value)

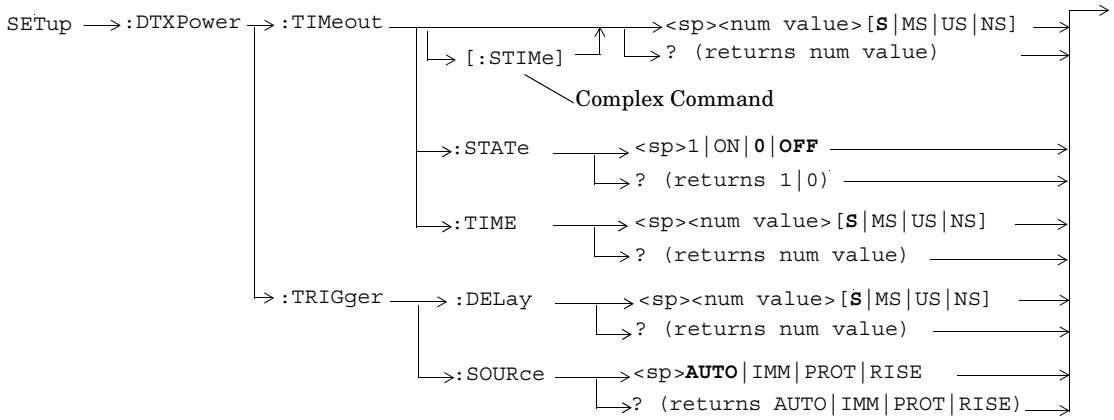
SETup:DPOWer



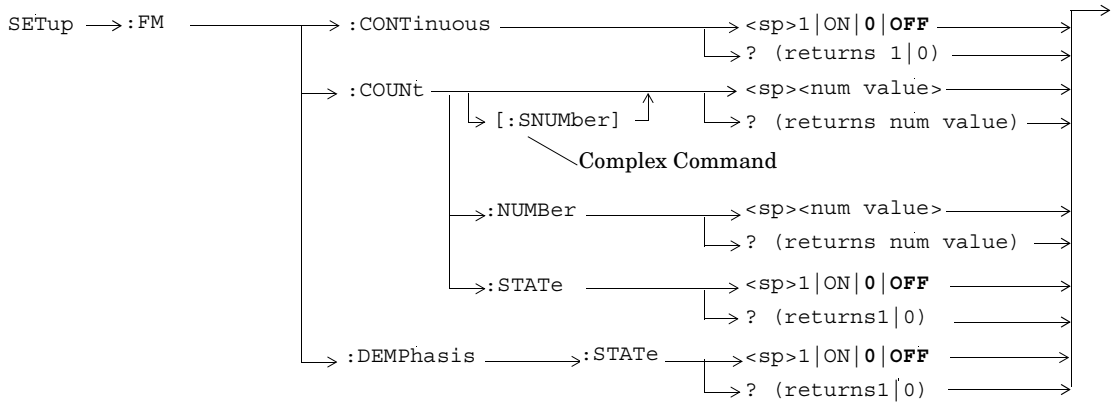

SETup:DPOWer
:TIMeout
[:STIMe]
Complex Command
[:SELected]
:DIGital136
<sp><num value>[S|MS]
? (returns num value)
:STATe
[:SELected]
:DIGital136
<sp>1|ON|0|OFF
? (returns 1|0)
:TIME
[:SELected]
:DIGital136
<sp><num value>[S|MS]
? (returns num value)

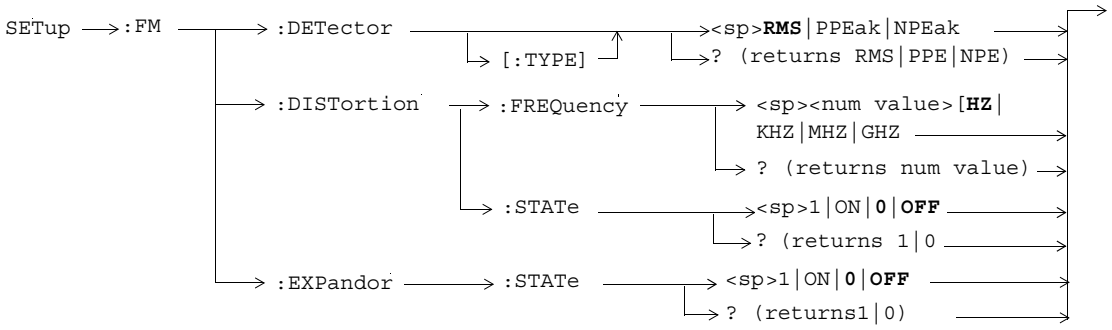
SETup:DTXPower

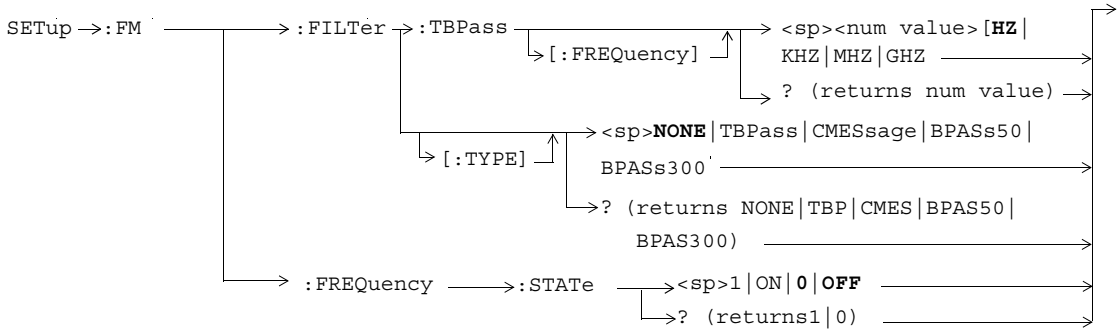


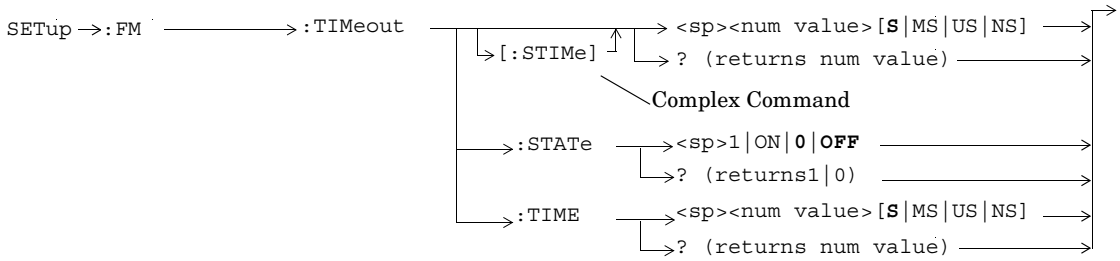

SETup
:DTXPower
:TIMeout
[:STIMe]
Complex Command
<sp><num value>[**S**|MS|US|NS]
? (returns num value)
:STATe
<sp>1|ON|**0**|**OFF**
? (returns 1|0)
:TIME
<sp><num value>[**S**|MS|US|NS]
? (returns num value)
:TRIGger
:DELay
<sp><num value>[**S**|MS|US|NS]
? (returns num value)
:SOURce
<sp>**AUTO**|IMM|PROT|RISE
? (returns AUTO|IMM|PROT|RISE)

SETup:FM

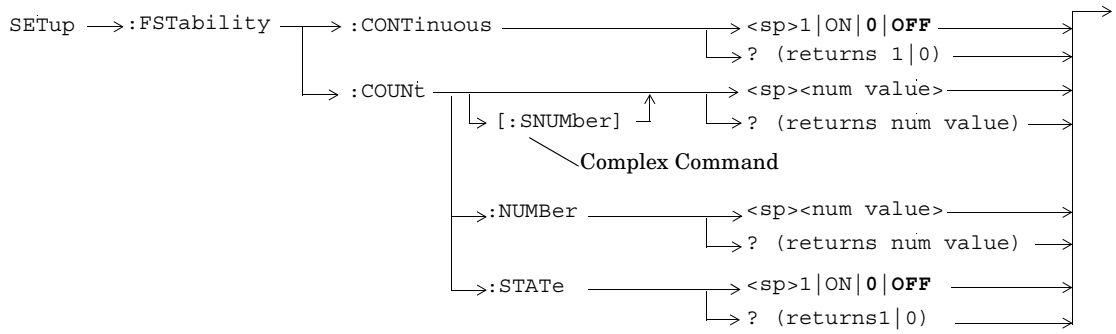

SETup
:FM
:CONTinuous
<sp>1|ON|0|OFF
? (returns 1|0)
:COUNt
<sp><num value>
[:SNUMber]
? (returns num value)
Complex Command
:NUMBer
<sp><num value>
? (returns num value)
:STATe
<sp>1|ON|0|OFF
? (returns1|0)
:DEMPhasis
:STATe
<sp>1|ON|0|OFF
? (returns1|0)

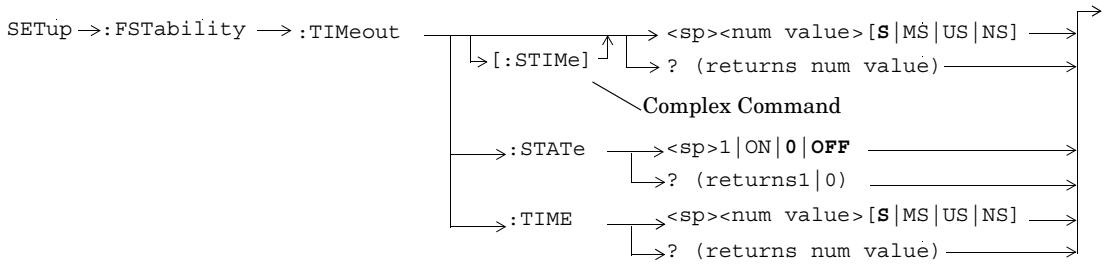

SETup
:FM
:DETector
[:TYPE]
<sp>RMS|PPEak|NPEak
? (returns RMS|PPE|NPE)
:DISTortion
:FREQuency
<sp><num value>[HZ|
KHZ|MHZ|GHZ
? (returns num value)
:STATe
<sp>1|ON|0|OFF
? (returns 1|0
:EXPandor
:STATe
<sp>1|ON|0|OFF
? (returns1|0)


SETup
:FM
:FILTer
:TBPass
[:FREQuency]
<sp><num value>[HZ|
KHZ|MHZ|GHZ
? (returns num value)
[:TYPE]
<sp>NONE|TBPass|CMESsage|BPASs50|
BPASs300
? (returns NONE|TBP|CMES|BPAS50|
BPAS300)
:FREQuency
:STATe
<sp>1|ON|0|OFF
? (returns1|0)

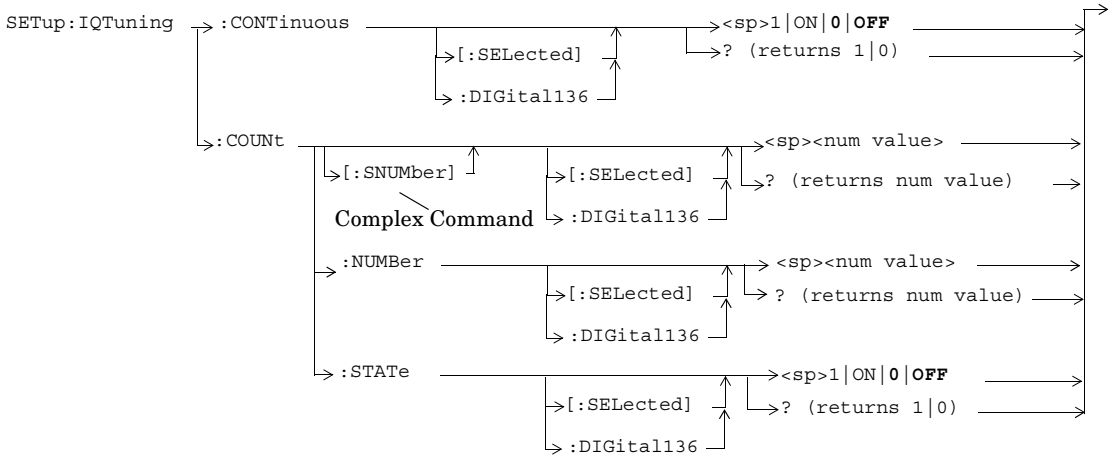

SETup
:FM
:TIMeout
[:STIMe]
<sp><num value>[S|MS|US|NS]
? (returns num value)
Complex Command
:STATe
<sp>1|ON|0|OFF
? (returns1|0)
:TIME
<sp><num value>[S|MS|US|NS]
? (returns num value)

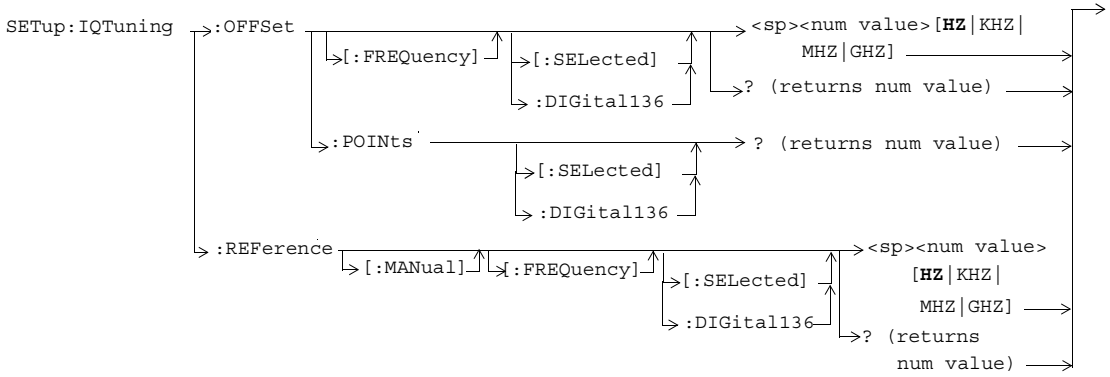
SETup:FSTability

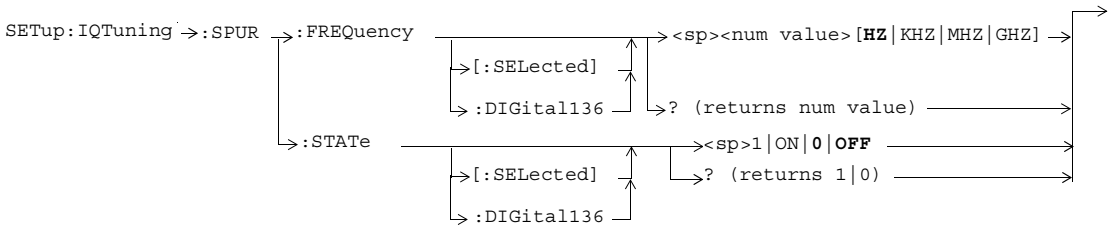


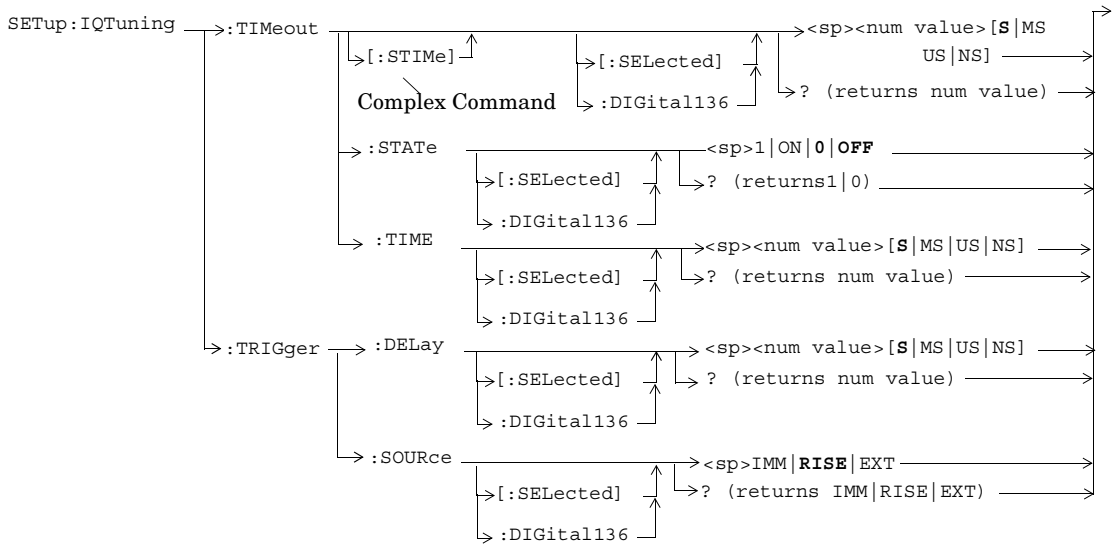

SETup
:FSTability
:TIMeout
[:STIMe]
<sp><num value>[S|MS|US|NS]
? (returns num value)
Complex Command
:STATe
<sp>1|ON|0|OFF
? (returns1|0)
:TIME
<sp><num value>[S|MS|US|NS]
? (returns num value)

SETup:IQTuning

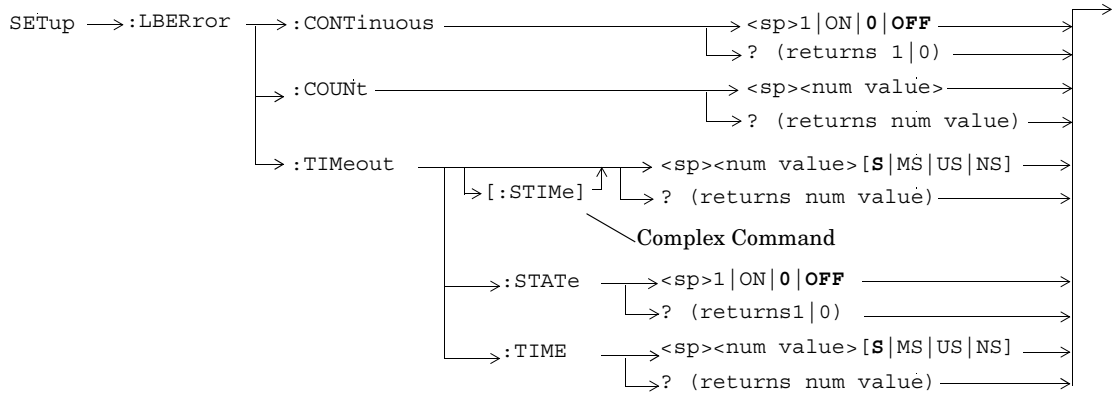



SETup:IQTuning
:OFFSet
[:FREQuency]
[:SELected]
:DIGital136
<sp><num value>[HZ|KHZ|
MHZ|GHZ]
? (returns num value)
:POINts
? (returns num value)
[:SELected]
:DIGital136
:REFerence
[:MANual]
[:FREQuency]
[:SELected]
:DIGital136
<sp><num value>
[HZ|KHZ|
MHZ|GHZ]
? (returns
num value)

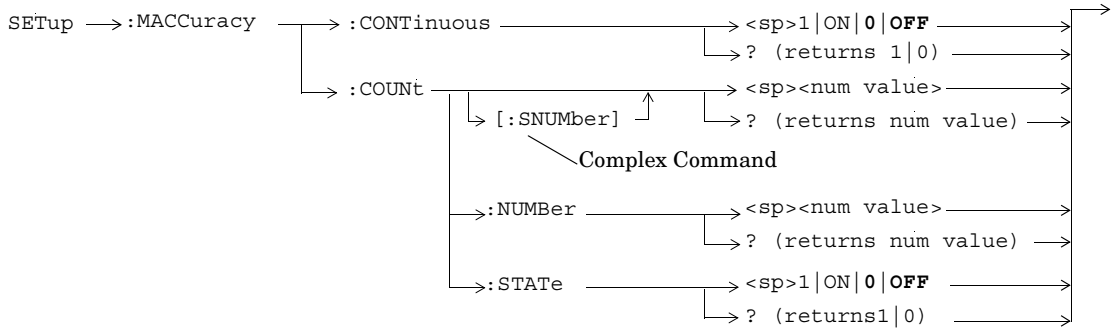

SETup:IQTuning
:SPUR
:FREQuency
[:SELected]
:DIGital136
<sp><num value>[**HZ**|KHZ|MHZ|GHZ]
? (returns num value)
:STATe
[:SELected]
:DIGital136
<sp>1|ON|**0**|**OFF**
? (returns 1|0)

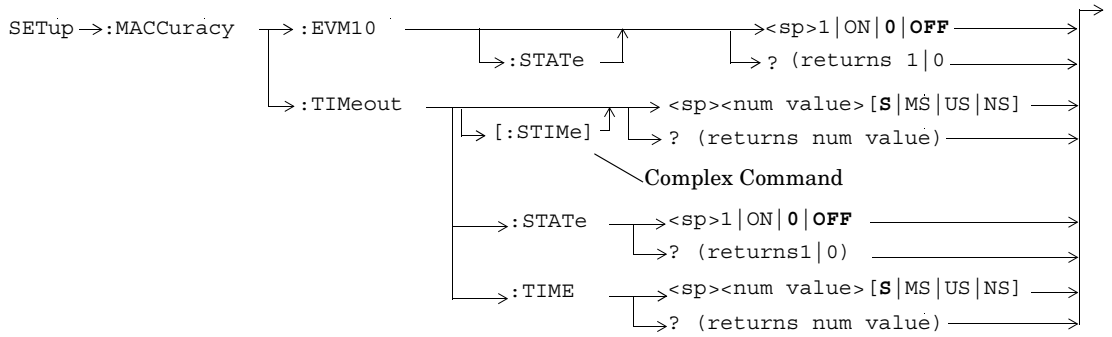

SETup:IQTuning
:TIMeout
[:STIMe]
Complex Command
[:SELected]
:DIGital136
<sp><num value>[S|MS
US|NS]
? (returns num value)
:STATe
[:SELected]
:DIGital136
<sp>1|ON|0|OFF
? (returns1|0)
:TIME
[:SELected]
:DIGital136
<sp><num value>[S|MS|US|NS]
? (returns num value)
:TRIGger
:DELay
[:SELected]
:DIGital136
<sp><num value>[S|MS|US|NS]
? (returns num value)
:SOURce
[:SELected]
:DIGital136
<sp>IMM|RISE|EXT
? (returns IMM|RISE|EXT)

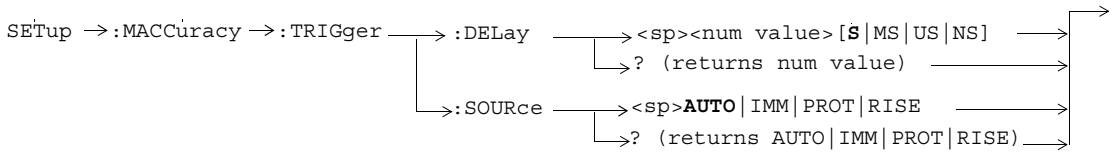
SETup:LBERror



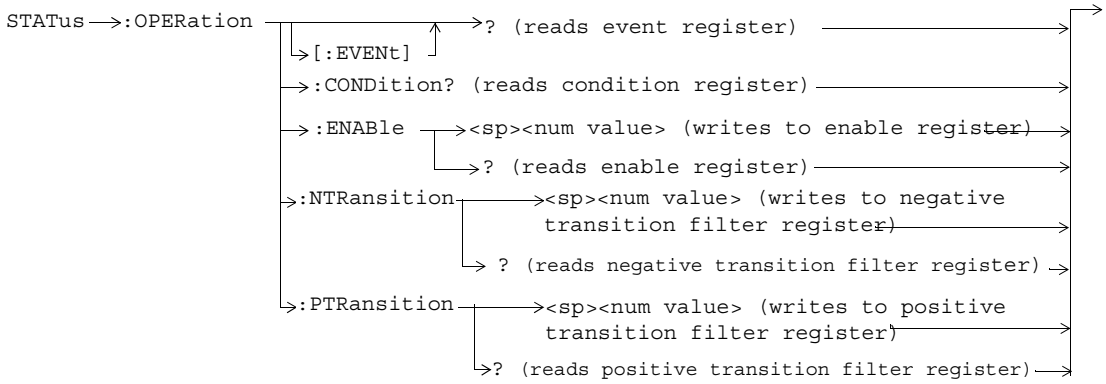
SETup:MACCuracy

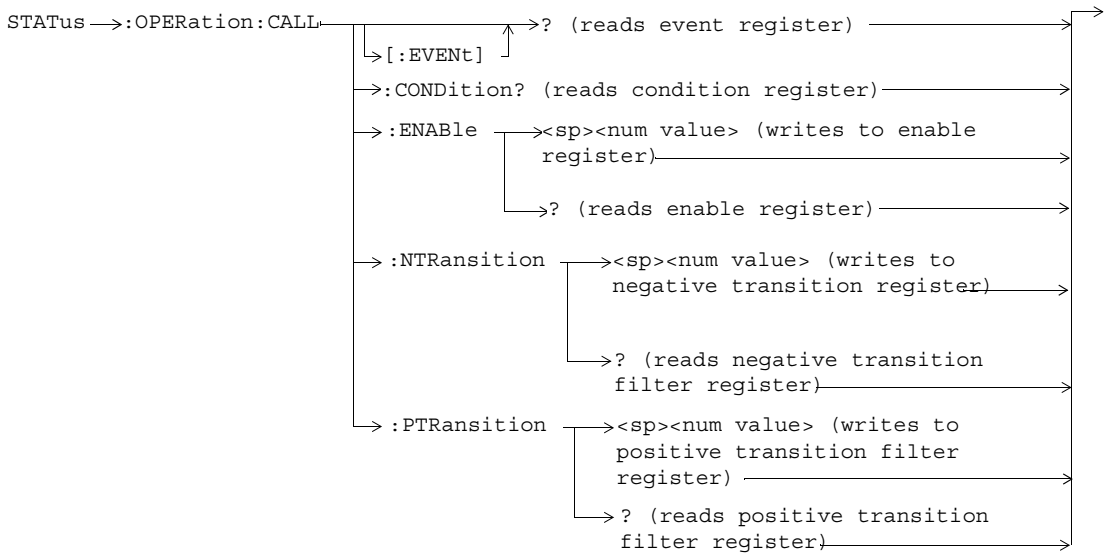


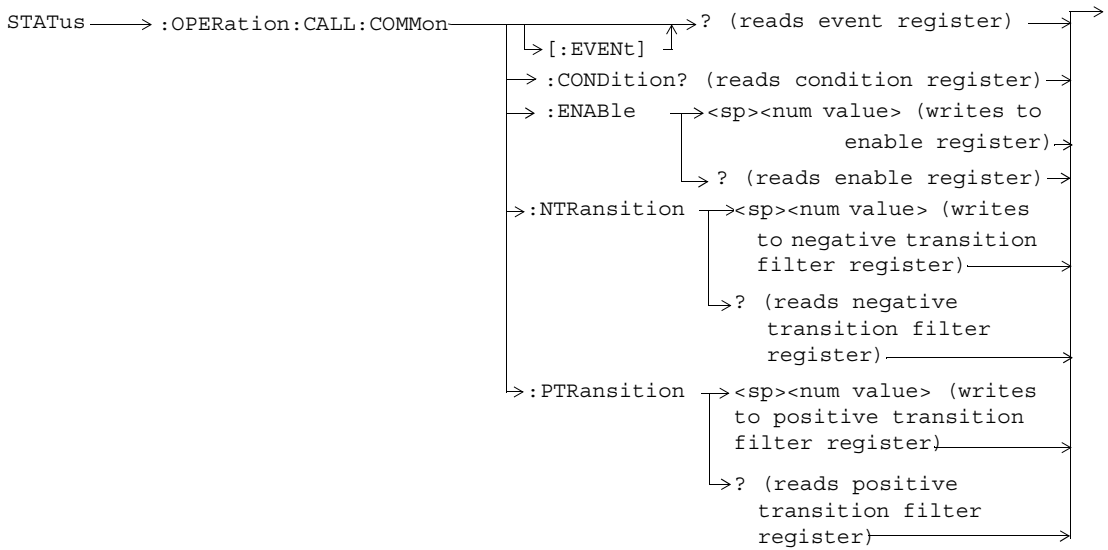

SETup
:MACCuracy
:EVM10
:STATe
<sp>1|ON|0|OFF
? (returns 1|0
:TIMeout
[:STIMe]
<sp><num value>[S|MS|US|NS]
? (returns num value)
Complex Command
:STATe
<sp>1|ON|0|OFF
? (returns1|0)
:TIME
<sp><num value>[S|MS|US|NS]
? (returns num value)

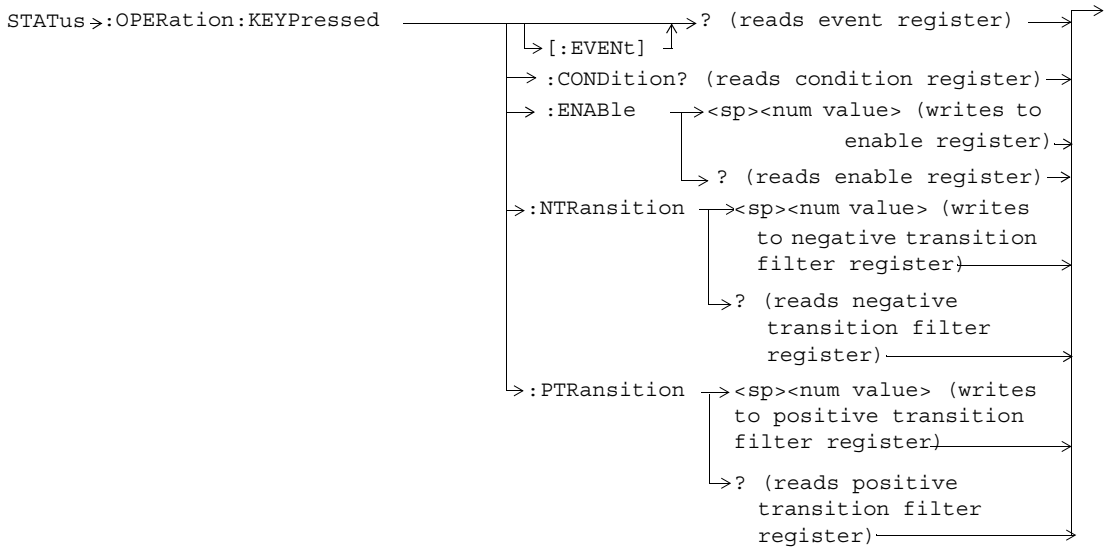

SETup
:MACCuracy
:TRIGger
:DELay
<sp><num value>[S|MS|US|NS]
? (returns num value)
:SOURce
<sp>AUTO|IMM|PROT|RISE
? (returns AUTO|IMM|PROT|RISE)

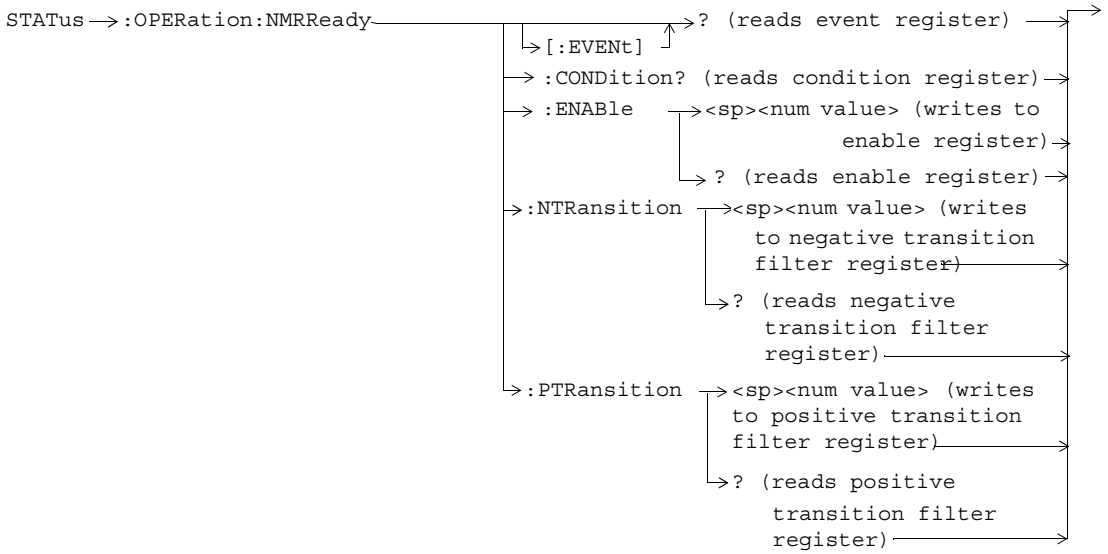
STATus:OPERation

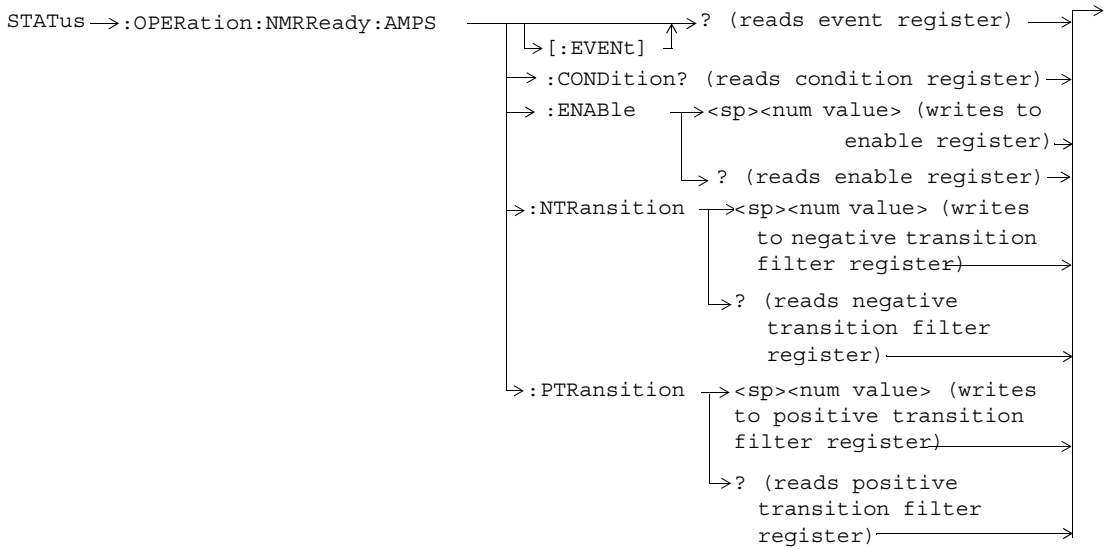


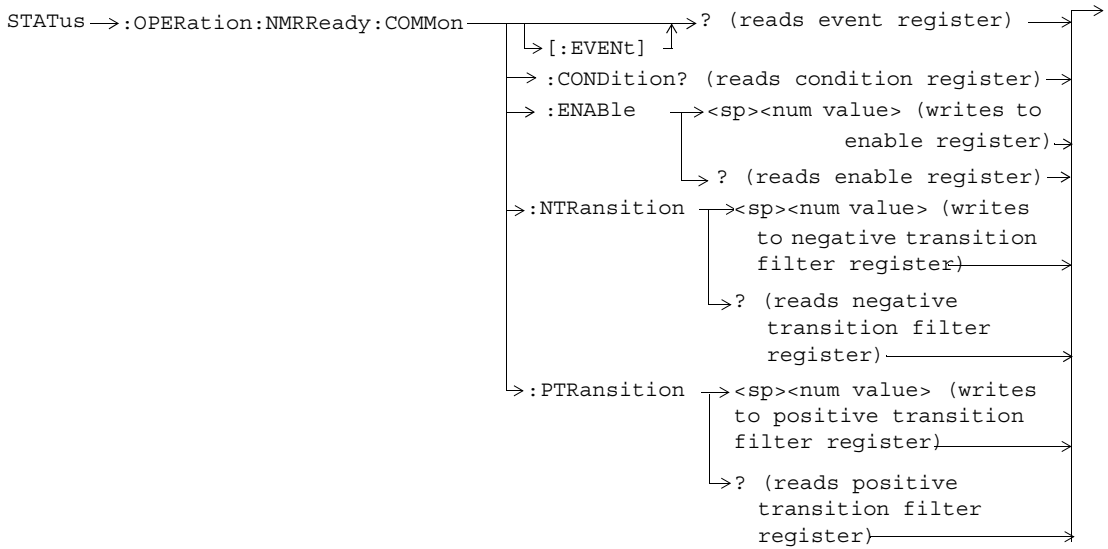

STATus
:OPERation:CALL
? (reads event register)
[:EVENt]
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to negative transition register)
? (reads negative transition filter register)
:PTRansition
<sp><num value> (writes to positive transition filter register)
? (reads positive transition filter register)

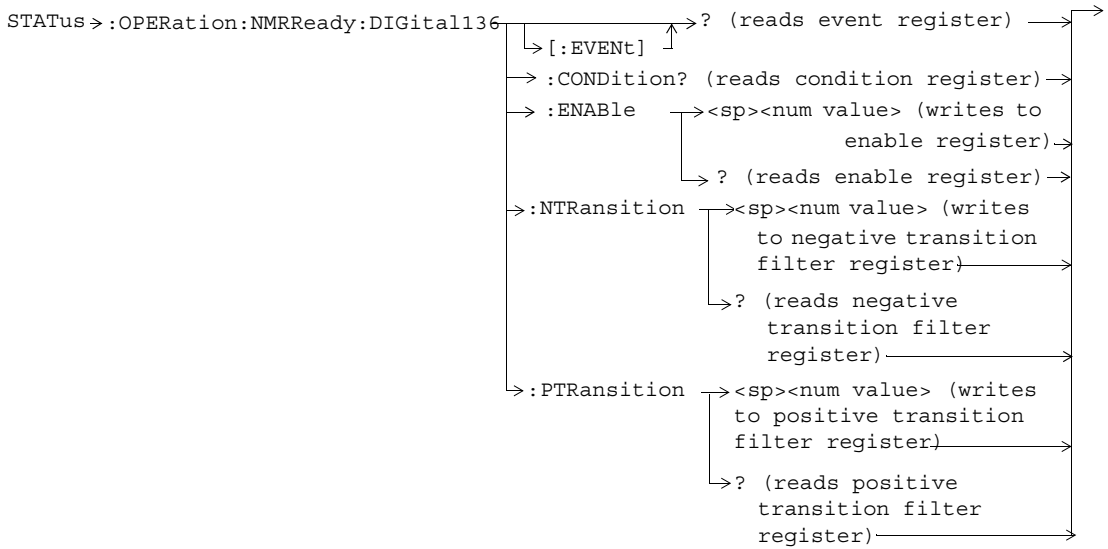

STATus
:OPERation:CALL:COMMon
[:EVENt]
? (reads event register)
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to negative transition filter register)
? (reads negative transition filter register)
:PTRansition
<sp><num value> (writes to positive transition filter register)
? (reads positive transition filter register)


STATus:OPERation:KEYPressed
[:EVENt]
? (reads event register)
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to negative transition filter register)
? (reads negative transition filter register)
:PTRansition
<sp><num value> (writes to positive transition filter register)
? (reads positive transition filter register)

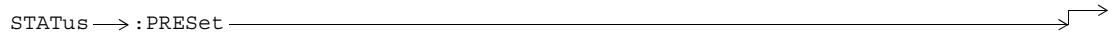

STATus
:OPERation:NMRReady
[:EVENt]
? (reads event register)
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to negative transition filter register)
? (reads negative transition filter register)
:PTRansition
<sp><num value> (writes to positive transition filter register)
? (reads positive transition filter register)


STATus
:OPERation:NMRReady:AMPS
[:EVENt]
? (reads event register)
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to negative transition filter register)
? (reads negative transition filter register)
:PTRansition
<sp><num value> (writes to positive transition filter register)
? (reads positive transition filter register)


STATus
:OPERation:NMRReady:COMMon
[:EVENt]
? (reads event register)
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to negative transition filter register)
? (reads negative transition filter register)
:PTRansition
<sp><num value> (writes to positive transition filter register)
? (reads positive transition filter register)


STATus
:OPERation:NMRReady:DIGital136
? (reads event register)
[:EVENt]
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to negative transition filter register)
? (reads negative transition filter register)
:PTRansition
<sp><num value> (writes to positive transition filter register)
? (reads positive transition filter register)

STATus:PRESet

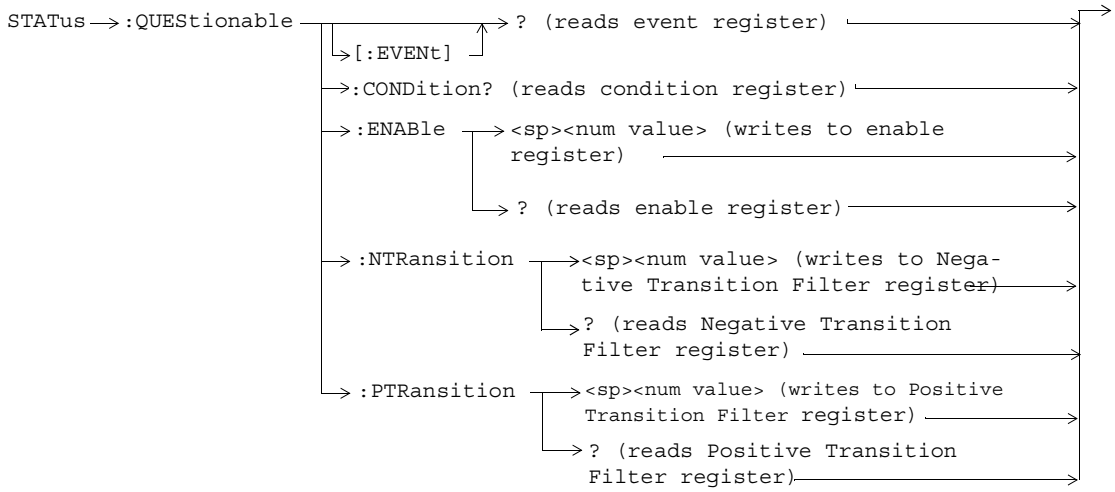


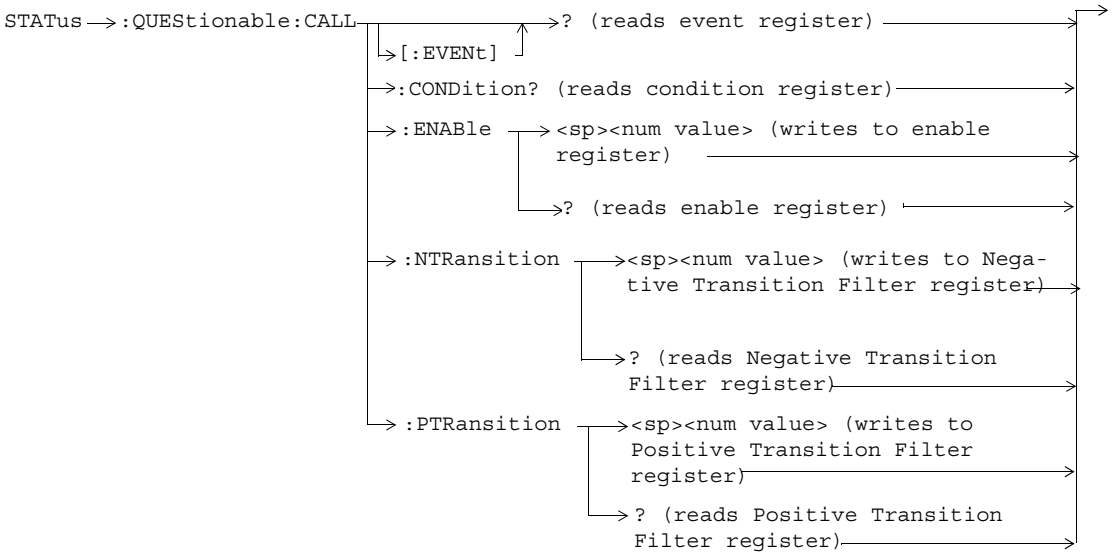
"Diagram Conventions" on page 7

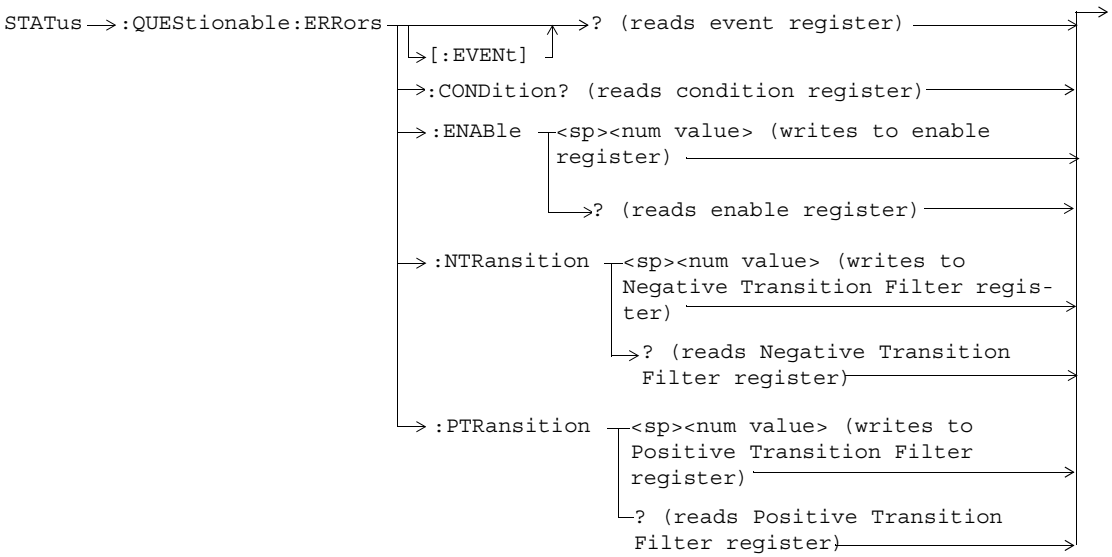
STATus:PRESet

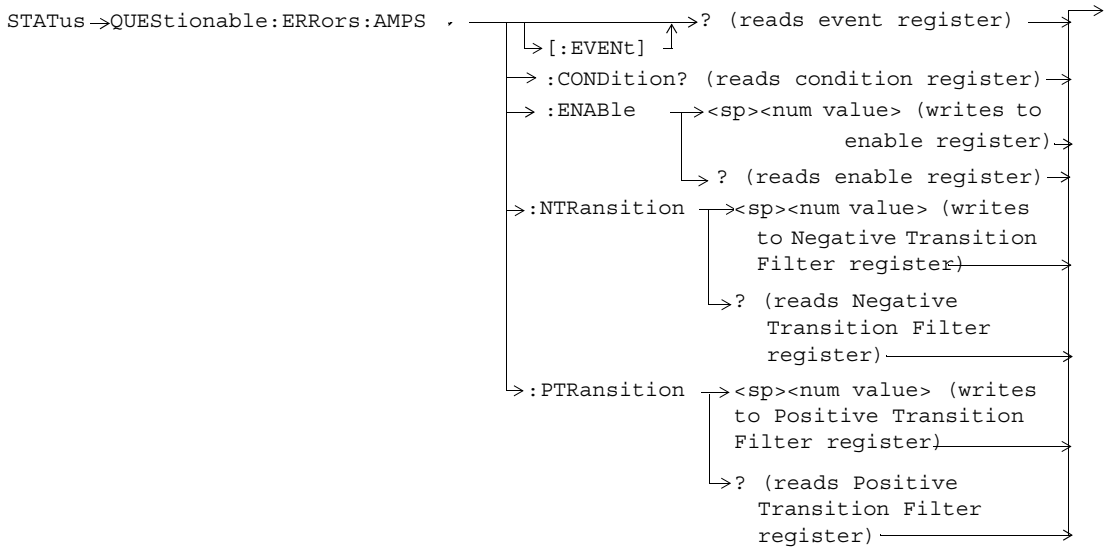
Function	Presets the Status Subsystem Sets all Enable Registers to 0 (not enabled). Sets all Positive Transition Registers (PTR) to 1 (positive transitions enabled). Sets all Negative Transition Registers (NTR) to 0 (negative transitions disabled).

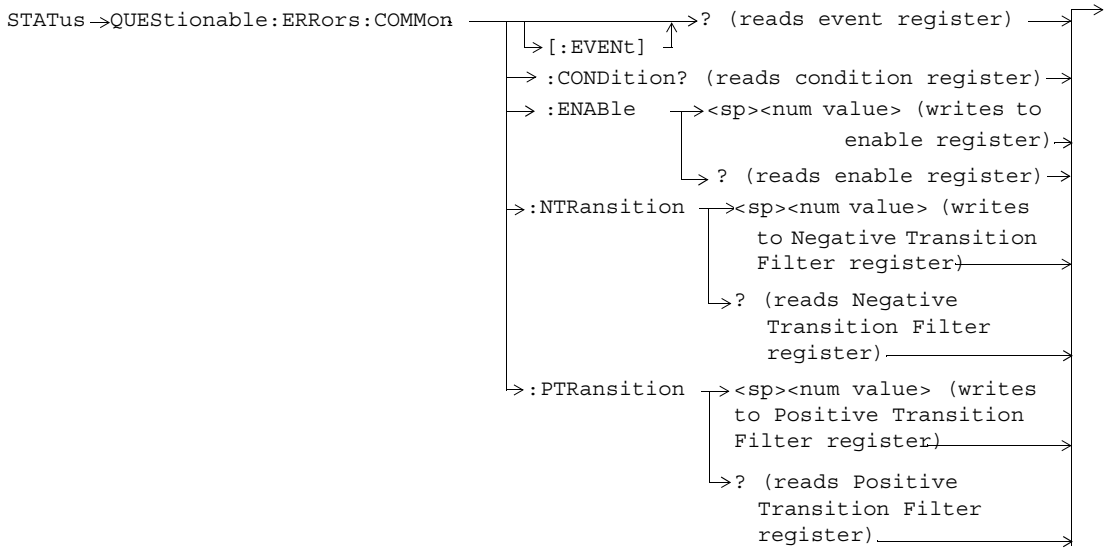
STATus:QUEStionable

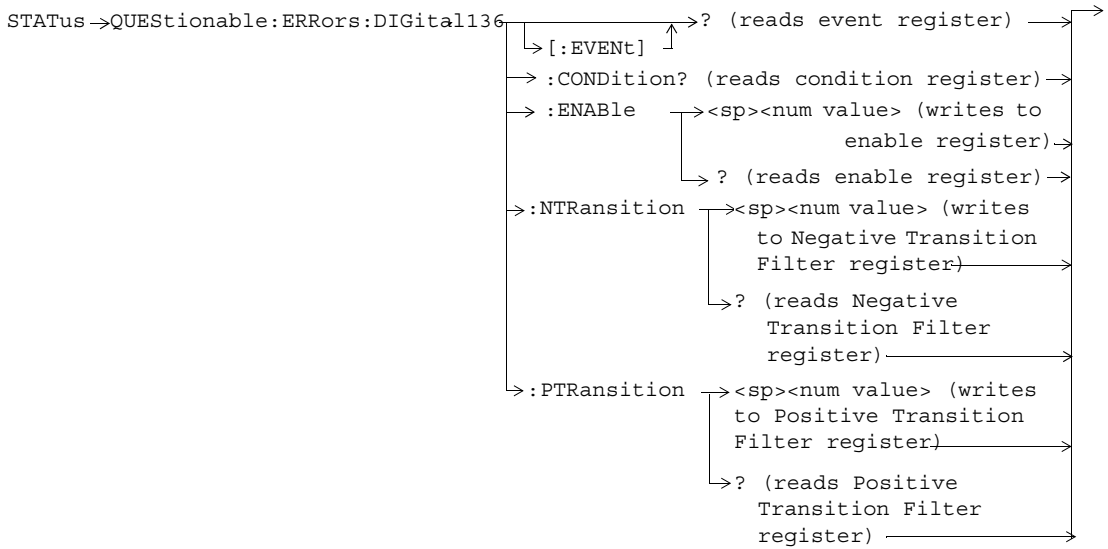


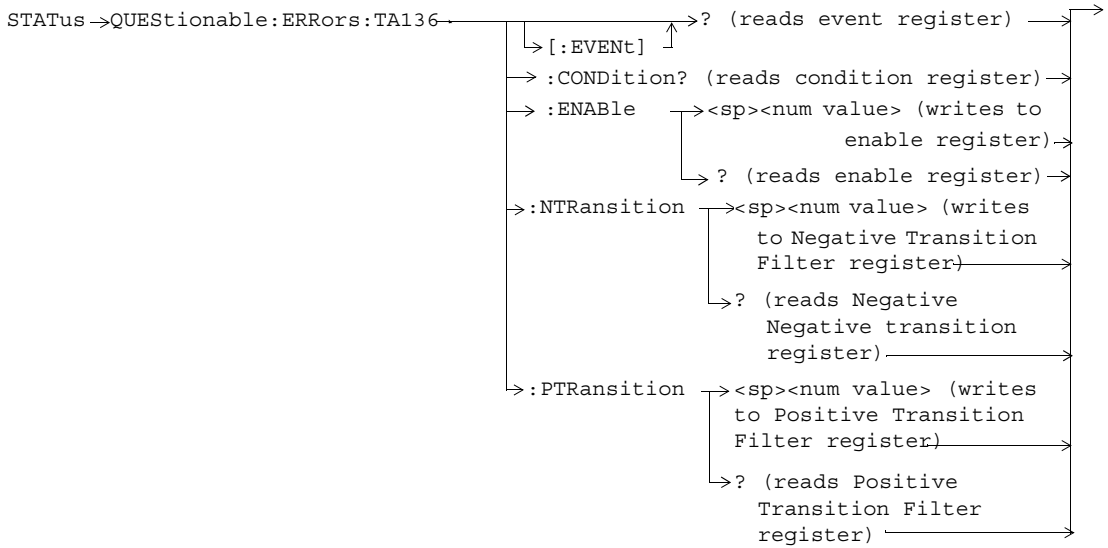

STATus
:QUEStionable:CALL
[:EVENt]
? (reads event register)
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to Negative Transition Filter register)
? (reads Negative Transition Filter register)
:PTRansition
<sp><num value> (writes to Positive Transition Filter register)
? (reads Positive Transition Filter register)

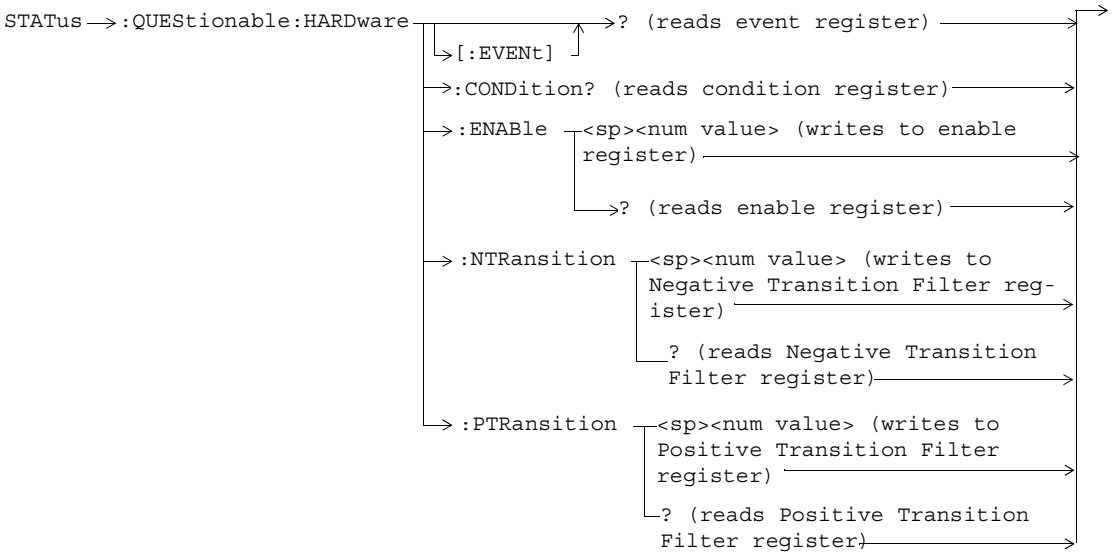

STATus
:QUEStionable:ERRors
[:EVENt]
? (reads event register)
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to Negative Transition Filter register)
? (reads Negative Transition Filter register)
:PTRansition
<sp><num value> (writes to Positive Transition Filter register)
? (reads Positive Transition Filter register)


STATus
QUEStionable:ERRors:AMPS
,
? (reads event register)
[:EVENt]
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to Negative Transition Filter register)
? (reads Negative Transition Filter register)
:PTRansition
<sp><num value> (writes to Positive Transition Filter register)
? (reads Positive Transition Filter register)


STATus
QUEStionable:ERRors:COMMon
? (reads event register)
[:EVENt]
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to Negative Transition Filter register)
? (reads Negative Transition Filter register)
:PTRansition
<sp><num value> (writes to Positive Transition Filter register)
? (reads Positive Transition Filter register)


STATus
QUEStionable:ERRors:DIGital1136
? (reads event register)
[:EVENt]
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to Negative Transition Filter register)
? (reads Negative Transition Filter register)
:PTRansition
<sp><num value> (writes to Positive Transition Filter register)
? (reads Positive Transition Filter register)


STATus
QUEStionable:ERRors:TA136
[:EVENt]
? (reads event register)
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to Negative Transition Filter register)
? (reads Negative Negative transition register)
:PTRansition
<sp><num value> (writes to Positive Transition Filter register)
? (reads Positive Transition Filter register)


STATus
:QUEStionable:HARDware
? (reads event register)
[:EVENt]
:CONDition? (reads condition register)
:ENABle
<sp><num value> (writes to enable register)
? (reads enable register)
:NTRansition
<sp><num value> (writes to Negative Transition Filter register)
? (reads Negative Transition Filter register)
:PTRansition
<sp><num value> (writes to Positive Transition Filter register)
? (reads Positive Transition Filter register)

```
OUTPUT 714;"STATUS:QUESTIONABLE:EVENT?" !Queries and clears the Questionable Event
                                        !Register
OUTPUT 714;"STATUS:QUESTIONABLE:CONDITION?" !Queries and clears the Questionable Condition
                                           !Register
OUTPUT 714;"STATUS:QUESTIONABLE:ENABLE 1024" !Sets the Questionable Enable Register
                                             !for bit 10
OUTPUT 714;"STATUS:QUESTIONABLE:NTRANSITION 2" !Sets the Questionable Negative
                                               !Transition Filter Register for bit 1
OUTPUT 714;"STATUS:QUESTIONABLE:PTRANSITION 2" !Sets the Questionable Positive
                                               !Transition Filter Register for bit 1


OUTPUT 714;"STATUS:QUESTIONABLE:CALL:EVENT?" !Queries and clears the Questionable
                                             !Call Event Register
OUTPUT 714;"STATUS:QUESTIONABLE:CALL:CONDITION?" !Queries and clears the Questionable
                                                 !Call Condition Register
OUTPUT 714;"STATUS:QUESTIONABLE:CALL:ENABLE 1024" !Sets the Questionable
                                                  !Call Enable Register for bit 10
OUTPUT 714;"STATUS:QUESTIONABLE:CALL:NTRANSITION 2" !Sets the Questionable Call
                                                    !Negative Transition Filter Register
                                                    !for bit 1
OUTPUT 714;"STATUS:QUESTIONABLE:CALL:PTRANSITION 2" !Sets the Questionable Call
                                                    !Positive Transition Filter Register
                                                    !for bit 1
```

```
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:EVENT?" !Queries and clears the Questionable
                                               !Errors Event Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:CONDITION?" !Queries and clears the Questionable
                                                   !Errors Condition Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:ENABLE 1024" !Sets the Questionable
                                                    !Errors Enable
                                                    !Register for bit 10
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:NTRANSITION 2" !Sets the Questionable Errors
                                                      !Negative Transition Filter Register
                                                      !for bit 1
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:PTRANSITION 2" !Sets the Questionable Errors
                                                      !Positive Transition Filter Register
                                                      !for bit 1


OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:COMMON:EVENT?" !Queries and clears the Questionable
                                                      !Errors Common Event Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:COMMON:CONDITION?" !Queries the and clears the
                                                          !Questionable Errors
                                                          !Common Condition Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:COMMON:ENABLE 1024" !Sets the Questionable
                                                           !Errors Common Enable
                                                           !Register for bit 10
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:COMMON:NTRANSITION 2" !Sets the Questionable Errors
                                                             !Common Negative Transition
                                                             !Register for bit 1
```

```
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:COMMON:PTRANSITION 2" !Sets the Questionable Errors
                                                             !Common Positive Tranasition
                                                             !Register for bit 1

OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:AMPS:EVENT?" !Queries and clears the Questionable
                                                    !Errors AMPS Event Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:AMPS:CONDITION?" !Queries and clears the Questionable
                                                        !Errors AMPS Condition Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:AMPS:ENABLE 1024" !Sets the Questionable
                                                         !Errors AMPS Enable
                                                         !Register for bit 10
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:AMPS:NTRANSITION 2" !Sets the Questionable Errors
                                                           !AMPS Negative Transition
                                                           !Filter Register for bit 1
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:AMPS:PTRANSITION 2" !Sets the Questionable Errors
                                                           !AMPS Positive Tranasition
                                                           !Filter Register for bit 1

OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:DIGITAL136:EVENT?" !Queries and clears the
                                                          !Questionable Errors
                                                          !Digital136 Event Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:DIGITAL136:CONDITION?" !Queries and clears the
                                                              !Qestionable Errors
                                                              !Digital136 Condition Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:DIGITAL136:ENABLE 1024" !Sets the Questionable
                                                               !Errors Digital136 Enable
```

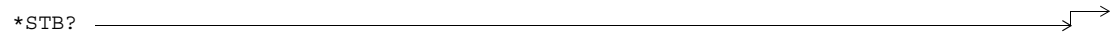
```
                                                                !Register for bit 10
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:DIGITAL136:NTRANSITION 2" !Sets the Questionable
                                                                 !Errors Digital136 Negative
                                                                 !Transition Filter
                                                                 !Register for bit 1
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:DIGITAL136:PTRANSITION 2" !Sets the Questionable
                                                                 !Errors Digital136 Positive
                                                                 !Transition Filter
                                                                 !Register for bit 1

OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:TA136:EVENT?" !Queries and clears the
                                                     !Questionable Errors
                                                     !TA136 Event Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:TA136:CONDITION?" !Queries and clears the
                                                         !Qestionable Errors
                                                         !TA136 Condition Register
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:TA136:ENABLE 1024" !Sets the Questionable
                                                          !Errors TA136 Enable
                                                          !Register for bit 10
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:TA136:NTRANSITION 2" !Sets the Questionable
                                                            !Errors TA136 Negative
                                                            !Transition Filter
                                                            !Register for bit 1
OUTPUT 714;"STATUS:QUESTIONABLE:ERRORS:TA136:PTRANSITION 2" !Sets the Questionable
                                                            !Errors TA136 Positive
```

```
                                                        !Transition Filter
                                                        !Register for bit 1
OUTPUT 714;"STATUS:QUESTIONABLE:HARDWARE:EVENT?" !Queries and clears the Questionable
                                                 !Hardware Event Register
OUTPUT 714;"STATUS:QUESTIONABLE:HARDWARE:CONDITION?" !Queries and clears the Questionable
                                                     !Hardware Condition Register
OUTPUT 714;"STATUS:QUESTIONABLE:HARDWARE:ENABLE 1024" !Sets the Questionable
                                                      !Hardware Enable
                                                      !Register for bit 10
OUTPUT 714;"STATUS:QUESTIONABLE:HARDWARE:NTRANSITION 2" !Sets the Questionable
                                                        !Hardware Negative Transition Filter
                                                        !Register for bit 1
OUTPUT 714;"STATUS:QUESTIONABLE:HARDWARE:PTRANSITION 2" !Sets the Questionable
                                                        !Hardware Positive Transition Filter
                                                        !Register for bit 1
```

Status Byte Register

*STB?



NOTE The Status Byte Register can also be read with a serial poll. For example, the command “Status_byte = SPOLL(714)” would perform a serial poll of the Status Byte Register, returning and releasing RQS (bit 6).

Status Byte Register Bit Assignments

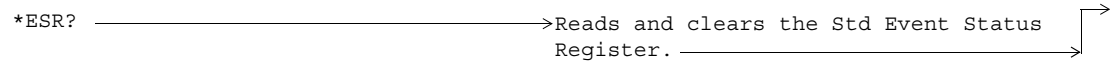
Bit Number	Binary Weighting	Label	Description
7	128	STATus: OPERation	Summarizes the STATus: OPERation Status Register, which fans out to the NMRReady and CALL Status Registers.
6	64	RQS (SRQ TRUE?)/Master Summary Status	RQS is read by a serial poll (SPOLL) Master Summary Status is read by a *STB? query - defined by IEEE 488.2
5	32	Standard Event Status Register	Summarizes the Standard Event Status Register
4	16	Message Available	SCPI - Defined
3	8	STATus: QUEStionable Status Register	Summary Message comes from the STATus: QUEStionable Status Register, which fans out to the CALL and HARDware Status Registers
2	4	Error/ Event Queue	SCPI - Defined

Bit Number	Binary Weighting	Label	Description
1	2	Reserved	
0	1	Reserved	

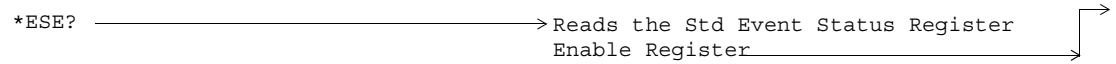
Status Byte Register Bit Assignments

Standard Event Status Register

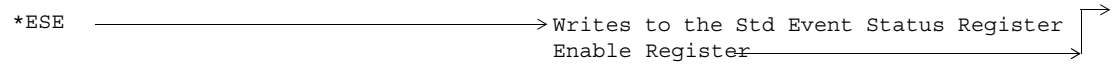
***ESR?**



***ESE?**



***ESE**



“Diagram Conventions” on page 7

Standard Event Status Register Bit Assignment

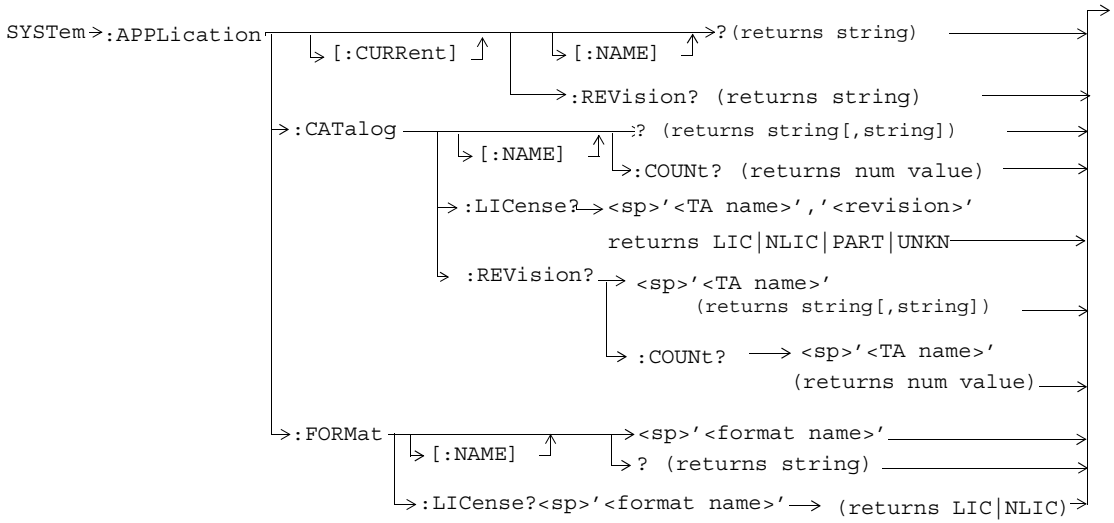
Bit Number	Binary Weighting	Condition	Description
15	32768	Reserved by IEEE.	This bit will always be 0.
14	16384	Reserved by IEEE.	This bit will always be 0.
13	8192	Reserved by IEEE.	This bit will always be 0.
12	4096	Reserved by IEEE.	This bit will always be 0.
11	2048	Reserved by IEEE.	This bit will always be 0.
10	1024	Reserved by IEEE.	This bit will always be 0.
9	512	Reserved by IEEE.	This bit will always be 0.
8	256	Reserved by IEEE.	This bit will always be 0.
7	128	Power On	This bit is set to 1 if the power supply has been turned off and on since the last time this register was read or otherwise cleared. Defined in "IEEE Std. 488.2-1992",11.5.1.1.2
6	64	Reserved for future use.	This bit will always be 0.

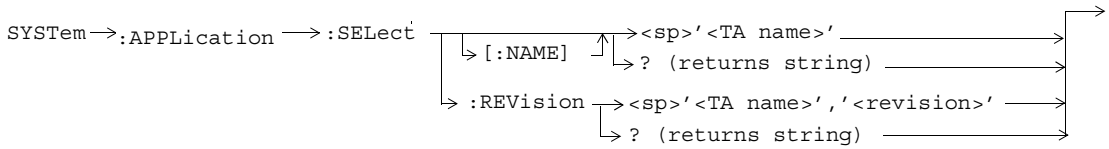
Bit Number	Binary Weighting	Condition	Description
5	32	Command Error	This bit is set to 1 if the test set detects an error while trying to process a command. The following events cause a command error: • An IEEE 488.2 syntax error. The test set received a message that did not follow the syntax defined by the standard. • A semantic error. For example the test set received an incorrectly spelled command. • The test set received a group execution trigger (GET) inside a program message

Bit Number	Binary Weighting	Condition	Description
4	16	Execution Error	This bit is set to 1 if the test set detects an error while trying to execute a command. The following events cause a execution error: • A <PROGRAM DATA> element received in a command is outside the legal range for the test set, or it is inconsistent with the operation of the test set. • The test set could not execute a valid command due to some test set hardware/firmware condition.
3	8	Device Dependent Error	This bit is set to 1 if a test set operation does not execute properly due to an internal condition (such as, overrange). This bit indicates that the error was not a command, query, or execution error.

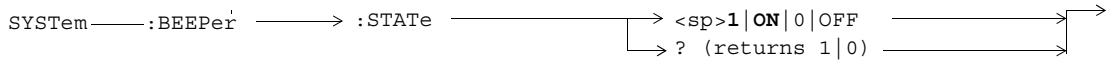
Bit Number	Binary Weighting	Condition	Description
2	4	Query Error	This bit is set to 1 if an error has occurred while trying to read the test set's output queue. The following events cause a query error: • An attempt is made to read data from the output queue when no data is present or is pending. • Data in the output queue has been lost. An example of this would be an output queue overflow.
1	2	Reserved for future use.	This bit will always be 0.
0	1	Operation Complete	This bit is set to 1 when the test set has completed all pending operations and is ready to accept new commands. This bit is only generated in response to the *OPC IEEE 488.2 common command.

SYSTem:APPLication

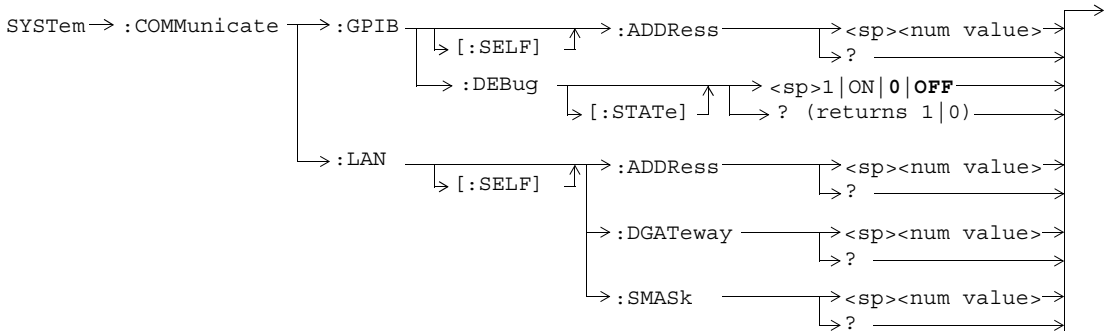



SYSTem
:APPLication
:SELect
[:NAME]
<sp>'<TA name>'
? (returns string)
:REVision
<sp>'<TA name>','<revision>'
? (returns string)

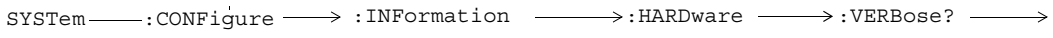
SYSTem:BEEPer



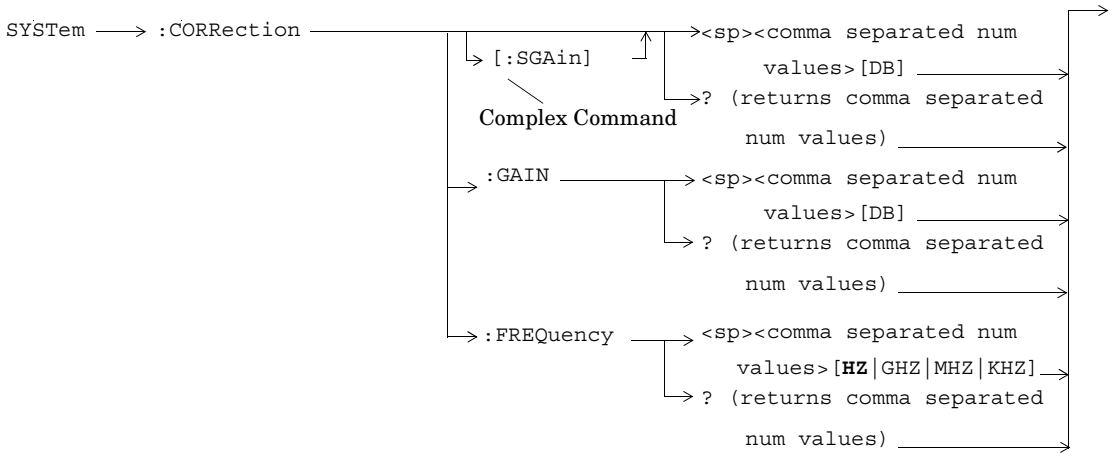
SYSTem:COMMunicate

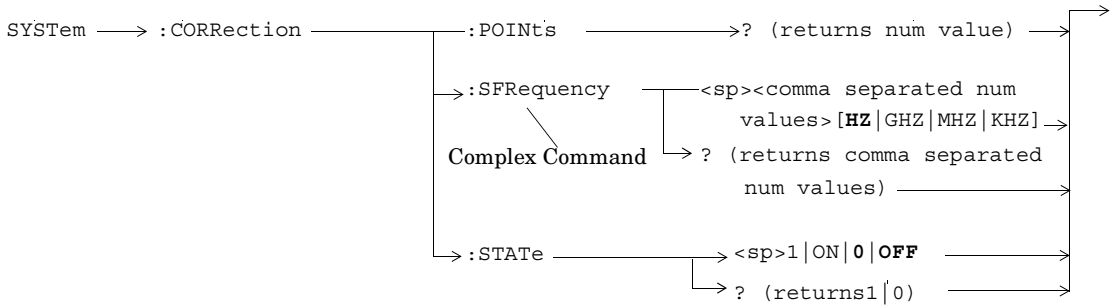


SYSTem:CONFigure:INFormation:HARDware:VERBose?

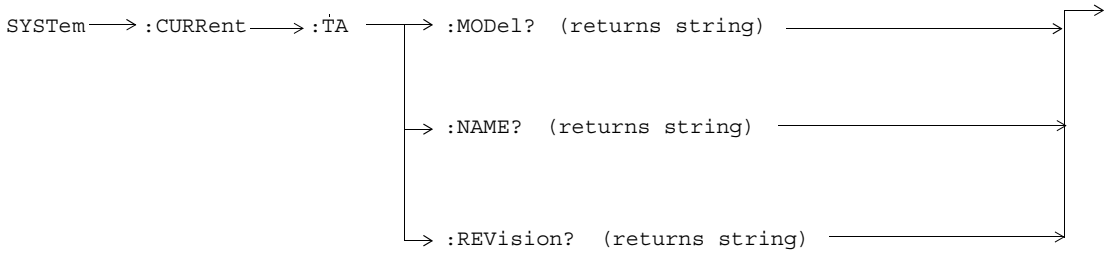


SYSTem:CORRection

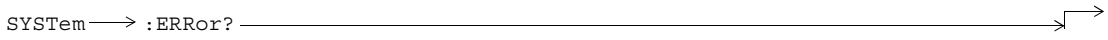



SYSTem
:CORRection
:POINts
? (returns num value)
:SFRequency
<sp><comma separated num values>[HZ|GHZ|MHZ|KHZ]
? (returns comma separated num values)
Complex Command
:STATe
<sp>1|ON|0|OFF
? (returns1|0)

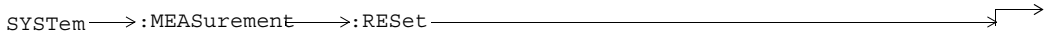
SYSTem:CURRent:TA



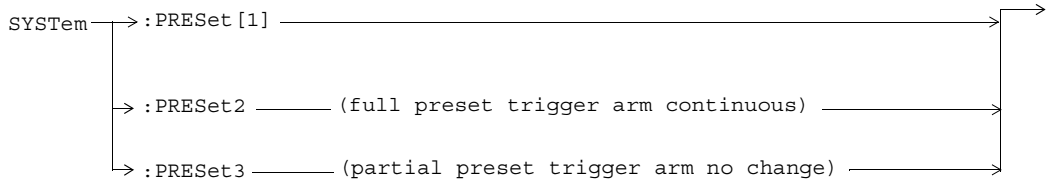
SYSTem:ERRor?



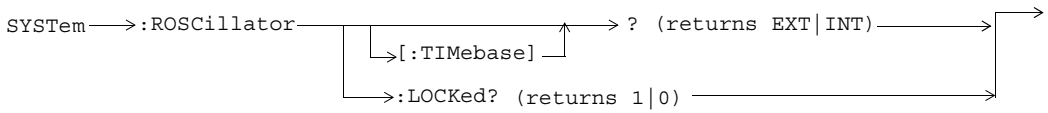
SYSTem:MEASurement:RESet



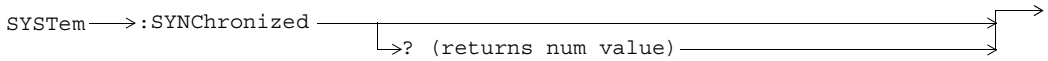
SYSTem:PRESet



SYSTem:ROSCillator



SYSTem:SYNChronized



IEEE 488.2 Common Commands

Description

***CLS**

The *CLS, clear status command, is defined in “IEEE Std 488.2-1992”, 10.3. This command will also clear and close the error message screen on the test set’s display.

***ESE**

The *ESE, standard event status enable command, is defined in “IEEE Std 488.2-1992”, 10.10.

***ESE?**

The *ESE?, standard event status enable query, is defined in “IEEE Std 488.2-1992”, 10.11.

***ESR?**

The *ESR?, standard event status register query, is defined in “IEEE Std 488.2-1992 “,10.12.

***IDN?**

The *IDN?, identification queryis defined in “IEEE Std 488.2-1992”, 10.14.*IDN? is used to retrieve information about the test set in ASCII format.

*IDN?, returns ASCII codes 32 through 126 excluding comma and semicolon in four comma separated fields. Field 1 returns the manufacturer, field 2 returns the instrument model number, field 3 returns the serial number, field 4 returns 0.

***OPC**

The *OPC, operation complete command, is defined in “IEEE 488.2-1992”, 10.18. *OPC causes the test set to continuously sense the No Operation Pending flag. When the No Operation Pending flag becomes TRUE, the OPC event bit in the standard event status register (ESR) is set to indicate that the state of all pending operations is completed. The *OPC common command is not recommended for use as an overlapped command.

***OPC?**

The *OPC?, operation complete query, is defined in “IEEE Std 488.2-1992”, 10.19. The *OPC? query allows synchronization between the controller and the test set using either the message available (MAV) bit in the status byte, or a read of the output OPC?. The *OPC? query does not effect the OPC event bit in the Standard Event Status Register (ESR). The *OPC? common command is not recommended for use as an overlapped command.

***OPT?**

The *OPT?, option identification query, is defined in “IEEE Std 488.2-1992”, 10.20. Each option will have a unique name, that name will be retuned with the query.

***RST**

The *RST, full preset command, is defined in “IEEE Std 488.2-1992”, 10.32. A full preset is also accomplished

using the SYStem:PRESet2 command. A full preset restores the majority of settings to their default values and sets measurement trigger arm to single.

***SRE**

The *SRE, service request enable command, is defined in “IEEE Std 488.2-1992”, 10.34. The parameter range for this command is 0 through 255.

***SRE?**

The *SRE?, service request enable query, is defined in “IEEE Std 488.2-1992”, 10.35. Values returned by this query range from 0 through 255.

***STB?**

The *STB?, read status byte query, is defined in “IEEE Std 488.2-1992”, 10.36. Values returned by this query range from 0 through 255.

***WAI**

The *WAI, wait-to-continue command, is defined in “IEEE Std 488.2-1992”, 10.39. The *WAI command prevents the test set from executing any further commands or queries until all pending operation flags are false. The *WAI common command is not recommended for use as an overlapped command.

Syntax Equivalents

Numerics

100 Hz BW BPF Center Frequency, 93
  audio analyzer, 82
1st Alternate, 49
2nd Alternate, 49

A

ACC (control channel type), 17
ACC Channel, 15
ACP
  See also Adjacent Channel Power
Active Cell, 32
Active Cell Status, 18
Adjacent Channel Power
  1st Alternate (Lower, Upper), 49
  2nd Alternate (Lower, Upper), 49
  Adjacent (Lower, Upper), 49
  average, 73
Adjacent Channel Power Setup
  Measurement Timeout, 80
  Multi-Measurement Count, 79
  Trigger Arm, 79
  Trigger Delay, 80
  Trigger Source, 80
Amplitude, 13
Amplitude Offset RF In/Out Port, 139
Analog MS TX Level (AVC setup info), 42
Analog MS TX Level (current AVC info), 45
Analog Transmit Power
  average, 73
  integrity, 54
  intermediate count, 54
  Mininum, Maximum, Average, Std Dev, 54
Analog Transmit Power Setup
  Measurement Timeout, 86
  Multi-Measurement Count, 85
  Trigger Arm, 85
ATXP
  See also Analog Transmit Power
Audio Analyzer
  Audio Frequency, 51
  Audio Level, 52
  Distortion, 51
  integrity, 51
  intermediate count, 51
  SINAD, 52
Audio Analyzer Setup
  100 Hz BW BPF Center Frequency, 82
  De-Emphasis State, 81
  Detector Type, 82
  Expandor Reference Level, 82
  Filter Type, 82
  Measurement Timeout, 82
  Multi-Measurement Count, 81
  SINAD/Distortion Fundamental Frequency, 82
  SINAD/Distortion State, 82
  Trigger Arm, 81
Audio Frequency, 51
Audio Generator, 13
Audio Level, 52
  average, 73
AVC (traffic channel type), 46
AVC Channel, 41
AVC Channel (current AVC info), 45
AVC Setup Info
  Analog MS TX Level, 42
  AVC Channel, 41
  SAT Color Code, 41
AVC Test, 32

B

Band Pass Filter Frequency
  FM, 93
Beeper State, 136
Bit Error Count, 64, 74
Bit Error Ratio, 64, 74

C

Cal. first IQ mod, 14
Cal. second IQ mod, 14
Cell Power, 37
Cell Power (CW parameters), 37
Cell Power State (CW parameters), 37
Cellular (DCCH band), 20
Cellular (DTC band), 41
Change MS TX Level, 27
Clear Message Log, 47

Control Ch Type
  ACC, 17
  DCCH, 17
Coupling, 13
Current AVC Info
  Analog MS TX Level, 45
  AVC Channel, 45
  SAT Color Code, 45
Current DTC Info
  Digital MS TX Level, 45
  DTC Band, 45
  DTC Channel, 45
  DTC Vocoder Type, 45
Current Traffic Channel Type, 45
CW, 32
CW Parameters
  Cell Power, 37
  Cell Power State, 37
  RF Generator Frequency, 39

D

DCCH (control channel type), 17
DCCH Band
  Cellular, 20
  PCS, 20
DCCH Channel, 20
debug feature, 11
Debug State, 137
De-Emphasis State
  audio analyzer, 81
  FM, 91
Default Gateway, 137
Detector Type
  Peak -, 92
  Peak (audio analyzer), 82
  Peak +, 92
  RMS (audio analyzer), 82
  RMS (FM), 92
Digital MS Protocol Version (mobile station information), 31
Digital MS TX Level (current DTC info), 45
Digital MS TX Level (DTC setup info), 42
Digital Transmit Power
  average, 73
  intermediate count, 57
  Mininum, Maximum, Average, Std Dev, 57
Digital Transmit Power Setup
  Measurement Timeout, 90
  Multi-Measurement Count, 89
  Trigger Arm, 89
  Trigger Delay, 90
  Trigger Source, 90
DigitalTransmit Power
  integrity, 57
Disable MAHO Reporting, 28
Display Brightness
  High, 47
  Medium, 47
Display Mode
  Fast, 47
  Track, 47
Display mode, 47
Distortion
  audio, 51
  audio (average), 73
  FM, 60
  FM (average), 73
Distortion (%)
  Mininum, Maximum, Average, 60
Distortion Fundamental Frequency, 92
Distortion State, 92
DTC (traffic channel type), 46
DTC Band
  Cellular, 41
  PCS, 41
DTC Band (current DTC info), 45
DTC Burst Size
  Normal, 21
  Shortened, 21
DTC Channel, 41
DTC Channel (current DTC info), 45
DTC Induced Err, 23
DTC Setup Info
  Digital MS TX Level, 42
DTC Test, 32
DTC Vocoder Type
  ACELP(CC1), 41
  VSELP, 41
DTC Vocoder Type (current DTC info), 45

DTXP
See also Digital Transmit Power

E

Enable MAHO Reporting, 28
End Call, 22
ESN (Decimal) (mobile station information), 27
ESN (Hex) (mobile station information), 27
EVM
See Also RMS EVM
EVM1
maximum, 74
EVM10
maximum, 74
EVM10 Measurement, 99, 103
Execute Handoff, 25
Expandor Reference Level
audio analyzer, 82
Expandor State, 92
Expected CW Power (receiver control), 75
Expected Power Control (receiver control), 75
Ext FM State, 24

F

Filter Type
100 Hz BW BPF (audio analyzer), 82
100 Hz BW BPF (FM), 93
300 Hz to 15 kHz (audio analyzer), 82
300 to 15 k (FM), 93
50 Hz to 15 kHz (audio analyzer), 82
50 to 15 k (FM), 93
C-Message (audio analyzer), 82
C-Message (FM), 93
None (audio analyzer), 82
None (FM), 93
FM Dev (kHz) RMS
Mininum, Maximum, Average, 59
FM Deviation, 59
average, 73
Freq Error (Hz)
Minimum, Maximum, Average, 67
Mininum, Maximum, Average, 62
Frequency, 13
frequency stability, 62
frequency stability (average), 74
Frequency (kHz)
Mininum, Maximum, Average, 62
Frequency Error, 62
modulation accuracy, 67
Frequency Modulation
Distortion, 60
FM Deviation, 59
integrity, 60
intermediate count, 60
Modulation Frequency, 60
Frequency Modulation Setup
100 Hz BW BPF Center Frequency, 93
De-Emphasis State, 91
Detector Type, 92
Distortion Fundamental Frequency, 92
Distortion State, 92
Expandor State, 92
Filter Type, 93
Measurement Timeout, 93
Multi-Measurement Count, 91
Trigger Arm, 91
Frequency Stability
Frequency, 62
Frequency Error, 62
integrity, 62
intermediate count, 62
Worst Frequency Error, 62
Frequency Stability Setup
Measurement Timeout, 96
Multi-Measurement Count, 95
Trigger Arm, 95

G

GPIB Address, 137

I

Instrument Information
Last Calibration, 14
Test Application, 141
Instrument Setup
Beeper State, 136

Default Gateway, 137
Display Brightness, 47
Display Mode, 47
GPIB Address, 137
LAN IP Address, 137
Maskable Message Display State, 47
Subnet Mask, 137
Int FM Dev, 24
Int FM Freq, 24
integrity
analog transmit power, 54
audio measurements, 51
digital transmit power, 57
FM, 60
frequency stability, 62
intermediate count
analog transmit power, 54
audio measurements, 51
digital transmit power, 57
FM, 60
frequency stability, 62
IQ Tuning
FETCh, 63

L

LAN IP Address, 137
Last Calibration, 14
Loopback BER
Bit Error Count, 64
Bit Error Ratio, 64
bits tested, 74
Loopback BER Setup
Measurement Timeout, 101
Number of bits to test, 101
Trigger Arm, 101

M

MACC
See also Modulation Accuracy
Magnitude Error
maximum, 74
See Also RMS Magnitude Error
MAHO BER, 28
MAHO Info, 28
MAHO Reported Information, 28
MAHO RSSI, 28
Manual Analog Power Cellular Band (receiver control), 76
Manual Digital Power Cellular Band (receiver control), 76
Manual Digital Power PCS Band (receiver control), 76
Maskable Message Display State, 47
MCC, 26
Measurement Frequency
Auto (receiver control), 75
manual (receiver control), 77
Measurement Timeout
adjacent channel power, 80
analog transmit power, 86
audio analyzer, 82
digital transmit power, 90
FM, 93
frequency stability, 96
Loopback BER, 101
modulation accuracy, 99, 103
Message Log, 142
Min 1 (Hex) (mobile station information), 31
Min 2 (Hex) (mobile station information), 31
Mobile Station Information
Digital MS Protocol Version, 31
ESN (Decimal), 27
ESN (Hex), 27
Min 1 (Hex), 31
Min 2 (Hex), 31
MS Called Party Number, 31
Phone Number, 31
Rate Capability, 31
SCM, 31
Modulation Accuracy
Frequency Error, 67
Origin Offset, 69
RMS EVM1, 66
RMS EVM10, 66
RMS Magnitude Error, 67
RMS Phase Error, 69
Modulation Accuracy Setup
EVM10 Measurement, 99, 103

Measurement Timeout, 99, 103
Multi-Measurement Count, 98, 103
Trigger Arm, 98, 103
Trigger Delay, 99, 103
Trigger Source, 99, 103
Modulation Frequency, 60
MS Called Party Number (mobile station information), 31
MS TX Level, 42
Multi-Measurement Count
adjacent channel power, 79
analog transmit power, 85
audio analyzer, 81
digital transmit power, 89
FM, 91
frequency stability, 95
modulation accuracy, 98, 103

N

Normal (DTC burst size), 21
Number of bits to test
Loopback BER, 101

O

Operating Mode
Active Cell, 32
AVC Test, 32
CW, 32
DTC Test, 32
Origin Offset
maximum, 74
modulation accuracy, 69
Origin Offset (dBc)
Minimum, Maximum, Average, 69
Originate Call, 33

P

Paging Number, 34, 35, 38
PCS (DCCH band), 20
PCS (DTC band), 41
Phase Error
maximum, 74
See Also RMS Phase Error
Phone Number (mobile station information), 31
Port Configuration
Amplitude Offset (RF In/Out port), 139
programming, debug feature, 11
Pulse, 13

R

Rate Capability (mobile station information), 31
Receiver Control
Expected CW Power, 75
Expected Power Control, 75
Manual Analog Power Cellular Band, 76
Manual Digital Power Cellular Band, 76
Manual Digital Power PCS Band, 76
Measurement Frequency
Auto, 75
manual, 77
Uplink Frequency
Auto, 75
manual, 77
Rev, License, 135
RF Gen Freq, 39
RF Gen Freq Ctrl, 19
RF Generator Frequency (CW parameters), 39
RMS EVM1
modulation accuracy, 66
RMS EVM1 (%)
Minimum, Maximum, Average, 66
RMS EVM10
modulation accuracy, 66
RMS EVM10 (%)
Minimum, Maximum, Average, 66
RMS Mag Error (%)
Minimum, Maximum, Average, 67
RMS Magnitude Error
modulation accuracy, 67
RMS Phase Error (degrees)
Minimum, Maximum, Average, 69
RMSPhase Error
modulation accuracy, 69

Syntax Equivalents

S

SAT Color Code, 41
SAT Color Code (current AVC info), 45
SAT State, 16
SCM (mobile station information), 31
Shortened (DTC burst size), 21
SINAD, 52
  average, 73
SINAD/Distortion Fundamental Frequency, 82
  audio analyzer, 82
SINAD/Distortion State, 82
  audio analyzer, 82
SOC, 44
Subnet Mask, 137
System ID, 43

T

Test Application
  Test Application, Revision, License, 135
Test Application (instrument information), 141
Test Application Setup, 135
Traffic Ch Type
  AVC, 46
  DTC, 46
Trigger Arm
  adjacent channel power, 79
  analog transmit power, 85
  audio analyzer, 81
  digital transmit power, 89
  FM, 91
  frequency stabiltiy, 95
  Loopback BER, 101
  modulation accuracy, 98, 103
Trigger Delay
  adjacent channel power, 80
  digital transmit power, 90
  IQ Tuning, 99
  MACC, 103
Trigger Source
  adjacent channel power, 80
  digital transmit power, 90
  IQ Tuning, 99
  MACC, 103

U

Uplink Frequency
  Auto (receiver control), 75
  manual (receiver control), 77

V

Vocoder Type
  ACELP(CC1), 41
  VSELP, 41

W

Worst Frequency Error, 62, 74