

PAV2250ADII

1.0.0.1

Generated by Doxygen 1.8.5

Wed Oct 16 2013 17:48:47

Contents

- 1 Module Index** **1**
- 1.1 Modules 1

- 2 File Index** **3**
- 2.1 File List 3

- 3 Module Documentation** **5**
- 3.1 Connect/Disconnect Functions 5
- 3.1.1 Detailed Description 5
- 3.1.2 Function Documentation 5
- 3.1.2.1 PAV2250A_ConnectViaEthernet 5
- 3.1.2.2 PAV2250A_ConnectViaIEEE 6
- 3.1.2.3 PAV2250A_ConnectViaUSB 6
- 3.1.2.4 PAV2250A_DisconnectEthernet 7
- 3.1.2.5 PAV2250A_DisconnectIEEE 7
- 3.1.2.6 PAV2250A_DisconnectUSB 7
- 3.1.2.7 PAV2250A_GetPAV2250ADeviceIDN 8
- 3.1.2.8 PAV2250A_GetPAV2250AUSBDeviceCnt 8
- 3.2 Command Functions 9
- 3.2.1 Detailed Description 9
- 3.2.2 Function Documentation 9
- 3.2.2.1 PAV2250A_IsStable 9
- 3.2.2.2 PAV2250A_PerformGetID 10
- 3.2.2.3 PAV2250A_PerformGetTotalData 11
- 3.2.2.4 PAV2250A_PerformGetTotalDataRow 12
- 3.2.2.5 PAV2250A_PerformGroupExecuteTrigger 12
- 3.2.2.6 PAV2250A_ResetDefaultValues 12
- 3.3 Harmonic Functions 14
- 3.3.1 Detailed Description 15
- 3.3.2 Function Documentation 15
- 3.3.2.1 PAV2250A_GetHarmonicInPhase 15
- 3.3.2.2 PAV2250A_GetHarmonicMagnitude 16

3.3.2.3	PAV2250A_GetHarmonicPhase	16
3.3.2.4	PAV2250A_GetHarmonicQuad	17
3.3.2.5	PAV2250A_GetHarmonicRatioInPhase	17
3.3.2.6	PAV2250A_GetHarmonicRatioMagnitude	17
3.3.2.7	PAV2250A_GetHarmonicRatioPhase	18
3.3.2.8	PAV2250A_GetHarmonicRatioQuad	18
3.3.2.9	PAV2250A_GetHarmonicRatioState	19
3.3.2.10	PAV2250A_PerformGetHarmonicDataRaw	19
3.3.2.11	PAV2250A_PerformGetHarmonics	20
3.3.2.12	PAV2250A_PerformGetHarmonicsRatio	20
3.3.2.13	PAV2250A_SetHarmonicRatioState	21
3.3.2.14	PAV2250A_ViewHarmonic	21
3.3.2.15	PAV2250A_ViewNextHarmonicGroup	22
3.3.2.16	PAV2250A_ViewPrevHarmonicGroup	22
3.4	Reference Range Functions	23
3.4.1	Detailed Description	24
3.4.2	Function Documentation	24
3.4.2.1	PAV2250A_GetRefAutoRange	24
3.4.2.2	PAV2250A_GetRefRangeActualIndex	24
3.4.2.3	PAV2250A_GetRefRangeConfigIndex	25
3.4.2.4	PAV2250A_GetRefRangeIndexSettings	26
3.4.2.5	PAV2250A_GetRefRangeString	27
3.4.2.6	PAV2250A_SetRefAutoRange	28
3.4.2.7	PAV2250A_SetRefRange	29
3.4.2.8	PAV2250A_SetRefRange100MV	29
3.4.2.9	PAV2250A_SetRefRange100V	30
3.4.2.10	PAV2250A_SetRefRange10V	30
3.4.2.11	PAV2250A_SetRefRange1V	30
3.4.2.12	PAV2250A_SetRefRange200MV	31
3.4.2.13	PAV2250A_SetRefRange200V	31
3.4.2.14	PAV2250A_SetRefRange20V	31
3.4.2.15	PAV2250A_SetRefRange2V	32
3.4.2.16	PAV2250A_SetRefRange500MV	32
3.4.2.17	PAV2250A_SetRefRange500V	32
3.4.2.18	PAV2250A_SetRefRange50MV	33
3.4.2.19	PAV2250A_SetRefRange50V	33
3.4.2.20	PAV2250A_SetRefRange5V	33
3.5	Signal Range Functions	35
3.5.1	Detailed Description	36
3.5.2	Function Documentation	36

3.5.2.1	PAV2250A_GetSigAutoRange	36
3.5.2.2	PAV2250A_GetSigRangeActualIndex	36
3.5.2.3	PAV2250A_GetSigRangeConfigIndex	37
3.5.2.4	PAV2250A_GetSigRangeIndexSettings	38
3.5.2.5	PAV2250A_GetSigRangeString	39
3.5.2.6	PAV2250A_SetSigAutoRange	40
3.5.2.7	PAV2250A_SetSigRange	40
3.5.2.8	PAV2250A_SetSigRange100MV	41
3.5.2.9	PAV2250A_SetSigRange100V	41
3.5.2.10	PAV2250A_SetSigRange10V	41
3.5.2.11	PAV2250A_SetSigRange1V	42
3.5.2.12	PAV2250A_SetSigRange200MV	42
3.5.2.13	PAV2250A_SetSigRange200V	42
3.5.2.14	PAV2250A_SetSigRange20V	43
3.5.2.15	PAV2250A_SetSigRange2V	43
3.5.2.16	PAV2250A_SetSigRange500MV	43
3.5.2.17	PAV2250A_SetSigRange500V	44
3.5.2.18	PAV2250A_SetSigRange50MV	44
3.5.2.19	PAV2250A_SetSigRange50V	44
3.5.2.20	PAV2250A_SetSigRange5V	45
3.6	IEEE Functions	46
3.6.1	Detailed Description	46
3.6.2	Function Documentation	46
3.6.2.1	PAV2250A_IEEECLS	46
3.6.2.2	PAV2250A_IEEEGetErrors	46
3.6.2.3	PAV2250A_IEEEReset	47
3.7	Configuration Functions	48
3.7.1	Detailed Description	52
3.7.2	Function Documentation	52
3.7.2.1	PAV2250A_GetCommState	52
3.7.2.2	PAV2250A_GetCustView1	52
3.7.2.3	PAV2250A_GetCustView1Text	53
3.7.2.4	PAV2250A_GetCustView2	53
3.7.2.5	PAV2250A_GetCustView2Text	54
3.7.2.6	PAV2250A_GetCustView3	55
3.7.2.7	PAV2250A_GetCustView3Text	55
3.7.2.8	PAV2250A_GetCustView4	56
3.7.2.9	PAV2250A_GetCustView4Text	57
3.7.2.10	PAV2250A_GetHoldDataState	57
3.7.2.11	PAV2250A_GetIEEELang	58

3.7.2.12	PAV2250A_GetIEEELangText	58
3.7.2.13	PAV2250A_GetMainView	58
3.7.2.14	PAV2250A_GetMainViewText	59
3.7.2.15	PAV2250A_GetReadMode	60
3.7.2.16	PAV2250A_GetReadModeText	60
3.7.2.17	PAV2250A_GetScreenBrightness	61
3.7.2.18	PAV2250A_GetTabView	61
3.7.2.19	PAV2250A_GetTabViewText	62
3.7.2.20	PAV2250A_GoToLocal	62
3.7.2.21	PAV2250A_SetCustView1	62
3.7.2.22	PAV2250A_SetCustView1FundMag	63
3.7.2.23	PAV2250A_SetCustView1InPhase	63
3.7.2.24	PAV2250A_SetCustView1Phase	64
3.7.2.25	PAV2250A_SetCustView1Quad	64
3.7.2.26	PAV2250A_SetCustView1RefVolt	64
3.7.2.27	PAV2250A_SetCustView1SigVolt	65
3.7.2.28	PAV2250A_SetCustView1THD	65
3.7.2.29	PAV2250A_SetCustView2	65
3.7.2.30	PAV2250A_SetCustView2FundMag	66
3.7.2.31	PAV2250A_SetCustView2InPhase	66
3.7.2.32	PAV2250A_SetCustView2Phase	67
3.7.2.33	PAV2250A_SetCustView2Quad	67
3.7.2.34	PAV2250A_SetCustView2RefVolt	67
3.7.2.35	PAV2250A_SetCustView2SigVolt	68
3.7.2.36	PAV2250A_SetCustView2THD	68
3.7.2.37	PAV2250A_SetCustView3	68
3.7.2.38	PAV2250A_SetCustView3FundMag	69
3.7.2.39	PAV2250A_SetCustView3InPhase	69
3.7.2.40	PAV2250A_SetCustView3Phase	70
3.7.2.41	PAV2250A_SetCustView3Quad	70
3.7.2.42	PAV2250A_SetCustView3RefVolt	70
3.7.2.43	PAV2250A_SetCustView3SigVolt	71
3.7.2.44	PAV2250A_SetCustView3THD	71
3.7.2.45	PAV2250A_SetCustView4	71
3.7.2.46	PAV2250A_SetCustView4FundMag	72
3.7.2.47	PAV2250A_SetCustView4InPhase	72
3.7.2.48	PAV2250A_SetCustView4Phase	73
3.7.2.49	PAV2250A_SetCustView4Quad	73
3.7.2.50	PAV2250A_SetCustView4RefVolt	73
3.7.2.51	PAV2250A_SetCustView4SigVolt	74

3.7.2.52	PAV2250A_SetCustView4THD	74
3.7.2.53	PAV2250A_SetHoldDataState	74
3.7.2.54	PAV2250A_SetIEEELang	75
3.7.2.55	PAV2250A_SetIEEELang2250ANative	75
3.7.2.56	PAV2250A_SetIEEELang2250Legacy	76
3.7.2.57	PAV2250A_SetMainView	76
3.7.2.58	PAV2250A_SetMainViewFundMag	77
3.7.2.59	PAV2250A_SetMainViewInPhase	77
3.7.2.60	PAV2250A_SetMainViewPhase	78
3.7.2.61	PAV2250A_SetMainViewQuad	78
3.7.2.62	PAV2250A_SetMainViewRefVolt	78
3.7.2.63	PAV2250A_SetMainViewSigVolt	79
3.7.2.64	PAV2250A_SetMainViewTHD	79
3.7.2.65	PAV2250A_SetReadMode	79
3.7.2.66	PAV2250A_SetReadModeRefRef	80
3.7.2.67	PAV2250A_SetReadModeRefSig	80
3.7.2.68	PAV2250A_SetReadModeSigRef	81
3.7.2.69	PAV2250A_SetReadModeSigSig	81
3.7.2.70	PAV2250A_SetRemoteEthernet	81
3.7.2.71	PAV2250A_SetRemoteIEEE	82
3.7.2.72	PAV2250A_SetRemoteJ1	82
3.7.2.73	PAV2250A_SetRemoteUSB	83
3.7.2.74	PAV2250A_SetScreenBrightness	84
3.7.2.75	PAV2250A_SetTabView	84
3.7.2.76	PAV2250A_SetTabViewCustom	85
3.7.2.77	PAV2250A_SetTabViewHarmonics	85
3.7.2.78	PAV2250A_SetTabViewLVDT	85
3.7.2.79	PAV2250A_SetTabViewMain	86
3.7.2.80	PAV2250A_SetTabViewReference	86
3.8	Calibration Functions	87
3.8.1	Detailed Description	87
3.8.2	Function Documentation	87
3.8.2.1	PAV2250A_Calibrate	87
3.8.2.2	PAV2250A_GetCalState	87
3.9	Miscellaneous Functions	88
3.9.1	Detailed Description	88
3.9.2	Function Documentation	88
3.9.2.1	PAV2250A_LastCmdSent	88
3.9.2.2	PAV2250A_MaxRetry	88
3.9.2.3	PAV2250A_QueryCommand	88

3.9.2.4	PAV2250A_WriteCommand	89
3.10	Time Window Functions	90
3.10.1	Detailed Description	90
3.10.2	Function Documentation	90
3.10.2.1	PAV2250A_GetTimeWndActual	90
3.10.2.2	PAV2250A_GetTimeWndAuto	90
3.10.2.3	PAV2250A_GetTimeWndOverride	91
3.10.2.4	PAV2250A_SetTimeWndAuto	91
3.10.2.5	PAV2250A_SetTimeWndOverride	91
3.11	Setup Options Functions	93
3.11.1	Detailed Description	94
3.11.2	Function Documentation	94
3.11.2.1	PAV2250A_GetAutoSaveOption	94
3.11.2.2	PAV2250A_GetAutoSaveOptionText	95
3.11.2.3	PAV2250A_GetAutoUnitsOption	95
3.11.2.4	PAV2250A_GetAutoUnitsOptionText	96
3.11.2.5	PAV2250A_GetDateDisplayOption	96
3.11.2.6	PAV2250A_GetDateDisplayOptionText	97
3.11.2.7	PAV2250A_GetMainDisplayOption	97
3.11.2.8	PAV2250A_GetMainDisplayOptionText	98
3.11.2.9	PAV2250A_GetNullMeterRangePercent	98
3.11.2.10	PAV2250A_GetSignalInputOption	99
3.11.2.11	PAV2250A_GetSignalInputOptionText	99
3.11.2.12	PAV2250A_GetTimeDisplayOption	100
3.11.2.13	PAV2250A_GetTimeDisplayOptionText	100
3.11.2.14	PAV2250A_GetTouchscreenOption	101
3.11.2.15	PAV2250A_GetTouchscreenOptionText	101
3.11.2.16	PAV2250A_SetAutoSaveDisable	101
3.11.2.17	PAV2250A_SetAutoSaveEnable	102
3.11.2.18	PAV2250A_SetAutoSaveOption	102
3.11.2.19	PAV2250A_SetAutoUnitsDisable	103
3.11.2.20	PAV2250A_SetAutoUnitsEnable	103
3.11.2.21	PAV2250A_SetAutoUnitsOption	103
3.11.2.22	PAV2250A_SetDateDisplayNumeric	104
3.11.2.23	PAV2250A_SetDateDisplayOption	104
3.11.2.24	PAV2250A_SetDateDisplayText	105
3.11.2.25	PAV2250A_SetMainDisplayIndependent	106
3.11.2.26	PAV2250A_SetMainDisplayLinked	106
3.11.2.27	PAV2250A_SetMainDisplayOption	106
3.11.2.28	PAV2250A_SetNullMeterRangePercent	107

3.11.2.29 PAV2250A_SetSignalInputBack	107
3.11.2.30 PAV2250A_SetSignalInputFront	108
3.11.2.31 PAV2250A_SetSignalInputOption	108
3.11.2.32 PAV2250A_SetTimeDisplayAMPM	108
3.11.2.33 PAV2250A_SetTimeDisplayMilitary	109
3.11.2.34 PAV2250A_SetTimeDisplayOption	109
3.11.2.35 PAV2250A_SetTouchscreenDisable	110
3.11.2.36 PAV2250A_SetTouchscreenEnable	111
3.11.2.37 PAV2250A_SetTouchscreenOption	111
3.12 View Configuration Functions	112
3.12.1 Detailed Description	113
3.12.2 Function Documentation	113
3.12.2.1 PAV2250A_GetViewFrequencyConfig	113
3.12.2.2 PAV2250A_GetViewFundMagConfig	113
3.12.2.3 PAV2250A_GetViewIndexConfig	114
3.12.2.4 PAV2250A_GetViewInPhaseConfig	115
3.12.2.5 PAV2250A_GetViewMainConfig	116
3.12.2.6 PAV2250A_GetViewPhaseConfig	117
3.12.2.7 PAV2250A_GetViewQuadConfig	118
3.12.2.8 PAV2250A_GetViewRefVoltConfig	118
3.12.2.9 PAV2250A_GetViewSigOffsetConfig	119
3.12.2.10 PAV2250A_GetViewSigVoltConfig	119
3.12.2.11 PAV2250A_GetViewTHDConfig	120
3.12.2.12 PAV2250A_GetViewTotalRatioConfig	120
3.13 View Max Field Width Functions	122
3.13.1 Detailed Description	123
3.13.2 Function Documentation	123
3.13.2.1 PAV2250A_GetViewFrequencyMaxFieldWidth	123
3.13.2.2 PAV2250A_GetViewFundMagMaxFieldWidth	123
3.13.2.3 PAV2250A_GetViewInPhaseMaxFieldWidth	123
3.13.2.4 PAV2250A_GetViewMainMaxFieldWidth	124
3.13.2.5 PAV2250A_GetViewPhaseMaxFieldWidth	124
3.13.2.6 PAV2250A_GetViewQuadMaxFieldWidth	125
3.13.2.7 PAV2250A_GetViewRefVoltMaxFieldWidth	125
3.13.2.8 PAV2250A_GetViewSigOffsetMaxFieldWidth	125
3.13.2.9 PAV2250A_GetViewSigVoltMaxFieldWidth	126
3.13.2.10 PAV2250A_GetViewTHDMaxFieldWidth	126
3.13.2.11 PAV2250A_GetViewTotalRatioMaxFieldWidth	126
3.13.2.12 PAV2250A_SetViewFrequencyMaxFieldWidth	127
3.13.2.13 PAV2250A_SetViewFundMagMaxFieldWidth	127

3.13.2.14	PAV2250A_SetViewInPhaseMaxFieldWidth	128
3.13.2.15	PAV2250A_SetViewMainMaxFieldWidth	129
3.13.2.16	PAV2250A_SetViewPhaseMaxFieldWidth	129
3.13.2.17	PAV2250A_SetViewQuadMaxFieldWidth	130
3.13.2.18	PAV2250A_SetViewRefVoltMaxFieldWidth	131
3.13.2.19	PAV2250A_SetViewSigOffsetMaxFieldWidth	131
3.13.2.20	PAV2250A_SetViewSigVoltMaxFieldWidth	132
3.13.2.21	PAV2250A_SetViewTHDMaxFieldWidth	133
3.13.2.22	PAV2250A_SetViewTotalRatioMaxFieldWidth	133
3.14	View Units Functions	134
3.14.1	Detailed Description	136
3.14.2	Function Documentation	136
3.14.2.1	PAV2250A_GetViewFrequencyUnits	136
3.14.2.2	PAV2250A_GetViewFrequencyUnitsText	137
3.14.2.3	PAV2250A_GetViewFundMagUnits	137
3.14.2.4	PAV2250A_GetViewFundMagUnitsText	138
3.14.2.5	PAV2250A_GetViewInPhaseUnits	139
3.14.2.6	PAV2250A_GetViewInPhaseUnitsText	139
3.14.2.7	PAV2250A_GetViewMainUnits	140
3.14.2.8	PAV2250A_GetViewMainUnitsText	141
3.14.2.9	PAV2250A_GetViewPhaseUnits	142
3.14.2.10	PAV2250A_GetViewPhaseUnitsText	142
3.14.2.11	PAV2250A_GetViewQuadUnits	143
3.14.2.12	PAV2250A_GetViewQuadUnitsText	143
3.14.2.13	PAV2250A_GetViewRefVoltUnits	144
3.14.2.14	PAV2250A_GetViewRefVoltUnitsText	144
3.14.2.15	PAV2250A_GetViewSigOffsetUnits	145
3.14.2.16	PAV2250A_GetViewSigOffsetUnitsText	145
3.14.2.17	PAV2250A_GetViewSigVoltUnits	146
3.14.2.18	PAV2250A_GetViewSigVoltUnitsText	146
3.14.2.19	PAV2250A_GetViewTHDUnits	147
3.14.2.20	PAV2250A_GetViewTHDUnitsText	147
3.14.2.21	PAV2250A_GetViewTotalRatioUnits	148
3.14.2.22	PAV2250A_GetViewTotalRatioUnitsText	148
3.14.2.23	PAV2250A_SetViewFrequencyHZ	149
3.14.2.24	PAV2250A_SetViewFrequencyKHZ	149
3.14.2.25	PAV2250A_SetViewFundMagDB	149
3.14.2.26	PAV2250A_SetViewFundMagMV	150
3.14.2.27	PAV2250A_SetViewFundMagPercent	150
3.14.2.28	PAV2250A_SetViewFundMagRatio	150

3.14.2.29 PAV2250A_SetViewFundMagV	151
3.14.2.30 PAV2250A_SetViewInPhaseDB	151
3.14.2.31 PAV2250A_SetViewInPhaseMV	151
3.14.2.32 PAV2250A_SetViewInPhasePercent	152
3.14.2.33 PAV2250A_SetViewInPhaseRatio	152
3.14.2.34 PAV2250A_SetViewInPhaseV	152
3.14.2.35 PAV2250A_SetViewMain180	153
3.14.2.36 PAV2250A_SetViewMain360	153
3.14.2.37 PAV2250A_SetViewMainDB	154
3.14.2.38 PAV2250A_SetViewMainMV	155
3.14.2.39 PAV2250A_SetViewMainPercent	155
3.14.2.40 PAV2250A_SetViewMainRatio	156
3.14.2.41 PAV2250A_SetViewMainV	157
3.14.2.42 PAV2250A_SetViewPhase180	157
3.14.2.43 PAV2250A_SetViewPhase360	157
3.14.2.44 PAV2250A_SetViewQuadDB	158
3.14.2.45 PAV2250A_SetViewQuadMV	158
3.14.2.46 PAV2250A_SetViewQuadPercent	158
3.14.2.47 PAV2250A_SetViewQuadRatio	159
3.14.2.48 PAV2250A_SetViewQuadV	159
3.14.2.49 PAV2250A_SetViewRefVoltMV	159
3.14.2.50 PAV2250A_SetViewRefVoltV	160
3.14.2.51 PAV2250A_SetViewSigOffsetMV	160
3.14.2.52 PAV2250A_SetViewSigOffsetV	160
3.14.2.53 PAV2250A_SetViewSigVoltMV	161
3.14.2.54 PAV2250A_SetViewSigVoltV	161
3.14.2.55 PAV2250A_SetViewTHDDB	161
3.14.2.56 PAV2250A_SetViewTHDPercent	162
3.14.2.57 PAV2250A_SetViewTotalRatioDB	162
3.14.2.58 PAV2250A_SetViewTotalRatioPercent	162
3.14.2.59 PAV2250A_SetViewTotalRatioRatio	163
3.15 View Offset Functions	164
3.15.1 Detailed Description	164
3.15.2 Function Documentation	164
3.15.2.1 PAV2250A_GetViewFundMagOffset	164
3.15.2.2 PAV2250A_GetViewInPhaseOffset	165
3.15.2.3 PAV2250A_GetViewMainOffset	165
3.15.2.4 PAV2250A_GetViewPhaseOffset	165
3.15.2.5 PAV2250A_GetViewQuadOffset	166
3.15.2.6 PAV2250A_GetViewRefVoltOffset	166

3.15.2.7	PAV2250A_GetViewSigVoltOffset	167
3.15.2.8	PAV2250A_SetViewFundMagOffset	168
3.15.2.9	PAV2250A_SetViewInPhaseOffset	168
3.15.2.10	PAV2250A_SetViewMainOffset	169
3.15.2.11	PAV2250A_SetViewPhaseOffset	170
3.15.2.12	PAV2250A_SetViewQuadOffset	170
3.15.2.13	PAV2250A_SetViewRefVoltOffset	171
3.15.2.14	PAV2250A_SetViewSigVoltOffset	172
3.16	View Scale Functions	173
3.16.1	Detailed Description	173
3.16.2	Function Documentation	173
3.16.2.1	PAV2250A_GetViewFundMagScale	173
3.16.2.2	PAV2250A_GetViewInPhaseScale	174
3.16.2.3	PAV2250A_GetViewMainScale	174
3.16.2.4	PAV2250A_GetViewPhaseScale	174
3.16.2.5	PAV2250A_GetViewQuadScale	175
3.16.2.6	PAV2250A_GetViewRefVoltScale	175
3.16.2.7	PAV2250A_GetViewSigVoltScale	176
3.16.2.8	PAV2250A_SetViewFundMagScale	177
3.16.2.9	PAV2250A_SetViewInPhaseScale	177
3.16.2.10	PAV2250A_SetViewMainScale	178
3.16.2.11	PAV2250A_SetViewPhaseScale	179
3.16.2.12	PAV2250A_SetViewQuadScale	179
3.16.2.13	PAV2250A_SetViewRefVoltScale	180
3.16.2.14	PAV2250A_SetViewSigVoltScale	181
3.17	View by Index Functions	182
3.17.1	Detailed Description	182
3.17.2	Function Documentation	182
3.17.2.1	PAV2250A_GetViewIndexMaxFieldWidth	182
3.17.2.2	PAV2250A_GetViewIndexOffset	183
3.17.2.3	PAV2250A_GetViewIndexScale	184
3.17.2.4	PAV2250A_GetViewIndexUnits	185
3.17.2.5	PAV2250A_GetViewIndexUnitsText	186
3.17.2.6	PAV2250A_SetViewIndexMaxFieldWidth	188
3.17.2.7	PAV2250A_SetViewIndexOffset	189
3.17.2.8	PAV2250A_SetViewIndexScale	190
3.17.2.9	PAV2250A_SetViewIndexUnits	191
3.18	Reference Voltage Functions	193
3.18.1	Detailed Description	193
3.18.2	Function Documentation	193

3.18.2.1	PAV2250A_GetRef	193
3.18.2.2	PAV2250A_GetRefDC	194
3.18.2.3	PAV2250A_GetRefText	194
3.18.2.4	PAV2250A_GetRefTotalRMS_AC	195
3.18.2.5	PAV2250A_GetRefTotalRMS_ACDC	196
3.18.2.6	PAV2250A_GetRefTotalSum	196
3.18.2.7	PAV2250A_SetRef	197
3.18.2.8	PAV2250A_SetRefDC	198
3.18.2.9	PAV2250A_SetRefTotalRMS_AC	198
3.18.2.10	PAV2250A_SetRefTotalRMS_ACDC	199
3.18.2.11	PAV2250A_SetRefTotalSum	199
3.19	Signal Voltage Functions	200
3.19.1	Detailed Description	200
3.19.2	Function Documentation	200
3.19.2.1	PAV2250A_GetSig	200
3.19.2.2	PAV2250A_GetSigDC	201
3.19.2.3	PAV2250A_GetSigText	201
3.19.2.4	PAV2250A_GetSigTotalRMS_AC	202
3.19.2.5	PAV2250A_GetSigTotalRMS_ACDC	203
3.19.2.6	PAV2250A_GetSigTotalSum	203
3.19.2.7	PAV2250A_SetSig	204
3.19.2.8	PAV2250A_SetSigDC	205
3.19.2.9	PAV2250A_SetSigTotalRMS_AC	205
3.19.2.10	PAV2250A_SetSigTotalRMS_ACDC	206
3.19.2.11	PAV2250A_SetSigTotalSum	206
3.20	Independent Component Functions	207
3.20.1	Detailed Description	207
3.20.2	Function Documentation	207
3.20.2.1	PAV2250A_GetFrequency	207
3.20.2.2	PAV2250A_GetSampleRateIndex	207
3.20.2.3	PAV2250A_GetSigOffset	208
3.20.2.4	PAV2250A_GetTHD	208
3.20.2.5	PAV2250A_GetTotalRatio	208
3.21	Data Buffer Functions	210
3.21.1	Detailed Description	210
3.21.2	Function Documentation	210
3.21.2.1	HandleBufferDataSetup	210
3.21.2.2	HandleBufferedDataState	211
3.21.2.3	HandleBufferedDataValues	213
3.21.2.4	PAV2250A_BufferCapture	214

3.21.2.5	PAV2250A_BufferGet	216
3.21.2.6	PAV2250A_BufferStop	216
3.21.2.7	PAV2250A_GetBufferedData	216
3.21.2.8	PAV2250A_GetBufferedDataState	217
3.21.2.9	PAV2250A_GetBufferedPageIndex	217
3.21.2.10	PAV2250A_SetBufferedDataState	218
3.21.2.11	PAV2250A_SetBufferedPageIndex	220
3.22	Internal Reference Functions	221
3.22.1	Detailed Description	221
3.22.2	Function Documentation	221
3.22.2.1	PAV2250A_GetIntRefFreq	221
3.22.2.2	PAV2250A_GetIntRefMeasCur	222
3.22.2.3	PAV2250A_GetIntRefOutputState	222
3.22.2.4	PAV2250A_GetIntRefOverCurState	222
3.22.2.5	PAV2250A_GetIntRefRemoteSense	223
3.22.2.6	PAV2250A_GetIntRefSenseDir	223
3.22.2.7	PAV2250A_GetIntRefVolt	224
3.22.2.8	PAV2250A_ResetIntRefOverCur	224
3.22.2.9	PAV2250A_SetIntRefFreq	225
3.22.2.10	PAV2250A_SetIntRefOutputState	226
3.22.2.11	PAV2250A_SetIntRefRemoteSense	226
3.22.2.12	PAV2250A_SetIntRefSenseDir	227
3.22.2.13	PAV2250A_SetIntRefVolt	227
3.23	Local Functions (not exported)	228
3.23.1	Detailed Description	228
3.23.2	Function Documentation	228
3.23.2.1	ExecuteRemoteCmd	228
3.23.2.2	ExecuteRemoteCmdBool	229
3.23.2.3	ExecuteRemoteCmdFloat	229
3.23.2.4	ExecuteRemoteCmdInt	230
3.23.2.5	ExecuteRemoteCmdString	231
3.23.2.6	GetRange	232
3.23.2.7	HexStr2Declnt	232
3.23.2.8	IsLanguageTypeLegacy	233
3.23.2.9	IsLanguageTypeNative	233
3.23.2.10	ParseForCommaSeparatedDataElements	233
3.23.2.11	SendIEEEMessage	233
3.23.2.12	WaitForResponse	234
3.24	LVDT Functions	235
3.24.1	Detailed Description	236

3.24.2	Function Documentation	236
3.24.2.1	PAV2250A_GetLVDT4WireAlgorithm	236
3.24.2.2	PAV2250A_GetLVDT4WireAlgorithmText	236
3.24.2.3	PAV2250A_GetLVDTEnabled	237
3.24.2.4	PAV2250A_GetLVDTEnabledText	237
3.24.2.5	PAV2250A_GetLVDTPOFF	237
3.24.2.6	PAV2250A_GetLVDTPosition	238
3.24.2.7	PAV2250A_GetLVDTScale	238
3.24.2.8	PAV2250A_GetLVDTSignal	238
3.24.2.9	PAV2250A_GetLVDTSignalText	239
3.24.2.10	PAV2250A_GetLVDTType	239
3.24.2.11	PAV2250A_GetLVDTTypeText	240
3.24.2.12	PAV2250A_GetLVDTVA	240
3.24.2.13	PAV2250A_GetLVDTVB	240
3.24.2.14	PAV2250A_SetLVDT4WireAlgorithm	241
3.24.2.15	PAV2250A_SetLVDT4WireVA_VAPLUSVB	241
3.24.2.16	PAV2250A_SetLVDT4WireVA_VB	242
3.24.2.17	PAV2250A_SetLVDTDisabled	243
3.24.2.18	PAV2250A_SetLVDTEnabled	243
3.24.2.19	PAV2250A_SetLVDTPOFF	243
3.24.2.20	PAV2250A_SetLVDTScale	244
3.24.2.21	PAV2250A_SetLVDTSignal	244
3.24.2.22	PAV2250A_SetLVDTSignalFund	245
3.24.2.23	PAV2250A_SetLVDTSignalINPH	245
3.24.2.24	PAV2250A_SetLVDTSignalTotal	245
3.24.2.25	PAV2250A_SetLVDTType	245
3.24.2.26	PAV2250A_SetLVDTType2Wire	246
3.24.2.27	PAV2250A_SetLVDTType3Wire	246
3.24.2.28	PAV2250A_SetLVDTType4Wire	247
3.25	Reference Only Functions	249
3.25.1	Detailed Description	249
3.25.2	Function Documentation	249
3.25.2.1	PAV2250A_GetRefFundMag	249
3.25.2.2	PAV2250A_GetRefFundMagRatio	249
3.25.2.3	PAV2250A_GetRefInPhase	250
3.25.2.4	PAV2250A_GetRefInPhaseRatio	250
3.25.2.5	PAV2250A_GetRefPhase	250
3.25.2.6	PAV2250A_GetRefQuad	251
3.25.2.7	PAV2250A_GetRefQuadRatio	251
3.25.2.8	PAV2250A_GetRefTHD	251

3.25.2.9	PAV2250A_GetRefTotal	252
3.25.2.10	PAV2250A_GetRefTotalRatio	252
4	File Documentation	253
4.1	E:/(BLACKFIN CODE)/Instruments/PAV-2250A/Driver_VS2010/Source/PAV2250ADII/PAV2250A-Dll.cpp File Reference	253
4.1.1	Function Documentation	274
4.1.1.1	DllMain	274
4.1.2	Variable Documentation	274
4.1.2.1	PAV2250ADII_VERSION	274
4.2	E:/(BLACKFIN CODE)/Instruments/PAV-2250A/Driver_VS2010/Source/PAV2250ADII/PAV2250A-Dll.h File Reference	274
4.2.1	Macro Definition Documentation	295
4.2.1.1	_PAV2250AClass	295
4.2.1.2	_PAV2250AFUNC	295
4.2.1.3	bool	295
4.2.2	Enumeration Type Documentation	295
4.2.2.1	PAV_IEEE_LANGUAGES	295
4.2.2.2	PAV_INT_REF_OUTPUT_STATE	295
4.2.2.3	PAV_INT_REF_OVER_CURRENT_STATE	295
4.2.2.4	PAV_INT_REF_REMOTE_GEN_DIR	296
4.2.2.5	PAV_LVDT_4WIRE_ALGORITHM	296
4.2.2.6	PAV_LVDT_SIGNAL	296
4.2.2.7	PAV_LVDT_TYPE	296
4.2.2.8	PAV_OPTIONS_AUTOSAVE	296
4.2.2.9	PAV_OPTIONS_AUTOUNITS	296
4.2.2.10	PAV_OPTIONS_DATEDISPLAY	296
4.2.2.11	PAV_OPTIONS_MAINDISPLAY	297
4.2.2.12	PAV_OPTIONS_RATIO	297
4.2.2.13	PAV_OPTIONS_SIGNALINPUT	297
4.2.2.14	PAV_OPTIONS_TIMEDISPLAY	297
4.2.2.15	PAV_OPTIONS_TOUCHSCREEN	297
4.2.2.16	PAV_Ranges	297
4.2.2.17	PAV_ReadModes	298
4.2.2.18	PAV_REMOTE_SENSE_STATE	298
4.2.2.19	PAV_STATUS	298
4.2.2.20	PAV_Tabs	298
4.2.2.21	PAV_Units	299
4.2.2.22	PAV_Views	299
4.2.2.23	PAV_VoltViews	299

[Index](#)

300

Chapter 1

Module Index

1.1 Modules

Here is a list of all modules:

- Connect/Disconnect Functions 5
- Command Functions 9
- Harmonic Functions 14
- Reference Range Functions 23
- Signal Range Functions 35
- IEEE Functions 46
- Configuration Functions 48
- Calibration Functions 87
- Miscellaneous Functions 88
- Time Window Functions 90
- Setup Options Functions 93
- View Configuration Functions 112
- View Max Field Width Functions 122
- View Units Functions 134
- View Offset Functions 164
- View Scale Functions 173
- View by Index Functions 182
- Reference Voltage Functions 193
- Signal Voltage Functions 200
- Independent Component Functions 207
- Data Buffer Functions 210
- Internal Reference Functions 221
- Local Functions (not exported) 228
- LVDT Functions 235
- Reference Only Functions 249

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

E:\(BLACKFIN CODE)\Instruments\PAV-2250A\Driver_VS2010\Source\PAV2250ADII\PAV2250ADII.cpp 253
E:\(BLACKFIN CODE)\Instruments\PAV-2250A\Driver_VS2010\Source\PAV2250ADII\PAV2250ADII.h . 274

Chapter 3

Module Documentation

3.1 Connect/Disconnect Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_ConnectViaIEEE](#) (int nPAVNo, int nIEEEAddr, int nIEEELang)
PAV2250A_ConnectViaIEEE sets up and opens the connection to communicate to the PAV2250A via IEEE. The IEEE supports the following language protocols:
- [_PAV2250AFUNC](#) int [PAV2250A_ConnectViaUSB](#) (int nPAVNo, int nDeviceNo)
PAV2250A_ConnectViaUSB sets up and opens the connection to communicate to the PAV2250A via USB. Prior to calling this function, make calls to the [PAV2250A_GetPAV2250AUSBDeviceCnt\(\)](#) routine to determine the number of Cypress USB Devices detected in your system and the [PAV2250A_GetPAV2250ADeviceIDN\(\)](#) routine to determine the appropriate PAV identifier (nPAVNo) associated with the Cypress USB Devices that are connected to PAV2250A via USB.
- [_PAV2250AFUNC](#) int [PAV2250A_ConnectViaEthernet](#) (int nPAVNo, char *szIPAddr, int nPort)
PAV2250A_ConnectViaEthernet sets up and opens the connection to communicate to the 2250A via Ethernet.
- [_PAV2250AFUNC](#) int [PAV2250A_DisconnectIEEE](#) (int nPAVNo)
PAV2250A_DisconnectIEEE closes the connection to communicate to the 2250A via IEEE
- [_PAV2250AFUNC](#) int [PAV2250A_DisconnectUSB](#) (int nPAVNo)
PAV2250A_DisconnectUSB closes the connection to communicate to the 2250A via USB.
- [_PAV2250AFUNC](#) int [PAV2250A_DisconnectEthernet](#) (int nPAVNo)
PAV2250A_DisconnectEthernet closes the connection to communicate to the 2250A via Ethernet.
- [_PAV2250AFUNC](#) int [PAV2250A_GetPAV2250AUSBDeviceCnt](#) (int *pnUSBDeviceCnt)
PAV2250A_GetPAV2250AUSBDeviceCnt invokes the Cypress driver and returns the number of Cypress USB Devices detected with your computer system.
- [_PAV2250AFUNC](#) int [PAV2250A_GetPAV2250ADeviceIDN](#) (int nDeviceNo, char *pszID)
PAV2250A_GetPAV2250ADeviceIDN sends the IDN command to get Device ID string for the device. The ID returned includes the manufacturer (NORTH ATLANTIC), the 2250A module, serial number, and revision information. NOTE: this function does not require an Open call to have already been performed, but it does assume a USB physical connection. If you are connecting to the instrument using one of the other methods (IEEE or Ethernet) you must first "connect" to the device using that method and then call [PAV2250A_PerformGetID](#).

3.1.1 Detailed Description

3.1.2 Function Documentation

3.1.2.1 [_PAV2250AFUNC](#) int [PAV2250A_ConnectViaEthernet](#) (int nPAVNo, char * szIPAddr, int nPort)

[PAV2250A_ConnectViaEthernet](#) sets up and opens the connection to communicate to the 2250A via Ethernet.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>szIPAddr</i>	: (Input) IPv4 Address to be used to connect to 2250A.
<i>nPort</i>	: (Input) Port to be used to connect to 2250A.

Returns

- PAV_SUCCESS : Successful connection via Ethernet
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to perform ethernet login to 2250A
- PAV_ERROR_ETHER_CONNECTION : Ethernet connection error

See Also

CreateClientSocket, Ethernet_WriteMsg, [PAV2250A_SetRemoteEthernet](#)

3.1.2.2 _PAV2250AFUNC int PAV2250A_ConnectViaIEEE (int nPAVNo, int nIEEEAddr, int nIEEELang)

PAV2250A_ConnectViaIEEE sets up and opens the connection to communicate to the PAV2250A via IEEE. The IEEE supports the following language protocols:

- PAV-2250A Native
- PAV-2250 Native (Legacy)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nIEEEAddr</i>	: (Input) IEEE Address to be used to connect to PAV2250A. (0-30)
<i>nIEEELang</i>	: (Input) Language Protocol to be used to communicate via IEEE to PAV2250A. <ul style="list-style-type: none"> • 2250A Language Types: <ul style="list-style-type: none"> • PAV2250A_NATIVE 0 • PAV2250_LEGACY 1

Returns

- PAV_SUCCESS : Successful connection via IEEE using specified address and language protocol
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_ADDRS : Invalid IEEE Address parameter
- PAV_ERROR_LANG : Invalid PAV2250A Language parameter
- PAV_ERROR_OPEN_PAV_SESSION : IEEE connection or configuration error

See Also

[PAV2250A_SetIEEELang](#), [PAV2250A_SetRemoteIEEE](#)

3.1.2.3 _PAV2250AFUNC int PAV2250A_ConnectViaUSB (int nPAVNo, int nDeviceNo)

PAV2250A_ConnectViaUSB sets up and opens the connection to communicate to the PAV2250A via USB. Prior to calling this function, make calls to the [PAV2250A_GetPAV2250AUSBDeviceCnt\(\)](#) routine to determine the number of Cypress USB Devices detected in your system and the [PAV2250A_GetPAV2250ADeviceIDN\(\)](#) routine to determine the appropriate PAV identifier (nPAVNo) associated with the Cypress USB Devices that are connected to PAV2250A via USB.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nDeviceNo</i>	: (Input) USB device number to open and communication via USB with 2250A.

Returns

- PAV_SUCCESS : Successful connection via USB
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_USB_CONNECTION : USB connection or configuration error

See Also

USB_Connect, [PAV2250A_SetRemoteUSB](#)

3.1.2.4 **_PAV2250AFUNC** int PAV2250A_DisconnectEthernet (int *nPAVNo*)

PAV2250A_DisconnectEthernet closes the connection to communicate to the 2250A via Ethernet.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

See Also

CloseClientSocket

3.1.2.5 **_PAV2250AFUNC** int PAV2250A_DisconnectIEEE (int *nPAVNo*)

PAV2250A_DisconnectIEEE closes the connection to communicate to the 2250A via IEEE

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

See Also

PAV2250A_GoToLocal(nPAVNo)

3.1.2.6 **_PAV2250AFUNC** int PAV2250A_DisconnectUSB (int *nPAVNo*)

PAV2250A_DisconnectUSB closes the connection to communicate to the 2250A via USB.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

See Also

USB_Disconnect, PAV2250A_GoToLocal(nPAVNo)

3.1.2.7 `_PAV2250AFUNC int PAV2250A_GetPAV2250ADeviceIDN (int nDeviceNo, char * pszID)`

PAV2250A_GetPAV2250ADeviceIDN sends the IDN command to get Device ID string for the device. The ID returned includes the manufacturer (NORTH ATLANTIC), the 2250A module, serial number, and revision information. NOTE: this function does not require an Open call to have already been performed, but it does assume a USB physical connection. If you are connecting to the instrument using one of the other methods (IEEE or Ethernet) you must first "connect" to the device using that method and then call PAV2250A_PerformGetID.

Parameters

<i>nDeviceNo</i>	: (Input) USB device number to open and communication via USB with 2250A.
<i>pszID</i>	: (Output) Device ID of the specified PAV

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

3.1.2.8 `_PAV2250AFUNC int PAV2250A_GetPAV2250AUSBDeviceCnt (int * pnUSBDeviceCnt)`

PAV2250A_GetPAV2250AUSBDeviceCnt invokes the Cypress driver and returns the number of Cypress USB Devices detected with your computer system.

Parameters

<i>pnUSBDevice-Cnt</i>	: (Output) Number of Cypress USB Devices detected.
------------------------	--

Returns

- PAV_SUCCESS : Function is successful

See Also

USB_GetCypressUSBDeviceCnt

3.2 Command Functions

Functions

- [_PAV2250AFUNC](#)** int **[PAV2250A_PerformGetID](#)** (int nPAVNo, char *pszID)
PAV2250A_PerformGetID sends the IDN command to get Device ID string for the device. The ID returned includes the manufacturer (NORTH ATLANTIC), the 2250A module, serial number, and revision information.
- [_PAV2250AFUNC](#)** int **[PAV2250A_IsStable](#)** (int nPAVNo, bool *pbStable)
PAV2250A_IsStable returns whether or not the PAV is considered to be "stable". (i.e. the unit has settled long enough to allow for accurate readings to be taken.)
- [_PAV2250AFUNC](#)** int **[PAV2250A_PerformGetTotalData](#)** (int nPAVNo, char *pszTotalData)
PAV2250A_PerformGetTotalData sends the Total Data command to get Total Data for the device. NOTE: This function should only be called if connecting via Ethernet or IEEE. If connecting by USB, you should fetch the individual components that make up "TotalData" one at a time (Total Ratio, Ref Total RMS AC, Sig Total RMS AC, THD, Frequency, Sample Rate Index, Ref Range, Sig Range, Total Sig Offset, Sig Total Sum, Ref Total RMS AC/DC, Sig Total RMS AC/DC, RefDC and SigDC). Current USB driver in use has a restriction on amount of data it can return from one call.
- [_PAV2250AFUNC](#)** int **[PAV2250A_PerformGetTotalDataRaw](#)** (int nPAVNo, int nReceiveBufferSize, char *pszTotalData)
PAV2250A_PerformGetTotalDataRaw sends the Total Data command to get Raw Total Data for the device. NOTE: This function should only be called if connecting via Ethernet or IEEE. Current USB driver in use has a restriction on amount of data it can return from one call.
- [_PAV2250AFUNC](#)** int **[PAV2250A_PerformGroupExecuteTrigger](#)** (int nPAVNo, int nIEEEAddr)
PAV2250A_PerformGroupExecuteTrigger performs a group execute trigger.
- [_PAV2250AFUNC](#)** int **[PAV2250A_ResetDefaultValues](#)** (int nPAVNo)
PAV2250A_ResetDefaultValues sends the command to set the device settings back to the factory default values.

3.2.1 Detailed Description

3.2.2 Function Documentation

3.2.2.1 **[_PAV2250AFUNC](#)** int **[PAV2250A_IsStable](#)** (int nPAVNo, bool * pbStable)

[PAV2250A_IsStable](#) returns whether or not the PAV is considered to be "stable". (i.e. the unit has settled long enough to allow for accurate readings to be taken.)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbStable</i>	: (Output) <ul style="list-style-type: none"> • true : if stable • false : if NOT stable

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function supported only with PAV2250A_NATIVE
- PAV_ERROR_WRITE : Unable to send command to 2250A

See Also

[ExecuteRemoteCmdBool](#)

3.2.2.2 `_PAV2250AFUNC` int `PAV2250A_PerformGetID` (int *nPAVNo*, char * *pszID*)

`PAV2250A_PerformGetID` sends the IDN command to get Device ID string for the device. The ID returned includes the manufacturer (NORTH ATLANTIC), the 2250A module, serial number, and revision information.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszID</i>	: (Output) Device ID of the specified PAV

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.2.2.3 **_PAV2250AFUNC** int PAV2250A_PerformGetTotalData (int *nPAVNo*, char * *pszTotalData*)

PAV2250A_PerformGetTotalData sends the Total Data command to get Total Data for the device. NOTE: This function should only be called if connecting via Ethernet or IEEE. If connecting by USB, you should fetch the individual components that make up "TotalData" one at a time (Total Ratio, Ref Total RMS AC, Sig Total RMS AC, THD, Frequency, Sample Rate Index, Ref Range, Sig Range, Total Sig Offset, Sig Total Sum, Ref Total RMS AC/DC, Sig Total RMS AC/DC, RefDC and SigDC). Current USB driver in use has a restriction on amount of data it can return from one call.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszTotalData</i>	: (Output) Comma delimited string containing: <ul style="list-style-type: none"> • Total Ratio • Total Ref Volt • Total Sig Volt • THD • Frequency • Sample Rate Index • Ref Range • Sig Range

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function supported only with PAV2250A_NATIVE
- PAV_ERROR_WRITE : Unable to send command to 2250A

See Also

[ExecuteRemoteCmdString](#)

3.2.2.4 `_PAV2250AFUNC` int PAV2250A_PerformGetTotalDataRow (int *nPAVNo*, int *nReceiveBufferSize*, char * *pszTotalData*)

PAV2250A_PerformGetTotalDataRow sends the Total Data command to get Raw Total Data for the device. NOTE: This function should only be called if connecting via Ethernet or IEEE. Current USB driver in use has a restriction on amount of data it can return from one call.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nReceiveBufferSize</i>	: (Input) Specifies how big the the receive buffer is.
<i>pszTotalData</i>	: (Input) Pointer to location to return the Total data. <ul style="list-style-type: none"> The following data will be returned as floating point values (4 bytes) each: Total Ratio Total Ref Volt Total Sig Volt THD Frequency The following data will be returned as unsigned short values (2 bytes) each: Sample Rate Index Ref Range Sig Range The following Harmonic data will be returned (4 bytes): (Ethernet - all 16 harmonics) (USB - only fundamental) Phase Amplitude In Phase Quadrature AmplitudeRatio In PhaseRatio QuadratureRatio

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

USB_WriteMsgRawData, [ExecuteRemoteCmd](#)

3.2.2.5 `_PAV2250AFUNC` int PAV2250A_PerformGroupExecuteTrigger (int *nPAVNo*, int *nIEEEAddr*)

PAV2250A_PerformGroupExecuteTrigger performs a group execute trigger.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nIEEEAddr</i>	: (Input) IEEE Address to be used to connect to PAV2250A. (0-30)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_ADDRS : Invalid IEEE Address parameter
- PAV_ERROR_TRIGGER : Error executing the Trigger command

See Also

Trigger

3.2.2.6 `_PAV2250AFUNC` int PAV2250A_ResetDefaultValues (int *nPAVNo*)

PAV2250A_ResetDefaultValues sends the command to set the device settings back to the factory default values.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function supported only with PAV2250A_NATIVE
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[IsLanguageTypeNative](#), [ExecuteRemoteCmd](#)

3.3 Harmonic Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonics](#) (int nPAVNo, int nHarmonic, char *pszHarmonics)
PAV2250A_PerformGetHarmonics sends the Harmonics command to get Harmonics Data for the device. The Harmonics are returned in a comma separated string and includes the Harmonic Phase, Magnitude, In Phase and Quad for the given harmonic number. (valid values 0 (Fundamental) - 15)
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicPhase](#) (int nPAVNo, int nHarmonic, float *pfPhase)
PAV2250A_GetHarmonicPhase is responsible for returning just the Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicMagnitude](#) (int nPAVNo, int nHarmonic, float *pfMag)
PAV2250A_GetHarmonicMagnitude is responsible for returning just the Magnitude (Amplitude) value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicInPhase](#) (int nPAVNo, int nHarmonic, float *pfInPhase)
PAV2250A_GetHarmonicInPhase is responsible for returning just the In Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicQuad](#) (int nPAVNo, int nHarmonic, float *pfQuad)
PAV2250A_GetHarmonicQuad is responsible for returning just the Quadrature value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonicsRatio](#) (int nPAVNo, int nHarmonic, char *pszHarmonicsRatio)
PAV2250A_PerformGetHarmonicsRatio sends the Harmonics command to get Harmonics Data for the device. The Harmonics are returned in a comma separated string and includes the Harmonic Phase, Magnitude Ratio, In Phase Ratio and Quad Ratio for the given harmonic number. (valid values 0 (Fundamental) - 15)
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioPhase](#) (int nPAVNo, int nHarmonic, float *pfPhase)
PAV2250A_GetHarmonicRatioPhase is responsible for returning just the Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioMagnitude](#) (int nPAVNo, int nHarmonic, float *pfMag)
PAV2250A_GetHarmonicRatioMagnitude is responsible for returning just the Magnitude Ratio (Amplitude) value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioInPhase](#) (int nPAVNo, int nHarmonic, float *pfInPhase)
PAV2250A_GetHarmonicRatioInPhase is responsible for returning just the In Phase Ratio value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioQuad](#) (int nPAVNo, int nHarmonic, float *pfQuad)
PAV2250A_GetHarmonicRatioQuad is responsible for returning just the Quadrature Ratio value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonicDataRaw](#) (int nPAVNo, int nHarmGroup, int nReceiveBufferSize, char *pszHarmonicData)
PAV2250A_PerformGetHarmonicDataRaw sends the Harmonic Data command to get Raw Harmonic Data for the specified Harmonic Group for the device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioState](#) (int nPAVNo, bool *pbHarmRatio)
PAV2250A_GetHarmonicRatioState sends the command to get the Harmonic Ratio State to the PAV2250A device. Harmonic Ratio State controls whether Ratio or Absolute values are shown for Harmonic Display.
- [_PAV2250AFUNC](#) int [PAV2250A_SetHarmonicRatioState](#) (int nPAVNo, bool bHarmRatio)
PAV2250A_SetHarmonicRatioState sends the command to set the Harmonic Ratio State to the PAV2250A device. Harmonic Ratio State controls whether Ratio or Absolute values are shown for Harmonic Display.
- [_PAV2250AFUNC](#) int [PAV2250A_ViewPrevHarmonicGroup](#) (int nPAVNo)
PAV2250A_ViewPrevHarmonicGroup sends the command to set the Harmonic View to the previous group of harmonics.
- [_PAV2250AFUNC](#) int [PAV2250A_ViewNextHarmonicGroup](#) (int nPAVNo)
PAV2250A_ViewNextHarmonicGroup sends the command to set the Harmonic View to the next group of harmonics.
- [_PAV2250AFUNC](#) int [PAV2250A_ViewHarmonic](#) (int nPAVNo, int nHarmonic)
PAV2250A_ViewHarmonic sends the command to set the Harmonic View to show the desired harmonic. Valid values are 0 - 15.

3.3.1 Detailed Description

3.3.2 Function Documentation

3.3.2.1 `_PAV2250AFUNC int PAV2250A_GetHarmonicInPhase (int nPAVNo, int nHarmonic, float * pInPhase)`

PAV2250A_GetHarmonicInPhase is responsible for returning just the In Phase value for the provided Harmonic.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pfInPhase</i>	: (Output) In Phase floating point value for the provided Harmonic

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.3.2.2 _PAV2250AFUNC int PAV2250A_GetHarmonicMagnitude (int nPAVNo, int nHarmonic, float * pfMag)

PAV2250A_GetHarmonicMagnitude is responsible for returning just the Magnitude (Amplitude) value for the provided Harmonic.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pfMag</i>	: (Output) Magnitude floating point value for the provided Harmonic

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.3.2.3 _PAV2250AFUNC int PAV2250A_GetHarmonicPhase (int nPAVNo, int nHarmonic, float * pfPhase)

PAV2250A_GetHarmonicPhase is responsible for returning just the Phase value for the provided Harmonic.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pfPhase</i>	: (Output) Phase floating point value for the provided Harmonic

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.3.2.4 `_PAV2250AFUNC int PAV2250A_GetHarmonicQuad (int nPAVNo, int nHarmonic, float * pfQuad)`

`PAV2250A_GetHarmonicQuad` is responsible for returning just the Quadrature value for the provided Harmonic.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pfQuad</i>	: (Output) Quad floating point value for the provided Harmonic

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid `nPAVNo` parameter
- `PAV_ERROR_WRITE` : Unable to send command to 2250A
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.3.2.5 `_PAV2250AFUNC int PAV2250A_GetHarmonicRatioInPhase (int nPAVNo, int nHarmonic, float * pfInPhase)`

`PAV2250A_GetHarmonicRatioInPhase` is responsible for returning just the In Phase Ratio value for the provided Harmonic.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pfInPhase</i>	: (Output) In Phase Ratio floating point value for the provided Harmonic

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid `nPAVNo` parameter
- `PAV_ERROR_WRITE` : Unable to send command to 2250A
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.3.2.6 `_PAV2250AFUNC int PAV2250A_GetHarmonicRatioMagnitude (int nPAVNo, int nHarmonic, float * pfMag)`

`PAV2250A_GetHarmonicRatioMagnitude` is responsible for returning just the Magnitude Ratio (Amplitude) value for the provided Harmonic.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pfMag</i>	: (Output) Magnitude Ratio floating point value for the provided Harmonic

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.3.2.7 `_PAV2250AFUNC int PAV2250A_GetHarmonicRatioPhase (int nPAVNo, int nHarmonic, float * pfPhase)`

PAV2250A_GetHarmonicRatioPhase is responsible for returning just the Phase value for the provided Harmonic.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pfPhase</i>	: (Output) Phase floating point value for the provided Harmonic

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.3.2.8 `_PAV2250AFUNC int PAV2250A_GetHarmonicRatioQuad (int nPAVNo, int nHarmonic, float * pfQuad)`

PAV2250A_GetHarmonicRatioQuad is responsible for returning just the Quadrature Ratio value for the provided Harmonic.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pfQuad</i>	: (Output) Quad Ratio floating point value for the provided Harmonic

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.3.2.9 `_PAV2250AFUNC int PAV2250A_GetHarmonicRatioState (int nPAVNo, bool * pbHarmRatio)`

`PAV2250A_GetHarmonicRatioState` sends the command to get the Harmonic Ratio State to the PAV2250A device. Harmonic Ratio State controls whether Ratio or Absolute values are shown for Harmonic Display.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbHarmRatio</i>	: (Output) True if currently displaying "Ratio" values for Harmonics; False otherwise

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid `nPAVNo` parameter
- `PAV_ERROR_WRITE` : Unable to send command to 2250A
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.3.2.10 `_PAV2250AFUNC int PAV2250A_PerformGetHarmonicDataRow (int nPAVNo, int nHarmGroup, int nReceiveBufferSize, char * pszHarmonicData)`

`PAV2250A_PerformGetHarmonicDataRow` sends the Harmonic Data command to get Raw Harmonic Data for the specified Harmonic Group for the device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmGroup</i>	: (Input) Harmonic Group Number for which to retrieve data. <ul style="list-style-type: none"> • 1 = Pass back 1st and 2nd harmonic • 2 = Pass back 3rd and 4th harmonic • 3 = Pass back 5th and 6th harmonic • 4 = Pass back 7th and 8th harmonic • 5 = Pass back 9th and 10th harmonic • 6 = Pass back 11th and 12th harmonic • 7 = Pass back 13th and 14th harmonic • 8 = Pass back 15th

<i>nReceiveBuffer-Size</i>	: (Input) Size of receive buffer
<i>pszHarmonic-Data</i>	: (Output) Raw Hex values in string form separated by spaces

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

USB_WriteMsgRawData, [ExecuteRemoteCmd](#)

3.3.2.11 `_PAV2250AFUNC int PAV2250A_PerformGetHarmonics (int nPAVNo, int nHarmonic, char * pszHarmonics)`

PAV2250A_PerformGetHarmonics sends the Harmonics command to get Harmonics Data for the device. The Harmonics are returned in a comma separated string and includes the Harmonic Phase, Magnitude, In Phase and Quad for the given harmonic number. (valid values 0 (Fundamental) - 15)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pszHarmonics</i>	: (Output) Comma separated string containing: <ul style="list-style-type: none"> • Phase • Magnitude • In Phase • Quad

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.3.2.12 `_PAV2250AFUNC int PAV2250A_PerformGetHarmonicsRatio (int nPAVNo, int nHarmonic, char * pszHarmonicsRatio)`

PAV2250A_PerformGetHarmonicsRatio sends the Harmonics command to get Harmonics Data for the device. The Harmonics are returned in a comma separated string and includes the Harmonic Phase, Magnitude Ratio, In Phase Ratio and Quad Ratio for the given harmonic number. (valid values 0 (Fundamental) - 15)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number for which to retrieve data. (valid values 0 (Fundamental) - 15)
<i>pszHarmonics-Ratio</i>	: (Output) Comma separated string containing: <ul style="list-style-type: none"> • Phase • Magnitude Ratio • In Phase Ratio • Quad Ratio

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

See Also

3.3.2.13 `_PAV2250AFUNC int PAV2250A_SetHarmonicRatioState (int nPAVNo, bool bHarmRatio)`

PAV2250A_SetHarmonicRatioState sends the command to set the Harmonic Ratio State to the PAV2250A device. Harmonic Ratio State controls whether Ratio or Absolute values are shown for Harmonic Display.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>bHarmRatio</i>	: (Input) Harmonic Ratio State controls whether Ratio (True) or Absolute values (False) are shown for Harmonic Display.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.3.2.14 `_PAV2250AFUNC int PAV2250A_ViewHarmonic (int nPAVNo, int nHarmonic)`

PAV2250A_ViewHarmonic sends the command to set the Harmonic View to show the desired harmonic. Valid values are 0 - 15.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nHarmonic</i>	: (Input) Harmonic Number to view. (valid values 0 (Fundamental) - 15)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.3.2.15 `_PAV2250AFUNC int PAV2250A_ViewNextHarmonicGroup (int nPAVNo)`

PAV2250A_ViewNextHarmonicGroup sends the command to set the Harmonic View to the next group of harmonics.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.3.2.16 `_PAV2250AFUNC int PAV2250A_ViewPrevHarmonicGroup (int nPAVNo)`

PAV2250A_ViewPrevHarmonicGroup sends the command to set the Harmonic View to the previous group of harmonics.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4 Reference Range Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetRefAutoRange](#) (int nPAVNo)
PAV2250A_SetRefAutoRange sends the command to force the Reference to be in "Auto" range mode.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefAutoRange](#) (int nPAVNo, bool *pbAutoRange)
PAV2250A_GetRefAutoRange determines whether or not the Reference is in "Auto" range mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange](#) (int nPAVNo, int nRangeIndex)
PAV2250A_SetRefRange sends the command to force the Reference into the specified Range based on the Range Index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeString](#) (int nPAVNo, char *pszRefRange)
PAV2250A_GetRefRangeString is responsible for returning back the Reference range the PAV is currently operating at in string form. Return values will have the word "AUTO" precede the actual range value when the PAV is in "AUTO" range mode and will have the word "OVR" precede the actual range when the actual range is over the configured range. Example: Auto Range with an actual range of 2 Volts will return: "AUTO 2.000". If the range is not "AUTO" and an Over range was not detected, the configured range will be returned such as: "2.000" for the 2 volt range.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeIndexSettings](#) (int nPAVNo, bool *pbAutoRange, bool *pbRangeMismatch, int *pnRangeIndex)
PAV2250A_GetRefRangeIndexSettings is responsible for returning back Reference range information. Return values indicate whether or not the reference is in "Auto" range mode, whether or not there is a range mismatch (i.e. the actual range is different than the desired configured range), and the actual range index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeConfigIndex](#) (int nPAVNo, int *pnRangeIndex)
PAV2250A_GetRefRangeConfigIndex is responsible for returning the index of the Reference range the PAV was configured with.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeActualIndex](#) (int nPAVNo, int *pnRangeIndex)
PAV2250A_GetRefRangeConfigIndex is responsible for returning the index of the Reference range the PAV was actually operating in.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange50MV](#) (int nPAVNo)
PAV2250A_SetRefRange50MV sends the command to force the PAV Reference to be 50MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange100MV](#) (int nPAVNo)
PAV2250A_SetRefRange100MV sends the command to force the PAV Reference to be 100MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange200MV](#) (int nPAVNo)
PAV2250A_SetRefRange200MV sends the command to force the PAV Reference to be 200MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange500MV](#) (int nPAVNo)
PAV2250A_SetRefRange500MV sends the command to force the PAV Reference to be 500MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange1V](#) (int nPAVNo)
PAV2250A_SetRefRange1V sends the command to force the PAV Reference to be 1V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange2V](#) (int nPAVNo)
PAV2250A_SetRefRange2V sends the command to force the PAV Reference to be 2V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange5V](#) (int nPAVNo)
PAV2250A_SetRefRange5V sends the command to force the PAV Reference to be 5V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange10V](#) (int nPAVNo)
PAV2250A_SetRefRange10V sends the command to force the PAV Reference to be 10V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange20V](#) (int nPAVNo)
PAV2250A_SetRefRange20V sends the command to force the PAV Reference to be 20V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange50V](#) (int nPAVNo)
PAV2250A_SetRefRange50V sends the command to force the PAV Reference to be 50V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange100V](#) (int nPAVNo)
PAV2250A_SetRefRange100V sends the command to force the PAV Reference to be 100V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange200V](#) (int nPAVNo)
PAV2250A_SetRefRange200V sends the command to force the PAV Reference to be 200V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange500V](#) (int nPAVNo)
PAV2250A_SetRefRange500V sends the command to force the PAV Reference to be 500V.

3.4.1 Detailed Description

3.4.2 Function Documentation

3.4.2.1 `_PAV2250AFUNC int PAV2250A_GetRefAutoRange (int nPAVNo, bool * pbAutoRange)`

PAV2250A_GetRefAutoRange determines whether or not the Reference is in "Auto" range mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbAutoRange</i>	: (Output) Will reflect the value True if Reference is in "Auto" mode; otherwise False

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdBool](#)

3.4.2.2 `_PAV2250AFUNC int PAV2250A_GetRefRangeActualIndex (int nPAVNo, int * pnRangeIndex)`

PAV2250A_GetRefRangeConfigIndex is responsible for returning the index of the Reference range the PAV was is actually operating in.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnRangeIndex</i>	: (Output) Range Index (0 - 12) <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.4.2.3 _PAV2250AFUNC int PAV2250A_GetRefRangeConfigIndex (int nPAVNo, int * pnRangeIndex)

PAV2250A_GetRefRangeConfigIndex is responsible for returning the index of the Reference range the PAV was configured with.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnRangeIndex</i>	: (Output) Range Index (0 - 12) or 65535 for "AUTO" <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.4.2.4 **_PAV2250AFUNC** int PAV2250A_GetRefRangeIndexSettings (int *nPAVNo*, bool * *pbAutoRange*, bool * *pbRangeMismatch*, int * *pnRangeIndex*)

PAV2250A_GetRefRangeIndexSettings is responsible for returning back Reference range information. Return values indicate whether or not the reference is in "Auto" range mode, whether or not there is a range mismatch (i.e. the actual range is different than the desired configured range), and the actual range index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbAutoRange</i>	: (Output) True if the PAV is configured for AUTO ranging
<i>pbRange-Mismatch</i>	: (Output) True if there is a range mismatch between the actual range and desired configured range; False otherwise
<i>pnRangeIndex</i>	: (Output) The actual range index the reference is operating in <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[GetRange](#)

3.4.2.5 `_PAV2250AFUNC int PAV2250A_GetRefRangeString (int nPAVNo, char * pszRefRange)`

PAV2250A_GetRefRangeString is responsible for returning back the Reference range the PAV is currently operating at in string form. Return values will have the word "AUTO" precede the actual range value when the PAV is in "AUTO" range mode and will have the word "OVR" precede the actual range when the actual range is over the configured range. Example: Auto Range with an actual range of 2 Volts will return: "AUTO 2.000". If the range is not "AUTO" and an Over range was not detected, the configured range will be returned such as: "2.000" for the 2 volt range.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

<i>pszRefRange</i>	: (Output) Current Reference range the PAV is operating in. Examples of 2V range include: <ul style="list-style-type: none"> • AUTO 2.000 • OVR 2.000 • 2.000
--------------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.4.2.6 `_PAV2250AFUNC int PAV2250A_SetRefAutoRange (int nPAVNo)`

PAV2250A_SetRefAutoRange sends the command to force the Reference to be in "Auto" range mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)3.4.2.7 `_PAV2250AFUNC` int `PAV2250A_SetRefRange` (int `nPAVNo`, int `nRangeIndex`)

`PAV2250A_SetRefRange` sends the command to force the Reference into the specified Range based on the Range Index.

Parameters

<code>nPAVNo</code>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<code>nRangeIndex</code>	: (Input) Zero-based index of Range to put PAV Reference into. <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid `nPAVNo` parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)3.4.2.8 `_PAV2250AFUNC` int `PAV2250A_SetRefRange100MV` (int `nPAVNo`)

`PAV2250A_SetRefRange100MV` sends the command to force the PAV Reference to be 100MV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.9 _PAV2250AFUNC int PAV2250A_SetRefRange100V (int nPAVNo)

PAV2250A_SetRefRange100V sends the command to force the PAV Reference to be 100V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.10 _PAV2250AFUNC int PAV2250A_SetRefRange10V (int nPAVNo)

PAV2250A_SetRefRange10V sends the command to force the PAV Reference to be 10V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.11 _PAV2250AFUNC int PAV2250A_SetRefRange1V (int nPAVNo)

PAV2250A_SetRefRange1V sends the command to force the PAV Reference to be 1V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.12 `_PAV2250AFUNC int PAV2250A_SetRefRange200MV (int nPAVNo)`

PAV2250A_SetRefRange200MV sends the command to force the PAV Reference to be 200MV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.13 `_PAV2250AFUNC int PAV2250A_SetRefRange200V (int nPAVNo)`

PAV2250A_SetRefRange200V sends the command to force the PAV Reference to be 200V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.14 `_PAV2250AFUNC int PAV2250A_SetRefRange20V (int nPAVNo)`

PAV2250A_SetRefRange20V sends the command to force the PAV Reference to be 20V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.15 __PAV2250AFUNC int PAV2250A_SetRefRange2V (int nPAVNo)

PAV2250A_SetRefRange2V sends the command to force the PAV Reference to be 2V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.16 __PAV2250AFUNC int PAV2250A_SetRefRange500MV (int nPAVNo)

PAV2250A_SetRefRange500MV sends the command to force the PAV Reference to be 500MV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.17 __PAV2250AFUNC int PAV2250A_SetRefRange500V (int nPAVNo)

PAV2250A_SetRefRange500V sends the command to force the PAV Reference to be 500V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.18 `_PAV2250AFUNC int PAV2250A_SetRefRange50MV (int nPAVNo)`

PAV2250A_SetRefRange50MV sends the command to force the PAV Reference to be 50MV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.19 `_PAV2250AFUNC int PAV2250A_SetRefRange50V (int nPAVNo)`

PAV2250A_SetRefRange50V sends the command to force the PAV Reference to be 50V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.4.2.20 `_PAV2250AFUNC int PAV2250A_SetRefRange5V (int nPAVNo)`

PAV2250A_SetRefRange5V sends the command to force the PAV Reference to be 5V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5 Signal Range Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetSigAutoRange](#) (int nPAVNo)
PAV2250A_SetSigAutoRange sends the command to force the Signal to be in "Auto" range mode.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigAutoRange](#) (int nPAVNo, bool *pbAutoRange)
PAV2250A_GetSigAutoRange determines whether or not the Signal is in "Auto" range mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange](#) (int nPAVNo, int nRangeIndex)
PAV2250A_SetSigRange sends the command to force the Signal into the specified Range based on the Range Index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeString](#) (int nPAVNo, char *pszSigRange)
PAV2250A_GetSigRangeString is responsible for returning back the Signal range the PAV is currently operating at in string form. Return values will have the word "AUTO" precede the actual range value when the PAV is in "AUTO" range mode and will have the word "OVR" precede the actual range when the actual range is over the configured range. Example: Auto Range with an actual range of 2 Volots will return: "AUTO 2.000". If the range is not "AUTO" and an Over range was not detected, the configured range will be returned such as: "2.000" for the 2 volt range.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeIndexSettings](#) (int nPAVNo, bool *pbAutoRange, bool *pbRangeMismatch, int *pnRangeIndex)
PAV2250A_GetSigRangeIndexSettings is responsible for returning back Signal range information. Return values indicate whether or not the signal is in "Auto" range mode, whether or not there is a range mismatch (i.e. the actual range is different than the desired configured range), and the actual range index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeConfigIndex](#) (int nPAVNo, int *pnRangeIndex)
PAV2250A_GetSigRangeConfigIndex is responsible for returning the index of the Signal range the PAV was configured with.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeActualIndex](#) (int nPAVNo, int *pnRangeIndex)
PAV2250A_GetSigRangeConfigIndex is responsible for returning the index of the Signal range the PAV was actually operating in.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange50MV](#) (int nPAVNo)
PAV2250A_SetSigRange50MV sends the command to force the PAV Signal to be 50MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange100MV](#) (int nPAVNo)
PAV2250A_SetSigRange100MV sends the command to force the PAV Signal to be 100MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange200MV](#) (int nPAVNo)
PAV2250A_SetSigRange200MV sends the command to force the PAV Signal to be 200MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange500MV](#) (int nPAVNo)
PAV2250A_SetSigRange500MV sends the command to force the PAV Signal to be 500MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange1V](#) (int nPAVNo)
PAV2250A_SetSigRange1V sends the command to force the PAV Signal to be 1V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange2V](#) (int nPAVNo)
PAV2250A_SetSigRange2V sends the command to force the PAV Signal to be 2V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange5V](#) (int nPAVNo)
PAV2250A_SetSigRange5V sends the command to force the PAV Signal to be 5V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange10V](#) (int nPAVNo)
PAV2250A_SetSigRange10V sends the command to force the PAV Signal to be 10V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange20V](#) (int nPAVNo)
PAV2250A_SetSigRange20V sends the command to force the PAV Signal to be 20V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange50V](#) (int nPAVNo)
PAV2250A_SetSigRange50V sends the command to force the PAV Signal to be 50V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange100V](#) (int nPAVNo)
PAV2250A_SetSigRange100V sends the command to force the PAV Signal to be 100V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange200V](#) (int nPAVNo)
PAV2250A_SetSigRange200V sends the command to force the PAV Signal to be 200V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange500V](#) (int nPAVNo)
PAV2250A_SetSigRange500V sends the command to force the PAV Signal to be 500V.

3.5.1 Detailed Description

3.5.2 Function Documentation

3.5.2.1 `_PAV2250AFUNC int PAV2250A_GetSigAutoRange (int nPAVNo, bool * pbAutoRange)`

PAV2250A_GetSigAutoRange determines whether or not the Signal is in "Auto" range mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbAutoRange</i>	: (Output) Will reflect the value true if Signal is in "Auto" mode; otherwise false

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdBool](#)

3.5.2.2 `_PAV2250AFUNC int PAV2250A_GetSigRangeActualIndex (int nPAVNo, int * pnRangeIndex)`

PAV2250A_GetSigRangeConfigIndex is responsible for returning the index of the Signal range the PAV was is actually operating in.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnRangeIndex</i>	: (Output) Range Index (0 - 12) <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.5.2.3 _PAV2250AFUNC int PAV2250A_GetSigRangeConfigIndex (int nPAVNo, int * pnRangeIndex)

PAV2250A_GetSigRangeConfigIndex is responsible for returning the index of the Signal range the PAV was configured with.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnRangeIndex</i>	: (Output) Range Index (0 - 12) or 65535 for "AUTO" <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.5.2.4 **_PAV2250AFUNC** int PAV2250A_GetSigRangeIndexSettings (int *nPAVNo*, bool * *pbAutoRange*, bool * *pbRangeMismatch*, int * *pnRangeIndex*)

PAV2250A_GetSigRangeIndexSettings is responsible for returning back Signal range information. Return values indicate whether or not the signal is in "Auto" range mode, whether or not there is a range mismatch (i.e. the actual range is different than the desired configured range), and the actual range index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbAutoRange</i>	: (Output) True if the PAV is configured for AUTO ranging
<i>pbRange-Mismatch</i>	: (Output) True if there is a range mismatch between the actual range and desired configured range; False otherwise
<i>pnRangeIndex</i>	: (Output) The actual range index the signal is operating in <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[GetRange](#)

3.5.2.5 `_PAV2250AFUNC int PAV2250A_GetSigRangeString (int nPAVNo, char * pszSigRange)`

PAV2250A_GetSigRangeString is responsible for returning back the Signal range the PAV is currently operating at in string form. Return values will have the word "AUTO" precede the actual range value when the PAV is in "AUTO" range mode and will have the word "OVR" precede the actual range when the actual range is over the configured range. Example: Auto Range with an actual range of 2 Volts will return: "AUTO 2.000". If the range is not "AUTO" and an Over range was not detected, the configured range will be returned such as: "2.000" for the 2 volt range.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszSigRange</i>	: (Output) Current Signal range the PAV is operating in. Examples of 2V range include: <ul style="list-style-type: none"> • AUTO 2.000 • OVR 2.000 • 2.000

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.5.2.6 `_PAV2250AFUNC int PAV2250A_SetSigAutoRange (int nPAVNo)`

PAV2250A_SetSigAutoRange sends the command to force the Signal to be in "Auto" range mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.7 `_PAV2250AFUNC int PAV2250A_SetSigRange (int nPAVNo, int nRangeIndex)`

PAV2250A_SetSigRange sends the command to force the Signal into the specified Range based on the Range Index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nRangeIndex</i>	: (Input) Zero-based index of Range to put PAV Signal into. <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.8 _PAV2250AFUNC int PAV2250A_SetSigRange100MV (int nPAVNo)

PAV2250A_SetSigRange100MV sends the command to force the PAV Signal to be 100MV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.9 _PAV2250AFUNC int PAV2250A_SetSigRange100V (int nPAVNo)

PAV2250A_SetSigRange100V sends the command to force the PAV Signal to be 100V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.10 _PAV2250AFUNC int PAV2250A_SetSigRange10V (int nPAVNo)

PAV2250A_SetSigRange10V sends the command to force the PAV Signal to be 10V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.11 `_PAV2250AFUNC int PAV2250A_SetSigRange1V (int nPAVNo)`

PAV2250A_SetSigRange1V sends the command to force the PAV Signal to be 1V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.12 `_PAV2250AFUNC int PAV2250A_SetSigRange200MV (int nPAVNo)`

PAV2250A_SetSigRange200MV sends the command to force the PAV Signal to be 200MV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.13 `_PAV2250AFUNC int PAV2250A_SetSigRange200V (int nPAVNo)`

PAV2250A_SetSigRange200V sends the command to force the PAV Signal to be 200V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.14 `_PAV2250AFUNC int PAV2250A_SetSigRange20V (int nPAVNo)`

PAV2250A_SetSigRange20V sends the command to force the PAV Signal to be 20V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.15 `_PAV2250AFUNC int PAV2250A_SetSigRange2V (int nPAVNo)`

PAV2250A_SetSigRange2V sends the command to force the PAV Signal to be 2V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.16 `_PAV2250AFUNC int PAV2250A_SetSigRange500MV (int nPAVNo)`

PAV2250A_SetSigRange500MV sends the command to force the PAV Signal to be 500MV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.17 `_PAV2250AFUNC int PAV2250A_SetSigRange500V (int nPAVNo)`

PAV2250A_SetSigRange500V sends the command to force the PAV Signal to be 500V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.18 `_PAV2250AFUNC int PAV2250A_SetSigRange50MV (int nPAVNo)`

PAV2250A_SetSigRange50MV sends the command to force the PAV Signal to be 50MV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.19 `_PAV2250AFUNC int PAV2250A_SetSigRange50V (int nPAVNo)`

PAV2250A_SetSigRange50V sends the command to force the PAV Signal to be 50V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.5.2.20 `_PAV2250AFUNC int PAV2250A_SetSigRange5V (int nPAVNo)`

PAV2250A_SetSigRange5V sends the command to force the PAV Signal to be 5V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.6 IEEE Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_IEEEReset](#) (int nPAVNo, char *pszResults)
PAV2250A_IEEEReset sends the command to reset the 2250A device and set the device setting back to the factory default settings. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_IEEEGetErrors](#) (int nPAVNo, char *pszErrors)
PAV2250A_IEEEGetErrors sends the ERR command to get error from the error queue for the device. No error is returned when there are no errors on the queue.
- [_PAV2250AFUNC](#) int [PAV2250A_IEEECLS](#) (int nPAVNo)
PAV2250A_IEEECLS is responsible for clearing the IEEE - Clears Event Status Registers and Error Message Queue.

3.6.1 Detailed Description

3.6.2 Function Documentation

3.6.2.1 [_PAV2250AFUNC](#) int [PAV2250A_IEEECLS](#) (int nPAVNo)

[PAV2250A_IEEECLS](#) is responsible for clearing the IEEE - Clears Event Status Registers and Error Message Queue.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.6.2.2 [_PAV2250AFUNC](#) int [PAV2250A_IEEEGetErrors](#) (int nPAVNo, char * pszErrors)

[PAV2250A_IEEEGetErrors](#) sends the ERR command to get error from the error queue for the device. No error is returned when there are no errors on the queue.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszErrors</i>	: (Output) Pointer to location to return the error string

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.6.2.3 `_PAV2250AFUNC` int PAV2250A_IEEEReset (int *nPAVNo*, char * *pszResults*)

PAV2250A_IEEEReset sends the command to reset the 2250A device and set the device setting back to the factory default settings. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszResults</i>	: (Output) pointer to location to return the results of the reset command. Return results: <ul style="list-style-type: none">• Reset Complete : Reset has been successful.• Reset Not Performed : Reset has not been successful.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid *nPAVNo* parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7 Configuration Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetIEEELang](#) (int nPAVNo, int *pnIEEELang)
PAV2250A_GetIEEELang sends the command to get the IEEE language protocol set in the 2250A.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIEEELangText](#) (int nPAVNo, char *pszIEEELang)
PAV2250A_GetIEEELangText sends the command to get the IEEE language protocol set in the 2250A.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang](#) (int nPAVNo, int nIEEELang)
PAV2250A_SetIEEELang sends the command to set the IEEE protocol language to accept when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang2250ANative](#) (int nPAVNo)
PAV2250A_SetIEEELang2250ANative sends the command to set the IEEE protocol to the 2250A Native language when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang2250Legacy](#) (int nPAVNo)
PAV2250A_SetIEEELang2250Legacy sends the command to set the IEEE protocol to the 2250A Native language when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_GetCommState](#) (int nPAVNo, char *pszCommState)
PAV2250A_GetCommState sends the command to get the communication mode set in the 2250A.
- [_PAV2250AFUNC](#) int [PAV2250A_GoToLocal](#) (int nPAVNo)
PAV2250A_GoToLocal sends the command to set the communication mode to Local mode. In Local mode, remote set commands will not be accepted.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteUSB](#) (int nPAVNo)
PAV2250A_SetRemoteUSB sends the command to set the communication mode to Remote USB mode. In Remote USB mode, remote set commands will be accepted if the command is received from the USB interface.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteEthernet](#) (int nPAVNo)
PAV2250A_SetRemoteEthernet sends the command to set the communication mode to Remote Ethernet mode. In Remote Ethernet mode, remote set commands will be accepted if the command is received from the Ethernet interface.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemotIEEE](#) (int nPAVNo)
PAV2250A_SetRemotIEEE sends the command to set the communication mode to Remote IEEE mode. In Remote IEEE mode, remote set commands will be accepted if the command is received from the IEEE interface.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteJ1](#) (int nPAVNo)
PAV2250A_SetRemoteJ1 sends the command to set the communication mode to Remote J1 mode. In Remote J1 mode, remote set commands will be accepted if the command is received from the J1 interface.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTabView](#) (int nPAVNo, int *pnTabIndex)
PAV2250A_GetTabView sends the command requesting the current Tab View Index to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTabViewText](#) (int nPAVNo, char *pszTabViewText)
PAV2250A_GetTabViewText sends the command requesting the current Tab View text to the PAV2250A device. Text is reflective of the currently selected tab label. (Main, Harmonics, Quad View)
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabView](#) (int nPAVNo, int nTabIndex)
PAV2250A_SetTabView sends the command to set the Tab View Index to the PAV2250A device. Tab View Index should be reflective of which tab the PAV should display as the active tab. Index is zero-based.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewMain](#) (int nPAVNo)
PAV2250A_SetTabViewMain sends the command to set the Tab View to the Main tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewReference](#) (int nPAVNo)
PAV2250A_SetTabViewReference sends the command to set the Tab View to the Reference tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewHarmonics](#) (int nPAVNo)
PAV2250A_SetTabViewHarmonics sends the command to set the Tab View to the Harmonics tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewCustom](#) (int nPAVNo)
PAV2250A_SetTabViewCustom sends the command to set the Tab View to the Custom View tab.

- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewLVDT](#) (int nPAVNo)
PAV2250A_SetTabViewLVDT sends the command to set the Tab View to the LVDT View tab.
- [_PAV2250AFUNC](#) int [PAV2250A_GetReadMode](#) (int nPAVNo, int *pnReadModeIndex)
PAV2250A_GetReadMode sends the command requesting the current Read Mode Index to the PAV2250A device. Read Mode index indicates whether the PAV is showing SIG/REF, REF, SIG or REF/SIG. Index is zero-based.
- [_PAV2250AFUNC](#) int [PAV2250A_GetReadModeText](#) (int nPAVNo, char *pszReadModeText)
PAV2250A_GetReadModeText sends the command requesting the current Read Mode text to the PAV2250A device. Text is reflective of the currently selected Reading Mode (SIG/REF, REF, SIG, or REF/SIG).
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadMode](#) (int nPAVNo, int nReadModeIndex)
PAV2250A_SetReadMode sends the command to set the Read Mode Index to the PAV2250A device. Read Mode index indicates whether the PAV is showing SIG/REF, REF, SIG or REF/SIG. Index is zero-based.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeSigRef](#) (int nPAVNo)
PAV2250A_SetReadModeSigRef sends the command to set the Read Mode to Sig/Ref.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeRefRef](#) (int nPAVNo)
PAV2250A_SetReadModeRefRef sends the command to set the Read Mode to Ref.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeSigSig](#) (int nPAVNo)
PAV2250A_SetReadModeSigSig sends the command to set the Read Mode to Sig.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeRefSig](#) (int nPAVNo)
PAV2250A_SetReadModeRefSig sends the command to set the Read Mode to Ref/Sig.
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainView](#) (int nPAVNo, int *pnMainViewIndex)
PAV2250A_GetMainView sends the command requesting the current Main View Index to the PAV2250A device. Main View index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainViewText](#) (int nPAVNo, char *pszMainViewText)
PAV2250A_GetMainViewText sends the command requesting the current Main View text to the PAV2250A device. Text is reflective of the currently selected Main View: (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainView](#) (int nPAVNo, int nMainViewIndex)
PAV2250A_SetMainView sends the command to set the Main View Index to the PAV2250A device. Main View index indicates the view currently shown in the main display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewFundMag](#) (int nPAVNo)
PAV2250A_SetMainViewFundMag sends the command to set the Main View to Fundamental Magnitude.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewInPhase](#) (int nPAVNo)
PAV2250A_SetMainViewInPhase sends the command to set the Main View to In Phase.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewQuad](#) (int nPAVNo)
PAV2250A_SetMainViewQuad sends the command to set the Main View to Quadrature.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewPhase](#) (int nPAVNo)
PAV2250A_SetMainViewPhase sends the command to set the Main View to Phase.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewTHD](#) (int nPAVNo)
PAV2250A_SetMainViewTHD sends the command to set the Main View to THD.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewSigVolt](#) (int nPAVNo)
PAV2250A_SetMainViewSigVolt sends the command to set the Main View to Signal Voltage.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewRefVolt](#) (int nPAVNo)
PAV2250A_SetMainViewRefVolt sends the command to set the Main View to Reference Voltage.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHoldDataState](#) (int nPAVNo, bool *pbHoldData)
PAV2250A_GetHoldDataState sends the command to get the Hold Data State to the PAV2250A device. Hold Data when true indicates no screen refreshes are taking place.
- [_PAV2250AFUNC](#) int [PAV2250A_SetHoldDataState](#) (int nPAVNo, bool bHoldData)
PAV2250A_SetHoldDataState sends the command to set the Hold Data State to the PAV2250A device. Hold Data when true indicates no screen refreshes are taking place.
- [_PAV2250AFUNC](#) int [PAV2250A_GetScreenBrightness](#) (int nPAVNo, int *pnBrightness)

- PAV2250A_GetScreenBrightness* sends the command requesting the current Screen Brightness to the PAV2250A device. Screen Brightness indicates how bright the screen is illuminated. Valid values are between 15-100.
- [_PAV2250AFUNC](#) int [PAV2250A_SetScreenBrightness](#) (int nPAVNo, int nBrightness)

PAV2250A_SetScreenBrightness sends the command setting the current Screen Brightness on the PAV2250A device. Screen Brightness indicates how bright the screen is illuminated. Valid values are between 15-100.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView1](#) (int nPAVNo, int *pnCustView1Index)

PAV2250A_GetCustView1 sends the command requesting the current Quad View 1 Index to the PAV2250A device. Quad View 1 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView1Text](#) (int nPAVNo, char *pszCustView1Text)

PAV2250A_GetCustView1Text sends the command requesting the current Quad View 1 Text to the PAV2250A device. Quad View 1 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1](#) (int nPAVNo, int nCustView1Index)

PAV2250A_SetCustView1 sends the command to set the Quad View 1 Index to the PAV2250A device. Quad View 1 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1FundMag](#) (int nPAVNo)

PAV2250A_SetCustView1FundMag sends the command to set the Quad View 1 slot to Fundamental Magnitude.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1InPhase](#) (int nPAVNo)

PAV2250A_SetCustView1InPhase sends the command to set the Quad View 1 slot to In Phase.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1Quad](#) (int nPAVNo)

PAV2250A_SetCustView1Quad sends the command to set the Quad View 1 slot to Quadrature.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1Phase](#) (int nPAVNo)

PAV2250A_SetCustView1Phase sends the command to set the Quad View 1 slot to Phase Angle.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1THD](#) (int nPAVNo)

PAV2250A_SetCustView1THD sends the command to set the Quad View 1 slot to THD.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1SigVolt](#) (int nPAVNo)

PAV2250A_SetCustView1SigVolt sends the command to set the Quad View 1 slot to Signal Voltage.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1RefVolt](#) (int nPAVNo)

PAV2250A_SetCustView1RefVolt sends the command to set the Quad View 1 slot to Reference Voltage.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView2](#) (int nPAVNo, int *pnCustView2Index)

PAV2250A_GetCustView2 sends the command requesting the current Quad View 2 Index to the PAV2250A device. Quad View 2 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView2Text](#) (int nPAVNo, char *pszCustView2Text)

PAV2250A_GetCustView2Text sends the command requesting the current Quad View 2 Text to the PAV2250A device. Quad View 2 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2](#) (int nPAVNo, int nCustView2Index)

PAV2250A_SetCustView2 sends the command to set the Quad View 2 Index to the PAV2250A device. Quad View 2 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2FundMag](#) (int nPAVNo)

PAV2250A_SetCustView2FundMag sends the command to set the Quad View 2 slot to Fundamental Magnitude.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2InPhase](#) (int nPAVNo)

PAV2250A_SetCustView2InPhase sends the command to set the Quad View 2 slot to In Phase.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2Quad](#) (int nPAVNo)

PAV2250A_SetCustView2Quad sends the command to set the Quad View 2 slot to Quadrature.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2Phase](#) (int nPAVNo)

PAV2250A_SetCustView2Phase sends the command to set the Quad View 2 slot to Phase Angle.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2THD](#) (int nPAVNo)

PAV2250A_SetCustView2THD sends the command to set the Quad View 2 slot to THD.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2SigVolt](#) (int nPAVNo)

- PAV2250A_SetCustView2SigVolt* sends the command to set the Quad View 2 slot to Signal Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2RefVolt](#) (int nPAVNo)

PAV2250A_SetCustView2RefVolt sends the command to set the Quad View 2 slot to Reference Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView3](#) (int nPAVNo, int *pnCustView3Index)

PAV2250A_GetCustView3 sends the command requesting the current Quad View 3 Index to the PAV2250A device. Quad View 3 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView3Text](#) (int nPAVNo, char *pszCustView3Text)

PAV2250A_GetCustView3Text sends the command requesting the current Quad View 3 Text to the PAV2250A device. Quad View 3 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3](#) (int nPAVNo, int nCustView3Index)

PAV2250A_SetCustView3 sends the command to set the Quad View 3 Index to the PAV2250A device. Quad View 3 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3FundMag](#) (int nPAVNo)

PAV2250A_SetCustView3FundMag sends the command to set the Quad View 3 slot to Fundamental Magnitude.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3InPhase](#) (int nPAVNo)

PAV2250A_SetCustView3InPhase sends the command to set the Quad View 3 slot to In Phase.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3Quad](#) (int nPAVNo)

PAV2250A_SetCustView3Quad sends the command to set the Quad View 3 slot to Quadrature.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3Phase](#) (int nPAVNo)

PAV2250A_SetCustView3Phase sends the command to set the Quad View 3 slot to Phase Angle.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3THD](#) (int nPAVNo)

PAV2250A_SetCustView3THD sends the command to set the Quad View 3 slot to THD.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3SigVolt](#) (int nPAVNo)

PAV2250A_SetCustView3SigVolt sends the command to set the Quad View 3 slot to Signal Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3RefVolt](#) (int nPAVNo)

PAV2250A_SetCustView3RefVolt sends the command to set the Quad View 3 slot to Reference Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView4](#) (int nPAVNo, int *pnCustView4Index)

PAV2250A_GetCustView4 sends the command requesting the current Quad View 4 Index to the PAV2250A device. Quad View 4 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView4Text](#) (int nPAVNo, char *pszCustView4Text)

PAV2250A_GetCustView4Text sends the command requesting the current Quad View 4 Text to the PAV2250A device. Quad View 4 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4](#) (int nPAVNo, int nCustView4Index)

PAV2250A_SetCustView4 sends the command to set the Quad View 4 Index to the PAV2250A device. Quad View 4 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4FundMag](#) (int nPAVNo)

PAV2250A_SetCustView4FundMag sends the command to set the Quad View 4 slot to Fundamental Magnitude.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4InPhase](#) (int nPAVNo)

PAV2250A_SetCustView4InPhase sends the command to set the Quad View 4 slot to In Phase.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4Quad](#) (int nPAVNo)

PAV2250A_SetCustView4Quad sends the command to set the Quad View 4 slot to Quadrature.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4Phase](#) (int nPAVNo)

PAV2250A_SetCustView4Phase sends the command to set the Quad View 4 slot to Phase Angle.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4THD](#) (int nPAVNo)

PAV2250A_SetCustView4THD sends the command to set the Quad View 4 slot to THD.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4SigVolt](#) (int nPAVNo)

PAV2250A_SetCustView4SigVolt sends the command to set the Quad View 4 slot to Signal Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4RefVolt](#) (int nPAVNo)

PAV2250A_SetCustView4RefVolt sends the command to set the Quad View 4 slot to Reference Voltage.

3.7.1 Detailed Description

3.7.2 Function Documentation

3.7.2.1 `_PAV2250AFUNC int PAV2250A_GetCommState (int nPAVNo, char * pszCommState)`

PAV2250A_GetCommState sends the command to get the communication mode set in the 2250A.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszCommState</i>	: (Input) pointer to location to return the communication mode string. Return values: <ul style="list-style-type: none"> • "Local Mode" • "Remote IEEE Addr: <addr>, Lang: <language>" • "Remote USB" • "Remote Ethernet" • "Remote J1"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7.2.2 `_PAV2250AFUNC int PAV2250A_GetCustView1 (int nPAVNo, int * pnCustView1Index)`

PAV2250A_GetCustView1 sends the command requesting the current Quad View 1 Index to the PAV2250A device. Quad View 1 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase,2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnCustView1-Index</i>	: (Output) Index of view currently being displayed for this view <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.7.2.3 **_PAV2250AFUNC** int PAV2250A_GetCustView1Text (int nPAVNo, char * pszCustView1Text)

PAV2250A_GetCustView1Text sends the command requesting the current Quad View 1 Text to the PAV2250A device. Quad View 1 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszCustView1-Text</i>	: (Output) Text name of view currently being displayed for this view <ul style="list-style-type: none"> • Fund Mag • In Phase • Quad • Phase • THD • Sig Volt • Ref Volt

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7.2.4 **_PAV2250AFUNC** int PAV2250A_GetCustView2 (int nPAVNo, int * pnCustView2Index)

PAV2250A_GetCustView2 sends the command requesting the current Quad View 2 Index to the PAV2250A device. Quad View 2 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase,2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnCustView2-Index</i>	: (Output) Index of view currently being displayed for this view <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.7.2.5 _PAV2250AFUNC int PAV2250A_GetCustView2Text (int nPAVNo, char * pszCustView2Text)

PAV2250A_GetCustView2Text sends the command requesting the current Quad View 2 Text to the PAV2250A device. Quad View 2 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszCustView2-Text</i>	: (Output) Text name of view currently being displayed for this view <ul style="list-style-type: none"> • Fund Mag • In Phase • Quad • Phase • THD • Sig Volt • Ref Volt

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7.2.6 `_PAV2250AFUNC int PAV2250A_GetCustView3 (int nPAVNo, int * pnCustView3Index)`

PAV2250A_GetQuadVies3 sends the command requesting the current Quad View 3 Index to the PAV2250A device. Quad View 3 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase,2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnCustView3-Index</i>	: (Output) Index of view currently being displayed for this view <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.7.2.7 `_PAV2250AFUNC int PAV2250A_GetCustView3Text (int nPAVNo, char * pszCustView3Text)`

PAV2250A_GetCustView3Text sends the command requesting the current Quad View 3 Text to the PAV2250A device. Quad View 3 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

<i>pszCustView3-Text</i>	: (Output) Text name of view currently being displayed for this view <ul style="list-style-type: none"> • Fund Mag • In Phase • Quad • Phase • THD • Sig Volt • Ref Volt
--------------------------	---

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7.2.8 **_PAV2250AFUNC** int PAV2250A_GetCustView4 (int nPAVNo, int * pnCustView4Index)

PAV2250A_GetCustView4 sends the command requesting the current Quad View 4 Index to the PAV2250A device. Quad View 4 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnCustView4-Index</i>	: (Output) Index of view currently being displayed for this view <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.7.2.9 `_PAV2250AFUNC int PAV2250A_GetCustView4Text (int nPAVNo, char * pszCustView4Text)`

PAV2250A_GetCustView4Text sends the command requesting the current Quad View 4 Text to the PAV2250A device. Quad View 4 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszCustView4-Text</i>	: (Output) Text name of view currently being displayed for this view <ul style="list-style-type: none"> • Fund Mag • In Phase • Quad • Phase • THD • Sig Volt • Ref Volt

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7.2.10 `_PAV2250AFUNC int PAV2250A_GetHoldDataState (int nPAVNo, bool * pbHoldData)`

PAV2250A_GetHoldDataState sends the command to get the Hold Data State to the PAV2250A device. Hold Data when true indicates no screen refreshes are taking place.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbHoldData</i>	: (Output) True if PAV is currently in the "Hold" state. False otherwise

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.11 `_PAV2250AFUNC int PAV2250A_GetIEEELang (int nPAVNo, int * pnIEEELang)`

PAV2250A_GetIEEELang sends the command to get the IEEE language protocol set in the 2250A.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnIEEELang</i>	: (Output) IEEE protocol index. <ul style="list-style-type: none"> • 0 : PAV-2250A Native • 1 : PAV-2250 Legacy

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function supported only with PAV2250A_NATIVE
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.7.2.12 `_PAV2250AFUNC int PAV2250A_GetIEEELangText (int nPAVNo, char * pszIEEELang)`

PAV2250A_GetIEEELangText sends the command to get the IEEE language protocol set in the 2250A.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszIEEELang</i>	: (Output) IEEE protocol string. <ul style="list-style-type: none"> • 2250ANATIVE: PAV-2250A Native • 2250LEGACY : PAV-2250 Legacy

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function supported only with PAV2250A_NATIVE
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.13 `_PAV2250AFUNC int PAV2250A_GetMainView (int nPAVNo, int * pnMainViewIndex)`

PAV2250A_GetMainView sends the command requesting the current Main View Index to the PAV2250A device. Main View index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMainView-Index</i>	: (Output) Main View Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.7.2.14 `_PAV2250AFUNC int PAV2250A_GetMainViewText (int nPAVNo, char * pszMainViewText)`

PAV2250A_GetMainViewText sends the command requesting the current Main View text to the PAV2250A device. Text is reflective of the currently selected Main View: (0=Fund Mag, 1=In Phase,2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszMainView-Text</i>	: (Output) Main View Text. Valid Values: <ul style="list-style-type: none"> • "Fund Mag" • "In Phase" • "Quad" • "Phase" • "THD" • "Sig Volt" • "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7.2.15 `_PAV2250AFUNC int PAV2250A_GetReadMode (int nPAVNo, int * pnReadModeIndex)`

PAV2250A_GetReadMode sends the command requesting the current Read Mode Index to the PAV2250A device. Read Mode index indicates whether the PAV is showing SIG/REF, REF, SIG or REF/SIG. Index is zero-based.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnReadMode-Index</i>	: (Output) ReadMode Index. Valid Values: <ul style="list-style-type: none"> • 0 : "SIG/REF" • 1 : "REF" • 2 : "SIG" • 3 : "REF/SIG"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.7.2.16 `_PAV2250AFUNC int PAV2250A_GetReadModeText (int nPAVNo, char * pszReadModeText)`

PAV2250A_GetReadModeText sends the command requesting the current Read Mode text to the PAV2250A device. Text is reflective of the currently selected Reading Mode (SIG/REF, REF, SIG, or REF/SIG).

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszReadMode-Text</i>	: (Output) Read Mode Text. Valid Values: <ul style="list-style-type: none"> • "SIG/REF" • "REF" • "SIG" • "REF/SIG"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7.2.17 `_PAV2250AFUNC int PAV2250A_GetScreenBrightness (int nPAVNo, int * pnBrightness)`

PAV2250A_GetScreenBrightness sends the command requesting the current Screen Brightness to the PAV2250A device. Screen Brightness indicates how bright the screen is illuminated. Valid values are between 15-100.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnBrightness</i>	: (Output) Current level of screen brightness - valid values: 15 - 100

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.18 `_PAV2250AFUNC int PAV2250A_GetTabView (int nPAVNo, int * pnTabIndex)`

PAV2250A_GetTabView sends the command requesting the current Tab View Index to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnTabIndex</i>	: (Output) Tab Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Main Tab" • 1 : "Reference Tab" (FUTURE) • 2 : "Harmonics Tab" • 3 : "QUAD Tab"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.7.2.19 `_PAV2250AFUNC` int PAV2250A_GetTabViewText (int *nPAVNo*, char * *pszTabViewText*)

PAV2250A_GetTabViewText sends the command requesting the current Tab View text to the PAV2250A device. Text is reflective of the currently selected tab label. (Main, Harmonics, Quad View)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszTabViewText</i>	: (Output) Tab View Text. Valid Values: <ul style="list-style-type: none"> • "Main" • "Reference" • "Harmonics" • "Custom"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.7.2.20 `_PAV2250AFUNC` int PAV2250A_GoToLocal (int *nPAVNo*)

PAV2250A_GoToLocal sends the command to set the communication mode to Local mode. In Local mode, remote set commands will not be accepted.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.21 `_PAV2250AFUNC` int PAV2250A_SetCustView1 (int *nPAVNo*, int *nCustView1Index*)

PAV2250A_SetCustView1 sends the command to set the Quad View 1 Index to the PAV2250A device. Quad View 1 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nCustView1-Index</i>	: (Input) Index of the desired view to display <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.22 `_PAV2250AFUNC int PAV2250A_SetCustView1FundMag (int nPAVNo)`

PAV2250A_SetCustView1FundMag sends the command to set the Quad View 1 slot to Fundamental Magnitude.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.23 `_PAV2250AFUNC int PAV2250A_SetCustView1InPhase (int nPAVNo)`

PAV2250A_SetCustView1InPhase sends the command to set the Quad View 1 slot to In Phase.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.24 `_PAV2250AFUNC int PAV2250A_SetCustView1Phase (int nPAVNo)`

PAV2250A_SetCustView1Phase sends the command to set the Quad View 1 slot to Phase Angle.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.25 `_PAV2250AFUNC int PAV2250A_SetCustView1Quad (int nPAVNo)`

PAV2250A_SetCustView1Quad sends the command to set the Quad View 1 slot to Quadrature.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.26 `_PAV2250AFUNC int PAV2250A_SetCustView1RefVolt (int nPAVNo)`

PAV2250A_SetCustView1RefVolt sends the command to set the Quad View 1 slot to Reference Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.27 `_PAV2250AFUNC int PAV2250A_SetCustView1SigVolt (int nPAVNo)`

PAV2250A_SetCustView1SigVolt sends the command to set the Quad View 1 slot to Signal Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.28 `_PAV2250AFUNC int PAV2250A_SetCustView1THD (int nPAVNo)`

PAV2250A_SetCustView1THD sends the command to set the Quad View 1 slot to THD.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.29 `_PAV2250AFUNC int PAV2250A_SetCustView2 (int nPAVNo, int nCustView2Index)`

PAV2250A_SetCustView2 sends the command to set the Quad View 2 Index to the PAV2250A device. Quad View 2 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nCustView2-Index</i>	: (Input) Index of the desired view to display <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.30 `_PAV2250AFUNC int PAV2250A_SetCustView2FundMag (int nPAVNo)`

PAV2250A_SetCustView2FundMag sends the command to set the Quad View 2 slot to Fundamental Magnitude.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.31 `_PAV2250AFUNC int PAV2250A_SetCustView2InPhase (int nPAVNo)`

PAV2250A_SetCustView2InPhase sends the command to set the Quad View 2 slot to In Phase.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.32 `_PAV2250AFUNC int PAV2250A_SetCustView2Phase (int nPAVNo)`

PAV2250A_SetCustView2Phase sends the command to set the Quad View 2 slot to Phase Angle.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.33 `_PAV2250AFUNC int PAV2250A_SetCustView2Quad (int nPAVNo)`

PAV2250A_SetCustView2Quad sends the command to set the Quad View 2 slot to Quadrature.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.34 `_PAV2250AFUNC int PAV2250A_SetCustView2RefVolt (int nPAVNo)`

PAV2250A_SetCustView2RefVolt sends the command to set the Quad View 2 slot to Reference Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.35 _PAV2250AFUNC int PAV2250A_SetCustView2SigVolt (int nPAVNo)

PAV2250A_SetCustView2SigVolt sends the command to set the Quad View 2 slot to Signal Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.36 _PAV2250AFUNC int PAV2250A_SetCustView2THD (int nPAVNo)

PAV2250A_SetCustView2THD sends the command to set the Quad View 2 slot to THD.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.37 _PAV2250AFUNC int PAV2250A_SetCustView3 (int nPAVNo, int nCustView3Index)

PAV2250A_SetCustView3 sends the command to set the Quad View 3 Index to the PAV2250A device. Quad View 3 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nCustView3-Index</i>	: (Input) Index of the desired view to display <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.38 `_PAV2250AFUNC int PAV2250A_SetCustView3FundMag (int nPAVNo)`

PAV2250A_SetCustView3FundMag sends the command to set the Quad View 3 slot to Fundamental Magnitude.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.39 `_PAV2250AFUNC int PAV2250A_SetCustView3InPhase (int nPAVNo)`

PAV2250A_SetCustView3InPhase sends the command to set the Quad View 3 slot to In Phase.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.40 `_PAV2250AFUNC int PAV2250A_SetCustView3Phase (int nPAVNo)`

PAV2250A_SetCustView3Phase sends the command to set the Quad View 3 slot to Phase Angle.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.41 `_PAV2250AFUNC int PAV2250A_SetCustView3Quad (int nPAVNo)`

PAV2250A_SetCustView3Quad sends the command to set the Quad View 3 slot to Quadrature.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.42 `_PAV2250AFUNC int PAV2250A_SetCustView3RefVolt (int nPAVNo)`

PAV2250A_SetCustView3RefVolt sends the command to set the Quad View 3 slot to Reference Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.43 `_PAV2250AFUNC int PAV2250A_SetCustView3SigVolt (int nPAVNo)`

PAV2250A_SetCustView3SigVolt sends the command to set the Quad View 3 slot to Signal Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.44 `_PAV2250AFUNC int PAV2250A_SetCustView3THD (int nPAVNo)`

PAV2250A_SetCustView3THD sends the command to set the Quad View 3 slot to THD.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.45 `_PAV2250AFUNC int PAV2250A_SetCustView4 (int nPAVNo, int nCustView4Index)`

PAV2250A_SetCustView4 sends the command to set the Quad View 4 Index to the PAV2250A device. Quad View 4 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nCustView4-Index</i>	: (Input) Index of the desired view to display <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.46 `_PAV2250AFUNC int PAV2250A_SetCustView4FundMag (int nPAVNo)`

PAV2250A_SetCustView4FundMag sends the command to set the Quad View 4 slot to Fundamental Magnitude.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.47 `_PAV2250AFUNC int PAV2250A_SetCustView4InPhase (int nPAVNo)`

PAV2250A_SetCustView4InPhase sends the command to set the Quad View 4 slot to In Phase.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.48 `_PAV2250AFUNC int PAV2250A_SetCustView4Phase (int nPAVNo)`

PAV2250A_SetCustView4Phase sends the command to set the Quad View 4 slot to Phase Angle.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.49 `_PAV2250AFUNC int PAV2250A_SetCustView4Quad (int nPAVNo)`

PAV2250A_SetCustView4Quad sends the command to set the Quad View 4 slot to Quadrature.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.50 `_PAV2250AFUNC int PAV2250A_SetCustView4RefVolt (int nPAVNo)`

PAV2250A_SetCustView4RefVolt sends the command to set the Quad View 4 slot to Reference Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.51 `_PAV2250AFUNC int PAV2250A_SetCustView4SigVolt (int nPAVNo)`

PAV2250A_SetCustView4SigVolt sends the command to set the Quad View 4 slot to Signal Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.52 `_PAV2250AFUNC int PAV2250A_SetCustView4THD (int nPAVNo)`

PAV2250A_SetCustView4THD sends the command to set the Quad View 4 slot to THD.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.53 `_PAV2250AFUNC int PAV2250A_SetHoldDataState (int nPAVNo, bool bHoldData)`

PAV2250A_SetHoldDataState sends the command to set the Hold Data State to the PAV2250A device. Hold Data when true indicates no screen refreshes are taking place.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>bHoldData</i>	: (Input) True to force PAV to be put in "Hold" state (no screen refreshes) False to allow screen refreshes

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.54 `_PAV2250AFUNC int PAV2250A_SetIEEELang (int nPAVNo, int nIEEELang)`

PAV2250A_SetIEEELang sends the command to set the IEEE protocol language to accept when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nIEEELang</i>	: (Input) Language Protocol to be used to communicate via IEEE to 2250A. 2250A Language Types: <ul style="list-style-type: none"> • PAV2250A_NATIVE : 0 • PAV2250_LEGACY : 1

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.55 `_PAV2250AFUNC int PAV2250A_SetIEEELang2250ANative (int nPAVNo)`

PAV2250A_SetIEEELang2250ANative sends the command to set the IEEE protocol to the 2250A Native language when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

<param name="nIEEELang" : (Input) Language Protocol to be used to communicate via IEEE to 2250A. 2250A Language Types:

- PAV2250A_NATIVE : 0
- PAV2250_LEGACY : 1

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.56 _PAV2250AFUNC int PAV2250A_SetIEEELang2250Legacy (int nPAVNo)

PAV2250A_SetIEEELang2250Legacy sends the command to set the IEEE protocol to the 2250A Native language when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

<param name="nIEEELang" : (Input) Language Protocol to be used to communicate via IEEE to 2250A. 2250A Language Types:

- PAV2250A_NATIVE : 0
- PAV2250_LEGACY : 1

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.57 _PAV2250AFUNC int PAV2250A_SetMainView (int nPAVNo, int nMainViewIndex)

PAV2250A_SetMainView sends the command to set the Main View Index to the PAV2250A device. Main View index indicates the view currently shown in the main display. Index is zero-based. (0=Fund Mag, 1=In Phase,2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMainViewIndex</i>	: (Input) Main View Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.58 `_PAV2250AFUNC int PAV2250A_SetMainViewFundMag (int nPAVNo)`

PAV2250A_SetMainViewFundMag sends the command to set the Main View to Fundamental Magnitude.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.59 `_PAV2250AFUNC int PAV2250A_SetMainViewInPhase (int nPAVNo)`

PAV2250A_SetMainViewInPhase sends the command to set the Main View to In Phase.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.60 `_PAV2250AFUNC int PAV2250A_SetMainViewPhase (int nPAVNo)`

PAV2250A_SetMainViewPhase sends the command to set the Main View to Phase.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.61 `_PAV2250AFUNC int PAV2250A_SetMainViewQuad (int nPAVNo)`

PAV2250A_SetMainViewQuad sends the command to set the Main View to Quadrature.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.62 `_PAV2250AFUNC int PAV2250A_SetMainViewRefVolt (int nPAVNo)`

PAV2250A_SetMainViewRefVolt sends the command to set the Main View to Reference Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.63 `_PAV2250AFUNC int PAV2250A_SetMainViewSigVolt (int nPAVNo)`

PAV2250A_SetMainViewSigVolt sends the command to set the Main View to Signal Voltage.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.64 `_PAV2250AFUNC int PAV2250A_SetMainViewTHD (int nPAVNo)`

PAV2250A_SetMainViewTHD sends the command to set the Main View to THD.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.65 `_PAV2250AFUNC int PAV2250A_SetReadMode (int nPAVNo, int nReadModelIndex)`

PAV2250A_SetReadMode sends the command to set the Read Mode Index to the PAV2250A device. Read Mode index indicates whether the PAV is showing SIG/REF, REF, SIG or REF/SIG. Index is zero-based.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nReadMode-Index</i>	: (Input) ReadMode Index. Valid Values: <ul style="list-style-type: none"> • 0 : "SIG/REF" • 1 : "REF" • 2 : "SIG" • 3 : "REF/SIG"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.66 `_PAV2250AFUNC int PAV2250A_SetReadModeRefRef (int nPAVNo)`

PAV2250A_SetReadModeRefRef sends the command to set the Read Mode to Ref.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.67 `_PAV2250AFUNC int PAV2250A_SetReadModeRefSig (int nPAVNo)`

PAV2250A_SetReadModeRefSig sends the command to set the Read Mode to Ref/Sig.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.68 _PAV2250AFUNC int PAV2250A_SetReadModeSigRef (int nPAVNo)

PAV2250A_SetReadModeSigRef sends the command to set the Read Mode to Sig/Ref.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.69 _PAV2250AFUNC int PAV2250A_SetReadModeSigSig (int nPAVNo)

PAV2250A_SetReadModeSigSig sends the command to set the Read Mode to Sig.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.70 _PAV2250AFUNC int PAV2250A_SetRemoteEthernet (int nPAVNo)

PAV2250A_SetRemoteEthernet sends the command to set the communication mode to Remote Ethernet mode. In Remote Ethernet mode, remote set commands will be accepted if the command is received from the Ethernet interface.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.71 _PAV2250AFUNC int PAV2250A_SetRemoteIEEE (int nPAVNo)

PAV2250A_SetRemoteIEEE sends the command to set the communication mode to Remote IEEE mode. In Remote IEEE mode, remote set commands will be accepted if the command is received from the IEEE interface.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.72 _PAV2250AFUNC int PAV2250A_SetRemoteJ1 (int nPAVNo)

PAV2250A_SetRemoteJ1 sends the command to set the communication mode to Remote J1 mode. In Remote J1 mode, remote set commands will be accepted if the command is received from the J1 interface.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.73 `_PAV2250AFUNC int PAV2250A_SetRemoteUSB (int nPAVNo)`

PAV2250A_SetRemoteUSB sends the command to set the communication mode to Remote USB mode. In Remote USB mode, remote set commands will be accepted if the command is received from the USB interface.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.74 `_PAV2250AFUNC int PAV2250A_SetScreenBrightness (int nPAVNo, int nBrightness)`

PAV2250A_SetScreenBrightness sends the command setting the current Screen Brightness on the PAV2250A device. Screen Brightness indicates how bright the screen is illuminated. Valid values are between 15-100.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nBrightness</i>	: (Input) Level of desired screen brightness - valid values: 15 - 100

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.75 `_PAV2250AFUNC int PAV2250A_SetTabView (int nPAVNo, int nTabViewIndex)`

PAV2250A_SetTabView sends the command to set the Tab View Index to the PAV2250A device. Tab View Index should be reflective of which tab the PAV should display as the active tab. Index is zero-based.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nTabViewIndex</i>	: (Input) Tab View Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Main" • 1 : "Reference" • 2 : "Harmonics" • 3 : "CustView"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.76 _PAV2250AFUNC int PAV2250A_SetTabViewCustom (int nPAVNo)

PAV2250A_SetTabViewCustom sends the command to set the Tab View to the Custom View tab.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.77 _PAV2250AFUNC int PAV2250A_SetTabViewHarmonics (int nPAVNo)

PAV2250A_SetTabViewHarmonics sends the command to set the Tab View to the Harmonics tab.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.78 _PAV2250AFUNC int PAV2250A_SetTabViewLVDT (int nPAVNo)

PAV2250A_SetTabViewLVDT sends the command to set the Tab View to the LVDT View tab.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.79 `_PAV2250AFUNC int PAV2250A_SetTabViewMain (int nPAVNo)`

PAV2250A_SetTabViewMain sends the command to set the Tab View to the Main tab.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.7.2.80 `_PAV2250AFUNC int PAV2250A_SetTabViewReference (int nPAVNo)`

PAV2250A_SetTabViewReference sends the command to set the Tab View to the Reference tab.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.8 Calibration Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_Calibrate](#) (int nPAVNo)
PAV2250A_Calibrate sends the command to force the PAV to perform calibration.
- [_PAV2250AFUNC](#) int [PAV2250A_GetCalState](#) (int nPAVNo, char *pszCalState)
PAV2250A_GetCalState sends a command to the PAV 2250A to retrieve its current calibration state.

3.8.1 Detailed Description

3.8.2 Function Documentation

3.8.2.1 [_PAV2250AFUNC](#) int [PAV2250A_Calibrate](#) (int nPAVNo)

[PAV2250A_Calibrate](#) sends the command to force the PAV to perform calibration.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.8.2.2 [_PAV2250AFUNC](#) int [PAV2250A_GetCalState](#) (int nPAVNo, char * pszCalState)

[PAV2250A_GetCalState](#) sends a command to the PAV 2250A to retrieve its current calibration state.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszCalState</i>	: (Output) Possible values: <ul style="list-style-type: none"> • "CALIBRATING" if the PAV is currently calibrating • "READY" if the PAV is ready to go

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function is not supported
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.9 Miscellaneous Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_MaxRetry](#) (int nMaxRetry)
PAV2250A_MaxRetry sets the maximum retries to send a command or read a response that will be made when communicating via IEEE. The default value is 0.
- [_PAV2250AFUNC](#) int [PAV2250A_LastCmdSent](#) (int nPAVNo, char szLastCommand[])
PAV2250A_LastCmdSent returns the last command sent via IEEE, USB or Ethernet to the 2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_WriteCommand](#) (int nPAVNo, char szCommand[])
PAV2250A_WriteCommand sends the command to the 2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_QueryCommand](#) (int nPAVNo, char szCommand[], char *pszResponse)
PAV2250A_QueryCommand sends the command to the 2250A device and waits for the 2250A to respond.

3.9.1 Detailed Description

3.9.2 Function Documentation

3.9.2.1 [_PAV2250AFUNC](#) int [PAV2250A_LastCmdSent](#) (int nPAVNo, char szLastCommand[])

[PAV2250A_LastCmdSent](#) returns the last command sent via IEEE, USB or Ethernet to the 2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>szLastCommand</i>	: (Output) Last command sent to 2250A

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

3.9.2.2 [_PAV2250AFUNC](#) int [PAV2250A_MaxRetry](#) (int nMaxRetry)

[PAV2250A_MaxRetry](#) sets the maximum retries to send a command or read a response that will be made when communicating via IEEE. The default value is 0.

Parameters

<i>nMaxRetry</i>	: (Input) Maximum retries for IEEE communication
------------------	--

Returns

- PAV_SUCCESS : Function is successful

/*

3.9.2.3 [_PAV2250AFUNC](#) int [PAV2250A_QueryCommand](#) (int nPAVNo, char szCommand[], char * pszResponse)

[PAV2250A_QueryCommand](#) sends the command to the 2250A device and waits for the 2250A to respond.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>szCommand</i>	: (Input) command to send to 2250A.
<i>pszResponse</i>	: (Output) 2250A response to the command sent.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.9.2.4 `_PAV2250AFUNC int PAV2250A_WriteCommand (int nPAVNo, char szCommand[])`

PAV2250A_WriteCommand sends the command to the 2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>szCommand</i>	: (Input) Command to send to 2250A

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.10 Time Window Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeWndAuto](#) (int nPAVNo, bool bAuto)
PAV2250A_SetTimeWndAuto sends a command to the PAV 2250A to force the time window to be in "Auto" mode. The system will decide the data refresh rate.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndAuto](#) (int nPAVNo, bool *pbAuto)
PAV2250A_GetTimeWndAuto sends a command to the PAV 2250A to retrieve whether or not the Time Window is in "Auto" mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeWndOverride](#) (int nPAVNo, float fOverrideInterval)
PAV2250A_SetTimeWndOverride sends a command to the PAV 2250A to force the Time Window to be a specific value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndOverride](#) (int nPAVNo, float *pfOverrideInterval)
PAV2250A_GetTimeWndOverride sends a command to the PAV 2250A to retrieve the current setting for the Time Window override interval.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndActual](#) (int nPAVNo, float *pfActualInterval)
PAV2250A_GetTimeWndActual sends a command to the PAV 2250A to retrieve the actual TimeWindow interval. NOTE: If the PAV's TimeWindo is configured to "Auto", the system decides the appropriate TimeWindow.

3.10.1 Detailed Description

3.10.2 Function Documentation

3.10.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndActual](#) (int nPAVNo, float * pfActualInterval)

[PAV2250A_GetTimeWndActual](#) sends a command to the PAV 2250A to retrieve the actual TimeWindow interval.
NOTE: If the PAV's TimeWindo is configured to "Auto", the system decides the appropriate TimeWindow.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfActualInterval</i>	: (Output) Value of the actual Time Window interval.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.10.2.2 [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndAuto](#) (int nPAVNo, bool * pbAuto)

[PAV2250A_GetTimeWndAuto](#) sends a command to the PAV 2250A to retrieve whether or not the Time Window is in "Auto" mode.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbAuto</i>	: (Output) True if Time Window is currently set to "Auto"; False otherwise

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.10.2.3 `_PAV2250AFUNC int PAV2250A_GetTimeWndOverride (int nPAVNo, float * pfOverrideInterval)`

PAV2250A_GetTimeWndOverride sends a command to the PAV 2250A to retrieve the current settign for the Time Window override interval.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfOverride-Interval</i>	: (Output) Value of the override Time Window interval.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.10.2.4 `_PAV2250AFUNC int PAV2250A_SetTimeWndAuto (int nPAVNo, bool bAuto)`

PAV2250A_SetTimeWndAuto sends a command to the PAV 2250A to force the time window to be in "Auto" mode. The system will decide the data refresh rate.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>bAuto</i>	: (Input) True to force "Auto" mode; False to put the Time Window in manual mode

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.10.2.5 `_PAV2250AFUNC int PAV2250A_SetTimeWndOverride (int nPAVNo, float fOverrideInterval)`

PAV2250A_SetTimeWndOverride sends a command to the PAV 2250A to force the Time Window to be a specific value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fOverrideInterval</i>	: (Input) Value to override Time Window to.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11 Setup Options Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetSignalInputOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetSignalInputOption sends a command to the PAV 2250A requesting the current Signal Input Value to the PAV2250A device.
- int [PAV2250A_GetSignalInputOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetSignalInputOptionText sends the command requesting the current Signal Input Text value to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetSignalInputOption sends the command to set the current Signal Input Value to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputFront](#) (int nPAVNo)
PAV2250A_SetSignalInputFront sends the command to set the current Signal Input Value to the Front Panel.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputBack](#) (int nPAVNo)
PAV2250A_SetSignalInputBack sends the command to set the current Signal Input Value to the Back Panel.
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetMainDisplayOption sends the command requesting the current Main Display Index to the PAV2250A device. Main Display can either be 0 ("Independent View") or 1 ("Linked View")
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetMainDisplayOptionText sends the command requesting the current Main Display Option Text to the PAV2250A device. Main Display can either be "Independent View" or "Linked View".
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetMainDisplayOption sends the command to set the Main Display Index Value to the PAV2250A device. Main Display can either be 0 ("Independent View") or 1 ("Linked View").
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayIndependent](#) (int nPAVNo)
PAV2250A_SetMainDisplayIndependent sends the command to set the Main Display Value to Independent View.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayLinked](#) (int nPAVNo)
PAV2250A_SetMainDisplayLinked sends the command to set the Main Display Value to Linked View.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetTimeDisplayOption sends the command requesting the current Time Display Option Index to the PAV2250A device. Time Display can either be 0 ("AM/PM") or 1 ("Military")
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetTimeDisplayOptionText sends the command requesting the current Time Display Option Text to the PAV2250A device. Time Display can either be "AM/PM" or "Military".
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetTimeDisplayOption sends the command to set the Time Display Option Value to the PAV2250A device. Time Display can either be 0 ("AM/PM") or 1 ("Military")
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayAMPM](#) (int nPAVNo)
PAV2250A_SetTimeDisplayAMPM sends the command to set the Time Display Option Value to AM/PM.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayMilitary](#) (int nPAVNo)
PAV2250A_SetTimeDisplayMilitary sends the command to set the Time Display Option Value to Military.
- [_PAV2250AFUNC](#) int [PAV2250A_GetDateDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetDateDisplayOption
- [_PAV2250AFUNC](#) int [PAV2250A_GetDateDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetDateDisplayOptionText sends the command requesting the current Date Display Option Index to the PAV2250A device. Date Display can either be "Text Date"(MON/DD/YYYY) or "Numeric Date"(01/01/11).
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetDateDisplayOption sends the command to set the Date Display Option Value to the PAV2250A device. Date Display can either be 0 ("Text Date") or 1 ("Numeric Date")
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayText](#) (int nPAVNo)
PAV2250A_SetDateDisplayText sends the command to set the Date Display Option Value to Text format.
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayNumeric](#) (int nPAVNo)

- PAV2250A_SetDateDisplayNumeric* sends the command to set the Date Display Option Value to Numeric format.
- [_PAV2250AFUNC](#) int [PAV2250A_GetAutoSaveOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetAutoSaveOption sends the command requesting the current Auto Save Option Index to the PAV2250-A device. Auto Save Options can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_GetAutoSaveOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetAutoSaveOptionText
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetAutoSaveOption sends the command to set the Auto Save Option Index to the PAV2250A device. Auto Save can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveEnable](#) (int nPAVNo)
PAV2250A_SetAutoSaveEnable sends the command to set the Auto Save Option Index to Enabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveDisable](#) (int nPAVNo)
PAV2250A_SetAutoSaveDisable sends the command to set the Auto Save Option Index to Disabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetAutoUnitsOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetAutoUnitsOption sends the command requesting the current Auto Units Option Index to the PAV2250-A device. Auto Units Options can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_GetAutoUnitsOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetAutoUnitsOptionText sends the command requesting the current Auto Units Option Index to the PAV2250A device. Auto Units can either be "Enabled" or "Disabled"
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetAutoUnitsOption sends the command to set the Auto Units Option Index to the PAV2250A device. Auto Units can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsEnable](#) (int nPAVNo)
PAV2250A_SetAutoUnitsEnable sends the command to set the Auto Units Option Index to Enabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsDisable](#) (int nPAVNo)
PAV2250A_SetAutoUnitsDisable sends the command to set the Auto Units Option Index to Disabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetTouchscreenOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetTouchscreenOption sends the command requesting the current Touchscreen Option Index to the PAV2250A device. Touchscreen Options can either be 0 ("Disabled") or 1 ("Enabled")
 - [_PAV2250AFUNC](#) int [PAV2250A_GetTouchscreenOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetTouchscreenOptionText sends the command requesting the current Touchscreen Option Index to the PAV2250A device. Touchscreen can either be "Enabled" or "Disabled"
 - [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetTouchscreenOption sends the command to set the Touchscreen Option Index to the PAV2250A device. Touchscreen can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenEnable](#) (int nPAVNo)
PAV2250A_SetTouchscreenEnable sends the command to set the Touchscreen Option Index to Enabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenDisable](#) (int nPAVNo)
PAV2250A_SetTouchscreenDisable sends the command to set the Touchscreen Option Index to Disabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetNullMeterRangePercent](#) (int nPAVNo, float fRangePercent)
PAV2250A_SetNullMeterRangePercent sends the command to set the Null Meter Range Percent option.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetNullMeterRangePercent](#) (int nPAVNo, float *pfRangePercent)
PAV2250A_GetNullMeterRangePercent sends the command to get the Null Meter Range Percent option.

3.11.1 Detailed Description

3.11.2 Function Documentation

3.11.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetAutoSaveOption](#) (int nPAVNo, int * pnOptionIndex)

[PAV2250A_GetAutoSaveOption](#) sends the command requesting the current Auto Save Option Index to the PAV2250A device. Auto Save Options can either be 0 ("Disabled") or 1 ("Enabled").

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOptionIndex</i>	: (Output) Auto Save Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Disabled" • 1 : "Enabled"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.11.2.2 `_PAV2250AFUNC int PAV2250A_GetAutoSaveOptionText (int nPAVNo, char * pszOptionText)`

PAV2250A_GetAutoSaveOptionText

to the PAV2250A device. Auto Save can either be "Enabled" or "Disabled"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszOptionText</i>	: (Output) Auto Save Option Text. Valid Values: <ul style="list-style-type: none"> • "Enabled" • "Disabled"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.11.2.3 `_PAV2250AFUNC int PAV2250A_GetAutoUnitsOption (int nPAVNo, int * pnOptionIndex)`

PAV2250A_GetAutoUnitsOption sends the command requesting the current Auto Units Option Index to the PAV2250A device. Auto Units Options can either be 0 ("Disabled") or 1 ("Enabled").

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOptionIndex</i>	: (Output) Auto Units Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Disabled" • 1 : "Enabled"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.11.2.4 _PAV2250AFUNC int PAV2250A_GetAutoUnitsOptionText (int *nPAVNo*, char * *pszOptionText*)

PAV2250A_GetAutoUnitsOptionText sends the command requesting the current Auto Units Option Index to the PAV2250A device. Auto Units can either be "Enabled" or "Disabled"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszOptionText</i>	: (Output) Auto Units Option Text. Valid Values: <ul style="list-style-type: none"> • "Enabled" • "Disabled"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.11.2.5 _PAV2250AFUNC int PAV2250A_GetDateDisplayOption (int *nPAVNo*, int * *pnOptionIndex*)

PAV2250A_GetDateDisplayOption

to the PAV2250A device. Date Display can either be 0 ("Text Date") or 1 ("Numeric Date")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOptionIndex</i>	: (Output) Date Display Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Text Date" • 1 : "Numeric Date"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.11.2.6 `_PAV2250AFUNC int PAV2250A_GetDateDisplayOptionText (int nPAVNo, char * pszOptionText)`

PAV2250A_GetDateDisplayOptionText sends the command requesting the current Date Display Option Index to the PAV2250A device. Date Display can either be "Text Date"(MON/DD/YYYY) or "Numeric Date"(01/01/11).

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszOptionText</i>	: (Output) Date Display Option Text. Valid Values: <ul style="list-style-type: none"> • "Text Date" • "Numeric Date"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.11.2.7 `_PAV2250AFUNC int PAV2250A_GetMainDisplayOption (int nPAVNo, int * pnOptionIndex)`

PAV2250A_GetMainDisplayOption sends the command requesting the current Main Display Index to the PAV2250-A device. Main Display can either be 0 ("Independent View") or 1 ("Linked View")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOptionIndex</i>	: (Output) Main Display Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Independent View" • 1 : "Linked View"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.11.2.8 _PAV2250AFUNC int PAV2250A_GetMainDisplayOptionText (int nPAVNo, char * pszOptionText)

PAV2250A_GetMainDisplayOptionText sends the command requesting the current Main Display Option Text to the PAV2250A device. Main Display can either be "Independent View" or "Linked View".

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszOptionText</i>	: (Output) Main Display Option Text. Valid Values: <ul style="list-style-type: none"> • "Independent View" • "Linked View"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.11.2.9 _PAV2250AFUNC int PAV2250A_GetNullMeterRangePercent (int nPAVNo, float * pfRangePercent)

PAV2250A_GetNullMeterRangePercent sends the command to get the Null Meter Range Percent option.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfRangePercent</i>	: (Output) Percentage of full range the Null meter is currently set to.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.10 `_PAV2250AFUNC int PAV2250A_GetSignalInputOption (int nPAVNo, int * pnOptionIndex)`

PAV2250A_GetSignalInputOption sends a command to the PAV 2250A requesting the current Signal Input Value to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOptionIndex</i>	: (Output) Signal Input Value index. Valid Values: <ul style="list-style-type: none"> • 0 : "Front Panel" • 1 : "Back Panel"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.11.2.11 `int PAV2250A_GetSignalInputOptionText (int nPAVNo, char * pszOptionText)`

PAV2250A_GetSignalInputOptionText sends the command requesting the current Signal Input Text value to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszOptionText</i>	: (Output) Signal Option Text value. Valid Values: <ul style="list-style-type: none"> • "Front Panel" • "Back Panel"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.11.2.12 `_PAV2250AFUNC int PAV2250A_GetTimeDisplayOption (int nPAVNo, int * pnOptionIndex)`

PAV2250A_GetTimeDisplayOption sends the command requesting the current Time Display Option Index to the PAV2250A device. Time Display can either be 0 ("AM/PM") or 1 ("Military")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOptionIndex</i>	: (Output) Time Display Option Valid Values: <ul style="list-style-type: none"> • 0 : "AM/PM" • 1 : "Military"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.11.2.13 `_PAV2250AFUNC int PAV2250A_GetTimeDisplayOptionText (int nPAVNo, char * pszOptionText)`

PAV2250A_GetTimeDisplayOptionText sends the command requesting the current Time Display Option Text to the PAV2250A device. Time Display can either be "AM/PM" or "Military".

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszOptionText</i>	: (Output) Time Display Option Text. Valid Values: <ul style="list-style-type: none"> • "AM/PM" • "Military"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.11.2.14 `_PAV2250AFUNC int PAV2250A_GetTouchscreenOption (int nPAVNo, int * pnOptionIndex)`

PAV2250A_GetTouchscreenOption sends the command requesting the current Touchscreen Option Index to the PAV2250A device. Touchscreen Options can either be 0 ("Disabled") or 1 ("Enabled")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOptionIndex</i>	: (Output) Auto Touchscreen Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Disabled" • 1 : "Enabled"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.11.2.15 `_PAV2250AFUNC int PAV2250A_GetTouchscreenOptionText (int nPAVNo, char * pszOptionText)`

PAV2250A_GetTouchscreenOptionText sends the command requesting the current Touchscreen Option Index to the PAV2250A device. Touchscreen can either be "Enabled" or "Disabled"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszOptionText</i>	: (Output) Touchscreen Option Text. Valid Values: <ul style="list-style-type: none"> • "Enabled" • "Disabled"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.11.2.16 `_PAV2250AFUNC int PAV2250A_SetAutoSaveDisable (int nPAVNo)`

PAV2250A_SetAutoSaveDisable sends the command to set the Auto Save Option Index to Disabled.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.17 `_PAV2250AFUNC int PAV2250A_SetAutoSaveEnable (int nPAVNo)`

PAV2250A_SetAutoSaveEnable sends the command to set the Auto Save Option Index to Enabled.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.18 `_PAV2250AFUNC int PAV2250A_SetAutoSaveOption (int nPAVNo, int nOptionIndex)`

PAV2250A_SetAutoSaveOption sends the command to set the Auto Save Option Index to the PAV2250A device. Auto Save can either be 0 ("Disabled") or 1 ("Enabled").

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nOptionIndex</i>	: (Input) Auto Save Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Disable" • 1 : "Enable"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.19 `_PAV2250AFUNC int PAV2250A_SetAutoUnitsDisable (int nPAVNo)`

PAV2250A_SetAutoUnitsDisable sends the command to set the Auto Units Option Index to Disabled.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid *nPAVNo* parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.20 `_PAV2250AFUNC int PAV2250A_SetAutoUnitsEnable (int nPAVNo)`

PAV2250A_SetAutoUnitsEnable sends the command to set the Auto Units Option Index to Enabled.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid *nPAVNo* parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.21 `_PAV2250AFUNC int PAV2250A_SetAutoUnitsOption (int nPAVNo, int nOptionIndex)`

PAV2250A_SetAutoUnitsOption sends the command to set the Auto Units Option Index to the PAV2250A device. Auto Units can either be 0 ("Disabled") or 1 ("Enabled").

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nOptionIndex</i>	: (Input) Auto Units Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Disabled" • 1 : "Enabled"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.22 `_PAV2250AFUNC` int PAV2250A_SetDateDisplayNumeric (int *nPAVNo*)

PAV2250A_SetDateDisplayNumeric sends the command to set the Date Display Option Value to Numeric format.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.23 `_PAV2250AFUNC` int PAV2250A_SetDateDisplayOption (int *nPAVNo*, int *nOptionIndex*)

PAV2250A_SetDateDisplayOption sends the command to set the Date Display Option Value to the PAV2250A device. Date Display can either be 0 ("Text Date") or 1 ("Numeric Date")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nOptionIndex</i>	: (Input) Date Display Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Text Date" • 1 : "Numeric Date"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.24 `_PAV2250AFUNC` int `PAV2250A_SetDateDisplayText` (int *nPAVNo*)

`PAV2250A_SetDateDisplayText` sends the command to set the Date Display Option Value to Text format.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.25 `_PAV2250AFUNC int PAV2250A_SetMainDisplayIndependent (int nPAVNo)`

PAV2250A_SetMainDisplayIndependent sends the command to set the Main Display Value to Independent View.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.26 `_PAV2250AFUNC int PAV2250A_SetMainDisplayLinked (int nPAVNo)`

PAV2250A_SetMainDisplayLinked sends the command to set the Main Display Value to Linked View.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.27 `_PAV2250AFUNC int PAV2250A_SetMainDisplayOption (int nPAVNo, int nOptionIndex)`

PAV2250A_SetMainDisplayOption sends the command to set the Main Display Index Value to the PAV2250A device. Main Display can either be 0 ("Independent View") or 1 ("Linked View").

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nOptionIndex</i>	: (Input) Main Display Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Independent View" • 1 : "Linked View"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.28 `_PAV2250AFUNC int PAV2250A_SetNullMeterRangePercent (int nPAVNo, float fRangePercent)`

PAV2250A_SetNullMeterRangePercent sends the command to set the Null Meter Range Percent option.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fRangePercent</i>	: (Input) Percentage of full range the Null meter should be set to.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.29 `_PAV2250AFUNC int PAV2250A_SetSignalInputBack (int nPAVNo)`

PAV2250A_SetSignalInputBack sends the command to set the current Signal Input Value to the Back Panel.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.30 `_PAV2250AFUNC` int `PAV2250A_SetSignalInputFront` (int *nPAVNo*)

`PAV2250A_SetSignalInputFront` sends the command to set the current Signal Input Value to the Front Panel.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid *nPAVNo* parameter
- `PAV_ERROR_WRITE` : Unable to send command to 2250A
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.31 `_PAV2250AFUNC` int `PAV2250A_SetSignalInputOption` (int *nPAVNo*, int *nOptionIndex*)

`PAV2250A_SetSignalInputOption` sends the command to set the current Signal Input Value to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nOptionIndex</i>	: (Input) Signal Input Value index. Valid Values: <ul style="list-style-type: none"> • 0 : "Front Panel" • 1 : "Back Panel"

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid *nPAVNo* parameter
- `PAV_ERROR_WRITE` : Unable to send command to 2250A
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.32 `_PAV2250AFUNC` int `PAV2250A_SetTimeDisplayAMPM` (int *nPAVNo*)

`PAV2250A_SetTimeDisplayAMPM` sends the command to set the Time Display Option Value to AM/PM.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.33 `_PAV2250AFUNC` int PAV2250A_SetTimeDisplayMilitary (int *nPAVNo*)

PAV2250A_SetTimeDisplayMilitary sends the command to set the Time Display Option Value to Military.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.34 `_PAV2250AFUNC` int PAV2250A_SetTimeDisplayOption (int *nPAVNo*, int *nOptionIndex*)

PAV2250A_SetTimeDisplayOption sends the command to set the Time Display Option Value to the PAV2250A device. Time Display can either be 0 ("AM/PM") or 1 ("Military")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nOptionIndex</i>	: (Input) Time Display Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "AM/PM" • 1 : "Military"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.35 `_PAV2250AFUNC` int `PAV2250A_SetTouchscreenDisable` (int *nPAVNo*)

`PAV2250A_SetTouchscreenDisable` sends the command to set the Touchscreen Option Index to Disabled.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.36 `_PAV2250AFUNC int PAV2250A_SetTouchscreenEnable (int nPAVNo)`

PAV2250A_SetTouchscreenEnable sends the command to set the Touchscreen Option Index to Enabled.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.11.2.37 `_PAV2250AFUNC int PAV2250A_SetTouchscreenOption (int nPAVNo, int nOptionIndex)`

PAV2250A_SetTouchscreenOption sends the command to set the Touchscreen Option Index to the PAV2250A device. Touchscreen can either be 0 ("Disabled") or 1 ("Enabled").

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nOptionIndex</i>	: (Input) Touchscreen Option Index. Valid Values: <ul style="list-style-type: none"> • 0 : "Disabled" • 1 : "Enabled"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.12 View Configuration Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexConfig](#) (int nPAVNo, int nViewIndex, char *pszViewConfig)

PAV2250A_GetViewIndexConfig sends the command to get the view configuration for the specified view index. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewFundMagConfig sends the command to get the view configuration for the Fundamental Magnitude. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewInPhaseConfig sends the command to get the view configuration for the In Phase component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewQuadConfig sends the command to get the view configuration for the Quad component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewPhaseConfig sends the command to get the view configuration for the Phase component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewTHDConfig sends the command to get the view configuration for the THD component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewSigVoltConfig sends the command to get the view configuration for the Sig Volt component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewRefVoltConfig sends the command to get the view configuration for the Ref Volt component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewSigOffsetConfig sends the command to get the view configuration for the Sig Offset component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewTotalRatioConfig sends the command to get the view configuration for the Total Ratio component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewFrequencyConfig sends the command to get the view configuration for the Frequency component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewMainConfig sends the command to get the view configuration for the Main view. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

3.12.1 Detailed Description

3.12.2 Function Documentation

3.12.2.1 `_PAV2250AFUNC int PAV2250A_GetViewFrequencyConfig (int nPAVNo, char * pszViewConfig)`

`PAV2250A_GetViewFrequencyConfig` sends the command to get the view configuration for the Frequency component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid `nPAVNo` parameter
- `PAV_ERROR_WRITE` : Unable to send command to 2250A
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.2 `_PAV2250AFUNC int PAV2250A_GetViewFundMagConfig (int nPAVNo, char * pszViewConfig)`

`PAV2250A_GetViewFundMagConfig` sends the command to get the view configuration for the Fundamental Magnitude. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.3 `_PAV2250AFUNC int PAV2250A_GetViewIndexConfig (int nPAVNo, int nViewIndex, char * pszViewConfig)`

PAV2250A_GetViewIndexConfig sends the command to get the view configuration for the specified view index. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view for which to retrieve configuration data. <ul style="list-style-type: none"> • 0 : "FundMag" • 1 : "InPhase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "SigVolt" • 6 : "RefVolt" • 7 : "SigOffset" • 8 : "TotalRatio" • 9 : "Freq" • 10 : "Main"

<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale
----------------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.4 _PAV2250AFUNC int PAV2250A_GetViewInPhaseConfig (int nPAVNo, char * pszViewConfig)

PAV2250A_GetViewInPhaseConfig sends the command to get the view configuration for the In Phase component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.5 `_PAV2250AFUNC int PAV2250A_GetViewMainConfig (int nPAVNo, char * pszViewConfig)`

`PAV2250A_GetViewMainConfig` sends the command to get the view configuration for the Main view. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.6 `_PAV2250AFUNC int PAV2250A_GetViewPhaseConfig (int nPAVNo, char * pszViewConfig)`

PAV2250A_GetViewPhaseConfig sends the command to get the view configuration for the Phase component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.7 `_PAV2250AFUNC int PAV2250A_GetViewQuadConfig (int nPAVNo, char * pszViewConfig)`

PAV2250A_GetViewQuadConfig sends the command to get the view configuration for the Quad component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.8 `_PAV2250AFUNC int PAV2250A_GetViewRefVoltConfig (int nPAVNo, char * pszViewConfig)`

PAV2250A_GetViewRefVoltConfig sends the command to get the view configuration for the Ref Volt component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.9 `_PAV2250AFUNC int PAV2250A_GetViewSigOffsetConfig (int nPAVNo, char * pszViewConfig)`

PAV2250A_GetViewSigOffsetConfig sends the command to get the view configuration for the Sig Offset component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.10 `_PAV2250AFUNC int PAV2250A_GetViewSigVoltConfig (int nPAVNo, char * pszViewConfig)`

PAV2250A_GetViewSigVoltConfig sends the command to get the view configuration for the Sig Volt component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.11 `_PAV2250AFUNC int PAV2250A_GetViewTHDConfig (int nPAVNo, char * pszViewConfig)`

PAV2250A_GetViewTHDConfig sends the command to get the view configuration for the THD component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none"> • View Name • Unit Index • Resolution • Offset • Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.12.2.12 `_PAV2250AFUNC int PAV2250A_GetViewTotalRatioConfig (int nPAVNo, char * pszViewConfig)`

PAV2250A_GetViewTotalRatioConfig sends the command to get the view configuration for the Total Ratio component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewConfig</i>	: (Output) Return string - comma separated value containing field values: <ul style="list-style-type: none">• View Name• Unit Index• Resolution• Offset• Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.13 View Max Field Width Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewFundMagMaxFieldWidth sends the command to set the max field width for Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewFundMagMaxFieldWidth sends the command to get the max field width for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewInPhaseMaxFieldWidth sends the command to set the max field width for In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewInPhaseMaxFieldWidth sends the command to get the max field width for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewQuadMaxFieldWidth sends the command to set the max field width for Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewQuadMaxFieldWidth sends the command to get the max field width for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhaseMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewPhaseMaxFieldWidth sends the command to set the max field width for Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewPhaseMaxFieldWidth sends the command to get the max field width for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewTHDMaxFieldWidth sends the command to set the max field width for THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewTHDMaxFieldWidth sends the command to get the max field width for the THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewSigVoltMaxFieldWidth sends the command to set the max field width for Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewSigVoltMaxFieldWidth sends the command to get the max field width for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewRefVoltMaxFieldWidth sends the command to set the max field width for Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewRefVoltMaxFieldWidth sends the command to get the max field width for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewSigOffsetMaxFieldWidth sends the command to set the max field width for Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewSigOffsetMaxFieldWidth sends the command to get the max field width for the SigOffset view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewTotalRatioMaxFieldWidth sends the command to set the max field width for Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewTotalRatioMaxFieldWidth sends the command to get the max field width for the TotalRatio view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewFrequencyMaxFieldWidth sends the command to set the max field width for Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewFrequencyMaxFieldWidth sends the command to get the max field width for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewMainMaxFieldWidth sends the command to set the max field width for Main view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewMainMaxFieldWidth sends the command to get the max field width for the Main view.

3.13.1 Detailed Description

3.13.2 Function Documentation

3.13.2.1 `_PAV2250AFUNC int PAV2250A_GetViewFrequencyMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)`

`PAV2250A_GetViewFrequencyMaxFieldWidth` sends the command to get the max field width for the Frequency view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid `nPAVNo` parameter
- `PAV_ERROR_WRITE` : Unable to send command to 2250A
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.2 `_PAV2250AFUNC int PAV2250A_GetViewFundMagMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)`

`PAV2250A_GetViewFundMagMaxFieldWidth` sends the command to get the max field width for the Fundamental Magnitude view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid `nPAVNo` parameter
- `PAV_ERROR_WRITE` : Unable to send command to 2250A
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.3 `_PAV2250AFUNC int PAV2250A_GetViewInPhaseMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)`

`PAV2250A_GetViewInPhaseMaxFieldWidth` sends the command to get the max field width for the In Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.4 _PAV2250AFUNC int PAV2250A_GetViewMainMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)

PAV2250A_GetViewMainMaxFieldWidth sends the command to get the max field width for the Main view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.5 _PAV2250AFUNC int PAV2250A_GetViewPhaseMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)

PAV2250A_GetViewPhaseMaxFieldWidth sends the command to get the max field width for the Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.6 `_PAV2250AFUNC int PAV2250A_GetViewQuadMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)`

PAV2250A_GetViewQuadMaxFieldWidth sends the command to get the max field width for the Quad view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.7 `_PAV2250AFUNC int PAV2250A_GetViewRefVoltMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)`

PAV2250A_GetViewRefVoltMaxFieldWidth sends the command to get the max field width for the RefVolt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.8 `_PAV2250AFUNC int PAV2250A_GetViewSigOffsetMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)`

PAV2250A_GetViewSigOffsetMaxFieldWidth sends the command to get the max field width for the SigOffset view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.9 `_PAV2250AFUNC` int PAV2250A_GetViewSigVoltMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)

PAV2250A_GetViewSigVoltMaxFieldWidth sends the command to get the max field width for the SigVolt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.10 `_PAV2250AFUNC` int PAV2250A_GetViewTHDMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)

PAV2250A_GetViewTHDMaxFieldWidth sends the command to get the max field width for the THD view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.11 `_PAV2250AFUNC` int PAV2250A_GetViewTotalRatioMaxFieldWidth (int nPAVNo, int * pnMaxFieldWidth)

PAV2250A_GetViewTotalRatioMaxFieldWidth sends the command to get the max field width for the TotalRatio view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnMaxField-Width</i>	: (Output) Max Field Width currently set for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.13.2.12 `_PAV2250AFUNC int PAV2250A_SetViewFrequencyMaxFieldWidth (int nPAVNo, int nMaxFieldWidth)`

PAV2250A_SetViewFrequencyMaxFieldWidth sends the command to set the max field width for Frequency view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.13 `_PAV2250AFUNC int PAV2250A_SetViewFundMagMaxFieldWidth (int nPAVNo, int nMaxFieldWidth)`

PAV2250A_SetViewFundMagMaxFieldWidth sends the command to set the max field width for Fundamental Magnitude view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.14 `_PAV2250AFUNC` int `PAV2250A_SetViewInPhaseMaxFieldWidth` (int *nPAVNo*, int *nMaxFieldWidth*)

`PAV2250A_SetViewInPhaseMaxFieldWidth` sends the command to set the max field width for In Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.15 `_PAV2250AFUNC int PAV2250A_SetViewMainMaxFieldWidth (int nPAVNo, int nMaxFieldWidth)`

PAV2250A_SetViewMainMaxFieldWidth sends the command to set the max field width for Main view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.16 `_PAV2250AFUNC int PAV2250A_SetViewPhaseMaxFieldWidth (int nPAVNo, int nMaxFieldWidth)`

PAV2250A_SetViewPhaseMaxFieldWidth sends the command to set the max field width for Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.17 `_PAV2250AFUNC` int `PAV2250A_SetViewQuadMaxFieldWidth` (int *nPAVNo*, int *nMaxFieldWidth*)

`PAV2250A_SetViewQuadMaxFieldWidth` sends the command to set the max field width for Quad view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.18 `_PAV2250AFUNC int PAV2250A_SetViewRefVoltMaxFieldWidth (int nPAVNo, int nMaxFieldWidth)`

PAV2250A_SetViewRefVoltMaxFieldWidth sends the command to set the max field width for Ref Volt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.19 `_PAV2250AFUNC int PAV2250A_SetViewSigOffsetMaxFieldWidth (int nPAVNo, int nMaxFieldWidth)`

PAV2250A_SetViewSigOffsetMaxFieldWidth sends the command to set the max field width for Sig Offset view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.20 `_PAV2250AFUNC` int `PAV2250A_SetViewSigVoltMaxFieldWidth` (int *nPAVNo*, int *nMaxFieldWidth*)

`PAV2250A_SetViewSigVoltMaxFieldWidth` sends the command to set the max field width for Sig Volt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.21 `_PAV2250AFUNC int PAV2250A_SetViewTHDMaxFieldWidth (int nPAVNo, int nMaxFieldWidth)`

PAV2250A_SetViewTHDMaxFieldWidth sends the command to set the max field width for THD view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.13.2.22 `_PAV2250AFUNC int PAV2250A_SetViewTotalRatioMaxFieldWidth (int nPAVNo, int nMaxFieldWidth)`

PAV2250A_SetViewTotalRatioMaxFieldWidth sends the command to set the max field width for Total Ratio view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nMaxFieldWidth</i>	: (Input) Desired max field width

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14 View Units Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagV](#) (int nPAVNo)
PAV2250A_SetViewFundMagV sends the command to set Fundamental Magnitude units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagMV](#) (int nPAVNo)
PAV2250A_SetViewFundMagMV sends the command to set Fundamental Magnitude units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagPercent](#) (int nPAVNo)
PAV2250A_SetViewFundMagPercent sends the command to set Fundamental Magnitude units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagDB](#) (int nPAVNo)
PAV2250A_SetViewFundMagDB sends the command to set Fundamental Magnitude units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagRatio](#) (int nPAVNo)
PAV2250A_SetViewFundMagRatio sends the command to set Fundamental Magnitude units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewFundMagUnits sends the command to get the view units index for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewFundMagUnitsText sends the command to get the view units for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseV](#) (int nPAVNo)
PAV2250A_SetViewInPhaseV sends the command to set InPhase units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseMV](#) (int nPAVNo)
PAV2250A_SetViewInPhaseMV sends the command to set In Phase units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhasePercent](#) (int nPAVNo)
PAV2250A_SetViewInPhasePercent sends the command to set In Phase units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseDB](#) (int nPAVNo)
PAV2250A_SetViewInPhaseDB sends the command to set In Phase units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseRatio](#) (int nPAVNo)
PAV2250A_SetViewInPhaseRatio sends the command to set Fundamental In Phase units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewInPhaseUnits sends the command to get the view units index for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewInPhaseUnitsText sends the command to get the view units for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadV](#) (int nPAVNo)
PAV2250A_SetViewQuadV sends the command to set Quad units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadMV](#) (int nPAVNo)
PAV2250A_SetViewQuadMV sends the command to set Quad units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadPercent](#) (int nPAVNo)
PAV2250A_SetViewQuadPercent sends the command to set Quad units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadDB](#) (int nPAVNo)
PAV2250A_SetViewQuadDB sends the command to set Quad units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadRatio](#) (int nPAVNo)
PAV2250A_SetViewQuadRatio sends the command to set Quad units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewQuadUnits sends the command to get the view units index for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewQuadUnitsText sends the command to get the view units for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhase360](#) (int nPAVNo)
PAV2250A_SetViewPhase360 sends the command to set Phase units to +-360.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhase180](#) (int nPAVNo)

- PAV2250A_SetViewPhase180 sends the command to set Phase units to +/-180.*

 - [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewPhaseUnits sends the command to get the view units index for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewPhaseUnitsText sends the command to get the view units for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDPercent](#) (int nPAVNo)

PAV2250A_SetViewTHDPercent sends the command to set THD units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDDB](#) (int nPAVNo)

PAV2250A_SetViewTHDDB sends the command to set THD units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewTHDUnits sends the command to get the view units index for the THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewTHDUnitsText sends the command to get the view units for the THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltV](#) (int nPAVNo)

PAV2250A_SetViewSigVoltV sends the command to set Signal Voltage units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltMV](#) (int nPAVNo)

PAV2250A_SetViewSigVoltMV sends the command to set Signal Voltage units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewSigVoltUnits sends the command to get the view units index for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewSigVoltUnitsText sends the command to get the view units for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltV](#) (int nPAVNo)

PAV2250A_SetViewRefVoltV sends the command to set Reference Voltage units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltMV](#) (int nPAVNo)

PAV2250A_SetViewRefVoltMV sends the command to set Reference Voltage units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewRefVoltUnits sends the command to get the view units index for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewRefVoltUnitsText sends the command to get the view units for the RefVolt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetV](#) (int nPAVNo)

PAV2250A_SetViewSigOffsetV sends the command to set Signal Offset units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetMV](#) (int nPAVNo)

PAV2250A_SetViewSigOffsetMV sends the command to set Signal Offset units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewSigOffsetUnits sends the command to get the view units index for the Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewSigOffsetUnitsText sends the command to get the view units for the Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioPercent](#) (int nPAVNo)

PAV2250A_SetViewTotalRatioPercent sends the command to set Total Ratio units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioDB](#) (int nPAVNo)

PAV2250A_SetViewTotalRatioDB sends the command to set Total Ratio units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioRatio](#) (int nPAVNo)

PAV2250A_SetViewTotalRatioRatio sends the command to set Total Ratio units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewTotalRatioUnits sends the command to get the view units index for the Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewTotalRatioUnitsText sends the command to get the view units for the Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyHZ](#) (int nPAVNo)

PAV2250A_SetViewFrequencyHZ sends the command to set Frequency units to Hz.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyKHZ](#) (int nPAVNo)

PAV2250A_SetViewFrequencyKHZ sends the command to set Frequency units to KHz.

- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewFrequencyUnits sends the command to get the view units index for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewFrequencyUnitsText sends the command to get the view units for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainV](#) (int nPAVNo)
PAV2250A_SetViewMainV sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainMV](#) (int nPAVNo)
PAV2250A_SetViewMainMV sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainPercent](#) (int nPAVNo)
PAV2250A_SetViewMainPercent sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainDB](#) (int nPAVNo)
PAV2250A_SetViewMainDB sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainRatio](#) (int nPAVNo)
PAV2250A_SetViewMainRatio sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMain360](#) (int nPAVNo)
PAV2250A_SetViewMain360 sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMain180](#) (int nPAVNo)
PAV2250A_SetViewMain180 sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewMainUnits sends the command to get the view units index for the Main view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewMainUnitsText sends the command to get the view units for the Main view.

3.14.1 Detailed Description

3.14.2 Function Documentation

3.14.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyUnits](#) (int nPAVNo, int * pnUnits)

[PAV2250A_GetViewFrequencyUnits](#) sends the command to get the view units index for the Frequency view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 7 : "Hz" • 8 : "KHz"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.2 `_PAV2250AFUNC int PAV2250A_GetViewFrequencyUnitsText (int nPAVNo, char * pszUnits)`

PAV2250A_GetViewFrequencyUnitsText sends the command to get the view units for the Frequency view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • Hz • KHz

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.3 `_PAV2250AFUNC int PAV2250A_GetViewFundMagUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewFundMagUnits sends the command to get the view units index for the Fundamental Magnitude view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 0 : "V" • 1 : "mV" • 2 : "Ratio" • 3 : "Percent" • 4 : "dB"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.4 `_PAV2250AFUNC int PAV2250A_GetViewFundMagUnitsText (int nPAVNo, char * pszUnits)`

`PAV2250A_GetViewFundMagUnitsText` sends the command to get the view units for the Fundamental Magnitude view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • V • mV • Ratio • Percent • dB

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.5 `_PAV2250AFUNC int PAV2250A_GetViewInPhaseUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewInPhaseUnits sends the command to get the view units index for the In Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 0 : "V" • 1 : "mV" • 2 : "Ratio" • 3 : "Percent" • 4 : "DB"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.6 `_PAV2250AFUNC int PAV2250A_GetViewInPhaseUnitsText (int nPAVNo, char * pszUnits)`

PAV2250A_GetViewInPhaseUnitsText sends the command to get the view units for the In Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • V • mV • Ratio • Percent • dB

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.7 `_PAV2250AFUNC int PAV2250A_GetViewMainUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewMainUnits sends the command to get the view units index for the Main view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 0 : "V" • 1 : "mV" • 2 : "Ratio" • 3 : "Percent" • 4 : "DB" • 5 : "360" • 6 : "180"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.8 `_PAV2250AFUNC int PAV2250A_GetViewMainUnitsText (int nPAVNo, char * pszUnits)`

`PAV2250A_GetViewMainUnitsText` sends the command to get the view units for the Main view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • V • mV • Ratio • Percent • dB • 360 • 180

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.9 `_PAV2250AFUNC int PAV2250A_GetViewPhaseUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewPhaseUnits sends the command to get the view units index for the Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 5 : "360" • 6 : "180"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.10 `_PAV2250AFUNC int PAV2250A_GetViewPhaseUnitsText (int nPAVNo, char * pszUnits)`

PAV2250A_GetViewPhaseUnitsText sends the command to get the view units for the Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • 360 • 180

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.11 `_PAV2250AFUNC int PAV2250A_GetViewQuadUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewQuadUnits sends the command to get the view units index for the Quad view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 0 : "V" • 1 : "mV" • 2 : "Ratio" • 3 : "Percent" • 4 : "DB"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.12 `_PAV2250AFUNC int PAV2250A_GetViewQuadUnitsText (int nPAVNo, char * pszUnits)`

PAV2250A_GetViewQuadUnitsText sends the command to get the view units for the Quad view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • V • mV • Ratio • Percent • dB

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.13 `_PAV2250AFUNC int PAV2250A_GetViewRefVoltUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewRefVoltUnits sends the command to get the view units index for the Ref Volt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 0 : "V" • 1 : "mV"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.14 `_PAV2250AFUNC int PAV2250A_GetViewRefVoltUnitsText (int nPAVNo, char * pszUnits)`

PAV2250A_GetViewRefVoltUnitsText sends the command to get the view units for the RefVolt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • V • mV

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.15 `_PAV2250AFUNC int PAV2250A_GetViewSigOffsetUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewSigOffsetUnits sends the command to get the view units index for the Sig Offset view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 0 : "V" • 1 : "mV"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.16 `_PAV2250AFUNC int PAV2250A_GetViewSigOffsetUnitsText (int nPAVNo, char * pszUnits)`

PAV2250A_GetViewSigOffsetUnitsText sends the command to get the view units for the Sig Offset view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • V • mV

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.17 `_PAV2250AFUNC int PAV2250A_GetViewSigVoltUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewSigVoltUnits sends the command to get the view units index for the Sig Volt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 0 : "V" • 1 : "mV"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.18 `_PAV2250AFUNC int PAV2250A_GetViewSigVoltUnitsText (int nPAVNo, char * pszUnits)`

PAV2250A_GetViewSigVoltUnitsText sends the command to get the view units for the Sig Volt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • V • mV
-----------------	---

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.19 _PAV2250AFUNC int PAV2250A_GetViewTHDUnits (int nPAVNo, int * pnUnits)

PAV2250A_GetViewTHDUnits sends the command to get the view units index for the THD view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 3 : "Percent" • 4 : "DB"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.20 _PAV2250AFUNC int PAV2250A_GetViewTHDUnitsText (int nPAVNo, char * pszUnits)

PAV2250A_GetViewTHDUnitsText sends the command to get the view units for the THD view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • Percent • dB

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.21 `_PAV2250AFUNC int PAV2250A_GetViewTotalRatioUnits (int nPAVNo, int * pnUnits)`

PAV2250A_GetViewTotalRatioUnits sends the command to get the view units index for the Total Ratio view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnUnits</i>	: (Output) Returns the Units Index for this view <ul style="list-style-type: none"> • 2 : "Ratio" • 3 : "Percent" • 4 : "DB"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.14.2.22 `_PAV2250AFUNC int PAV2250A_GetViewTotalRatioUnitsText (int nPAVNo, char * pszUnits)`

PAV2250A_GetViewTotalRatioUnitsText sends the command to get the view units for the Total Ratio view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszUnits</i>	: (Output) Text name of Units this view is currently configured to. Valid Values: <ul style="list-style-type: none"> • Ratio • Percent • dB

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.14.2.23 `_PAV2250AFUNC` int PAV2250A_SetViewFrequencyHZ (int nPAVNo)

PAV2250A_SetViewFrequencyHZ sends the command to set Frequency units to Hz.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.24 `_PAV2250AFUNC` int PAV2250A_SetViewFrequencyKHZ (int nPAVNo)

PAV2250A_SetViewFrequencyKHZ sends the command to set Frequency units to KHz.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.25 `_PAV2250AFUNC` int PAV2250A_SetViewFundMagDB (int nPAVNo)

PAV2250A_SetViewFundMagDB sends the command to set Fundamental Magnitude units to dB.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.26 `_PAV2250AFUNC` int PAV2250A_SetViewFundMagMV (int *nPAVNo*)

PAV2250A_SetViewFundMagMV sends the command to set Fundamental Magnitude units to mV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.27 `_PAV2250AFUNC` int PAV2250A_SetViewFundMagPercent (int *nPAVNo*)

PAV2250A_SetViewFundMagPercent sends the command to set Fundamental Magnitude units to %.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.28 `_PAV2250AFUNC` int PAV2250A_SetViewFundMagRatio (int *nPAVNo*)

PAV2250A_SetViewFundMagRatio sends the command to set Fundamental Magnitude units to Ratio.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.29 `_PAV2250AFUNC` int PAV2250A_SetViewFundMagV (int *nPAVNo*)

PAV2250A_SetViewFundMagV sends the command to set Fundamental Magnitude units to V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.30 `_PAV2250AFUNC` int PAV2250A_SetViewInPhaseDB (int *nPAVNo*)

PAV2250A_SetViewInPhaseDB sends the command to set In Phase units to dB.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.31 `_PAV2250AFUNC` int PAV2250A_SetViewInPhaseMV (int *nPAVNo*)

PAV2250A_SetViewInPhaseMV sends the command to set In Phase units to mV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.32 `_PAV2250AFUNC int PAV2250A_SetViewInPhasePercent (int nPAVNo)`

PAV2250A_SetViewInPhasePercent sends the command to set In Phase units to %.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.33 `_PAV2250AFUNC int PAV2250A_SetViewInPhaseRatio (int nPAVNo)`

PAV2250A_SetViewInPhaseRatio sends the command to set Fundamental In Phase units to Ratio.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.34 `_PAV2250AFUNC int PAV2250A_SetViewInPhaseV (int nPAVNo)`

PAV2250A_SetViewInPhaseV sends the command to set InPhase units to V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.35 `_PAV2250AFUNC int PAV2250A_SetViewMain180 (int nPAVNo)`

PAV2250A_SetViewMain180 sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.36 `_PAV2250AFUNC int PAV2250A_SetViewMain360 (int nPAVNo)`

PAV2250A_SetViewMain360 sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.37 `_PAV2250AFUNC` int `PAV2250A_SetViewMainDB` (int *nPAVNo*)

`PAV2250A_SetViewMainDB` sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.38 `_PAV2250AFUNC int PAV2250A_SetViewMainMV (int nPAVNo)`

PAV2250A_SetViewMainMV sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.39 `_PAV2250AFUNC int PAV2250A_SetViewMainPercent (int nPAVNo)`

PAV2250A_SetViewMainPercent sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.40 `_PAV2250AFUNC` int `PAV2250A_SetViewMainRatio` (int *nPAVNo*)

`PAV2250A_SetViewMainRatio` sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.41 `_PAV2250AFUNC int PAV2250A_SetViewMainV (int nPAVNo)`

PAV2250A_SetViewMainV sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.42 `_PAV2250AFUNC int PAV2250A_SetViewPhase180 (int nPAVNo)`

PAV2250A_SetViewPhase180 sends the command to set Phase units to +-180.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.43 `_PAV2250AFUNC int PAV2250A_SetViewPhase360 (int nPAVNo)`

PAV2250A_SetViewPhase360 sends the command to set Phase units to +-360.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.44 `_PAV2250AFUNC int PAV2250A_SetViewQuadDB (int nPAVNo)`

PAV2250A_SetViewQuadDB sends the command to set Quad units to dB.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.45 `_PAV2250AFUNC int PAV2250A_SetViewQuadMV (int nPAVNo)`

PAV2250A_SetViewQuadMV sends the command to set Quad units to mV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.46 `_PAV2250AFUNC int PAV2250A_SetViewQuadPercent (int nPAVNo)`

PAV2250A_SetViewQuadPercent sends the command to set Quad units to %.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.47 `_PAV2250AFUNC int PAV2250A_SetViewQuadRatio (int nPAVNo)`

PAV2250A_SetViewQuadRatio sends the command to set Quad units to Ratio.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.48 `_PAV2250AFUNC int PAV2250A_SetViewQuadV (int nPAVNo)`

PAV2250A_SetViewQuadV sends the command to set Quad units to V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.49 `_PAV2250AFUNC int PAV2250A_SetViewRefVoltMV (int nPAVNo)`

PAV2250A_SetViewRefVoltMV sends the command to set Reference Voltage units to mV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.50 `_PAV2250AFUNC` int PAV2250A_SetViewRefVoltV (int *nPAVNo*)

PAV2250A_SetViewRefVoltV sends the command to set Reference Voltage units to V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.51 `_PAV2250AFUNC` int PAV2250A_SetViewSigOffsetMV (int *nPAVNo*)

PAV2250A_SetViewSigOffsetMV sends the command to set Signal Offset units to mV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.52 `_PAV2250AFUNC` int PAV2250A_SetViewSigOffsetV (int *nPAVNo*)

PAV2250A_SetViewSigOffsetV sends the command to set Signal Offset units to V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.53 `_PAV2250AFUNC` int PAV2250A_SetViewSigVoltMV (int *nPAVNo*)

PAV2250A_SetViewSigVoltMV sends the command to set Signal Voltage units to mV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.54 `_PAV2250AFUNC` int PAV2250A_SetViewSigVoltV (int *nPAVNo*)

PAV2250A_SetViewSigVoltV sends the command to set Signal Voltage units to V.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.55 `_PAV2250AFUNC` int PAV2250A_SetViewTHDDB (int *nPAVNo*)

PAV2250A_SetViewTHDDB sends the command to set THD units to dB.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.56 `_PAV2250AFUNC int PAV2250A_SetViewTHDPercent (int nPAVNo)`

PAV2250A_SetViewTHDPercent sends the command to set THD units to %.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.57 `_PAV2250AFUNC int PAV2250A_SetViewTotalRatioDB (int nPAVNo)`

PAV2250A_SetViewTotalRatioDB sends the command to set Total Ratio units to dB.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.58 `_PAV2250AFUNC int PAV2250A_SetViewTotalRatioPercent (int nPAVNo)`

PAV2250A_SetViewTotalRatioPercent sends the command to set Total Ratio units to %.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.14.2.59 `_PAV2250AFUNC int PAV2250A_SetViewTotalRatioRatio (int nPAVNo)`

PAV2250A_SetViewTotalRatioRatio sends the command to set Total Ratio units to Ratio.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.15 View Offset Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagOffset](#) (int nPAVNo, float fOffset)
PAV2250A_SetViewFundMagOffset sends the command to set the offset for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagOffset](#) (int nPAVNo, float *pfOffset)
PAV2250A_GetViewFundMagOffset sends the command to get the offset for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseOffset](#) (int nPAVNo, float fOffset)
PAV2250A_SetViewInPhaseOffset sends the command to set the offset for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseOffset](#) (int nPAVNo, float *pfOffset)
PAV2250A_GetViewInPhaseOffset sends the command to get the offset for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadOffset](#) (int nPAVNo, float fOffset)
PAV2250A_SetViewQuadOffset sends the command to set the offset for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadOffset](#) (int nPAVNo, float *pfOffset)
PAV2250A_GetViewQuadOffset sends the command to get the offset for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhaseOffset](#) (int nPAVNo, float fOffset)
PAV2250A_SetViewPhaseOffset sends the command to set the offset for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseOffset](#) (int nPAVNo, float *pfOffset)
PAV2250A_GetViewPhaseOffset sends the command to get the offset for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltOffset](#) (int nPAVNo, float fOffset)
PAV2250A_SetViewSigVoltOffset sends the command to set the offset for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltOffset](#) (int nPAVNo, float *pfOffset)
PAV2250A_GetViewSigVoltOffset sends the command to get the offset for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltOffset](#) (int nPAVNo, float fOffset)
PAV2250A_SetViewRefVoltOffset sends the command to set the offset for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltOffset](#) (int nPAVNo, float *pfOffset)
PAV2250A_GetViewRefVoltOffset sends the command to get the offset for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainOffset](#) (int nPAVNo, float fOffset)
PAV2250A_SetViewMainOffset sends the command to set the offset for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainOffset](#) (int nPAVNo, float *pfOffset)
PAV2250A_GetViewMainOffset sends the command to get the offset for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")

3.15.1 Detailed Description

3.15.2 Function Documentation

3.15.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagOffset](#) (int nPAVNo, float * pfOffset)

[PAV2250A_GetViewFundMagOffset](#) sends the command to get the offset for the Fundamental Magnitude view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfOffset</i>	: (Output) Offset currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.15.2.2 `_PAV2250AFUNC int PAV2250A_GetViewInPhaseOffset (int nPAVNo, float * pfOffset)`

PAV2250A_GetViewInPhaseOffset sends the command to get the offset for the In Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfOffset</i>	: (Output) Offset currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.15.2.3 `_PAV2250AFUNC int PAV2250A_GetViewMainOffset (int nPAVNo, float * pfOffset)`

PAV2250A_GetViewMainOffset sends the command to get the offset for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfOffset</i>	: (Output) Offset currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.15.2.4 `_PAV2250AFUNC int PAV2250A_GetViewPhaseOffset (int nPAVNo, float * pfOffset)`

PAV2250A_GetViewPhaseOffset sends the command to get the offset for the Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfOffset</i>	: (Output) Offset currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.15.2.5 `_PAV2250AFUNC int PAV2250A_GetViewQuadOffset (int nPAVNo, float * pfOffset)`

PAV2250A_GetViewQuadOffset sends the command to get the offset for the Quad view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfOffset</i>	: (Output) Offset currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.15.2.6 `_PAV2250AFUNC int PAV2250A_GetViewRefVoltOffset (int nPAVNo, float * pfOffset)`

PAV2250A_GetViewRefVoltOffset sends the command to get the offset for the RefVolt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfOffset</i>	: (Output) Offset currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.15.2.7 `_PAV2250AFUNC int PAV2250A_GetViewSigVoltOffset (int nPAVNo, float * pfOffset)`

PAV2250A_GetViewSigVoltOffset sends the command to get the offset for the SigVolt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfOffset</i>	: (Output) Offset currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.15.2.8 _PAV2250AFUNC int PAV2250A_SetViewFundMagOffset (int nPAVNo, float fOffset)

PAV2250A_SetViewFundMagOffset sends the command to set the offset for the Fundamental Magnitude view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fOffset</i>	: (Input) Desired Offset to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.15.2.9 _PAV2250AFUNC int PAV2250A_SetViewInPhaseOffset (int nPAVNo, float fOffset)

PAV2250A_SetViewInPhaseOffset sends the command to set the offset for the In Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fOffset</i>	: (Input) Desired Offset to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.15.2.10 `_PAV2250AFUNC` int `PAV2250A_SetViewMainOffset` (int *nPAVNo*, float *fOffset*)

`PAV2250A_SetViewMainOffset` sends the command to set the offset for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fOffset</i>	: (Input) Desired Offset to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.15.2.11 `_PAV2250AFUNC int PAV2250A_SetViewPhaseOffset (int nPAVNo, float fOffset)`

PAV2250A_SetViewPhaseOffset sends the command to set the offset for the Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fOffset</i>	: (Input) Desired Offset to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.15.2.12 `_PAV2250AFUNC int PAV2250A_SetViewQuadOffset (int nPAVNo, float fOffset)`

PAV2250A_SetViewQuadOffset sends the command to set the offset for the Quad view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fOffset</i>	: (Input) Desired Offset to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.15.2.13 `_PAV2250AFUNC` int `PAV2250A_SetViewRefVoltOffset` (int *nPAVNo*, float *fOffset*)

`PAV2250A_SetViewRefVoltOffset` sends the command to set the offset for the Ref Volt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fOffset</i>	: (Input) Desired Offset to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.15.2.14 `_PAV2250AFUNC` int PAV2250A_SetViewSigVoltOffset (int *nPAVNo*, float *fOffset*)

PAV2250A_SetViewSigVoltOffset sends the command to set the offset for the Sig Volt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fOffset</i>	: (Input) Desired Offset to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.16 View Scale Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewFundMagScale sends the command to set the scale for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewFundMagScale sends the command to get the scale for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewInPhaseScale sends the command to set the scale for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewInPhaseScale sends the command to get the scale for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewQuadScale sends the command to set the scale for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewQuadScale sends the command to get the scale for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhaseScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewPhaseScale sends the command to set the scale for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewPhaseScale sends the command to get the scale for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewSigVoltScale sends the command to set the scale for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewSigVoltScale sends the command to get the scale for the SigVolt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewRefVoltScale sends the command to set the scale for the RefVolt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewRefVoltScale sends the command to get the scale for the RefVolt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewMainScale sends the command to set the scale for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewMainScale sends the command to get the scale for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")

3.16.1 Detailed Description

3.16.2 Function Documentation

3.16.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagScale](#) (int nPAVNo, float * pfScale)

[PAV2250A_GetViewFundMagScale](#) sends the command to get the scale for the Fundamental Magnitude view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfScale</i>	: (Output) Scale currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.16.2.2 `_PAV2250AFUNC int PAV2250A_GetViewInPhaseScale (int nPAVNo, float * pfScale)`

PAV2250A_GetViewInPhaseScale sends the command to get the scale for the In Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfScale</i>	: (Output) Scale currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.16.2.3 `_PAV2250AFUNC int PAV2250A_GetViewMainScale (int nPAVNo, float * pfScale)`

PAV2250A_GetViewMainScale sends the command to get the scale for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfScale</i>	: (Output) Scale currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.16.2.4 `_PAV2250AFUNC int PAV2250A_GetViewPhaseScale (int nPAVNo, float * pfScale)`

PAV2250A_GetViewPhaseScale sends the command to get the scale for the Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfScale</i>	: (Output) Scale currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.16.2.5 `_PAV2250AFUNC int PAV2250A_GetViewQuadScale (int nPAVNo, float * pfScale)`

PAV2250A_GetViewQuadScale sends the command to get the scale for the Quad view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfScale</i>	: (Output) Scale currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.16.2.6 `_PAV2250AFUNC int PAV2250A_GetViewRefVoltScale (int nPAVNo, float * pfScale)`

PAV2250A_GetViewRefVoltScale sends the command to get the scale for the RefVolt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfScale</i>	: (Output) Scale currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.16.2.7 `_PAV2250AFUNC int PAV2250A_GetViewSigVoltScale (int nPAVNo, float * pfScale)`

`PAV2250A_GetViewSigVoltScale` sends the command to get the scale for the SigVolt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfScale</i>	: (Output) Scale currently configured for this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.16.2.8 _PAV2250AFUNC int PAV2250A_SetViewFundMagScale (int nPAVNo, float fScale)

PAV2250A_SetViewFundMagScale sends the command to set the scale for the Fundamental Magnitude view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fScale</i>	: (Input) Desired Scale to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.16.2.9 _PAV2250AFUNC int PAV2250A_SetViewInPhaseScale (int nPAVNo, float fScale)

PAV2250A_SetViewInPhaseScale sends the command to set the scale for the In Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fScale</i>	: (Input) Desired Scale to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.16.2.10 `_PAV2250AFUNC` int `PAV2250A_SetViewMainScale` (int *nPAVNo*, float *fScale*)

`PAV2250A_SetViewMainScale` sends the command to set the scale for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fScale</i>	: (Input) Desired Scale to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.16.2.11 `_PAV2250AFUNC int PAV2250A_SetViewPhaseScale (int nPAVNo, float fScale)`

PAV2250A_SetViewPhaseScale sends the command to set the scale for the Phase view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fScale</i>	: (Input) Desired Scale to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.16.2.12 `_PAV2250AFUNC int PAV2250A_SetViewQuadScale (int nPAVNo, float fScale)`

PAV2250A_SetViewQuadScale sends the command to set the scale for the Quad view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fScale</i>	: (Input) Desired Scale to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.16.2.13 `_PAV2250AFUNC` int `PAV2250A_SetViewRefVoltScale` (int *nPAVNo*, float *fScale*)

`PAV2250A_SetViewRefVoltScale` sends the command to set the scale for the RefVolt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fScale</i>	: (Input) Desired Scale to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.16.2.14 `_PAV2250AFUNC int PAV2250A_SetViewSigVoltScale (int nPAVNo, float fScale)`

PAV2250A_SetViewSigVoltScale sends the command to set the scale for the Sig Volt view.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fScale</i>	: (Input) Desired Scale to be applied to this view

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.17 View by Index Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexMaxFieldWidth](#) (int nPAVNo, int nViewIndex, int nMaxFieldWidth)

PAV2250A_SetViewIndexMaxFieldWidth sends the command to set the max field width for the specified view index.

- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexMaxFieldWidth](#) (int nPAVNo, int nViewIndex, int *pnMaxFieldWidth)

PAV2250A_GetViewIndexMaxFieldWidth sends the command to get the max field width for the specified view index.

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexUnits](#) (int nPAVNo, int nViewIndex, int nUnits)

PAV2250A_SetViewIndexUnits sends the command to set the units for the specified view index.

- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexUnits](#) (int nPAVNo, int nViewIndex, int *pnUnits)

PAV2250A_GetViewIndexUnits sends the command to get the units for the specified view index.

- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexUnitsText](#) (int nPAVNo, int nViewIndex, char *pszUnits)

PAV2250A_GetViewIndexUnitsText sends the command to get the view units for the view text specified by the index.

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexOffset](#) (int nPAVNo, int nViewIndex, float fOffset)

PAV2250A_SetViewIndexOffset sends the command to set the offset for the specified view index.

- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexOffset](#) (int nPAVNo, int nViewIndex, float *pfOffset)

PAV2250A_GetViewIndexOffset sends the command to get the offset for the specified view index.

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexScale](#) (int nPAVNo, int nViewIndex, float fScale)

PAV2250A_SetViewIndexScale sends the command to set the scale for the specified view index.

- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexScale](#) (int nPAVNo, int nViewIndex, float *pfScale)

PAV2250A_GetViewIndexScale sends the command to get the scale for the specified view index.

3.17.1 Detailed Description

3.17.2 Function Documentation

- #### 3.17.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexMaxFieldWidth](#) (int nPAVNo, int nViewIndex, int * pnMaxFieldWidth)

PAV2250A_GetViewIndexMaxFieldWidth sends the command to get the max field width for the specified view index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to get max field width <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"
<i>pnMaxField-Width</i>	: (Output) Max Field Width this view is currently configured with

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.17.2.2 `_PAV2250AFUNC int PAV2250A_GetViewIndexOffset (int nPAVNo, int nViewIndex, float * pfOffset)`

PAV2250A_GetViewIndexOffset sends the command to get the offset for the specified view index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to get the offset for <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"
<i>pfOffset</i>	: (Output) Offset value to the specified view is currently configured with

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.17.2.3 `_PAV2250AFUNC int PAV2250A_GetViewIndexScale (int nPAVNo, int nViewIndex, float * pfScale)`

PAV2250A_GetViewIndexScale sends the command to get the scale for the specified view index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to get the scale for <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"
<i>pfScale</i>	: (Output) Scale value to the specified view is currently configured with

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.17.2.4 `_PAV2250AFUNC int PAV2250A_GetViewIndexUnits (int nPAVNo, int nViewIndex, int * pnUnits)`

PAV2250A_GetViewIndexUnits sends the command to get the units for the specified view index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to get Units <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"
<i>pnUnits</i>	: (Output) Resolution this view is currently configured with <ul style="list-style-type: none"> • 0 : "V" • 1 : "mV" • 2 : "Ratio" • 3 : "Percent" • 4 : "DB" • 5 : "360" • 6 : "180" • 7 : "Hz" • 8 : "KHz"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.17.2.5 _PAV2250AFUNC int PAV2250A_GetViewIndexUnitsText (int nPAVNo, int nViewIndex, char * pszUnits)

PAV2250A_GetViewIndexUnitsText sends the command to get the view units for the view text specified by the index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to get Units Text <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"
<i>pszUnits</i>	: (Output) Units Text the specified view is configured with <ul style="list-style-type: none"> • V • mV • Ratio • Percent • DB • 360 • 180 • Hz • KHz

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.17.2.6 `_PAV2250AFUNC int PAV2250A_SetViewIndexMaxFieldWidth (int nPAVNo, int nViewIndex, int nMaxFieldWidth)`

PAV2250A_SetViewIndexMaxFieldWidth sends the command to set the max field width for the specified view index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to set max field width <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"

<i>nMaxFieldWidth</i>	: (Input) Desired max field width (Values 1 - 7)
-----------------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.17.2.7 _PAV2250AFUNC int PAV2250A_SetViewIndexOffset (int nPAVNo, int nViewIndex, float fOffset)

PAV2250A_SetViewIndexOffset sends the command to set the offset for the specified view index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to set the offset <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"

<i>fOffset</i>	: (Input) Offset value to configure the specified view with
----------------	---

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.17.2.8 _PAV2250AFUNC int PAV2250A_SetViewIndexScale (int nPAVNo, int nViewIndex, float fScale)

PAV2250A_SetViewIndexScale sends the command to set the scale for the specified view index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to set the scale <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"

<i>fScale</i>	: (Input) Scale value to configure the specified view with
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.17.2.9 _PAV2250AFUNC int PAV2250A_SetViewIndexUnits (int nPAVNo, int nViewIndex, int nUnits)

PAV2250A_SetViewIndexUnits sends the command to set the units for the specified view index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Index of view to set Units <ul style="list-style-type: none"> • 0 : "Fund Mag" • 1 : "In Phase" • 2 : "Quad" • 3 : "Phase" • 4 : "THD" • 5 : "Sig Volt" • 6 : "Ref Volt" • 7 : "Sig Offset" • 8 : "Total Ratio" • 9 : "Freq" • 10 : "Main"

<i>nUnits</i>	<p>: (Input) Desired units.</p> <ul style="list-style-type: none">• 0 : "V"• 1 : "mV"• 2 : "Ratio"• 3 : "Percent"• 4 : "DB"• 5 : "360"• 6 : "180"• 7 : "Hz"• 8 " "KHz"
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.18 Reference Voltage Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetRef](#) (int nPAVNo, int *pnViewIndex)
PAV2250A_GetRef sends the command requesting the current Ref Voltage View Index to the PAV2250A device. Index options (1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefText](#) (int nPAVNo, char *pszViewText)
PAV2250A_GetRef sends the command requesting the current Ref Voltage View Text to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRef](#) (int nPAVNo, int nViewIndex)
PAV2250A_SetRef sends the command to set the Ref Voltage View Index to the PAV2250A device. Index options (1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalSum](#) (int nPAVNo)
PAV2250A_SetRefTotalSum sends the command to set the Ref Voltage View to TotalSum to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalRMS_AC](#) (int nPAVNo)
PAV2250A_SetRefTotalRMS_AC
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalRMS_ACDC](#) (int nPAVNo)
PAV2250A_SetRefTotalRMS_ACDC sends the command to set the Ref Voltage View to TotalRMS_ACDC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefDC](#) (int nPAVNo)
PAV2250A_SetRefDC sends the command to set the Ref Voltage View to DC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalSum](#) (int nPAVNo, float *pfRefTotalSum)
PAV2250A_GetRefTotalSum sends the command to get the Ref Voltage Total Sum value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalRMS_AC](#) (int nPAVNo, float *pfRefRMS)
PAV2250A_GetRefTotalRMS_AC sends the command to get the Ref Voltage Total RMS AC value from the PAV2250-A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalRMS_ACDC](#) (int nPAVNo, float *pfRefRMS_ACDC)
PAV2250A_GetRefTotalRMS_ACDC sends the command to get the Ref Voltage Total RMS AC+DC value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefDC](#) (int nPAVNo, float *pfRefDC)
PAV2250A_GetRefDC sends the command to get the Ref Voltage DC value from the PAV2250A device.

3.18.1 Detailed Description

3.18.2 Function Documentation

3.18.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetRef](#) (int nPAVNo, int * pnViewIndex)

[PAV2250A_GetRef](#) sends the command requesting the current Ref Voltage View Index to the PAV2250A device. Index options (1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnViewIndex</i>	: (Output) Returns the current configured RefVolt view. Valid Values: <ul style="list-style-type: none"> • 1 : "Ref Total RMS (AC)" • 2 : "Ref Total RMS (AC+DC)" • 3 : "Ref DC"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.18.2.2 `_PAV2250AFUNC int PAV2250A_GetRefDC (int nPAVNo, float * pfRefDC)`

PAV2250A_GetRefDC sends the command to get the Ref Voltage DC value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfRefDC</i>	: (Output) Reference DC value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.18.2.3 `_PAV2250AFUNC int PAV2250A_GetRefText (int nPAVNo, char * pszViewText)`

PAV2250A_GetRef sends the command requesting the current Ref Voltage View Text to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewText</i>	: (Output) Returns the current configured RefVolt view text. Valid Values: <ul style="list-style-type: none"> • Ref Total RMS (AC) • Ref Total RMS (AC+DC) • Ref DC

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.18.2.4 `_PAV2250AFUNC` int `PAV2250A_GetRefTotalRMS_AC` (int *nPAVNo*, float * *pfRefRMS*)

`PAV2250A_GetRefTotalRMS_AC` sends the command to get the Ref Voltage Total RMS AC value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfRefRMS</i>	: (Output) Reference RMS value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.18.2.5 `_PAV2250AFUNC int PAV2250A_GetRefTotalRMS_ACDC (int nPAVNo, float * pfRefRMS_ACDC)`

PAV2250A_GetRefTotalRMS_ACDC sends the command to get the Ref Voltage Total RMS AC+DC value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfRefRMS_AC-DC</i>	: (Output) Reference RMS AC+DC value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.18.2.6 `_PAV2250AFUNC int PAV2250A_GetRefTotalSum (int nPAVNo, float * pfRefTotalSum)`

PAV2250A_GetRefTotalSum sends the command to get the Ref Voltage Total Sum value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfRefTotalSum</i>	: (Output) Reference Total Sum value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.18.2.7 `_PAV2250AFUNC` int `PAV2250A_SetRef` (int *nPAVNo*, int *nViewIndex*)

`PAV2250A_SetRef` sends the command to set the Ref Voltage View Index to the PAV2250A device. Index options (1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Desired RefVolt view to configure this view with. Valid Values: <ul style="list-style-type: none"> • 1 : "Ref Total RMS (AC)" • 2 : "Ref Total RMS (AC+DC)" • 3 : "Ref DC"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.18.2.8 `_PAV2250AFUNC int PAV2250A_SetRefDC (int nPAVNo)`

PAV2250A_SetRefDC sends the command to set the Ref Voltage View to DC to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.18.2.9 `_PAV2250AFUNC int PAV2250A_SetRefTotalRMS_AC (int nPAVNo)`

PAV2250A_SetRefTotalRMS_AC

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also[ExecuteRemoteCmd](#)**3.18.2.10 _PAV2250AFUNC int PAV2250A_SetRefTotalRMS_ACDC (int nPAVNo)**

PAV2250A_SetRefTotalRMS_ACDC sends the command to set the Ref Voltage View to TotalRMS_ACDC to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also[ExecuteRemoteCmd](#)**3.18.2.11 _PAV2250AFUNC int PAV2250A_SetRefTotalSum (int nPAVNo)**

PAV2250A_SetRefTotalSum sends the command to set the Ref Voltage View to TotalSum to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also[ExecuteRemoteCmd](#)

3.19 Signal Voltage Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetSig](#) (int nPAVNo, int *pnViewIndex)
PAV2250A_GetSig sends the command requesting the current Sig Voltage View Index to the PAV2250A device. Index options (0=TotalSum 1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigText](#) (int nPAVNo, char *pszViewText)
PAV2250A_GetSig sends the command requesting the current Sig Voltage View Text to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSig](#) (int nPAVNo, int nViewIndex)
PAV2250A_SetSig sends the command to set the Sig Voltage View Index to the PAV2250A device. Index options (0=TotalSum, 1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalSum](#) (int nPAVNo)
PAV2250A_SetSigTotalSum sends the command to set the Sig Voltage View to TotalSum to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalRMS_AC](#) (int nPAVNo)
PAV2250A_SetSigTotalRMS_AC sends the command to set the Sig Voltage View to TotalRMS_AC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalRMS_ACDC](#) (int nPAVNo)
PAV2250A_SetSigTotalRMS_ACDC sends the command to set the Sig Voltage View to TotalRMS_ACDC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigDC](#) (int nPAVNo)
PAV2250A_SetSigDC sends the command to set the Sig Voltage View to DC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalSum](#) (int nPAVNo, float *pfSigTotalSum)
PAV2250A_GetSigTotalSum sends the command to get the Sig Voltage Total Sum value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalRMS_AC](#) (int nPAVNo, float *pfSigRMS)
PAV2250A_GetSigTotalRMS_AC sends the command to get the Sig Voltage Total RMS AC value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalRMS_ACDC](#) (int nPAVNo, float *pfSigRMS_ACDC)
PAV2250A_GetSigTotalRMS_ACDC sends the command to get the Sig Voltage Total RMS AC+DC value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigDC](#) (int nPAVNo, float *pfSigDC)
PAV2250A_GetSigDC sends the command to get the Sig Voltage DC value from the PAV2250A device.

3.19.1 Detailed Description

3.19.2 Function Documentation

3.19.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetSig](#) (int nPAVNo, int * pnViewIndex)

[PAV2250A_GetSig](#) sends the command requesting the current Sig Voltage View Index to the PAV2250A device. Index options (0=TotalSum 1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnViewIndex</i>	: (Output) Returns the current configured SigVolt view. Valid Values: <ul style="list-style-type: none"> • 0 : "Sig Total Sum" • 1 : "Sig Total RMS (AC)" • 2 : "Sig Total RMS (AC+DC)" • 3 : "Sig DC"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.19.2.2 `_PAV2250AFUNC int PAV2250A_GetSigDC (int nPAVNo, float * pfSigDC)`

PAV2250A_GetSigDC sends the command to get the Sig Voltage DC value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfSigDC</i>	: (Output) Signal DC value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.19.2.3 `_PAV2250AFUNC int PAV2250A_GetSigText (int nPAVNo, char * pszViewText)`

PAV2250A_GetSig sends the command requesting the current Sig Voltage View Text to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszViewText</i>	: (Output) Returns the current configured SigVolt view text. Valid Values: <ul style="list-style-type: none"> • Sig Total Sum • Sig Total RMS (AC) • Sig Total RMS (AC+DC) • Sig DC

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.19.2.4 `_PAV2250AFUNC int PAV2250A_GetSigTotalRMS_AC (int nPAVNo, float * pfSigRMS)`

`PAV2250A_GetSigTotalRMS_AC` sends the command to get the Sig Voltage Total RMS AC value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfSigRMS</i>	: (Output) Signal Total RMS AC value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.19.2.5 `_PAV2250AFUNC int PAV2250A_GetSigTotalRMS_ACDC (int nPAVNo, float * pfSigRMS_ACDC)`

PAV2250A_GetSigTotalRMS_ACDC sends the command to get the Sig Voltage Total RMS AC+DC value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfSigRMS_ACD-C</i>	: (Output) Signal Total RMS AC+DC value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.19.2.6 `_PAV2250AFUNC int PAV2250A_GetSigTotalSum (int nPAVNo, float * pfSigTotalSum)`

PAV2250A_GetSigTotalSum sends the command to get the Sig Voltage Total Sum value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfSigTotalSum</i>	: (Output) Signal Total Sum value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.19.2.7 `_PAV2250AFUNC int PAV2250A_SetSig (int nPAVNo, int nViewIndex)`

PAV2250A_SetSig sends the command to set the Sig Voltage View Index to the PAV2250A device. Index options (0=TotalSum, 1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nViewIndex</i>	: (Input) Desired SigVolt view to configure this view with. Valid Values: <ul style="list-style-type: none"> • 0 : "Sig Total Sum" • 1 : "Sig Total RMS (AC)" • 2 : "Sig Total RMS (AC+DC)" • 3 : "Sig DC"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.19.2.8 `_PAV2250AFUNC int PAV2250A_SetSigDC (int nPAVNo)`

PAV2250A_SetSigDC sends the command to set the Sig Voltage View to DC to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.19.2.9 `_PAV2250AFUNC int PAV2250A_SetSigTotalRMS_AC (int nPAVNo)`

PAV2250A_SetSigTotalRMS_AC sends the command to set the Sig Voltage View to TotalRMS_AC to the PAV2250-A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.19.2.10 _PAV2250AFUNC int PAV2250A_SetSigTotalRMS_ACDC (int nPAVNo)

PAV2250A_SetSigTotalRMS_ACDC sends the command to set the Sig Voltage View to TotalRMS_ACDC to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.19.2.11 _PAV2250AFUNC int PAV2250A_SetSigTotalSum (int nPAVNo)

PAV2250A_SetSigTotalSum sends the command to set the Sig Voltage View to TotalSum to the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.20 Independent Component Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetSigOffset](#) (int nPAVNo, float *pfSigOffset)
PAV2250A_GetSigOffset sends the command to get the Sig Offset value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTotalRatio](#) (int nPAVNo, float *pfTotalRatio)
PAV2250A_GetTotalRatio sends the command to get the Total Ratio value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTHD](#) (int nPAVNo, float *pfTHD)
PAV2250A_GetTHD sends the command to get the THD value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetFrequency](#) (int nPAVNo, float *pfFreq)
PAV2250A_GetFrequency sends the command to get the Frequency value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSampleRateIndex](#) (int nPAVNo, int *pnSampleRateIndex)
PAV2250A_GetSampleRateIndex sends the command to get the Sample Rate Index value from the PAV2250A device.

3.20.1 Detailed Description

3.20.2 Function Documentation

3.20.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetFrequency](#) (int nPAVNo, float * pfFreq)

[PAV2250A_GetFrequency](#) sends the command to get the Frequency value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfFreq</i>	: (Output) Frequency value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.20.2.2 [_PAV2250AFUNC](#) int [PAV2250A_GetSampleRateIndex](#) (int nPAVNo, int * pnSampleRateIndex)

[PAV2250A_GetSampleRateIndex](#) sends the command to get the Sample Rate Index value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnSampleRate-Index</i>	: (Output) Sample Rate Index value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.20.2.3 `_PAV2250AFUNC int PAV2250A_GetSigOffset (int nPAVNo, float * pfSigOffset)`

PAV2250A_GetSigOffset sends the command to get the Sig Offset value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfSigOffset</i>	: (Output) Signal Offset value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.20.2.4 `_PAV2250AFUNC int PAV2250A_GetTHD (int nPAVNo, float * pfTHD)`

PAV2250A_GetTHD sends the command to get the THD value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfTHD</i>	: (Output) THD value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.20.2.5 `_PAV2250AFUNC int PAV2250A_GetTotalRatio (int nPAVNo, float * pfTotalRatio)`

PAV2250A_GetTotalRatio sends the command to get the Total Ratio value from the PAV2250A device.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfTotalRatio</i>	: (Output) Total Ratio value

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.21 Data Buffer Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedDataState](#) (int nPAVNo, bool *pbDataBufferEnabled, bool *pbDataBufferComplete, int *pnBufferedDataType, int *pnSampleRate, int *pnBufferSize, bool *pbDataBufferReady)

FOR INTERNAL USE ONLY! PAV2250A_GetBufferedDataState is responsible for returning data buffering state information.
- [_PAV2250AFUNC](#) int [PAV2250A_SetBufferedDataState](#) (int nPAVNo, bool bDataBufferEnabled, int nBufferedDataType, int nSampleRate, int nBufferSize)

FOR INTERNAL USE ONLY! PAV2250A_SetBufferedDataState is responsible for setting the buffer collection parameters of DataType, SampleRate, BufferSize and BufferEnabled.
- [_PAV2250AFUNC](#) int [PAV2250A_BufferCapture](#) (int nPAVNo, int nDataType, int nSampleRate, int nSize)

FOR INTERNAL USE ONLY! PAV2250A_BufferCapture is responsible for Starting a data buffer capture.
- [_PAV2250AFUNC](#) int [PAV2250A_BufferStop](#) (int nPAVNo)

FOR INTERNAL USE ONLY! PAV2250A_BufferStop is responsible for stopping all buffer captures for the given PAV.
- [_PAV2250AFUNC](#) int [PAV2250A_BufferGet](#) (int nPAVNo, int nStart, int nCount, char *pszData)

FOR INTERNAL USE ONLY! PAV2250A_BufferGet is responsible for retrieving a captured buffer.
- [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedPageIndex](#) (int nPAVNo, int *pnPageIndex)

FOR INTERNAL USE ONLY! PAV2250A_GetBufferedPageIndex returns the current buffered page index.
- [_PAV2250AFUNC](#) int [PAV2250A_SetBufferedPageIndex](#) (int nPAVNo, int nPageIndex)

FOR INTERNAL USE ONLY! PAV2250A_SetBufferedPageIndex is responsible for setting the current page buffer.
- [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedData](#) (int nPAVNo, int nPageIndex, int nRequestedBufferSize, int nTotalOverallDataCnt, int *pnElementCnt, int *paBufferData)

FOR INTERNAL USE ONLY! PAV2250A_GetBufferedData is responsible for retrieving the buffered data.
- int [HandleBufferedDataState](#) (int nPAVNo, bool *pbDataBufferEnabled, bool *pbDataBufferComplete, int *pnBufferedDataType, int *pnSampleRate, int *pnBufferSize, bool *pbDataBufferReady)

FOR INTERNAL USE ONLY! HandleBufferedDataState returns Data Buffering configuration information.
- int [HandleBufferDataSetup](#) (int nPAVNo, int nStartRec, int nEndRec)

FOR INTERNAL USE ONLY! HandleBufferDataSetup is responsible for setting some parameters needed to capture data.
- int [HandleBufferedDataValues](#) (int nPAVNo, int nStartRec, int *pnDataCnt, int *paBufferData)

FOR INTERNAL USE ONLY! HandleBufferedDataValues is responsible for loading the buffered data values into the output buffer.

3.21.1 Detailed Description

3.21.2 Function Documentation

3.21.2.1 int HandleBufferDataSetup (int nPAVNo, int nStartRec, int nEndRec)

FOR INTERNAL USE ONLY! HandleBufferDataSetup is responsible for setting some parameters needed to capture data.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nStartRec</i>	: (Input) Start record
<i>nEndRec</i>	: (Input) End record

Returns

- PAV_SUCCESS : Function is successful

- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.21.2.2 `int HandleBufferedDataState (int nPAVNo, bool * pbDataBufferEnabled, bool * pbDataBufferComplete, int * pnBufferedDataType, int * pnSampleRate, int * pnBufferSize, bool * pbDataBufferReady)`

FOR INTERNAL USE ONLY! HandleBufferedDataState returns Data Buffering configuration information.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pbDataBuffer-Enabled</i>	: (Output) True if Data Buffer is Enabled; False otherwise
<i>pbDataBuffer-Complete</i>	: (Output) True if Data Buffering has completed; False otherwise
<i>pnBufferedData-Type</i>	: (Output) <ul style="list-style-type: none"> • 0 : "MOD3 SigCos" • 1 : "MOD2 RefCos" • 2 : "MOD2 RefSig" • 3 : "MOD2 SigCos" • 4 : "MOD3 RefCos" • 5 : "MOD3 RefSin"
<i>pnSampleRate</i>	: (Output) <ul style="list-style-type: none"> • -1 : "Auto Sample Rate" • 0 : "2.5 MHz_0" • 1 : "2.5 MHz_1" • 2 : "2.5 MHz_2" • 3 : "2.5 MHz_3" • 4 : "2.5 MHz_4" • 5 : "1.25 MHz" • 6 : "625 kHz" • 7 : "312.5 kHz" • 8 : "156.25 kHz" • 9 : "78.125 kHz" • 10 : "39.0625 kHz" • 11 : "19.53125 kHz" • 12 : "9.765625 kHz" • 13 : "4.8828125 kHz" • 14 : "2.44140625 kHz" • 15 : "1.220703125 kHz" • 16 : "610.6515625 Hz" • 17 : "305.17578125 Hz" • 18 : "152.587890625 Hz" • 19 : "76.2939453125 Hz" • 20 : "38.1469726525 Hz" • 21 : "19.07348632625 Hz" • 22 : "9.536743163125 Hz"
<i>pnBufferSize</i>	: (Output) Size of buffer

<i>pbDataBuffer-Ready</i>	: (Output) True if data buffer is ready; False otherwise
---------------------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.21.2.3 int HandleBufferedDataValues (int nPAVNo, int nStartRec, int * pnDataCnt, int * paBufferData)

FOR INTERNAL USE ONLY! HandleBufferedDataValues is responsible for loading the buffered data values into the output buffer.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>nStartRec</i>	: (Input)
<i>pnDataCnt</i>	: (Output)
<i>paBufferData</i>	: (Output)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ParseForCommaSeparatedDataElements](#)

3.21.2.4 `_PAV2250AFUNC` `int PAV2250A_BufferCapture (int nPAVNo, int nDataType, int nSampleRate, int nSize)`

FOR INTERNAL USE ONLY! `PAV2250A_BufferCapture` is responsible for Starting a data buffer capture.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nDataType</i>	: (Input) Type of data buffering to conduct: <ul style="list-style-type: none"> • 0 : "MOD3 SigCos" • 1 : "MOD2 RefCos" • 2 : "MOD2 RefSig" • 3 : "MOD2 SigCos" • 4 : "MOD3 RefCos" • 5 : "MOD3 RefSin"
<i>nSampleRate</i>	: (Input) Rate at which data is buffered <ul style="list-style-type: none"> • -1 : "Auto Sample Rate" • 0 : "2.5 MHz_0" • 1 : "2.5 MHz_1" • 2 : "2.5 MHz_2" • 3 : "2.5 MHz_3" • 4 : "2.5 MHz_4" • 5 : "1.25 MHz" • 6 : "625 kHz" • 7 : "312.5 kHz" • 8 : "156.25 kHz" • 9 : "78.125 kHz" • 10 : "39.0625 kHz" • 11 : "19.53125 kHz" • 12 : "9.765625 kHz" • 13 : "4.8828125 kHz" • 14 : "2.44140625 kHz" • 15 : "1.220703125 kHz" • 16 : "610.6515625 Hz" • 17 : "305.17578125 Hz" • 18 : "152.587890625 Hz" • 19 : "76.2939453125 Hz" • 20 : "38.1469726525 Hz" • 21 : "19.07348632625 Hz" • 22 : "9.536743163125 Hz"

<i>nSize</i>	: (Input) Size of the data buffer to collect in Bytes
--------------	---

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.21.2.5 `_PAV2250AFUNC int PAV2250A_BufferGet (int nPAVNo, int nStart, int nCount, char * pszData)`

FOR INTERNAL USE ONLY! PAV2250A_BufferGet is responsible for retrieving a captured buffer.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nStart</i>	: (Input) Byte of which to start retrieving buffer from
<i>nCount</i>	: (Input) Number of bytes to retrieve
<i>pszData</i>	: (Output) Collected data

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.21.2.6 `_PAV2250AFUNC int PAV2250A_BufferStop (int nPAVNo)`

FOR INTERNAL USE ONLY! PAV2250A_BufferStop is responsible for stopping all buffer captures for the given PAV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.21.2.7 `_PAV2250AFUNC int PAV2250A_GetBufferedData (int nPAVNo, int nPageIndex, int nRequestedBufferSize, int nTotalOverallDataCnt, int * pnElementCnt, int * paBufferData)`

FOR INTERNAL USE ONLY! PAV2250A_GetBufferedData is responsible for retrieving the buffered data.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nPageIndex</i>	: (Input) Index of buffered page
<i>nRequested-BufferSize</i>	: (Input) Size of buffer to store data
<i>nTotalOverall-DataCnt</i>	: (Input) Total amount of data collected
<i>pnElementCnt</i>	: (Output) Number of elements in the return buffer
<i>paBufferData</i>	: (Output) Requested data

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.21.2.8 `_PAV2250AFUNC int PAV2250A_GetBufferedDataState (int nPAVNo, bool * pbDataBufferEnabled, bool * pbDataBufferComplete, int * pnBufferedDataType, int * pnSampleRate, int * pnBufferSize, bool * pbDataBufferReady)`

FOR INTERNAL USE ONLY! PAV2250A_GetBufferedDataState is responsible for returning data buffering state information.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbDataBuffer-Enabled</i>	: (Output) True if data buffering is enabled; False otherwise
<i>pbDataBuffer-Complete</i>	: (Output) True if data buffering has completed; False otherwise
<i>pnBufferedData-Type</i>	: (Output) Type of data being buffered
<i>pnSampleRate</i>	: (Output) Rate at which data is buffered
<i>pnBufferSize</i>	: (Output) Current size of the data buffer in Bytes
<i>pbDataBuffer-Ready</i>	: (Output) True if the data buffer is ready; False otherwise

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#), [HandleBufferedDataState](#)

3.21.2.9 `_PAV2250AFUNC int PAV2250A_GetBufferedPageIndex (int nPAVNo, int * pnPageIndex)`

FOR INTERNAL USE ONLY! PAV2250A_GetBufferedPageIndex returns the current buffered page index.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnPageIndex</i>	: (Output) Index of buffered page

Returns

- PAV_SUCCESS : Function is successful

- PAV_ERROR_PAVNO : Invalid nPAVNo parameter

- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.21.2.10 `_PAV2250AFUNC int PAV2250A_SetBufferedDataState (int nPAVNo, bool bDataBufferEnabled, int nBufferedDataType, int nSampleRate, int nBufferSize)`

FOR INTERNAL USE ONLY! PAV2250A_SetBufferedDataState is responsible for setting the buffer collection parameters of DataType, SampleRate, BufferSize and BufferEnabled.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>bDataBuffer-Enabled</i>	: (Input) True if data buffering is enabled; False otherwise
<i>nBufferedData-Type</i>	: (Input) Type of data buffering to conduct: <ul style="list-style-type: none"> • 0 : "MOD3 SigCos" • 1 : "MOD2 RefCos" • 2 : "MOD2 RefSig" • 3 : "MOD2 SigCos" • 4 : "MOD3 RefCos" • 5 : "MOD3 RefSin"
<i>nSampleRate</i>	: (Input) Rate at which data is buffered <ul style="list-style-type: none"> • -1 : "Auto Sample Rate" • 0 : "2.5 MHz_0" • 1 : "2.5 MHz_1" • 2 : "2.5 MHz_2" • 3 : "2.5 MHz_3" • 4 : "2.5 MHz_4" • 5 : "1.25 MHz" • 6 : "625 kHz" • 7 : "312.5 kHz" • 8 : "156.25 kHz" • 9 : "78.125 kHz" • 10 : "39.0625 kHz" • 11 : "19.53125 kHz" • 12 : "9.765625 kHz" • 13 : "4.8828125 kHz" • 14 : "2.44140625 kHz" • 15 : "1.220703125 kHz" • 16 : "610.6515625 Hz" • 17 : "305.17578125 Hz" • 18 : "152.587890625 Hz" • 19 : "76.2939453125 Hz" • 20 : "38.1469726525 Hz" • 21 : "19.07348632625 Hz" • 22 : "9.536743163125 Hz"
<i>nBufferSize</i>	: (Input) Size of the data buffer to collect in Bytes

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.21.2.11 `_PAV2250AFUNC` int PAV2250A_SetBufferedPageIndex (int *nPAVNo*, int *nPageIndex*)

FOR INTERNAL USE ONLY! PAV2250A_SetBufferedPageIndex is responsible for setting the current page buffer.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>nPageIndex</i>	: (Index) Index of buffered page

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.22 Internal Reference Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefFreq](#) (int nPAVNo, float fFreq)
PAV2250A_SetIntRefFreq sends a command to the PAV 2250A to force the internal reference frequency to the specified value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefFreq](#) (int nPAVNo, float *pfFreq)
PAV2250A_GetIntRefFreq is responsible for returning internal reference frequency value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefVolt](#) (int nPAVNo, float fVolt)
PAV2250A_SetIntRefVolt sends a command to the PAV 2250A to force the internal reference voltage to the specified value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefVolt](#) (int nPAVNo, float *pfVolt)
PAV2250A_GetIntRefVolt is responsible for returning internal reference voltage value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefOutputState](#) (int nPAVNo, int nOutputState)
PAV2250A_SetIntRefOutputState is responsible for setting the output state. (OFF/ON).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefOutputState](#) (int nPAVNo, int *pnOutputState)
PAV2250A_GetIntRefOutputState is responsible for returning the index associated with the current reference state output.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefOverCurState](#) (int nPAVNo, int *pnOvrCurState)
PAV2250A_GetIntRefOutputState is responsible for returning the over current state.
- [_PAV2250AFUNC](#) int [PAV2250A_ResetIntRefOverCur](#) (int nPAVNo)
PAV2250A_ResetIntRefOverCur is responsible for resetting the over current flag.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefSenseDir](#) (int nPAVNo, int nSenseDir)
PAV2250A_SetIntRefSenseDir is responsible for setting the sense direction. (Front/Back).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefSenseDir](#) (int nPAVNo, int *pnSenseDir)
PAV2250A_GetIntRefSenseDir is responsible for returning the index associated with the current Sense line direction.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefRemoteSense](#) (int nPAVNo, int nRemoteSense)
PAV2250A_SetRefGenRmtSenseState is responsible for setting the the sense state. (Disable/Enable).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefRemoteSense](#) (int nPAVNo, int *pnRemoteSense)
PAV2250A_GetIntRefRemoteSense is responsible for returning the index associated with the current Sense line state.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefMeasCur](#) (int nPAVNo, int *pnMeasCur)
PAV2250A_GetIntRefMeasCur is responsible for returning the internal reference measured current.

3.22.1 Detailed Description

3.22.2 Function Documentation

3.22.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefFreq](#) (int nPAVNo, float * pfFreq)

[PAV2250A_GetIntRefFreq](#) is responsible for returning internal reference frequency value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfFreq</i>	: (Output) Frequency floating point value for the internal reference

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.22.2.2 `_PAV2250AFUNC int PAV2250A_GetIntRefMeasCur (int nPAVNo, int * pnMeasCur)`

PAV2250A_GetIntRefMeasCur is responsible for returning the internal reference measured current.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pnMeasCur</i>	: (Output) Measured current in mA

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.22.2.3 `_PAV2250AFUNC int PAV2250A_GetIntRefOutputState (int nPAVNo, int * pnOutputState)`

PAV2250A_GetIntRefOutputState is responsible for returning the index associated with the current reference state output.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOutputState</i>	: (Output) Reference Output State (0 or 1) <ul style="list-style-type: none"> • PAV_INT_REF_OUT_NOT_AVAILABLE (0) • PAV_INT_REF_OUT_AVAILABLE (1)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.22.2.4 `_PAV2250AFUNC int PAV2250A_GetIntRefOverCurState (int nPAVNo, int * pnOvrCurState)`

PAV2250A_GetIntRefOutputState is responsible for returning the over current state.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnOvrCurState</i>	: (Output) Over Current State (0 or 1) <ul style="list-style-type: none"> • PAV_NO_OVER_CURRENT (0) • PAV_OVER_CURRENT (1)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.22.2.5 `_PAV2250AFUNC int PAV2250A_GetIntRefRemoteSense (int nPAVNo, int * pnRemoteSense)`

PAV2250A_GetIntRefRemoteSense is responsible for returning the index associated with the current Sense line state.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnRemoteSense</i>	: (Output) Sense state (0 or 1) <ul style="list-style-type: none"> • PAV_RMT_SENSE_DISABLE (0) • PAV_RMT_SENSE_ENABLE (1)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.22.2.6 `_PAV2250AFUNC int PAV2250A_GetIntRefSenseDir (int nPAVNo, int * pnSenseDir)`

PAV2250A_GetIntRefSenseDir is responsible for returning the index associated with the current Sense line direction.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

<i>pnSenseDir</i>	: (Output) Sense Direction (0 or 1) <ul style="list-style-type: none"> • PAV_REF_GEN_DIR_BACK (0) • PAV_REF_GEN_DIR_FRONT (1)
-------------------	---

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.22.2.7 `_PAV2250AFUNC int PAV2250A_GetIntRefVolt (int nPAVNo, float * pfVolt)`

PAV2250A_GetIntRefVolt is responsible for returning internal reference voltage value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfVolt</i>	: (Output) Voltage floating point value for the internal reference

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_WRITE : Unable to send command to 2250A
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.22.2.8 `_PAV2250AFUNC int PAV2250A_ResetIntRefOverCur (int nPAVNo)`

PAV2250A_ResetIntRefOverCur is responsible for resetting the over current flag.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.22.2.9 `_PAV2250AFUNC int PAV2250A_SetIntRefFreq (int nPAVNo, float fFreq)`

`PAV2250A_SetIntRefFreq` sends a command to the PAV 2250A to force the internal reference frequency to the specified value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fFreq</i>	: (Input) Value to set internal reference frequency to

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.22.2.10 `_PAV2250AFUNC int PAV2250A_SetIntRefOutputState (int nPAVNo, int nOutputState)`

PAV2250A_SetIntRefOutputState is responsible for setting the output state. (OFF/ON).

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>nOutputState</i>	: (Input) <ul style="list-style-type: none"> • 0 : "PAV_INT_REF_OUT_NOT_AVAILABLE" • 1 : "PAV_INT_REF_OUT_AVAILABLE"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_RANGE : Supplied parameter value was out of expected range
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.22.2.11 `_PAV2250AFUNC int PAV2250A_SetIntRefRemoteSense (int nPAVNo, int nRemoteSense)`

PAV2250A_SetRefGenRmtSenseState is responsible for setting the the sense state. (Disable/Enable).

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>nRemoteSense</i>	: (Input) Specifies whether or not remote sense is enabled or disabled. <ul style="list-style-type: none"> • 0 : "PAV_RMT_SENSE_DISABLE" • 1 : "PAV_RMT_SENSE_ENABLE"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_RANGE : Supplied parameter value was out of expected range
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ParseForCommaSeparatedDataElements](#)

3.22.2.12 `_PAV2250AFUNC int PAV2250A_SetIntRefSenseDir (int nPAVNo, int nSenseDir)`

PAV2250A_SetIntRefSenseDir is responsible for setting the sense direction. (Front/Back).

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>nSenseDir</i>	: (Input) <ul style="list-style-type: none"> • 0 : "PAV_REF_GEN_DIR_BACK" • 1 : "PAV_REF_GEN_DIR_FRONT"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_RANGE : Supplied parameter value was out of expected range
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ParseForCommaSeparatedDataElements](#)

3.22.2.13 `_PAV2250AFUNC int PAV2250A_SetIntRefVolt (int nPAVNo, float fVolt)`

PAV2250A_SetIntRefVolt sends a command to the PAV 2250A to force the internal reference voltage to the specified value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fVolt</i>	: (Input) Value to set internal reference voltage to

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.23 Local Functions (not exported)

Functions

- [bool SendIEEEMessage](#) (int nPAVNo)

SendIEEEMessage sends the command currently found in "cmdBuf" via IEEE to the PAV 2250A.
- [int GetRange](#) (char szCommand[], int nPAVNo, [bool](#) &bAutoRange, [bool](#) &bRangeMismatch, int &nRangeIndex)

GetRange performs the work to fetch range information from the PAV and return pertinent information to the caller.
- [bool IsLanguageTypeNative](#) (int nPAVNo)

IsLanguageTypeNative returns whether or not the current IEEE language is configured for "Native".
- [bool IsLanguageTypeLegacy](#) (int nPAVNo)

IsLanguageTypeLegacy returns whether or not the current IEEE language is configured for "Legacy".
- [int ExecuteRemoteCmd](#) (char szCommand[], int nPAVNo, [bool](#) bExpectReply, int nReceiveBufferSize)

ExecuteRemoteCmd is responsible for executing the command found in szCommand and filling tempbuf with any response (if one is expected).
- [int ExecuteRemoteCmdFloat](#) (char szCommand[], int nPAVNo, float *pfValue)

ExecuteRemoteCmdFloat is responsible for executing the command found in szCommand and converting the return value to a floating point value.
- [int ExecuteRemoteCmdInt](#) (char szCommand[], int nPAVNo, int *pnValue)

ExecuteRemoteCmdInt is responsible for executing the command found in szCommand and converting the return value to an Int value.
- [int ExecuteRemoteCmdBool](#) (char szCommand[], int nPAVNo, [bool](#) *pbValue)

ExecuteRemoteCmdBool is responsible for executing the command found in szCommand and converting the return value to a Bool value.
- [int ExecuteRemoteCmdString](#) (char szCommand[], int nPAVNo, char *pszValue)

ExecuteRemoteCmdString is responsible for executing the command found in szCommand and copying the return value into the passed in string buffer.
- [bool WaitForResponse](#) (int nPAVNo)

WaitForResponse is responsible for waiting for a response from the PAV.
- [void ParseForCommaSeparatedDataElements](#) (int nPAVNo, int *pnNumberOfDataElements)

ParseForCommaSeparatedDataElements is a utility function that parses the tempbuf buffer for data values and places each value into the valueBuf data structure.
- [unsigned int HexStr2Declnt](#) (char *hexvalue)

HexStr2Declnt converts a string containing a Hex value into a 32-bit decimal value.

3.23.1 Detailed Description

3.23.2 Function Documentation

3.23.2.1 int ExecuteRemoteCmd (char szCommand[], int nPAVNo, bool bExpectReply, int nReceiveBufferSize)

ExecuteRemoteCmd is responsible for executing the command found in szCommand and filling tempbuf with any response (if one is expected).

Parameters

<i>szCommand</i>	: (Input) Command to execute.
<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>bExpectReply</i>	: (Input) True if a reply is expected; False otherwise

<i>nReceiveBuffer-Size</i>	: (Input) Size of return buffer
----------------------------	---------------------------------

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.23.2.2 int ExecuteRemoteCmdBool (char szCommand[], int nPAVNo, bool * pbValue)

ExecuteRemoteCmdBool is responsible for executing the command found in szCommand and converting the return value to a Bool value.

Parameters

<i>szCommand</i>	: (Input) Command to execute.
<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbValue</i>	: (Output) Bool return value.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.23.2.3 int ExecuteRemoteCmdFloat (char szCommand[], int nPAVNo, float * pfValue)

ExecuteRemoteCmdFloat is responsible for executing the command found in szCommand and converting the return value to a floating point value.

Parameters

<i>szCommand</i>	: (Input) Command to execute.
<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pfValue</i>	: (Output) Floating point return value.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.23.2.4 `int ExecuteRemoteCmdInt (char szCommand[], int nPAVNo, int * pnValue)`

`ExecuteRemoteCmdInt` is responsible for executing the command found in `szCommand` and converting the return value to an `Int` value.

Parameters

<i>szCommand</i>	: (Input) Command to execute.
<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pnValue</i>	: (Output) Int return value.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.23.2.5 int ExecuteRemoteCmdString (char *szCommand*[], int *nPAVNo*, char * *pszValue*)

ExecuteRemoteCmdString is responsible for executing the command found in *szCommand* and copying the return value into the passed in string buffer.

Parameters

<i>szCommand</i>	: (Input) Command to execute.
<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pszValue</i>	: (Output) Char string return value.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.23.2.6 int GetRange (char *szCommand*[], int *nPAVNo*, bool & *bAutoRange*, bool & *bRangeMismatch*, int & *nRangeIndex*)

GetRange performs the work to fetch range information from the PAV and return pertinent information to the caller.

Parameters

<i>szCommand</i>	: (Input) Range command to execute
<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>bAutoRange</i>	: (Output) True if the PAV is configured for AUTO ranging
<i>bRange-Mismatch</i>	: (Output) True if there is a range mismatch between the actual range and desired configured range; False otherwise
<i>nRangeIndex</i>	: (Output) The actual range index the reference is operating in <ul style="list-style-type: none"> • PAV_RANGE_50MV (0) • PAV_RANGE_100MV (1) • PAV_RANGE_200MV (2) • PAV_RANGE_500MV (3) • PAV_RANGE_1V (4) • PAV_RANGE_2V (5) • PAV_RANGE_5V (6) • PAV_RANGE_10V (7) • PAV_RANGE_20V (8) • PAV_RANGE_50V (9) • PAV_RANGE_100V (10) • PAV_RANGE_200V (11) • PAV_RANGE_500V (12)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#), [ParseForCommaSeparatedDataElements](#)

3.23.2.7 unsigned int HexStr2DecInt (char * *hexvalue*)

HexStr2DecInt converts a string containing a Hex value into a 32-bit decimal value.

Parameters

<i>hexvalue</i>	: (Input) string to convert from Hex to a 32-bit decimal value
-----------------	--

Returns

unsigned int : converted value

3.23.2.8 bool IsLanguageTypeLegacy (int *nPAVNo*)

IsLanguageTypeLegacy returns whether or not the current IEEE language is configured for "Legacy".

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- True : If IEEE language is configured for "Legacy"
- False : otherwise

3.23.2.9 bool IsLanguageTypeNative (int *nPAVNo*)

IsLanguageTypeNative returns whether or not the current IEEE language is configured for "Native".

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- True : If IEEE language is configured for "Native"
- False : otherwise

3.23.2.10 void ParseForCommaSeparatedDataElements (int *nPAVNo*, int * *pnNumberOfDataElements*)

ParseForCommaSeparatedDataElements is a utility function that parses the tempbuf buffer for data values and places each value into the valueBuf data structure.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pnNumberOfDataElements</i>	: (output) Number of data elements parsed.

Returns

VOID

3.23.2.11 bool SendIEEEMessage (int *nPAVNo*)

SendIEEEMessage sends the command currently found in "cmdBuf" via IEEE to the PAV 2250A.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
---------------	--

Returns

- True : if success
- False : otherwise

3.23.2.12 bool WaitForResponse (int *nPAVNo*)

WaitForResponse is responsible for waiting for a response from the PAV.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
---------------	---

Returns

3.24 LVDT Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTEnabled](#) (int nPAVNo, bool *pbLVDTEnabled)
PAV2250A_GetLVDTEnabled returns whether or not the PAV's LVDT calculations are enabled or disabled.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTEnabledText](#) (int nPAVNo, char *pszLVDTEnabledText)
PAV2250A_GetLVDTEnabledText retrieves the LVDT enabled text: "Disabled" or "Enabled"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTEnabled](#) (int nPAVNo)
PAV2250A_SetLVDTEnabled enables the LVDT calculations : 1 ("Enabled")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTDisabled](#) (int nPAVNo)
PAV2250A_SetLVDTDisabled disables the LVDT calculations : 0 ("Disabled")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTPosition](#) (int nPAVNo, float *pfPosition)
PAV2250A_GetLVDTPosition retrieves the current LVDT position.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTType](#) (int nPAVNo, int *pnLVDTType)
PAV2250A_GetLVDTType retrieves the LVDT mode type: 2 = 2-Wire, 3 = 3-Wire, 4 = 4-Wire
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTTypeText](#) (int nPAVNo, char *pszLVDTTypeText)
PAV2250A_GetLVDTTypeText retrieves the LVDT mode type text: "2-Wire", "3-Wire", "4-Wire"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType](#) (int nPAVNo, int nLVDTTypeIndex)
PAV2250A_SetLVDTType sets the LVDT mode type: 2 = "2-Wire", 3 = "3-Wire", 4 = "4-Wire"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType2Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType2Wire sets the LVDT mode type to: 2 ("2-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType3Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType3Wire sets the LVDT mode type to: 3 ("3-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType4Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType4Wire sets the LVDT mode type to: 4 ("4-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTSignal](#) (int nPAVNo, int *pnLVDTSignal)
PAV2250A_GetLVDTSignal retrieves the LVDT signal: 0 = Fund, 1 = Total, 2 = INPH
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTSignalText](#) (int nPAVNo, char *pszLVDTSignalText)
PAV2250A_GetLVDTSignalText retrieves the LVDT signal text: "Fund", "Total", "INPH"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignal](#) (int nPAVNo, int nLVDTSignalIndex)
PAV2250A_SetLVDTSignal sets the LVDT signal: 0 = "Fund", 1 = "Total", 2 = "INPH"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalINPH](#) (int nPAVNo)
PAV2250A_SetLVDTSignalINPH sets the LVDT signal to: 2 ("INPH")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalFund](#) (int nPAVNo)
PAV2250A_SetLVDTSignalFund sets the LVDT signal to: 0 ("Fund")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalTotal](#) (int nPAVNo)
PAV2250A_SetLVDTSignalTotal sets the LVDT signal to: 1 ("Total")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTVA](#) (int nPAVNo, float *pfLVDTVA)
PAV2250A_GetLVDTVA retrieves the current LVDT VA value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTVB](#) (int nPAVNo, float *pfLVDTVB)
PAV2250A_GetLVDTVB retrieves the current LVDT VB value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTPOFF](#) (int nPAVNo, float fLVDTPOFF)
PAV2250A_SetLVDTPOFF sends a command to the PAV 2250A to set a Phase Offset of the desired value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTPOFF](#) (int nPAVNo, float *pfLVDTPOFF)
PAV2250A_GetLVDTPOFF retrieves the current LVDT Phase Offset (POFF) value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTScale](#) (int nPAVNo, float fLVDTScale)
PAV2250A_SetLVDTScale sends a command to the PAV 2250A to set a Scale of the desired value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTScale](#) (int nPAVNo, float *pfLVDTScale)
PAV2250A_GetLVDTScale retrieves the current LVDT Scale value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDT4WireAlgorithm](#) (int nPAVNo, int *pnLVDT4WireAlgorithm)

- PAV2250A_GetLVDT4WireAlgorithm* retrieves the LVDT 4-Wire Algorithm: 0 = V(a), V(a) + V(b), 1 = V(a), V(b)

 - [_PAV2250AFUNC](#) int [PAV2250A_GetLVDT4WireAlgorithmText](#) (int nPAVNo, char *pszLVDT4WireAlgorithmText)

PAV2250A_GetLVDT4WireAlgorithmText retrieves the LVDT 4-Wire Algorithm text: "V(a), V(a) + V(b)", "V(a), V(b)"
 - [_PAV2250AFUNC](#) int [PAV2250A_SetLVDT4WireAlgorithm](#) (int nPAVNo, int nLVDT4WireAlgorithmIndex)

PAV2250A_SetLVDT4WireAlgorithm sets the LVDT 4-Wire Algorithm: 0 = "V(a), V(a) + V(b)", 1 = "V(a), V(b)"
 - [_PAV2250AFUNC](#) int [PAV2250A_SetLVDT4WireVA_VAPLUSVB](#) (int nPAVNo)

PAV2250A_SetLVDT4WireVA_VAPLUSVB sets the LVDT 4-Wire Algorithm to: 0 ("V(a), V(a) + V(b)")
 - [_PAV2250AFUNC](#) int [PAV2250A_SetLVDT4WireVA_VB](#) (int nPAVNo)

PAV2250A_SetLVDT4WireVA_VB sets the LVDT 4-Wire Algorithm to: 1 ("V(a), V(b)")

3.24.1 Detailed Description

3.24.2 Function Documentation

3.24.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetLVDT4WireAlgorithm](#) (int nPAVNo, int * pnLVDT4WireAlgorithm)

[PAV2250A_GetLVDT4WireAlgorithm](#) retrieves the LVDT 4-Wire Algorithm: 0 = V(a), V(a) + V(b), 1 = V(a), V(b)

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pnLVDT4WireAlgorithm</i>	: (output) LVDT 4-Wire Algorithm (0, or 1) . <ul style="list-style-type: none"> PAV_LVDT_4WIRE_VA_VAPLUSVB (0) PAV_LVDT_4WIRE_VA_VB (1)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.24.2.2 [_PAV2250AFUNC](#) int [PAV2250A_GetLVDT4WireAlgorithmText](#) (int nPAVNo, char * pszLVDT4WireAlgorithmText)

[PAV2250A_GetLVDT4WireAlgorithmText](#) retrieves the LVDT 4-Wire Algorithm text: "V(a), V(a) + V(b)", "V(a), V(b)"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pszLVDT4WireAlgorithmText</i>	: (output) LVDT 4-Wire Algorithm text ("V(a), V(a) + V(b)", "V(a), V(b)") .

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.24.2.3 `_PAV2250AFUNC int PAV2250A_GetLVDTEnabled (int nPAVNo, bool * pbLVDTEnabled)`

PAV2250A_GetLVDTEnabled returns whether or not the PAV's LVDT calculations are enabled or disabled.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>pbLVDTEnabled</i>	: (Output) <ul style="list-style-type: none"> • true : if enabled • false : if NOT enabled

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid *nPAVNo* parameter
- PAV_ERROR_FUNC_NOT_SUPPORTED : Function supported only with PAV2250A_NATIVE
- PAV_ERROR_WRITE : Unable to send command to 2250A

See Also

[ExecuteRemoteCmdBool](#)

3.24.2.4 `_PAV2250AFUNC int PAV2250A_GetLVDTEnabledText (int nPAVNo, char * pszLVDTEnabledText)`

PAV2250A_GetLVDTEnabledText retrieves the LVDT enabled text: "Disabled" or "Enabled"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pszLVDT-EnabledText</i>	: (output) LVDT enabled state (Disabled or Enabled) .

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid *nPAVNo* parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.24.2.5 `_PAV2250AFUNC int PAV2250A_GetLVDTPOFF (int nPAVNo, float * pfLVDTPOFF)`

PAV2250A_GetLVDTPOFF retrieves the current LVDT Phase Offset (POFF) value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
---------------	---

<i>pfLVDTPOFF</i>	: (output) LVDT Phase Offset value.
-------------------	-------------------------------------

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.24.2.6 `_PAV2250AFUNC int PAV2250A_GetLVDTPosition (int nPAVNo, float * pfPosition)`

PAV2250A_GetLVDTPosition retrieves the current LVDT position.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfPosition</i>	: (output) LVDT Position.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.24.2.7 `_PAV2250AFUNC int PAV2250A_GetLVDTScale (int nPAVNo, float * pfLVDTScale)`

PAV2250A_GetLVDTScale retrieves the current LVDT Scale value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfLVDTScale</i>	: (output) LVDT Scale value.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.24.2.8 `_PAV2250AFUNC int PAV2250A_GetLVDTSignal (int nPAVNo, int * pnLVDTSignal)`

PAV2250A_GetLVDTSignal retrieves the LVDT signal: 0 = Fund, 1 = Total, 2 = INPH

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pnLVDTSignal</i>	: (output) LVDT signal (0, 1, 2) . <ul style="list-style-type: none"> • PAV_LVDT_SIGNAL_FUND (0) • PAV_LVDT_SIGNAL_TOTAL (1) • PAV_LVDT_SIGNAL_INPH (2)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.24.2.9 `_PAV2250AFUNC int PAV2250A_GetLVDTSignalText (int nPAVNo, char * pszLVDTSignalText)`

PAV2250A_GetLVDTSignalText retrieves the LVDT signal text: "Fund", "Total", "INPH"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pszLVDTSignal-Text</i>	: (output) LVDT signal text (Fund or Total) .

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.24.2.10 `_PAV2250AFUNC int PAV2250A_GetLVDTType (int nPAVNo, int * pnLVDTType)`

PAV2250A_GetLVDTType retrieves the LVDT mode type: 2 = 2-Wire, 3 = 3-Wire, 4 = 4-Wire

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pnLVDTType</i>	: (output) LVDT mode type (2, 3 or 4) . <ul style="list-style-type: none"> • PAV_LVDT_TYPE_2WIRE (2) • PAV_LVDT_TYPE_3WIRE (3) • PAV_LVDT_TYPE_4WIRE (4)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdInt](#)

3.24.2.11 `_PAV2250AFUNC int PAV2250A_GetLVDTTypeText (int nPAVNo, char * pszLVDTTypeText)`

PAV2250A_GetLVDTTypeText retrieves the LVDT mode type text: "2-Wire", "3-Wire", "4-Wire"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pszLVDTType-Text</i>	: (output) LVDT mode type text (2-Wire, 3-Wire or 4-Wire) .

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdString](#)

3.24.2.12 `_PAV2250AFUNC int PAV2250A_GetLVDTVA (int nPAVNo, float * pfLVDTVA)`

PAV2250A_GetLVDTVA retrieves the current LVDT VA value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfLVDTVA</i>	: (output) LVDT VA value.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.24.2.13 `_PAV2250AFUNC int PAV2250A_GetLVDTVB (int nPAVNo, float * pfLVDTVB)`

PAV2250A_GetLVDTVB retrieves the current LVDT VB value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfLVDTVB</i>	: (output) LVDT VB value.

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.24.2.14 `_PAV2250AFUNC int PAV2250A_SetLVDT4WireAlgorithm (int nPAVNo, int nLVDT4WireAlgorithmIndex)`

PAV2250A_SetLVDT4WireAlgorithm sets the LVDT 4-Wire Algorithm: 0 = "V(a), V(a) + V(b)", 1 = "V(a), V(b)"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>nLVDT4Wire-AlgorithmIndex</i>	: (Input) <ul style="list-style-type: none"> • 0 : "PAV_LVDT_4WIRE_VA_VAPLUSVB" • 1 : "PAV_LVDT_4WIRE_VA_VB"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_RANGE : Supplied parameter value was out of expected range
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.15 `_PAV2250AFUNC int PAV2250A_SetLVDT4WireVA_VAPLUSVB (int nPAVNo)`

PAV2250A_SetLVDT4WireVA_VAPLUSVB sets the LVDT 4-Wire Algorithm to: 0 ("V(a), V(a) + V(b)")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.16 `_PAV2250AFUNC` int `PAV2250A_SetLVDT4WireVA_VB` (int *nPAVNo*)

`PAV2250A_SetLVDT4WireVA_VB` sets the LVDT 4-Wire Algorithm to: 1 ("V(a), V(b)")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.17 `_PAV2250AFUNC int PAV2250A_SetLVDTDisabled (int nPAVNo)`

PAV2250A_SetLVDTDisabled disables the LVDT calculations : 0 ("Disabled")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.18 `_PAV2250AFUNC int PAV2250A_SetLVDTEnabled (int nPAVNo)`

PAV2250A_SetLVDTEnabled enables the LVDT calculations : 1 ("Enabled")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.19 `_PAV2250AFUNC int PAV2250A_SetLVDTPOFF (int nPAVNo, float fLVDTPOFF)`

PAV2250A_SetLVDTPOFF sends a command to the PAV 2250A to set a Phase Offset of the desired value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fLVDTPOFF</i>	: (Input) Value of which to set the Phase Offset

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.20 _PAV2250AFUNC int PAV2250A_SetLVDTScale (int *nPAVNo*, float *fLVDTScale*)

PAV2250A_SetLVDTScale sends a command to the PAV 2250A to set a Scale of the desired value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
<i>fLVDTScale</i>	: (Input) Value of which to set the Scale

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.21 _PAV2250AFUNC int PAV2250A_SetLVDTSignal (int *nPAVNo*, int *nLVDTSignalIndex*)

PAV2250A_SetLVDTSignal sets the LVDT signal: 0 = "Fund", 1 = "Total", 2 = "INPH"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>nLVDTSignal-Index</i>	: (Input) <ul style="list-style-type: none"> • 0 : "PAV_LVDT_SIGNAL_FUND" • 1 : "PAV_LVDT_SIGNAL_TOTAL" • 2 : "PAV_LVDT_SIGNAL_INPH"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_RANGE : Supplied parameter value was out of expected range
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.22 `_PAV2250AFUNC` int `PAV2250A_SetLVDTSignalFund` (int *nPAVNo*)

`PAV2250A_SetLVDTSignalFund` sets the LVDT signal to: 0 ("Fund")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid *nPAVNo* parameter
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.23 `_PAV2250AFUNC` int `PAV2250A_SetLVDTSignalINPH` (int *nPAVNo*)

`PAV2250A_SetLVDTSignalINPH` sets the LVDT signal to: 2 ("INPH")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid *nPAVNo* parameter
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.24 `_PAV2250AFUNC` int `PAV2250A_SetLVDTSignalTotal` (int *nPAVNo*)

`PAV2250A_SetLVDTSignalTotal` sets the LVDT signal to: 1 ("Total")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- `PAV_SUCCESS` : Function is successful
- `PAV_ERROR_PAVNO` : Invalid *nPAVNo* parameter
- `PAV_ERROR_DATA` : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.25 `_PAV2250AFUNC` int `PAV2250A_SetLVDTTType` (int *nPAVNo*, int *nLVDTTTypeIndex*)

`PAV2250A_SetLVDTTType` sets the LVDT mode type: 2 = "2-Wire", 3 = "3-Wire", 4 = "4-Wire"

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>nLVDTType-Index</i>	: (Input) <ul style="list-style-type: none"> • 2 : "PAV_LVDT_TYPE_2WIRE" • 3 : "PAV_LVDT_TYPE_3WIRE" • 4 : "PAV_LVDT_TYPE_4WIRE"

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_RANGE : Supplied parameter value was out of expected range
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.26 `_PAV2250AFUNC int PAV2250A_SetLVDTType2Wire (int nPAVNo)`

PAV2250A_SetLVDTType2Wire sets the LVDT mode type to: 2 ("2-Wire")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.27 `_PAV2250AFUNC int PAV2250A_SetLVDTType3Wire (int nPAVNo)`

PAV2250A_SetLVDTType3Wire sets the LVDT mode type to: 3 ("3-Wire")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.24.2.28 `_PAV2250AFUNC` int `PAV2250A_SetLVDTType4Wire` (int *nPAVNo*)

`PAV2250A_SetLVDTType4Wire` sets the LVDT mode type to: 4 ("4-Wire")

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV number assigned to connection with PAV2250A. (1-MAX_PAV)
---------------	--

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmd](#)

3.25 Reference Only Functions

Functions

- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotal](#) (int nPAVNo, float *pfTotal)
PAV2250A_GetRefTotal retrieves the reference Total value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefFundMag](#) (int nPAVNo, float *pfFundMag)
PAV2250A_GetRefFundMag retrieves the reference Fundamental (Magnitude) value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefInPhase](#) (int nPAVNo, float *pfInPhase)
PAV2250A_GetRefInPhase retrieves the reference InPhase value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefQuad](#) (int nPAVNo, float *pfQuad)
PAV2250A_GetRefQuad retrieves the reference Quad value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefPhase](#) (int nPAVNo, float *pfPhase)
PAV2250A_GetRefPhase retrieves the reference Phase value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTHD](#) (int nPAVNo, float *pfTHD)
PAV2250A_GetRefTHD retrieves the reference THD value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalRatio](#) (int nPAVNo, float *pfTotalRatio)
PAV2250A_GetRefTotalRatio retrieves the reference Total Ratio value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefFundMagRatio](#) (int nPAVNo, float *pfFundMagRatio)
PAV2250A_GetRefFundMagRatio retrieves the reference Fundamental (Magnitude) Ratio value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefInPhaseRatio](#) (int nPAVNo, float *pfInPhaseRatio)
PAV2250A_GetRefInPhaseRatio retrieves the reference InPhase Ratio value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefQuadRatio](#) (int nPAVNo, float *pfQuadRatio)
PAV2250A_GetRefQuadRatio retrieves the reference Quad Ratio value.

3.25.1 Detailed Description

3.25.2 Function Documentation

3.25.2.1 [_PAV2250AFUNC](#) int [PAV2250A_GetRefFundMag](#) (int nPAVNo, float * pfFundMag)

[PAV2250A_GetRefFundMag](#) retrieves the reference Fundamental (Magnitude) value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfFundMag</i>	: (output) Fundamental (Magnitude)

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.2 [_PAV2250AFUNC](#) int [PAV2250A_GetRefFundMagRatio](#) (int nPAVNo, float * pfFundMagRatio)

[PAV2250A_GetRefFundMagRatio](#) retrieves the reference Fundamental (Magnitude) Ratio value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfFundMagRatio</i>	: (output) Fundamental (Magnitude) Ratio

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.3 `_PAV2250AFUNC int PAV2250A_GetRefInPhase (int nPAVNo, float * pfInPhase)`

PAV2250A_GetRefInPhase retrieves the reference InPhase value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfInPhase</i>	: (output) InPhase

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.4 `_PAV2250AFUNC int PAV2250A_GetRefInPhaseRatio (int nPAVNo, float * pfInPhaseRatio)`

PAV2250A_GetRefInPhaseRatio retrieves the reference InPhase Ratio value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfInPhaseRatio</i>	: (output) InPhase Ratio

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.5 `_PAV2250AFUNC int PAV2250A_GetRefPhase (int nPAVNo, float * pfPhase)`

PAV2250A_GetRefPhase retrieves the reference Phase value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfPhase</i>	: (output) Phase

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.6 `_PAV2250AFUNC int PAV2250A_GetRefQuad (int nPAVNo, float * pfQuad)`

PAV2250A_GetRefQuad retrieves the reference Quad value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfQuad</i>	: (output) Quad

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.7 `_PAV2250AFUNC int PAV2250A_GetRefQuadRatio (int nPAVNo, float * pfQuadRatio)`

PAV2250A_GetRefQuadRatio retrieves the reference Quad Ratio value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfQuadRatio</i>	: (output) Quad Ratio

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.8 `_PAV2250AFUNC int PAV2250A_GetRefTHD (int nPAVNo, float * pfTHD)`

PAV2250A_GetRefTHD retrieves the reference THD value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfTHD</i>	: (output) THD

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.9 _PAV2250AFUNC int PAV2250A_GetRefTotal (int *nPAVNo*, float * *pfTotal*)

PAV2250A_GetRefTotal retrieves the reference Total value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfTotal</i>	: (output) Reference Total

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

3.25.2.10 _PAV2250AFUNC int PAV2250A_GetRefTotalRatio (int *nPAVNo*, float * *pfTotalRatio*)

PAV2250A_GetRefTotalRatio retrieves the reference Total Ratio value.

Parameters

<i>nPAVNo</i>	: (Input) Logical PAV Index assigned to connection with PAV2250A. (0-(MAX_PAV-1))
<i>pfTotalRatio</i>	: (output) Total Ratio

Returns

- PAV_SUCCESS : Function is successful
- PAV_ERROR_PAVNO : Invalid nPAVNo parameter
- PAV_ERROR_DATA : Data returned from 2250A is not valid for command sent

See Also

[ExecuteRemoteCmdFloat](#)

Chapter 4

File Documentation

4.1 E:/(BLACKFIN CODE)/Instruments/PAV-2250A/Driver_VS2010/Source/PAV2250ADII/-PAV2250ADII.cpp File Reference

```
#include <winsock2.h>
#include <ws2tcpip.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
#include "decl-32.h"
#include "USBComm.h"
#include "Ethernet.h"
#include "PAV2250AD11.h"
```

Functions

- **BOOL** APIENTRY **DIIMain** (HANDLE hModule, DWORD ul_reason_for_call, LPVOID lpReserved)
- **bool** **SendIEEEMessage** (int nPAVNo)
SendIEEEMessage sends the command currently found in "cmdBuf" via IEEE to the PAV 2250A.
- int **GetRange** (char szCommand[], int nPAVNo, **bool** &bAutoRange, **bool** &bRangeMismatch, int &nRangeIndex)
GetRange performs the work to fetch range information from the PAV and return pertinent information to the caller.
- **bool** **IsLanguageTypeNative** (int nPAVNo)
IsLanguageTypeNative returns whether or not the current IEEE language is configured for "Native".
- **bool** **IsLanguageTypeLegacy** (int nPAVNo)
IsLanguageTypeLegacy returns whether or not the current IEEE language is configured for "Legacy".
- int **ExecuteRemoteCmd** (char szCommand[], int nPAVNo, **bool** bExpectReply, int nReceiveBufferSize)
ExecuteRemoteCmd is responsible for executing the command found in szCommand and filling tempbuf with any response (if one is expected).
- int **ExecuteRemoteCmdFloat** (char szCommand[], int nPAVNo, float *pfValue)
ExecuteRemoteCmdFloat is responsible for executing the command found in szCommand and converting the return value to a floating point value.
- int **ExecuteRemoteCmdInt** (char szCommand[], int nPAVNo, int *pnValue)
ExecuteRemoteCmdInt is responsible for executing the command found in szCommand and converting the return value to an Int value.
- int **ExecuteRemoteCmdBool** (char szCommand[], int nPAVNo, **bool** *pbValue)
ExecuteRemoteCmdBool is responsible for executing the command found in szCommand and converting the return value to a Bool value.

- int [ExecuteRemoteCmdString](#) (char szCommand[], int nPAVNo, char *pszValue)

ExecuteRemoteCmdString is responsible for executing the command found in szCommand and copying the return value into the passed in string buffer.
- bool [WaitForResponse](#) (int nPAVNo)

WaitForResponse is responsible for waiting for a response from the PAV.
- void [ParseForCommaSeparatedDataElements](#) (int nPAVNo, int *pnNumberOfDataElements)

ParseForCommaSeparatedDataElements is a utility function that parses the tempbuf buffer for data values and places each value into the valueBuf data structure.
- unsigned int [HexStr2Declnt](#) (char *hexvalue)

HexStr2Declnt converts a string containing a Hex value into a 32-bit decimal value.
- [_PAV2250AFUNC](#) int [PAV2250A_ConnectViaIEEE](#) (int nPAVNo, int nIEEEAddr, int nIEEELang)

PAV2250A_ConnectViaIEEE sets up and opens the connection to communicate to the PAV2250A via IEEE. The IEEE supports the following language protocols:
- [_PAV2250AFUNC](#) int [PAV2250A_ConnectViaUSB](#) (int nPAVNo, int nDeviceNo)

PAV2250A_ConnectViaUSB sets up and opens the connection to communicate to the PAV2250A via USB. Prior to calling this function, make calls to the [PAV2250A_GetPAV2250AUSBDeviceCnt\(\)](#) routine to determine the number of Cypress USB Devices detected in your system and the [PAV2250A_GetPAV2250ADeviceIDN\(\)](#) routine to determine the appropriate PAV identifier (nPAVNo) associated with the Cypress USB Devices that are connected to PAV2250A via USB.
- [_PAV2250AFUNC](#) int [PAV2250A_ConnectViaEthernet](#) (int nPAVNo, char *szIPAddr, int nPort)

PAV2250A_ConnectViaEthernet sets up and opens the connection to communicate to the 2250A via Ethernet.
- [_PAV2250AFUNC](#) int [PAV2250A_DisconnectIEEE](#) (int nPAVNo)

PAV2250A_DisconnectIEEE closes the connection to communicate to the 2250A via IEEE
- [_PAV2250AFUNC](#) int [PAV2250A_DisconnectUSB](#) (int nPAVNo)

PAV2250A_DisconnectUSB closes the connection to communicate to the 2250A via USB.
- [_PAV2250AFUNC](#) int [PAV2250A_DisconnectEthernet](#) (int nPAVNo)

PAV2250A_DisconnectEthernet closes the connection to communicate to the 2250A via Ethernet.
- [_PAV2250AFUNC](#) int [PAV2250A_GetPAV2250AUSBDeviceCnt](#) (int *pnUSBDeviceCnt)

PAV2250A_GetPAV2250AUSBDeviceCnt invokes the Cypress driver and returns the number of Cypress USB Devices detected with your computer system.
- [_PAV2250AFUNC](#) int [PAV2250A_GetPAV2250ADeviceIDN](#) (int nDeviceNo, char *pszID)

PAV2250A_GetPAV2250ADeviceIDN sends the IDN command to get Device ID string for the device. The ID returned includes the manufacturer (NORTH ATLANTIC), the 2250A module, serial number, and revision information. NOTE: this function does not require an Open call to have already been performed, but it does assume a USB physical connection. If you are connecting to the instrument using one of the other methods (IEEE or Ethernet) you must first "connect" to the device using that method and then call [PAV2250A_PerformGetID](#).
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetID](#) (int nPAVNo, char *pszID)

PAV2250A_PerformGetID sends the IDN command to get Device ID string for the device. The ID returned includes the manufacturer (NORTH ATLANTIC), the 2250A module, serial number, and revision information.
- [_PAV2250AFUNC](#) int [PAV2250A_IsStable](#) (int nPAVNo, bool *pbStable)

PAV2250A_IsStable returns whether or not the PAV is considered to be "stable". (i.e. the unit has settled long enough to allow for accurate readings to be taken.)
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetTotalData](#) (int nPAVNo, char *pszTotalData)

PAV2250A_PerformGetTotalData sends the Total Data command to get Total Data for the device. NOTE: This function should only be called if connecting via Ethernet or IEEE. If connecting by USB, you should fetch the individual components that make up "TotalData" one at a time (Total Ratio, Ref Total RMS AC, Sig Total RMS AC, THD, Frequency, Sample Rate Index, Ref Range, Sig Range, Total Sig Offset, Sig Total Sum, Ref Total RMS AC/DC, Sig Total RMS AC/DC, RefDC, and SigDC). Current USB driver in use has a restriction on amount of data it can return from one call.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetTotalDataRaw](#) (int nPAVNo, int nReceiveBufferSize, char *pszTotalData)

PAV2250A_PerformGetTotalDataRaw sends the Total Data command to get Raw Total Data for the device. NOTE: This function should only be called if connecting via Ethernet or IEEE. Current USB driver in use has a restriction on amount of data it can return from one call.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGroupExecuteTrigger](#) (int nPAVNo, int nIEEEAddr)

- PAV2250A_PerformGroupExecuteTrigger* performs a group execute trigger.
- [_PAV2250AFUNC](#) int [PAV2250A_ResetDefaultValues](#) (int nPAVNo)
PAV2250A_ResetDefaultValues sends the command to set the device settings back to the factory default values.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonics](#) (int nPAVNo, int nHarmonic, char *pszHarmonics)
PAV2250A_PerformGetHarmonics sends the Harmonics command to get Harmonics Data for the device. The Harmonics are returned in a comma separated string and includes the Harmonic Phase, Magnitude, In Phase and Quad for the given harmonic number. (valid values 0 (Fundamental) - 15)
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicPhase](#) (int nPAVNo, int nHarmonic, float *pfPhase)
PAV2250A_GetHarmonicPhase is responsible for returning just the Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicMagnitude](#) (int nPAVNo, int nHarmonic, float *pfMag)
PAV2250A_GetHarmonicMagnitude is responsible for returning just the Magnitude (Amplitude) value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicInPhase](#) (int nPAVNo, int nHarmonic, float *pfInPhase)
PAV2250A_GetHarmonicInPhase is responsible for returning just the In Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicQuad](#) (int nPAVNo, int nHarmonic, float *pfQuad)
PAV2250A_GetHarmonicQuad is responsible for returning just the Quadrature value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonicsRatio](#) (int nPAVNo, int nHarmonic, char *pszHarmonicsRatio)
PAV2250A_PerformGetHarmonicsRatio sends the Harmonics command to get Harmonics Data for the device. The Harmonics are returned in a comma separated string and includes the Harmonic Phase, Magnitude Ratio, In Phase Ratio and Quad Ratio for the given harmonic number. (valid values 0 (Fundamental) - 15)
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioPhase](#) (int nPAVNo, int nHarmonic, float *pfPhase)
PAV2250A_GetHarmonicRatioPhase is responsible for returning just the Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioMagnitude](#) (int nPAVNo, int nHarmonic, float *pfMag)
PAV2250A_GetHarmonicRatioMagnitude is responsible for returning just the Magnitude Ratio (Amplitude) value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioInPhase](#) (int nPAVNo, int nHarmonic, float *pfInPhase)
PAV2250A_GetHarmonicRatioInPhase is responsible for returning just the In Phase Ratio value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioQuad](#) (int nPAVNo, int nHarmonic, float *pfQuad)
PAV2250A_GetHarmonicRatioQuad is responsible for returning just the Quadrature Ratio value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonicDataRaw](#) (int nPAVNo, int nHarmGroup, int nReceiveBufferSize, char *pszHarmonicData)
PAV2250A_PerformGetHarmonicDataRaw sends the Harmonic Data command to get Raw Harmonic Data for the specified Harmonic Group for the device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioState](#) (int nPAVNo, bool *pbHarmRatio)
PAV2250A_GetHarmonicRatioState sends the command to get the Harmonic Ratio State to the PAV2250A device. Harmonic Ratio State controls whether Ratio or Absolute values are shown for Harmonic Display.
- [_PAV2250AFUNC](#) int [PAV2250A_SetHarmonicRatioState](#) (int nPAVNo, bool bHarmRatio)
PAV2250A_SetHarmonicRatioState sends the command to set the Harmonic Ratio State to the PAV2250A device. Harmonic Ratio State controls whether Ratio or Absolute values are shown for Harmonic Display.
- [_PAV2250AFUNC](#) int [PAV2250A_ViewPrevHarmonicGroup](#) (int nPAVNo)
PAV2250A_ViewPrevHarmonicGroup sends the command to set the Harmonic View to the previous group of harmonics.
- [_PAV2250AFUNC](#) int [PAV2250A_ViewNextHarmonicGroup](#) (int nPAVNo)
PAV2250A_ViewNextHarmonicGroup sends the command to set the Harmonic View to the next group of harmonics.
- [_PAV2250AFUNC](#) int [PAV2250A_ViewHarmonic](#) (int nPAVNo, int nHarmonic)
PAV2250A_ViewHarmonic sends the command to set the Harmonic View to show the desired harmonic. Valid values are 0 - 15.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefAutoRange](#) (int nPAVNo)
PAV2250A_SetRefAutoRange sends the command to force the Reference to be in "Auto" range mode.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefAutoRange](#) (int nPAVNo, bool *pbAutoRange)
PAV2250A_GetRefAutoRange determines whether or not the Reference is in "Auto" range mode.

- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange](#) (int nPAVNo, int nRangeIndex)

PAV2250A_SetRefRange sends the command to force the Reference into the specified Range based on the Range Index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeString](#) (int nPAVNo, char *pszRefRange)

PAV2250A_GetRefRangeString is responsible for returning back the Reference range the PAV is currently operating at in string form. Return values will have the word "AUTO" precede the actual range value when the PAV is in "AUTO" range mode and will have the word "OVR" precede the actual range when the actual range is over the configured range. Example: Auto Range with an actual range of 2 Volts will return: "AUTO 2.000". If the range is not "AUTO" and an Over range was not detected, the configured range will be returned such as: "2.000" for the 2 volt range.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeIndexSettings](#) (int nPAVNo, bool *pbAutoRange, bool *pbRangeMismatch, int *pnRangeIndex)

PAV2250A_GetRefRangeIndexSettings is responsible for returning back Reference range information. Return values indicate whether or not the reference is in "Auto" range mode, whether or not there is a range mismatch (i.e. the actual range is different than the desired configured range), and the actual range index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeConfigIndex](#) (int nPAVNo, int *pnRangeIndex)

PAV2250A_GetRefRangeConfigIndex is responsible for returning the index of the Reference range the PAV was configured with.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeActualIndex](#) (int nPAVNo, int *pnRangeIndex)

PAV2250A_GetRefRangeConfigIndex is responsible for returning the index of the Reference range the PAV was actually operating in.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange50MV](#) (int nPAVNo)

PAV2250A_SetRefRange50MV sends the command to force the PAV Reference to be 50MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange100MV](#) (int nPAVNo)

PAV2250A_SetRefRange100MV sends the command to force the PAV Reference to be 100MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange200MV](#) (int nPAVNo)

PAV2250A_SetRefRange200MV sends the command to force the PAV Reference to be 200MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange500MV](#) (int nPAVNo)

PAV2250A_SetRefRange500MV sends the command to force the PAV Reference to be 500MV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange1V](#) (int nPAVNo)

PAV2250A_SetRefRange1V sends the command to force the PAV Reference to be 1V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange2V](#) (int nPAVNo)

PAV2250A_SetRefRange2V sends the command to force the PAV Reference to be 2V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange5V](#) (int nPAVNo)

PAV2250A_SetRefRange5V sends the command to force the PAV Reference to be 5V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange10V](#) (int nPAVNo)

PAV2250A_SetRefRange10V sends the command to force the PAV Reference to be 10V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange20V](#) (int nPAVNo)

PAV2250A_SetRefRange20V sends the command to force the PAV Reference to be 20V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange50V](#) (int nPAVNo)

PAV2250A_SetRefRange50V sends the command to force the PAV Reference to be 50V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange100V](#) (int nPAVNo)

PAV2250A_SetRefRange100V sends the command to force the PAV Reference to be 100V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange200V](#) (int nPAVNo)

PAV2250A_SetRefRange200V sends the command to force the PAV Reference to be 200V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange500V](#) (int nPAVNo)

PAV2250A_SetRefRange500V sends the command to force the PAV Reference to be 500V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigAutoRange](#) (int nPAVNo)

PAV2250A_SetSigAutoRange sends the command to force the Signal to be in "Auto" range mode.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigAutoRange](#) (int nPAVNo, bool *pbAutoRange)

PAV2250A_GetSigAutoRange determines whether or not the Signal is in "Auto" range mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange](#) (int nPAVNo, int nRangeIndex)

PAV2250A_SetSigRange sends the command to force the Signal into the specified Range based on the Range Index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeString](#) (int nPAVNo, char *pszSigRange)

- PAV2250A_GetSigRangeString* is responsible for returning back the Signal range the PAV is currently operating at in string form. Return values will have the word "AUTO" precede the actual range value when the PAV is in "AUTO" range mode and will have the word "OVR" precede the actual range when the actual range is over the configured range. Example: Auto Range with an actual range of 2 Volots will return: "AUTO 2.000". If the range is not "AUTO" and an Over range was not detected, the configured range will be returned such as: "2.000" for the 2 volt range.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeIndexSettings](#) (int nPAVNo, bool *pbAutoRange, bool *pbRangeMismatch, int *pnRangeIndex)
PAV2250A_GetSigRangeIndexSettings is responsible for returning back Signal range information. Return values indicate whether or not the signal is in "Auto" range mode, whether or not there is a range mismatch (i.e. the actual range is different than the desired configured range), and the actual range index.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeConfigIndex](#) (int nPAVNo, int *pnRangeIndex)
PAV2250A_GetSigRangeConfigIndex is responsible for returning the index of the Signal range the PAV was configured with.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeActualIndex](#) (int nPAVNo, int *pnRangeIndex)
PAV2250A_GetSigRangeConfigIndex is responsible for returning the index of the Signal range the PAV is actually operating in.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange50MV](#) (int nPAVNo)
PAV2250A_SetSigRange50MV sends the command to force the PAV Signal to be 50MV.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange100MV](#) (int nPAVNo)
PAV2250A_SetSigRange100MV sends the command to force the PAV Signal to be 100MV.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange200MV](#) (int nPAVNo)
PAV2250A_SetSigRange200MV sends the command to force the PAV Signal to be 200MV.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange500MV](#) (int nPAVNo)
PAV2250A_SetSigRange500MV sends the command to force the PAV Signal to be 500MV.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange1V](#) (int nPAVNo)
PAV2250A_SetSigRange1V sends the command to force the PAV Signal to be 1V.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange2V](#) (int nPAVNo)
PAV2250A_SetSigRange2V sends the command to force the PAV Signal to be 2V.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange5V](#) (int nPAVNo)
PAV2250A_SetSigRange5V sends the command to force the PAV Signal to be 5V.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange10V](#) (int nPAVNo)
PAV2250A_SetSigRange10V sends the command to force the PAV Signal to be 10V.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange20V](#) (int nPAVNo)
PAV2250A_SetSigRange20V sends the command to force the PAV Signal to be 20V.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange50V](#) (int nPAVNo)
PAV2250A_SetSigRange50V sends the command to force the PAV Signal to be 50V.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange100V](#) (int nPAVNo)
PAV2250A_SetSigRange100V sends the command to force the PAV Signal to be 100V.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange200V](#) (int nPAVNo)
PAV2250A_SetSigRange200V sends the command to force the PAV Signal to be 200V.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange500V](#) (int nPAVNo)
PAV2250A_SetSigRange500V sends the command to force the PAV Signal to be 500V.
 - [_PAV2250AFUNC](#) int [PAV2250A_IEEEReset](#) (int nPAVNo, char *pszResults)
PAV2250A_IEEEReset sends the command to reset the 2250A device and set the device setting back to the factory default settings. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
 - [_PAV2250AFUNC](#) int [PAV2250A_IEEEGetErrors](#) (int nPAVNo, char *pszErrors)
PAV2250A_IEEEGetErrors sends the ERR command to get error from the error queue for the device. No error is returned when there are no errors on the queue.
 - [_PAV2250AFUNC](#) int [PAV2250A_IEEECLS](#) (int nPAVNo)
PAV2250A_IEEECLS is responsible for clearing the IEEE - Clears Event Status Registers and Error Message Queue.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetIEEELang](#) (int nPAVNo, int *pnIEEELang)
PAV2250A_GetIEEELang sends the command to get the IEEE language protocol set in the 2250A.

- [_PAV2250AFUNC](#) int [PAV2250A_GetIEEELangText](#) (int nPAVNo, char *pszIEEELang)
PAV2250A_GetIEEELangText sends the command to get the IEEE language protocol set in the 2250A.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang](#) (int nPAVNo, int nIEEELang)
PAV2250A_SetIEEELang sends the command to set the IEEE protocol language to accept when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang2250ANative](#) (int nPAVNo)
PAV2250A_SetIEEELang2250ANative sends the command to set the IEEE protocol to the 2250A Native language when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang2250Legacy](#) (int nPAVNo)
PAV2250A_SetIEEELang2250Legacy sends the command to set the IEEE protocol to the 2250A Native language when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_GetCommState](#) (int nPAVNo, char *pszCommState)
PAV2250A_GetCommState sends the command to get the communication mode set in the 2250A.
- [_PAV2250AFUNC](#) int [PAV2250A_GoToLocal](#) (int nPAVNo)
PAV2250A_GoToLocal sends the command to set the communication mode to Local mode. In Local mode, remote set commands will not be accepted.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteUSB](#) (int nPAVNo)
PAV2250A_SetRemoteUSB sends the command to set the communication mode to Remote USB mode. In Remote USB mode, remote set commands will be accepted if the command is received from the USB interface.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteEthernet](#) (int nPAVNo)
PAV2250A_SetRemoteEthernet sends the command to set the communication mode to Remote Ethernet mode. In Remote Ethernet mode, remote set commands will be accepted if the command is received from the Ethernet interface.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteIEEE](#) (int nPAVNo)
PAV2250A_SetRemoteIEEE sends the command to set the communication mode to Remote IEEE mode. In Remote IEEE mode, remote set commands will be accepted if the command is received from the IEEE interface.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteJ1](#) (int nPAVNo)
PAV2250A_SetRemoteJ1 sends the command to set the communication mode to Remote J1 mode. In Remote J1 mode, remote set commands will be accepted if the command is received from the J1 interface.
- [_PAV2250AFUNC](#) int [PAV2250A_Calibrate](#) (int nPAVNo)
PAV2250A_Calibrate sends the command to force the PAV to perform calibration.
- [_PAV2250AFUNC](#) int [PAV2250A_GetCalState](#) (int nPAVNo, char *pszCalState)
PAV2250A_GetCalState sends a command to the PAV 2250A to retrieve its current calibration state.
- [_PAV2250AFUNC](#) int [PAV2250A_MaxRetry](#) (int nMaxRetry)
PAV2250A_MaxRetry sets the maximum retries to send a command or read a response that will be made when communicating via IEEE. The default value is 0.
- [_PAV2250AFUNC](#) int [PAV2250A_LastCmdSent](#) (int nPAVNo, char szLastCommand[])
PAV2250A_LastCmdSent returns the last command sent via IEEE, USB or Ethernet to the 2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_WriteCommand](#) (int nPAVNo, char szCommand[])
PAV2250A_WriteCommand sends the command to the 2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_QueryCommand](#) (int nPAVNo, char szCommand[], char *pszResponse)
PAV2250A_QueryCommand sends the command to the 2250A device and waits for the 2250A to respond.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeWndAuto](#) (int nPAVNo, bool bAuto)
PAV2250A_SetTimeWndAuto sends a command to the PAV 2250A to force the time window to be in "Auto" mode. The system will decide the data refresh rate.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndAuto](#) (int nPAVNo, bool *pbAuto)
PAV2250A_GetTimeWndAuto sends a command to the PAV 2250A to retrieve whether or not the Time Window is in "Auto" mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeWndOverride](#) (int nPAVNo, float fOverrideInterval)
PAV2250A_SetTimeWndOverride sends a command to the PAV 2250A to force the Time Window to be a specific value.

- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndOverride](#) (int nPAVNo, float *pfOverrideInterval)
PAV2250A_GetTimeWndOverride sends a command to the PAV 2250A to retrieve the current setting for the Time Window override interval.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndActual](#) (int nPAVNo, float *pfActualInterval)
*PAV2250A_GetTimeWndActual sends a command to the PAV 2250A to retrieve the actual TimeWindow interval.
 NOTE: If the PAV's TimeWindo is configured to "Auto", the system decides the appropriate TimeWindow.*
- [_PAV2250AFUNC](#) int [PAV2250A_GetSignalInputOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetSignalInputOption sends a command to the PAV 2250A requesting the current Signal Input Value to the PAV2250A device.
- int [PAV2250A_GetSignalInputOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetSignalInputOptionText sends the command requesting the current Signal Input Text value to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetSignalInputOption sends the command to set the current Signal Input Value to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputFront](#) (int nPAVNo)
PAV2250A_SetSignalInputFront sends the command to set the current Signal Input Value to the Front Panel.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputBack](#) (int nPAVNo)
PAV2250A_SetSignalInputBack sends the command to set the current Signal Input Value to the Back Panel.
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetMainDisplayOption sends the command requesting the current Main Display Index to the PAV2250A device. Main Display can either be 0 ("Independent View") or 1 ("Linked View")
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetMainDisplayOptionText sends the command requesting the current Main Display Option Text to the PAV2250A device. Main Display can either be "Independent View" or "Linked View".
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetMainDisplayOption sends the command to set the Main Display Index Value to the PAV2250A device. Main Display can either be 0 ("Independent View") or 1 ("Linked View").
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayIndependent](#) (int nPAVNo)
PAV2250A_SetMainDisplayIndependent sends the command to set the Main Display Value to Independent View.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayLinked](#) (int nPAVNo)
PAV2250A_SetMainDisplayLinked sends the command to set the Main Display Value to Linked View.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetTimeDisplayOption sends the command requesting the current Time Display Option Index to the PAV2250A device. Time Display can either be 0 ("AM/PM") or 1 ("Military")
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetTimeDisplayOptionText sends the command requesting the current Time Display Option Text to the PAV2250A device. Time Display can either be "AM/PM" or "Military".
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetTimeDisplayOption sends the command to set the Time Display Option Value to the PAV2250A device. Time Display can either be 0 ("AM/PM") or 1 ("Military")
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayAMPM](#) (int nPAVNo)
PAV2250A_SetTimeDisplayAMPM sends the command to set the Time Display Option Value to AM/PM.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayMilitary](#) (int nPAVNo)
PAV2250A_SetTimeDisplayMilitary sends the command to set the Time Display Option Value to Military.
- [_PAV2250AFUNC](#) int [PAV2250A_GetDateDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetDateDisplayOption
- [_PAV2250AFUNC](#) int [PAV2250A_GetDateDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetDateDisplayOptionText sends the command requesting the current Date Display Option Index to the PAV2250A device. Date Display can either be "Text Date"(MON/DD/YYYY) or "Numeric Date"(01/01/11).
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetDateDisplayOption sends the command to set the Date Display Option Value to the PAV2250A device. Date Display can either be 0 ("Text Date") or 1 ("Numeric Date")
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayText](#) (int nPAVNo)

- PAV2250A_SetDateDisplayText* sends the command to set the Date Display Option Value to Text format.
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayNumeric](#) (int nPAVNo)
PAV2250A_SetDateDisplayNumeric sends the command to set the Date Display Option Value to Numeric format.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetAutoSaveOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetAutoSaveOption sends the command requesting the current Auto Save Option Index to the PAV2250-A device. Auto Save Options can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_GetAutoSaveOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetAutoSaveOptionText
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetAutoSaveOption sends the command to set the Auto Save Option Index to the PAV2250A device. Auto Save can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveEnable](#) (int nPAVNo)
PAV2250A_SetAutoSaveEnable sends the command to set the Auto Save Option Index to Enabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveDisable](#) (int nPAVNo)
PAV2250A_SetAutoSaveDisable sends the command to set the Auto Save Option Index to Disabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetAutoUnitsOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetAutoUnitsOption sends the command requesting the current Auto Units Option Index to the PAV2250-A device. Auto Units Options can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_GetAutoUnitsOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetAutoUnitsOptionText sends the command requesting the current Auto Units Option Index to the PAV2250A device. Auto Units can either be "Enabled" or "Disabled"
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetAutoUnitsOption sends the command to set the Auto Units Option Index to the PAV2250A device. Auto Units can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsEnable](#) (int nPAVNo)
PAV2250A_SetAutoUnitsEnable sends the command to set the Auto Units Option Index to Enabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsDisable](#) (int nPAVNo)
PAV2250A_SetAutoUnitsDisable sends the command to set the Auto Units Option Index to Disabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetTouchscreenOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetTouchscreenOption sends the command requesting the current Touchscreen Option Index to the PAV2250A device. Touchscreen Options can either be 0 ("Disabled") or 1 ("Enabled")
 - [_PAV2250AFUNC](#) int [PAV2250A_GetTouchscreenOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetTouchscreenOptionText sends the command requesting the current Touchscreen Option Index to the PAV2250A device. Touchscreen can either be "Enabled" or "Disabled"
 - [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetTouchscreenOption sends the command to set the Touchscreen Option Index to the PAV2250A device. Touchscreen can either be 0 ("Disabled") or 1 ("Enabled").
 - [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenEnable](#) (int nPAVNo)
PAV2250A_SetTouchscreenEnable sends the command to set the Touchscreen Option Index to Enabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenDisable](#) (int nPAVNo)
PAV2250A_SetTouchscreenDisable sends the command to set the Touchscreen Option Index to Disabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetNullMeterRangePercent](#) (int nPAVNo, float fRangePercent)
PAV2250A_SetNullMeterRangePercent sends the command to set the Null Meter Range Percent option.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetNullMeterRangePercent](#) (int nPAVNo, float *pfRangePercent)
PAV2250A_GetNullMeterRangePercent sends the command to get the Null Meter Range Percent option.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetTabView](#) (int nPAVNo, int *pnTabIndex)
PAV2250A_GetTabView sends the command requesting the current Tab View Index to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetTabViewText](#) (int nPAVNo, char *pszTabViewText)
PAV2250A_GetTabViewText sends the command requesting the current Tab View text to the PAV2250A device. Text is reflective of the currently selected tab label. (Main, Harmonics, Quad View)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetTabView](#) (int nPAVNo, int nTabIndex)
PAV2250A_SetTabView sends the command to set the Tab View Index to the PAV2250A device. Tab View Index should be reflective of which tab the PAV should display as the active tab. Index is zero-based.

- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewMain](#) (int nPAVNo)
PAV2250A_SetTabViewMain sends the command to set the Tab View to the Main tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewReference](#) (int nPAVNo)
PAV2250A_SetTabViewReference sends the command to set the Tab View to the Reference tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewHarmonics](#) (int nPAVNo)
PAV2250A_SetTabViewHarmonics sends the command to set the Tab View to the Harmonics tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewCustom](#) (int nPAVNo)
PAV2250A_SetTabViewCustom sends the command to set the Tab View to the Custom View tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewLVDT](#) (int nPAVNo)
PAV2250A_SetTabViewLVDT sends the command to set the Tab View to the LVDT View tab.
- [_PAV2250AFUNC](#) int [PAV2250A_GetReadMode](#) (int nPAVNo, int *pnReadModeIndex)
PAV2250A_GetReadMode sends the command requesting the current Read Mode Index to the PAV2250A device. Read Mode index indicates whether the PAV is showing SIG/REF, REF, SIG or REF/SIG. Index is zero-based.
- [_PAV2250AFUNC](#) int [PAV2250A_GetReadModeText](#) (int nPAVNo, char *pszReadModeText)
PAV2250A_GetReadModeText sends the command requesting the current Read Mode text to the PAV2250A device. Text is reflective of the currently selected Reading Mode (SIG/REF, REF, SIG, or REF/SIG).
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadMode](#) (int nPAVNo, int nReadModeIndex)
PAV2250A_SetReadMode sends the command to set the Read Mode Index to the PAV2250A device. Read Mode index indicates whether the PAV is showing SIG/REF, REF, SIG or REF/SIG. Index is zero-based.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeSigRef](#) (int nPAVNo)
PAV2250A_SetReadModeSigRef sends the command to set the Read Mode to Sig/Ref.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeRefRef](#) (int nPAVNo)
PAV2250A_SetReadModeRefRef sends the command to set the Read Mode to Ref.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeSigSig](#) (int nPAVNo)
PAV2250A_SetReadModeSigSig sends the command to set the Read Mode to Sig.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeRefSig](#) (int nPAVNo)
PAV2250A_SetReadModeRefSig sends the command to set the Read Mode to Ref/Sig.
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainView](#) (int nPAVNo, int *pnMainViewIndex)
PAV2250A_GetMainView sends the command requesting the current Main View Index to the PAV2250A device. Main View index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainViewText](#) (int nPAVNo, char *pszMainViewText)
PAV2250A_GetMainViewText sends the command requesting the current Main View text to the PAV2250A device. Text is reflective of the currently selected Main View: (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainView](#) (int nPAVNo, int nMainViewIndex)
PAV2250A_SetMainView sends the command to set the Main View Index to the PAV2250A device. Main View index indicates the view currently shown in the main display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewFundMag](#) (int nPAVNo)
PAV2250A_SetMainViewFundMag sends the command to set the Main View to Fundamental Magnitude.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewInPhase](#) (int nPAVNo)
PAV2250A_SetMainViewInPhase sends the command to set the Main View to In Phase.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewQuad](#) (int nPAVNo)
PAV2250A_SetMainViewQuad sends the command to set the Main View to Quadrature.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewPhase](#) (int nPAVNo)
PAV2250A_SetMainViewPhase sends the command to set the Main View to Phase.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewTHD](#) (int nPAVNo)
PAV2250A_SetMainViewTHD sends the command to set the Main View to THD.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewSigVolt](#) (int nPAVNo)
PAV2250A_SetMainViewSigVolt sends the command to set the Main View to Signal Voltage.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewRefVolt](#) (int nPAVNo)

- PAV2250A_SetMainViewRefVolt* sends the command to set the Main View to Reference Voltage.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHoldDataState](#) (int nPAVNo, bool *pbHoldData)
PAV2250A_GetHoldDataState sends the command to get the Hold Data State to the PAV2250A device. Hold Data when true indicates no screen refreshes are taking place.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetHoldDataState](#) (int nPAVNo, bool bHoldData)
PAV2250A_SetHoldDataState sends the command to set the Hold Data State to the PAV2250A device. Hold Data when true indicates no screen refreshes are taking place.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetScreenBrightness](#) (int nPAVNo, int *pnBrightness)
PAV2250A_GetScreenBrightness sends the command requesting the current Screen Brightness to the PAV2250A device. Screen Brightness indicates how bright the screen is illuminated. Valid values are between 15-100.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetScreenBrightness](#) (int nPAVNo, int nBrightness)
PAV2250A_SetScreenBrightness sends the command setting the current Screen Brightness on the PAV2250A device. Screen Brightness indicates how bright the screen is illuminated. Valid values are between 15-100.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView1](#) (int nPAVNo, int *pnCustView1Index)
PAV2250A_GetCustView1 sends the command requesting the current Quad View 1 Index to the PAV2250A device. Quad View 1 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView1Text](#) (int nPAVNo, char *pszCustView1Text)
PAV2250A_GetCustView1Text sends the command requesting the current Quad View 1 Text to the PAV2250A device. Quad View 1 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1](#) (int nPAVNo, int nCustView1Index)
PAV2250A_SetCustView1 sends the command to set the Quad View 1 Index to the PAV2250A device. Quad View 1 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1FundMag](#) (int nPAVNo)
PAV2250A_SetCustView1FundMag sends the command to set the Quad View 1 slot to Fundamental Magnitude.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1InPhase](#) (int nPAVNo)
PAV2250A_SetCustView1InPhase sends the command to set the Quad View 1 slot to In Phase.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1Quad](#) (int nPAVNo)
PAV2250A_SetCustView1Quad sends the command to set the Quad View 1 slot to Quadrature.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1Phase](#) (int nPAVNo)
PAV2250A_SetCustView1Phase sends the command to set the Quad View 1 slot to Phase Angle.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1THD](#) (int nPAVNo)
PAV2250A_SetCustView1THD sends the command to set the Quad View 1 slot to THD.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1SigVolt](#) (int nPAVNo)
PAV2250A_SetCustView1SigVolt sends the command to set the Quad View 1 slot to Signal Voltage.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1RefVolt](#) (int nPAVNo)
PAV2250A_SetCustView1RefVolt sends the command to set the Quad View 1 slot to Reference Voltage.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView2](#) (int nPAVNo, int *pnCustView2Index)
PAV2250A_GetCustView2 sends the command requesting the current Quad View 2 Index to the PAV2250A device. Quad View 2 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView2Text](#) (int nPAVNo, char *pszCustView2Text)
PAV2250A_GetCustView2Text sends the command requesting the current Quad View 2 Text to the PAV2250A device. Quad View 2 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2](#) (int nPAVNo, int nCustView2Index)
PAV2250A_SetCustView2 sends the command to set the Quad View 2 Index to the PAV2250A device. Quad View 2 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2FundMag](#) (int nPAVNo)
PAV2250A_SetCustView2FundMag sends the command to set the Quad View 2 slot to Fundamental Magnitude.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2InPhase](#) (int nPAVNo)

- PAV2250A_SetCustView2InPhase* sends the command to set the Quad View 2 slot to In Phase.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2Quad](#) (int nPAVNo)

PAV2250A_SetCustView2Quad sends the command to set the Quad View 2 slot to Quadrature.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2Phase](#) (int nPAVNo)

PAV2250A_SetCustView2Phase sends the command to set the Quad View 2 slot to Phase Angle.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2THD](#) (int nPAVNo)

PAV2250A_SetCustView2THD sends the command to set the Quad View 2 slot to THD.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2SigVolt](#) (int nPAVNo)

PAV2250A_SetCustView2SigVolt sends the command to set the Quad View 2 slot to Signal Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2RefVolt](#) (int nPAVNo)

PAV2250A_SetCustView2RefVolt sends the command to set the Quad View 2 slot to Reference Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView3](#) (int nPAVNo, int *pnCustView3Index)

PAV2250A_GetQuadVies3 sends the command requesting the current Quad View 3 Index to the PAV2250A device. Quad View 3 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView3Text](#) (int nPAVNo, char *pszCustView3Text)

PAV2250A_GetCustView3Text sends the command requesting the current Quad View 3 Text to the PAV2250A device. Quad View 3 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3](#) (int nPAVNo, int nCustView3Index)

PAV2250A_SetCustView3 sends the command to set the Quad View 3 Index to the PAV2250A device. Quad View 3 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3FundMag](#) (int nPAVNo)

PAV2250A_SetCustView3FundMag sends the command to set the Quad View 3 slot to Fundamental Magnitude.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3InPhase](#) (int nPAVNo)

PAV2250A_SetCustView3InPhase sends the command to set the Quad View 3 slot to In Phase.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3Quad](#) (int nPAVNo)

PAV2250A_SetCustView3Quad sends the command to set the Quad View 3 slot to Quadrature.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3Phase](#) (int nPAVNo)

PAV2250A_SetCustView3Phase sends the command to set the Quad View 3 slot to Phase Angle.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3THD](#) (int nPAVNo)

PAV2250A_SetCustView3THD sends the command to set the Quad View 3 slot to THD.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3SigVolt](#) (int nPAVNo)

PAV2250A_SetCustView3SigVolt sends the command to set the Quad View 3 slot to Signal Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3RefVolt](#) (int nPAVNo)

PAV2250A_SetCustView3RefVolt sends the command to set the Quad View 3 slot to Reference Voltage.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView4](#) (int nPAVNo, int *pnCustView4Index)

PAV2250A_GetCustView4 sends the command requesting the current Quad View 4 Index to the PAV2250A device. Quad View 4 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView4Text](#) (int nPAVNo, char *pszCustView4Text)

PAV2250A_GetCustView4Text sends the command requesting the current Quad View 4 Text to the PAV2250A device. Quad View 4 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4](#) (int nPAVNo, int nCustView4Index)

PAV2250A_SetCustView4 sends the command to set the Quad View 4 Index to the PAV2250A device. Quad View 4 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4FundMag](#) (int nPAVNo)

PAV2250A_SetCustView4FundMag sends the command to set the Quad View 4 slot to Fundamental Magnitude.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4InPhase](#) (int nPAVNo)

PAV2250A_SetCustView4InPhase sends the command to set the Quad View 4 slot to In Phase.

- [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4Quad](#) (int nPAVNo)
PAV2250A_SetCustView4Quad sends the command to set the Quad View 4 slot to Quadrature.
- [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4Phase](#) (int nPAVNo)
PAV2250A_SetCustView4Phase sends the command to set the Quad View 4 slot to Phase Angle.
- [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4THD](#) (int nPAVNo)
PAV2250A_SetCustView4THD sends the command to set the Quad View 4 slot to THD.
- [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4SigVolt](#) (int nPAVNo)
PAV2250A_SetCustView4SigVolt sends the command to set the Quad View 4 slot to Signal Voltage.
- [_PAV2250AFUNC](#) int [PAV2250A_SetCustView4RefVolt](#) (int nPAVNo)
PAV2250A_SetCustView4RefVolt sends the command to set the Quad View 4 slot to Reference Voltage.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexConfig](#) (int nPAVNo, int nViewIndex, char *pszViewConfig)
PAV2250A_GetViewIndexConfig sends the command to get the view configuration for the specified view index. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewFundMagConfig sends the command to get the view configuration for the Fundamental Magnitude. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewInPhaseConfig sends the command to get the view configuration for the In Phase component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewQuadConfig sends the command to get the view configuration for the Quad component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewPhaseConfig sends the command to get the view configuration for the Phase component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewTHDConfig sends the command to get the view configuration for the THD component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewSigVoltConfig sends the command to get the view configuration for the Sig Volt component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewRefVoltConfig sends the command to get the view configuration for the Ref Volt component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewSigOffsetConfig sends the command to get the view configuration for the Sig Offset component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewTotalRatioConfig sends the command to get the view configuration for the Total Ratio component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewFrequencyConfig sends the command to get the view configuration for the Frequency component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainConfig](#) (int nPAVNo, char *pszViewConfig)

PAV2250A_GetViewMainConfig sends the command to get the view configuration for the Main view. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewFundMagMaxFieldWidth sends the command to set the max field width for Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewFundMagMaxFieldWidth sends the command to get the max field width for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewInPhaseMaxFieldWidth sends the command to set the max field width for In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewInPhaseMaxFieldWidth sends the command to get the max field width for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewQuadMaxFieldWidth sends the command to set the max field width for Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewQuadMaxFieldWidth sends the command to get the max field width for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhaseMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewPhaseMaxFieldWidth sends the command to set the max field width for Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewPhaseMaxFieldWidth sends the command to get the max field width for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewTHDMaxFieldWidth sends the command to set the max field width for THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewTHDMaxFieldWidth sends the command to get the max field width for the THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewSigVoltMaxFieldWidth sends the command to set the max field width for Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewSigVoltMaxFieldWidth sends the command to get the max field width for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewRefVoltMaxFieldWidth sends the command to set the max field width for Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewRefVoltMaxFieldWidth sends the command to get the max field width for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewSigOffsetMaxFieldWidth sends the command to set the max field width for Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewSigOffsetMaxFieldWidth sends the command to get the max field width for the SigOffset view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewTotalRatioMaxFieldWidth sends the command to set the max field width for Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewTotalRatioMaxFieldWidth sends the command to get the max field width for the TotalRatio view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewFrequencyMaxFieldWidth sends the command to set the max field width for Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewFrequencyMaxFieldWidth sends the command to get the max field width for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewMainMaxFieldWidth sends the command to set the max field width for Main view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewMainMaxFieldWidth sends the command to get the max field width for the Main view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagV](#) (int nPAVNo)
PAV2250A_SetViewFundMagV sends the command to set Fundamental Magnitude units to V.

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagMV](#) (int nPAVNo)
PAV2250A_SetViewFundMagMV sends the command to set Fundamental Magnitude units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagPercent](#) (int nPAVNo)
PAV2250A_SetViewFundMagPercent sends the command to set Fundamental Magnitude units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagDB](#) (int nPAVNo)
PAV2250A_SetViewFundMagDB sends the command to set Fundamental Magnitude units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagRatio](#) (int nPAVNo)
PAV2250A_SetViewFundMagRatio sends the command to set Fundamental Magnitude units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewFundMagUnits sends the command to get the view units index for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewFundMagUnitsText sends the command to get the view units for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseV](#) (int nPAVNo)
PAV2250A_SetViewInPhaseV sends the command to set InPhase units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseMV](#) (int nPAVNo)
PAV2250A_SetViewInPhaseMV sends the command to set In Phase units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhasePercent](#) (int nPAVNo)
PAV2250A_SetViewInPhasePercent sends the command to set In Phase units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseDB](#) (int nPAVNo)
PAV2250A_SetViewInPhaseDB sends the command to set In Phase units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseRatio](#) (int nPAVNo)
PAV2250A_SetViewInPhaseRatio sends the command to set Fundamental In Phase units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewInPhaseUnits sends the command to get the view units index for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewInPhaseUnitsText sends the command to get the view units for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadV](#) (int nPAVNo)
PAV2250A_SetViewQuadV sends the command to set Quad units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadMV](#) (int nPAVNo)
PAV2250A_SetViewQuadMV sends the command to set Quad units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadPercent](#) (int nPAVNo)
PAV2250A_SetViewQuadPercent sends the command to set Quad units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadDB](#) (int nPAVNo)
PAV2250A_SetViewQuadDB sends the command to set Quad units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadRatio](#) (int nPAVNo)
PAV2250A_SetViewQuadRatio sends the command to set Quad units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewQuadUnits sends the command to get the view units index for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewQuadUnitsText sends the command to get the view units for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhase360](#) (int nPAVNo)
PAV2250A_SetViewPhase360 sends the command to set Phase units to +-360.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhase180](#) (int nPAVNo)
PAV2250A_SetViewPhase180 sends the command to set Phase units to +-180.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewPhaseUnits sends the command to get the view units index for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewPhaseUnitsText sends the command to get the view units for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDPercent](#) (int nPAVNo)

- PAV2250A_SetViewTHDPercent sends the command to set THD units to %.*

 - [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDDB](#) (int nPAVNo)

PAV2250A_SetViewTHDDB sends the command to set THD units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewTHDUnits sends the command to get the view units index for the THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewTHDUnitsText sends the command to get the view units for the THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltV](#) (int nPAVNo)

PAV2250A_SetViewSigVoltV sends the command to set Signal Voltage units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltMV](#) (int nPAVNo)

PAV2250A_SetViewSigVoltMV sends the command to set Signal Voltage units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewSigVoltUnits sends the command to get the view units index for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewSigVoltUnitsText sends the command to get the view units for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltV](#) (int nPAVNo)

PAV2250A_SetViewRefVoltV sends the command to set Reference Voltage units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltMV](#) (int nPAVNo)

PAV2250A_SetViewRefVoltMV sends the command to set Reference Voltage units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewRefVoltUnits sends the command to get the view units index for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewRefVoltUnitsText sends the command to get the view units for the RefVolt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetV](#) (int nPAVNo)

PAV2250A_SetViewSigOffsetV sends the command to set Signal Offset units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetMV](#) (int nPAVNo)

PAV2250A_SetViewSigOffsetMV sends the command to set Signal Offset units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewSigOffsetUnits sends the command to get the view units index for the Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewSigOffsetUnitsText sends the command to get the view units for the Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioPercent](#) (int nPAVNo)

PAV2250A_SetViewTotalRatioPercent sends the command to set Total Ratio units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioDB](#) (int nPAVNo)

PAV2250A_SetViewTotalRatioDB sends the command to set Total Ratio units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioRatio](#) (int nPAVNo)

PAV2250A_SetViewTotalRatioRatio sends the command to set Total Ratio units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewTotalRatioUnits sends the command to get the view units index for the Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewTotalRatioUnitsText sends the command to get the view units for the Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyHZ](#) (int nPAVNo)

PAV2250A_SetViewFrequencyHZ sends the command to set Frequency units to Hz.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyKHZ](#) (int nPAVNo)

PAV2250A_SetViewFrequencyKHZ sends the command to set Frequency units to KHz.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyUnits](#) (int nPAVNo, int *pnUnits)

PAV2250A_GetViewFrequencyUnits sends the command to get the view units index for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyUnitsText](#) (int nPAVNo, char *pszUnits)

PAV2250A_GetViewFrequencyUnitsText sends the command to get the view units for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainV](#) (int nPAVNo)

- PAV2250A_SetViewMainV* sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)
- `_PAV2250AFUNC` int `PAV2250A_SetViewMainMV` (int nPAVNo)
PAV2250A_SetViewMainMV sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)
 - `_PAV2250AFUNC` int `PAV2250A_SetViewMainPercent` (int nPAVNo)
PAV2250A_SetViewMainPercent sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)
 - `_PAV2250AFUNC` int `PAV2250A_SetViewMainDB` (int nPAVNo)
PAV2250A_SetViewMainDB sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)
 - `_PAV2250AFUNC` int `PAV2250A_SetViewMainRatio` (int nPAVNo)
PAV2250A_SetViewMainRatio sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)
 - `_PAV2250AFUNC` int `PAV2250A_SetViewMain360` (int nPAVNo)
PAV2250A_SetViewMain360 sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)
 - `_PAV2250AFUNC` int `PAV2250A_SetViewMain180` (int nPAVNo)
PAV2250A_SetViewMain180 sends the command to set the view units for the Main view. (only important when main view mode is set to `INDEPENDENT_VIEW`)
 - `_PAV2250AFUNC` int `PAV2250A_GetViewMainUnits` (int nPAVNo, int *pnUnits)
PAV2250A_GetViewMainUnits sends the command to get the view units index for the Main view.
 - `_PAV2250AFUNC` int `PAV2250A_GetViewMainUnitsText` (int nPAVNo, char *pszUnits)
PAV2250A_GetViewMainUnitsText sends the command to get the view units for the Main view.
 - `_PAV2250AFUNC` int `PAV2250A_SetViewFundMagOffset` (int nPAVNo, float fOffset)
PAV2250A_SetViewFundMagOffset sends the command to set the offset for the Fundamental Magnitude view.
 - `_PAV2250AFUNC` int `PAV2250A_GetViewFundMagOffset` (int nPAVNo, float *pfOffset)
PAV2250A_GetViewFundMagOffset sends the command to get the offset for the Fundamental Magnitude view.
 - `_PAV2250AFUNC` int `PAV2250A_SetViewInPhaseOffset` (int nPAVNo, float fOffset)
PAV2250A_SetViewInPhaseOffset sends the command to set the offset for the In Phase view.
 - `_PAV2250AFUNC` int `PAV2250A_GetViewInPhaseOffset` (int nPAVNo, float *pfOffset)
PAV2250A_GetViewInPhaseOffset sends the command to get the offset for the In Phase view.
 - `_PAV2250AFUNC` int `PAV2250A_SetViewQuadOffset` (int nPAVNo, float fOffset)
PAV2250A_SetViewQuadOffset sends the command to set the offset for the Quad view.
 - `_PAV2250AFUNC` int `PAV2250A_GetViewQuadOffset` (int nPAVNo, float *pfOffset)
PAV2250A_GetViewQuadOffset sends the command to get the offset for the Quad view.
 - `_PAV2250AFUNC` int `PAV2250A_SetViewPhaseOffset` (int nPAVNo, float fOffset)
PAV2250A_SetViewPhaseOffset sends the command to set the offset for the Phase view.
 - `_PAV2250AFUNC` int `PAV2250A_GetViewPhaseOffset` (int nPAVNo, float *pfOffset)
PAV2250A_GetViewPhaseOffset sends the command to get the offset for the Phase view.
 - `_PAV2250AFUNC` int `PAV2250A_SetViewSigVoltOffset` (int nPAVNo, float fOffset)
PAV2250A_SetViewSigVoltOffset sends the command to set the offset for the Sig Volt view.
 - `_PAV2250AFUNC` int `PAV2250A_GetViewSigVoltOffset` (int nPAVNo, float *pfOffset)
PAV2250A_GetViewSigVoltOffset sends the command to get the offset for the Sig Volt view.
 - `_PAV2250AFUNC` int `PAV2250A_SetViewRefVoltOffset` (int nPAVNo, float fOffset)
PAV2250A_SetViewRefVoltOffset sends the command to set the offset for the Ref Volt view.
 - `_PAV2250AFUNC` int `PAV2250A_GetViewRefVoltOffset` (int nPAVNo, float *pfOffset)
PAV2250A_GetViewRefVoltOffset sends the command to get the offset for the Ref Volt view.
 - `_PAV2250AFUNC` int `PAV2250A_SetViewMainOffset` (int nPAVNo, float fOffset)
PAV2250A_SetViewMainOffset sends the command to set the offset for the Main view. (only comes into play when the main view option is set to `"INDEPENDENT_VIEW"`)
 - `_PAV2250AFUNC` int `PAV2250A_GetViewMainOffset` (int nPAVNo, float *pfOffset)

- PAV2250A_GetViewMainOffset* sends the command to get the offset for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewFundMagScale sends the command to set the scale for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewFundMagScale sends the command to get the scale for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewInPhaseScale sends the command to set the scale for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewInPhaseScale sends the command to get the scale for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewQuadScale sends the command to set the scale for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewQuadScale sends the command to get the scale for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhaseScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewPhaseScale sends the command to set the scale for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewPhaseScale sends the command to get the scale for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewSigVoltScale sends the command to set the scale for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewSigVoltScale sends the command to get the scale for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewRefVoltScale sends the command to set the scale for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewRefVoltScale sends the command to get the scale for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewMainScale sends the command to set the scale for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewMainScale sends the command to get the scale for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexMaxFieldWidth](#) (int nPAVNo, int nViewIndex, int nMaxFieldWidth)
PAV2250A_SetViewIndexMaxFieldWidth sends the command to set the max field width for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexMaxFieldWidth](#) (int nPAVNo, int nViewIndex, int *pnMaxFieldWidth)
PAV2250A_GetViewIndexMaxFieldWidth sends the command to get the max field width for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexUnits](#) (int nPAVNo, int nViewIndex, int nUnits)
PAV2250A_SetViewIndexUnits sends the command to set the units for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexUnits](#) (int nPAVNo, int nViewIndex, int *pnUnits)
PAV2250A_GetViewIndexUnits sends the command to get the units for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexUnitsText](#) (int nPAVNo, int nViewIndex, char *pszUnits)
PAV2250A_GetViewIndexUnitsText sends the command to get the view units for the view text specified by the index.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexOffset](#) (int nPAVNo, int nViewIndex, float fOffset)
PAV2250A_SetViewIndexOffset sends the command to set the offset for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexOffset](#) (int nPAVNo, int nViewIndex, float *pfOffset)
PAV2250A_GetViewIndexOffset sends the command to get the offset for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexScale](#) (int nPAVNo, int nViewIndex, float fScale)
PAV2250A_SetViewIndexScale sends the command to set the scale for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexScale](#) (int nPAVNo, int nViewIndex, float *pfScale)

- PAV2250A_GetViewIndexScale* sends the command to get the scale for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRef](#) (int nPAVNo, int *pnViewIndex)
PAV2250A_GetRef sends the command requesting the current Ref Voltage View Index to the PAV2250A device. Index options (1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefText](#) (int nPAVNo, char *pszViewText)
PAV2250A_GetRef sends the command requesting the current Ref Voltage View Text to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRef](#) (int nPAVNo, int nViewIndex)
PAV2250A_SetRef sends the command to set the Ref Voltage View Index to the PAV2250A device. Index options (1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalSum](#) (int nPAVNo)
PAV2250A_SetRefTotalSum sends the command to set the Ref Voltage View to TotalSum to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalRMS_AC](#) (int nPAVNo)
PAV2250A_SetRefTotalRMS_AC
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalRMS_ACDC](#) (int nPAVNo)
PAV2250A_SetRefTotalRMS_ACDC sends the command to set the Ref Voltage View to TotalRMS_ACDC to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefDC](#) (int nPAVNo)
PAV2250A_SetRefDC sends the command to set the Ref Voltage View to DC to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalSum](#) (int nPAVNo, float *pfRefTotalSum)
PAV2250A_GetRefTotalSum sends the command to get the Ref Voltage Total Sum value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalRMS_AC](#) (int nPAVNo, float *pfRefRMS)
PAV2250A_GetRefTotalRMS_AC sends the command to get the Ref Voltage Total RMS AC value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalRMS_ACDC](#) (int nPAVNo, float *pfRefRMS_ACDC)
PAV2250A_GetRefTotalRMS_ACDC sends the command to get the Ref Voltage Total RMS AC+DC value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefDC](#) (int nPAVNo, float *pfRefDC)
PAV2250A_GetRefDC sends the command to get the Ref Voltage DC value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSig](#) (int nPAVNo, int *pnViewIndex)
PAV2250A_GetSig sends the command requesting the current Sig Voltage View Index to the PAV2250A device. Index options (0=TotalSum 1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigText](#) (int nPAVNo, char *pszViewText)
PAV2250A_GetSig sends the command requesting the current Sig Voltage View Text to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSig](#) (int nPAVNo, int nViewIndex)
PAV2250A_SetSig sends the command to set the Sig Voltage View Index to the PAV2250A device. Index options (0=TotalSum, 1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalSum](#) (int nPAVNo)
PAV2250A_SetSigTotalSum sends the command to set the Sig Voltage View to TotalSum to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalRMS_AC](#) (int nPAVNo)
PAV2250A_SetSigTotalRMS_AC sends the command to set the Sig Voltage View to TotalRMS_AC to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalRMS_ACDC](#) (int nPAVNo)
PAV2250A_SetSigTotalRMS_ACDC sends the command to set the Sig Voltage View to TotalRMS_ACDC to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigDC](#) (int nPAVNo)
PAV2250A_SetSigDC sends the command to set the Sig Voltage View to DC to the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalSum](#) (int nPAVNo, float *pfSigTotalSum)
PAV2250A_GetSigTotalSum sends the command to get the Sig Voltage Total Sum value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalRMS_AC](#) (int nPAVNo, float *pfSigRMS)
PAV2250A_GetSigTotalRMS_AC sends the command to get the Sig Voltage Total RMS AC value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalRMS_ACDC](#) (int nPAVNo, float *pfSigRMS_ACDC)

- PAV2250A_GetSigTotalRMS_ACDC sends the command to get the Sig Voltage Total RMS AC+DC value from the PAV2250A device.*

 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigDC](#) (int nPAVNo, float *pfSigDC)
PAV2250A_GetSigDC sends the command to get the Sig Voltage DC value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigOffset](#) (int nPAVNo, float *pfSigOffset)
PAV2250A_GetSigOffset sends the command to get the Sig Offset value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetTotalRatio](#) (int nPAVNo, float *pfTotalRatio)
PAV2250A_GetTotalRatio sends the command to get the Total Ratio value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetTHD](#) (int nPAVNo, float *pfTHD)
PAV2250A_GetTHD sends the command to get the THD value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetFrequency](#) (int nPAVNo, float *pfFreq)
PAV2250A_GetFrequency sends the command to get the Frequency value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetSampleRateIndex](#) (int nPAVNo, int *pnSampleRateIndex)
PAV2250A_GetSampleRateIndex sends the command to get the Sample Rate Index value from the PAV2250A device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedDataState](#) (int nPAVNo, bool *pbDataBufferEnabled, bool *pbDataBufferComplete, int *pnBufferedDataType, int *pnSampleRate, int *pnBufferSize, bool *pbDataBufferReady)
FOR INTERNAL USE ONLY! PAV2250A_GetBufferedDataState is responsible for returning data buffering state information.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetBufferedDataState](#) (int nPAVNo, bool bDataBufferEnabled, int nBufferedDataType, int nSampleRate, int nBufferSize)
FOR INTERNAL USE ONLY! PAV2250A_SetBufferedDataState is responsible for setting the buffer collection parameters of DataType, SampleRate, BufferSize and BufferEnabled.
 - [_PAV2250AFUNC](#) int [PAV2250A_BufferCapture](#) (int nPAVNo, int nDataType, int nSampleRate, int nSize)
FOR INTERNAL USE ONLY! PAV2250A_BufferCapture is responsible for Starting a data buffer capture.
 - [_PAV2250AFUNC](#) int [PAV2250A_BufferStop](#) (int nPAVNo)
FOR INTERNAL USE ONLY! PAV2250A_BufferStop is responsible for stopping all buffer captures for the given PAV.
 - [_PAV2250AFUNC](#) int [PAV2250A_BufferGet](#) (int nPAVNo, int nStart, int nCount, char *pszData)
FOR INTERNAL USE ONLY! PAV2250A_BufferGet is responsible for retrieving a captured buffer.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedPageIndex](#) (int nPAVNo, int *pnPageIndex)
FOR INTERNAL USE ONLY! PAV2250A_GetBufferedPageIndex returns the current buffered page index.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetBufferedPageIndex](#) (int nPAVNo, int nPageIndex)
FOR INTERNAL USE ONLY! PAV2250A_SetBufferedPageIndex is responsible for setting the current page buffer.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedData](#) (int nPAVNo, int nPageIndex, int nRequestedBufferSize, int nTotalOverallDataCnt, int *pnElementCnt, int *paBufferData)
FOR INTERNAL USE ONLY! PAV2250A_GetBufferedData is responsible for retrieving the buffered data.
 - int [HandleBufferedDataState](#) (int nPAVNo, bool *pbDataBufferEnabled, bool *pbDataBufferComplete, int *pnBufferedDataType, int *pnSampleRate, int *pnBufferSize, bool *pbDataBufferReady)
FOR INTERNAL USE ONLY! HandleBufferedDataState returns Data Buffering configuration information.
 - int [HandleBufferDataSetup](#) (int nPAVNo, int nStartRec, int nEndRec)
FOR INTERNAL USE ONLY! HandleBufferDataSetup is responsible for setting some parameters needed to capture data.
 - int [HandleBufferedDataValues](#) (int nPAVNo, int nStartRec, int *pnDataCnt, int *paBufferData)
FOR INTERNAL USE ONLY! HandleBufferedDataValues is responsible for loading the buffered data values into the output buffer.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefFreq](#) (int nPAVNo, float fFreq)
PAV2250A_SetIntRefFreq sends a command to the PAV 2250A to force the internal reference frequency to the specified value.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefFreq](#) (int nPAVNo, float *pfFreq)
PAV2250A_GetIntRefFreq is responsible for returning internal reference frequency value.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefVolt](#) (int nPAVNo, float fVolt)

PAV2250A_SetIntRefVolt sends a command to the PAV 2250A to force the internal reference voltage to the specified value.

- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefVolt](#) (int nPAVNo, float *pfVolt)
PAV2250A_GetIntRefVolt is responsible for returning internal reference voltage value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefOutputState](#) (int nPAVNo, int nOutputState)
PAV2250A_SetIntRefOutputState is responsible for setting the output state. (OFF/ON).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefOutputState](#) (int nPAVNo, int *pnOutputState)
PAV2250A_GetIntRefOutputState is responsible for returning the index associated with the current reference state output.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefOverCurState](#) (int nPAVNo, int *pnOvrCurState)
PAV2250A_GetIntRefOutputState is responsible for returning the over current state.
- [_PAV2250AFUNC](#) int [PAV2250A_ResetIntRefOverCur](#) (int nPAVNo)
PAV2250A_ResetIntRefOverCur is responsible for resetting the over current flag.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefSenseDir](#) (int nPAVNo, int nSenseDir)
PAV2250A_SetIntRefSenseDir is responsible for setting the sense direction. (Front/Back).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefSenseDir](#) (int nPAVNo, int *pnSenseDir)
PAV2250A_GetIntRefSenseDir is responsible for returning the index associated with the current Sense line direction.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefRemoteSense](#) (int nPAVNo, int nRemoteSense)
PAV2250A_SetRefGenRmtSenseState is responsible for setting the the sense state. (Disable/Enable).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefRemoteSense](#) (int nPAVNo, int *pnRemoteSense)
PAV2250A_GetIntRefRemoteSense is responsible for returning the index associated with the current Sense line state.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefMeasCur](#) (int nPAVNo, int *pnMeasCur)
PAV2250A_GetIntRefMeasCur is responsible for returning the internal reference measured current.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTEnabled](#) (int nPAVNo, bool *pbLVDTEnabled)
PAV2250A_GetLVDTEnabled returns whether or not the PAV's LVDT calculations are enabled or disabled.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTEnabledText](#) (int nPAVNo, char *pszLVDTEnabledText)
PAV2250A_GetLVDTEnabledText retrieves the LVDT enabled text: "Disabled" or "Enabled"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTEnabled](#) (int nPAVNo)
PAV2250A_SetLVDTEnabled enables the LVDT calculations : 1 ("Enabled")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTDisabled](#) (int nPAVNo)
PAV2250A_SetLVDTDisabled disables the LVDT calculations : 0 ("Disabled")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTPosition](#) (int nPAVNo, float *pfPosition)
PAV2250A_GetLVDTPosition retrieves the current LVDT position.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTType](#) (int nPAVNo, int *pnLVDTType)
PAV2250A_GetLVDTType retrieves the LVDT mode type: 2 = 2-Wire, 3 = 3-Wire, 4 = 4-Wire
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTTypeText](#) (int nPAVNo, char *pszLVDTTypeText)
PAV2250A_GetLVDTTypeText retrieves the LVDT mode type text: "2-Wire", "3-Wire", "4-Wire"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType](#) (int nPAVNo, int nLVDTTypeIndex)
PAV2250A_SetLVDTType sets the LVDT mode type: 2 = "2-Wire", 3 = "3-Wire", 4 = "4-Wire"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType2Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType2Wire sets the LVDT mode type to: 2 ("2-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType3Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType3Wire sets the LVDT mode type to: 3 ("3-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType4Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType4Wire sets the LVDT mode type to: 4 ("4-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTSignal](#) (int nPAVNo, int *pnLVDTSignal)
PAV2250A_GetLVDTSignal retrieves the LVDT signal: 0 = Fund, 1 = Total, 2 = INPH
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTSignalText](#) (int nPAVNo, char *pszLVDTSignalText)
PAV2250A_GetLVDTSignalText retrieves the LVDT signal text: "Fund", "Total", "INPH"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignal](#) (int nPAVNo, int nLVDTSignalIndex)

- PAV2250A_SetLVDTSignal* sets the LVDT signal: 0 = "Fund", 1 = "Total", 2 = "INPH"

 - [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalINPH](#) (int nPAVNo)
PAV2250A_SetLVDTSignalINPH sets the LVDT signal to: 2 ("INPH")
 - [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalFund](#) (int nPAVNo)
PAV2250A_SetLVDTSignalFund sets the LVDT signal to: 0 ("Fund")
 - [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalTotal](#) (int nPAVNo)
PAV2250A_SetLVDTSignalTotal sets the LVDT signal to: 1 ("Total")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTVA](#) (int nPAVNo, float *pfLVDTVA)
PAV2250A_GetLVDTVA retrieves the current LVDT VA value.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTVB](#) (int nPAVNo, float *pfLVDTVB)
PAV2250A_GetLVDTVB retrieves the current LVDT VB value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTPOFF](#) (int nPAVNo, float fLVDTPOFF)
PAV2250A_SetLVDTPOFF sends a command to the PAV 2250A to set a Phase Offset of the desired value.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTPOFF](#) (int nPAVNo, float *pfLVDTPOFF)
PAV2250A_GetLVDTPOFF retrieves the current LVDT Phase Offset (POFF) value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTScale](#) (int nPAVNo, float fLVDTScale)
PAV2250A_SetLVDTScale sends a command to the PAV 2250A to set a Scale of the desired value.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTScale](#) (int nPAVNo, float *pfLVDTScale)
PAV2250A_GetLVDTScale retrieves the current LVDT Scale value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDT4WireAlgorithm](#) (int nPAVNo, int *pnLVDT4WireAlgorithm)
PAV2250A_GetLVDT4WireAlgorithm retrieves the LVDT 4-Wire Algorithm: 0 = V(a), V(a) + V(b), 1 = V(a), V(b)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetLVDT4WireAlgorithmText](#) (int nPAVNo, char *pszLVDT4WireAlgorithmText)
PAV2250A_GetLVDT4WireAlgorithmText retrieves the LVDT 4-Wire Algorithm text: "V(a), V(a) + V(b)", "V(a), V(b)"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDT4WireAlgorithm](#) (int nPAVNo, int nLVDT4WireAlgorithmIndex)
PAV2250A_SetLVDT4WireAlgorithm sets the LVDT 4-Wire Algorithm: 0 = "V(a), V(a) + V(b)", 1 = "V(a), V(b)"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDT4WireVA_VAPLUSVB](#) (int nPAVNo)
PAV2250A_SetLVDT4WireVA_VAPLUSVB sets the LVDT 4-Wire Algorithm to: 0 ("V(a), V(a) + V(b)")
 - [_PAV2250AFUNC](#) int [PAV2250A_SetLVDT4WireVA_VB](#) (int nPAVNo)
PAV2250A_SetLVDT4WireVA_VB sets the LVDT 4-Wire Algorithm to: 1 ("V(a), V(b)")
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotal](#) (int nPAVNo, float *pfTotal)
PAV2250A_GetRefTotal retrieves the reference Total value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefFundMag](#) (int nPAVNo, float *pfFundMag)
PAV2250A_GetRefFundMag retrieves the reference Fundamental (Magnitude) value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefInPhase](#) (int nPAVNo, float *pfInPhase)
PAV2250A_GetRefInPhase retrieves the reference InPhase value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefQuad](#) (int nPAVNo, float *pfQuad)
PAV2250A_GetRefQuad retrieves the reference Quad value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefPhase](#) (int nPAVNo, float *pfPhase)
PAV2250A_GetRefPhase retrieves the reference Phase value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTHD](#) (int nPAVNo, float *pfTHD)
PAV2250A_GetRefTHD retrieves the reference THD value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalRatio](#) (int nPAVNo, float *pfTotalRatio)
PAV2250A_GetRefTotalRatio retrieves the reference Total Ratio value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefFundMagRatio](#) (int nPAVNo, float *pfFundMagRatio)
PAV2250A_GetRefFundMagRatio retrieves the reference Fundamental (Magnitude) Ratio value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefInPhaseRatio](#) (int nPAVNo, float *pfInPhaseRatio)
PAV2250A_GetRefInPhaseRatio retrieves the reference InPhase Ratio value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefQuadRatio](#) (int nPAVNo, float *pfQuadRatio)
PAV2250A_GetRefQuadRatio retrieves the reference Quad Ratio value.

Variables

- const double `PAV2250ADII_VERSION` = 1.00

4.1.1 Function Documentation

- 4.1.1.1 `BOOL APIENTRY DIIMain (HANDLE hModule, DWORD ul_reason_for_call, LPVOID lpReserved)`

4.1.2 Variable Documentation

- 4.1.2.1 const double `PAV2250ADII_VERSION` = 1.00

4.2 E:/(BLACKFIN CODE)/Instruments/PAV-2250A/Driver_VS2010/Source/PAV2250ADII/-PAV2250ADII.h File Reference

```
#include "PAV2250AD11_BuildOption.h"
#include <windows.h>
#include <Winnt.h>
#include <string.h>
```

Macros

- #define `_PAV2250ACCLASS` __declspec(dllexport)
- #define `_PAV2250AFUNC` __declspec(dllexport)
- #define `bool` BOOL

Enumerations

- enum `PAV_STATUS` {
`PAV_SUCCESS` = 0, `PAV_ERROR_OPEN_PAV_SESSION` = 1, `PAV_ERROR_PAVNO` = 2, `PAV_ERROR_ADDRS` = 3,
`PAV_ERROR_LANG` = 4, `PAV_ERROR_DATA` = 5, `PAV_ERROR_RANGE` = 6, `PAV_ERROR_WRITE` = 7,
`PAV_ERROR_USB_CONNECTION` = 8, `PAV_ERROR_ETHER_CONNECTION` = 9, `PAV_ERROR_FUNC_NOT_SUPPORTED` = 10, `PAV_ERROR_TRIGGER` = 11,
`PAV_STATUS_LAST` = 12 }
- enum `PAV_IEEE_LANGUAGES` { `PAV2250A_NATIVE` = 0, `PAV2250_LEGACY` = 1 }
- enum `PAV_Tabs` { `PAV_TAB_MAIN` = 0, `PAV_TAB_REFERENCE` = 1, `PAV_TAB_HARMONICS` = 2, `PAV_TAB_CUSTOM` = 3 }
- enum `PAV_ReadModes` { `PAV_READMODE_SIGREF` = 0, `PAV_READMODE_REFREF` = 1, `PAV_READMODE_SIGSIG` = 2, `PAV_READMODE_REFSIG` = 3 }
- enum `PAV_Views` {
`PAV_VIEW_FUNDMAG` = 0, `PAV_VIEW_INPHASE` = 1, `PAV_VIEW_QUAD` = 2, `PAV_VIEW_PHASE` = 3,
`PAV_VIEW_THD` = 4, `PAV_VIEW_SIGVOLT` = 5, `PAV_VIEW_REFVOLT` = 6, `PAV_VIEW_SIGOFFSET` = 7,
`PAV_VIEW_TOTALRATIO` = 8, `PAV_VIEW_FREQ` = 9, `PAV_VIEW_MAIN` = 10 }
- enum `PAV_VoltViews` { `PAV_VOLTVIEW_TOTALSUM` = 0, `PAV_VOLTVIEW_TOTALRMS_AC` = 1, `PAV_VOLTVIEW_TOTALRMS_ACDC` = 2, `PAV_VOLTVIEW_DC` = 3 }
- enum `PAV_Units` {
`PAV_UNITS_V` = 0, `PAV_UNITS_MV` = 1, `PAV_UNITS_RATIO` = 2, `PAV_UNITS_PERCENT` = 3,
`PAV_UNITS_DB` = 4, `PAV_UNITS_360` = 5, `PAV_UNITS_180` = 6, `PAV_UNITS_HZ` = 7,
`PAV_UNITS_KHZ` = 8 }

- enum `PAV_Ranges` {
`PAV_RANGE_50MV = 0, PAV_RANGE_100MV = 1, PAV_RANGE_200MV = 2, PAV_RANGE_500MV = 3,`
`PAV_RANGE_1V = 4, PAV_RANGE_2V = 5, PAV_RANGE_5V = 6, PAV_RANGE_10V = 7,`
`PAV_RANGE_20V = 8, PAV_RANGE_50V = 9, PAV_RANGE_100V = 10, PAV_RANGE_200V = 11,`
`PAV_RANGE_500V = 12 }`
- enum `PAV_OPTIONS_SIGNALINPUT` { `PAV_SIGNALINPUT_FRONT = 0, PAV_SIGNALINPUT_BACK = 1` }
- enum `PAV_OPTIONS_MAINDISPLAY` { `PAV_MAINDISPLAY_INDEPENDENT = 0, PAV_MAINDISPLAY_LINKED = 1` }
- enum `PAV_OPTIONS_RATIO` { `PAV_RATIO_APPLYONLYTOMAIN = 0, PAV_RATIO_APPLYTORELEVANT = 1` }
- enum `PAV_OPTIONS_TIMEDISPLAY` { `PAV_TIMEDISPLAY_AMPM = 0, PAV_TIMEDISPLAY_MILITARY = 1` }
- enum `PAV_OPTIONS_DATEDISPLAY` { `PAV_DATEDISPLAY_TEXT = 0, PAV_DATEDISPLAY_NUMERIC = 1` }
- enum `PAV_OPTIONS_AUTOSAVE` { `PAV_AUTOSAVE_DISABLE = 0, PAV_AUTOSAVE_ENABLE = 1` }
- enum `PAV_OPTIONS_AUTOUNITS` { `PAV_AUTOUNITS_DISABLE = 0, PAV_AUTOUNITS_ENABLE = 1` }
- enum `PAV_OPTIONS_TOUCHSCREEN` { `PAV_TOUCHSCREEN_DISABLE = 0, PAV_TOUCHSCREEN_ENABLE = 1` }
- enum `PAV_REMOTE_SENSE_STATE` { `PAV_RMT_SENSE_DISABLE = 0, PAV_RMT_SENSE_ENABLE = 1` }
- enum `PAV_INT_REF_REMOTE_GEN_DIR` { `PAV_REF_GEN_DIR_BACK = 0, PAV_REF_GEN_DIR_FRONT = 1` }
- enum `PAV_INT_REF_OVER_CURRENT_STATE` { `PAV_NO_OVER_CURRENT = 0, PAV_OVER_CURRENT = 1` }
- enum `PAV_INT_REF_OUTPUT_STATE` { `PAV_INT_REF_OUT_NOT_AVAILABLE = 0, PAV_INT_REF_OUT_AVAILABLE = 1` }
- enum `PAV_LVDT_TYPE` { `PAV_LVDT_TYPE_2WIRE = 2, PAV_LVDT_TYPE_3WIRE = 3, PAV_LVDT_TYPE_4WIRE = 4` }
- enum `PAV_LVDT_SIGNAL` { `PAV_LVDT_SIGNAL_FUND = 0, PAV_LVDT_SIGNAL_TOTAL = 1, PAV_LVDT_SIGNAL_INPH = 2` }
- enum `PAV_LVDT_4WIRE_ALGORITHM` { `PAV_LVDT_4WIRE_VA_VAPLUSVB = 0, PAV_LVDT_4WIRE_VA_VB = 1` }

Functions

- `_PAV2250AFUNC` int `PAV2250A_ConnectViaIEEE` (int nPAVNo, int nIEEEAddr, int nIEEELanguage)
PAV2250A_ConnectViaIEEE sets up and opens the connection to communicate to the PAV2250A via IEEE. The IEEE supports the following language protocols:
- `_PAV2250AFUNC` int `PAV2250A_ConnectViaUSB` (int nPAVNo, int nDeviceNo)
PAV2250A_ConnectViaUSB sets up and opens the connection to communicate to the PAV2250A via USB. Prior to calling this function, make calls to the `PAV2250A_GetPAV2250AUSBDeviceCnt()` routine to determine the number of Cypress USB Devices detected in your system and the `PAV2250A_GetPAV2250ADeviceIDN()` routine to determine the appropriate PAV identifier (nPAVNo) associated with the Cypress USB Devices that are connected to PAV2250A via USB.
- `_PAV2250AFUNC` int `PAV2250A_ConnectViaEthernet` (int nPAVNo, char *szIPAddr, int nPort)
PAV2250A_ConnectViaEthernet sets up and opens the connection to communicate to the 2250A via Ethernet.
- `_PAV2250AFUNC` int `PAV2250A_DisconnectIEEE` (int nPAVNo)
PAV2250A_DisconnectIEEE closes the connection to communicate to the 2250A via IEEE
- `_PAV2250AFUNC` int `PAV2250A_DisconnectUSB` (int nPAVNo)
PAV2250A_DisconnectUSB closes the connection to communicate to the 2250A via USB.
- `_PAV2250AFUNC` int `PAV2250A_DisconnectEthernet` (int nPAVNo)
PAV2250A_DisconnectEthernet closes the connection to communicate to the 2250A via Ethernet.
- `_PAV2250AFUNC` int `PAV2250A_GetPAV2250AUSBDeviceCnt` (int *pnUSBDeviceCnt)

PAV2250A_GetPAV2250AUSBDeviceCnt invokes the Cypress driver and returns the number of Cypress USB Devices detected with your computer system.

- [_PAV2250AFUNC](#) int [PAV2250A_GetPAV2250ADeviceIDN](#) (int nPAVNo, char *pszID)

PAV2250A_GetPAV2250ADeviceIDN sends the IDN command to get Device ID string for the device. The ID returned includes the manufacturer (NORTH ATLANTIC), the 2250A module, serial number, and revision information. NOTE: this function does not require an Open call to have already been performed, but it does assume a USB physical connection. If you are connecting to the instrument using one of the other methods (IEEE or Ethernet) you must first "connect" to the device using that method and then call *PAV2250A_PerformGetID*.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetID](#) (int nPAVNo, char *pszID)

PAV2250A_PerformGetID sends the IDN command to get Device ID string for the device. The ID returned includes the manufacturer (NORTH ATLANTIC), the 2250A module, serial number, and revision information.
- [_PAV2250AFUNC](#) int [PAV2250A_IsStable](#) (int nPAVNo, bool *pbStable)

PAV2250A_IsStable returns whether or not the PAV is considered to be "stable". (i.e. the unit has settled long enough to allow for accurate readings to be taken.)
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetTotalData](#) (int nPAVNo, char *pszTotalData)

PAV2250A_PerformGetTotalData sends the Total Data command to get Total Data for the device. NOTE: This function should only be called if connecting via Ethernet or IEEE. If connecting by USB, you should fetch the individual components that make up "TotalData" one at a time (Total Ratio, Ref Total RMS AC, Sig Total RMS AC, THD, Frequency, Sample Rate Index, Ref Range, Sig Range, Total Sig Offset, Sig Total Sum, Ref Total RMS AC/DC, Sig Total RMS AC/DC, RefDC and SigDC). Current USB driver in use has a restriction on amount of data it can return from one call.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetTotalDataRaw](#) (int nPAVNo, int nReceiveBufferSize, char *pszTotalData)

PAV2250A_PerformGetTotalDataRaw sends the Total Data command to get Raw Total Data for the device. NOTE: This function should only be called if connecting via Ethernet or IEEE. Current USB driver in use has a restriction on amount of data it can return from one call.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGroupExecuteTrigger](#) (int nPAVNo, int nIEEEAddr)

PAV2250A_PerformGroupExecuteTrigger performs a group execute trigger.
- [_PAV2250AFUNC](#) int [PAV2250A_ResetDefaultValues](#) (int nPAVNo)

PAV2250A_ResetDefaultValues sends the command to set the device settings back to the factory default values.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonics](#) (int nPAVNo, int nHarmonic, char *pszHarmonics)

PAV2250A_PerformGetHarmonics sends the Harmonics command to get Harmonics Data for the device. The Harmonics are returned in a comma separated string and includes the Harmonic Phase, Magnitude, In Phase and Quad for the given harmonic number. (valid values 0 (Fundamental) - 15)
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicPhase](#) (int nPAVNo, int nHarmonic, float *pfPhase)

PAV2250A_GetHarmonicPhase is responsible for returning just the Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicMagnitude](#) (int nPAVNo, int nHarmonic, float *pfMag)

PAV2250A_GetHarmonicMagnitude is responsible for returning just the Magnitude (Amplitude) value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicInPhase](#) (int nPAVNo, int nHarmonic, float *pfInPhase)

PAV2250A_GetHarmonicInPhase is responsible for returning just the In Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicQuad](#) (int nPAVNo, int nHarmonic, float *pfQuad)

PAV2250A_GetHarmonicQuad is responsible for returning just the Quadrature value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonicsRatio](#) (int nPAVNo, int nHarmonic, char *pszHarmonicsRatio)

PAV2250A_PerformGetHarmonicsRatio sends the Harmonics command to get Harmonics Data for the device. The Harmonics are returned in a comma separated string and includes the Harmonic Phase, Magnitude Ratio, In Phase Ratio and Quad Ratio for the given harmonic number. (valid values 0 (Fundamental) - 15)
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioPhase](#) (int nPAVNo, int nHarmonic, float *pfPhase)

PAV2250A_GetHarmonicRatioPhase is responsible for returning just the Phase value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioMagnitude](#) (int nPAVNo, int nHarmonic, float *pfMag)

PAV2250A_GetHarmonicRatioMagnitude is responsible for returning just the Magnitude Ratio (Amplitude) value for the provided Harmonic.
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioInPhase](#) (int nPAVNo, int nHarmonic, float *pfInPhase)

- PAV2250A_GetHarmonicRatioInPhase is responsible for returning just the In Phase Ratio value for the provided Harmonic.*
- [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioQuad](#) (int nPAVNo, int nHarmonic, float *pfQuad)
PAV2250A_GetHarmonicRatioQuad is responsible for returning just the Quadrature Ratio value for the provided Harmonic.
 - [_PAV2250AFUNC](#) int [PAV2250A_PerformGetHarmonicDataRaw](#) (int nPAVNo, int nHarmGroup, int nReceiveBufferSize, char *pszHarmonicData)
PAV2250A_PerformGetHarmonicDataRaw sends the Harmonic Data command to get Raw Harmonic Data for the specified Harmonic Group for the device.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetHarmonicRatioState](#) (int nPAVNo, bool *pbHarmRatio)
PAV2250A_GetHarmonicRatioState sends the command to get the Harmonic Ratio State to the PAV2250A device. Harmonic Ratio State controls whether Ratio or Absolute values are shown for Harmonic Display.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetHarmonicRatioState](#) (int nPAVNo, bool bHarmRatio)
PAV2250A_SetHarmonicRatioState sends the command to set the Harmonic Ratio State to the PAV2250A device. Harmonic Ratio State controls whether Ratio or Absolute values are shown for Harmonic Display.
 - [_PAV2250AFUNC](#) int [PAV2250A_ViewPrevHarmonicGroup](#) (int nPAVNo)
PAV2250A_ViewPrevHarmonicGroup sends the command to set the Harmonic View to the previous group of harmonics.
 - [_PAV2250AFUNC](#) int [PAV2250A_ViewNextHarmonicGroup](#) (int nPAVNo)
PAV2250A_ViewNextHarmonicGroup sends the command to set the Harmonic View to the next group of harmonics.
 - [_PAV2250AFUNC](#) int [PAV2250A_ViewHarmonic](#) (int nPAVNo, int nHarmonic)
PAV2250A_ViewHarmonic sends the command to set the Harmonic View to show the desired harmonic. Valid values are 0 - 15.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefAutoRange](#) (int nPAVNo)
PAV2250A_SetRefAutoRange sends the command to force the Reference to be in "Auto" range mode.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefAutoRange](#) (int nPAVNo, bool *pbAutoRange)
PAV2250A_GetRefAutoRange determines whether or not the Reference is in "Auto" range mode.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange](#) (int nPAVNo, int nRangeIndex)
PAV2250A_SetRefRange sends the command to force the Reference into the specified Range based on the Range Index.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeString](#) (int nPAVNo, char *pszRefRange)
PAV2250A_GetRefRangeString is responsible for returning back the Reference range the PAV is currently operating at in string form. Return values will have the word "AUTO" precede the actual range value when the PAV is in "AUTO" range mode and will have the word "OVR" precede the actual range when the actual range is over the configured range. Example: Auto Range with an actual range of 2 Volts will return: "AUTO 2.000". If the range is not "AUTO" and an Over range was not detected, the configured range will be returned such as: "2.000" for the 2 volt range.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeIndexSettings](#) (int nPAVNo, bool *pbAutoRange, bool *pbRangeMismatch, int *pnRangeIndex)
PAV2250A_GetRefRangeIndexSettings is responsible for returning back Reference range information. Return values indicate whether or not the reference is in "Auto" range mode, whether or not there is a range mismatch (i.e. the actual range is different than the desired configured range), and the actual range index.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeConfigIndex](#) (int nPAVNo, int *pnRangeIndex)
PAV2250A_GetRefRangeConfigIndex is responsible for returning the index of the Reference range the PAV was configured with.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetRefRangeActualIndex](#) (int nPAVNo, int *pnRangeIndex)
PAV2250A_GetRefRangeConfigIndex is responsible for returning the index of the Reference range the PAV was is actually operating in.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange50MV](#) (int nPAVNo)
PAV2250A_SetRefRange50MV sends the command to force the PAV Reference to be 50MV.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange100MV](#) (int nPAVNo)
PAV2250A_SetRefRange100MV sends the command to force the PAV Reference to be 100MV.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange200MV](#) (int nPAVNo)
PAV2250A_SetRefRange200MV sends the command to force the PAV Reference to be 200MV.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange500MV](#) (int nPAVNo)

- PAV2250A_SetRefRange500MV sends the command to force the PAV Reference to be 500MV.*

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange1V](#) (int nPAVNo)

PAV2250A_SetRefRange1V sends the command to force the PAV Reference to be 1V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange2V](#) (int nPAVNo)

PAV2250A_SetRefRange2V sends the command to force the PAV Reference to be 2V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange5V](#) (int nPAVNo)

PAV2250A_SetRefRange5V sends the command to force the PAV Reference to be 5V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange10V](#) (int nPAVNo)

PAV2250A_SetRefRange10V sends the command to force the PAV Reference to be 10V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange20V](#) (int nPAVNo)

PAV2250A_SetRefRange20V sends the command to force the PAV Reference to be 20V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange50V](#) (int nPAVNo)

PAV2250A_SetRefRange50V sends the command to force the PAV Reference to be 50V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange100V](#) (int nPAVNo)

PAV2250A_SetRefRange100V sends the command to force the PAV Reference to be 100V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange200V](#) (int nPAVNo)

PAV2250A_SetRefRange200V sends the command to force the PAV Reference to be 200V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetRefRange500V](#) (int nPAVNo)

PAV2250A_SetRefRange500V sends the command to force the PAV Reference to be 500V.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigAutoRange](#) (int nPAVNo)

PAV2250A_SetSigAutoRange sends the command to force the Signal to be in "Auto" range mode.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigAutoRange](#) (int nPAVNo, bool *pbAutoRange)

PAV2250A_GetSigAutoRange determines whether or not the Signal is in "Auto" range mode.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange](#) (int nPAVNo, int nRangeIndex)

PAV2250A_SetSigRange sends the command to force the Signal into the specified Range based on the Range Index.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeString](#) (int nPAVNo, char *pszSigRange)

PAV2250A_GetSigRangeString is responsible for returning back the Signal range the PAV is currently operating at in string form. Return values will have the word "AUTO" precede the actual range value when the PAV is in "AUTO" range mode and will have the word "OVR" precede the actual range when the actual range is over the configured range. Example: Auto Range with an actual range of 2 Volots will return: "AUTO 2.000". If the range is not "AUTO" and an Over range was not detected, the configured range will be returned such as: "2.000" for the 2 volt range.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeIndexSettings](#) (int nPAVNo, bool *pbAutoRange, bool *pbRangeMismatch, int *pnRangeIndex)

PAV2250A_GetSigRangeIndexSettings is responsible for returning back Signal range information. Return values indicate whether or not the signal is in "Auto" range mode, whether or not there is a range mismatch (i.e. the actual range is different than the desired configured range), and the actual range index.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeConfigIndex](#) (int nPAVNo, int *pnRangeIndex)

PAV2250A_GetSigRangeConfigIndex is responsible for returning the index of the Signal range the PAV was configured with.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetSigRangeActualIndex](#) (int nPAVNo, int *pnRangeIndex)

PAV2250A_GetSigRangeActualIndex is responsible for returning the index of the Signal range the PAV was actually operating in.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange50MV](#) (int nPAVNo)

PAV2250A_SetSigRange50MV sends the command to force the PAV Signal to be 50MV.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange100MV](#) (int nPAVNo)

PAV2250A_SetSigRange100MV sends the command to force the PAV Signal to be 100MV.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange200MV](#) (int nPAVNo)

PAV2250A_SetSigRange200MV sends the command to force the PAV Signal to be 200MV.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange500MV](#) (int nPAVNo)

PAV2250A_SetSigRange500MV sends the command to force the PAV Signal to be 500MV.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange1V](#) (int nPAVNo)

PAV2250A_SetSigRange1V sends the command to force the PAV Signal to be 1V.

- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange2V](#) (int nPAVNo)
PAV2250A_SetSigRange2V sends the command to force the PAV Signal to be 2V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange5V](#) (int nPAVNo)
PAV2250A_SetSigRange5V sends the command to force the PAV Signal to be 5V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange10V](#) (int nPAVNo)
PAV2250A_SetSigRange10V sends the command to force the PAV Signal to be 10V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange20V](#) (int nPAVNo)
PAV2250A_SetSigRange20V sends the command to force the PAV Signal to be 20V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange50V](#) (int nPAVNo)
PAV2250A_SetSigRange50V sends the command to force the PAV Signal to be 50V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange100V](#) (int nPAVNo)
PAV2250A_SetSigRange100V sends the command to force the PAV Signal to be 100V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange200V](#) (int nPAVNo)
PAV2250A_SetSigRange200V sends the command to force the PAV Signal to be 200V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigRange500V](#) (int nPAVNo)
PAV2250A_SetSigRange500V sends the command to force the PAV Signal to be 500V.
- [_PAV2250AFUNC](#) int [PAV2250A_IEEEReset](#) (int nPAVNo, char *pszResults)
PAV2250A_IEEEReset sends the command to reset the 2250A device and set the device setting back to the factory default settings. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_IEEEGetErrors](#) (int nPAVNo, char *pszErrors)
PAV2250A_IEEEGetErrors sends the ERR command to get error from the error queue for the device. No error is returned when there are no errors on the queue.
- [_PAV2250AFUNC](#) int [PAV2250A_IEEECLS](#) (int nPAVNo)
PAV2250A_IEEECLS is responsible for clearing the IEEE - Clears Event Status Registers and Error Message Queue.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIEEELang](#) (int nPAVNo, int *pnIEEELang)
PAV2250A_GetIEEELang sends the command to get the IEEE language protocol set in the 2250A.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIEEELangText](#) (int nPAVNo, char *szIEEELang)
PAV2250A_GetIEEELangText sends the command to get the IEEE language protocol set in the 2250A.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang](#) (int nPAVNo, int nIEEELang)
PAV2250A_SetIEEELang sends the command to set the IEEE protocol language to accept when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang2250ANative](#) (int nPAVNo)
PAV2250A_SetIEEELang2250ANative sends the command to set the IEEE protocol to the 2250A Native language when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIEEELang2250Legacy](#) (int nPAVNo)
PAV2250A_SetIEEELang2250Legacy sends the command to set the IEEE protocol to the 2250A Native language when communicating via IEEE. Note the 2250A device will not accept the command if its remote communication configuration does not match the communication connection mode.
- [_PAV2250AFUNC](#) int [PAV2250A_GetCommState](#) (int nPAVNo, char *szCommState)
PAV2250A_GetCommState sends the command to get the communication mode set in the 2250A.
- [_PAV2250AFUNC](#) int [PAV2250A_GoToLocal](#) (int nPAVNo)
PAV2250A_GoToLocal sends the command to set the communication mode to Local mode. In Local mode, remote set commands will not be accepted.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteUSB](#) (int nPAVNo)
PAV2250A_SetRemoteUSB sends the command to set the communication mode to Remote USB mode. In Remote USB mode, remote set commands will be accepted if the command is received from the USB interface.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteEthernet](#) (int nPAVNo)
PAV2250A_SetRemoteEthernet sends the command to set the communication mode to Remote Ethernet mode. In Remote Ethernet mode, remote set commands will be accepted if the command is received from the Ethernet interface.

- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteIEEE](#) (int nPAVNo)
PAV2250A_SetRemoteIEEE sends the command to set the communication mode to Remote IEEE mode. In Remote IEEE mode, remote set commands will be accepted if the command is received from the IEEE interface.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRemoteJ1](#) (int nPAVNo)
PAV2250A_SetRemoteJ1 sends the command to set the communication mode to Remote J1 mode. In Remote J1 mode, remote set commands will be accepted if the command is received from the J1 interface.
- [_PAV2250AFUNC](#) int [PAV2250A_Calibrate](#) (int nPAVNo)
PAV2250A_Calibrate sends the command to force the PAV to perform calibration.
- [_PAV2250AFUNC](#) int [PAV2250A_GetCalState](#) (int nPAVNo, char *pszCalState)
PAV2250A_GetCalState sends a command to the PAV 2250A to retrieve its current calibration state.
- [_PAV2250AFUNC](#) int [PAV2250A_MaxRetry](#) (int nMaxRetry)
PAV2250A_MaxRetry sets the maximum retries to send a command or read a response that will be made when communicating via IEEE. The default value is 0.
- [_PAV2250AFUNC](#) int [PAV2250A_LastCmdSent](#) (int nPAVNo, char szLastCommand[])
PAV2250A_LastCmdSent returns the last command sent via IEEE, USB or Ethernet to the 2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_WriteCommand](#) (int nPAVNo, char szCommand[])
PAV2250A_WriteCommand sends the command to the 2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_QueryCommand](#) (int nPAVNo, char szCommand[], char *pszResponse)
PAV2250A_QueryCommand sends the command to the 2250A device and waits for the 2250A to respond.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeWndAuto](#) (int nPAVNo, bool bAuto)
PAV2250A_SetTimeWndAuto sends a command to the PAV 2250A to force the time window to be in "Auto" mode. The system will decide the data refresh rate.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndAuto](#) (int nPAVNo, bool *pbAuto)
PAV2250A_GetTimeWndAuto sends a command to the PAV 2250A to retrieve whether or not the Time Window is in "Auto" mode.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeWndOverride](#) (int nPAVNo, float fOverrideInterval)
PAV2250A_SetTimeWndOverride sends a command to the PAV 2250A to force the Time Window to be a specific value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndOverride](#) (int nPAVNo, float *pfOverrideInterval)
PAV2250A_GetTimeWndOverride sends a command to the PAV 2250A to retrieve the current setting for the Time Window override interval.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeWndActual](#) (int nPAVNo, float *pfActualInterval)
PAV2250A_GetTimeWndActual sends a command to the PAV 2250A to retrieve the actual TimeWindow interval. NOTE: If the PAV's TimeWindow is configured to "Auto", the system decides the appropriate TimeWindow.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSignalInputOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetSignalInputOption sends a command to the PAV 2250A requesting the current Signal Input Value to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSignalInputOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetSignalInputOptionText sends the command requesting the current Signal Input Text value to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputOption](#) (int nPAVNo, int nOptionIndex)
PAV2250A_SetSignalInputOption sends the command to set the current Signal Input Value to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputFront](#) (int nPAVNo)
PAV2250A_SetSignalInputFront sends the command to set the current Signal Input Value to the Front Panel.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSignalInputBack](#) (int nPAVNo)
PAV2250A_SetSignalInputBack sends the command to set the current Signal Input Value to the Back Panel.
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
PAV2250A_GetMainDisplayOption sends the command requesting the current Main Display Index to the PAV2250A device. Main Display can either be 0 ("Independent View") or 1 ("Linked View")
- [_PAV2250AFUNC](#) int [PAV2250A_GetMainDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
PAV2250A_GetMainDisplayOptionText sends the command requesting the current Main Display Option Text to the PAV2250A device. Main Display can either be "Independent View" or "Linked View".
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayOption](#) (int nPAVNo, int nOptionIndex)

- PAV2250A_SetMainDisplayOption* sends the command to set the Main Display Index Value to the PAV2250A device. Main Display can either be 0 ("Independent View") or 1 ("Linked View").
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayIndependent](#) (int nPAVNo)
 - PAV2250A_SetMainDisplayIndependent* sends the command to set the Main Display Value to Independent View.
- [_PAV2250AFUNC](#) int [PAV2250A_SetMainDisplayLinked](#) (int nPAVNo)
 - PAV2250A_SetMainDisplayLinked* sends the command to set the Main Display Value to Linked View.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
 - PAV2250A_GetTimeDisplayOption* sends the command requesting the current Time Display Option Index to the PAV2250A device. Time Display can either be 0 ("AM/PM") or 1 ("Military")
- [_PAV2250AFUNC](#) int [PAV2250A_GetTimeDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
 - PAV2250A_GetTimeDisplayOptionText* sends the command requesting the current Time Display Option Text to the PAV2250A device. Time Display can either be "AM/PM" or "Military".
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayOption](#) (int nPAVNo, int nOptionIndex)
 - PAV2250A_SetTimeDisplayOption* sends the command to set the Time Display Option Value to the PAV2250A device. Time Display can either be 0 ("AM/PM") or 1 ("Military")
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayAMPM](#) (int nPAVNo)
 - PAV2250A_SetTimeDisplayAMPM* sends the command to set the Time Display Option Value to AM/PM.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTimeDisplayMilitary](#) (int nPAVNo)
 - PAV2250A_SetTimeDisplayMilitary* sends the command to set the Time Display Option Value to Military.
- [_PAV2250AFUNC](#) int [PAV2250A_GetDateDisplayOption](#) (int nPAVNo, int *pnOptionIndex)
 - PAV2250A_GetDateDisplayOption*
- [_PAV2250AFUNC](#) int [PAV2250A_GetDateDisplayOptionText](#) (int nPAVNo, char *pszOptionText)
 - PAV2250A_GetDateDisplayOptionText* sends the command requesting the current Date Display Option Index to the PAV2250A device. Date Display can either be "Text Date"(MON/DD/YYYY) or "Numeric Date"(01/01/11).
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayOption](#) (int nPAVNo, int nOptionIndex)
 - PAV2250A_SetDateDisplayOption* sends the command to set the Date Display Option Value to the PAV2250A device. Date Display can either be 0 ("Text Date") or 1 ("Numeric Date")
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayText](#) (int nPAVNo)
 - PAV2250A_SetDateDisplayText* sends the command to set the Date Display Option Value to Text format.
- [_PAV2250AFUNC](#) int [PAV2250A_SetDateDisplayNumeric](#) (int nPAVNo)
 - PAV2250A_SetDateDisplayNumeric* sends the command to set the Date Display Option Value to Numeric format.
- [_PAV2250AFUNC](#) int [PAV2250A_GetAutoSaveOption](#) (int nPAVNo, int *pnOptionIndex)
 - PAV2250A_GetAutoSaveOption* sends the command requesting the current Auto Save Option Index to the PAV2250-A device. Auto Save Options can either be 0 ("Disabled") or 1 ("Enabled").
- [_PAV2250AFUNC](#) int [PAV2250A_GetAutoSaveOptionText](#) (int nPAVNo, char *pszOptionText)
 - PAV2250A_GetAutoSaveOptionText*
- [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveOption](#) (int nPAVNo, int nOptionIndex)
 - PAV2250A_SetAutoSaveOption* sends the command to set the Auto Save Option Index to the PAV2250A device. Auto Save can either be 0 ("Disabled") or 1 ("Enabled").
- [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveEnable](#) (int nPAVNo)
 - PAV2250A_SetAutoSaveEnable* sends the command to set the Auto Save Option Index to Enabled.
- [_PAV2250AFUNC](#) int [PAV2250A_SetAutoSaveDisable](#) (int nPAVNo)
 - PAV2250A_SetAutoSaveDisable* sends the command to set the Auto Save Option Index to Disabled.
- [_PAV2250AFUNC](#) int [PAV2250A_GetAutoUnitsOption](#) (int nPAVNo, int *pnOptionIndex)
 - PAV2250A_GetAutoUnitsOption* sends the command requesting the current Auto Units Option Index to the PAV2250-A device. Auto Units Options can either be 0 ("Disabled") or 1 ("Enabled").
- [_PAV2250AFUNC](#) int [PAV2250A_GetAutoUnitsOptionText](#) (int nPAVNo, char *pszOptionText)
 - PAV2250A_GetAutoUnitsOptionText* sends the command requesting the current Auto Units Option Index to the PAV2250A device. Auto Units can either be "Enabled" or "Disabled"
- [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsOption](#) (int nPAVNo, int nOptionIndex)
 - PAV2250A_SetAutoUnitsOption* sends the command to set the Auto Units Option Index to the PAV2250A device. Auto Units can either be 0 ("Disabled") or 1 ("Enabled").
- [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsEnable](#) (int nPAVNo)

- PAV2250A_SetAutoUnitsEnable* sends the command to set the Auto Units Option Index to Enabled.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetAutoUnitsDisable](#) (int nPAVNo)

PAV2250A_SetAutoUnitsDisable sends the command to set the Auto Units Option Index to Disabled.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTouchscreenOption](#) (int nPAVNo, int *pnOptionIndex)

PAV2250A_GetTouchscreenOption sends the command requesting the current Touchscreen Option Index to the PAV2250A device. Touchscreen Options can either be 0 ("Disabled") or 1 ("Enabled")
- [_PAV2250AFUNC](#) int [PAV2250A_GetTouchscreenOptionText](#) (int nPAVNo, char *pszOptionText)

PAV2250A_GetTouchscreenOptionText sends the command requesting the current Touchscreen Option Index to the PAV2250A device. Touchscreen can either be "Enabled" or "Disabled"
- [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenOption](#) (int nPAVNo, int nOptionIndex)

PAV2250A_SetTouchscreenOption sends the command to set the Touchscreen Option Index to the PAV2250A device. Touchscreen can either be 0 ("Disabled") or 1 ("Enabled").
- [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenEnable](#) (int nPAVNo)

PAV2250A_SetTouchscreenEnable sends the command to set the Touchscreen Option Index to Enabled.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTouchscreenDisable](#) (int nPAVNo)

PAV2250A_SetTouchscreenDisable sends the command to set the Touchscreen Option Index to Disabled.
- [_PAV2250AFUNC](#) int [PAV2250A_SetNullMeterRangePercent](#) (int nPAVNo, float fRangePercent)

PAV2250A_SetNullMeterRangePercent sends the command to set the Null Meter Range Percent option.
- [_PAV2250AFUNC](#) int [PAV2250A_GetNullMeterRangePercent](#) (int nPAVNo, float *pfRangePercent)

PAV2250A_GetNullMeterRangePercent sends the command to get the Null Meter Range Percent option.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTabView](#) (int nPAVNo, int *pnTabIndex)

PAV2250A_GetTabView sends the command requesting the current Tab View Index to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTabViewText](#) (int nPAVNo, char *pszTabViewText)

PAV2250A_GetTabViewText sends the command requesting the current Tab View text to the PAV2250A device. Text is reflective of the currently selected tab label. (Main, Harmonics, Quad View)
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabView](#) (int nPAVNo, int nTabIndex)

PAV2250A_SetTabView sends the command to set the Tab View Index to the PAV2250A device. Tab View Index should be reflective of which tab the PAV should display as the active tab. Index is zero-based.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewMain](#) (int nPAVNo)

PAV2250A_SetTabViewMain sends the command to set the Tab View to the Main tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewReference](#) (int nPAVNo)

PAV2250A_SetTabViewReference sends the command to set the Tab View to the Reference tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewHarmonics](#) (int nPAVNo)

PAV2250A_SetTabViewHarmonics sends the command to set the Tab View to the Harmonics tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewCustom](#) (int nPAVNo)

PAV2250A_SetTabViewCustom sends the command to set the Tab View to the Custom View tab.
- [_PAV2250AFUNC](#) int [PAV2250A_SetTabViewLVDT](#) (int nPAVNo)

PAV2250A_SetTabViewLVDT sends the command to set the Tab View to the LVDT View tab.
- [_PAV2250AFUNC](#) int [PAV2250A_GetReadMode](#) (int nPAVNo, int *pnReadModeIndex)

PAV2250A_GetReadMode sends the command requesting the current Read Mode Index to the PAV2250A device. Read Mode index indicates whether the PAV is showing SIG/REF, REF, SIG or REF/SIG. Index is zero-based.
- [_PAV2250AFUNC](#) int [PAV2250A_GetReadModeText](#) (int nPAVNo, char *pszReadModeText)

PAV2250A_GetReadModeText sends the command requesting the current Read Mode text to the PAV2250A device. Text is reflective of the currently selected Reading Mode (SIG/REF, REF, SIG, or REF/SIG).
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadMode](#) (int nPAVNo, int nReadModeIndex)

PAV2250A_SetReadMode sends the command to set the Read Mode Index to the PAV2250A device. Read Mode index indicates whether the PAV is showing SIG/REF, REF, SIG or REF/SIG. Index is zero-based.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeSigRef](#) (int nPAVNo)

PAV2250A_SetReadModeSigRef sends the command to set the Read Mode to Sig/Ref.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeRefRef](#) (int nPAVNo)

PAV2250A_SetReadModeRefRef sends the command to set the Read Mode to Ref.
- [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeSigSig](#) (int nPAVNo)

- PAV2250A_SetReadModeSigSig sends the command to set the Read Mode to Sig.*

 - [_PAV2250AFUNC](#) int [PAV2250A_SetReadModeRefSig](#) (int nPAVNo)
PAV2250A_SetReadModeRefSig sends the command to set the Read Mode to Ref/Sig.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetMainView](#) (int nPAVNo, int *pnMainViewIndex)
PAV2250A_GetMainView sends the command requesting the current Main View Index to the PAV2250A device. Main View index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetMainViewText](#) (int nPAVNo, char *pszMainViewText)
PAV2250A_GetMainViewText sends the command requesting the current Main View text to the PAV2250A device. Text is reflective of the currently selected Main View: (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetMainView](#) (int nPAVNo, int nMainViewIndex)
PAV2250A_SetMainView sends the command to set the Main View Index to the PAV2250A device. Main View index indicates the view currently shown in the main display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewFundMag](#) (int nPAVNo)
PAV2250A_SetMainViewFundMag sends the command to set the Main View to Fundamental Magnitude.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewInPhase](#) (int nPAVNo)
PAV2250A_SetMainViewInPhase sends the command to set the Main View to In Phase.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewQuad](#) (int nPAVNo)
PAV2250A_SetMainViewQuad sends the command to set the Main View to Quadrature.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewPhase](#) (int nPAVNo)
PAV2250A_SetMainViewPhase sends the command to set the Main View to Phase.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewTHD](#) (int nPAVNo)
PAV2250A_SetMainViewTHD sends the command to set the Main View to THD.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewSigVolt](#) (int nPAVNo)
PAV2250A_SetMainViewSigVolt sends the command to set the Main View to Signal Voltage.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetMainViewRefVolt](#) (int nPAVNo)
PAV2250A_SetMainViewRefVolt sends the command to set the Main View to Reference Voltage.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetHoldDataState](#) (int nPAVNo, bool *pbHoldData)
PAV2250A_GetHoldDataState sends the command to get the Hold Data State to the PAV2250A device. Hold Data when true indicates no screen refreshes are taking place.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetHoldDataState](#) (int nPAVNo, bool bHoldData)
PAV2250A_SetHoldDataState sends the command to set the Hold Data State to the PAV2250A device. Hold Data when true indicates no screen refreshes are taking place.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetScreenBrightness](#) (int nPAVNo, int *pnBrightness)
PAV2250A_GetScreenBrightness sends the command requesting the current Screen Brightness to the PAV2250A device. Screen Brightness indicates how bright the screen is illuminated. Valid values are between 15-100.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetScreenBrightness](#) (int nPAVNo, int nBrightness)
PAV2250A_SetScreenBrightness sends the command setting the current Screen Brightness on the PAV2250A device. Screen Brightness indicates how bright the screen is illuminated. Valid values are between 15-100.
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView1](#) (int nPAVNo, int *pnCustView1Index)
PAV2250A_GetCustView1 sends the command requesting the current Quad View 1 Index to the PAV2250A device. Quad View 1 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView1Text](#) (int nPAVNo, char *pszCustView1Text)
PAV2250A_GetCustView1Text sends the command requesting the current Quad View 1 Text to the PAV2250A device. Quad View 1 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1](#) (int nPAVNo, int nCustView1Index)
PAV2250A_SetCustView1 sends the command to set the Quad View 1 Index to the PAV2250A device. Quad View 1 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1FundMag](#) (int nPAVNo)

- PAV2250A_SetCustView1FundMag* sends the command to set the Quad View 1 slot to Fundamental Magnitude.

 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1InPhase](#) (int nPAVNo)
PAV2250A_SetCustView1InPhase sends the command to set the Quad View 1 slot to In Phase.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1Quad](#) (int nPAVNo)
PAV2250A_SetCustView1Quad sends the command to set the Quad View 1 slot to Quadrature.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1Phase](#) (int nPAVNo)
PAV2250A_SetCustView1Phase sends the command to set the Quad View 1 slot to Phase Angle.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1THD](#) (int nPAVNo)
PAV2250A_SetCustView1THD sends the command to set the Quad View 1 slot to THD.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1SigVolt](#) (int nPAVNo)
PAV2250A_SetCustView1SigVolt sends the command to set the Quad View 1 slot to Signal Voltage.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView1RefVolt](#) (int nPAVNo)
PAV2250A_SetCustView1RefVolt sends the command to set the Quad View 1 slot to Reference Voltage.
- [_PAV2250AFUNC](#) int [PAV2250A_GetCustView2](#) (int nPAVNo, int *pnCustView2Index)
PAV2250A_GetCustView2 sends the command requesting the current Quad View 2 Index to the PAV2250A device. Quad View 2 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView2Text](#) (int nPAVNo, char *pszCustView2Text)
PAV2250A_GetCustView2Text sends the command requesting the current Quad View 2 Text to the PAV2250A device. Quad View 2 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2](#) (int nPAVNo, int nCustView2Index)
PAV2250A_SetCustView2 sends the command to set the Quad View 2 Index to the PAV2250A device. Quad View 2 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2FundMag](#) (int nPAVNo)
PAV2250A_SetCustView2FundMag sends the command to set the Quad View 2 slot to Fundamental Magnitude.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2InPhase](#) (int nPAVNo)
PAV2250A_SetCustView2InPhase sends the command to set the Quad View 2 slot to In Phase.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2Quad](#) (int nPAVNo)
PAV2250A_SetCustView2Quad sends the command to set the Quad View 2 slot to Quadrature.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2Phase](#) (int nPAVNo)
PAV2250A_SetCustView2Phase sends the command to set the Quad View 2 slot to Phase Angle.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2THD](#) (int nPAVNo)
PAV2250A_SetCustView2THD sends the command to set the Quad View 2 slot to THD.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2SigVolt](#) (int nPAVNo)
PAV2250A_SetCustView2SigVolt sends the command to set the Quad View 2 slot to Signal Voltage.
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView2RefVolt](#) (int nPAVNo)
PAV2250A_SetCustView2RefVolt sends the command to set the Quad View 2 slot to Reference Voltage.
- [_PAV2250AFUNC](#) int [PAV2250A_GetCustView3](#) (int nPAVNo, int *pnCustView3Index)
PAV2250A_GetCustView3 sends the command requesting the current Quad View 3 Index to the PAV2250A device. Quad View 3 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_GetCustView3Text](#) (int nPAVNo, char *pszCustView3Text)
PAV2250A_GetCustView3Text sends the command requesting the current Quad View 3 Text to the PAV2250A device. Quad View 3 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3](#) (int nPAVNo, int nCustView3Index)
PAV2250A_SetCustView3 sends the command to set the Quad View 3 Index to the PAV2250A device. Quad View 3 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
 - [_PAV2250AFUNC](#) int [PAV2250A_SetCustView3FundMag](#) (int nPAVNo)
PAV2250A_SetCustView3FundMag sends the command to set the Quad View 3 slot to Fundamental Magnitude.

- `_PAV2250AFUNC` int `PAV2250A_SetCustView3InPhase` (int nPAVNo)
PAV2250A_SetCustView3InPhase sends the command to set the Quad View 3 slot to In Phase.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView3Quad` (int nPAVNo)
PAV2250A_SetCustView3Quad sends the command to set the Quad View 3 slot to Quadrature.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView3Phase` (int nPAVNo)
PAV2250A_SetCustView3Phase sends the command to set the Quad View 3 slot to Phase Angle.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView3THD` (int nPAVNo)
PAV2250A_SetCustView3THD sends the command to set the Quad View 3 slot to THD.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView3SigVolt` (int nPAVNo)
PAV2250A_SetCustView3SigVolt sends the command to set the Quad View 3 slot to Signal Voltage.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView3RefVolt` (int nPAVNo)
PAV2250A_SetCustView3RefVolt sends the command to set the Quad View 3 slot to Reference Voltage.
- `_PAV2250AFUNC` int `PAV2250A_GetCustView4` (int nPAVNo, int *pnCustView4Index)
PAV2250A_GetCustView4 sends the command requesting the current Quad View 4 Index to the PAV2250A device. Quad View 4 Index indicates the view index of the view currently showing on the PAV. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
- `_PAV2250AFUNC` int `PAV2250A_GetCustView4Text` (int nPAVNo, char *pszCustView4Text)
PAV2250A_GetCustView4Text sends the command requesting the current Quad View 4 Text to the PAV2250A device. Quad View 4 Text indicates the view name of the view currently showing on the PAV. (Fund Mag, In Phase, Quad, Phase, THD, Sig Volt, Ref Volt)
- `_PAV2250AFUNC` int `PAV2250A_SetCustView4` (int nPAVNo, int nCustView4Index)
PAV2250A_SetCustView4 sends the command to set the Quad View 4 Index to the PAV2250A device. Quad View 4 index indicates the view currently shown in the Quad display. Index is zero-based. (0=Fund Mag, 1=In Phase, 2=Quad, 3=Phase, 4=THD, 5=Sig Volt, 6=Ref Volt)
- `_PAV2250AFUNC` int `PAV2250A_SetCustView4FundMag` (int nPAVNo)
PAV2250A_SetCustView4FundMag sends the command to set the Quad View 4 slot to Fundamental Magnitude.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView4InPhase` (int nPAVNo)
PAV2250A_SetCustView4InPhase sends the command to set the Quad View 4 slot to In Phase.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView4Quad` (int nPAVNo)
PAV2250A_SetCustView4Quad sends the command to set the Quad View 4 slot to Quadrature.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView4Phase` (int nPAVNo)
PAV2250A_SetCustView4Phase sends the command to set the Quad View 4 slot to Phase Angle.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView4THD` (int nPAVNo)
PAV2250A_SetCustView4THD sends the command to set the Quad View 4 slot to THD.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView4SigVolt` (int nPAVNo)
PAV2250A_SetCustView4SigVolt sends the command to set the Quad View 4 slot to Signal Voltage.
- `_PAV2250AFUNC` int `PAV2250A_SetCustView4RefVolt` (int nPAVNo)
PAV2250A_SetCustView4RefVolt sends the command to set the Quad View 4 slot to Reference Voltage.
- `_PAV2250AFUNC` int `PAV2250A_GetViewIndexConfig` (int nPAVNo, int nViewIndex, char *pszViewConfig)
PAV2250A_GetViewIndexConfig sends the command to get the view configuration for the specified view index. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- `_PAV2250AFUNC` int `PAV2250A_GetViewFundMagConfig` (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewFundMagConfig sends the command to get the view configuration for the Fundamental Magnitude. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- `_PAV2250AFUNC` int `PAV2250A_GetViewInPhaseConfig` (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewInPhaseConfig sends the command to get the view configuration for the In Phase component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- `_PAV2250AFUNC` int `PAV2250A_GetViewQuadConfig` (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewQuadConfig sends the command to get the view configuration for the Quad component. The view configuration consists of the view name, unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.

- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewPhaseConfig sends the command to get the view configuration for the Phase component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewTHDConfig sends the command to get the view configuration for the THD component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewSigVoltConfig sends the command to get the view configuration for the Sig Volt component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewRefVoltConfig sends the command to get the view configuration for the Ref Volt component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewSigOffsetConfig sends the command to get the view configuration for the Sig Offset component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewTotalRatioConfig sends the command to get the view configuration for the Total Ratio component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewFrequencyConfig sends the command to get the view configuration for the Frequency component. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainConfig](#) (int nPAVNo, char *pszViewConfig)
PAV2250A_GetViewMainConfig sends the command to get the view configuration for the Main view. The view configuration consists of the view name,unit index, resolution, offset, scale. The returned string will be a comma delimited string containing the fields depicted above.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewFundMagMaxFieldWidth sends the command to set the max field width for Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewFundMagMaxFieldWidth sends the command to get the max field width for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewInPhaseMaxFieldWidth sends the command to set the max field width for In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewInPhaseMaxFieldWidth sends the command to get the max field width for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewQuadMaxFieldWidth sends the command to set the max field width for Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewQuadMaxFieldWidth sends the command to get the max field width for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhaseMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewPhaseMaxFieldWidth sends the command to set the max field width for Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewPhaseMaxFieldWidth sends the command to get the max field width for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewTHDMaxFieldWidth sends the command to set the max field width for THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewTHDMaxFieldWidth sends the command to get the max field width for the THD view.

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewSigVoltMaxFieldWidth sends the command to set the max field width for Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewSigVoltMaxFieldWidth sends the command to get the max field width for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewRefVoltMaxFieldWidth sends the command to set the max field width for Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewRefVoltMaxFieldWidth sends the command to get the max field width for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewSigOffsetMaxFieldWidth sends the command to set the max field width for Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewSigOffsetMaxFieldWidth sends the command to get the max field width for the SigOffset view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewTotalRatioMaxFieldWidth sends the command to set the max field width for Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewTotalRatioMaxFieldWidth sends the command to get the max field width for the TotalRatio view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewFrequencyMaxFieldWidth sends the command to set the max field width for Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewFrequencyMaxFieldWidth sends the command to get the max field width for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainMaxFieldWidth](#) (int nPAVNo, int nMaxFieldWidth)
PAV2250A_SetViewMainMaxFieldWidth sends the command to set the max field width for Main view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainMaxFieldWidth](#) (int nPAVNo, int *pnMaxFieldWidth)
PAV2250A_GetViewMainMaxFieldWidth sends the command to get the max field width for the Main view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagV](#) (int nPAVNo)
PAV2250A_SetViewFundMagV sends the command to set Fundamental Magnitude units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagMV](#) (int nPAVNo)
PAV2250A_SetViewFundMagMV sends the command to set Fundamental Magnitude units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagPercent](#) (int nPAVNo)
PAV2250A_SetViewFundMagPercent sends the command to set Fundamental Magnitude units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagDB](#) (int nPAVNo)
PAV2250A_SetViewFundMagDB sends the command to set Fundamental Magnitude units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagRatio](#) (int nPAVNo)
PAV2250A_SetViewFundMagRatio sends the command to set Fundamental Magnitude units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewFundMagUnits sends the command to get the view units index for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewFundMagUnitsText sends the command to get the view units for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseV](#) (int nPAVNo)
PAV2250A_SetViewInPhaseV sends the command to set InPhase units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseMV](#) (int nPAVNo)
PAV2250A_SetViewInPhaseMV sends the command to set In Phase units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhasePercent](#) (int nPAVNo)
PAV2250A_SetViewInPhasePercent sends the command to set In Phase units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseDB](#) (int nPAVNo)
PAV2250A_SetViewInPhaseDB sends the command to set In Phase units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseRatio](#) (int nPAVNo)
PAV2250A_SetViewInPhaseRatio sends the command to set Fundamental In Phase units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseUnits](#) (int nPAVNo, int *pnUnits)

- PAV2250A_GetViewInPhaseUnits* sends the command to get the view units index for the In Phase view.

 - [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewInPhaseUnitsText sends the command to get the view units for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadV](#) (int nPAVNo)
PAV2250A_SetViewQuadV sends the command to set Quad units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadMV](#) (int nPAVNo)
PAV2250A_SetViewQuadMV sends the command to set Quad units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadPercent](#) (int nPAVNo)
PAV2250A_SetViewQuadPercent sends the command to set Quad units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadDB](#) (int nPAVNo)
PAV2250A_SetViewQuadDB sends the command to set Quad units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadRatio](#) (int nPAVNo)
PAV2250A_SetViewQuadRatio sends the command to set Quad units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewQuadUnits sends the command to get the view units index for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewQuadUnitsText sends the command to get the view units for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhase360](#) (int nPAVNo)
PAV2250A_SetViewPhase360 sends the command to set Phase units to +-360.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhase180](#) (int nPAVNo)
PAV2250A_SetViewPhase180 sends the command to set Phase units to +-180.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewPhaseUnits sends the command to get the view units index for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewPhaseUnitsText sends the command to get the view units for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDPercent](#) (int nPAVNo)
PAV2250A_SetViewTHDPercent sends the command to set THD units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTHDDB](#) (int nPAVNo)
PAV2250A_SetViewTHDDB sends the command to set THD units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewTHDUnits sends the command to get the view units index for the THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTHDUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewTHDUnitsText sends the command to get the view units for the THD view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltV](#) (int nPAVNo)
PAV2250A_SetViewSigVoltV sends the command to set Signal Voltage units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltMV](#) (int nPAVNo)
PAV2250A_SetViewSigVoltMV sends the command to set Signal Voltage units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewSigVoltUnits sends the command to get the view units index for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewSigVoltUnitsText sends the command to get the view units for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltV](#) (int nPAVNo)
PAV2250A_SetViewRefVoltV sends the command to set Reference Voltage units to V.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltMV](#) (int nPAVNo)
PAV2250A_SetViewRefVoltMV sends the command to set Reference Voltage units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewRefVoltUnits sends the command to get the view units index for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewRefVoltUnitsText sends the command to get the view units for the RefVolt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetV](#) (int nPAVNo)
PAV2250A_SetViewSigOffsetV sends the command to set Signal Offset units to V.

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigOffsetMV](#) (int nPAVNo)
PAV2250A_SetViewSigOffsetMV sends the command to set Signal Offset units to mV.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewSigOffsetUnits sends the command to get the view units index for the Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigOffsetUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewSigOffsetUnitsText sends the command to get the view units for the Sig Offset view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioPercent](#) (int nPAVNo)
PAV2250A_SetViewTotalRatioPercent sends the command to set Total Ratio units to %.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioDB](#) (int nPAVNo)
PAV2250A_SetViewTotalRatioDB sends the command to set Total Ratio units to dB.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewTotalRatioRatio](#) (int nPAVNo)
PAV2250A_SetViewTotalRatioRatio sends the command to set Total Ratio units to Ratio.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewTotalRatioUnits sends the command to get the view units index for the Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewTotalRatioUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewTotalRatioUnitsText sends the command to get the view units for the Total Ratio view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyHZ](#) (int nPAVNo)
PAV2250A_SetViewFrequencyHZ sends the command to set Frequency units to Hz.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFrequencyKHZ](#) (int nPAVNo)
PAV2250A_SetViewFrequencyKHZ sends the command to set Frequency units to KHz.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewFrequencyUnits sends the command to get the view units index for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFrequencyUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewFrequencyUnitsText sends the command to get the view units for the Frequency view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainV](#) (int nPAVNo)
PAV2250A_SetViewMainV sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainMV](#) (int nPAVNo)
PAV2250A_SetViewMainMV sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainPercent](#) (int nPAVNo)
PAV2250A_SetViewMainPercent sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainDB](#) (int nPAVNo)
PAV2250A_SetViewMainDB sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainRatio](#) (int nPAVNo)
PAV2250A_SetViewMainRatio sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMain360](#) (int nPAVNo)
PAV2250A_SetViewMain360 sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMain180](#) (int nPAVNo)
PAV2250A_SetViewMain180 sends the command to set the view units for the Main view. (only important when main view mode is set to INDEPENDENT_VIEW)
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainUnits](#) (int nPAVNo, int *pnUnits)
PAV2250A_GetViewMainUnits sends the command to get the view units index for the Main view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainUnitsText](#) (int nPAVNo, char *pszUnits)
PAV2250A_GetViewMainUnitsText sends the command to get the view units for the Main view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagOffset](#) (int nPAVNo, float fOffset)
PAV2250A_SetViewFundMagOffset sends the command to set the offset for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagOffset](#) (int nPAVNo, float *pfOffset)

- PAV2250A_GetViewFundMagOffset sends the command to get the offset for the Fundamental Magnitude view.*

 - [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseOffset](#) (int nPAVNo, float fOffset)

PAV2250A_SetViewInPhaseOffset sends the command to set the offset for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseOffset](#) (int nPAVNo, float *pfOffset)

PAV2250A_GetViewInPhaseOffset sends the command to get the offset for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadOffset](#) (int nPAVNo, float fOffset)

PAV2250A_SetViewQuadOffset sends the command to set the offset for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadOffset](#) (int nPAVNo, float *pfOffset)

PAV2250A_GetViewQuadOffset sends the command to get the offset for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhaseOffset](#) (int nPAVNo, float fOffset)

PAV2250A_SetViewPhaseOffset sends the command to set the offset for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseOffset](#) (int nPAVNo, float *pfOffset)

PAV2250A_GetViewPhaseOffset sends the command to get the offset for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltOffset](#) (int nPAVNo, float fOffset)

PAV2250A_SetViewSigVoltOffset sends the command to set the offset for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltOffset](#) (int nPAVNo, float *pfOffset)

PAV2250A_GetViewSigVoltOffset sends the command to get the offset for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltOffset](#) (int nPAVNo, float fOffset)

PAV2250A_SetViewRefVoltOffset sends the command to set the offset for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltOffset](#) (int nPAVNo, float *pfOffset)

PAV2250A_GetViewRefVoltOffset sends the command to get the offset for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainOffset](#) (int nPAVNo, float fOffset)

PAV2250A_SetViewMainOffset sends the command to set the offset for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainOffset](#) (int nPAVNo, float *pfOffset)

PAV2250A_GetViewMainOffset sends the command to get the offset for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewFundMagScale](#) (int nPAVNo, float fScale)

PAV2250A_SetViewFundMagScale sends the command to set the scale for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewFundMagScale](#) (int nPAVNo, float *pfScale)

PAV2250A_GetViewFundMagScale sends the command to get the scale for the Fundamental Magnitude view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewInPhaseScale](#) (int nPAVNo, float fScale)

PAV2250A_SetViewInPhaseScale sends the command to set the scale for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewInPhaseScale](#) (int nPAVNo, float *pfScale)

PAV2250A_GetViewInPhaseScale sends the command to get the scale for the In Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewQuadScale](#) (int nPAVNo, float fScale)

PAV2250A_SetViewQuadScale sends the command to set the scale for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewQuadScale](#) (int nPAVNo, float *pfScale)

PAV2250A_GetViewQuadScale sends the command to get the scale for the Quad view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewPhaseScale](#) (int nPAVNo, float fScale)

PAV2250A_SetViewPhaseScale sends the command to set the scale for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewPhaseScale](#) (int nPAVNo, float *pfScale)

PAV2250A_GetViewPhaseScale sends the command to get the scale for the Phase view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewSigVoltScale](#) (int nPAVNo, float fScale)

PAV2250A_SetViewSigVoltScale sends the command to set the scale for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewSigVoltScale](#) (int nPAVNo, float *pfScale)

PAV2250A_GetViewSigVoltScale sends the command to get the scale for the Sig Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewRefVoltScale](#) (int nPAVNo, float fScale)

PAV2250A_SetViewRefVoltScale sends the command to set the scale for the Ref Volt view.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewRefVoltScale](#) (int nPAVNo, float *pfScale)

PAV2250A_GetViewRefVoltScale sends the command to get the scale for the Ref Volt view.

- [_PAV2250AFUNC](#) int [PAV2250A_SetViewMainScale](#) (int nPAVNo, float fScale)
PAV2250A_SetViewMainScale sends the command to set the scale for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewMainScale](#) (int nPAVNo, float *pfScale)
PAV2250A_GetViewMainScale sends the command to get the scale for the Main view. (only comes into play when the main view option is set to "INDEPENDENT_VIEW")
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexMaxFieldWidth](#) (int nPAVNo, int nViewIndex, int nMaxFieldWidth)
PAV2250A_SetViewIndexMaxFieldWidth sends the command to set the max field width for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexMaxFieldWidth](#) (int nPAVNo, int nViewIndex, int *pnMaxFieldWidth)
PAV2250A_GetViewIndexMaxFieldWidth sends the command to get the max field width for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexUnits](#) (int nPAVNo, int nViewIndex, int nUnits)
PAV2250A_SetViewIndexUnits sends the command to set the units for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexUnits](#) (int nPAVNo, int nViewIndex, int *pnUnits)
PAV2250A_GetViewIndexUnits sends the command to get the units for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexUnitsText](#) (int nPAVNo, int nViewIndex, char *pszUnits)
PAV2250A_GetViewIndexUnitsText sends the command to get the view units for the view text specified by the index.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexOffset](#) (int nPAVNo, int nViewIndex, float fOffset)
PAV2250A_SetViewIndexOffset sends the command to set the offset for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexOffset](#) (int nPAVNo, int nViewIndex, float *pfOffset)
PAV2250A_GetViewIndexOffset sends the command to get the offset for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_SetViewIndexScale](#) (int nPAVNo, int nViewIndex, float fScale)
PAV2250A_SetViewIndexScale sends the command to set the scale for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetViewIndexScale](#) (int nPAVNo, int nViewIndex, float *pfScale)
PAV2250A_GetViewIndexScale sends the command to get the scale for the specified view index.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRef](#) (int nPAVNo, int *pnViewIndex)
PAV2250A_GetRef sends the command requesting the current Ref Voltage View Index to the PAV2250A device. Index options (1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefText](#) (int nPAVNo, char *pszViewText)
PAV2250A_GetRef sends the command requesting the current Ref Voltage View Text to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRef](#) (int nPAVNo, int nViewIndex)
PAV2250A_SetRef sends the command to set the Ref Voltage View Index to the PAV2250A device. Index options (1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalSum](#) (int nPAVNo)
PAV2250A_SetRefTotalSum sends the command to set the Ref Voltage View to TotalSum to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalRMS_AC](#) (int nPAVNo)
PAV2250A_SetRefTotalRMS_AC
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefTotalRMS_ACDC](#) (int nPAVNo)
PAV2250A_SetRefTotalRMS_ACDC sends the command to set the Ref Voltage View to TotalRMS_ACDC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetRefDC](#) (int nPAVNo)
PAV2250A_SetRefDC sends the command to set the Ref Voltage View to DC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalSum](#) (int nPAVNo, float *pfRefTotalSum)
PAV2250A_GetRefTotalSum sends the command to get the Ref Voltage Total Sum value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalRMS_AC](#) (int nPAVNo, float *pfRefRMS)
PAV2250A_GetRefTotalRMS_AC sends the command to get the Ref Voltage Total RMS AC value from the PAV2250-A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefTotalRMS_ACDC](#) (int nPAVNo, float *pfRefRMS_ACDC)
PAV2250A_GetRefTotalRMS_ACDC sends the command to get the Ref Voltage Total RMS AC+DC value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetRefDC](#) (int nPAVNo, float *pfRefDC)

- PAV2250A_GetRefDC* sends the command to get the Ref Voltage DC value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSig](#) (int nPAVNo, int *pnViewIndex)

PAV2250A_GetSig sends the command requesting the current Sig Voltage View Index to the PAV2250A device. Index options (0=TotalSum 1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigText](#) (int nPAVNo, char *pszViewText)

PAV2250A_GetSig sends the command requesting the current Sig Voltage View Text to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSig](#) (int nPAVNo, int nViewIndex)

PAV2250A_SetSig sends the command to set the Sig Voltage View Index to the PAV2250A device. Index options (0=TotalSum, 1=TotalRMS_AC, 2=TotalRMS_ACDC, 3=DC)
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalSum](#) (int nPAVNo)

PAV2250A_SetSigTotalSum sends the command to set the Sig Voltage View to TotalSum to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalRMS_AC](#) (int nPAVNo)

PAV2250A_SetSigTotalRMS_AC sends the command to set the Sig Voltage View to TotalRMS_AC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigTotalRMS_ACDC](#) (int nPAVNo)

PAV2250A_SetSigTotalRMS_ACDC sends the command to set the Sig Voltage View to TotalRMS_ACDC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_SetSigDC](#) (int nPAVNo)

PAV2250A_SetSigDC sends the command to set the Sig Voltage View to DC to the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalSum](#) (int nPAVNo, float *pfSigTotalSum)

PAV2250A_GetSigTotalSum sends the command to get the Sig Voltage Total Sum value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalRMS_AC](#) (int nPAVNo, float *pfSigRMS)

PAV2250A_GetSigTotalRMS_AC sends the command to get the Sig Voltage Total RMS AC value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigTotalRMS_ACDC](#) (int nPAVNo, float *pfSigRMS_ACDC)

PAV2250A_GetSigTotalRMS_ACDC sends the command to get the Sig Voltage Total RMS AC+DC value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigDC](#) (int nPAVNo, float *pfSigDC)

PAV2250A_GetSigDC sends the command to get the Sig Voltage DC value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSigOffset](#) (int nPAVNo, float *pfSigOffset)

PAV2250A_GetSigOffset sends the command to get the Sig Offset value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTotalRatio](#) (int nPAVNo, float *pfTotalRatio)

PAV2250A_GetTotalRatio sends the command to get the Total Ratio value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetTHD](#) (int nPAVNo, float *pfTHD)

PAV2250A_GetTHD sends the command to get the THD value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetFrequency](#) (int nPAVNo, float *pfFreq)

PAV2250A_GetFrequency sends the command to get the Frequency value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetSampleRateIndex](#) (int nPAVNo, int *pnSampleRateIndex)

PAV2250A_GetSampleRateIndex sends the command to get the Sample Rate Index value from the PAV2250A device.
- [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedDataState](#) (int nPAVNo, bool *pbDataBufferEnabled, bool *pbDataBufferComplete, int *pnBufferedDataType, int *pnSampleRate, int *pnBufferSize, bool *pbDataBufferReady)

FOR INTERNAL USE ONLY! *PAV2250A_GetBufferedDataState* is responsible for returning data buffering state information.
- [_PAV2250AFUNC](#) int [PAV2250A_SetBufferedDataState](#) (int nPAVNo, bool bDataBufferEnabled, int nBufferedDataType, int nSampleRate, int nBufferSize)

FOR INTERNAL USE ONLY! *PAV2250A_SetBufferedDataState* is responsible for setting the buffer collection parameters of DataType, SampleRate, BufferSize and BufferEnabled.
- [_PAV2250AFUNC](#) int [PAV2250A_BufferCapture](#) (int nPAVNo, int nDataType, int nSampleRate, int nSize)

FOR INTERNAL USE ONLY! *PAV2250A_BufferCapture* is responsible for Starting a data buffer capture.
- [_PAV2250AFUNC](#) int [PAV2250A_BufferStop](#) (int nPAVNo)

FOR INTERNAL USE ONLY! *PAV2250A_BufferStop* is responsible for stopping all buffer captures for the given PAV.

- [_PAV2250AFUNC](#) int [PAV2250A_BufferGet](#) (int nPAVNo, int nStart, int nCount, char *pszData)
FOR INTERNAL USE ONLY! PAV2250A_BufferGet is responsible for retrieving a captured buffer.
- [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedPageIndex](#) (int nPAVNo, int *pnPageIndex)
FOR INTERNAL USE ONLY! PAV2250A_GetBufferedPageIndex returns the current buffered page index.
- [_PAV2250AFUNC](#) int [PAV2250A_SetBufferedPageIndex](#) (int nPAVNo, int nPageIndex)
FOR INTERNAL USE ONLY! PAV2250A_SetBufferedPageIndex is responsible for setting the current page buffer.
- [_PAV2250AFUNC](#) int [PAV2250A_GetBufferedData](#) (int nPAVNo, int nPageIndex, int nRequestedBufferSize, int nTotalOverallDataCnt, int *pDataCnt, int *paBufferData)
FOR INTERNAL USE ONLY! PAV2250A_GetBufferedData is responsible for retrieving the buffered data.
- int [HandleBufferedDataState](#) (int nPAVNo, bool *pbDataBufferEnabled, bool *pbDataBufferComplete, int *pnBufferedDataType, int *pnSampleRate, int *pnBufferSize, bool *pbDataBufferReady)
FOR INTERNAL USE ONLY! HandleBufferedDataState returns Data Buffering configuration information.
- int [HandleBufferDataSetup](#) (int nPAVNo, int nStartRec, int nEndRec)
FOR INTERNAL USE ONLY! HandleBufferDataSetup is responsible for setting some parameters needed to capture data.
- int [HandleBufferedDataValues](#) (int nPAVNo, int nStartRec, int *pnDataCnt, int *paBufferData)
FOR INTERNAL USE ONLY! HandleBufferedDataValues is responsible for loading the buffered data values into the output buffer.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefFreq](#) (int nPAVNo, float fFreq)
PAV2250A_SetIntRefFreq sends a command to the PAV 2250A to force the internal reference frequency to the specified value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefFreq](#) (int nPAVNo, float *pfFreq)
PAV2250A_GetIntRefFreq is responsible for returning internal reference frequency value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefVolt](#) (int nPAVNo, float fVolt)
PAV2250A_SetIntRefVolt sends a command to the PAV 2250A to force the internal reference voltage to the specified value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefVolt](#) (int nPAVNo, float *pfVolt)
PAV2250A_GetIntRefVolt is responsible for returning internal reference voltage value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefOutputState](#) (int nPAVNo, int nOutputState)
PAV2250A_SetIntRefOutputState is responsible for setting the output state. (OFF/ON).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefOutputState](#) (int nPAVNo, int *pnOutputState)
PAV2250A_GetIntRefOutputState is responsible for returning the index associated with the current reference state output.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefOverCurState](#) (int nPAVNo, int *pnOvrCurState)
PAV2250A_GetIntRefOutputState is responsible for returning the over current state.
- [_PAV2250AFUNC](#) int [PAV2250A_ResetIntRefOverCur](#) (int nPAVNo)
PAV2250A_ResetIntRefOverCur is responsible for resetting the over current flag.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefRemoteSense](#) (int nPAVNo, int nRemoteSense)
PAV2250A_SetRefGenRmtSenseState is responsible for setting the the sense state. (Disable/Enable).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefRemoteSense](#) (int nPAVNo, int *pnRemoteSense)
PAV2250A_GetIntRefRemoteSense is responsible for returning the index associated with the current Sense line state.
- [_PAV2250AFUNC](#) int [PAV2250A_SetIntRefSenseDir](#) (int nPAVNo, int nSenseDir)
PAV2250A_SetIntRefSenseDir is responsible for setting the sense direction. (Front/Back).
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefSenseDir](#) (int nPAVNo, int *pnSenseDir)
PAV2250A_GetIntRefSenseDir is responsible for returning the index associated with the current Sense line direction.
- [_PAV2250AFUNC](#) int [PAV2250A_GetIntRefMeasCur](#) (int nPAVNo, int *pnMeasCur)
PAV2250A_GetIntRefMeasCur is responsible for returning the internal reference measured current.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTEEnabled](#) (int nPAVNo, bool *pbLVDTEEnabled)
PAV2250A_GetLVDTEEnabled returns whether or not the PAV's LVDT calculations are enabled or disabled.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTEEnabledText](#) (int nPAVNo, char *pszLVDTEEnabledText)
PAV2250A_GetLVDTEEnabledText retrieves the LVDT enabled text: "Disabled" or "Enabled"

- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTEnabled](#) (int nPAVNo)
PAV2250A_SetLVDTEnabled enables the LVDT calculations : 1 ("Enabled")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTDisabled](#) (int nPAVNo)
PAV2250A_SetLVDTDisabled disables the LVDT calculations : 0 ("Disabled")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTPosition](#) (int nPAVNo, float *pfPosition)
PAV2250A_GetLVDTPosition retrieves the current LVDT position.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTType](#) (int nPAVNo, int *pnLVDTType)
PAV2250A_GetLVDTType retrieves the LVDT mode type: 2 = 2-Wire, 3 = 3-Wire, 4 = 4-Wire
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTTypeText](#) (int nPAVNo, char *pszLVDTTypeText)
PAV2250A_GetLVDTTypeText retrieves the LVDT mode type text: "2-Wire", "3-Wire", "4-Wire"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType](#) (int nPAVNo, int nLVDTTypeIndex)
PAV2250A_SetLVDTType sets the LVDT mode type: 2 = "2-Wire", 3 = "3-Wire", 4 = "4-Wire"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType2Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType2Wire sets the LVDT mode type to: 2 ("2-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType3Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType3Wire sets the LVDT mode type to: 3 ("3-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTType4Wire](#) (int nPAVNo)
PAV2250A_SetLVDTType4Wire sets the LVDT mode type to: 4 ("4-Wire")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTSignal](#) (int nPAVNo, int *pnLVDTSignal)
PAV2250A_GetLVDTSignal retrieves the LVDT signal: 0 = Fund, 1 = Total, 2 = INPH
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTSignalText](#) (int nPAVNo, char *pszLVDTSignalText)
PAV2250A_GetLVDTSignalText retrieves the LVDT signal text: "Fund", "Total", "INPH"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignal](#) (int nPAVNo, int nLVDTSignalIndex)
PAV2250A_SetLVDTSignal sets the LVDT signal: 0 = "Fund", 1 = "Total", 2 = "INPH"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalINPH](#) (int nPAVNo)
PAV2250A_SetLVDTSignalINPH sets the LVDT signal to: 2 ("INPH")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalFund](#) (int nPAVNo)
PAV2250A_SetLVDTSignalFund sets the LVDT signal to: 0 ("Fund")
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTSignalTotal](#) (int nPAVNo)
PAV2250A_SetLVDTSignalTotal sets the LVDT signal to: 1 ("Total")
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTVA](#) (int nPAVNo, float *pfLVDTVA)
PAV2250A_GetLVDTVA retrieves the current LVDT VA value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTVB](#) (int nPAVNo, float *pfLVDTVB)
PAV2250A_GetLVDTVB retrieves the current LVDT VB value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTPOFF](#) (int nPAVNo, float fLVDTPOFF)
PAV2250A_SetLVDTPOFF sends a command to the PAV 2250A to set a Phase Offset of the desired value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTPOFF](#) (int nPAVNo, float *pfLVDTPOFF)
PAV2250A_GetLVDTPOFF retrieves the current LVDT Phase Offset (POFF) value.
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDTScale](#) (int nPAVNo, float fLVDTScale)
PAV2250A_SetLVDTScale sends a command to the PAV 2250A to set a Scale of the desired value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDTScale](#) (int nPAVNo, float *pfLVDTScale)
PAV2250A_GetLVDTScale retrieves the current LVDT Scale value.
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDT4WireAlgorithm](#) (int nPAVNo, int *pnLVDT4WireAlgorithm)
PAV2250A_GetLVDT4WireAlgorithm retrieves the LVDT 4-Wire Algorithm: 0 = V(a), V(a) + V(b), 1 = V(a), V(b)
- [_PAV2250AFUNC](#) int [PAV2250A_GetLVDT4WireAlgorithmText](#) (int nPAVNo, char *pszLVDT4WireAlgorithmText)
PAV2250A_GetLVDT4WireAlgorithmText retrieves the LVDT 4-Wire Algorithm text: "V(a), V(a) + V(b)", "V(a), V(b)"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDT4WireAlgorithm](#) (int nPAVNo, int nLVDT4WireAlgorithmIndex)
PAV2250A_SetLVDT4WireAlgorithm sets the LVDT 4-Wire Algorithm: 0 = "V(a), V(a) + V(b)", 1 = "V(a), V(b)"
- [_PAV2250AFUNC](#) int [PAV2250A_SetLVDT4WireVA_VAPLUSVB](#) (int nPAVNo)
PAV2250A_SetLVDT4WireVA_VAPLUSVB sets the LVDT 4-Wire Algorithm to: 0 ("V(a), V(a) + V(b)")

- `_PAV2250AFUNC` int `PAV2250A_SetLVDT4WireVA_VB` (int nPAVNo)
PAV2250A_SetLVDT4WireVA_VB sets the LVDT 4-Wire Algorithm to: 1 ("V(a), V(b)")
- `_PAV2250AFUNC` int `PAV2250A_GetRefTotal` (int nPAVNo, float *pfTotal)
PAV2250A_GetRefTotal retrieves the reference Total value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefFundMag` (int nPAVNo, float *pfFundMag)
PAV2250A_GetRefFundMag retrieves the reference Fundamental (Magnitude) value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefInPhase` (int nPAVNo, float *pfInPhase)
PAV2250A_GetRefInPhase retrieves the reference InPhase value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefQuad` (int nPAVNo, float *pfQuad)
PAV2250A_GetRefQuad retrieves the reference Quad value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefPhase` (int nPAVNo, float *pfPhase)
PAV2250A_GetRefPhase retrieves the reference Phase value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefTHD` (int nPAVNo, float *pfTHD)
PAV2250A_GetRefTHD retrieves the reference THD value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefTotalRatio` (int nPAVNo, float *pfTotalRatio)
PAV2250A_GetRefTotalRatio retrieves the reference Total Ratio value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefFundMagRatio` (int nPAVNo, float *pfFundMagRatio)
PAV2250A_GetRefFundMagRatio retrieves the reference Fundamental (Magnitude) Ratio value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefInPhaseRatio` (int nPAVNo, float *pfInPhaseRatio)
PAV2250A_GetRefInPhaseRatio retrieves the reference InPhase Ratio value.
- `_PAV2250AFUNC` int `PAV2250A_GetRefQuadRatio` (int nPAVNo, float *pfQuadRatio)
PAV2250A_GetRefQuadRatio retrieves the reference Quad Ratio value.

4.2.1 Macro Definition Documentation

4.2.1.1 `#define _PAV2250AClass __declspec(dllexport)`

4.2.1.2 `#define _PAV2250AFUNC __declspec(dllexport)`

4.2.1.3 `#define bool BOOL`

4.2.2 Enumeration Type Documentation

4.2.2.1 enum `PAV_IEEE_LANGUAGES`

Enumerator

`PAV2250A_NATIVE`
`PAV2250_LEGACY`

4.2.2.2 enum `PAV_INT_REF_OUTPUT_STATE`

Enumerator

`PAV_INT_REF_OUT_NOT_AVAILABLE`
`PAV_INT_REF_OUT_AVAILABLE`

4.2.2.3 enum `PAV_INT_REF_OVER_CURRENT_STATE`

Enumerator

`PAV_NO_OVER_CURRENT`
`PAV_OVER_CURRENT`

4.2.2.4 enum PAV_INT_REF_REMOTE_GEN_DIR

Enumerator

PAV_REF_GEN_DIR_BACK
PAV_REF_GEN_DIR_FRONT

4.2.2.5 enum PAV_LVDT_4WIRE_ALGORITHM

Enumerator

PAV_LVDT_4WIRE_VA_VAPLUSVB
PAV_LVDT_4WIRE_VA_VB

4.2.2.6 enum PAV_LVDT_SIGNAL

Enumerator

PAV_LVDT_SIGNAL_FUND
PAV_LVDT_SIGNAL_TOTAL
PAV_LVDT_SIGNAL_INPH

4.2.2.7 enum PAV_LVDT_TYPE

Enumerator

PAV_LVDT_TYPE_2WIRE
PAV_LVDT_TYPE_3WIRE
PAV_LVDT_TYPE_4WIRE

4.2.2.8 enum PAV_OPTIONS_AUTOSAVE

Enumerator

PAV_AUTOSAVE_DISABLE
PAV_AUTOSAVE_ENABLE

4.2.2.9 enum PAV_OPTIONS_AUTOUNITS

Enumerator

PAV_AUTOUNITS_DISABLE
PAV_AUTOUNITS_ENABLE

4.2.2.10 enum PAV_OPTIONS_DATEDISPLAY

Enumerator

PAV_DATEDISPLAY_TEXT
PAV_DATEDISPLAY_NUMERIC

4.2.2.11 enum PAV_OPTIONS_MAINDISPLAY

Enumerator

PAV_MAINDISPLAY_INDEPENDENT
PAV_MAINDISPLAY_LINKED

4.2.2.12 enum PAV_OPTIONS_RATIO

Enumerator

PAV_RATIO_APPLYONLYTOMAIN
PAV_RATIO_APPLYTORELEVANT

4.2.2.13 enum PAV_OPTIONS_SIGNALINPUT

Enumerator

PAV_SIGNALINPUT_FRONT
PAV_SIGNALINPUT_BACK

4.2.2.14 enum PAV_OPTIONS_TIMEDISPLAY

Enumerator

PAV_TIMEDISPLAY_AMPM
PAV_TIMEDISPLAY_MILITARY

4.2.2.15 enum PAV_OPTIONS_TOUCHSCREEN

Enumerator

PAV_TOUCHSCREEN_DISABLE
PAV_TOUCHSCREEN_ENABLE

4.2.2.16 enum PAV_Ranges

Enumerator

PAV_RANGE_50MV
PAV_RANGE_100MV
PAV_RANGE_200MV
PAV_RANGE_500MV
PAV_RANGE_1V
PAV_RANGE_2V
PAV_RANGE_5V
PAV_RANGE_10V
PAV_RANGE_20V
PAV_RANGE_50V
PAV_RANGE_100V
PAV_RANGE_200V
PAV_RANGE_500V

4.2.2.17 enum PAV_ReadModes

Enumerator

PAV_READMODE_SIGREF
PAV_READMODE_REFREF
PAV_READMODE_SIGSIG
PAV_READMODE_REFSIG

4.2.2.18 enum PAV_REMOTE_SENSE_STATE

Enumerator

PAV_RMT_SENSE_DISABLE
PAV_RMT_SENSE_ENABLE

4.2.2.19 enum PAV_STATUS

Enumerator

PAV_SUCCESS
PAV_ERROR_OPEN_PAV_SESSION
PAV_ERROR_PAVNO
PAV_ERROR_ADDRS
PAV_ERROR_LANG
PAV_ERROR_DATA
PAV_ERROR_RANGE
PAV_ERROR_WRITE
PAV_ERROR_USB_CONNECTION
PAV_ERROR_ETHER_CONNECTION
PAV_ERROR_FUNC_NOT_SUPPORTED
PAV_ERROR_TRIGGER
PAV_STATUS_LAST

4.2.2.20 enum PAV_Tabs

Enumerator

PAV_TAB_MAIN
PAV_TAB_REFERENCE
PAV_TAB_HARMONICS
PAV_TAB_CUSTOM

4.2.2.21 enum PAV_Units

Enumerator

PAV_UNITS_V
PAV_UNITS_MV
PAV_UNITS_RATIO
PAV_UNITS_PERCENT
PAV_UNITS_DB
PAV_UNITS_360
PAV_UNITS_180
PAV_UNITS_HZ
PAV_UNITS_KHZ

4.2.2.22 enum PAV_Views

Enumerator

PAV_VIEW_FUNDMAG
PAV_VIEW_INPHASE
PAV_VIEW_QUAD
PAV_VIEW_PHASE
PAV_VIEW_THD
PAV_VIEW_SIGVOLT
PAV_VIEW_REFVOLT
PAV_VIEW_SIGOFFSET
PAV_VIEW_TOTALRATIO
PAV_VIEW_FREQ
PAV_VIEW_MAIN

4.2.2.23 enum PAV_VoltViews

Enumerator

PAV_VOLTVIEW_TOTALSUM
PAV_VOLTVIEW_TOTALRMS_AC
PAV_VOLTVIEW_TOTALRMS_ACDC
PAV_VOLTVIEW_DC

Index

- [_PAV2250AClass](#)
 - [PAV2250ADII.h, 295](#)
- [_PAV2250AFUNC](#)
 - [PAV2250ADII.h, 295](#)
- [bool](#)
 - [PAV2250ADII.h, 295](#)
- [Calibration Functions, 87](#)
 - [PAV2250A_Calibrate, 87](#)
 - [PAV2250A_GetCalState, 87](#)
- [Command Functions, 9](#)
 - [PAV2250A_IsStable, 9](#)
 - [PAV2250A_PerformGetID, 9](#)
 - [PAV2250A_PerformGetTotalData, 11](#)
 - [PAV2250A_PerformGetTotalDataRaw, 11](#)
 - [PAV2250A_PerformGroupExecuteTrigger, 12](#)
 - [PAV2250A_ResetDefaultValues, 12](#)
- [Configuration Functions, 48](#)
 - [PAV2250A_GetCommState, 52](#)
 - [PAV2250A_GetCustView1, 52](#)
 - [PAV2250A_GetCustView1Text, 53](#)
 - [PAV2250A_GetCustView2, 53](#)
 - [PAV2250A_GetCustView2Text, 54](#)
 - [PAV2250A_GetCustView3, 55](#)
 - [PAV2250A_GetCustView3Text, 55](#)
 - [PAV2250A_GetCustView4, 56](#)
 - [PAV2250A_GetCustView4Text, 57](#)
 - [PAV2250A_GetHoldDataState, 57](#)
 - [PAV2250A_GetIEEELang, 58](#)
 - [PAV2250A_GetIEEELangText, 58](#)
 - [PAV2250A_GetMainView, 58](#)
 - [PAV2250A_GetMainViewText, 59](#)
 - [PAV2250A_GetReadMode, 60](#)
 - [PAV2250A_GetReadModeText, 60](#)
 - [PAV2250A_GetScreenBrightness, 61](#)
 - [PAV2250A_GetTabView, 61](#)
 - [PAV2250A_GetTabViewText, 62](#)
 - [PAV2250A_GoToLocal, 62](#)
 - [PAV2250A_SetCustView1, 62](#)
 - [PAV2250A_SetCustView1FundMag, 63](#)
 - [PAV2250A_SetCustView1InPhase, 63](#)
 - [PAV2250A_SetCustView1Phase, 64](#)
 - [PAV2250A_SetCustView1Quad, 64](#)
 - [PAV2250A_SetCustView1RefVolt, 64](#)
 - [PAV2250A_SetCustView1SigVolt, 65](#)
 - [PAV2250A_SetCustView1THD, 65](#)
 - [PAV2250A_SetCustView2, 65](#)
 - [PAV2250A_SetCustView2FundMag, 66](#)
 - [PAV2250A_SetCustView2InPhase, 66](#)
 - [PAV2250A_SetCustView2Phase, 67](#)
 - [PAV2250A_SetCustView2Quad, 67](#)
 - [PAV2250A_SetCustView2RefVolt, 67](#)
 - [PAV2250A_SetCustView2SigVolt, 68](#)
 - [PAV2250A_SetCustView2THD, 68](#)
 - [PAV2250A_SetCustView3, 68](#)
 - [PAV2250A_SetCustView3FundMag, 69](#)
 - [PAV2250A_SetCustView3InPhase, 69](#)
 - [PAV2250A_SetCustView3Phase, 70](#)
 - [PAV2250A_SetCustView3Quad, 70](#)
 - [PAV2250A_SetCustView3RefVolt, 70](#)
 - [PAV2250A_SetCustView3SigVolt, 71](#)
 - [PAV2250A_SetCustView3THD, 71](#)
 - [PAV2250A_SetCustView4, 71](#)
 - [PAV2250A_SetCustView4FundMag, 72](#)
 - [PAV2250A_SetCustView4InPhase, 72](#)
 - [PAV2250A_SetCustView4Phase, 73](#)
 - [PAV2250A_SetCustView4Quad, 73](#)
 - [PAV2250A_SetCustView4RefVolt, 73](#)
 - [PAV2250A_SetCustView4SigVolt, 74](#)
 - [PAV2250A_SetCustView4THD, 74](#)
 - [PAV2250A_SetHoldDataState, 74](#)
 - [PAV2250A_SetIEEELang, 75](#)
 - [PAV2250A_SetIEEELang2250ANative, 75](#)
 - [PAV2250A_SetIEEELang2250Legacy, 76](#)
 - [PAV2250A_SetMainView, 76](#)
 - [PAV2250A_SetMainViewFundMag, 77](#)
 - [PAV2250A_SetMainViewInPhase, 77](#)
 - [PAV2250A_SetMainViewPhase, 78](#)
 - [PAV2250A_SetMainViewQuad, 78](#)
 - [PAV2250A_SetMainViewRefVolt, 78](#)
 - [PAV2250A_SetMainViewSigVolt, 79](#)
 - [PAV2250A_SetMainViewTHD, 79](#)
 - [PAV2250A_SetReadMode, 79](#)
 - [PAV2250A_SetReadModeRefRef, 80](#)
 - [PAV2250A_SetReadModeRefSig, 80](#)
 - [PAV2250A_SetReadModeSigRef, 81](#)
 - [PAV2250A_SetReadModeSigSig, 81](#)
 - [PAV2250A_SetRemoteEthernet, 81](#)
 - [PAV2250A_SetRemoteIEEE, 82](#)
 - [PAV2250A_SetRemoteJ1, 82](#)
 - [PAV2250A_SetRemoteUSB, 82](#)
 - [PAV2250A_SetScreenBrightness, 84](#)
 - [PAV2250A_SetTabView, 84](#)
 - [PAV2250A_SetTabViewCustom, 85](#)
 - [PAV2250A_SetTabViewHarmonics, 85](#)
 - [PAV2250A_SetTabViewLVDT, 85](#)
 - [PAV2250A_SetTabViewMain, 86](#)
 - [PAV2250A_SetTabViewReference, 86](#)

- Connect/Disconnect Functions, 5
 - PAV2250A_ConnectViaEthernet, 5
 - PAV2250A_ConnectViaIEEE, 6
 - PAV2250A_ConnectViaUSB, 6
 - PAV2250A_DisconnectEthernet, 7
 - PAV2250A_DisconnectIEEE, 7
 - PAV2250A_DisconnectUSB, 7
 - PAV2250A_GetPAV2250ADeviceIDN, 8
 - PAV2250A_GetPAV2250AUSBDeviceCnt, 8
- Data Buffer Functions, 210
 - HandleBufferDataSetup, 210
 - HandleBufferedDataState, 211
 - HandleBufferedDataValues, 213
 - PAV2250A_BufferCapture, 214
 - PAV2250A_BufferGet, 216
 - PAV2250A_BufferStop, 216
 - PAV2250A_GetBufferedData, 216
 - PAV2250A_GetBufferedDataState, 217
 - PAV2250A_GetBufferedPageIndex, 217
 - PAV2250A_SetBufferedDataState, 218
 - PAV2250A_SetBufferedPageIndex, 220
- DllMain
 - PAV2250ADll.cpp, 274
- ExecuteRemoteCmd
 - Local Functions (not exported), 228
- ExecuteRemoteCmdBool
 - Local Functions (not exported), 229
- ExecuteRemoteCmdFloat
 - Local Functions (not exported), 229
- ExecuteRemoteCmdInt
 - Local Functions (not exported), 229
- ExecuteRemoteCmdString
 - Local Functions (not exported), 231
- GetRange
 - Local Functions (not exported), 232
- HandleBufferDataSetup
 - Data Buffer Functions, 210
- HandleBufferedDataState
 - Data Buffer Functions, 211
- HandleBufferedDataValues
 - Data Buffer Functions, 213
- Harmonic Functions, 14
 - PAV2250A_GetHarmonicInPhase, 15
 - PAV2250A_GetHarmonicMagnitude, 16
 - PAV2250A_GetHarmonicPhase, 16
 - PAV2250A_GetHarmonicQuad, 17
 - PAV2250A_GetHarmonicRatioInPhase, 17
 - PAV2250A_GetHarmonicRatioMagnitude, 17
 - PAV2250A_GetHarmonicRatioPhase, 18
 - PAV2250A_GetHarmonicRatioQuad, 18
 - PAV2250A_GetHarmonicRatioState, 19
 - PAV2250A_PerformGetHarmonicDataRow, 19
 - PAV2250A_PerformGetHarmonics, 20
 - PAV2250A_PerformGetHarmonicsRatio, 20
 - PAV2250A_SetHarmonicRatioState, 21
 - PAV2250A_ViewHarmonic, 21
 - PAV2250A_ViewNextHarmonicGroup, 22
 - PAV2250A_ViewPrevHarmonicGroup, 22
- HexStr2DeclInt
 - Local Functions (not exported), 232
- IEEE Functions, 46
 - PAV2250A_IEEECLS, 46
 - PAV2250A_IEEEGetErrors, 46
 - PAV2250A_IEEEReset, 46
- Independent Component Functions, 207
 - PAV2250A_GetFrequency, 207
 - PAV2250A_GetSampleRateIndex, 207
 - PAV2250A_GetSigOffset, 208
 - PAV2250A_GetTHD, 208
 - PAV2250A_GetTotalRatio, 208
- Internal Reference Functions, 221
 - PAV2250A_GetIntRefFreq, 221
 - PAV2250A_GetIntRefMeasCur, 222
 - PAV2250A_GetIntRefOutputState, 222
 - PAV2250A_GetIntRefOverCurState, 222
 - PAV2250A_GetIntRefRemoteSense, 223
 - PAV2250A_GetIntRefSenseDir, 223
 - PAV2250A_GetIntRefVolt, 224
 - PAV2250A_ResetIntRefOverCur, 224
 - PAV2250A_SetIntRefFreq, 224
 - PAV2250A_SetIntRefOutputState, 226
 - PAV2250A_SetIntRefRemoteSense, 226
 - PAV2250A_SetIntRefSenseDir, 227
 - PAV2250A_SetIntRefVolt, 227
- IsLanguageTypeLegacy
 - Local Functions (not exported), 233
- IsLanguageTypeNative
 - Local Functions (not exported), 233
- LVDT Functions, 235
 - PAV2250A_GetLVDT4WireAlgorithm, 236
 - PAV2250A_GetLVDT4WireAlgorithmText, 236
 - PAV2250A_GetLVDTEnabled, 236
 - PAV2250A_GetLVDTEnabledText, 237
 - PAV2250A_GetLVDTPOFF, 237
 - PAV2250A_GetLVDTPosition, 238
 - PAV2250A_GetLVDTScale, 238
 - PAV2250A_GetLVDTSignal, 238
 - PAV2250A_GetLVDTSignalText, 239
 - PAV2250A_GetLVDTType, 239
 - PAV2250A_GetLVDTTypeText, 240
 - PAV2250A_GetLVDTVA, 240
 - PAV2250A_GetLVDTVB, 240
 - PAV2250A_SetLVDT4WireAlgorithm, 241
 - PAV2250A_SetLVDT4WireVA_VB, 241
 - PAV2250A_SetLVDTDisabled, 243
 - PAV2250A_SetLVDTEnabled, 243
 - PAV2250A_SetLVDTPOFF, 243
 - PAV2250A_SetLVDTScale, 244
 - PAV2250A_SetLVDTSignal, 244
 - PAV2250A_SetLVDTSignalFund, 244
 - PAV2250A_SetLVDTSignalINPH, 245
 - PAV2250A_SetLVDTSignalTotal, 245

- PAV2250A_SetLVDTType, 245
- PAV2250A_SetLVDTType2Wire, 246
- PAV2250A_SetLVDTType3Wire, 246
- PAV2250A_SetLVDTType4Wire, 246
- Local Functions (not exported), 228
 - ExecuteRemoteCmd, 228
 - ExecuteRemoteCmdBool, 229
 - ExecuteRemoteCmdFloat, 229
 - ExecuteRemoteCmdInt, 229
 - ExecuteRemoteCmdString, 231
 - GetRange, 232
 - HexStr2Declnt, 232
 - IsLanguageTypeLegacy, 233
 - IsLanguageTypeNative, 233
 - ParseForCommaSeparatedDataElements, 233
 - SendIEEEMessage, 233
 - WaitForResponse, 234
- Miscellaneous Functions, 88
 - PAV2250A_LastCmdSent, 88
 - PAV2250A_MaxRetry, 88
 - PAV2250A_QueryCommand, 88
 - PAV2250A_WriteCommand, 89
- PAV2250_LEGACY
 - PAV2250ADII.h, 295
- PAV2250A_NATIVE
 - PAV2250ADII.h, 295
- PAV2250ADII.h
 - PAV2250_LEGACY, 295
 - PAV2250A_NATIVE, 295
 - PAV_AUTOSAVE_DISABLE, 296
 - PAV_AUTOSAVE_ENABLE, 296
 - PAV_AUTOUNITS_DISABLE, 296
 - PAV_AUTOUNITS_ENABLE, 296
 - PAV_DATEDISPLAY_NUMERIC, 296
 - PAV_DATEDISPLAY_TEXT, 296
 - PAV_ERROR_ADDRS, 298
 - PAV_ERROR_DATA, 298
 - PAV_ERROR_ETHER_CONNECTION, 298
 - PAV_ERROR_FUNC_NOT_SUPPORTED, 298
 - PAV_ERROR_LANG, 298
 - PAV_ERROR_OPEN_PAV_SESSION, 298
 - PAV_ERROR_PAVNO, 298
 - PAV_ERROR_RANGE, 298
 - PAV_ERROR_TRIGGER, 298
 - PAV_ERROR_USB_CONNECTION, 298
 - PAV_ERROR_WRITE, 298
 - PAV_INT_REF_OUT_AVAILABLE, 295
 - PAV_INT_REF_OUT_NOT_AVAILABLE, 295
 - PAV_LVDT_4WIRE_VA_VAPLUSVB, 296
 - PAV_LVDT_4WIRE_VA_VB, 296
 - PAV_LVDT_SIGNAL_FUND, 296
 - PAV_LVDT_SIGNAL_INPH, 296
 - PAV_LVDT_SIGNAL_TOTAL, 296
 - PAV_LVDT_TYPE_2WIRE, 296
 - PAV_LVDT_TYPE_3WIRE, 296
 - PAV_LVDT_TYPE_4WIRE, 296
 - PAV_MAINDISPLAY_INDEPENDENT, 297
 - PAV_MAINDISPLAY_LINKED, 297
 - PAV_NO_OVER_CURRENT, 295
 - PAV_OVER_CURRENT, 295
 - PAV_RANGE_100MV, 297
 - PAV_RANGE_100V, 297
 - PAV_RANGE_10V, 297
 - PAV_RANGE_1V, 297
 - PAV_RANGE_200MV, 297
 - PAV_RANGE_200V, 297
 - PAV_RANGE_20V, 297
 - PAV_RANGE_2V, 297
 - PAV_RANGE_500MV, 297
 - PAV_RANGE_500V, 297
 - PAV_RANGE_50MV, 297
 - PAV_RANGE_50V, 297
 - PAV_RANGE_5V, 297
 - PAV_RATIO_APPLYONLYTOMAIN, 297
 - PAV_RATIO_APPLYTORELEVANT, 297
 - PAV_READMODE_REFREF, 298
 - PAV_READMODE_REFSIG, 298
 - PAV_READMODE_SIGREF, 298
 - PAV_READMODE_SIGSIG, 298
 - PAV_REF_GEN_DIR_BACK, 296
 - PAV_REF_GEN_DIR_FRONT, 296
 - PAV_RMT_SENSE_DISABLE, 298
 - PAV_RMT_SENSE_ENABLE, 298
 - PAV_SIGNALINPUT_BACK, 297
 - PAV_SIGNALINPUT_FRONT, 297
 - PAV_STATUS_LAST, 298
 - PAV_SUCCESS, 298
 - PAV_TAB_CUSTOM, 298
 - PAV_TAB_HARMONICS, 298
 - PAV_TAB_MAIN, 298
 - PAV_TAB_REFERENCE, 298
 - PAV_TIMEDISPLAY_AMP, 297
 - PAV_TIMEDISPLAY_MILITARY, 297
 - PAV_TOUCHSCREEN_DISABLE, 297
 - PAV_TOUCHSCREEN_ENABLE, 297
 - PAV_UNITS_180, 299
 - PAV_UNITS_360, 299
 - PAV_UNITS_DB, 299
 - PAV_UNITS_HZ, 299
 - PAV_UNITS_KHZ, 299
 - PAV_UNITS_MV, 299
 - PAV_UNITS_PERCENT, 299
 - PAV_UNITS_RATIO, 299
 - PAV_UNITS_V, 299
 - PAV_VIEW_FREQ, 299
 - PAV_VIEW_FUNDMAG, 299
 - PAV_VIEW_INPHASE, 299
 - PAV_VIEW_MAIN, 299
 - PAV_VIEW_PHASE, 299
 - PAV_VIEW_QUAD, 299
 - PAV_VIEW_REFVOLT, 299
 - PAV_VIEW_SIGOFFSET, 299
 - PAV_VIEW_SIGVOLT, 299
 - PAV_VIEW_THD, 299
 - PAV_VIEW_TOTALRATIO, 299

- PAV_VOLTVIEW_DC, [299](#)
- PAV_VOLTVIEW_TOTALRMS_AC, [299](#)
- PAV_VOLTVIEW_TOTALRMS_ACDC, [299](#)
- PAV_VOLTVIEW_TOTALSUM, [299](#)
- PAV_AUTOSAVE_DISABLE
 - PAV2250ADII.h, [296](#)
- PAV_AUTOSAVE_ENABLE
 - PAV2250ADII.h, [296](#)
- PAV_AUTOUNITS_DISABLE
 - PAV2250ADII.h, [296](#)
- PAV_AUTOUNITS_ENABLE
 - PAV2250ADII.h, [296](#)
- PAV_DATEDISPLAY_NUMERIC
 - PAV2250ADII.h, [296](#)
- PAV_DATEDISPLAY_TEXT
 - PAV2250ADII.h, [296](#)
- PAV_ERROR_ADDRS
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_DATA
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_ETHER_CONNECTION
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_FUNC_NOT_SUPPORTED
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_LANG
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_OPEN_PAV_SESSION
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_PAVNO
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_RANGE
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_TRIGGER
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_USB_CONNECTION
 - PAV2250ADII.h, [298](#)
- PAV_ERROR_WRITE
 - PAV2250ADII.h, [298](#)
- PAV_INT_REF_OUT_AVAILABLE
 - PAV2250ADII.h, [295](#)
- PAV_INT_REF_OUT_NOT_AVAILABLE
 - PAV2250ADII.h, [295](#)
- PAV_LVDT_4WIRE_VA_VAPLUSVB
 - PAV2250ADII.h, [296](#)
- PAV_LVDT_4WIRE_VA_VB
 - PAV2250ADII.h, [296](#)
- PAV_LVDT_SIGNAL_FUND
 - PAV2250ADII.h, [296](#)
- PAV_LVDT_SIGNAL_INPH
 - PAV2250ADII.h, [296](#)
- PAV_LVDT_SIGNAL_TOTAL
 - PAV2250ADII.h, [296](#)
- PAV_LVDT_TYPE_2WIRE
 - PAV2250ADII.h, [296](#)
- PAV_LVDT_TYPE_3WIRE
 - PAV2250ADII.h, [296](#)
- PAV_LVDT_TYPE_4WIRE
 - PAV2250ADII.h, [296](#)
- PAV_MAINDISPLAY_INDEPENDENT
 - PAV2250ADII.h, [297](#)
- PAV_MAINDISPLAY_LINKED
 - PAV2250ADII.h, [297](#)
- PAV_NO_OVER_CURRENT
 - PAV2250ADII.h, [295](#)
- PAV_OVER_CURRENT
 - PAV2250ADII.h, [295](#)
- PAV_RANGE_100MV
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_100V
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_10V
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_1V
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_200MV
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_200V
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_20V
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_2V
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_500MV
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_500V
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_50MV
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_50V
 - PAV2250ADII.h, [297](#)
- PAV_RANGE_5V
 - PAV2250ADII.h, [297](#)
- PAV_RATIO_APPLYONLYTOMAIN
 - PAV2250ADII.h, [297](#)
- PAV_RATIO_APPLYTORELEVANT
 - PAV2250ADII.h, [297](#)
- PAV_READMODE_REFREF
 - PAV2250ADII.h, [298](#)
- PAV_READMODE_REFSIG
 - PAV2250ADII.h, [298](#)
- PAV_READMODE_SIGREF
 - PAV2250ADII.h, [298](#)
- PAV_READMODE_SIGSIG
 - PAV2250ADII.h, [298](#)
- PAV_REF_GEN_DIR_BACK
 - PAV2250ADII.h, [296](#)
- PAV_REF_GEN_DIR_FRONT
 - PAV2250ADII.h, [296](#)
- PAV_RMT_SENSE_DISABLE
 - PAV2250ADII.h, [298](#)
- PAV_RMT_SENSE_ENABLE
 - PAV2250ADII.h, [298](#)
- PAV_SIGNALINPUT_BACK
 - PAV2250ADII.h, [297](#)
- PAV_SIGNALINPUT_FRONT
 - PAV2250ADII.h, [297](#)

- PAV_STATUS_LAST
PAV2250ADII.h, 298
- PAV_SUCCESS
PAV2250ADII.h, 298
- PAV_TAB_CUSTOM
PAV2250ADII.h, 298
- PAV_TAB_HARMONICS
PAV2250ADII.h, 298
- PAV_TAB_MAIN
PAV2250ADII.h, 298
- PAV_TAB_REFERENCE
PAV2250ADII.h, 298
- PAV_TIMEDISPLAY_AMP
PAV2250ADII.h, 297
- PAV_TIMEDISPLAY_MILITARY
PAV2250ADII.h, 297
- PAV_TOUCHSCREEN_DISABLE
PAV2250ADII.h, 297
- PAV_TOUCHSCREEN_ENABLE
PAV2250ADII.h, 297
- PAV_UNITS_180
PAV2250ADII.h, 299
- PAV_UNITS_360
PAV2250ADII.h, 299
- PAV_UNITS_DB
PAV2250ADII.h, 299
- PAV_UNITS_HZ
PAV2250ADII.h, 299
- PAV_UNITS_KHZ
PAV2250ADII.h, 299
- PAV_UNITS_MV
PAV2250ADII.h, 299
- PAV_UNITS_PERCENT
PAV2250ADII.h, 299
- PAV_UNITS_RATIO
PAV2250ADII.h, 299
- PAV_UNITS_V
PAV2250ADII.h, 299
- PAV_VIEW_FREQ
PAV2250ADII.h, 299
- PAV_VIEW_FUNDMAG
PAV2250ADII.h, 299
- PAV_VIEW_INPHASE
PAV2250ADII.h, 299
- PAV_VIEW_MAIN
PAV2250ADII.h, 299
- PAV_VIEW_PHASE
PAV2250ADII.h, 299
- PAV_VIEW_QUAD
PAV2250ADII.h, 299
- PAV_VIEW_REFVOLT
PAV2250ADII.h, 299
- PAV_VIEW_SIGOFFSET
PAV2250ADII.h, 299
- PAV_VIEW_SIGVOLT
PAV2250ADII.h, 299
- PAV_VIEW_THD
PAV2250ADII.h, 299
- PAV_VIEW_TOTALRATIO
PAV2250ADII.h, 299
- PAV_VOLTVIEW_DC
PAV2250ADII.h, 299
- PAV_VOLTVIEW_TOTALRMS_AC
PAV2250ADII.h, 299
- PAV_VOLTVIEW_TOTALRMS_ACDC
PAV2250ADII.h, 299
- PAV_VOLTVIEW_TOTALSUM
PAV2250ADII.h, 299
- PAV2250A_BufferCapture
Data Buffer Functions, 214
- PAV2250A_BufferGet
Data Buffer Functions, 216
- PAV2250A_BufferStop
Data Buffer Functions, 216
- PAV2250A_Calibrate
Calibration Functions, 87
- PAV2250A_ConnectViaEthernet
Connect/Disconnect Functions, 5
- PAV2250A_ConnectViaIEEE
Connect/Disconnect Functions, 6
- PAV2250A_ConnectViaUSB
Connect/Disconnect Functions, 6
- PAV2250A_DisconnectEthernet
Connect/Disconnect Functions, 7
- PAV2250A_DisconnectIEEE
Connect/Disconnect Functions, 7
- PAV2250A_DisconnectUSB
Connect/Disconnect Functions, 7
- PAV2250A_GetAutoSaveOption
Setup Options Functions, 94
- PAV2250A_GetAutoSaveOptionText
Setup Options Functions, 95
- PAV2250A_GetAutoUnitsOption
Setup Options Functions, 95
- PAV2250A_GetAutoUnitsOptionText
Setup Options Functions, 96
- PAV2250A_GetBufferedData
Data Buffer Functions, 216
- PAV2250A_GetBufferedDataState
Data Buffer Functions, 217
- PAV2250A_GetBufferedPageIndex
Data Buffer Functions, 217
- PAV2250A_GetCalState
Calibration Functions, 87
- PAV2250A_GetCommState
Configuration Functions, 52
- PAV2250A_GetCustView1
Configuration Functions, 52
- PAV2250A_GetCustView1Text
Configuration Functions, 53
- PAV2250A_GetCustView2
Configuration Functions, 53
- PAV2250A_GetCustView2Text
Configuration Functions, 54
- PAV2250A_GetCustView3
Configuration Functions, 55

- PAV2250A_GetCustView3Text
 - Configuration Functions, [55](#)
- PAV2250A_GetCustView4
 - Configuration Functions, [56](#)
- PAV2250A_GetCustView4Text
 - Configuration Functions, [57](#)
- PAV2250A_GetDateDisplayOption
 - Setup Options Functions, [96](#)
- PAV2250A_GetDateDisplayOptionText
 - Setup Options Functions, [97](#)
- PAV2250A_GetFrequency
 - Independent Component Functions, [207](#)
- PAV2250A_GetHarmonicInPhase
 - Harmonic Functions, [15](#)
- PAV2250A_GetHarmonicMagnitude
 - Harmonic Functions, [16](#)
- PAV2250A_GetHarmonicPhase
 - Harmonic Functions, [16](#)
- PAV2250A_GetHarmonicQuad
 - Harmonic Functions, [17](#)
- PAV2250A_GetHarmonicRatioInPhase
 - Harmonic Functions, [17](#)
- PAV2250A_GetHarmonicRatioMagnitude
 - Harmonic Functions, [17](#)
- PAV2250A_GetHarmonicRatioPhase
 - Harmonic Functions, [18](#)
- PAV2250A_GetHarmonicRatioQuad
 - Harmonic Functions, [18](#)
- PAV2250A_GetHarmonicRatioState
 - Harmonic Functions, [19](#)
- PAV2250A_GetHoldDataState
 - Configuration Functions, [57](#)
- PAV2250A_GetIEEELang
 - Configuration Functions, [58](#)
- PAV2250A_GetIEEELangText
 - Configuration Functions, [58](#)
- PAV2250A_GetIntRefFreq
 - Internal Reference Functions, [221](#)
- PAV2250A_GetIntRefMeasCur
 - Internal Reference Functions, [222](#)
- PAV2250A_GetIntRefOutputState
 - Internal Reference Functions, [222](#)
- PAV2250A_GetIntRefOverCurState
 - Internal Reference Functions, [222](#)
- PAV2250A_GetIntRefRemoteSense
 - Internal Reference Functions, [223](#)
- PAV2250A_GetIntRefSenseDir
 - Internal Reference Functions, [223](#)
- PAV2250A_GetIntRefVolt
 - Internal Reference Functions, [224](#)
- PAV2250A_GetLVDT4WireAlgorithm
 - LVDT Functions, [236](#)
- PAV2250A_GetLVDT4WireAlgorithmText
 - LVDT Functions, [236](#)
- PAV2250A_GetLVDTEnabled
 - LVDT Functions, [236](#)
- PAV2250A_GetLVDTEnabledText
 - LVDT Functions, [237](#)
- PAV2250A_GetLVDTPOFF
 - LVDT Functions, [237](#)
- PAV2250A_GetLVDTPosition
 - LVDT Functions, [238](#)
- PAV2250A_GetLVDTScale
 - LVDT Functions, [238](#)
- PAV2250A_GetLVDTSignal
 - LVDT Functions, [238](#)
- PAV2250A_GetLVDTSignalText
 - LVDT Functions, [239](#)
- PAV2250A_GetLVDTType
 - LVDT Functions, [239](#)
- PAV2250A_GetLVDTTypeText
 - LVDT Functions, [240](#)
- PAV2250A_GetLVDTVA
 - LVDT Functions, [240](#)
- PAV2250A_GetLVDTVB
 - LVDT Functions, [240](#)
- PAV2250A_GetMainDisplayOption
 - Setup Options Functions, [97](#)
- PAV2250A_GetMainDisplayOptionText
 - Setup Options Functions, [98](#)
- PAV2250A_GetMainView
 - Configuration Functions, [58](#)
- PAV2250A_GetMainViewText
 - Configuration Functions, [59](#)
- PAV2250A_GetNullMeterRangePercent
 - Setup Options Functions, [98](#)
- PAV2250A_GetPAV2250ADeviceIDN
 - Connect/Disconnect Functions, [8](#)
- PAV2250A_GetPAV2250AUSBDeviceCnt
 - Connect/Disconnect Functions, [8](#)
- PAV2250A_GetReadMode
 - Configuration Functions, [60](#)
- PAV2250A_GetReadModeText
 - Configuration Functions, [60](#)
- PAV2250A_GetRef
 - Reference Voltage Functions, [193](#)
- PAV2250A_GetRefAutoRange
 - Reference Range Functions, [24](#)
- PAV2250A_GetRefDC
 - Reference Voltage Functions, [194](#)
- PAV2250A_GetRefFundMag
 - Reference Only Functions, [249](#)
- PAV2250A_GetRefFundMagRatio
 - Reference Only Functions, [249](#)
- PAV2250A_GetRefInPhase
 - Reference Only Functions, [250](#)
- PAV2250A_GetRefInPhaseRatio
 - Reference Only Functions, [250](#)
- PAV2250A_GetRefPhase
 - Reference Only Functions, [250](#)
- PAV2250A_GetRefQuad
 - Reference Only Functions, [251](#)
- PAV2250A_GetRefQuadRatio
 - Reference Only Functions, [251](#)
- PAV2250A_GetRefRangeActualIndex
 - Reference Range Functions, [24](#)

- PAV2250A_GetRefRangeConfigIndex
 - Reference Range Functions, [25](#)
- PAV2250A_GetRefRangeIndexSettings
 - Reference Range Functions, [25](#)
- PAV2250A_GetRefRangeString
 - Reference Range Functions, [27](#)
- PAV2250A_GetRefTHD
 - Reference Only Functions, [251](#)
- PAV2250A_GetRefText
 - Reference Voltage Functions, [194](#)
- PAV2250A_GetRefTotal
 - Reference Only Functions, [252](#)
- PAV2250A_GetRefTotalRMS_AC
 - Reference Voltage Functions, [194](#)
- PAV2250A_GetRefTotalRMS_ACDC
 - Reference Voltage Functions, [196](#)
- PAV2250A_GetRefTotalRatio
 - Reference Only Functions, [252](#)
- PAV2250A_GetRefTotalSum
 - Reference Voltage Functions, [196](#)
- PAV2250A_GetSampleRateIndex
 - Independent Component Functions, [207](#)
- PAV2250A_GetScreenBrightness
 - Configuration Functions, [61](#)
- PAV2250A_GetSig
 - Signal Voltage Functions, [200](#)
- PAV2250A_GetSigAutoRange
 - Signal Range Functions, [36](#)
- PAV2250A_GetSigDC
 - Signal Voltage Functions, [201](#)
- PAV2250A_GetSigOffset
 - Independent Component Functions, [208](#)
- PAV2250A_GetSigRangeActualIndex
 - Signal Range Functions, [36](#)
- PAV2250A_GetSigRangeConfigIndex
 - Signal Range Functions, [37](#)
- PAV2250A_GetSigRangeIndexSettings
 - Signal Range Functions, [37](#)
- PAV2250A_GetSigRangeString
 - Signal Range Functions, [39](#)
- PAV2250A_GetSigText
 - Signal Voltage Functions, [201](#)
- PAV2250A_GetSigTotalRMS_AC
 - Signal Voltage Functions, [201](#)
- PAV2250A_GetSigTotalRMS_ACDC
 - Signal Voltage Functions, [203](#)
- PAV2250A_GetSigTotalSum
 - Signal Voltage Functions, [203](#)
- PAV2250A_GetSignalInputOption
 - Setup Options Functions, [99](#)
- PAV2250A_GetSignalInputOptionText
 - Setup Options Functions, [99](#)
- PAV2250A_GetTHD
 - Independent Component Functions, [208](#)
- PAV2250A_GetTabView
 - Configuration Functions, [61](#)
- PAV2250A_GetTabViewText
 - Configuration Functions, [62](#)
- PAV2250A_GetTimeDisplayOption
 - Setup Options Functions, [100](#)
- PAV2250A_GetTimeDisplayOptionText
 - Setup Options Functions, [100](#)
- PAV2250A_GetTimeWndActual
 - Time Window Functions, [90](#)
- PAV2250A_GetTimeWndAuto
 - Time Window Functions, [90](#)
- PAV2250A_GetTimeWndOverride
 - Time Window Functions, [91](#)
- PAV2250A_GetTotalRatio
 - Independent Component Functions, [208](#)
- PAV2250A_GetTouchscreenOption
 - Setup Options Functions, [101](#)
- PAV2250A_GetTouchscreenOptionText
 - Setup Options Functions, [101](#)
- PAV2250A_GetViewFrequencyConfig
 - View Configuration Functions, [113](#)
- PAV2250A_GetViewFrequencyMaxFieldWidth
 - View Max Field Width Functions, [123](#)
- PAV2250A_GetViewFrequencyUnits
 - View Units Functions, [136](#)
- PAV2250A_GetViewFrequencyUnitsText
 - View Units Functions, [137](#)
- PAV2250A_GetViewFundMagConfig
 - View Configuration Functions, [113](#)
- PAV2250A_GetViewFundMagMaxFieldWidth
 - View Max Field Width Functions, [123](#)
- PAV2250A_GetViewFundMagOffset
 - View Offset Functions, [164](#)
- PAV2250A_GetViewFundMagScale
 - View Scale Functions, [173](#)
- PAV2250A_GetViewFundMagUnits
 - View Units Functions, [137](#)
- PAV2250A_GetViewFundMagUnitsText
 - View Units Functions, [137](#)
- PAV2250A_GetViewInPhaseConfig
 - View Configuration Functions, [115](#)
- PAV2250A_GetViewInPhaseMaxFieldWidth
 - View Max Field Width Functions, [123](#)
- PAV2250A_GetViewInPhaseOffset
 - View Offset Functions, [165](#)
- PAV2250A_GetViewInPhaseScale
 - View Scale Functions, [174](#)
- PAV2250A_GetViewInPhaseUnits
 - View Units Functions, [139](#)
- PAV2250A_GetViewInPhaseUnitsText
 - View Units Functions, [139](#)
- PAV2250A_GetViewIndexConfig
 - View Configuration Functions, [114](#)
- PAV2250A_GetViewIndexMaxFieldWidth
 - View by Index Functions, [182](#)
- PAV2250A_GetViewIndexOffset
 - View by Index Functions, [183](#)
- PAV2250A_GetViewIndexScale
 - View by Index Functions, [184](#)
- PAV2250A_GetViewIndexUnits
 - View by Index Functions, [185](#)

- PAV2250A_GetViewIndexUnitsText
 - View by Index Functions, [186](#)
- PAV2250A_GetViewMainConfig
 - View Configuration Functions, [115](#)
- PAV2250A_GetViewMainMaxFieldWidth
 - View Max Field Width Functions, [124](#)
- PAV2250A_GetViewMainOffset
 - View Offset Functions, [165](#)
- PAV2250A_GetViewMainScale
 - View Scale Functions, [174](#)
- PAV2250A_GetViewMainUnits
 - View Units Functions, [140](#)
- PAV2250A_GetViewMainUnitsText
 - View Units Functions, [140](#)
- PAV2250A_GetViewPhaseConfig
 - View Configuration Functions, [117](#)
- PAV2250A_GetViewPhaseMaxFieldWidth
 - View Max Field Width Functions, [124](#)
- PAV2250A_GetViewPhaseOffset
 - View Offset Functions, [165](#)
- PAV2250A_GetViewPhaseScale
 - View Scale Functions, [174](#)
- PAV2250A_GetViewPhaseUnits
 - View Units Functions, [142](#)
- PAV2250A_GetViewPhaseUnitsText
 - View Units Functions, [142](#)
- PAV2250A_GetViewQuadConfig
 - View Configuration Functions, [117](#)
- PAV2250A_GetViewQuadMaxFieldWidth
 - View Max Field Width Functions, [124](#)
- PAV2250A_GetViewQuadOffset
 - View Offset Functions, [166](#)
- PAV2250A_GetViewQuadScale
 - View Scale Functions, [175](#)
- PAV2250A_GetViewQuadUnits
 - View Units Functions, [143](#)
- PAV2250A_GetViewQuadUnitsText
 - View Units Functions, [143](#)
- PAV2250A_GetViewRefVoltConfig
 - View Configuration Functions, [118](#)
- PAV2250A_GetViewRefVoltMaxFieldWidth
 - View Max Field Width Functions, [125](#)
- PAV2250A_GetViewRefVoltOffset
 - View Offset Functions, [166](#)
- PAV2250A_GetViewRefVoltScale
 - View Scale Functions, [175](#)
- PAV2250A_GetViewRefVoltUnits
 - View Units Functions, [144](#)
- PAV2250A_GetViewRefVoltUnitsText
 - View Units Functions, [144](#)
- PAV2250A_GetViewSigOffsetConfig
 - View Configuration Functions, [119](#)
- PAV2250A_GetViewSigOffsetMaxFieldWidth
 - View Max Field Width Functions, [125](#)
- PAV2250A_GetViewSigOffsetUnits
 - View Units Functions, [145](#)
- PAV2250A_GetViewSigOffsetUnitsText
 - View Units Functions, [145](#)
- PAV2250A_GetViewSigVoltConfig
 - View Configuration Functions, [119](#)
- PAV2250A_GetViewSigVoltMaxFieldWidth
 - View Max Field Width Functions, [126](#)
- PAV2250A_GetViewSigVoltOffset
 - View Offset Functions, [166](#)
- PAV2250A_GetViewSigVoltScale
 - View Scale Functions, [175](#)
- PAV2250A_GetViewSigVoltUnits
 - View Units Functions, [146](#)
- PAV2250A_GetViewSigVoltUnitsText
 - View Units Functions, [146](#)
- PAV2250A_GetViewTHDConfig
 - View Configuration Functions, [120](#)
- PAV2250A_GetViewTHDMaxFieldWidth
 - View Max Field Width Functions, [126](#)
- PAV2250A_GetViewTHDUnits
 - View Units Functions, [147](#)
- PAV2250A_GetViewTHDUnitsText
 - View Units Functions, [147](#)
- PAV2250A_GetViewTotalRatioConfig
 - View Configuration Functions, [120](#)
- PAV2250A_GetViewTotalRatioMaxFieldWidth
 - View Max Field Width Functions, [126](#)
- PAV2250A_GetViewTotalRatioUnits
 - View Units Functions, [148](#)
- PAV2250A_GetViewTotalRatioUnitsText
 - View Units Functions, [148](#)
- PAV2250A_GoToLocal
 - Configuration Functions, [62](#)
- PAV2250A_IEEECLS
 - IEEE Functions, [46](#)
- PAV2250A_IEEEGetErrors
 - IEEE Functions, [46](#)
- PAV2250A_IEEEReset
 - IEEE Functions, [46](#)
- PAV2250A_IsStable
 - Command Functions, [9](#)
- PAV2250A_LastCmdSent
 - Miscellaneous Functions, [88](#)
- PAV2250A_MaxRetry
 - Miscellaneous Functions, [88](#)
- PAV2250A_PerformGetHarmonicDataRow
 - Harmonic Functions, [19](#)
- PAV2250A_PerformGetHarmonics
 - Harmonic Functions, [20](#)
- PAV2250A_PerformGetHarmonicsRatio
 - Harmonic Functions, [20](#)
- PAV2250A_PerformGetID
 - Command Functions, [9](#)
- PAV2250A_PerformGetTotalData
 - Command Functions, [11](#)
- PAV2250A_PerformGetTotalDataRow
 - Command Functions, [11](#)
- PAV2250A_PerformGroupExecuteTrigger
 - Command Functions, [12](#)
- PAV2250A_QueryCommand
 - Miscellaneous Functions, [88](#)

- PAV2250A_ResetDefaultValues
Command Functions, [12](#)
- PAV2250A_ResetIntRefOverCur
Internal Reference Functions, [224](#)
- PAV2250A_SetAutoSaveDisable
Setup Options Functions, [101](#)
- PAV2250A_SetAutoSaveEnable
Setup Options Functions, [102](#)
- PAV2250A_SetAutoSaveOption
Setup Options Functions, [102](#)
- PAV2250A_SetAutoUnitsDisable
Setup Options Functions, [102](#)
- PAV2250A_SetAutoUnitsEnable
Setup Options Functions, [103](#)
- PAV2250A_SetAutoUnitsOption
Setup Options Functions, [103](#)
- PAV2250A_SetBufferedDataState
Data Buffer Functions, [218](#)
- PAV2250A_SetBufferedPageIndex
Data Buffer Functions, [220](#)
- PAV2250A_SetCustView1
Configuration Functions, [62](#)
- PAV2250A_SetCustView1FundMag
Configuration Functions, [63](#)
- PAV2250A_SetCustView1InPhase
Configuration Functions, [63](#)
- PAV2250A_SetCustView1Phase
Configuration Functions, [64](#)
- PAV2250A_SetCustView1Quad
Configuration Functions, [64](#)
- PAV2250A_SetCustView1RefVolt
Configuration Functions, [64](#)
- PAV2250A_SetCustView1SigVolt
Configuration Functions, [65](#)
- PAV2250A_SetCustView1THD
Configuration Functions, [65](#)
- PAV2250A_SetCustView2
Configuration Functions, [65](#)
- PAV2250A_SetCustView2FundMag
Configuration Functions, [66](#)
- PAV2250A_SetCustView2InPhase
Configuration Functions, [66](#)
- PAV2250A_SetCustView2Phase
Configuration Functions, [67](#)
- PAV2250A_SetCustView2Quad
Configuration Functions, [67](#)
- PAV2250A_SetCustView2RefVolt
Configuration Functions, [67](#)
- PAV2250A_SetCustView2SigVolt
Configuration Functions, [68](#)
- PAV2250A_SetCustView2THD
Configuration Functions, [68](#)
- PAV2250A_SetCustView3
Configuration Functions, [68](#)
- PAV2250A_SetCustView3FundMag
Configuration Functions, [69](#)
- PAV2250A_SetCustView3InPhase
Configuration Functions, [69](#)
- PAV2250A_SetCustView3Phase
Configuration Functions, [70](#)
- PAV2250A_SetCustView3Quad
Configuration Functions, [70](#)
- PAV2250A_SetCustView3RefVolt
Configuration Functions, [70](#)
- PAV2250A_SetCustView3SigVolt
Configuration Functions, [71](#)
- PAV2250A_SetCustView3THD
Configuration Functions, [71](#)
- PAV2250A_SetCustView4
Configuration Functions, [71](#)
- PAV2250A_SetCustView4FundMag
Configuration Functions, [72](#)
- PAV2250A_SetCustView4InPhase
Configuration Functions, [72](#)
- PAV2250A_SetCustView4Phase
Configuration Functions, [73](#)
- PAV2250A_SetCustView4Quad
Configuration Functions, [73](#)
- PAV2250A_SetCustView4RefVolt
Configuration Functions, [73](#)
- PAV2250A_SetCustView4SigVolt
Configuration Functions, [74](#)
- PAV2250A_SetCustView4THD
Configuration Functions, [74](#)
- PAV2250A_SetDateDisplayNumeric
Setup Options Functions, [104](#)
- PAV2250A_SetDateDisplayOption
Setup Options Functions, [104](#)
- PAV2250A_SetDateDisplayText
Setup Options Functions, [104](#)
- PAV2250A_SetHarmonicRatioState
Harmonic Functions, [21](#)
- PAV2250A_SetHoldDataState
Configuration Functions, [74](#)
- PAV2250A_SetIEEELang
Configuration Functions, [75](#)
- PAV2250A_SetIEEELang2250ANative
Configuration Functions, [75](#)
- PAV2250A_SetIEEELang2250Legacy
Configuration Functions, [76](#)
- PAV2250A_SetIntRefFreq
Internal Reference Functions, [224](#)
- PAV2250A_SetIntRefOutputState
Internal Reference Functions, [226](#)
- PAV2250A_SetIntRefRemoteSense
Internal Reference Functions, [226](#)
- PAV2250A_SetIntRefSenseDir
Internal Reference Functions, [227](#)
- PAV2250A_SetIntRefVolt
Internal Reference Functions, [227](#)
- PAV2250A_SetLVDT4WireAlgorithm
LVDT Functions, [241](#)
- PAV2250A_SetLVDT4WireVA_VB
LVDT Functions, [241](#)
- PAV2250A_SetLVDTDisabled
LVDT Functions, [243](#)

- PAV2250A_SetLVDTEnabled
 - LVDT Functions, [243](#)
- PAV2250A_SetLVDTPOFF
 - LVDT Functions, [243](#)
- PAV2250A_SetLVDTScale
 - LVDT Functions, [244](#)
- PAV2250A_SetLVDTSignal
 - LVDT Functions, [244](#)
- PAV2250A_SetLVDTSignalFund
 - LVDT Functions, [244](#)
- PAV2250A_SetLVDTSignalINPH
 - LVDT Functions, [245](#)
- PAV2250A_SetLVDTSignalTotal
 - LVDT Functions, [245](#)
- PAV2250A_SetLVDTType
 - LVDT Functions, [245](#)
- PAV2250A_SetLVDTType2Wire
 - LVDT Functions, [246](#)
- PAV2250A_SetLVDTType3Wire
 - LVDT Functions, [246](#)
- PAV2250A_SetLVDTType4Wire
 - LVDT Functions, [246](#)
- PAV2250A_SetMainDisplayIndependent
 - Setup Options Functions, [106](#)
- PAV2250A_SetMainDisplayLinked
 - Setup Options Functions, [106](#)
- PAV2250A_SetMainDisplayOption
 - Setup Options Functions, [106](#)
- PAV2250A_SetMainView
 - Configuration Functions, [76](#)
- PAV2250A_SetMainViewFundMag
 - Configuration Functions, [77](#)
- PAV2250A_SetMainViewInPhase
 - Configuration Functions, [77](#)
- PAV2250A_SetMainViewPhase
 - Configuration Functions, [78](#)
- PAV2250A_SetMainViewQuad
 - Configuration Functions, [78](#)
- PAV2250A_SetMainViewRefVolt
 - Configuration Functions, [78](#)
- PAV2250A_SetMainViewSigVolt
 - Configuration Functions, [79](#)
- PAV2250A_SetMainViewTHD
 - Configuration Functions, [79](#)
- PAV2250A_SetNullMeterRangePercent
 - Setup Options Functions, [107](#)
- PAV2250A_SetReadMode
 - Configuration Functions, [79](#)
- PAV2250A_SetReadModeRefRef
 - Configuration Functions, [80](#)
- PAV2250A_SetReadModeRefSig
 - Configuration Functions, [80](#)
- PAV2250A_SetReadModeSigRef
 - Configuration Functions, [81](#)
- PAV2250A_SetReadModeSigSig
 - Configuration Functions, [81](#)
- PAV2250A_SetRef
 - Reference Voltage Functions, [196](#)
- PAV2250A_SetRefAutoRange
 - Reference Range Functions, [28](#)
- PAV2250A_SetRefDC
 - Reference Voltage Functions, [198](#)
- PAV2250A_SetRefRange
 - Reference Range Functions, [29](#)
- PAV2250A_SetRefRange100MV
 - Reference Range Functions, [29](#)
- PAV2250A_SetRefRange100V
 - Reference Range Functions, [30](#)
- PAV2250A_SetRefRange10V
 - Reference Range Functions, [30](#)
- PAV2250A_SetRefRange1V
 - Reference Range Functions, [30](#)
- PAV2250A_SetRefRange200MV
 - Reference Range Functions, [31](#)
- PAV2250A_SetRefRange200V
 - Reference Range Functions, [31](#)
- PAV2250A_SetRefRange20V
 - Reference Range Functions, [31](#)
- PAV2250A_SetRefRange2V
 - Reference Range Functions, [32](#)
- PAV2250A_SetRefRange500MV
 - Reference Range Functions, [32](#)
- PAV2250A_SetRefRange500V
 - Reference Range Functions, [32](#)
- PAV2250A_SetRefRange50MV
 - Reference Range Functions, [33](#)
- PAV2250A_SetRefRange50V
 - Reference Range Functions, [33](#)
- PAV2250A_SetRefRange5V
 - Reference Range Functions, [33](#)
- PAV2250A_SetRefTotalRMS_AC
 - Reference Voltage Functions, [198](#)
- PAV2250A_SetRefTotalRMS_ACDC
 - Reference Voltage Functions, [199](#)
- PAV2250A_SetRefTotalSum
 - Reference Voltage Functions, [199](#)
- PAV2250A_SetRemoteEthernet
 - Configuration Functions, [81](#)
- PAV2250A_SetRemoteIEEE
 - Configuration Functions, [82](#)
- PAV2250A_SetRemoteJ1
 - Configuration Functions, [82](#)
- PAV2250A_SetRemoteUSB
 - Configuration Functions, [82](#)
- PAV2250A_SetScreenBrightness
 - Configuration Functions, [84](#)
- PAV2250A_SetSig
 - Signal Voltage Functions, [203](#)
- PAV2250A_SetSigAutoRange
 - Signal Range Functions, [40](#)
- PAV2250A_SetSigDC
 - Signal Voltage Functions, [205](#)
- PAV2250A_SetSigRange
 - Signal Range Functions, [40](#)
- PAV2250A_SetSigRange100MV
 - Signal Range Functions, [41](#)

- PAV2250A_SetSigRange100V
 - Signal Range Functions, [41](#)
- PAV2250A_SetSigRange10V
 - Signal Range Functions, [41](#)
- PAV2250A_SetSigRange1V
 - Signal Range Functions, [42](#)
- PAV2250A_SetSigRange200MV
 - Signal Range Functions, [42](#)
- PAV2250A_SetSigRange200V
 - Signal Range Functions, [42](#)
- PAV2250A_SetSigRange20V
 - Signal Range Functions, [43](#)
- PAV2250A_SetSigRange2V
 - Signal Range Functions, [43](#)
- PAV2250A_SetSigRange500MV
 - Signal Range Functions, [43](#)
- PAV2250A_SetSigRange500V
 - Signal Range Functions, [44](#)
- PAV2250A_SetSigRange50MV
 - Signal Range Functions, [44](#)
- PAV2250A_SetSigRange50V
 - Signal Range Functions, [44](#)
- PAV2250A_SetSigRange5V
 - Signal Range Functions, [45](#)
- PAV2250A_SetSigTotalRMS_AC
 - Signal Voltage Functions, [205](#)
- PAV2250A_SetSigTotalRMS_ACDC
 - Signal Voltage Functions, [206](#)
- PAV2250A_SetSigTotalSum
 - Signal Voltage Functions, [206](#)
- PAV2250A_SetSignalInputBack
 - Setup Options Functions, [107](#)
- PAV2250A_SetSignalInputFront
 - Setup Options Functions, [107](#)
- PAV2250A_SetSignalInputOption
 - Setup Options Functions, [108](#)
- PAV2250A_SetTabView
 - Configuration Functions, [84](#)
- PAV2250A_SetTabViewCustom
 - Configuration Functions, [85](#)
- PAV2250A_SetTabViewHarmonics
 - Configuration Functions, [85](#)
- PAV2250A_SetTabViewLVDT
 - Configuration Functions, [85](#)
- PAV2250A_SetTabViewMain
 - Configuration Functions, [86](#)
- PAV2250A_SetTabViewReference
 - Configuration Functions, [86](#)
- PAV2250A_SetTimeDisplayAMPM
 - Setup Options Functions, [108](#)
- PAV2250A_SetTimeDisplayMilitary
 - Setup Options Functions, [109](#)
- PAV2250A_SetTimeDisplayOption
 - Setup Options Functions, [109](#)
- PAV2250A_SetTimeWndAuto
 - Time Window Functions, [91](#)
- PAV2250A_SetTimeWndOverride
 - Time Window Functions, [91](#)
- PAV2250A_SetTouchscreenDisable
 - Setup Options Functions, [109](#)
- PAV2250A_SetTouchscreenEnable
 - Setup Options Functions, [111](#)
- PAV2250A_SetTouchscreenOption
 - Setup Options Functions, [111](#)
- PAV2250A_SetViewFrequencyHZ
 - View Units Functions, [149](#)
- PAV2250A_SetViewFrequencyKHZ
 - View Units Functions, [149](#)
- PAV2250A_SetViewFrequencyMaxFieldWidth
 - View Max Field Width Functions, [127](#)
- PAV2250A_SetViewFundMagDB
 - View Units Functions, [149](#)
- PAV2250A_SetViewFundMagMV
 - View Units Functions, [150](#)
- PAV2250A_SetViewFundMagMaxFieldWidth
 - View Max Field Width Functions, [127](#)
- PAV2250A_SetViewFundMagOffset
 - View Offset Functions, [168](#)
- PAV2250A_SetViewFundMagPercent
 - View Units Functions, [150](#)
- PAV2250A_SetViewFundMagRatio
 - View Units Functions, [150](#)
- PAV2250A_SetViewFundMagScale
 - View Scale Functions, [177](#)
- PAV2250A_SetViewFundMagV
 - View Units Functions, [151](#)
- PAV2250A_SetViewInPhaseDB
 - View Units Functions, [151](#)
- PAV2250A_SetViewInPhaseMV
 - View Units Functions, [151](#)
- PAV2250A_SetViewInPhaseMaxFieldWidth
 - View Max Field Width Functions, [127](#)
- PAV2250A_SetViewInPhaseOffset
 - View Offset Functions, [168](#)
- PAV2250A_SetViewInPhasePercent
 - View Units Functions, [152](#)
- PAV2250A_SetViewInPhaseRatio
 - View Units Functions, [152](#)
- PAV2250A_SetViewInPhaseScale
 - View Scale Functions, [177](#)
- PAV2250A_SetViewInPhaseV
 - View Units Functions, [152](#)
- PAV2250A_SetViewIndexMaxFieldWidth
 - View by Index Functions, [188](#)
- PAV2250A_SetViewIndexOffset
 - View by Index Functions, [189](#)
- PAV2250A_SetViewIndexScale
 - View by Index Functions, [190](#)
- PAV2250A_SetViewIndexUnits
 - View by Index Functions, [191](#)
- PAV2250A_SetViewMain180
 - View Units Functions, [153](#)
- PAV2250A_SetViewMain360
 - View Units Functions, [153](#)
- PAV2250A_SetViewMainDB
 - View Units Functions, [153](#)

- PAV2250A_SetViewMainMV
 - View Units Functions, [155](#)
- PAV2250A_SetViewMainMaxFieldWidth
 - View Max Field Width Functions, [129](#)
- PAV2250A_SetViewMainOffset
 - View Offset Functions, [168](#)
- PAV2250A_SetViewMainPercent
 - View Units Functions, [155](#)
- PAV2250A_SetViewMainRatio
 - View Units Functions, [155](#)
- PAV2250A_SetViewMainScale
 - View Scale Functions, [177](#)
- PAV2250A_SetViewMainV
 - View Units Functions, [157](#)
- PAV2250A_SetViewPhase180
 - View Units Functions, [157](#)
- PAV2250A_SetViewPhase360
 - View Units Functions, [157](#)
- PAV2250A_SetViewPhaseMaxFieldWidth
 - View Max Field Width Functions, [129](#)
- PAV2250A_SetViewPhaseOffset
 - View Offset Functions, [170](#)
- PAV2250A_SetViewPhaseScale
 - View Scale Functions, [179](#)
- PAV2250A_SetViewQuadDB
 - View Units Functions, [158](#)
- PAV2250A_SetViewQuadMV
 - View Units Functions, [158](#)
- PAV2250A_SetViewQuadMaxFieldWidth
 - View Max Field Width Functions, [129](#)
- PAV2250A_SetViewQuadOffset
 - View Offset Functions, [170](#)
- PAV2250A_SetViewQuadPercent
 - View Units Functions, [158](#)
- PAV2250A_SetViewQuadRatio
 - View Units Functions, [159](#)
- PAV2250A_SetViewQuadScale
 - View Scale Functions, [179](#)
- PAV2250A_SetViewQuadV
 - View Units Functions, [159](#)
- PAV2250A_SetViewRefVoltMV
 - View Units Functions, [159](#)
- PAV2250A_SetViewRefVoltMaxFieldWidth
 - View Max Field Width Functions, [131](#)
- PAV2250A_SetViewRefVoltOffset
 - View Offset Functions, [170](#)
- PAV2250A_SetViewRefVoltScale
 - View Scale Functions, [179](#)
- PAV2250A_SetViewRefVoltV
 - View Units Functions, [160](#)
- PAV2250A_SetViewSigOffsetMV
 - View Units Functions, [160](#)
- PAV2250A_SetViewSigOffsetMaxFieldWidth
 - View Max Field Width Functions, [131](#)
- PAV2250A_SetViewSigOffsetV
 - View Units Functions, [160](#)
- PAV2250A_SetViewSigVoltMV
 - View Units Functions, [161](#)
- PAV2250A_SetViewSigVoltMaxFieldWidth
 - View Max Field Width Functions, [131](#)
- PAV2250A_SetViewSigVoltOffset
 - View Offset Functions, [172](#)
- PAV2250A_SetViewSigVoltScale
 - View Scale Functions, [181](#)
- PAV2250A_SetViewSigVoltV
 - View Units Functions, [161](#)
- PAV2250A_SetViewTHDDB
 - View Units Functions, [161](#)
- PAV2250A_SetViewTHDMaxFieldWidth
 - View Max Field Width Functions, [133](#)
- PAV2250A_SetViewTHDPercent
 - View Units Functions, [162](#)
- PAV2250A_SetViewTotalRatioDB
 - View Units Functions, [162](#)
- PAV2250A_SetViewTotalRatioMaxFieldWidth
 - View Max Field Width Functions, [133](#)
- PAV2250A_SetViewTotalRatioPercent
 - View Units Functions, [162](#)
- PAV2250A_SetViewTotalRatioRatio
 - View Units Functions, [163](#)
- PAV2250A_ViewHarmonic
 - Harmonic Functions, [21](#)
- PAV2250A_ViewNextHarmonicGroup
 - Harmonic Functions, [22](#)
- PAV2250A_ViewPrevHarmonicGroup
 - Harmonic Functions, [22](#)
- PAV2250A_WriteCommand
 - Miscellaneous Functions, [89](#)
- PAV2250ADII.cpp
 - DIIIMain, [274](#)
 - PAV2250ADII_VERSION, [274](#)
- PAV2250ADII.h
 - _PAV2250AClass, [295](#)
 - _PAV2250AFUNC, [295](#)
 - bool, [295](#)
 - PAV_LVDT_SIGNAL, [296](#)
 - PAV_LVDT_TYPE, [296](#)
 - PAV_Ranges, [297](#)
 - PAV_ReadModes, [297](#)
 - PAV_STATUS, [298](#)
 - PAV_Tabs, [298](#)
 - PAV_Units, [298](#)
 - PAV_Views, [299](#)
 - PAV_VoltViews, [299](#)
- PAV2250ADII_VERSION
 - PAV2250ADII.cpp, [274](#)
- PAV_LVDT_SIGNAL
 - PAV2250ADII.h, [296](#)
- PAV_LVDT_TYPE
 - PAV2250ADII.h, [296](#)
- PAV_OPTIONS_RATIO
 - PAV2250ADII.h, [297](#)
- PAV_Ranges
 - PAV2250ADII.h, [297](#)
- PAV_ReadModes
 - PAV2250ADII.h, [297](#)

- PAV_STATUS
 - PAV2250ADII.h, 298
- PAV_Tabs
 - PAV2250ADII.h, 298
- PAV_Units
 - PAV2250ADII.h, 298
- PAV_Views
 - PAV2250ADII.h, 299
- PAV_VoltViews
 - PAV2250ADII.h, 299
- ParseForCommaSeparatedDataElements
 - Local Functions (not exported), 233
- Reference Only Functions, 249
 - PAV2250A_GetRefFundMag, 249
 - PAV2250A_GetRefFundMagRatio, 249
 - PAV2250A_GetRefInPhase, 250
 - PAV2250A_GetRefInPhaseRatio, 250
 - PAV2250A_GetRefPhase, 250
 - PAV2250A_GetRefQuad, 251
 - PAV2250A_GetRefQuadRatio, 251
 - PAV2250A_GetRefTHD, 251
 - PAV2250A_GetRefTotal, 252
 - PAV2250A_GetRefTotalRatio, 252
- Reference Range Functions, 23
 - PAV2250A_GetRefAutoRange, 24
 - PAV2250A_GetRefRangeActualIndex, 24
 - PAV2250A_GetRefRangeConfigIndex, 25
 - PAV2250A_GetRefRangeIndexSettings, 25
 - PAV2250A_GetRefRangeString, 27
 - PAV2250A_SetRefAutoRange, 28
 - PAV2250A_SetRefRange, 29
 - PAV2250A_SetRefRange100MV, 29
 - PAV2250A_SetRefRange100V, 30
 - PAV2250A_SetRefRange10V, 30
 - PAV2250A_SetRefRange1V, 30
 - PAV2250A_SetRefRange200MV, 31
 - PAV2250A_SetRefRange200V, 31
 - PAV2250A_SetRefRange20V, 31
 - PAV2250A_SetRefRange2V, 32
 - PAV2250A_SetRefRange500MV, 32
 - PAV2250A_SetRefRange500V, 32
 - PAV2250A_SetRefRange50MV, 33
 - PAV2250A_SetRefRange50V, 33
 - PAV2250A_SetRefRange5V, 33
- Reference Voltage Functions, 193
 - PAV2250A_GetRef, 193
 - PAV2250A_GetRefDC, 194
 - PAV2250A_GetRefText, 194
 - PAV2250A_GetRefTotalRMS_AC, 194
 - PAV2250A_GetRefTotalRMS_ACDC, 196
 - PAV2250A_GetRefTotalSum, 196
 - PAV2250A_SetRef, 196
 - PAV2250A_SetRefDC, 198
 - PAV2250A_SetRefTotalRMS_AC, 198
 - PAV2250A_SetRefTotalRMS_ACDC, 199
 - PAV2250A_SetRefTotalSum, 199
- SendIEEEMessage
 - Local Functions (not exported), 233
- Setup Options Functions, 93
 - PAV2250A_GetAutoSaveOption, 94
 - PAV2250A_GetAutoSaveOptionText, 95
 - PAV2250A_GetAutoUnitsOption, 95
 - PAV2250A_GetAutoUnitsOptionText, 96
 - PAV2250A_GetDateDisplayOption, 96
 - PAV2250A_GetDateDisplayOptionText, 97
 - PAV2250A_GetMainDisplayOption, 97
 - PAV2250A_GetMainDisplayOptionText, 98
 - PAV2250A_GetNullMeterRangePercent, 98
 - PAV2250A_GetSignalInputOption, 99
 - PAV2250A_GetSignalInputOptionText, 99
 - PAV2250A_GetTimeDisplayOption, 100
 - PAV2250A_GetTimeDisplayOptionText, 100
 - PAV2250A_GetTouchscreenOption, 101
 - PAV2250A_GetTouchscreenOptionText, 101
 - PAV2250A_SetAutoSaveDisable, 101
 - PAV2250A_SetAutoSaveEnable, 102
 - PAV2250A_SetAutoSaveOption, 102
 - PAV2250A_SetAutoUnitsDisable, 102
 - PAV2250A_SetAutoUnitsEnable, 103
 - PAV2250A_SetAutoUnitsOption, 103
 - PAV2250A_SetDateDisplayNumeric, 104
 - PAV2250A_SetDateDisplayOption, 104
 - PAV2250A_SetDateDisplayText, 104
 - PAV2250A_SetMainDisplayIndependent, 106
 - PAV2250A_SetMainDisplayLinked, 106
 - PAV2250A_SetMainDisplayOption, 106
 - PAV2250A_SetNullMeterRangePercent, 107
 - PAV2250A_SetSignalInputBack, 107
 - PAV2250A_SetSignalInputFront, 107
 - PAV2250A_SetSignalInputOption, 108
 - PAV2250A_SetTimeDisplayAMPM, 108
 - PAV2250A_SetTimeDisplayMilitary, 109
 - PAV2250A_SetTimeDisplayOption, 109
 - PAV2250A_SetTouchscreenDisable, 109
 - PAV2250A_SetTouchscreenEnable, 111
 - PAV2250A_SetTouchscreenOption, 111
- Signal Range Functions, 35
 - PAV2250A_GetSigAutoRange, 36
 - PAV2250A_GetSigRangeActualIndex, 36
 - PAV2250A_GetSigRangeConfigIndex, 37
 - PAV2250A_GetSigRangeIndexSettings, 37
 - PAV2250A_GetSigRangeString, 39
 - PAV2250A_SetSigAutoRange, 40
 - PAV2250A_SetSigRange, 40
 - PAV2250A_SetSigRange100MV, 41
 - PAV2250A_SetSigRange100V, 41
 - PAV2250A_SetSigRange10V, 41
 - PAV2250A_SetSigRange1V, 42
 - PAV2250A_SetSigRange200MV, 42
 - PAV2250A_SetSigRange200V, 42
 - PAV2250A_SetSigRange20V, 43
 - PAV2250A_SetSigRange2V, 43
 - PAV2250A_SetSigRange500MV, 43
 - PAV2250A_SetSigRange500V, 44
 - PAV2250A_SetSigRange50MV, 44

- PAV2250A_SetSigRange50V, 44
- PAV2250A_SetSigRange5V, 45
- Signal Voltage Functions, 200
 - PAV2250A_GetSig, 200
 - PAV2250A_GetSigDC, 201
 - PAV2250A_GetSigText, 201
 - PAV2250A_GetSigTotalRMS_AC, 201
 - PAV2250A_GetSigTotalRMS_ACDC, 203
 - PAV2250A_GetSigTotalSum, 203
 - PAV2250A_SetSig, 203
 - PAV2250A_SetSigDC, 205
 - PAV2250A_SetSigTotalRMS_AC, 205
 - PAV2250A_SetSigTotalRMS_ACDC, 206
 - PAV2250A_SetSigTotalSum, 206
- Time Window Functions, 90
 - PAV2250A_GetTimeWndActual, 90
 - PAV2250A_GetTimeWndAuto, 90
 - PAV2250A_GetTimeWndOverride, 91
 - PAV2250A_SetTimeWndAuto, 91
 - PAV2250A_SetTimeWndOverride, 91
- View by Index Functions, 182
 - PAV2250A_GetViewIndexMaxFieldWidth, 182
 - PAV2250A_GetViewIndexOffset, 183
 - PAV2250A_GetViewIndexScale, 184
 - PAV2250A_GetViewIndexUnits, 185
 - PAV2250A_GetViewIndexUnitsText, 186
 - PAV2250A_SetViewIndexMaxFieldWidth, 188
 - PAV2250A_SetViewIndexOffset, 189
 - PAV2250A_SetViewIndexScale, 190
 - PAV2250A_SetViewIndexUnits, 191
- View Configuration Functions, 112
 - PAV2250A_GetViewFrequencyConfig, 113
 - PAV2250A_GetViewFundMagConfig, 113
 - PAV2250A_GetViewInPhaseConfig, 115
 - PAV2250A_GetViewIndexConfig, 114
 - PAV2250A_GetViewMainConfig, 115
 - PAV2250A_GetViewPhaseConfig, 117
 - PAV2250A_GetViewQuadConfig, 117
 - PAV2250A_GetViewRefVoltConfig, 118
 - PAV2250A_GetViewSigOffsetConfig, 119
 - PAV2250A_GetViewSigVoltConfig, 119
 - PAV2250A_GetViewTHDConfig, 120
 - PAV2250A_GetViewTotalRatioConfig, 120
- View Max Field Width Functions, 122
 - PAV2250A_GetViewFrequencyMaxFieldWidth, 123
 - PAV2250A_GetViewFundMagMaxFieldWidth, 123
 - PAV2250A_GetViewInPhaseMaxFieldWidth, 123
 - PAV2250A_GetViewMainMaxFieldWidth, 124
 - PAV2250A_GetViewPhaseMaxFieldWidth, 124
 - PAV2250A_GetViewQuadMaxFieldWidth, 124
 - PAV2250A_GetViewRefVoltMaxFieldWidth, 125
 - PAV2250A_GetViewSigOffsetMaxFieldWidth, 125
 - PAV2250A_GetViewSigVoltMaxFieldWidth, 126
 - PAV2250A_GetViewTHDMaxFieldWidth, 126
 - PAV2250A_GetViewTotalRatioMaxFieldWidth, 126
 - PAV2250A_GetViewFrequencyMaxFieldWidth, 127
 - PAV2250A_GetViewFundMagMaxFieldWidth, 127
 - PAV2250A_GetViewInPhaseMaxFieldWidth, 127
 - PAV2250A_GetViewMainMaxFieldWidth, 129
 - PAV2250A_GetViewPhaseMaxFieldWidth, 129
 - PAV2250A_GetViewQuadMaxFieldWidth, 129
 - PAV2250A_GetViewRefVoltMaxFieldWidth, 131
 - PAV2250A_GetViewSigOffsetMaxFieldWidth, 131
 - PAV2250A_GetViewSigVoltMaxFieldWidth, 131
 - PAV2250A_GetViewTHDMaxFieldWidth, 133
 - PAV2250A_GetViewTotalRatioMaxFieldWidth, 133
- View Offset Functions, 164
 - PAV2250A_GetViewFundMagOffset, 164
 - PAV2250A_GetViewInPhaseOffset, 165
 - PAV2250A_GetViewMainOffset, 165
 - PAV2250A_GetViewPhaseOffset, 165
 - PAV2250A_GetViewQuadOffset, 166
 - PAV2250A_GetViewRefVoltOffset, 166
 - PAV2250A_GetViewSigVoltOffset, 166
 - PAV2250A_SetViewFundMagOffset, 168
 - PAV2250A_SetViewInPhaseOffset, 168
 - PAV2250A_SetViewMainOffset, 168
 - PAV2250A_SetViewPhaseOffset, 170
 - PAV2250A_SetViewQuadOffset, 170
 - PAV2250A_SetViewRefVoltOffset, 170
 - PAV2250A_SetViewSigVoltOffset, 172
- View Scale Functions, 173
 - PAV2250A_GetViewFundMagScale, 173
 - PAV2250A_GetViewInPhaseScale, 174
 - PAV2250A_GetViewMainScale, 174
 - PAV2250A_GetViewPhaseScale, 174
 - PAV2250A_GetViewQuadScale, 175
 - PAV2250A_GetViewRefVoltScale, 175
 - PAV2250A_GetViewSigVoltScale, 175
 - PAV2250A_SetViewFundMagScale, 177
 - PAV2250A_SetViewInPhaseScale, 177
 - PAV2250A_SetViewMainScale, 177
 - PAV2250A_SetViewPhaseScale, 179
 - PAV2250A_SetViewQuadScale, 179
 - PAV2250A_SetViewRefVoltScale, 179
 - PAV2250A_SetViewSigVoltScale, 181
- View Units Functions, 134
 - PAV2250A_GetViewFrequencyUnits, 136
 - PAV2250A_GetViewFrequencyUnitsText, 137
 - PAV2250A_GetViewFundMagUnits, 137
 - PAV2250A_GetViewFundMagUnitsText, 137
 - PAV2250A_GetViewInPhaseUnits, 139
 - PAV2250A_GetViewInPhaseUnitsText, 139
 - PAV2250A_GetViewMainUnits, 140
 - PAV2250A_GetViewMainUnitsText, 140
 - PAV2250A_GetViewPhaseUnits, 142
 - PAV2250A_GetViewPhaseUnitsText, 142
 - PAV2250A_GetViewQuadUnits, 143
 - PAV2250A_GetViewQuadUnitsText, 143
 - PAV2250A_GetViewRefVoltUnits, 144
 - PAV2250A_GetViewRefVoltUnitsText, 144
 - PAV2250A_GetViewSigOffsetUnits, 145

PAV2250A_GetViewSigOffsetUnitsText, [145](#)
PAV2250A_GetViewSigVoltUnits, [146](#)
PAV2250A_GetViewSigVoltUnitsText, [146](#)
PAV2250A_GetViewTHDUnits, [147](#)
PAV2250A_GetViewTHDUnitsText, [147](#)
PAV2250A_GetViewTotalRatioUnits, [148](#)
PAV2250A_GetViewTotalRatioUnitsText, [148](#)
PAV2250A_SetViewFrequencyHZ, [149](#)
PAV2250A_SetViewFrequencyKHZ, [149](#)
PAV2250A_SetViewFundMagDB, [149](#)
PAV2250A_SetViewFundMagMV, [150](#)
PAV2250A_SetViewFundMagPercent, [150](#)
PAV2250A_SetViewFundMagRatio, [150](#)
PAV2250A_SetViewFundMagV, [151](#)
PAV2250A_SetViewInPhaseDB, [151](#)
PAV2250A_SetViewInPhaseMV, [151](#)
PAV2250A_SetViewInPhasePercent, [152](#)
PAV2250A_SetViewInPhaseRatio, [152](#)
PAV2250A_SetViewInPhaseV, [152](#)
PAV2250A_SetViewMain180, [153](#)
PAV2250A_SetViewMain360, [153](#)
PAV2250A_SetViewMainDB, [153](#)
PAV2250A_SetViewMainMV, [155](#)
PAV2250A_SetViewMainPercent, [155](#)
PAV2250A_SetViewMainRatio, [155](#)
PAV2250A_SetViewMainV, [157](#)
PAV2250A_SetViewPhase180, [157](#)
PAV2250A_SetViewPhase360, [157](#)
PAV2250A_SetViewQuadDB, [158](#)
PAV2250A_SetViewQuadMV, [158](#)
PAV2250A_SetViewQuadPercent, [158](#)
PAV2250A_SetViewQuadRatio, [159](#)
PAV2250A_SetViewQuadV, [159](#)
PAV2250A_SetViewRefVoltMV, [159](#)
PAV2250A_SetViewRefVoltV, [160](#)
PAV2250A_SetViewSigOffsetMV, [160](#)
PAV2250A_SetViewSigOffsetV, [160](#)
PAV2250A_SetViewSigVoltMV, [161](#)
PAV2250A_SetViewSigVoltV, [161](#)
PAV2250A_SetViewTHDDB, [161](#)
PAV2250A_SetViewTHDPercent, [162](#)
PAV2250A_SetViewTotalRatioDB, [162](#)
PAV2250A_SetViewTotalRatioPercent, [162](#)
PAV2250A_SetViewTotalRatioRatio, [163](#)

WaitForResponse

Local Functions (not exported), [234](#)