

Axel-hub

2.3.0.17

by Teodor Krastev for Imperial College London

1 Namespace Index	1
1.1 Packages	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 File Index	7
4.1 File List	7
5 Namespace Documentation	9
5.1 Axel_hub Namespace Reference	9
5.1.1 Function Documentation	12
5.1.1.1 ActiveRemote()	12
5.1.1.2 backMME()	12
5.1.1.3 calcContrast()	13
5.1.1.4 centreFringe()	13
5.1.1.5 deconstructAccel()	14
5.1.1.6 dispatcherTimer_Tick()	15
5.1.1.7 fillReport()	15
5.1.1.8 Flip()	15
5.1.1.9 GetBufferSize()	16
5.1.1.10 GetSamplingPeriod()	16
5.1.1.11 Image_MouseDown()	16
5.1.1.12 Init()	16
5.1.1.13 InitOptions()	17
5.1.1.14 LogEvent()	17
5.1.1.15 LogHandler()	17
5.1.1.16 nextShot()	18
5.1.1.17 OnActiveComm()	18
5.1.1.18 OnAsyncSend()	18
5.1.1.19 OnJumboRepeat()	19
5.1.1.20 OnRealSampling()	19
5.1.1.21 OnReceive()	19
5.1.1.22 OpenConfigFile()	20
5.1.1.23 PID()	20
5.1.1.24 RemoteEvent()	20
5.1.1.25 RemoteHandler()	20
5.1.1.26 RemoteModeEvent()	21
5.1.1.27 Reset()	21
5.1.1.28 SaveConfigFile()	21
5.1.1.29 scanClass()	21

5.1.1.30 SendJson()	21
5.1.1.31 SetActivity()	22
5.1.1.32 SetFringeParams()	22
5.1.1.33 SetSamplingRate()	22
5.1.1.34 StartDelegate()	23
5.1.1.35 StartEvent()	23
5.1.1.36 StartHandler()	23
5.1.1.37 Status()	23
5.1.1.38 strobesUC()	23
5.1.1.39 UpdateModes()	24
5.1.1.40 UserControl_Loaded()	24
5.1.1.41 zeroFringe()	24
5.1.2 Variable Documentation	24
5.1.2.1 _jumboButton	24
5.1.2.2 _PID_Enabled	25
5.1.2.3 _Running	25
5.1.2.4 ArrangedPartner	25
5.1.2.5 configFile	25
5.1.2.6 currentTime	25
5.1.2.7 Down	25
5.1.2.8 dStack	26
5.1.2.9 dTimer	26
5.1.2.10 genOptions	26
5.1.2.11 grpMME	26
5.1.2.12 iStack	26
5.1.2.13 jumboButton	26
5.1.2.14 lastContrast	27
5.1.2.15 lastMMEin	27
5.1.2.16 lastMMEout	27
5.1.2.17 logger	27
5.1.2.18 Low	27
5.1.2.19 OnActiveRemote	27
5.1.2.20 OnLog	28
5.1.2.21 OnRemote	28
5.1.2.22 OnRemoteMode	28
5.1.2.23 OnStart	28
5.1.2.24 PID_Enabled	28
5.1.2.25 realSampling	29
5.1.2.26 remote	29
5.1.2.27 remoteMode	29
5.1.2.28 runI	29
5.1.2.29 Running	29

5.1.2.30 scanModes	30
5.1.2.31 set	30
5.1.2.32 Titles	30
5.1.2.33 totalTime	30
5.1.2.34 Up	30
5.2 Axel_hub::Properties Namespace Reference	30
5.3 OptionsNS Namespace Reference	31
5.3.1 Enumeration Type Documentation	31
5.3.1.1 RemoteMode	31
5.4 XamlGeneratedNamespace Namespace Reference	31
6 Class Documentation	33
6.1 Axel_hub.accelCalibr Struct Reference	33
6.1.1 Detailed Description	33
6.1.2 Member Function Documentation	33
6.1.2.1 accel()	33
6.1.3 Member Data Documentation	34
6.1.3.1 cK0	34
6.1.3.2 cK1	34
6.1.3.3 model	34
6.1.3.4 pK0	34
6.1.3.5 pK1	35
6.1.3.6 rAccel	35
6.1.3.7 rTemper	35
6.1.3.8 SN	35
6.2 Axel_hub::App Class Reference	35
6.2.1 Detailed Description	36
6.3 Axel_hub.AxelAxesClass Class Reference	36
6.3.1 Detailed Description	37
6.3.2 Constructor & Destructor Documentation	37
6.3.2.1 AxelAxesClass()	37
6.3.3 Member Function Documentation	38
6.3.3.1 AddAxis()	38
6.3.3.2 byName()	38
6.3.3.3 Clear()	39
6.3.3.4 Closing()	39
6.3.3.5 DoAcquire()	39
6.3.3.6 DoAcquireTemperature()	40
6.3.3.7 DoJumboScan()	40
6.3.3.8 DoRemote()	40
6.3.3.9 jumboRepeat()	41
6.3.3.10 LogEvent()	41

6.3.3.11 LogHandler()	41
6.3.3.12 prfIdx()	41
6.3.3.13 SaveDefaultModes()	42
6.3.3.14 set2startADC24()	42
6.3.3.15 SetChartStrobes()	42
6.3.3.16 startADC()	43
6.3.3.17 UpdateFromOptions()	43
6.3.4 Member Data Documentation	43
6.3.4.1 axelMems	43
6.3.5 Property Documentation	43
6.3.5.1 memsRunning	44
6.3.5.2 rCount	44
6.3.6 Event Documentation	44
6.3.6.1 OnLog	44
6.4 Axel_hub::AxelAxisClass Class Reference	44
6.4.1 Detailed Description	45
6.5 Axel_hub::AxelChart Class Reference	45
6.5.1 Detailed Description	45
6.6 Axel_hub::AxelChartClass Class Reference	45
6.6.1 Detailed Description	46
6.7 Axel_hub.AxelMems Class Reference	46
6.7.1 Detailed Description	47
6.7.2 Member Enumeration Documentation	47
6.7.2.1 TimingModes	47
6.7.3 Constructor & Destructor Documentation	48
6.7.3.1 AxelMems()	48
6.7.4 Member Function Documentation	48
6.7.4.1 AcquireEvent()	48
6.7.4.2 AcquireHandler()	48
6.7.4.3 configureVITask()	49
6.7.4.4 isDevicePlugged()	49
6.7.4.5 readBurst()	49
6.7.4.6 RealConvRate()	49
6.7.4.7 RealSamplingEvent()	50
6.7.4.8 RealSamplingHandler()	50
6.7.4.9 Reset()	50
6.7.4.10 SetStopwatch()	50
6.7.4.11 StartAcquisition()	50
6.7.4.12 StartStopwatch()	51
6.7.4.13 StopAcquisition()	51
6.7.4.14 TimeElapsed()	51
6.7.5 Member Data Documentation	51

6.7.5.1 AdjustTimelineToStopwatch	51
6.7.5.2 FixConvRate	52
6.7.5.3 hw	52
6.7.5.4 memsX	52
6.7.5.5 rawData	52
6.7.5.6 Timeout	52
6.7.5.7 TimingMode	52
6.7.6 Property Documentation	53
6.7.6.1 activeChannel	53
6.7.6.2 nSamples	53
6.7.6.3 running	53
6.7.6.4 sampleRate	53
6.7.7 Event Documentation	53
6.7.7.1 OnAcquire	53
6.7.7.2 OnRealSampling	54
6.8 Axel_hub.AxelMemsTemperature Class Reference	54
6.8.1 Detailed Description	54
6.8.2 Constructor & Destructor Documentation	54
6.8.2.1 AxelMemsTemperature()	54
6.8.3 Member Function Documentation	54
6.8.3.1 TakeTheTemperature()	55
6.8.4 Member Data Documentation	55
6.8.4.1 hw	55
6.9 Axel_hub.DataStack Class Reference	55
6.9.1 Detailed Description	57
6.9.2 Constructor & Destructor Documentation	57
6.9.2.1 DataStack()	57
6.9.3 Member Function Documentation	58
6.9.3.1 Add()	58
6.9.3.2 AddPoint()	58
6.9.3.3 AddRange()	59
6.9.3.4 Clear()	59
6.9.3.5 Clone()	59
6.9.3.6 Compress()	60
6.9.3.7 CopyEach()	60
6.9.3.8 ExportToArray()	60
6.9.3.9 fillSamples()	61
6.9.3.10 Fit2Limit()	61
6.9.3.11 ImportFromArray()	61
6.9.3.12 importFromArrays()	62
6.9.3.13 indexByX()	62
6.9.3.14 OpenPair()	62

6.9.3.15 pointSDev()	63
6.9.3.16 pointXs()	63
6.9.3.17 pointYs()	64
6.9.3.18 Portion()	64
6.9.3.19 RefreshEvent()	64
6.9.3.20 RefreshHandler()	64
6.9.3.21 Rescale()	65
6.9.3.22 SavePair()	66
6.9.3.23 statsByIdx()	66
6.9.3.24 statsByTime()	67
6.9.3.25 TimePortion()	67
6.9.4 Member Data Documentation	68
6.9.4.1 generalIdx	68
6.9.4.2 logger	68
6.9.4.3 maxDepth	68
6.9.4.4 RefFileStats	68
6.9.4.5 stopWatch	68
6.9.4.6 visualCountLimit	69
6.9.5 Property Documentation	69
6.9.5.1 Depth	69
6.9.5.2 First	69
6.9.5.3 Last	69
6.9.5.4 lastError	69
6.9.5.5 prefix	70
6.9.5.6 rem	70
6.9.5.7 Running	70
6.9.5.8 StackMode	70
6.9.5.9 TimeSeriesMode	70
6.9.6 Event Documentation	70
6.9.6.1 OnRefresh	71
6.10 Axel_hub::FringeParams Struct Reference	71
6.10.1 Detailed Description	71
6.10.2 Member Data Documentation	71
6.10.2.1 offset	71
6.10.2.2 period	72
6.10.2.3 phase	72
6.11 OptionsNS.GeneralOptions Class Reference	72
6.11.1 Detailed Description	73
6.11.2 Member Enumeration Documentation	73
6.11.2.1 SaveModes	73
6.11.3 Member Function Documentation	73
6.11.3.1 Save()	74

6.11.4 Member Data Documentation	74
6.11.4.1 saveModes	74
6.11.5 Property Documentation	74
6.11.5.1 AxesChannels	74
6.11.5.2 followPID	74
6.11.5.3 intN2	74
6.11.5.4 JumboRepeat	75
6.11.5.5 JumboScan	75
6.11.5.6 LogFilePrec	75
6.11.5.7 Mems2SignDelay	75
6.11.5.8 Mems2SignLen	75
6.11.5.9 MemsHw	75
6.11.5.10 MemsInJumbo	76
6.11.5.11 RawSignalAvg	76
6.11.5.12 SaveFilePrec	76
6.11.5.13 saveVisuals	76
6.11.5.14 ShowMemsIfRunning	76
6.11.5.15 SignalCursorPrec	76
6.11.5.16 SignalTablePrec	77
6.11.5.17 TemperatureCompensation	77
6.11.5.18 TemperatureEnabled	77
6.11.5.19 TemperatureHw	77
6.11.5.20 TrendSignalLen	77
6.12 XamlGeneratedNamespace::GeneratedInternalTypeHelper Class Reference	78
6.12.1 Detailed Description	78
6.13 Axel_hub::JoinOptimClass Class Reference	78
6.13.1 Detailed Description	78
6.14 Axel_hub::MainWindow Class Reference	79
6.14.1 Detailed Description	79
6.15 OptionsNS.Modes Class Reference	79
6.15.1 Detailed Description	80
6.15.2 Member Function Documentation	80
6.15.2.1 Save()	81
6.15.3 Property Documentation	81
6.15.3.1 AutoScaleBottom	81
6.15.3.2 AutoScaleMiddle	81
6.15.3.3 Background	81
6.15.3.4 ChartUpdate	81
6.15.3.5 DarkCurrent	81
6.15.3.6 DoubleStrobe	82
6.15.3.7 JoinLog	82
6.15.3.8 JumboBy	82

6.15.3.9 JumboCycles	82
6.15.3.10 JumboFrom	82
6.15.3.11 JumboTo	82
6.15.3.12 Kcoeff	83
6.15.3.13 kD	83
6.15.3.14 kl	83
6.15.3.15 kP	83
6.15.3.16 MemsEnabled	83
6.15.3.17 MiddleFrame	83
6.15.3.18 N1	84
6.15.3.19 N2	84
6.15.3.20 Ntot	84
6.15.3.21 offset	84
6.15.3.22 phi0	84
6.15.3.23 PID_Enabled	84
6.15.3.24 PowerCoeff	85
6.15.3.25 RN1	85
6.15.3.26 RN2	85
6.15.3.27 RollMean	85
6.15.3.28 RsltChrtUpdate	85
6.15.3.29 RsltTblUpdate	85
6.15.3.30 scale	86
6.15.3.31 ShowFreq	86
6.15.3.32 SignalLog	86
6.15.3.33 StackDepth	86
6.15.3.34 StdDev	86
6.15.3.35 TblUpdate	86
6.15.3.36 TopFrame	87
6.15.3.37 TopOfTopFrame	87
6.16 OptionsNS::OptionsWindow Class Reference	87
6.16.1 Detailed Description	87
6.17 Axel_hub::Properties::Resources Class Reference	87
6.17.1 Detailed Description	88
6.18 Axel_hub::scanClass Class Reference	88
6.18.1 Detailed Description	88
6.19 OptionsNS.ScanModes Class Reference	88
6.19.1 Detailed Description	89
6.19.2 Member Function Documentation	89
6.19.2.1 Save()	89
6.19.3 Member Data Documentation	89
6.19.3.1 remoteMode	89
6.19.4 Property Documentation	89

6.19.4.1 Height	90
6.19.4.2 Left	90
6.19.4.3 SamplingFreq	90
6.19.4.4 SizeLimit	90
6.19.4.5 SizeLimitMode	90
6.19.4.6 TimeLimit	90
6.19.4.7 TimeLimitMode	91
6.19.4.8 Top	91
6.19.4.9 Width	91
6.20 Axel_hub::Properties::Settings Class Reference	91
6.20.1 Detailed Description	91
6.21 Axel_hub.ShotList Class Reference	92
6.21.1 Detailed Description	92
6.21.2 Constructor & Destructor Documentation	93
6.21.2.1 ShotList()	93
6.21.3 Member Function Documentation	93
6.21.3.1 Add()	93
6.21.3.2 archiScan()	93
6.21.3.3 resetScan()	94
6.21.3.4 Save()	94
6.21.4 Member Data Documentation	94
6.21.4.1 conditions	94
6.21.4.2 depth	95
6.21.5 Property Documentation	95
6.21.5.1 archiveMode	95
6.21.5.2 enabled	95
6.21.5.3 FileCount	95
6.21.5.4 filename	95
6.21.5.5 lastIdx	96
6.21.5.6 savingMode	96
6.22 Axel_hub::signalClass Class Reference	96
6.22.1 Detailed Description	96
6.23 Axel_hub.SingleShot Class Reference	96
6.23.1 Detailed Description	97
6.23.2 Constructor & Destructor Documentation	97
6.23.2.1 SingleShot() [1/4]	97
6.23.2.2 SingleShot() [2/4]	98
6.23.2.3 SingleShot() [3/4]	98
6.23.2.4 SingleShot() [4/4]	98
6.23.3 Member Function Documentation	98
6.23.3.1 deconstructAccel()	98
6.23.3.2 idxByTime()	99

6.23.3.3 IsEmpty()	99
6.23.3.4 memsPortion()	99
6.23.3.5 memsWeightAccel()	100
6.23.4 Member Data Documentation	100
6.23.4.1 precision	100
6.23.4.2 quant	100
6.23.5 Property Documentation	100
6.23.5.1 AsString	101
6.23.5.2 mems	101
6.24 Axel_hub::strokesUC Class Reference	101
6.24.1 Detailed Description	101
7 File Documentation	103
7.1 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/App.xaml.cs File Reference	103
7.2 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AssemblyInfo.cs File Reference	103
7.3 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelAxes.cs File Reference	103
7.4 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelAxisUC.xaml.cs File Reference	104
7.5 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelChartUC.xaml.cs File Reference	104
7.6 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelHMemcs File Reference	104
7.7 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/DataPrimitives.cs File Reference	105
7.8 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/DataStackLib.cs File Reference	105
7.9 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/MainWindow.xaml.cs File Reference	105
7.10 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/App.g.cs File Reference	106
7.11 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/App.g.i.cs File Reference	106
7.12 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Axel-hub_Content.g.i.cs File Reference	106
7.13 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelAxisUC.g.cs File Reference	106
7.14 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelAxisUC.g.i.cs File Reference	107
7.15 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChart.g.i.cs File Reference	107
7.16 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChartUC.g.cs File Reference	107
7.17 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChartUC.g.i.cs File Reference	108
7.18 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/GeneratedInternalTypeHelper.g.cs File Reference	108
7.19 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/GeneratedInternalTypeHelper.g.i.cs File Reference	108
7.20 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/JoinOptimUC.g.i.cs File Reference	108
7.21 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/MainWindow.g.cs File Reference	109
7.22 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/MainWindow.g.i.cs File Reference	109
7.23 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options/Options.g.cs File Reference	109
7.24 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options/Options.g.i.cs File Reference	110
7.25 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options.g.i.cs File Reference	110
7.26 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/scanUC.g.cs File Reference	110
7.27 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/scanUC.g.i.cs File Reference	111
7.28 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/signalUC.g.cs File Reference	111

7.29 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/signalUC.g.i.cs File Reference	111
7.30 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strobeControlUC.g.i.cs File Reference	112
7.31 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strobesUC.g.cs File Reference	112
7.32 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strobesUC.g.i.cs File Reference	112
7.33 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/Options.xaml.cs File Reference	113
7.34 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/OptionsType.cs File Reference	113
7.35 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Properties/Resources.Designer.cs File Reference	113
7.36 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Properties/Settings.Designer.cs File Reference	114
7.37 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/scanUC.xaml.cs File Reference	114
7.38 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/signalUC.xaml.cs File Reference	116
7.39 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/strobesUC.xaml.cs File Reference	116

Index	119
--------------	------------

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

Axel_hub	9
Axel_hub::Properties	30
OptionsNS	31
XamlGeneratedNamespace	31

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Axel_hub.accelCalibr	33
Application	
Axel_hub::App	35
Axel_hub::App	35
Axel_hub::App	35
ApplicationSettingsBase	
Axel_hub::Properties::Settings	91
Axel_hub.AxelMems	46
Axel_hub.AxelMemsTemperature	54
Axel_hub::FringeParams	71
OptionsNS.GeneralOptions	72
IComponentConnector	
Axel_hub::AxelAxisClass	44
Axel_hub::AxelAxisClass	44
Axel_hub::AxelChart	45
Axel_hub::AxelChartClass	45
Axel_hub::AxelChartClass	45
Axel_hub::JoinOptimClass	78
Axel_hub::MainWindow	79
Axel_hub::MainWindow	79
Axel_hub::scanClass	88
Axel_hub::scanClass	88
Axel_hub::signalClass	96
Axel_hub::signalClass	96
Axel_hub::strobesUC	101
Axel_hub::strobesUC	101
Axel_hub::strobesUC	101
OptionsNS::OptionsWindow	87
OptionsNS::OptionsWindow	87
OptionsNS::OptionsWindow	87
InternalTypeHelper	
XamlGeneratedNamespace::GeneratedInternalTypeHelper	78
List	
Axel_hub.AxelAxesClass	36
Axel_hub.DataStack	55

Axel_hub.ShotList	92
OptionsNS.Modes	79
Axel_hub::Properties::Resources	87
OptionsNS.ScanModes	88
Axel_hub.SingleShot	96
UserControl	
Axel_hub::AxelAxisClass	44
Axel_hub::AxelAxisClass	44
Axel_hub::AxelChart	45
Axel_hub::AxelChartClass	45
Axel_hub::AxelChartClass	45
Axel_hub::JoinOptimClass	78
Axel_hub::scanClass	88
Axel_hub::scanClass	88
Axel_hub::signalClass	96
Axel_hub::signalClass	96
Axel_hub::strobesUC	101
Axel_hub::strobesUC	101
Axel_hub::strobesUC	101
UserControl	
Axel_hub::AxelAxisClass	44
Axel_hub::AxelChartClass	45
Axel_hub::signalClass	96
Window	
Axel_hub::MainWindow	79
Axel_hub::MainWindow	79
Axel_hub::MainWindow	79
OptionsNS::OptionsWindow	87
OptionsNS::OptionsWindow	87
OptionsNS::OptionsWindow	87
OptionsNS::OptionsWindow	87

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Axel_hub.accelCalibr	Acceleration calibration with optional temperature compensation particular to each MEMS device	33
Axel_hub::App	Interaction logic for App.xaml	35
Axel_hub.AxelAxesClass	Intermediator between incoming data flow from ucScan user component and AxelAxis user components	36
Axel_hub::AxelAxisClass	Interaction logic for AxelAxisUC.xaml AxelAxisClass represents a single axis of acceleration encapsulated and accessible in AxelAxes list of AxelAxisClass Future intermediary abstract movement (linear or rotation) component will be implemented	44
Axel_hub::AxelChart	AxelChart	45
Axel_hub::AxelChartClass	Interaction logic for AxelChart.xaml	45
Axel_hub.AxelMems	The hardware abstraction for MEMS with ADC24 (NI9251) device	46
Axel_hub.AxelMemsTemperature	The temperature in a class abstraction	54
Axel_hub.DataStack	You (developer) need to set TimeMode and one of SizeLimit or TimeLimit TimeMode is about the way DataStack limits its size The output is from standard List method ToArray in order to set DataSource of Graph	55
Axel_hub::FringeParams	$\text{fringes}(\phi) = \cos(\text{period} * \phi + \text{phase}) + \text{offset}$	71
OptionsNS.GeneralOptions	general options from Options dialog window accessible everywhere	72
XamlGeneratedNamespace::GeneratedInternalTypeHelper	GeneratedInternalTypeHelper	78
Axel_hub::JoinOptimClass	JoinOptimClass	78
Axel_hub::MainWindow	Interaction logic for MainWindow.xaml command line arguments (space separated): - remote:partner -hw:config.file where partner is remote partner name title; hw<c>hardware, config.file.hw is in Config folder	79

OptionsNS.Modes	
Visuals and prameters for Top: Axel-chart Middle: Signal panel charts Bottom: Scan and Accel trend tabs/charts	79
OptionsNS::OptionsWindow	
OptionsWindow	87
Axel_hub::Properties::Resources	
A strongly-typed resource class, for looking up localized strings, etc	87
Axel_hub::scanClass	
scanClass	88
OptionsNS.ScanModes	
visuals for the app, MEMS acquisition params and scan modes	88
Axel_hub::Properties::Settings	91
Axel_hub.ShotList	
List / series of single shots	92
Axel_hub::signalClass	
signalClass	96
Axel_hub.SingleShot	
Class representing single shot with both components quant (MOT) and MEMS (ADC24)	96
Axel_hub::strobesUC	
strobesUC	101

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/App.xaml.cs	103
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AssemblyInfo.cs	103
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelAxes.cs	103
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelAxisUC.xaml.cs	104
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelChartUC.xaml.cs	104
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelHMMems.cs	104
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/DataPrimitives.cs	105
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/DataStackLib.cs	105
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/MainWindow.xaml.cs	105
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/scanUC.xaml.cs	114
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/signalUC.xaml.cs	116
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/strobesUC.xaml.cs	116
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/App.g.cs	106
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/App.g.i.cs	106
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Axel-hub_Content.g.i.cs	106
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelAxisUC.g.cs	106
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelAxisUC.g.i.cs	107
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChart.g.i.cs	107
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChartUC.g.cs	107
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChartUC.g.i.cs	108
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/GeneratedInternalTypeHelper.g.cs	108
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/GeneratedInternalTypeHelper.g.i.cs	108
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/JoinOptimUC.g.i.cs	108
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/MainWindow.g.cs	109
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/MainWindow.g.i.cs	109
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options.g.i.cs	110
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/scanUC.g.cs	110
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/scanUC.g.i.cs	111
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/signalUC.g.cs	111
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/signalUC.g.i.cs	111
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strobeControlUC.g.i.cs	112
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strobesUC.g.cs	112
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strobesUC.g.i.cs	112
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options/Options.g.cs	109
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options/Options.g.i.cs	110

E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/ Options.xaml.cs	113
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/ OptionsType.cs	113
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Properties/ Resources.Designer.cs	113
E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Properties/ Settings.Designer.cs	114

Chapter 5

Namespace Documentation

5.1 Axel_hub Namespace Reference

Namespaces

- [Properties](#)

Classes

- struct [accelCalibr](#)
Acceleration calibration with optional temperature compensation particular to each MEMS device
- class [App](#)
Interaction logic for App.xaml
- class [AxelAxesClass](#)
Intermediator between incoming data flow from ucScan user component and AxelAxis user components
- class [AxelAxisClass](#)
Interaction logic for AxelAxisUC.xaml [AxelAxisClass](#) represents a single axis of acceleration encapsulated and accessible in AxelAxes list of [AxelAxisClass](#) Future intermediary abstract movement (linear or rotation) component will be implemented.
- class [AxelChart](#)
[AxelChart](#)
- class [AxelChartClass](#)
Interaction logic for AxelChart.xaml
- class [AxelMems](#)
The hardware abstraction for MEMS with ADC24 (NI9251) device
- class [AxelMemsTemperature](#)
The temperature in a class abstraction
- class **calcAccel**
Library for calculating acceleration from fringes, phase, etc
- class [DataStack](#)
You (developer) need to set TimeMode and one of SizeLimit or TimeLimit TimeMode is about the way [DataStack](#) limits its size The output is from standart List method ToArray in order to set DataSource of Graph
- struct [FringeParams](#)
 *$fringes(phi) = \cos(period * phi + phase) + offset$*
- class [JoinOptimClass](#)

- [JoinOptimClass](#)
- class [MainWindow](#)

Interaction logic for MainWindow.xaml command line arguments (space separated): -remote:partner -hw↵:config.file where partner is remote partner name title; hw<c>hardware, config.file.hw is in Config folder
- class **MMDataConverter**

Averaging the photo diode signals {"N2", "NTot", "B2", "BTot", "Bg"}
- class [scanClass](#)

scanClass
- class [ShotList](#)

List / series of single shots
- class [signalClass](#)

signalClass
- class [SingleShot](#)

Class representing single shot with both components quant (MOT) and MEMS (ADC24)
- class [strobesUC](#)

strobesUC

Functions

- public delegate void [StartDelegate](#) ()
- public [scanClass](#) ()

Class constructor - set defaults
- public void [InitOptions](#) (ref GeneralOptions _genOptions, ref ScanModes _scanModes)

Initialize - set genOptions
- public void [UpdateModes](#) ()

Set internal from visual modes
- public bool [SendJson](#) (string json, bool async=false)

Wrapper of remote.sendCommand
- public void [SetActivity](#) (string act)
- public void [SetSamplingRate](#) (int rate)
- public void [SetFringeParams](#) ([FringeParams](#) fp)

Show fringes params
- public void [OnRealSampling](#) (double _realSampling)
- private void [dispatcherTimer_Tick](#) (object sender, EventArgs e)

Shows visual progress of ADC24 acquisition
- private void [Status](#) (string sts)
- protected void [RemoteModeEvent](#) (RemoteMode oldMode, RemoteMode newMode)
- private bool [OnReceive](#) (string message)

Incomming from MM2/Axel-probe message
- public delegate void [StartHandler](#) (bool jumbo, bool down, double period, int sizeLimit)

Start/Stop group operation wity ADC24 params
- protected void [StartEvent](#) (bool jumbo, bool down, double period, int sizeLimit)
- public delegate void [RemoteHandler](#) (string msg)

Incomming message event thingy
- protected void [RemoteEvent](#) (string msg)
- public delegate void [LogHandler](#) (string txt, Color? clr=null)

Log into left text box
- protected void [LogEvent](#) (string txt, Color? clr=null)
- protected void [OnAsyncSend](#) (bool OK, string json2send)

Report sent message in log

- private void [Image_MouseDown](#) (object sender, MouseButtonEventArgs e)
- public double [GetSamplingPeriod](#) ()
Get the sampling period regardless the units
- public int [GetBufferSize](#) ()
Get the buffer size depending of settings
- protected void [ActiveRemote](#) (bool active)
- private void [OnActiveComm](#) (bool active, bool forced)
Event when the connection goes ON/OFF
- private void [UserControl_Loaded](#) (object sender, RoutedEventArgs e)
Some secondary to constructor initializations
- public [strokesUC](#) ()
Class constructor
- public void [Reset](#) ()
Initiale stroke for axel-probe simulated fringes
- public void [Flip](#) ()
Exchange UP/DOWN stroke positions
- public void [Init](#) (string _prefix)
Initiate stroke from file settings
- public void [OnJumboRepeat](#) (double _fringeScale, double _fringeShift, MMexec _grpMME, double contrastV)
Call this before each Jumbo Repeat for group MMexec and modes synchronization
- public double [centreFringe](#) ()
Calculating fringe centre
- public double [calcContrast](#) (double A)
Calculating contrast
- public double [zeroFringe](#) ()
Calculating zeroFringe - similar to centreFring but with phase shift compensation
- public Dictionary< string, double > [deconstructAccel](#) (double accel, double mems)
Deconstructing acceleration to acceleration components - see dictionary keys
- public double [nextShot](#) (int runID, double asymmetry, out double correction)
Calculated phaseCorr - corrected Raman phase (0 if not PID)
- public MMexec [backMME](#) (int runID, double asymmetry, MMexec mme=null)
Prepare back message with new Raman phase value
- private void [fillReport](#) (Dictionary< string, double > rpr)
Update table with strokes/PID calculation results
- public double [PID](#) (double disbalance)
Calculating the phase correction from the disbalance on strokes Ys
- public void [SaveConfigFile](#) ()
Save Config file in Config directory of Axel-hub
- public void [OpenConfigFile](#) ()
Open Config file from Config directory of Axel-hub

Variables

- public struct [Axel_hub::FringeParams](#) [realSampling](#)
Interaction logic for UserControl1.xaml
- private string [ArrangedPartner](#) = ""
- TimeSpan [totalTime](#)
- TimeSpan [currentTime](#)
- public DispatcherTimer [dTimer](#)
- GeneralOptions [genOptions](#) = null

- public ScanModes [scanModes](#) = null
- private bool [_Running](#)
- public bool [Running](#)
- Some visual adjustments when ADC24 starts/stops*
- public RemoteMode [remoteMode](#)
- Current remode mode - defines the context next group shots*
- public event RemoteModeHandler [OnRemoteMode](#)
- public RemoteMessaging [remote](#) { get
- [set](#)
- public event [StartHandler](#) [OnStart](#)
- public event [RemoteHandler](#) [OnRemote](#)
- public event [LogHandler](#) [OnLog](#)
- private bool [_jumboButton](#) = true
- private bool [jumboButton](#)
- Set the main scan button to Jumbo mode*
- public event ActiveRemoteHandler [OnActiveRemote](#)
- private MMexec [grpMME](#)
- private MMexec [lastMMEin](#)
- private MMexec [lastMMEout](#)
- private bool [_PID_Enabled](#)
- public bool [PID_Enabled](#)
- PID follow the strobe position*
- private double [lastContrast](#) = -1
- int [runI](#) = 0
- string [configFile](#)
- List< double > [iStack](#)
- List< double > [dStack](#)
- private FileLogger [logger](#)
- public Point [Down](#)
- public Point [Up](#)
- public Point [Low](#)
- string[] [Titles](#) = { "runI", "tP", "tI", "tD", "Down.X", "Up.X", "disbal", "corr", "iSD-R", "contrast" }

5.1.1 Function Documentation

5.1.1.1 ActiveRemote()

```
protected void Axel_hub::ActiveRemote (
    bool active )
```

Definition at line 460 of file scanUC.xaml.cs.

5.1.1.2 backMME()

```
public MMexec Axel_hub::backMME (
    int runID,
    double asymmetry,
    MMexec mme = null )
```

Prepare back message with new Raman phase value

Parameters

<i>runID</i>	Shot number
<i>asymmetry</i>	Asymmetry
<i>mme</i>	mme is ONLY for incoming axel-probe feed

Returns

lastMMEout.mmexec.Equals("");

Definition at line 368 of file strobesUC.xaml.cs.

5.1.1.3 calcContrast()

```
public double Axel_hub::calcContrast (
    double A )
```

Calculating contrast

Parameters

<i>A</i>	Asymetry (signal)
----------	-------------------

Returns

Calculated contrast

Definition at line 251 of file strobesUC.xaml.cs.

5.1.1.4 centreFringe()

```
public double Axel_hub::centreFringe ( )
```

Calculating fringe centre

Returns

Definition at line 241 of file strobesUC.xaml.cs.

5.1.1.5 deconstructAccel()

```
public Dictionary<string, double> Axel_hub::deconstructAccel (
    double accel,
    double mems )
```

Deconstructing accaleration to acceleration components - see dictionary keys

Parameters

<i>accel</i>	acceleration [mg] - target(real)
<i>mems</i>	mems accel.[mg] - measured (real + noise)

Returns

Definition at line 271 of file strobesUC.xaml.cs.

5.1.1.6 dispatcherTimer_Tick()

```
private void Axel_hub::dispatcherTimer_Tick (
    object sender,
    EventArgs e )
```

Shows visual progress of ADC24 acquisition

Parameters

<i>sender</i>	
<i>e</i>	

Definition at line 145 of file scanUC.xaml.cs.

5.1.1.7 fillReport()

```
private void Axel_hub::fillReport (
    Dictionary< string, double > rpr )
```

Update table with strobes/PID calculation results

Parameters

<i>rpr</i>	
------------	--

Definition at line 423 of file strobesUC.xaml.cs.

5.1.1.8 Flip()

```
public void Axel_hub::Flip ( )
```

Exchange UP/DOWN strobe positions

Definition at line 178 of file strobesUC.xaml.cs.

5.1.1.9 GetBufferSize()

```
public int Axel_hub::GetBufferSize ( )
```

Get the buffer size depending of settings

Returns

Definition at line 342 of file scanUC.xaml.cs.

5.1.1.10 GetSamplingPeriod()

```
public double Axel_hub::GetSamplingPeriod ( )
```

Get the sampling period regardless the units

Returns

[s]

Definition at line 315 of file scanUC.xaml.cs.

5.1.1.11 Image_MouseDown()

```
private void Axel_hub::Image_MouseDown (
    object sender,
    MouseButtonEventArgs e )
```

Definition at line 305 of file scanUC.xaml.cs.

5.1.1.12 Init()

```
public void Axel_hub::Init (
    string _prefix )
```

Initiate strobe from file settings

Parameters

<i>_prefix</i>	
----------------	--

Definition at line 189 of file strobesUC.xaml.cs.

5.1.1.13 InitOptions()

```
public void Axel_hub::InitOptions (
    ref GeneralOptions _genOptions,
    ref ScanModes _scanModes )
```

Initialize - set genOptions

Parameters

<i>_genOptions</i>	From options windows
<i>_scanModes</i>	From saved last used modes

Definition at line 76 of file scanUC.xaml.cs.

5.1.1.14 LogEvent()

```
public void Axel_hub::LogEvent (
    string txt,
    Color? clr = null )
```

Definition at line 289 of file scanUC.xaml.cs.

5.1.1.15 LogHandler()

```
public delegate void Axel_hub::LogHandler (
    string txt,
    Color? clr = null )
```

Log into left text box

Log event for message export

Parameters

<i>txt</i>	
<i>clr</i>	

5.1.1.16 nextShot()

```
public double Axel_hub::nextShot (
    int runID,
    double asymmetry,
    out double correction )
```

Calculated phaseCorr - corrected Raman phase (0 if not PID)

Parameters

<i>runID</i>	Shot number
<i>asymmetry</i>	Asymetry value
<i>correction</i>	The correction value

Returns

The corrected position

Definition at line 307 of file strobesUC.xaml.cs.

5.1.1.17 OnActiveComm()

```
private void Axel_hub::OnActiveComm (
    bool active,
    bool forced )
```

Event when the connection goes ON/OFF

Parameters

<i>active</i>	
---------------	--

Definition at line 469 of file scanUC.xaml.cs.

5.1.1.18 OnAsyncSend()

```
protected void Axel_hub::OnAsyncSend (
    bool OK,
    string json2send )
```

Report sent message in log

Parameters

<i>OK</i>	
<i>json2send</i>	

Definition at line 299 of file scanUC.xaml.cs.

5.1.1.19 OnJumboRepeat()

```
public void Axel_hub::OnJumboRepeat (
    double _fringeScale,
    double _fringeShift,
    MMexec _grpMME,
    double contrastV )
```

Call this before each Jumbo Repeat for group MMexec and modes synchronization

Parameters

<i>_fringeScale</i>	
<i>_fringeShift</i>	
<i>_grpMME</i>	
<i>contrastV</i>	

Definition at line 203 of file strobesUC.xaml.cs.

5.1.1.20 OnRealSampling()

```
public void Axel_hub::OnRealSampling (
    double _realSampling )
```

Definition at line 134 of file scanUC.xaml.cs.

5.1.1.21 OnReceive()

```
private bool Axel_hub::OnReceive (
    string message )
```

Incomming from MM2/Axel-probe message

Parameters

<i>message</i>	
----------------	--

Returns

Definition at line 242 of file scanUC.xaml.cs.

5.1.1.22 OpenConfigFile()

```
public void Axel_hub::OpenConfigFile ( )
```

Open Config file from Config directory of Axel-hub

Definition at line 516 of file strobesUC.xaml.cs.

5.1.1.23 PID()

```
public double Axel_hub::PID (
    double disbalance )
```

Calculating the phase correction from the disbalance on strobes Ys

Parameters

<i>disbalance</i>	
-------------------	--

Returns

Definition at line 457 of file strobesUC.xaml.cs.

5.1.1.24 RemoteEvent()

```
protected void Axel_hub::RemoteEvent (
    string msg )
```

Definition at line 277 of file scanUC.xaml.cs.

5.1.1.25 RemoteHandler()

```
public delegate void Axel_hub::RemoteHandler (
    string msg )
```

Incomming message event thingy

Parameters

<i>msg</i>	
------------	--

5.1.1.26 RemoteModeEvent()

```
protected void Axel_hub::RemoteModeEvent (
    RemoteMode oldMode,
    RemoteMode newMode )
```

Definition at line 231 of file scanUC.xaml.cs.

5.1.1.27 Reset()

```
public void Axel_hub::Reset ( )
```

Initaile strobe for axel-probe simulated fringes

Definition at line 169 of file strobesUC.xaml.cs.

5.1.1.28 SaveConfigFile()

```
public void Axel_hub::SaveConfigFile ( )
```

Save Config file in Config directory of Axel-hub

Definition at line 502 of file strobesUC.xaml.cs.

5.1.1.29 scanClass()

```
public Axel_hub::scanClass ( )
```

Class constructor - set defaults

Definition at line 55 of file scanUC.xaml.cs.

5.1.1.30 SendJson()

```
public bool Axel_hub::SendJson (
    string json,
    bool async = false )
```

Wrapper of remote.sendCommand

Parameters

<i>json</i>	
<i>async</i>	

Returns

Definition at line 107 of file scanUC.xaml.cs.

5.1.1.31 SetActivity()

```
public void Axel_hub::SetActivity (
    string act )
```

Definition at line 114 of file scanUC.xaml.cs.

5.1.1.32 SetFringeParams()

```
public void Axel_hub::SetFringeParams (
    FringeParams fp )
```

Show fringes params

Parameters

<i>fp</i>	fringes params
-----------	----------------

Definition at line 129 of file scanUC.xaml.cs.

5.1.1.33 SetSamplingRate()

```
public void Axel_hub::SetSamplingRate (
    int rate )
```

Definition at line 119 of file scanUC.xaml.cs.

5.1.1.34 StartDelegate()

```
public delegate void Axel_hub::StartDelegate ( )
```

5.1.1.35 StartEvent()

```
protected void Axel_hub::StartEvent (
    bool jumbo,
    bool down,
    double period,
    int sizeLimit )
```

Definition at line 266 of file scanUC.xaml.cs.

5.1.1.36 StartHandler()

```
public delegate void Axel_hub::StartHandler (
    bool jumbo,
    bool down,
    double period,
    int sizeLimit )
```

Start/Stop group operation wity ADC24 params

Parameters

<i>jumbo</i>	
<i>down</i>	
<i>period</i>	
<i>sizeLimit</i>	

5.1.1.37 Status()

```
private void Axel_hub::Status (
    string sts )
```

Definition at line 161 of file scanUC.xaml.cs.

5.1.1.38 strobesUC()

```
public Axel_hub::strobesUC ( )
```

Class constructor

Definition at line 157 of file strobesUC.xaml.cs.

5.1.1.39 UpdateModes()

```
public void Axel_hub::UpdateModes ( )
```

Set internal from visual modes

Definition at line 92 of file scanUC.xaml.cs.

5.1.1.40 UserControl_Loaded()

```
private void Axel_hub::UserControl_Loaded (
    object sender,
    RoutedEventArgs e )
```

Some secondary to contructor initialilizations

Parameters

<i>sender</i>	
<i>e</i>	

Definition at line 494 of file scanUC.xaml.cs.

5.1.1.41 zeroFringe()

```
public double Axel_hub::zeroFringe ( )
```

Calculating zeroFringe - similar to centreFring but woth phase shift compensation

Returns

Calculated zeroFringe [rad] [-pi..pi]

Definition at line 260 of file strobesUC.xaml.cs.

5.1.2 Variable Documentation

5.1.2.1 _jumboButton

```
private bool Axel_hub::_jumboButton = true
```

Definition at line 360 of file scanUC.xaml.cs.

5.1.2.2 _PID_Enabled

```
private bool Axel_hub::_PID_Enabled
```

Definition at line 114 of file strobesUC.xaml.cs.

5.1.2.3 _Running

```
private bool Axel_hub::_Running
```

Definition at line 166 of file scanUC.xaml.cs.

5.1.2.4 ArrangedPartner

```
private string Axel_hub::ArrangedPartner = ""
```

Definition at line 47 of file scanUC.xaml.cs.

5.1.2.5 configFile

```
string Axel_hub::configFile
```

Definition at line 146 of file strobesUC.xaml.cs.

5.1.2.6 currentTime

```
TimeSpan Axel_hub::currentTime
```

Definition at line 49 of file scanUC.xaml.cs.

5.1.2.7 Down

```
public Point Axel_hub::Down
```

Definition at line 150 of file strobesUC.xaml.cs.

5.1.2.8 dStack

```
List<double> Axel_hub::dStack
```

Definition at line 147 of file strobesUC.xaml.cs.

5.1.2.9 dTimer

```
public DispatcherTimer Axel_hub::dTimer
```

Definition at line 50 of file scanUC.xaml.cs.

5.1.2.10 genOptions

```
GeneralOptions Axel_hub::genOptions = null
```

Definition at line 68 of file scanUC.xaml.cs.

5.1.2.11 grpMME

```
private MMexec Axel_hub::grpMME
```

Definition at line 112 of file strobesUC.xaml.cs.

5.1.2.12 iStack

```
List<double> Axel_hub::iStack
```

Definition at line 147 of file strobesUC.xaml.cs.

5.1.2.13 jumboButton

```
private bool Axel_hub::jumboButton
```

Set the main scan button to Jumbo mode

Definition at line 365 of file scanUC.xaml.cs.

5.1.2.14 lastContrast

```
private double Axel_hub::lastContrast = -1
```

Definition at line 144 of file strobesUC.xaml.cs.

5.1.2.15 lastMMEin

```
private MMexec Axel_hub::lastMMEin
```

Definition at line 112 of file strobesUC.xaml.cs.

5.1.2.16 lastMMEout

```
private MMexec Axel_hub::lastMMEout
```

Definition at line 112 of file strobesUC.xaml.cs.

5.1.2.17 logger

```
private FileLogger Axel_hub::logger
```

Definition at line 148 of file strobesUC.xaml.cs.

5.1.2.18 Low

```
public Point Axel_hub::Low
```

Definition at line 152 of file strobesUC.xaml.cs.

5.1.2.19 OnActiveRemote

```
public event ActiveRemoteHandler Axel_hub::OnActiveRemote
```

Definition at line 459 of file scanUC.xaml.cs.

5.1.2.20 OnLog

public event [LogHandler](#) Axel_hub::OnLog

Definition at line 288 of file scanUC.xaml.cs.

5.1.2.21 OnRemote

public event [RemoteHandler](#) Axel_hub::OnRemote

Definition at line 276 of file scanUC.xaml.cs.

5.1.2.22 OnRemoteMode

public event [RemoteModeHandler](#) Axel_hub::OnRemoteMode

Definition at line 230 of file scanUC.xaml.cs.

5.1.2.23 OnStart

public event [StartHandler](#) Axel_hub::OnStart

Definition at line 265 of file scanUC.xaml.cs.

5.1.2.24 PID_Enabled

public bool Axel_hub::PID_Enabled

Initial value:

```
{
    get { return _PID_Enabled; }
    set
    {
        _PID_Enabled = value;
        if (value) lbTitle.Content = "PID - ON";
        else lbTitle.Content = "PID - OFF";
    }
}

private bool LogPID { get { return chkPIDlog.IsChecked.Value; } }
private bool Rpr2file { get { return chkRpr2file.IsChecked.Value; } }
public string prefix { get; private set; }
public double kP { get { return ndKP.Value; } private set { ndKP.Value = value; } }
public double kI { get { return ndKI.Value; } private set { ndKI.Value = value; } }
public double kD { get { return ndKD.Value; } private set { ndKD.Value = value; } }
public int PiWeight { get { return ndPiWeight.Value; } private set { ndPiWeight.Value = value; } }
public int kIdepth { get { return ndKIdepth.Value; } private set { ndKIdepth.Value = value; } }
public int kDdepth { get { return ndKDdepth.Value; } private set { ndKDdepth.Value = value; } }
public int FreqContrast { get { return ndFreqContrast.Value; } private set { ndFreqContrast.Value = value; } }
public Dictionary<string, double> accelSet { get; private set; }
public double fringeScale { get; private set; }
public double fringeShift { get; private set; }
public double disbalNorm { get; private set; }
private double refContrast = -1
```

PID follow the strobe position

Definition at line 119 of file strobesUC.xaml.cs.

5.1.2.25 realSampling

```
public struct Axel_hub::FringeParams Axel_hub::realSampling
```

Interaction logic for UserControl1.xaml

5.1.2.26 remote

```
public RemoteMessaging Axel_hub::remote { get
```

Definition at line 236 of file scanUC.xaml.cs.

5.1.2.27 remoteMode

```
public RemoteMode Axel_hub::remoteMode
```

Initial value:

```
{
    get { return _remoteMode; }
    set
    {
        lbMode.Content = "Oper.Mode: " + value.ToString();
        if (value == RemoteMode.Simple_Repeat || value == RemoteMode.Simple_Scan || value ==
RemoteMode.Disconnected) bbtnStart.Visibility = System.Windows.Visibility.Collapsed;
        else bbtnStart.Visibility = System.Windows.Visibility.Visible;
        RemoteMode tempRemoteMode = _remoteMode; _remoteMode = value; scanModes.remoteMode = value;
        if (!tempRemoteMode.Equals(value)) RemoteModeEvent(tempRemoteMode, value);
        tabControl.IsEnabled = (value == RemoteMode.Ready_To_Remote || value ==
RemoteMode.Disconnected);
    }
    public delegate void RemoteModeHandler(RemoteMode oldMode, RemoteMode newMode)
```

Current remode mode - defines the context next group shots

Definition at line 217 of file scanUC.xaml.cs.

5.1.2.28 runI

```
int Axel_hub::runI = 0
```

Definition at line 145 of file strobesUC.xaml.cs.

5.1.2.29 Running

```
public bool Axel_hub::Running
```

Some visual adjustments when ADC24 starts/stops

Definition at line 171 of file scanUC.xaml.cs.

5.1.2.30 scanModes

```
public ScanModes Axel_hub::scanModes = null
```

Definition at line 69 of file scanUC.xaml.cs.

5.1.2.31 set

```
Axel_hub::set
```

Definition at line 236 of file scanUC.xaml.cs.

5.1.2.32 Titles

```
string [] Axel_hub::Titles = { "runI", "tP", "tI", "tD", "Down.X", "Up.X", "disbal", "corr",  
"iSD-R", "contrast" }
```

Definition at line 418 of file strobesUC.xaml.cs.

5.1.2.33 totalTime

```
TimeSpan Axel_hub::totalTime
```

Definition at line 49 of file scanUC.xaml.cs.

5.1.2.34 Up

```
public Point Axel_hub::Up
```

Definition at line 151 of file strobesUC.xaml.cs.

5.2 Axel_hub::Properties Namespace Reference

Classes

- class [Resources](#)
A strongly-typed resource class, for looking up localized strings, etc.
- class [Settings](#)

5.3 OptionsNS Namespace Reference

Classes

- class [GeneralOptions](#)
general options from Options dialog window accesable everywhere
- class [Modes](#)
Visuals and prameters for Top: Axel-chart Middle: Signal panel charts Bottom: Scan and Accel trend tabs/charts
- class [OptionsWindow](#)
OptionsWindow
- class [ScanModes](#)
visuals for the app, MEMS acquisition params and scan modes

Enumerations

- enum [RemoteMode](#) {
[RemoteMode.Disconnected](#), [RemoteMode.Jumbo_Scan](#), [RemoteMode.Jumbo_Repeat](#), [RemoteMode.Simple_Scan](#),
[RemoteMode.Simple_Repeat](#), [RemoteMode.Ready_To_Remote](#) }
The mode negotiated with MM2

5.3.1 Enumeration Type Documentation

5.3.1.1 RemoteMode

```
enum OptionsNS.RemoteMode [strong]
```

The mode negotiated with MM2

Enumerator

Disconnected	
Jumbo_Scan	
Jumbo_Repeat	
Simple_Scan	
Simple_Repeat	
Ready_To_Remote	

Definition at line 24 of file OptionsType.cs.

5.4 XamlGeneratedNamespace Namespace Reference

Classes

- class [GeneratedInternalTypeHelper](#)
[GeneratedInternalTypeHelper](#)

Chapter 6

Class Documentation

6.1 Axel_hub.accelCalibr Struct Reference

Acceleration calibration with optional temperature compensation particular to each MEMS device

Public Member Functions

- double [accel](#) (double [accelV](#), double [temperV](#), bool [tempComp](#)=false)
The actual calibration from [V] to [mg] with optional temperature compensation

Public Attributes

- string [model](#)
- string [SN](#)
- double [rAccel](#)
- double [rTemper](#)
- double [cK0](#)
- double [cK1](#)
- double[] [pK0](#)
- double[] [pK1](#)

6.1.1 Detailed Description

Acceleration calibration with optional temperature compensation particular to each MEMS device

Definition at line 21 of file AxelHMems.cs.

6.1.2 Member Function Documentation

6.1.2.1 accel()

```
double Axel_hub.accelCalibr.accel (
    double accelV,
    double temperV,
    bool tempComp = false )
```

The actual calibration from [V] to [mg] with optional temperature compensation

Parameters

<i>accelV</i>	
<i>temperV</i>	
<i>tempComp</i>	

Returns

Definition at line 39 of file AxelHMems.cs.

6.1.3 Member Data Documentation

6.1.3.1 cK0

```
double Axel_hub.accelCalibr.cK0
```

Definition at line 27 of file AxelHMems.cs.

6.1.3.2 cK1

```
double Axel_hub.accelCalibr.cK1
```

Definition at line 28 of file AxelHMems.cs.

6.1.3.3 model

```
string Axel_hub.accelCalibr.model
```

Definition at line 23 of file AxelHMems.cs.

6.1.3.4 pK0

```
double [ ] Axel_hub.accelCalibr.pK0
```

Definition at line 29 of file AxelHMems.cs.

6.1.3.5 pK1

```
double [ ] Axel_hub.accelCalibr.pK1
```

Definition at line 30 of file AxelHMems.cs.

6.1.3.6 rAccel

```
double Axel_hub.accelCalibr.rAccel
```

Definition at line 25 of file AxelHMems.cs.

6.1.3.7 rTemper

```
double Axel_hub.accelCalibr.rTemper
```

Definition at line 26 of file AxelHMems.cs.

6.1.3.8 SN

```
string Axel_hub.accelCalibr.SN
```

Definition at line 24 of file AxelHMems.cs.

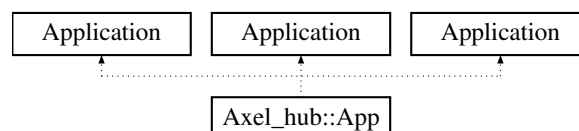
The documentation for this struct was generated from the following file:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/[AxelHMems.cs](#)

6.2 Axel_hub::App Class Reference

Interaction logic for App.xml

Inheritance diagram for Axel_hub::App:



6.2.1 Detailed Description

Interaction logic for App.xaml

[App](#)

Definition at line 14 of file App.xaml.cs.

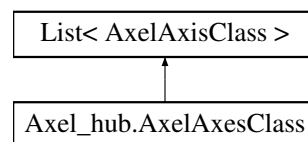
The documentation for this class was generated from the following files:

- [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/App.xaml.cs](#)
- [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/App.g.cs](#)
- [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/App.g.i.cs](#)

6.3 Axel_hub.AxelAxesClass Class Reference

Intermediator between incoming data flow from ucScan user component and AxelAxis user components

Inheritance diagram for Axel_hub.AxelAxesClass:



Public Member Functions

- int [prfIdx](#) (string prf)
Get an index from a prefix (X/Y)
- void [Clear](#) (bool Top=true, bool Middle=true, bool Bottom=true)
Clear and initialize visuals according to the switches
- [AxelAxesClass](#) (ref [GeneralOptions](#) _genOptions, ref [scanClass](#) _ucScan)
Class constructor
- void [AddAxis](#) (ref [AxelAxisClass](#) AxelAxis, string prefix)
The correct way to introduce new axis
- [AxelAxisClass](#) [byName](#) (char prefix)
Get an axis by prefix
- void [UpdateFromOptions](#) (bool activeComm)
When the options change, make everybody knows
- delegate void [LogHandler](#) (string txt, Color? clr=null)
The correct way to log a text on the text-box on the left
- void [set2startADC24](#) (bool down, double samplingPeriod, int InnerBufferSize)
Initialize ADC24 for a new measurement
- void [SaveDefaultModes](#) ()
Save the visual options
- void [startADC](#) (bool down, double period, int InnerBufferSize)
Start new measurement with ADC24
- void [DoAcquire](#) (List< Point > dt, out bool next)

- Get the acquisition buffer and distribute the data to axes*
- void [DoAcquireTemperature](#) (List< Point > dt, out bool next)
 - Acquire the temperature measurements and send the average to the corresponding axelChart*
- void [DoRemote](#) (string json)
 - The main MOT data getting method format shot.X / shot.Y OR shotData. IMPORTANT for Jumbo-repeat only .X/.Y if .X/.Y runID's are independant for each axis*
- void [DoJumboScan](#) (bool down)
 - When in Jumbo mode Start/Stop the first part of it*
- void [SetChartStrobes](#) (bool enabled)
- void [jumboRepeat](#) (int cycles)
 - When in Jumbo mode start the second part of it*
- void [Closing](#) (object sender, System.ComponentModel.CancelEventArgs e)
 - Not destroying anything, just preparing for closing*

Public Attributes

- [AxelMems axelMems](#) = null

Protected Member Functions

- void [LogEvent](#) (string txt, Color? clr=null)

Properties

- int [rCount](#) [get, set]
 - number of real (active) Axes*
- bool [memsRunning](#) [get, set]
 - Mask running for the active axelChart*

Events

- [LogHandler OnLog](#)

6.3.1 Detailed Description

Intermediator between incoming data flow from ucScan user component and AxelAxis user components

- encapsulate not-axis-specific objects (e.g axelMems) and operations (e.g. DoAcquire)

Definition at line 22 of file AxelAxes.cs.

6.3.2 Constructor & Destructor Documentation

6.3.2.1 AxelAxesClass()

```
Axel_hub.AxelAxesClass.AxelAxesClass (
    ref GeneralOptions _genOptions,
    ref scanClass _ucScan )
```

Class constructor

Parameters

<i>_genOptions</i>	general for the app options
<i>_ucScan</i>	the scan user user component ref

Definition at line 98 of file AxelAxes.cs.

6.3.3 Member Function Documentation

6.3.3.1 AddAxis()

```
void Axel_hub.AxelAxesClass.AddAxis (
    ref AxelAxisClass AxelAxis,
    string prefix )
```

The correct way to introduce new axis

Parameters

<i>AxelAxis</i>	
<i>prefix</i>	X or Y

Definition at line 115 of file AxelAxes.cs.

6.3.3.2 byName()

```
AxelAxisClass Axel_hub.AxelAxesClass.byName (
    char prefix )
```

Get an axis by prefix

Parameters

<i>prefix</i>	
---------------	--

Returns

Definition at line 128 of file AxelAxes.cs.

6.3.3.3 Clear()

```
void Axel_hub.AxelAxesClass.Clear (
    bool Top = true,
    bool Middle = true,
    bool Bottom = true )
```

Clear and initialize visuals according to the switches

Parameters

<i>Top</i>	top panel
<i>Middle</i>	middle panel
<i>Bottom</i>	bottom panel

Definition at line 86 of file AxelAxes.cs.

6.3.3.4 Closing()

```
void Axel_hub.AxelAxesClass.Closing (
    object sender,
    System.ComponentModel.CancelEventArgs e )
```

Not destroying anything, just preparing for closing

Parameters

<i>sender</i>	
<i>e</i>	

Definition at line 618 of file AxelAxes.cs.

6.3.3.5 DoAcquire()

```
void Axel_hub.AxelAxesClass.DoAcquire (
    List< Point > dt,
    out bool next )
```

Get the acquisition buffer and distribute the data to axes

Parameters

<i>dt</i>	
<i>next</i>	

Definition at line 246 of file AxelAxes.cs.

6.3.3.6 DoAcquireTemperature()

```
void Axel_hub.AxelAxesClass.DoAcquireTemperature (
    List< Point > dt,
    out bool next )
```

Acquire the temperature measurements and send the average to the corresponding axelChart

Parameters

<i>dt</i>	
<i>next</i>	

Definition at line 296 of file AxelAxes.cs.

6.3.3.7 DoJumboScan()

```
void Axel_hub.AxelAxesClass.DoJumboScan (
    bool down )
```

When in Jumbo mode Start/Stop the first part of it

Parameters

<i>down</i>	
-------------	--

Definition at line 473 of file AxelAxes.cs.

6.3.3.8 DoRemote()

```
void Axel_hub.AxelAxesClass.DoRemote (
    string json )
```

The main MOT data getting method format shot.X / shot.Y OR shotData. IMPORTANT for Jumbo-repeat only .X/.Y if .X / .Y runID's are independant for each axis

Parameters

<i>json</i>	Whatever is comming, it must be formatted according to "Book of JaSON"
-------------	--

Definition at line 351 of file AxelAxes.cs.

6.3.3.9 jumboRepeat()

```
void Axel_hub.AxelAxesClass.jumboRepeat (
    int cycles )
```

When in Jumbo mode start the second part of it

Parameters

<i>cycles</i>	set the number of shots or -1 to continues measurement
---------------	--

Definition at line 541 of file AxelAxes.cs.

6.3.3.10 LogEvent()

```
void Axel_hub.AxelAxesClass.LogEvent (
    string txt,
    Color? clr = null ) [protected]
```

Definition at line 161 of file AxelAxes.cs.

6.3.3.11 LogHandler()

```
delegate void Axel_hub.AxelAxesClass.LogHandler (
    string txt,
    Color? clr = null )
```

The correct way to log a text on the text-box on the left

Parameters

<i>txt</i>	
<i>clr</i>	

6.3.3.12 prfIdx()

```
int Axel_hub.AxelAxesClass.prfIdx (
    string prf )
```

Get an index from a prefix (X/Y)

Parameters

<i>prf</i>	
------------	--

Returns

Definition at line 53 of file AxelAxes.cs.

6.3.3.13 SaveDefaultModes()

```
void Axel_hub.AxelAxesClass.SaveDefaultModes ( )
```

Save the visual options

Definition at line 183 of file AxelAxes.cs.

6.3.3.14 set2startADC24()

```
void Axel_hub.AxelAxesClass.set2startADC24 (
    bool down,
    double samplingPeriod,
    int InnerBufferSize )
```

Initialize ADC24 for a new measurement

Parameters

<i>down</i>	
<i>samplingPeriod</i>	
<i>InnerBufferSize</i>	

Definition at line 172 of file AxelAxes.cs.

6.3.3.15 SetChartStrobes()

```
void Axel_hub.AxelAxesClass.SetChartStrobes (
    bool enabled )
```

Definition at line 531 of file AxelAxes.cs.

6.3.3.16 startADC()

```
void Axel_hub.AxelAxesClass.startADC (
    bool down,
    double period,
    int InnerBufferSize )
```

Start new measurement with ADC24

Parameters

<i>down</i>	
<i>period</i>	
<i>InnerBufferSize</i>	

Definition at line 195 of file AxelAxes.cs.

6.3.3.17 UpdateFromOptions()

```
void Axel_hub.AxelAxesClass.UpdateFromOptions (
    bool activeComm )
```

When the options change, make everybody knows

Parameters

<i>activeComm</i>	
-------------------	--

Definition at line 145 of file AxelAxes.cs.

6.3.4 Member Data Documentation**6.3.4.1 axelMems**

```
AxelMems Axel_hub.AxelAxesClass.axelMems = null
```

Definition at line 24 of file AxelAxes.cs.

6.3.5 Property Documentation

6.3.5.1 memsRunning

```
bool Axel_hub.AxelAxesClass.memsRunning [get], [set]
```

Mask running for the active axelChart

Definition at line 67 of file AxelAxes.cs.

6.3.5.2 rCount

```
int Axel_hub.AxelAxesClass.rCount [get], [set]
```

number of real (active) Axes

Definition at line 35 of file AxelAxes.cs.

6.3.6 Event Documentation

6.3.6.1 OnLog

```
LogHandler Axel_hub.AxelAxesClass.OnLog
```

Definition at line 160 of file AxelAxes.cs.

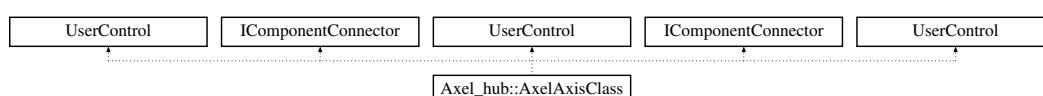
The documentation for this class was generated from the following file:

- E:/VSProjects/AxelSuite/Axel-hub/Axel-hub/[AxelAxes.cs](#)

6.4 Axel_hub::AxelAxisClass Class Reference

Interaction logic for AxelAxisUC.xaml [AxelAxisClass](#) represents a single axis of acceleration encapsulated and accesable in AxelAxes list of [AxelAxisClass](#) Future intermediary abstract movement (linear or rotation) component will be implemented.

Inheritance diagram for Axel_hub::AxelAxisClass:



6.4.1 Detailed Description

Interaction logic for AxelAxisUC.xaml [AxelAxisClass](#) represents a single axis of acceleration encapsulated and accesable in AxelAxes list of [AxelAxisClass](#) Future intermediary abstract movement (linear or rotation) component will be implemented.

[AxelAxisClass](#)

Definition at line 49 of file AxelAxisUC.xaml.cs.

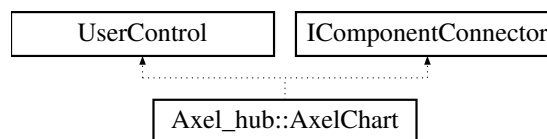
The documentation for this class was generated from the following files:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/[AxelAxisUC.xaml.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[AxelAxisUC.g.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[AxelAxisUC.g.i.cs](#)

6.5 Axel_hub::AxelChart Class Reference

[AxelChart](#)

Inheritance diagram for Axel_hub::AxelChart:



6.5.1 Detailed Description

[AxelChart](#)

Definition at line 42 of file AxelChart.g.i.cs.

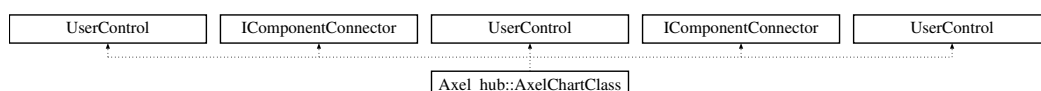
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[AxelChart.g.i.cs](#)

6.6 Axel_hub::AxelChartClass Class Reference

Interaction logic for AxelChart.xaml

Inheritance diagram for Axel_hub::AxelChartClass:



6.6.1 Detailed Description

Interaction logic for AxelChart.xaml

[AxelChartClass](#)

Definition at line 48 of file AxelChartUC.xaml.cs.

The documentation for this class was generated from the following files:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/[AxelChartUC.xaml.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[AxelChartUC.g.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[AxelChartUC.g.i.cs](#)

6.7 Axel_hub.AxelMems Class Reference

The hardware abstraction for MEMS with ADC24 (NI9251) device

Public Types

- enum [TimingModes](#) { [TimingModes.byNone](#), [TimingModes.byADCTimer](#), [TimingModes.byStopwatch](#), [TimingModes.byBoth](#) }

Public Member Functions

- [AxelMems](#) (string hwFile="", string memsFile="")
Class constructor
- void [StartStopwatch](#) ()
Stopwatch routines
- void [SetStopwatch](#) (Stopwatch ext_sw)
- double [TimeElapsed](#) ()
- double [RealConvRate](#) (double wantedCR)
Find nearest up sampling freq
- delegate void [RealSamplingHandler](#) (double [realSampling](#))
- double[,] [readBurst](#) (int nPoints)
Inner methods of continious (no gaps) data acquisition refer. NI9251 and related documentation
- void [configureVITask](#) (string physicalChn, int numbSamples, double samplingRate)
- bool [isDevicePlugged](#) ()
Check for device presence
- void [Reset](#) ()
Reset before new series of measurements
- delegate void [AcquireHandler](#) (List< Point > data, out bool next)
- void [StartAcquisition](#) (int samplesPerChannel, double samplingRate)
Set conditions for new data acquisition series
- void [StopAcquisition](#) ()

Public Attributes

- bool [AdjustTimelineToStopwatch](#) = false
false - use the set time interval between points true - adjust the time interval to stopwatch markers
- readonly double[] [FixConvRate](#)
NI9251 support fixed sampling freq listed here
- [TimingModes](#) [TimingMode](#) = [TimingModes.byNone](#)
- [accelCalibr](#) [memsX](#)
- Dictionary< string, string > [hw](#) = new Dictionary<string, string>()
- int [Timeout](#) = -1
- List< double > [rawData](#) = null

Protected Member Functions

- void [RealSamplingEvent](#) (double [realSampling](#))
- void [AcquireEvent](#) (List< Point > data, out bool next)

Properties

- int [nSamples](#) [get]
- double [sampleRate](#) [get]
- bool [running](#) [get]
- int [activeChannel](#) [get, set]

Events

- [RealSamplingHandler](#) [OnRealSampling](#)
- [AcquireHandler](#) [OnAcquire](#)

6.7.1 Detailed Description

The hardware abstraction for MEMS with ADC24 (NI9251) device

Definition at line 65 of file [AxelHMems.cs](#).

6.7.2 Member Enumeration Documentation

6.7.2.1 TimingModes

```
enum Axel\_hub.AxelMems.TimingModes [strong]
```

Enumerator

byNone	
byADCTimer	
byStopwatch	
byBoth	

Definition at line 80 of file AxelHMems.cs.

6.7.3 Constructor & Destructor Documentation

6.7.3.1 AxelMems()

```
Axel_hub.AxelMems.AxelMems (
    string hwFile = "",
    string memsFile = "" )
```

Class constructor

Parameters

<i>hwFile</i>	Hardware file (NI9251 settings)
<i>memsFile</i>	Mems calibration and teperature compensation

Definition at line 117 of file AxelHMems.cs.

6.7.4 Member Function Documentation

6.7.4.1 AcquireEvent()

```
void Axel_hub.AxelMems.AcquireEvent (
    List< Point > data,
    out bool next ) [protected]
```

Definition at line 291 of file AxelHMems.cs.

6.7.4.2 AcquireHandler()

```
delegate void Axel_hub.AxelMems.AcquireHandler (
    List< Point > data,
    out bool next )
```

6.7.4.3 configureVITask()

```
void Axel_hub.AxelMems.configureVITask (
    string physicalChn,
    int numbSamples,
    double samplingRate )
```

Definition at line 224 of file AxelHMems.cs.

6.7.4.4 isDevicePlugged()

```
bool Axel_hub.AxelMems.isDevicePlugged ( )
```

Check for device presence

Returns

Definition at line 244 of file AxelHMems.cs.

6.7.4.5 readBurst()

```
double [,] Axel_hub.AxelMems.readBurst (
    int nPoints )
```

Inner methods of continous (no gaps) data acquisition refer. NI9251 and related documentation

Definition at line 212 of file AxelHMems.cs.

6.7.4.6 RealConvRate()

```
double Axel_hub.AxelMems.RealConvRate (
    double wantedCR )
```

Find nearest up sampling freq

Parameters

<i>wantedCR</i>	Desired freq
-----------------	--------------

Returns

Definition at line 181 of file AxelHMems.cs.

6.7.4.7 RealSamplingEvent()

```
void Axel_hub.AxelMems.RealSamplingEvent (
    double realSampling ) [protected]
```

Definition at line 202 of file AxelHMems.cs.

6.7.4.8 RealSamplingHandler()

```
delegate void Axel_hub.AxelMems.RealSamplingHandler (
    double realSampling )
```

6.7.4.9 Reset()

```
void Axel_hub.AxelMems.Reset ( )
```

Reset before new series of measurements

Definition at line 263 of file AxelHMems.cs.

6.7.4.10 SetStopwatch()

```
void Axel_hub.AxelMems.SetStopwatch (
    Stopwatch ext_sw )
```

Definition at line 157 of file AxelHMems.cs.

6.7.4.11 StartAcquisition()

```
void Axel_hub.AxelMems.StartAcquisition (
    int samplesPerChannel,
    double samplingRate )
```

Set conditions for new data acquisition series

Parameters

<i>samplesPerChannel</i>	
<i>samplingRate</i>	

Definition at line 302 of file AxelHMems.cs.

6.7.4.12 StartStopwatch()

```
void Axel_hub.AxelMems.StartStopwatch ( )
```

Stopwatch routines

Definition at line 152 of file AxelHMems.cs.

6.7.4.13 StopAcquisition()

```
void Axel_hub.AxelMems.StopAcquisition ( )
```

Definition at line 448 of file AxelHMems.cs.

6.7.4.14 TimeElapsed()

```
double Axel_hub.AxelMems.TimeElapsed ( )
```

Definition at line 161 of file AxelHMems.cs.

6.7.5 Member Data Documentation

6.7.5.1 AdjustTimelineToStopwatch

```
bool Axel_hub.AxelMems.AdjustTimelineToStopwatch = false
```

false - use the set time interval between points true - adjust the time interval to stopwatch markers

Definition at line 72 of file AxelHMems.cs.

6.7.5.2 FixConvRate

```
readonly double [] Axel_hub.AxelMems.FixConvRate
```

Initial value:

```
= { 102400, 51200, 34133, 25600, 20480, 17067, 14629, 12800, 11378,  
    10240, 9309, 8533, 7314, 6400, 5689, 5120, 4655, 4267, 3657, 3200, 2844, 2560, 2327, 2133, 1829,  
    1600, 1422, 1280, 1164, 1067, 914, 800, 711, 640, 582, 533, 457, 400, 356, 320, 291, 267 }
```

NI9251 support fixed sampling freq listed here

Definition at line 76 of file AxelHMems.cs.

6.7.5.3 hw

```
Dictionary<string, string> Axel_hub.AxelMems.hw = new Dictionary<string, string>()
```

Definition at line 90 of file AxelHMems.cs.

6.7.5.4 memsX

```
accelCalibr Axel_hub.AxelMems.memsX
```

Definition at line 88 of file AxelHMems.cs.

6.7.5.5 rawData

```
List<double> Axel_hub.AxelMems.rawData = null
```

Definition at line 95 of file AxelHMems.cs.

6.7.5.6 Timeout

```
int Axel_hub.AxelMems.Timeout = -1
```

Definition at line 94 of file AxelHMems.cs.

6.7.5.7 TimingMode

```
TimingModes Axel_hub.AxelMems.TimingMode = TimingModes.byNone
```

Definition at line 81 of file AxelHMems.cs.

6.7.6 Property Documentation

6.7.6.1 activeChannel

```
int Axel_hub.AxelMems.activeChannel [get], [set]
```

Definition at line 174 of file AxelHMems.cs.

6.7.6.2 nSamples

```
int Axel_hub.AxelMems.nSamples [get]
```

Definition at line 92 of file AxelHMems.cs.

6.7.6.3 running

```
bool Axel_hub.AxelMems.running [get]
```

Definition at line 170 of file AxelHMems.cs.

6.7.6.4 sampleRate

```
double Axel_hub.AxelMems.sampleRate [get]
```

Definition at line 93 of file AxelHMems.cs.

6.7.7 Event Documentation

6.7.7.1 OnAcquire

```
AcquireHandler Axel_hub.AxelMems.OnAcquire
```

Definition at line 289 of file AxelHMems.cs.

6.7.7.2 OnRealSampling

[RealSamplingHandler](#) `Axel_hub.AxelMems.OnRealSampling`

Definition at line 200 of file `AxelHMems.cs`.

The documentation for this class was generated from the following file:

- `E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelHMems.cs`

6.8 Axel_hub.AxelMemsTemperature Class Reference

The temperature in a class abstraction

Public Member Functions

- [AxelMemsTemperature](#) (string hwFile="")
- double[] [TakeTheTemperature](#) ()

The actual temperature measurement

Public Attributes

- Dictionary< string, string > [hw](#) = new Dictionary<string, string>()

6.8.1 Detailed Description

The temperature in a class abstraction

Definition at line 458 of file `AxelHMems.cs`.

6.8.2 Constructor & Destructor Documentation

6.8.2.1 AxelMemsTemperature()

```
Axel_hub.AxelMemsTemperature.AxelMemsTemperature (
    string hwFile = "" )
```

Definition at line 462 of file `AxelHMems.cs`.

6.8.3 Member Function Documentation

6.8.3.1 TakeTheTemperature()

```
double [ ] Axel_hub.AxelMemsTemperature.TakeTheTemperature ( )
```

The actual temperature measurement

Returns

Definition at line 478 of file AxelHMems.cs.

6.8.4 Member Data Documentation

6.8.4.1 hw

```
Dictionary<string, string> Axel_hub.AxelMemsTemperature.hw = new Dictionary<string, string>()
```

Definition at line 460 of file AxelHMems.cs.

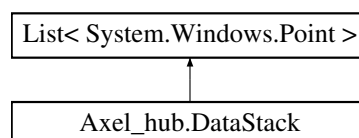
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/[AxelHMems.cs](#)

6.9 Axel_hub.DataStack Class Reference

You (developer) need to set TimeMode and one of SizeLimit or TimeLimit TimeMode is about the way [DataStack](#) limits its size The output is from standart List method ToArray in order to set DataSource of Graph

Inheritance diagram for Axel_hub.DataStack:



Public Member Functions

- [DataStack](#) (int depth=1000, string _prefix="")
Class constructor
- delegate void [RefreshHandler](#) ()
- int [Fit2Limit](#) ()
Restrict the size to about Depth length
- new void [Clear](#) ()
Clean everything up
- new int [Add](#) (Point pnt)
Overriding method to the base, assuming correct time order
- int [AddPoint](#) (double Y, double X=double.NaN)
Add point by coordinates assuming correct time order
- int [AddRange](#) (List< Point > pnts)
Add list of points assuming correct time order
- [DataStack CopyEach](#) (int each)
Copy subset of point (skipping some) to speed up visualization
- [DataStack Clone](#) (double offsetX=0, double offsetY=0)
Clone datastack with some offset applied
- [DataStack TimePortion](#) (double fromTime, double toTime)
Extract sub-DataStack for a time range
- [DataStack Portion](#) (int lastNPoints, int backFrom=-1)
Extract sub-DataStack for an index range
- [DataStack Compress](#) (int degree=5)
Another method (moving average) to extract subset of datastack for speed up visualization
- double[,] [ExportToArray](#) ()
Export data in array[,] format for NI library input
- bool [ImportFromArray](#) (double[,] da)
Import data from array[,] (NI routines)
- int [indexByX](#) (double X, bool smart=true)
Get index by time in time series
- bool [statsByIdx](#) (int FromIdx, int ToIdx, bool weightMean, out double Mean, out double stDev)
Statistics in an index range with averaging method
- bool [statsByTime](#) (double endOfTimeInterval, double duration, bool weightMean, out double Mean, out double stDev)
Statistics in a time range with averaging method
- double[] [pointXs](#) ()
Array of X coordinates
- void [Rescale](#) (double[] newXs, double offsetX=0)
Change the x scale with new one and offset
- double[] [pointYs](#) ()
Array of Y coordinates
- Point [pointSDev](#) (bool relativeY=false)
StandardDeviation by X and Y
- void [importFromArrays](#) (double[] xs, double[] ys)
Another way to import -> double[] and double[]
- void [fillSamples](#) (int n)
Fill with some random point, mostly for simulation
- bool [OpenPair](#) (string fn, ref GroupBox header, int rm=1)
Open tab separated x,y text file
- void [SavePair](#) (string fn, string rem="", string format="")
Save tab separated x,y text file

Public Attributes

- Dictionary< string, double > [RefFileStats](#)
- int [visualCountLimit](#) = -1
- int [generalIdx](#) = 0
- FileLogger [logger](#)
- Stopwatch [stopWatch](#)

Static Public Attributes

- const int [maxDepth](#) = 15000000

Protected Member Functions

- void [RefreshEvent](#) ()

Properties

- string [prefix](#) [get]
- string [rem](#) [get, set]
- string [lastError](#) [get, set]
- bool [StackMode](#) [get, set]
- bool [Running](#) [get, set]
Running the stopwatch and status
- int [Depth](#) [get, set]
- bool [TimeSeriesMode](#) [get, set]
- Point [First](#) [get]
First data point
- Point [Last](#) [get]
Last data point

Events

- [RefreshHandler OnRefresh](#)

6.9.1 Detailed Description

You (developer) need to set TimeMode and one of SizeLimit or TimeLimit TimeMode is about the way [DataStack](#) limits its size The output is from standart List method ToArray in order to set DataSource of Graph

Definition at line 38 of file DataStackLib.cs.

6.9.2 Constructor & Destructor Documentation

6.9.2.1 DataStack()

```
Axel_hub.DataStack.DataStack (
    int depth = 1000,
    string _prefix = "" )
```

Class constructor

Parameters

<i>depth</i>	-1 for Non-Stack modes
<i>_prefix</i>	

Definition at line 45 of file DataStackLib.cs.

6.9.3 Member Function Documentation

6.9.3.1 Add()

```
new int Axel_hub.DataStack.Add (  
    Point pnt )
```

Overriding method to the base, assuming correct time order

Parameters

<i>pnt</i>	
------------	--

Returns

Definition at line 152 of file DataStackLib.cs.

6.9.3.2 AddPoint()

```
int Axel_hub.DataStack.AddPoint (  
    double Y,  
    double X = double.NaN )
```

Add point by coordinates assuming correct time order

Parameters

<i>Y</i>	
<i>X</i>	

Returns

Definition at line 166 of file DataStackLib.cs.

6.9.3.3 AddRange()

```
int Axel_hub.DataStack.AddRange (
    List< Point > pnts )
```

Add list of points assuming correct time order

Parameters

<i>pnts</i>	
-------------	--

Returns

Definition at line 182 of file DataStackLib.cs.

6.9.3.4 Clear()

```
new void Axel_hub.DataStack.Clear ( )
```

Clean everything up

Definition at line 119 of file DataStackLib.cs.

6.9.3.5 Clone()

```
DataStack Axel_hub.DataStack.Clone (
    double offsetX = 0,
    double offsetY = 0 )
```

Clone datastack with some offset applied

Parameters

<i>offsetX</i>	
<i>offsetY</i>	

Returns

Definition at line 220 of file DataStackLib.cs.

6.9.3.6 Compress()

```
DataStack Axel_hub.DataStack.Compress (
    int degree = 5 )
```

Another method (moving average) to extract subset of datastack for speed up visualization

Parameters

<i>degree</i>	
---------------	--

Returns

Definition at line 288 of file DataStackLib.cs.

6.9.3.7 CopyEach()

```
DataStack Axel_hub.DataStack.CopyEach (
    int each )
```

Copy subset of point (skipping some) to speed up visualization

Parameters

<i>each</i>	
-------------	--

Returns

Definition at line 198 of file DataStackLib.cs.

6.9.3.8 ExportToArray()

```
double [,] Axel_hub.DataStack.ExportToArray ( )
```

Export data in array[,] format for NI library input

Returns

Definition at line 314 of file DataStackLib.cs.

6.9.3.9 fillSamples()

```
void Axel_hub.DataStack.fillSamples (
    int n )
```

Fill with some random point, mostly for simulation

Parameters

<i>n</i>	
----------	--

Definition at line 567 of file DataStackLib.cs.

6.9.3.10 Fit2Limit()

```
int Axel_hub.DataStack.Fit2Limit ( )
```

Restrict the size to about Depth length

Returns

Definition at line 108 of file DataStackLib.cs.

6.9.3.11 ImportFromArray()

```
bool Axel_hub.DataStack.ImportFromArray (
    double da[, ] )
```

Import data from array[,] (NI routines)

Parameters

<i>da</i>	
-----------	--

Returns

Definition at line 329 of file DataStackLib.cs.

6.9.3.12 importFromArrays()

```
void Axel_hub.DataStack.importFromArrays (
    double[] xs,
    double[] ys )
```

Another way to import -> double[] and double[]

Parameters

<i>xs</i>	
<i>ys</i>	

Definition at line 549 of file DataStackLib.cs.

6.9.3.13 indexByX()

```
int Axel_hub.DataStack.indexByX (
    double X,
    bool smart = true )
```

Get index by time in time series

Parameters

<i>X</i>	
<i>smart</i>	more direct way with equidistance asumption

Returns

Definition at line 363 of file DataStackLib.cs.

6.9.3.14 OpenPair()

```
bool Axel_hub.DataStack.OpenPair (
    string fn,
```

```
ref GroupBox header,  
int rm = 1 )
```

Open tab separated x,y text file

Parameters

<i>fn</i>	
<i>header</i>	
<i>rm</i>	

Returns

Definition at line 581 of file DataStackLib.cs.

6.9.3.15 pointSDev()

```
Point Axel_hub.DataStack.pointSDev (  
    bool relativeY = false )
```

StandardDeviation by X and Y

Parameters

<i>relativeY</i>	
------------------	--

Returns

Definition at line 533 of file DataStackLib.cs.

6.9.3.16 pointXs()

```
double [ ] Axel_hub.DataStack.pointXs ( )
```

Array of X coordinates

Returns

Definition at line 488 of file DataStackLib.cs.

6.9.3.17 pointYs()

```
double [ ] Axel_hub.DataStack.pointYs ( )
```

Array of Y coordinates

Returns

Definition at line 518 of file DataStackLib.cs.

6.9.3.18 Portion()

```
DataStack Axel_hub.DataStack.Portion (
    int lastNPoints,
    int backFrom = -1 )
```

Extract sub-DataStack for an index range

Parameters

<i>lastNPoints</i>	
<i>backFrom</i>	

Returns

Definition at line 266 of file DataStackLib.cs.

6.9.3.19 RefreshEvent()

```
void Axel_hub.DataStack.RefreshEvent ( ) [protected]
```

Definition at line 80 of file DataStackLib.cs.

6.9.3.20 RefreshHandler()

```
delegate void Axel_hub.DataStack.RefreshHandler ( )
```

6.9.3.21 Rescale()

```
void Axel_hub.DataStack.Rescale (
    double[] newXs,
    double offsetX = 0 )
```

Change the x scale with new one and offset

Parameters

<i>newXs</i>	
<i>offsetX</i>	

Definition at line 502 of file DataStackLib.cs.

6.9.3.22 SavePair()

```
void Axel_hub.DataStack.SavePair (
    string fn,
    string rem = "",
    string format = "" )
```

Save tab separated x,y text file

Parameters

<i>fn</i>	
<i>rem</i>	
<i>format</i>	

Definition at line 637 of file DataStackLib.cs.

6.9.3.23 statsByIdx()

```
bool Axel_hub.DataStack.statsByIdx (
    int FromIdx,
    int ToIdx,
    bool weightMean,
    out double Mean,
    out double stDev )
```

Statistics in an index range with averaging method

Parameters

<i>FromIdx</i>	
<i>ToIdx</i>	
<i>weightMean</i>	averaging method
<i>Mean</i>	
<i>stDev</i>	

Returns

Definition at line 395 of file DataStackLib.cs.

6.9.3.24 statsByTime()

```
bool Axel_hub.DataStack.statsByTime (
    double endOfTimeInterval,
    double duration,
    bool weightMean,
    out double Mean,
    out double stDev )
```

Statistics in a time range with averaging method

Parameters

<i>endOfTimeInterval</i>	
<i>duration</i>	
<i>weightMean</i>	averaging method
<i>Mean</i>	
<i>stDev</i>	

Returns

Definition at line 439 of file DataStackLib.cs.

6.9.3.25 TimePortion()

```
DataStack Axel_hub.DataStack.TimePortion (
    double fromTime,
    double toTime )
```

Extract sub-DataStack for a time range

Parameters

<i>fromTime</i>	
<i>toTime</i>	

Returns

Definition at line 239 of file DataStackLib.cs.

6.9.4 Member Data Documentation

6.9.4.1 generalIdx

```
int Axel_hub.DataStack.generalIdx = 0
```

Definition at line 65 of file DataStackLib.cs.

6.9.4.2 logger

```
FileLogger Axel_hub.DataStack.logger
```

Definition at line 74 of file DataStackLib.cs.

6.9.4.3 maxDepth

```
const int Axel_hub.DataStack.maxDepth = 15000000 [static]
```

Definition at line 99 of file DataStackLib.cs.

6.9.4.4 RefFileStats

```
Dictionary<string, double> Axel_hub.DataStack.RefFileStats
```

Definition at line 62 of file DataStackLib.cs.

6.9.4.5 stopWatch

```
Stopwatch Axel_hub.DataStack.stopWatch
```

Definition at line 75 of file DataStackLib.cs.

6.9.4.6 visualCountLimit

```
int Axel_hub.DataStack.visualCountLimit = -1
```

Definition at line 64 of file DataStackLib.cs.

6.9.5 Property Documentation

6.9.5.1 Depth

```
int Axel_hub.DataStack.Depth [get], [set]
```

Definition at line 100 of file DataStackLib.cs.

6.9.5.2 First

```
Point Axel_hub.DataStack.First [get]
```

First data point

Definition at line 346 of file DataStackLib.cs.

6.9.5.3 Last

```
Point Axel_hub.DataStack.Last [get]
```

Last data point

Definition at line 353 of file DataStackLib.cs.

6.9.5.4 lastError

```
string Axel_hub.DataStack.lastError [get], [set]
```

Definition at line 61 of file DataStackLib.cs.

6.9.5.5 prefix

```
string Axel_hub.DataStack.prefix [get]
```

Definition at line 59 of file DataStackLib.cs.

6.9.5.6 rem

```
string Axel_hub.DataStack.rem [get], [set]
```

Definition at line 60 of file DataStackLib.cs.

6.9.5.7 Running

```
bool Axel_hub.DataStack.Running [get], [set]
```

Running the stopwatch and status

Definition at line 90 of file DataStackLib.cs.

6.9.5.8 StackMode

```
bool Axel_hub.DataStack.StackMode [get], [set]
```

Definition at line 69 of file DataStackLib.cs.

6.9.5.9 TimeSeriesMode

```
bool Axel_hub.DataStack.TimeSeriesMode [get], [set]
```

Definition at line 102 of file DataStackLib.cs.

6.9.6 Event Documentation

6.9.6.1 OnRefresh

`RefreshHandler` `Axel_hub.DataStack.OnRefresh`

Definition at line 78 of file `DataStackLib.cs`.

The documentation for this class was generated from the following file:

- `E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/DataStackLib.cs`

6.10 Axel_hub::FringeParams Struct Reference

$\text{fringes}(\phi) = \cos(\text{period} * \phi + \text{phase}) + \text{offset}$

Public Attributes

- public double `period`
in mg per rad
- public double `phase`
the MEMS and the interferometer are not entirely paralel
- public double `offset`
phase offset [rad]

6.10.1 Detailed Description

$\text{fringes}(\phi) = \cos(\text{period} * \phi + \text{phase}) + \text{offset}$

Definition at line 26 of file `scanUC.xaml.cs`.

6.10.2 Member Data Documentation

6.10.2.1 offset

`public double Axel_hub::FringeParams::offset`

phase offset [rad]

Definition at line 39 of file `scanUC.xaml.cs`.

6.10.2.2 period

```
public double Axel_hub::FringeParams::period
```

in mg per rad

Definition at line 31 of file scanUC.xaml.cs.

6.10.2.3 phase

```
public double Axel_hub::FringeParams::phase
```

the MEMS and the interferometer are not entirely paralel

Definition at line 35 of file scanUC.xaml.cs.

The documentation for this struct was generated from the following file:

- E:/VSproujects/AxelSuite/Axel-hub/Axel-hub/[scanUC.xaml.cs](#)

6.11 OptionsNS.GeneralOptions Class Reference

general options from Options dialog window accesable everywhere

Public Types

- enum [SaveModes](#) { [SaveModes.save](#), [SaveModes.ask](#), [SaveModes.nosave](#) }

Public Member Functions

- void [Save](#) ()

Public Attributes

- [SaveModes saveModes](#)

Properties

- int [AxesChannels](#) [get, set]
- string [SignalCursorPrec](#) [get, set]
- string [SignalTablePrec](#) [get, set]
- string [SaveFilePrec](#) [get, set]
- string [LogFilePrec](#) [get, set]
- bool [intN2](#) [get, set]
- bool [saveVisuals](#) [get, set]
- bool [followPID](#) [get, set]
- int [TrendSignalLen](#) [get, set]
- int [RawSignalAvg](#) [get, set]
- bool [JumboScan](#) [get, set]
- bool [JumboRepeat](#) [get, set]
- bool [MemsInJumbo](#) [get, set]
- bool [ShowMemsIfRunning](#) [get, set]
- double [Mems2SignDelay](#) [get, set]
- double [Mems2SignLen](#) [get, set]
- bool [TemperatureEnabled](#) [get, set]
- bool [TemperatureCompensation](#) [get, set]
- string [MemsHw](#) [get, set]
- string [TemperatureHw](#) [get, set]

6.11.1 Detailed Description

general options from Options dialog window accesable everywhere

Definition at line 37 of file OptionsType.cs.

6.11.2 Member Enumeration Documentation

6.11.2.1 SaveModes

enum [OptionsNS.GeneralOptions.SaveModes](#) [strong]

Enumerator

save	
ask	
nosave	

Definition at line 39 of file OptionsType.cs.

6.11.3 Member Function Documentation

6.11.3.1 Save()

```
void OptionsNS.GeneralOptions.Save ( )
```

Definition at line 74 of file OptionsType.cs.

6.11.4 Member Data Documentation

6.11.4.1 saveModes

```
SaveModes OptionsNS.GeneralOptions.saveModes
```

Definition at line 59 of file OptionsType.cs.

6.11.5 Property Documentation

6.11.5.1 AxesChannels

```
int OptionsNS.GeneralOptions.AxesChannels [get], [set]
```

Definition at line 42 of file OptionsType.cs.

6.11.5.2 followPID

```
bool OptionsNS.GeneralOptions.followPID [get], [set]
```

Definition at line 52 of file OptionsType.cs.

6.11.5.3 intN2

```
bool OptionsNS.GeneralOptions.intN2 [get], [set]
```

Definition at line 49 of file OptionsType.cs.

6.11.5.4 JumboRepeat

```
bool OptionsNS.GeneralOptions.JumboRepeat [get], [set]
```

Definition at line 58 of file OptionsType.cs.

6.11.5.5 JumboScan

```
bool OptionsNS.GeneralOptions.JumboScan [get], [set]
```

Definition at line 57 of file OptionsType.cs.

6.11.5.6 LogFilePrec

```
string OptionsNS.GeneralOptions.LogFilePrec [get], [set]
```

Definition at line 47 of file OptionsType.cs.

6.11.5.7 Mems2SignDelay

```
double OptionsNS.GeneralOptions.Mems2SignDelay [get], [set]
```

Definition at line 65 of file OptionsType.cs.

6.11.5.8 Mems2SignLen

```
double OptionsNS.GeneralOptions.Mems2SignLen [get], [set]
```

Definition at line 66 of file OptionsType.cs.

6.11.5.9 MemsHw

```
string OptionsNS.GeneralOptions.MemsHw [get], [set]
```

Definition at line 71 of file OptionsType.cs.

6.11.5.10 MemsInJumbo

```
bool OptionsNS.GeneralOptions.MemsInJumbo [get], [set]
```

Definition at line 62 of file OptionsType.cs.

6.11.5.11 RawSignalAvg

```
int OptionsNS.GeneralOptions.RawSignalAvg [get], [set]
```

Definition at line 55 of file OptionsType.cs.

6.11.5.12 SaveFilePrec

```
string OptionsNS.GeneralOptions.SaveFilePrec [get], [set]
```

Definition at line 46 of file OptionsType.cs.

6.11.5.13 saveVisuals

```
bool OptionsNS.GeneralOptions.saveVisuals [get], [set]
```

Definition at line 50 of file OptionsType.cs.

6.11.5.14 ShowMemsIfRunning

```
bool OptionsNS.GeneralOptions.ShowMemsIfRunning [get], [set]
```

Definition at line 63 of file OptionsType.cs.

6.11.5.15 SignalCursorPrec

```
string OptionsNS.GeneralOptions.SignalCursorPrec [get], [set]
```

Definition at line 44 of file OptionsType.cs.

6.11.5.16 SignalTablePrec

```
string OptionsNS.GeneralOptions.SignalTablePrec [get], [set]
```

Definition at line 45 of file OptionsType.cs.

6.11.5.17 TemperatureCompensation

```
bool OptionsNS.GeneralOptions.TemperatureCompensation [get], [set]
```

Definition at line 69 of file OptionsType.cs.

6.11.5.18 TemperatureEnabled

```
bool OptionsNS.GeneralOptions.TemperatureEnabled [get], [set]
```

Definition at line 68 of file OptionsType.cs.

6.11.5.19 TemperatureHw

```
string OptionsNS.GeneralOptions.TemperatureHw [get], [set]
```

Definition at line 72 of file OptionsType.cs.

6.11.5.20 TrendSignalLen

```
int OptionsNS.GeneralOptions.TrendSignalLen [get], [set]
```

Definition at line 54 of file OptionsType.cs.

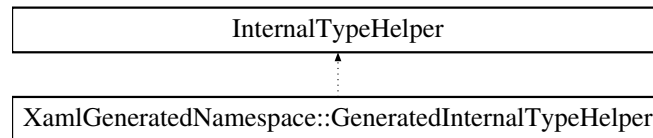
The documentation for this class was generated from the following file:

- <E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/OptionsType.cs>

6.12 XamlGeneratedNamespace::GeneratedInternalTypeHelper Class Reference

[GeneratedInternalTypeHelper](#)

Inheritance diagram for XamlGeneratedNamespace::GeneratedInternalTypeHelper:



6.12.1 Detailed Description

[GeneratedInternalTypeHelper](#)

Definition at line 20 of file `GeneratedInternalTypeHelper.g.i.cs`.

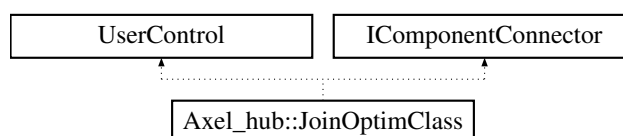
The documentation for this class was generated from the following file:

- `E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/GeneratedInternalTypeHelper.g.i.cs`

6.13 Axel_hub::JoinOptimClass Class Reference

[JoinOptimClass](#)

Inheritance diagram for `Axel_hub::JoinOptimClass`:



6.13.1 Detailed Description

[JoinOptimClass](#)

Definition at line 43 of file `JoinOptimUC.g.i.cs`.

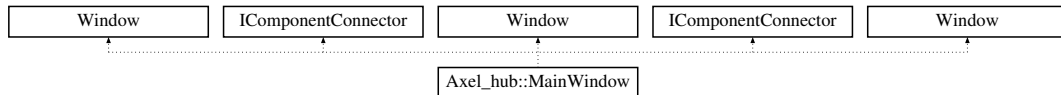
The documentation for this class was generated from the following file:

- `E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/JoinOptimUC.g.i.cs`

6.14 Axel_hub::MainWindow Class Reference

Interaction logic for MainWindow.xaml command line arguments (space separated): -remote:partner -hw↵:config.file where partner is remote partner name title; hw<c>hardware, config.file.hw is in Config folder

Inheritance diagram for Axel_hub::MainWindow:



6.14.1 Detailed Description

Interaction logic for MainWindow.xaml command line arguments (space separated): -remote:partner -hw↵:config.file where partner is remote partner name title; hw<c>hardware, config.file.hw is in Config folder

MainWindow

Definition at line 51 of file MainWindow.xaml.cs.

The documentation for this class was generated from the following files:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/[MainWindow.xaml.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[MainWindow.g.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[MainWindow.g.i.cs](#)

6.15 OptionsNS.Modes Class Reference

Visuals and prameters for Top: Axel-chart Middle: Signal panel charts Bottom: Scan and Accel trend tabs/charts

Public Member Functions

- void [Save](#) (string prefix)

Properties

- double [TopFrame](#) [get, set]
- double [TopOfTopFrame](#) [get, set]
- int [ShowFreq](#) [get, set]
- int [RollMean](#) [get, set]
- int [StackDepth](#) [get, set]
- bool [ChartUpdate](#) [get, set]
- bool [TblUpdate](#) [get, set]
- double [PowerCoeff](#) [get, set]
- double [MiddleFrame](#) [get, set]
- bool [AutoScaleMiddle](#) [get, set]
- bool [Background](#) [get, set]
- bool [DarkCurrent](#) [get, set]
- bool [StdDev](#) [get, set]
- bool [N1](#) [get, set]
- bool [N2](#) [get, set]
- bool [RN1](#) [get, set]
- bool [RN2](#) [get, set]
- bool [Ntot](#) [get, set]
- bool [RsltTblUpdate](#) [get, set]
- bool [RsltChrtUpdate](#) [get, set]
- bool [JoinLog](#) [get, set]
- bool [SignalLog](#) [get, set]
- double [JumboFrom](#) [get, set]
- double [JumboTo](#) [get, set]
- double [JumboBy](#) [get, set]
- int [JumboCycles](#) [get, set]
- bool [MemsEnabled](#) [get, set]
- double [Kcoeff](#) [get, set]
- double [phi0](#) [get, set]
- double [scale](#) [get, set]
- double [offset](#) [get, set]
- bool [AutoScaleBottom](#) [get, set]
- double [kP](#) [get, set]
- double [kI](#) [get, set]
- double [kD](#) [get, set]
- bool [PID_Enabled](#) [get, set]
- bool [DoubleStrobe](#) [get, set]

6.15.1 Detailed Description

Visuals and prameters for Top: Axel-chart Middle: Signal panel charts Bottom: Scan and Accel trend tabs/charts

Definition at line 113 of file OptionsType.cs.

6.15.2 Member Function Documentation

6.15.2.1 Save()

```
void OptionsNS.Modes.Save (
    string prefix )
```

Definition at line 160 of file OptionsType.cs.

6.15.3 Property Documentation

6.15.3.1 AutoScaleBottom

```
bool OptionsNS.Modes.AutoScaleBottom [get], [set]
```

Definition at line 153 of file OptionsType.cs.

6.15.3.2 AutoScaleMiddle

```
bool OptionsNS.Modes.AutoScaleMiddle [get], [set]
```

Definition at line 127 of file OptionsType.cs.

6.15.3.3 Background

```
bool OptionsNS.Modes.Background [get], [set]
```

Definition at line 128 of file OptionsType.cs.

6.15.3.4 ChartUpdate

```
bool OptionsNS.Modes.ChartUpdate [get], [set]
```

Definition at line 121 of file OptionsType.cs.

6.15.3.5 DarkCurrent

```
bool OptionsNS.Modes.DarkCurrent [get], [set]
```

Definition at line 129 of file OptionsType.cs.

6.15.3.6 DoubleStrobe

```
bool OptionsNS.Modes.DoubleStrobe [get], [set]
```

Definition at line 158 of file OptionsType.cs.

6.15.3.7 JoinLog

```
bool OptionsNS.Modes.JoinLog [get], [set]
```

Definition at line 138 of file OptionsType.cs.

6.15.3.8 JumboBy

```
double OptionsNS.Modes.JumboBy [get], [set]
```

Definition at line 144 of file OptionsType.cs.

6.15.3.9 JumboCycles

```
int OptionsNS.Modes.JumboCycles [get], [set]
```

Definition at line 145 of file OptionsType.cs.

6.15.3.10 JumboFrom

```
double OptionsNS.Modes.JumboFrom [get], [set]
```

Definition at line 142 of file OptionsType.cs.

6.15.3.11 JumboTo

```
double OptionsNS.Modes.JumboTo [get], [set]
```

Definition at line 143 of file OptionsType.cs.

6.15.3.12 Kcoeff

```
double OptionsNS.Modes.Kcoeff [get], [set]
```

Definition at line 148 of file OptionsType.cs.

6.15.3.13 kD

```
double OptionsNS.Modes.kD [get], [set]
```

Definition at line 156 of file OptionsType.cs.

6.15.3.14 kI

```
double OptionsNS.Modes.kI [get], [set]
```

Definition at line 155 of file OptionsType.cs.

6.15.3.15 kP

```
double OptionsNS.Modes.kP [get], [set]
```

Definition at line 154 of file OptionsType.cs.

6.15.3.16 MemsEnabled

```
bool OptionsNS.Modes.MemsEnabled [get], [set]
```

Definition at line 147 of file OptionsType.cs.

6.15.3.17 MiddleFrame

```
double OptionsNS.Modes.MiddleFrame [get], [set]
```

Definition at line 126 of file OptionsType.cs.

6.15.3.18 N1

```
bool OptionsNS.Modes.N1 [get], [set]
```

Definition at line 131 of file OptionsType.cs.

6.15.3.19 N2

```
bool OptionsNS.Modes.N2 [get], [set]
```

Definition at line 132 of file OptionsType.cs.

6.15.3.20 Ntot

```
bool OptionsNS.Modes.Ntot [get], [set]
```

Definition at line 135 of file OptionsType.cs.

6.15.3.21 offset

```
double OptionsNS.Modes.offset [get], [set]
```

Definition at line 151 of file OptionsType.cs.

6.15.3.22 phi0

```
double OptionsNS.Modes.phi0 [get], [set]
```

Definition at line 149 of file OptionsType.cs.

6.15.3.23 PID_Enabled

```
bool OptionsNS.Modes.PID_Enabled [get], [set]
```

Definition at line 157 of file OptionsType.cs.

6.15.3.24 PowerCoeff

```
double OptionsNS.Modes.PowerCoeff [get], [set]
```

Definition at line 123 of file OptionsType.cs.

6.15.3.25 RN1

```
bool OptionsNS.Modes.RN1 [get], [set]
```

Definition at line 133 of file OptionsType.cs.

6.15.3.26 RN2

```
bool OptionsNS.Modes.RN2 [get], [set]
```

Definition at line 134 of file OptionsType.cs.

6.15.3.27 RollMean

```
int OptionsNS.Modes.RollMean [get], [set]
```

Definition at line 119 of file OptionsType.cs.

6.15.3.28 RsltChrtUpdate

```
bool OptionsNS.Modes.RsltChrtUpdate [get], [set]
```

Definition at line 137 of file OptionsType.cs.

6.15.3.29 RsltTblUpdate

```
bool OptionsNS.Modes.RsltTblUpdate [get], [set]
```

Definition at line 136 of file OptionsType.cs.

6.15.3.30 scale

```
double OptionsNS.Modes.scale [get], [set]
```

Definition at line 150 of file OptionsType.cs.

6.15.3.31 ShowFreq

```
int OptionsNS.Modes.ShowFreq [get], [set]
```

Definition at line 118 of file OptionsType.cs.

6.15.3.32 SignalLog

```
bool OptionsNS.Modes.SignalLog [get], [set]
```

Definition at line 139 of file OptionsType.cs.

6.15.3.33 StackDepth

```
int OptionsNS.Modes.StackDepth [get], [set]
```

Definition at line 120 of file OptionsType.cs.

6.15.3.34 StdDev

```
bool OptionsNS.Modes.StdDev [get], [set]
```

Definition at line 130 of file OptionsType.cs.

6.15.3.35 TblUpdate

```
bool OptionsNS.Modes.TblUpdate [get], [set]
```

Definition at line 122 of file OptionsType.cs.

6.15.3.36 TopFrame

```
double OptionsNS.Modes.TopFrame [get], [set]
```

Definition at line 116 of file OptionsType.cs.

6.15.3.37 TopOfTopFrame

```
double OptionsNS.Modes.TopOfTopFrame [get], [set]
```

Definition at line 117 of file OptionsType.cs.

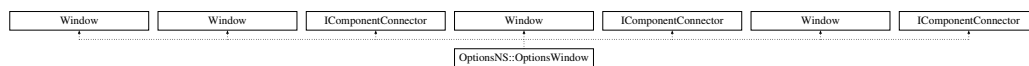
The documentation for this class was generated from the following file:

- [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/OptionsType.cs](#)

6.16 OptionsNS::OptionsWindow Class Reference

OptionsWindow

Inheritance diagram for OptionsNS::OptionsWindow:



6.16.1 Detailed Description

OptionsWindow

Interaction logic, load & save for [GeneralOptions](#) genOptions

Definition at line 42 of file Options.g.cs.

The documentation for this class was generated from the following files:

- [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options/Options.g.cs](#)
- [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options/Options.g.i.cs](#)
- [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/Options.xaml.cs](#)

6.17 Axel_hub::Properties::Resources Class Reference

A strongly-typed resource class, for looking up localized strings, etc.

6.17.1 Detailed Description

A strongly-typed resource class, for looking up localized strings, etc.

Definition at line 25 of file Resources.Designer.cs.

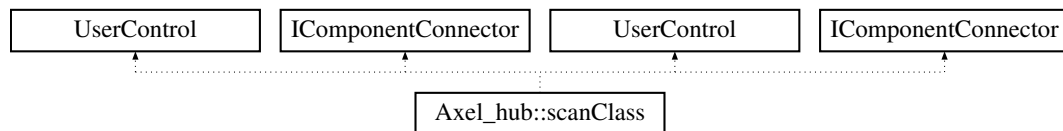
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Properties/[Resources.Designer.cs](#)

6.18 Axel_hub::scanClass Class Reference

[scanClass](#)

Inheritance diagram for Axel_hub::scanClass:



6.18.1 Detailed Description

[scanClass](#)

Definition at line 42 of file scanUC.g.cs.

The documentation for this class was generated from the following files:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[scanUC.g.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[scanUC.g.i.cs](#)

6.19 OptionsNS.ScanModes Class Reference

visuals for the app, MEMS acquisition params and scan modes

Public Member Functions

- void [Save](#) ()

Public Attributes

- [RemoteMode](#) remoteMode = [RemoteMode.Disconnected](#)

Properties

- double [Left](#) [get, set]
- double [Top](#) [get, set]
- double [Width](#) [get, set]
- double [Height](#) [get, set]
- int [SamplingFreq](#) [get, set]
- bool [TimeLimitMode](#) [get, set]
- int [TimeLimit](#) [get, set]
- bool [SizeLimitMode](#) [get, set]
- int [SizeLimit](#) [get, set]

6.19.1 Detailed Description

visuals for the app, MEMS acquisition params and scan modes

Definition at line 83 of file OptionsType.cs.

6.19.2 Member Function Documentation

6.19.2.1 Save()

```
void OptionsNS.ScanModes.Save ( )
```

Definition at line 101 of file OptionsType.cs.

6.19.3 Member Data Documentation

6.19.3.1 remoteMode

```
RemoteMode OptionsNS.ScanModes.remoteMode = RemoteMode.Disconnected
```

Definition at line 99 of file OptionsType.cs.

6.19.4 Property Documentation

6.19.4.1 Height

```
double OptionsNS.ScanModes.Height [get], [set]
```

Definition at line 89 of file OptionsType.cs.

6.19.4.2 Left

```
double OptionsNS.ScanModes.Left [get], [set]
```

Definition at line 86 of file OptionsType.cs.

6.19.4.3 SamplingFreq

```
int OptionsNS.ScanModes.SamplingFreq [get], [set]
```

Definition at line 92 of file OptionsType.cs.

6.19.4.4 SizeLimit

```
int OptionsNS.ScanModes.SizeLimit [get], [set]
```

Definition at line 96 of file OptionsType.cs.

6.19.4.5 SizeLimitMode

```
bool OptionsNS.ScanModes.SizeLimitMode [get], [set]
```

Definition at line 95 of file OptionsType.cs.

6.19.4.6 TimeLimit

```
int OptionsNS.ScanModes.TimeLimit [get], [set]
```

Definition at line 94 of file OptionsType.cs.

6.19.4.7 TimeLimitMode

```
bool OptionsNS.ScanModes.TimeLimitMode [get], [set]
```

Definition at line 93 of file OptionsType.cs.

6.19.4.8 Top

```
double OptionsNS.ScanModes.Top [get], [set]
```

Definition at line 87 of file OptionsType.cs.

6.19.4.9 Width

```
double OptionsNS.ScanModes.Width [get], [set]
```

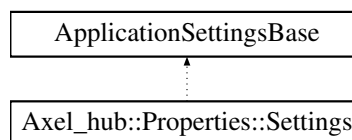
Definition at line 88 of file OptionsType.cs.

The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/[OptionsType.cs](#)

6.20 Axel_hub::Properties::Settings Class Reference

Inheritance diagram for Axel_hub::Properties::Settings:



6.20.1 Detailed Description

Definition at line 16 of file Settings.Designer.cs.

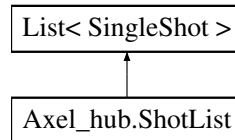
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Properties/[Settings.Designer.cs](#)

6.21 Axel_hub.ShotList Class Reference

List / series of single shots

Inheritance diagram for Axel_hub.ShotList:



Public Member Functions

- [ShotList](#) (bool arch=true, string FN="", string prefix="")
Class constructor if arch -> open file if FN not empty, or create FN if empty if not arch -> ignore FN and prefix
- new void [Add](#) ([SingleShot](#) ss)
New Add with optional log and size limit
- void [resetScan](#) ()
Reset scan of archive
- [SingleShot](#) [archiScan](#) (out bool next)
Get the next scan from a file
- void [Save](#) (string FN="")
Save a file with JSON of single shots per line read the format with ResetScan And ArchiScan

Public Attributes

- Dictionary< string, double > [conditions](#) = new Dictionary<string, double>()

Protected Attributes

- int [depth](#) = 10000

Properties

- string [filename](#) [get]
- bool [archiveMode](#) [get]
- bool [savingMode](#) [get]
- int [FileCount](#) [get]
- bool [enabled](#) [get, set]
Log ON/OFF
- int [lastIdx](#) [get]

6.21.1 Detailed Description

List / series of single shots

Definition at line 314 of file DataPrimitives.cs.

6.21.2 Constructor & Destructor Documentation

6.21.2.1 ShotList()

```
Axel_hub.ShotList.ShotList (
    bool arch = true,
    string FN = "",
    string prefix = "" )
```

Class constructor if arch -> open file if FN not empty, or create FN if empty if not arch -> ignore FN and prefix

Parameters

<i>arch</i>	log the incoming data
<i>FN</i>	
<i>prefix</i>	

Definition at line 336 of file DataPrimitives.cs.

6.21.3 Member Function Documentation

6.21.3.1 Add()

```
new void Axel_hub.ShotList.Add (
    SingleShot ss )
```

New Add with optional log and size limit

Parameters

<i>ss</i>	
-----------	--

Definition at line 362 of file DataPrimitives.cs.

6.21.3.2 archiScan()

```
SingleShot Axel_hub.ShotList.archiScan (
    out bool next )
```

Get the next scan from a file

Parameters

<i>next</i>	next is false on the last item
-------------	--------------------------------

Returns

Definition at line 427 of file DataPrimitives.cs.

6.21.3.3 resetScan()

```
void Axel_hub.ShotList.resetScan ( )
```

Reset scan of archive

Definition at line 396 of file DataPrimitives.cs.

6.21.3.4 Save()

```
void Axel_hub.ShotList.Save (
    string FN = "" )
```

Save a file with JSON of single shots per line read the format with ResetScan And ArchiScan

Parameters

<i>FN</i>	
-----------	--

Definition at line 450 of file DataPrimitives.cs.

6.21.4 Member Data Documentation**6.21.4.1 conditions**

```
Dictionary<string, double> Axel_hub.ShotList.conditions = new Dictionary<string, double>()
```

Definition at line 322 of file DataPrimitives.cs.

6.21.4.2 depth

```
int Axel_hub.ShotList.depth = 10000 [protected]
```

Definition at line 316 of file DataPrimitives.cs.

6.21.5 Property Documentation

6.21.5.1 archiveMode

```
bool Axel_hub.ShotList.archiveMode [get]
```

Definition at line 319 of file DataPrimitives.cs.

6.21.5.2 enabled

```
bool Axel_hub.ShotList.enabled [get], [set]
```

Log ON/OFF

Definition at line 380 of file DataPrimitives.cs.

6.21.5.3 FileCount

```
int Axel_hub.ShotList.FileCount [get]
```

Definition at line 324 of file DataPrimitives.cs.

6.21.5.4 filename

```
string Axel_hub.ShotList.filename [get]
```

Definition at line 317 of file DataPrimitives.cs.

6.21.5.5 lastIdx

```
int Axel_hub.ShotList.lastIdx [get]
```

Definition at line 392 of file DataPrimitives.cs.

6.21.5.6 savingMode

```
bool Axel_hub.ShotList.savingMode [get]
```

Definition at line 320 of file DataPrimitives.cs.

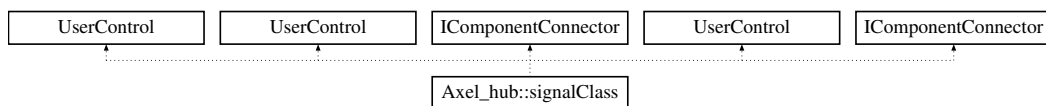
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/[DataPrimitives.cs](#)

6.22 Axel_hub::signalClass Class Reference

signalClass

Inheritance diagram for Axel_hub::signalClass:



6.22.1 Detailed Description

signalClass

Interaction logic for signalUC.xaml visualize the raw signal and signal trends {"N1", "N2", "RN1", "RN2", "NTot", "B2", "Btot"} table with last trend position and (optionally) some stats

Definition at line 42 of file signalUC.g.cs.

The documentation for this class was generated from the following files:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[signalUC.g.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[signalUC.g.i.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/[signalUC.xaml.cs](#)

6.23 Axel_hub.SingleShot Class Reference

Class representing single shot with both components quant (MOT) and MEMS (ADC24)

Public Member Functions

- [SingleShot](#) ()
Number of constructors
- [SingleShot](#) (double qTime, double qSignal)
- [SingleShot](#) (Point q)
- [SingleShot](#) (Point q, List< Point > m)
- bool [IsEmpty](#) ()
Check if empty
- int [idxByTime](#) (double tm, bool smart=true)
Find index for specific time
- List< Point > [memsPortion](#) (Range< double > rng)
Get part of mems within a reange
- double [memsWeightAccel](#) (double delay, double duration=-1, bool triangle=true)
Calculating mems acceleration time-related to quant point
- Dictionary< string, double > [deconstructAccel](#) (double fringeScale)
Accelerations components in a dictinary with order, resid, etc.

Public Attributes

- Point [quant](#)

Protected Attributes

- string [precision](#) = "G5"

Properties

- List< Point > [mems](#) [get, set]
- string [AsString](#) [get, set]
A single shot in JSON format for file import/export

6.23.1 Detailed Description

Class representing single shot with both components quant (MOT) and MEMS (ADC24)

Definition at line 126 of file DataPrimitives.cs.

6.23.2 Constructor & Destructor Documentation

6.23.2.1 SingleShot() [1/4]

```
Axel_hub.SingleShot.SingleShot ( )
```

Number of constructors

Definition at line 137 of file DataPrimitives.cs.

6.23.2.2 SingleShot() [2/4]

```
Axel_hub.SingleShot.SingleShot (
    double qTime,
    double qSignal )
```

Definition at line 142 of file DataPrimitives.cs.

6.23.2.3 SingleShot() [3/4]

```
Axel_hub.SingleShot.SingleShot (
    Point q )
```

Definition at line 147 of file DataPrimitives.cs.

6.23.2.4 SingleShot() [4/4]

```
Axel_hub.SingleShot.SingleShot (
    Point q,
    List< Point > m )
```

Definition at line 152 of file DataPrimitives.cs.

6.23.3 Member Function Documentation**6.23.3.1 deconstructAccel()**

```
Dictionary<string, double> Axel_hub.SingleShot.deconstructAccel (
    double fringeScale )
```

Accelerations components in a dictionary with order, resid, etc.

Parameters

<i>fringeScale</i>	
--------------------	--

Returns

Definition at line 255 of file DataPrimitives.cs.

6.23.3.2 idxByTime()

```
int Axel_hub.SingleShot.idxByTime (
    double tm,
    bool smart = true )
```

Find index for specific time

Parameters

<i>tm</i>	
<i>smart</i>	More direct way with some assumptions

Returns

Definition at line 174 of file DataPrimitives.cs.

6.23.3.3 IsEmpty()

```
bool Axel_hub.SingleShot.IsEmpty ( )
```

Check if empty

Returns

Definition at line 162 of file DataPrimitives.cs.

6.23.3.4 memsPortion()

```
List<Point> Axel_hub.SingleShot.memsPortion (
    Range< double > rng )
```

Get part of mems within a reange

Parameters

<i>rng</i>	
------------	--

Returns

Definition at line 201 of file DataPrimitives.cs.

6.23.3.5 memsWeightAccel()

```
double Axel_hub.SingleShot.memsWeightAccel (
    double delay,
    double duration = -1,
    bool triangle = true )
```

Calculating mems acceleration time-related to quant point

Parameters

<i>delay</i>	delay reference to quant.X
<i>duration</i>	the range length
<i>triangle</i>	alternative to triangle is uniform

Returns

Definition at line 216 of file DataPrimitives.cs.

6.23.4 Member Data Documentation

6.23.4.1 precision

```
string Axel_hub.SingleShot.precision = "G5" [protected]
```

Definition at line 128 of file DataPrimitives.cs.

6.23.4.2 quant

```
Point Axel_hub.SingleShot.quant
```

Definition at line 130 of file DataPrimitives.cs.

6.23.5 Property Documentation

6.23.5.1 AsString

```
string Axel_hub.SingleShot.AsString [get], [set]
```

A single shot in JSON format for file import/export

Definition at line 282 of file DataPrimitives.cs.

6.23.5.2 mems

```
List<Point> Axel_hub.SingleShot.mems [get], [set]
```

Definition at line 132 of file DataPrimitives.cs.

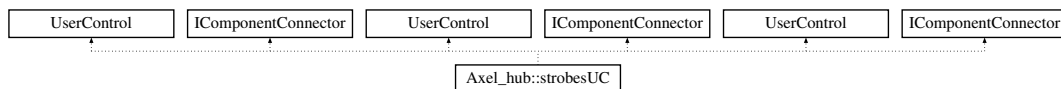
The documentation for this class was generated from the following file:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/[DataPrimitives.cs](#)

6.24 Axel_hub::strokesUC Class Reference

strokesUC

Inheritance diagram for Axel_hub::strokesUC:



6.24.1 Detailed Description

strokesUC

Definition at line 42 of file strokeControlUC.g.i.cs.

The documentation for this class was generated from the following files:

- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[strokeControlUC.g.i.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[strokesUC.g.cs](#)
- E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/[strokesUC.g.i.cs](#)

Chapter 7

File Documentation

7.1 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/App.xaml.cs File Reference

Classes

- class [Axel_hub::App](#)
Interaction logic for App.xaml

Namespaces

- [Axel_hub](#)

7.2 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AssemblyInfo.cs File Reference

7.3 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelAxes.cs File Reference

Classes

- class [Axel_hub.AxelAxesClass](#)
Intermediator between incomming data flow from ucScan user component and AxelAxis user components

Namespaces

- [Axel_hub](#)

7.4 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelAxisUC.xaml.cs File Reference

Classes

- class [Axel_hub::AxelAxisClass](#)

Interaction logic for AxelAxisUC.xaml [AxelAxisClass](#) represents a single axis of acceleration encapsulated and accesable in AxelAxes list of [AxelAxisClass](#) Future intermediary abstract movement (linear or rotation) component will be implemented.

Namespaces

- [Axel_hub](#)

7.5 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelChartUC.xaml.cs File Reference

Classes

- class [Axel_hub::AxelChartClass](#)

Interaction logic for AxelChart.xaml

Namespaces

- [Axel_hub](#)

7.6 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/AxelHMems.cs File Reference

Classes

- struct [Axel_hub.accelCalibr](#)

Acceleration calibration with optional temperature compensation particular to each MEMS device

- class [Axel_hub.AxelMems](#)

The hardware abstraction for MEMS with ADC24 (NI9251) device

- class [Axel_hub.AxelMemsTemperature](#)

The temperature in a class abstraction

Namespaces

- [Axel_hub](#)

7.7 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/DataPrimitives.cs File Reference

Classes

- class **Axel_hub.MMDataConverter**
Averaging the photo diode signals {"N2", "NTot", "B2", "BTot", "Bg"}
- class [Axel_hub.SingleShot](#)
Class representing single shot with both components quant (MOT) and MEMS (ADC24)
- class [Axel_hub.ShotList](#)
List / series of single shots

Namespaces

- [Axel_hub](#)

7.8 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/DataStackLib.cs File Reference

Classes

- class [Axel_hub.DataStack](#)
You (developer) need to set TimeMode and one of SizeLimit or TimeLimit TimeMode is about the way [DataStack](#) limits its size The output is from standart List method ToArray in order to set DataSource of Graph

Namespaces

- [Axel_hub](#)

7.9 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/MainWindow.xaml.cs File Reference

Classes

- class [Axel_hub::MainWindow](#)
Interaction logic for MainWindow.xaml command line arguments (space separated): -remote:partner -hw↔:config.file where partner is remote partner name title; hw<c>hardware, config.file.hw is in Config folder

Namespaces

- [Axel_hub](#)

Functions

- public delegate void [Axel_hub::StartDelegate](#) ()

7.10 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/App.g.cs File Reference

Classes

- class [Axel_hub::App](#)
Interaction logic for App.xaml

Namespaces

- [Axel_hub](#)

7.11 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/App.g.i.cs File Reference

Classes

- class [Axel_hub::App](#)
Interaction logic for App.xaml

Namespaces

- [Axel_hub](#)

7.12 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Axel_hub_Content.g.i.cs File Reference

7.13 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelAxisUC.g.cs File Reference

Classes

- class [Axel_hub::AxelAxisClass](#)
Interaction logic for AxelAxisUC.xaml [AxelAxisClass](#) represents a single axis of acceleration encapsulated and accesable in AxelAxes list of [AxelAxisClass](#) Future intermediator abstract movement (linear or rotation) component will be implemented.

Namespaces

- [Axel_hub](#)

7.14 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelAxisUC.g.i.cs File Reference

Classes

- class [Axel_hub::AxelAxisClass](#)

Interaction logic for AxelAxisUC.xaml [AxelAxisClass](#) represents a single axis of acceleration encapsulated and accesable in AxelAxes list of [AxelAxisClass](#) Future intermediary abstract movement (linear or rotation) component will be implemented.

Namespaces

- [Axel_hub](#)

7.15 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChart.g.i.cs File Reference

Classes

- class [Axel_hub::AxelChart](#)
[AxelChart](#)

Namespaces

- [Axel_hub](#)

7.16 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChartUC.g.cs File Reference

Classes

- class [Axel_hub::AxelChartClass](#)

Interaction logic for AxelChart.xaml

Namespaces

- [Axel_hub](#)

7.17 [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/AxelChartUC.g.i.cs File](#) Reference

Classes

- class [Axel_hub::AxelChartClass](#)
Interaction logic for AxelChart.xaml

Namespaces

- [Axel_hub](#)

7.18 [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/GeneratedInternalTypeHelper.g.cs File](#) Reference

7.19 [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/GeneratedInternalTypeHelper.g.i.cs File](#) Reference

Classes

- class [XamlGeneratedNamespace::GeneratedInternalTypeHelper](#)
GeneratedInternalTypeHelper

Namespaces

- [XamlGeneratedNamespace](#)

7.20 [E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/JoinOptimUC.g.i.cs File](#) Reference

Classes

- class [Axel_hub::JoinOptimClass](#)
JoinOptimClass

Namespaces

- [Axel_hub](#)

7.21 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/MainWindow.g.cs File Reference

Classes

- class [Axel_hub::MainWindow](#)

Interaction logic for MainWindow.xaml command line arguments (space separated): -remote:partner -hw↔:config.file where partner is remote partner name title; hw<c>hardware, config.file.hw is in Config folder

Namespaces

- [Axel_hub](#)

7.22 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/MainWindow.g.i.cs File Reference

Classes

- class [Axel_hub::MainWindow](#)

Interaction logic for MainWindow.xaml command line arguments (space separated): -remote:partner -hw↔:config.file where partner is remote partner name title; hw<c>hardware, config.file.hw is in Config folder

Namespaces

- [Axel_hub](#)

7.23 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options/Options.g.cs File Reference

Classes

- class [OptionsNS::OptionsWindow](#)
[OptionsWindow](#)

Namespaces

- [OptionsNS](#)

7.24 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options/↵ Options.g.i.cs File Reference

Classes

- class [OptionsNS::OptionsWindow](#)
[OptionsWindow](#)

Namespaces

- [OptionsNS](#)

7.25 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/Options.g.i.↵ cs File Reference

Classes

- class [OptionsNS::OptionsWindow](#)
[OptionsWindow](#)

Namespaces

- [OptionsNS](#)

7.26 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/scanUC.g.cs File Reference

Classes

- class [Axel_hub::scanClass](#)
[scanClass](#)

Namespaces

- [Axel_hub](#)

7.27 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/scanUC.g.i.cs File Reference

Classes

- class [Axel_hub::scanClass](#)
[scanClass](#)

Namespaces

- [Axel_hub](#)

7.28 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/signalUC.g.i.cs File Reference

Classes

- class [Axel_hub::signalClass](#)
[signalClass](#)

Namespaces

- [Axel_hub](#)

7.29 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/signalUC.g.i.cs File Reference

Classes

- class [Axel_hub::signalClass](#)
[signalClass](#)

Namespaces

- [Axel_hub](#)

7.30 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strobe↔ ControlUC.g.i.cs File Reference

Classes

- class [Axel_hub::strokesUC](#)
[strokesUC](#)

Namespaces

- [Axel_hub](#)

7.31 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strokes↔ UC.g.cs File Reference

Classes

- class [Axel_hub::strokesUC](#)
[strokesUC](#)

Namespaces

- [Axel_hub](#)

7.32 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/obj/Debug/strokes↔ UC.g.i.cs File Reference

Classes

- class [Axel_hub::strokesUC](#)
[strokesUC](#)

Namespaces

- [Axel_hub](#)

7.33 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/Options.xaml.cs File Reference

Classes

- class [OptionsNS::OptionsWindow](#)
OptionsWindow

Namespaces

- [OptionsNS](#)

7.34 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Options/OptionsType.cs File Reference

Classes

- class [OptionsNS.GeneralOptions](#)
general options from Options dialog window accesable everywhere
- class [OptionsNS.ScanModes](#)
visuals for the app, MEMS acquisition params and scan modes
- class [OptionsNS.Modes](#)
Visuals and prameters for Top: Axel-chart Middle: Signal panel charts Bottom: Scan and Accel trend tabs/charts

Namespaces

- [OptionsNS](#)

Enumerations

- enum [OptionsNS.RemoteMode](#) {
[OptionsNS.RemoteMode.Disconnected](#), [OptionsNS.RemoteMode.Jumbo_Scan](#), [OptionsNS.RemoteMode.Jumbo_Repeat](#),
[OptionsNS.RemoteMode.Simple_Scan](#),
[OptionsNS.RemoteMode.Simple_Repeat](#), [OptionsNS.RemoteMode.Ready_To_Remote](#) }
The mode negotiated with MM2

7.35 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Properties/Resources.Designer.cs File Reference

Classes

- class [Axel_hub::Properties::Resources](#)
A strongly-typed resource class, for looking up localized strings, etc.

Namespaces

- [Axel_hub](#)
- [Axel_hub::Properties](#)

7.36 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/Properties/Settings.Designer.cs File Reference

Classes

- class [Axel_hub::Properties::Settings](#)

Namespaces

- [Axel_hub](#)
- [Axel_hub::Properties](#)

7.37 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/scanUC.xaml.cs File Reference

Classes

- struct [Axel_hub::FringeParams](#)

$$\text{fringes}(\phi) = \cos(\text{period} * \phi + \text{phase}) + \text{offset}$$

Namespaces

- [Axel_hub](#)

Functions

- public [Axel_hub::scanClass](#) ()
Class constructor - set defaults
- public void [Axel_hub::InitOptions](#) (ref GeneralOptions _genOptions, ref ScanModes _scanModes)
Initialize - set genOptions
- public void [Axel_hub::UpdateModes](#) ()
Set internal from visual modes
- public bool [Axel_hub::SendJson](#) (string json, bool async=false)
Wrapper of remote.sendCommand
- public void [Axel_hub::SetActivity](#) (string act)
- public void [Axel_hub::SetSamplingRate](#) (int rate)
- public void [Axel_hub::SetFringeParams](#) (FringeParams fp)
Show fringes params
- public void [Axel_hub::OnRealSampling](#) (double _realSampling)

- private void [Axel_hub::dispatcherTimer_Tick](#) (object sender, EventArgs e)
Shows visual progress of ADC24 acquisition
- private void [Axel_hub::Status](#) (string sts)
- protected void [Axel_hub::RemoteModeEvent](#) (RemoteMode oldMode, RemoteMode newMode)
- private bool [Axel_hub::OnReceive](#) (string message)
Incomming from MM2/Axel-probe message
- public delegate void [Axel_hub::StartHandler](#) (bool jumbo, bool down, double period, int sizeLimit)
Start/Stop group operation wity ADC24 params
- protected void [Axel_hub::StartEvent](#) (bool jumbo, bool down, double period, int sizeLimit)
- public delegate void [Axel_hub::RemoteHandler](#) (string msg)
Incomming message event thingy
- protected void [Axel_hub::RemoteEvent](#) (string msg)
- public delegate void [Axel_hub::LogHandler](#) (string txt, Color? clr=null)
Log into left text box
- protected void [Axel_hub::LogEvent](#) (string txt, Color? clr=null)
- protected void [Axel_hub::OnAsyncSend](#) (bool OK, string json2send)
Report sent message in log
- private void [Axel_hub::Image_MouseDown](#) (object sender, MouseButtonEventArgs e)
- public double [Axel_hub::GetSamplingPeriod](#) ()
Get the sampling period regardless the units
- public int [Axel_hub::GetBufferSize](#) ()
Get the buffer size depending of settings
- protected void [Axel_hub::ActiveRemote](#) (bool active)
- private void [Axel_hub::OnActiveComm](#) (bool active, bool forced)
Event when the connection goes ON/OFF
- private void [Axel_hub::UserControl_Loaded](#) (object sender, RoutedEventArgs e)
Some secondary to contructor initializations

Variables

- public struct [Axel_hub::FringeParams](#) [Axel_hub::realSampling](#)
Interaction logic for UserControl1.xaml
- private string [Axel_hub::ArrangedPartner](#) = ""
- TimeSpan [Axel_hub::totalTime](#)
- TimeSpan [Axel_hub::currentTime](#)
- public DispatcherTimer [Axel_hub::dTimer](#)
- GeneralOptions [Axel_hub::genOptions](#) = null
- public ScanModes [Axel_hub::scanModes](#) = null
- private bool [Axel_hub::_Running](#)
- public bool [Axel_hub::Running](#)
Some visual adjustments when ADC24 starts/stops
- public RemoteMode [Axel_hub::remoteMode](#)
Current remode mode - defines the context next group shots
- public event RemoteModeHandler [Axel_hub::OnRemoteMode](#)
- public RemoteMessaging [Axel_hub::remote](#) { get
[Axel_hub::set](#)
- public event StartHandler [Axel_hub::OnStart](#)
- public event RemoteHandler [Axel_hub::OnRemote](#)
- public event LogHandler [Axel_hub::OnLog](#)
- private bool [Axel_hub::_jumboButton](#) = true
- private bool [Axel_hub::jumboButton](#)
Set the main scan button to Jumbo mode
- public event ActiveRemoteHandler [Axel_hub::OnActiveRemote](#)

7.38 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/signalUC.xaml.cs File Reference

Classes

- class [Axel_hub::signalClass](#)
signalClass

Namespaces

- [Axel_hub](#)

7.39 E:/VSprojects/AxelSuite/Axel-hub/Axel-hub/strobesUC.xaml.cs File Reference

Classes

- class **Axel_hub::calcAccel**
Library for calculating acceleration from fringes, phase, etc

Namespaces

- [Axel_hub](#)

Functions

- public [Axel_hub::strobesUC](#) ()
Class constructor
- public void [Axel_hub::Reset](#) ()
Initaile strobe for axel-probe simulated fringes
- public void [Axel_hub::Flip](#) ()
Exchange UP/DOWN strobe positions
- public void [Axel_hub::Init](#) (string _prefix)
Initiate strobe from file settings
- public void [Axel_hub::OnJumboRepeat](#) (double _fringeScale, double _fringeShift, MMexec_grpMME, double contrastV)
Call this before each Jumbo Repeat for group MMexec and modes synchronization
- public delegate void [Axel_hub::LogHandler](#) (string txt, Color? clr=null)
Log into left text box
- protected void [Axel_hub::LogEvent](#) (string txt, Color? clr=null)
- public double [Axel_hub::centreFringe](#) ()
Calculating fringe centre
- public double [Axel_hub::calcContrast](#) (double A)
Calculating contrast
- public double [Axel_hub::zeroFringe](#) ()
Calculating zeroFringe - similar to centreFring but woth phase shift compensation

- public Dictionary< string, double > [Axel_hub::deconstructAccel](#) (double accel, double mems)
Deconstructing accaleration to acceleration components - see dictionary keys
- public double [Axel_hub::nextShot](#) (int runID, double asymmetry, out double correction)
Calculated phaseCorr - corrected Raman phase (0 if not PID)
- public MMexec [Axel_hub::backMME](#) (int runID, double asymmetry, MMexec mme=null)
Prepare back message with new Raman phase value
- private void [Axel_hub::fillReport](#) (Dictionary< string, double > rpr)
Update table with strobes/PID calculation results
- public double [Axel_hub::PID](#) (double disbalance)
Calculating the phase correction from the disbalance on strobes Ys
- public void [Axel_hub::SaveConfigFile](#) ()
Save Config file in Config directory of Axel-hub
- public void [Axel_hub::OpenConfigFile](#) ()
Open Config file from Config directory of Axel-hub

Variables

- private MMexec [Axel_hub::grpMME](#)
- private MMexec [Axel_hub::lastMMEin](#)
- private MMexec [Axel_hub::lastMMEout](#)
- private bool [Axel_hub::_PID_Enabled](#)
- public bool [Axel_hub::PID_Enabled](#)
PID follow the strobe position
- private double [Axel_hub::lastContrast](#) = -1
- int [Axel_hub::runI](#) = 0
- string [Axel_hub::configFile](#)
- List< double > [Axel_hub::iStack](#)
- List< double > [Axel_hub::dStack](#)
- private FileLogger [Axel_hub::logger](#)
- public Point [Axel_hub::Down](#)
- public Point [Axel_hub::Up](#)
- public Point [Axel_hub::Low](#)
- string[] [Axel_hub::Titles](#) = { "runI", "tP", "tI", "tD", "Down.X", "Up.X", "disbal", "corr", "iSD-R", "contrast" }

Index

- [_PID_Enabled](#)
 - [Axel_hub, 24](#)
 - [_Running](#)
 - [Axel_hub, 25](#)
 - [_jumboButton](#)
 - [Axel_hub, 24](#)
- [accel](#)
 - [Axel_hub.accelCalibr, 33](#)
- [AcquireEvent](#)
 - [Axel_hub.AxelMems, 48](#)
- [AcquireHandler](#)
 - [Axel_hub.AxelMems, 48](#)
- [activeChannel](#)
 - [Axel_hub.AxelMems, 53](#)
- [ActiveRemote](#)
 - [Axel_hub, 12](#)
- [Add](#)
 - [Axel_hub.DataStack, 58](#)
 - [Axel_hub.ShotList, 93](#)
- [AddAxis](#)
 - [Axel_hub.AxelAxesClass, 38](#)
- [AddPoint](#)
 - [Axel_hub.DataStack, 58](#)
- [AddRange](#)
 - [Axel_hub.DataStack, 59](#)
- [AdjustTimelineToStopwatch](#)
 - [Axel_hub.AxelMems, 51](#)
- [archiScan](#)
 - [Axel_hub.ShotList, 93](#)
- [archiveMode](#)
 - [Axel_hub.ShotList, 95](#)
- [ArrangedPartner](#)
 - [Axel_hub, 25](#)
- [ask](#)
 - [OptionsNS.GeneralOptions, 73](#)
- [AsString](#)
 - [Axel_hub.SingleShot, 100](#)
- [AutoScaleBottom](#)
 - [OptionsNS.Modes, 81](#)
- [AutoScaleMiddle](#)
 - [OptionsNS.Modes, 81](#)
- [Axel_hub, 9](#)
 - [_PID_Enabled, 24](#)
 - [_Running, 25](#)
 - [_jumboButton, 24](#)
 - [ActiveRemote, 12](#)
 - [ArrangedPartner, 25](#)
 - [backMME, 12](#)
 - [calcContrast, 13](#)
 - [centreFringe, 13](#)
 - [configFile, 25](#)
 - [currentTime, 25](#)
 - [deconstructAccel, 13](#)
 - [dispatcherTimer_Tick, 15](#)
 - [Down, 25](#)
 - [dStack, 25](#)
 - [dTimer, 26](#)
 - [fillReport, 15](#)
 - [Flip, 15](#)
 - [genOptions, 26](#)
 - [GetBufferSize, 16](#)
 - [GetSamplingPeriod, 16](#)
 - [grpMME, 26](#)
 - [Image_MouseDown, 16](#)
 - [Init, 16](#)
 - [InitOptions, 17](#)
 - [iStack, 26](#)
 - [jumboButton, 26](#)
 - [lastContrast, 26](#)
 - [lastMMEin, 27](#)
 - [lastMMEout, 27](#)
 - [LogEvent, 17](#)
 - [logger, 27](#)
 - [LogHandler, 17](#)
 - [Low, 27](#)
 - [nextShot, 18](#)
 - [OnActiveComm, 18](#)
 - [OnActiveRemote, 27](#)
 - [OnAsyncSend, 18](#)
 - [OnJumboRepeat, 19](#)
 - [OnLog, 27](#)
 - [OnRealSampling, 19](#)
 - [OnReceive, 19](#)
 - [OnRemote, 28](#)
 - [OnRemoteMode, 28](#)
 - [OnStart, 28](#)
 - [OpenConfigFile, 20](#)
 - [PID, 20](#)
 - [PID_Enabled, 28](#)
 - [realSampling, 28](#)
 - [remote, 29](#)
 - [RemoteEvent, 20](#)
 - [RemoteHandler, 20](#)
 - [remoteMode, 29](#)
 - [RemoteModeEvent, 21](#)
 - [Reset, 21](#)
 - [runI, 29](#)
 - [Running, 29](#)

- SaveConfigFile, 21
- scanClass, 21
- scanModes, 29
- SendJson, 21
- set, 30
- SetActivity, 22
- SetFringeParams, 22
- SetSamplingRate, 22
- StartDelegate, 22
- StartEvent, 23
- StartHandler, 23
- Status, 23
- strokesUC, 23
- Titles, 30
- totalTime, 30
- Up, 30
- UpdateModes, 23
- UserControl_Loaded, 24
- zeroFringe, 24
- Axel_hub.accelCalibr, 33
 - accel, 33
 - cK0, 34
 - cK1, 34
 - model, 34
 - pK0, 34
 - pK1, 34
 - rAccel, 35
 - rTemper, 35
 - SN, 35
- Axel_hub.AxelAxesClass, 36
 - AddAxis, 38
 - AxelAxesClass, 37
 - axelMems, 43
 - byName, 38
 - Clear, 38
 - Closing, 39
 - DoAcquire, 39
 - DoAcquireTemperature, 40
 - DoJumboScan, 40
 - DoRemote, 40
 - jumboRepeat, 40
 - LogEvent, 41
 - LogHandler, 41
 - memsRunning, 43
 - OnLog, 44
 - prfIdx, 41
 - rCount, 44
 - SaveDefaultModes, 42
 - set2startADC24, 42
 - SetChartStrokes, 42
 - startADC, 42
 - UpdateFromOptions, 43
- Axel_hub.AxelMems, 46
 - AcquireEvent, 48
 - AcquireHandler, 48
 - activeChannel, 53
 - AdjustTimelineToStopwatch, 51
 - AxelMems, 48
 - byADCTimer, 47
 - byBoth, 47
 - byNone, 47
 - byStopwatch, 47
 - configureVITask, 48
 - FixConvRate, 51
 - hw, 52
 - isDevicePlugged, 49
 - memsX, 52
 - nSamples, 53
 - OnAcquire, 53
 - OnRealSampling, 53
 - rawData, 52
 - readBurst, 49
 - RealConvRate, 49
 - RealSamplingEvent, 50
 - RealSamplingHandler, 50
 - Reset, 50
 - running, 53
 - sampleRate, 53
 - SetStopwatch, 50
 - StartAcquisition, 50
 - StartStopwatch, 51
 - StopAcquisition, 51
 - TimeElapsed, 51
 - Timeout, 52
 - TimingMode, 52
 - TimingModes, 47
- Axel_hub.AxelMemsTemperature, 54
 - AxelMemsTemperature, 54
 - hw, 55
 - TakeTheTemperature, 54
- Axel_hub.DataStack, 55
 - Add, 58
 - AddPoint, 58
 - AddRange, 59
 - Clear, 59
 - Clone, 59
 - Compress, 60
 - CopyEach, 60
 - DataStack, 57
 - Depth, 69
 - ExportToArray, 60
 - fillSamples, 61
 - First, 69
 - Fit2Limit, 61
 - generalIdx, 68
 - ImportFromArray, 61
 - importFromArrays, 62
 - indexByX, 62
 - Last, 69
 - lastError, 69
 - logger, 68
 - maxDepth, 68
 - OnRefresh, 70
 - OpenPair, 62
 - pointSDev, 63
 - pointXs, 63

- pointYs, 63
- Portion, 64
- prefix, 69
- RefFileStats, 68
- RefreshEvent, 64
- RefreshHandler, 64
- rem, 70
- Rescale, 64
- Running, 70
- SavePair, 66
- StackMode, 70
- statsByIdx, 66
- statsByTime, 67
- stopWatch, 68
- TimePortion, 67
- TimeSeriesMode, 70
- visualCountLimit, 68
- Axel_hub.ShotList, 92
 - Add, 93
 - archiScan, 93
 - archiveMode, 95
 - conditions, 94
 - depth, 94
 - enabled, 95
 - FileCount, 95
 - filename, 95
 - lastIdx, 95
 - resetScan, 94
 - Save, 94
 - savingMode, 96
 - ShotList, 93
- Axel_hub.SingleShot, 96
 - AsString, 100
 - deconstructAccel, 98
 - idxByTime, 98
 - IsEmpty, 99
 - mems, 101
 - memsPortion, 99
 - memsWeightAccel, 100
 - precision, 100
 - quant, 100
 - SingleShot, 97, 98
- Axel_hub::App, 35
- Axel_hub::AxelAxisClass, 44
- Axel_hub::AxelChart, 45
- Axel_hub::AxelChartClass, 45
- Axel_hub::FringeParams, 71
 - offset, 71
 - period, 71
 - phase, 72
- Axel_hub::JoinOptimClass, 78
- Axel_hub::MainWindow, 79
- Axel_hub::Properties, 30
- Axel_hub::Properties::Resources, 87
- Axel_hub::Properties::Settings, 91
- Axel_hub::scanClass, 88
- Axel_hub::signalClass, 96
- Axel_hub::strobesUC, 101
- AxelAxesClass
 - Axel_hub.AxelAxesClass, 37
- AxelMems
 - Axel_hub.AxelMems, 48
- axelMems
 - Axel_hub.AxelAxesClass, 43
- AxelMemsTemperature
 - Axel_hub.AxelMemsTemperature, 54
- AxesChannels
 - OptionsNS.GeneralOptions, 74
- Background
 - OptionsNS.Modes, 81
- backMME
 - Axel_hub, 12
- byADCtimer
 - Axel_hub.AxelMems, 47
- byBoth
 - Axel_hub.AxelMems, 47
- byName
 - Axel_hub.AxelAxesClass, 38
- byNone
 - Axel_hub.AxelMems, 47
- byStopwatch
 - Axel_hub.AxelMems, 47
- calcContrast
 - Axel_hub, 13
- centreFringe
 - Axel_hub, 13
- ChartUpdate
 - OptionsNS.Modes, 81
- cK0
 - Axel_hub.accelCalibr, 34
- cK1
 - Axel_hub.accelCalibr, 34
- Clear
 - Axel_hub.AxelAxesClass, 38
 - Axel_hub.DataStack, 59
- Clone
 - Axel_hub.DataStack, 59
- Closing
 - Axel_hub.AxelAxesClass, 39
- Compress
 - Axel_hub.DataStack, 60
- conditions
 - Axel_hub.ShotList, 94
- configFile
 - Axel_hub, 25
- configureVITask
 - Axel_hub.AxelMems, 48
- CopyEach
 - Axel_hub.DataStack, 60
- currentTime
 - Axel_hub, 25
- DarkCurrent
 - OptionsNS.Modes, 81
- DataStack

- fillSamples
 - Axel_hub.DataStack, [61](#)
- First
 - Axel_hub.DataStack, [69](#)
- Fit2Limit
 - Axel_hub.DataStack, [61](#)
- FixConvRate
 - Axel_hub.AxelMems, [51](#)
- Flip
 - Axel_hub, [15](#)
- followPID
 - OptionsNS.GeneralOptions, [74](#)
- generalIdx
 - Axel_hub.DataStack, [68](#)
- genOptions
 - Axel_hub, [26](#)
- GetBufferSize
 - Axel_hub, [16](#)
- GetSamplingPeriod
 - Axel_hub, [16](#)
- grpMME
 - Axel_hub, [26](#)
- Height
 - OptionsNS.ScanModes, [89](#)
- hw
 - Axel_hub.AxelMems, [52](#)
 - Axel_hub.AxelMemsTemperature, [55](#)
- idxByTime
 - Axel_hub.SingleShot, [98](#)
- Image_MouseDown
 - Axel_hub, [16](#)
- ImportFromArray
 - Axel_hub.DataStack, [61](#)
- importFromArrays
 - Axel_hub.DataStack, [62](#)
- indexByX
 - Axel_hub.DataStack, [62](#)
- Init
 - Axel_hub, [16](#)
- InitOptions
 - Axel_hub, [17](#)
- intN2
 - OptionsNS.GeneralOptions, [74](#)
- isDevicePlugged
 - Axel_hub.AxelMems, [49](#)
- IsEmpty
 - Axel_hub.SingleShot, [99](#)
- iStack
 - Axel_hub, [26](#)
- JoinLog
 - OptionsNS.Modes, [82](#)
- Jumbo_Repeat
 - OptionsNS, [31](#)
- Jumbo_Scan
 - OptionsNS, [31](#)
- jumboButton
 - Axel_hub, [26](#)
- JumboBy
 - OptionsNS.Modes, [82](#)
- JumboCycles
 - OptionsNS.Modes, [82](#)
- JumboFrom
 - OptionsNS.Modes, [82](#)
- JumboRepeat
 - OptionsNS.GeneralOptions, [74](#)
- jumboRepeat
 - Axel_hub.AxelAxesClass, [40](#)
- JumboScan
 - OptionsNS.GeneralOptions, [75](#)
- JumboTo
 - OptionsNS.Modes, [82](#)
- Kcoeff
 - OptionsNS.Modes, [82](#)
- kD
 - OptionsNS.Modes, [83](#)
- kl
 - OptionsNS.Modes, [83](#)
- kP
 - OptionsNS.Modes, [83](#)
- Last
 - Axel_hub.DataStack, [69](#)
- lastContrast
 - Axel_hub, [26](#)
- lastError
 - Axel_hub.DataStack, [69](#)
- lastIdx
 - Axel_hub.ShotList, [95](#)
- lastMMEin
 - Axel_hub, [27](#)
- lastMMEout
 - Axel_hub, [27](#)
- Left
 - OptionsNS.ScanModes, [90](#)
- LogEvent
 - Axel_hub, [17](#)
 - Axel_hub.AxelAxesClass, [41](#)
- LogFilePrec
 - OptionsNS.GeneralOptions, [75](#)
- logger
 - Axel_hub, [27](#)
 - Axel_hub.DataStack, [68](#)
- LogHandler
 - Axel_hub, [17](#)
 - Axel_hub.AxelAxesClass, [41](#)
- Low
 - Axel_hub, [27](#)
- maxDepth
 - Axel_hub.DataStack, [68](#)
- mems
 - Axel_hub.SingleShot, [101](#)
- Mems2SignDelay

- OptionsNS.GeneralOptions, 75
- Mems2SignLen
 - OptionsNS.GeneralOptions, 75
- MemsEnabled
 - OptionsNS.Modes, 83
- MemsHw
 - OptionsNS.GeneralOptions, 75
- MemsInJumbo
 - OptionsNS.GeneralOptions, 75
- memsPortion
 - Axel_hub.SingleShot, 99
- memsRunning
 - Axel_hub.AxelAxesClass, 43
- memsWeightAccel
 - Axel_hub.SingleShot, 100
- memsX
 - Axel_hub.AxelMems, 52
- MiddleFrame
 - OptionsNS.Modes, 83
- model
 - Axel_hub.accelCalibr, 34
- N1
 - OptionsNS.Modes, 83
- N2
 - OptionsNS.Modes, 84
- nextShot
 - Axel_hub, 18
- nosave
 - OptionsNS.GeneralOptions, 73
- nSamples
 - Axel_hub.AxelMems, 53
- Ntot
 - OptionsNS.Modes, 84
- offset
 - Axel_hub::FringeParams, 71
 - OptionsNS.Modes, 84
- OnAcquire
 - Axel_hub.AxelMems, 53
- OnActiveComm
 - Axel_hub, 18
- OnActiveRemote
 - Axel_hub, 27
- OnAsyncSend
 - Axel_hub, 18
- OnJumboRepeat
 - Axel_hub, 19
- OnLog
 - Axel_hub, 27
 - Axel_hub.AxelAxesClass, 44
- OnRealSampling
 - Axel_hub, 19
 - Axel_hub.AxelMems, 53
- OnReceive
 - Axel_hub, 19
- OnRefresh
 - Axel_hub.DataStack, 70
- OnRemote
 - Axel_hub, 28
- OnRemoteMode
 - Axel_hub, 28
- OnStart
 - Axel_hub, 28
- OpenConfigFile
 - Axel_hub, 20
- OpenPair
 - Axel_hub.DataStack, 62
- OptionsNS, 31
 - Disconnected, 31
 - Jumbo_Repeat, 31
 - Jumbo_Scan, 31
 - Ready_To_Remote, 31
 - RemoteMode, 31
 - Simple_Repeat, 31
 - Simple_Scan, 31
- OptionsNS.GeneralOptions, 72
 - ask, 73
 - AxesChannels, 74
 - followPID, 74
 - intN2, 74
 - JumboRepeat, 74
 - JumboScan, 75
 - LogFilePrec, 75
 - Mems2SignDelay, 75
 - Mems2SignLen, 75
 - MemsHw, 75
 - MemsInJumbo, 75
 - nosave, 73
 - RawSignalAvg, 76
 - Save, 73
 - save, 73
 - SaveFilePrec, 76
 - SaveModes, 73
 - saveModes, 74
 - saveVisuals, 76
 - ShowMemsIfRunning, 76
 - SignalCursorPrec, 76
 - SignalTablePrec, 76
 - TemperatureCompensation, 77
 - TemperatureEnabled, 77
 - TemperatureHw, 77
 - TrendSignalLen, 77
- OptionsNS.Modes, 79
 - AutoScaleBottom, 81
 - AutoScaleMiddle, 81
 - Background, 81
 - ChartUpdate, 81
 - DarkCurrent, 81
 - DoubleStrobe, 81
 - JoinLog, 82
 - JumboBy, 82
 - JumboCycles, 82
 - JumboFrom, 82
 - JumboTo, 82
 - Kcoeff, 82
 - kD, 83

- kl, [83](#)
- kP, [83](#)
- MemsEnabled, [83](#)
- MiddleFrame, [83](#)
- N1, [83](#)
- N2, [84](#)
- Ntot, [84](#)
- offset, [84](#)
- phi0, [84](#)
- PID_Enabled, [84](#)
- PowerCoeff, [84](#)
- RN1, [85](#)
- RN2, [85](#)
- RollMean, [85](#)
- RsltChrtUpdate, [85](#)
- RsltTblUpdate, [85](#)
- Save, [80](#)
- scale, [85](#)
- ShowFreq, [86](#)
- SignalLog, [86](#)
- StackDepth, [86](#)
- StdDev, [86](#)
- TblUpdate, [86](#)
- TopFrame, [86](#)
- TopOfTopFrame, [87](#)
- OptionsNS.ScanModes, [88](#)
 - Height, [89](#)
 - Left, [90](#)
 - remoteMode, [89](#)
 - SamplingFreq, [90](#)
 - Save, [89](#)
 - SizeLimit, [90](#)
 - SizeLimitMode, [90](#)
 - TimeLimit, [90](#)
 - TimeLimitMode, [90](#)
 - Top, [91](#)
 - Width, [91](#)
- OptionsNS::OptionsWindow, [87](#)
- period
 - Axel_hub::FringeParams, [71](#)
- phase
 - Axel_hub::FringeParams, [72](#)
- phi0
 - OptionsNS.Modes, [84](#)
- PID
 - Axel_hub, [20](#)
- PID_Enabled
 - Axel_hub, [28](#)
 - OptionsNS.Modes, [84](#)
- pK0
 - Axel_hub.accelCalibr, [34](#)
- pK1
 - Axel_hub.accelCalibr, [34](#)
- pointSDev
 - Axel_hub.DataStack, [63](#)
- pointXs
 - Axel_hub.DataStack, [63](#)
- pointYs
 - Axel_hub.DataStack, [63](#)
- Portion
 - Axel_hub.DataStack, [64](#)
- PowerCoeff
 - OptionsNS.Modes, [84](#)
- precision
 - Axel_hub.SingleShot, [100](#)
- prefix
 - Axel_hub.DataStack, [69](#)
- prfIdx
 - Axel_hub.AxelAxesClass, [41](#)
- quant
 - Axel_hub.SingleShot, [100](#)
- rAccel
 - Axel_hub.accelCalibr, [35](#)
- rawData
 - Axel_hub.AxelMems, [52](#)
- RawSignalAvg
 - OptionsNS.GeneralOptions, [76](#)
- rCount
 - Axel_hub.AxelAxesClass, [44](#)
- readBurst
 - Axel_hub.AxelMems, [49](#)
- Ready_To_Remote
 - OptionsNS, [31](#)
- RealConvRate
 - Axel_hub.AxelMems, [49](#)
- realSampling
 - Axel_hub, [28](#)
- RealSamplingEvent
 - Axel_hub.AxelMems, [50](#)
- RealSamplingHandler
 - Axel_hub.AxelMems, [50](#)
- RefFileStats
 - Axel_hub.DataStack, [68](#)
- RefreshEvent
 - Axel_hub.DataStack, [64](#)
- RefreshHandler
 - Axel_hub.DataStack, [64](#)
- rem
 - Axel_hub.DataStack, [70](#)
- remote
 - Axel_hub, [29](#)
- RemoteEvent
 - Axel_hub, [20](#)
- RemoteHandler
 - Axel_hub, [20](#)
- RemoteMode
 - OptionsNS, [31](#)
- remoteMode
 - Axel_hub, [29](#)
 - OptionsNS.ScanModes, [89](#)
- RemoteModeEvent
 - Axel_hub, [21](#)
- Rescale
 - Axel_hub.DataStack, [64](#)
- Reset

- Axel_hub, 21
- Axel_hub.AxelMems, 50
- resetScan
 - Axel_hub.ShotList, 94
- RN1
 - OptionsNS.Modes, 85
- RN2
 - OptionsNS.Modes, 85
- RollMean
 - OptionsNS.Modes, 85
- RsltChrtUpdate
 - OptionsNS.Modes, 85
- RsltTblUpdate
 - OptionsNS.Modes, 85
- rTemper
 - Axel_hub.accelCalibr, 35
- runI
 - Axel_hub, 29
- Running
 - Axel_hub, 29
 - Axel_hub.DataStack, 70
- running
 - Axel_hub.AxelMems, 53
- sampleRate
 - Axel_hub.AxelMems, 53
- SamplingFreq
 - OptionsNS.ScanModes, 90
- Save
 - Axel_hub.ShotList, 94
 - OptionsNS.GeneralOptions, 73
 - OptionsNS.Modes, 80
 - OptionsNS.ScanModes, 89
- save
 - OptionsNS.GeneralOptions, 73
- SaveConfigFile
 - Axel_hub, 21
- SaveDefaultModes
 - Axel_hub.AxelAxesClass, 42
- SaveFilePrec
 - OptionsNS.GeneralOptions, 76
- SaveModes
 - OptionsNS.GeneralOptions, 73
- saveModes
 - OptionsNS.GeneralOptions, 74
- SavePair
 - Axel_hub.DataStack, 66
- saveVisuals
 - OptionsNS.GeneralOptions, 76
- savingMode
 - Axel_hub.ShotList, 96
- scale
 - OptionsNS.Modes, 85
- scanClass
 - Axel_hub, 21
- scanModes
 - Axel_hub, 29
- SendJson
 - Axel_hub, 21
- set
 - Axel_hub, 30
- set2startADC24
 - Axel_hub.AxelAxesClass, 42
- SetActivity
 - Axel_hub, 22
- SetChartStrobes
 - Axel_hub.AxelAxesClass, 42
- SetFringeParams
 - Axel_hub, 22
- SetSamplingRate
 - Axel_hub, 22
- SetStopwatch
 - Axel_hub.AxelMems, 50
- ShotList
 - Axel_hub.ShotList, 93
- ShowFreq
 - OptionsNS.Modes, 86
- ShowMemsIfRunning
 - OptionsNS.GeneralOptions, 76
- SignalCursorPrec
 - OptionsNS.GeneralOptions, 76
- SignalLog
 - OptionsNS.Modes, 86
- SignalTablePrec
 - OptionsNS.GeneralOptions, 76
- Simple_Repeat
 - OptionsNS, 31
- Simple_Scan
 - OptionsNS, 31
- SingleShot
 - Axel_hub.SingleShot, 97, 98
- SizeLimit
 - OptionsNS.ScanModes, 90
- SizeLimitMode
 - OptionsNS.ScanModes, 90
- SN
 - Axel_hub.accelCalibr, 35
- StackDepth
 - OptionsNS.Modes, 86
- StackMode
 - Axel_hub.DataStack, 70
- StartAcquisition
 - Axel_hub.AxelMems, 50
- startADC
 - Axel_hub.AxelAxesClass, 42
- StartDelegate
 - Axel_hub, 22
- StartEvent
 - Axel_hub, 23
- StartHandler
 - Axel_hub, 23
- StartStopwatch
 - Axel_hub.AxelMems, 51
- statsByIdx
 - Axel_hub.DataStack, 66
- statsByTime
 - Axel_hub.DataStack, 67

- Status
 - Axel_hub, [23](#)
- StdDev
 - OptionsNS.Modes, [86](#)
- StopAcquisition
 - Axel_hub.AxelMems, [51](#)
- stopWatch
 - Axel_hub.DataStack, [68](#)
- strobesUC
 - Axel_hub, [23](#)
- TakeTheTemperature
 - Axel_hub.AxelMemsTemperature, [54](#)
- TblUpdate
 - OptionsNS.Modes, [86](#)
- TemperatureCompensation
 - OptionsNS.GeneralOptions, [77](#)
- TemperatureEnabled
 - OptionsNS.GeneralOptions, [77](#)
- TemperatureHw
 - OptionsNS.GeneralOptions, [77](#)
- TimeElapsed
 - Axel_hub.AxelMems, [51](#)
- TimeLimit
 - OptionsNS.ScanModes, [90](#)
- TimeLimitMode
 - OptionsNS.ScanModes, [90](#)
- Timeout
 - Axel_hub.AxelMems, [52](#)
- TimePortion
 - Axel_hub.DataStack, [67](#)
- TimeSeriesMode
 - Axel_hub.DataStack, [70](#)
- TimingMode
 - Axel_hub.AxelMems, [52](#)
- TimingModes
 - Axel_hub.AxelMems, [47](#)
- Titles
 - Axel_hub, [30](#)
- Top
 - OptionsNS.ScanModes, [91](#)
- TopFrame
 - OptionsNS.Modes, [86](#)
- TopOfTopFrame
 - OptionsNS.Modes, [87](#)
- totalTime
 - Axel_hub, [30](#)
- TrendSignalLen
 - OptionsNS.GeneralOptions, [77](#)
- Up
 - Axel_hub, [30](#)
- UpdateFromOptions
 - Axel_hub.AxelAxesClass, [43](#)
- UpdateModes
 - Axel_hub, [23](#)
- UserControl_Loaded
 - Axel_hub, [24](#)
- visualCountLimit
 - Axel_hub.DataStack, [68](#)
- Width
 - OptionsNS.ScanModes, [91](#)
- XamlGeneratedNamespace, [31](#)
- XamlGeneratedNamespace::GeneratedInternalTypeHelper,
[78](#)
- zeroFringe
 - Axel_hub, [24](#)