[XPS Unified Command Interface 10](#_Toc425177522)

[GetVerCommand 10](#_Toc425177523)

[SetGantryMode 10](#_Toc425177524)

[GetGantryMode 11](#_Toc425177525)

[InitializeAndHomeXY 11](#_Toc425177526)

[InitializeAndHomeY 12](#_Toc425177527)

[InitializeAndHomeX 12](#_Toc425177528)

[MoveAbsolute 12](#_Toc425177529)

[MoveSlice 13](#_Toc425177530)

[WaitMotionEnd 13](#_Toc425177531)

[AbortMove 14](#_Toc425177532)

[GetAccParams 14](#_Toc425177533)

[SetAccParams 15](#_Toc425177534)

[GetVelParams 16](#_Toc425177535)

[SetVelParams 16](#_Toc425177536)

[GetXVelParams 17](#_Toc425177537)

[SetXVelParams 17](#_Toc425177538)

[GetYVelParams 18](#_Toc425177539)

[SetYVelParams 18](#_Toc425177540)

[SetJogVelocity 19](#_Toc425177541)

[GetJogVelocity 19](#_Toc425177542)

[SetJogAcceleration 20](#_Toc425177543)

[GetJogAcceleration 20](#_Toc425177544)

[StartJog 21](#_Toc425177545)

[EndJog 21](#_Toc425177546)

[GetCurrentPosition 22](#_Toc425177547)

[SetZone 22](#_Toc425177548)

[GetZone 23](#_Toc425177549)

[SetVarX 23](#_Toc425177550)

[SetVarXSecondary 24](#_Toc425177551)

[SetVarY 24](#_Toc425177552)

[GetVarX 25](#_Toc425177553)

[GetVarXSecondary 25](#_Toc425177554)

[GetVarY 26](#_Toc425177555)

[SetPiston 26](#_Toc425177556)

[GetPistonState 27](#_Toc425177557)

[SetBrake 27](#_Toc425177558)

[GetBrakeState 28](#_Toc425177559)

[ZygoStartBoardP2 28](#_Toc425177560)

[ZygoPositionGet 29](#_Toc425177561)

[ZygoErrorStatusStringGet 29](#_Toc425177562)

[ZygoErrorStatusGet 30](#_Toc425177563)

[ZygoStatusStringGet 30](#_Toc425177564)

[ZygoStatusGet 31](#_Toc425177565)

[ZygoGetVerInterfero 31](#_Toc425177566)

[ZygoResetX 32](#_Toc425177567)

[ZygoResetY 32](#_Toc425177568)

[ZygoSetOffsetX 32](#_Toc425177569)

[ZygoSetOffsetY 33](#_Toc425177570)

[ZygoSetPEGParams 33](#_Toc425177571)

[ZygoGetPEGLastCommunicationTime 34](#_Toc425177572)

[FirmwareVersionGet 34](#_Toc425177573)

[InstallerVersionGet 35](#_Toc425177574)

[Reboot 35](#_Toc425177575)

[RestartApplication 36](#_Toc425177576)

[CIEHeaderGet 36](#_Toc425177577)

[CIEReset 37](#_Toc425177578)

[ControllerMotionKernelTimeLoadGet 37](#_Toc425177579)

[ControllerRTTimeGet 38](#_Toc425177580)

[ControllerSlaveStatusGet 38](#_Toc425177581)

[ControllerSlaveStatusStringGet 39](#_Toc425177582)

[ControllerSynchronizeCorrectorISR 39](#_Toc425177583)

[ControllerStatusGet 40](#_Toc425177584)

[ControllerStatusRead 40](#_Toc425177585)

[ControllerStatusStringGet 41](#_Toc425177586)

[ElapsedTimeGet 41](#_Toc425177587)

[ErrorStringGet 42](#_Toc425177588)

[Login 42](#_Toc425177589)

[CloseAllOtherSockets 43](#_Toc425177590)

[HardwareDriverAndStageGet 43](#_Toc425177591)

[HardwareDateAndTimeGet 44](#_Toc425177592)

[HardwareDateAndTimeSet 44](#_Toc425177593)

[FileScriptHistoryRename 45](#_Toc425177594)

[FileGatheringRename 45](#_Toc425177595)

[INTServitudesStatusGet 45](#_Toc425177596)

[INTServitudesCommandGet 46](#_Toc425177597)

[TCLScriptExecute 46](#_Toc425177598)

[TCLScriptExecuteAndWait 47](#_Toc425177599)

[TCLScriptExecuteWithPriority 47](#_Toc425177600)

[TCLScriptKill 48](#_Toc425177601)

[TCLScriptKillAll 48](#_Toc425177602)

[TCLScriptRunningListGet 49](#_Toc425177603)

[TimerGet 49](#_Toc425177604)

[TimerSet 50](#_Toc425177605)

[GlobalArrayGet 50](#_Toc425177606)

[GlobalArraySet 51](#_Toc425177607)

[DoubleGlobalArrayGet 51](#_Toc425177608)

[DoubleGlobalArraySet 52](#_Toc425177609)

[PositionerMagneticTrackPositionAtHomeGet 52](#_Toc425177610)

[ZygoConnectToServer 53](#_Toc425177611)

[ZygoReset 53](#_Toc425177612)

[ZygoReadWord 54](#_Toc425177613)

[ZygoReadLong 54](#_Toc425177614)

[ZygoWriteWord 55](#_Toc425177615)

[ZygoWriteLong 55](#_Toc425177616)

[ZygoSendAndReceive 56](#_Toc425177617)

[ZygoDisconnectFromServer 56](#_Toc425177618)

[ZygoRegisterSet 57](#_Toc425177619)

[ZygoRegisterGet 57](#_Toc425177620)

[EventAdd 58](#_Toc425177621)

[EventGet 58](#_Toc425177622)

[EventRemove 59](#_Toc425177623)

[EventWait 59](#_Toc425177624)

[EventExtendedConfigurationTriggerSet 60](#_Toc425177625)

[EventExtendedConfigurationTriggerGet 60](#_Toc425177626)

[EventExtendedConfigurationActionSet 61](#_Toc425177627)

[EventExtendedConfigurationActionGet 61](#_Toc425177628)

[EventExtendedStart 62](#_Toc425177629)

[EventExtendedAllGet 62](#_Toc425177630)

[EventExtendedGet 63](#_Toc425177631)

[EventExtendedRemove 63](#_Toc425177632)

[EventExtendedWait 64](#_Toc425177633)

[GatheringConfigurationGet 64](#_Toc425177634)

[GatheringConfigurationSet 65](#_Toc425177635)

[GatheringCurrentNumberGet 65](#_Toc425177636)

[GatheringStopAndSave 66](#_Toc425177637)

[GatheringDataAcquire 66](#_Toc425177638)

[GatheringDataGet 67](#_Toc425177639)

[GatheringDataMultipleLinesGet 67](#_Toc425177640)

[GatheringReset 68](#_Toc425177641)

[GatheringRun 68](#_Toc425177642)

[GatheringRunAppend 69](#_Toc425177643)

[GatheringStop 69](#_Toc425177644)

[GatheringExternalConfigurationSet 70](#_Toc425177645)

[GatheringExternalConfigurationGet 70](#_Toc425177646)

[GatheringExternalCurrentNumberGet 71](#_Toc425177647)

[GatheringExternalDataGet 71](#_Toc425177648)

[GatheringExternalStopAndSave 72](#_Toc425177649)

[GPIOAnalogGet 72](#_Toc425177650)

[GPIOAnalogSet 73](#_Toc425177651)

[GPIOAnalogGainGet 73](#_Toc425177652)

[GPIOAnalogGainSet 74](#_Toc425177653)

[GPIOAnalogConfigurationGet 74](#_Toc425177654)

[GPIOAnalogConfigurationSet 75](#_Toc425177655)

[GPIODigitalGet 75](#_Toc425177656)

[GPIODigitalSet 76](#_Toc425177657)

[KillAll 76](#_Toc425177658)

[GroupAccelerationSetpointGet 77](#_Toc425177659)

[GroupAnalogTrackingModeEnable 77](#_Toc425177660)

[GroupAnalogTrackingModeDisable 78](#_Toc425177661)

[GroupCorrectorOutputGet 78](#_Toc425177662)

[GroupCurrentFollowingErrorGet 79](#_Toc425177663)

[GroupHomeSearch 79](#_Toc425177664)

[GroupHomeSearchAndRelativeMove 80](#_Toc425177665)

[GroupInitialize 80](#_Toc425177666)

[GroupInitializeNoEncoderReset 81](#_Toc425177667)

[GroupInitializeWithEncoderCalibration 81](#_Toc425177668)

[GroupInterlockDisable 82](#_Toc425177669)

[GroupInterlockEnable 82](#_Toc425177670)

[GroupJogParametersSet 83](#_Toc425177671)

[GroupJogParametersGet 83](#_Toc425177672)

[GroupJogCurrentGet 84](#_Toc425177673)

[GroupJogModeEnable 84](#_Toc425177674)

[GroupJogModeDisable 85](#_Toc425177675)

[GroupKill 85](#_Toc425177676)

[GroupMotionDisable 86](#_Toc425177677)

[GroupMotionEnable 86](#_Toc425177678)

[GroupMotionStatusGet 86](#_Toc425177679)

[GroupMoveAbort 87](#_Toc425177680)

[GroupMoveAbortFast 87](#_Toc425177681)

[GroupMoveAbsolute 88](#_Toc425177682)

[GroupMoveRelative 88](#_Toc425177683)

[GroupPositionCorrectedProfilerGet 89](#_Toc425177684)

[GroupPositionCurrentGet 90](#_Toc425177685)

[GroupPositionPCORawEncoderGet 90](#_Toc425177686)

[GroupPositionSetpointGet 91](#_Toc425177687)

[GroupPositionTargetGet 91](#_Toc425177688)

[GroupReferencingActionExecute 92](#_Toc425177689)

[GroupReferencingStart 92](#_Toc425177690)

[GroupReferencingStop 93](#_Toc425177691)

[GroupStatusGet 93](#_Toc425177692)

[GroupStatusStringGet 94](#_Toc425177693)

[GroupVelocityCurrentGet 94](#_Toc425177694)

[PositionerDriverStatusGet 95](#_Toc425177695)

[PositionerDriverStatusStringGet 95](#_Toc425177696)

[PositionerErrorGet 96](#_Toc425177697)

[PositionerErrorRead 96](#_Toc425177698)

[PositionerErrorStringGet 97](#_Toc425177699)

[PositionerHardwareStatusGet 97](#_Toc425177700)

[PositionerHardwareStatusStringGet 98](#_Toc425177701)

[PositionersEncoderIndexDifferenceGet 98](#_Toc425177702)

[PositionerGantryEndReferencingPositionGet 99](#_Toc425177703)

[PositionerStageParameterGet 99](#_Toc425177704)

[PositionerStageParameterSet 100](#_Toc425177705)

[PositionerCorrectorTypeGet 100](#_Toc425177706)

[PositionerWarningFollowingErrorSet 101](#_Toc425177707)

[PositionerWarningFollowingErrorGet 101](#_Toc425177708)

[PositionerCompensationPositionFilterSet 102](#_Toc425177709)

[PositionerCompensationPositionFilterGet 102](#_Toc425177710)

[PositionerCorrectorAutoTuning 103](#_Toc425177711)

[PositionerAccelerationAutoScaling 103](#_Toc425177712)

[PositionerExcitationSignalGet 104](#_Toc425177713)

[PositionerExcitationSignalSet 104](#_Toc425177714)

[PositionerCurrentVelocityAccelerationFiltersSet 105](#_Toc425177715)

[PositionerCurrentVelocityAccelerationFiltersGet 105](#_Toc425177716)

[PositionerEncoderAmplitudeValuesGet 106](#_Toc425177717)

[PositionerEncoderCalibrationParametersGet 107](#_Toc425177718)

[PositionerRawEncoderPositionGet 107](#_Toc425177719)

[PositionerBacklashSet 108](#_Toc425177720)

[PositionerBacklashGet 108](#_Toc425177721)

[PositionerBacklashEnable 109](#_Toc425177722)

[PositionerBacklashDisable 109](#_Toc425177723)

[PositionerPreCorrectorExcitationSignalGet 110](#_Toc425177724)

[PositionerPreCorrectorExcitationSignalSet 110](#_Toc425177725)

[PositionerMotionDoneGet 111](#_Toc425177726)

[PositionerMotionDoneSet 111](#_Toc425177727)

[PositionerHardInterpolatorFactorGet 112](#_Toc425177728)

[PositionerHardInterpolatorFactorSet 112](#_Toc425177729)

[PositionerHardInterpolatorPositionGet 113](#_Toc425177730)

[PositionerDriverFiltersGet 113](#_Toc425177731)

[PositionerDriverFiltersSet 114](#_Toc425177732)

[PositionerDriverPositionOffsetsGet 115](#_Toc425177733)

[PositionerPositionCompareGet 115](#_Toc425177734)

[PositionerPositionCompareSet 116](#_Toc425177735)

[PositionerPositionCompareEnable 116](#_Toc425177736)

[PositionerPositionCompareDisable 117](#_Toc425177737)

[PositionerPositionComparePulseParametersGet 117](#_Toc425177738)

[PositionerPositionComparePulseParametersSet 118](#_Toc425177739)

[PositionerCompensatedFastPCOAbort 118](#_Toc425177740)

[PositionerCompensatedFastPCOCurrentStatusGet 119](#_Toc425177741)

[PositionerCompensatedFastPCOEnable 119](#_Toc425177742)

[PositionerCompensatedFastPCOFromFile 120](#_Toc425177743)

[PositionerCompensatedFastPCOLoadToMemory 120](#_Toc425177744)

[PositionerCompensatedFastPCOMemoryReset 121](#_Toc425177745)

[PositionerCompensatedFastPCOPrepare 121](#_Toc425177746)

[PositionerCompensatedFastPCOSet 122](#_Toc425177747)

[PositionerCompensatedFastPCOPulseParametersGet 122](#_Toc425177748)

[PositionerCompensatedFastPCOPulseParametersSet 123](#_Toc425177749)

[PositionerCorrectorPIDFFAccelerationSet 123](#_Toc425177750)

[PositionerCorrectorPIDFFAccelerationGet 124](#_Toc425177751)

[PositionerCorrectorPIDAccelerationFilterSet 125](#_Toc425177752)

[PositionerCorrectorPIDAccelerationFilterGet 126](#_Toc425177753)

[PositionerCorrectorDamperFilterSet 126](#_Toc425177754)

[PositionerCorrectorDamperFilterGet 127](#_Toc425177755)

[PositionerCorrectorPostFFSet 127](#_Toc425177756)

[PositionerCorrectorPostFFGet 128](#_Toc425177757)

[PositionerCorrectorPIDBaseSet 128](#_Toc425177758)

[PositionerCorrectorPIDBaseGet 129](#_Toc425177759)

[PositionerPositionCompareScanAccelerationLimitGet 129](#_Toc425177760)

[PositionerPositionCompareScanAccelerationLimitSet 130](#_Toc425177761)

[PositionerCorrectorP2IDFFAccelerationSet 130](#_Toc425177762)

[PositionerCorrectorP2IDFFAccelerationGet 131](#_Toc425177763)

[PositionerCorrectorPIDDualFFVoltageSet 132](#_Toc425177764)

[PositionerCorrectorPIDDualFFVoltageGet 133](#_Toc425177765)

[PositionerCorrectorPIDFFVelocitySet 134](#_Toc425177766)

[PositionerCorrectorPIDFFVelocityGet 135](#_Toc425177767)

[PositionerCorrectorPIPositionSet 135](#_Toc425177768)

[PositionerCorrectorPIPositionGet 136](#_Toc425177769)

[PositionerCorrectorSR1AccelerationSet 136](#_Toc425177770)

[PositionerCorrectorSR1AccelerationGet 137](#_Toc425177771)

[PositionerCorrectorSR1ObserverAccelerationSet 138](#_Toc425177772)

[PositionerCorrectorSR1ObserverAccelerationGet 138](#_Toc425177773)

[PositionerCorrectorSR1OffsetAccelerationSet 139](#_Toc425177774)

[PositionerCorrectorSR1OffsetAccelerationGet 139](#_Toc425177775)

[PositionerCorrectorDualSet 140](#_Toc425177776)

[PositionerCorrectorDualGet 141](#_Toc425177777)

[PositionerCompensationDualLoopNotchFilterSet 141](#_Toc425177778)

[PositionerCompensationDualLoopNotchFilterGet 142](#_Toc425177779)

[PositionerCompensationDualLoopPhaseCorrectionFilterSet 142](#_Toc425177780)

[PositionerCompensationDualLoopPhaseCorrectionFilterGet 143](#_Toc425177781)

[PositionerCompensationNotchFilterSet 144](#_Toc425177782)

[PositionerCompensationNotchFilterGet 144](#_Toc425177783)

[PositionerCompensationPhaseCorrectionFilterSet 145](#_Toc425177784)

[PositionerCompensationPhaseCorrectionFilterGet 145](#_Toc425177785)

[PositionerCorrectorNotchFiltersSet 146](#_Toc425177786)

[PositionerCorrectorNotchFiltersGet 147](#_Toc425177787)

[PositionerCompensationPreFeedForwardFrequencyNotchFilterGet 147](#_Toc425177788)

[PositionerCompensationPreFeedForwardFrequencyNotchFilterSet 148](#_Toc425177789)

[PositionerCompensationPreFeedForwardSpatialNotchFilterGet 148](#_Toc425177790)

[PositionerCompensationPreFeedForwardSpatialNotchFilterSet 149](#_Toc425177791)

[PositionerCompensationPreFeedForwardPhaseCorrectionFilterGet 150](#_Toc425177792)

[PositionerCompensationPreFeedForwardPhaseCorrectionFilterSet 150](#_Toc425177793)

[PositionerCompensationFrequencyNotchsGet 151](#_Toc425177794)

[PositionerCompensationFrequencyNotchsSet 152](#_Toc425177795)

[PositionerCompensationSpatialPeriodicNotchsGet 152](#_Toc425177796)

[PositionerCompensationSpatialPeriodicNotchsSet 153](#_Toc425177797)

[PositionerCompensationPostExcitationLowPassFilterGet 154](#_Toc425177798)

[PositionerCompensationPostExcitationLowPassFilterSet 154](#_Toc425177799)

[PositionerCompensationPostExcitationFrequencyNotchFilterGet 155](#_Toc425177800)

[PositionerCompensationPostExcitationFrequencyNotchFilterSet 155](#_Toc425177801)

[PositionerCompensationPostExcitationNotchModeFilterGet 156](#_Toc425177802)

[PositionerCompensationPostExcitationNotchModeFilterSet 157](#_Toc425177803)

[PositionerCompensationPostExcitationPhaseCorrectionFilterGet 157](#_Toc425177804)

[PositionerCompensationPostExcitationPhaseCorrectionFilterSet 158](#_Toc425177805)

[PositionerCompensationLowPassTwoFilterGet 159](#_Toc425177806)

[PositionerCompensationLowPassTwoFilterSet 159](#_Toc425177807)

[PositionerCompensationNotchModeFiltersGet 160](#_Toc425177808)

[PositionerCompensationNotchModeFiltersSet 160](#_Toc425177809)

[PositionerCompensationPhaseCorrectionFiltersGet 161](#_Toc425177810)

[PositionerCompensationPhaseCorrectionFiltersSet 162](#_Toc425177811)

[PositionerAnalogTrackingPositionParametersGet 162](#_Toc425177812)

[PositionerAnalogTrackingPositionParametersSet 163](#_Toc425177813)

[PositionerAnalogTrackingVelocityParametersGet 163](#_Toc425177814)

[PositionerAnalogTrackingVelocityParametersSet 164](#_Toc425177815)

[PositionerJogMaximumVelocityAndAccelerationGet 165](#_Toc425177816)

[PositionerMaximumVelocityAndAccelerationGet 165](#_Toc425177817)

[PositionerUserTravelLimitsGet 166](#_Toc425177818)

[PositionerUserTravelLimitsSet 166](#_Toc425177819)

[PositionerSGammaExactVelocityAjustedDisplacementGet 167](#_Toc425177820)

[PositionerSGammaParametersGet 167](#_Toc425177821)

[PositionerSGammaParametersSet 168](#_Toc425177822)

[PositionerSGammaVelocityAndAccelerationSet 169](#_Toc425177823)

[PositionerSGammaPreviousMotionTimesGet 169](#_Toc425177824)

[MultipleAxesPVTVerification 170](#_Toc425177825)

[MultipleAxesPVTVerificationResultGet 170](#_Toc425177826)

[MultipleAxesPVTExecution 171](#_Toc425177827)

[MultipleAxesPVTParametersGet 171](#_Toc425177828)

[MultipleAxesPVTPulseOutputSet 172](#_Toc425177829)

[MultipleAxesPVTPulseOutputGet 172](#_Toc425177830)

[MultipleAxesPVTLoadToMemory 173](#_Toc425177831)

[MultipleAxesPVTResetInMemory 173](#_Toc425177832)

[XYPVTVerification 174](#_Toc425177833)

[XYPVTVerificationResultGet 174](#_Toc425177834)

[XYPVTExecution 175](#_Toc425177835)

[XYPVTParametersGet 175](#_Toc425177836)

[XYPVTPulseOutputSet 176](#_Toc425177837)

[XYPVTPulseOutputGet 177](#_Toc425177838)

[XYPVTLoadToMemory 177](#_Toc425177839)

[XYPVTResetInMemory 178](#_Toc425177840)

[TZPVTVerification 178](#_Toc425177841)

[TZPVTVerificationResultGet 179](#_Toc425177842)

[TZPVTExecution 179](#_Toc425177843)

[TZPVTParametersGet 180](#_Toc425177844)

[TZPVTPulseOutputSet 180](#_Toc425177845)

[TZPVTPulseOutputGet 181](#_Toc425177846)

[TZPVTLoadToMemory 181](#_Toc425177847)

[TZPVTResetInMemory 182](#_Toc425177848)

[XYLineArcVerification 182](#_Toc425177849)

[XYLineArcVerificationResultGet 183](#_Toc425177850)

[XYLineArcExecution 183](#_Toc425177851)

[XYLineArcParametersGet 184](#_Toc425177852)

[XYLineArcPulseOutputSet 184](#_Toc425177853)

[XYLineArcPulseOutputGet 185](#_Toc425177854)

[XYZSplineVerification 185](#_Toc425177855)

[XYZSplineVerificationResultGet 186](#_Toc425177856)

[XYZSplineExecution 187](#_Toc425177857)

[XYZSplineParametersGet 187](#_Toc425177858)

[SingleAxisSlaveModeEnable 188](#_Toc425177859)

[SingleAxisSlaveModeDisable 188](#_Toc425177860)

[SingleAxisSlaveParametersSet 189](#_Toc425177861)

[SingleAxisSlaveParametersGet 189](#_Toc425177862)

[SingleAxisThetaClampDisable 190](#_Toc425177863)

[SingleAxisThetaClampEnable 190](#_Toc425177864)

[SingleAxisThetaFeedforwardParametersGet 191](#_Toc425177865)

[SingleAxisThetaFeedforwardParametersSet 191](#_Toc425177866)

[SingleAxisThetaSlaveModeEnable 192](#_Toc425177867)

[SingleAxisThetaSlaveModeDisable 192](#_Toc425177868)

[SingleAxisThetaSlaveParametersGet 193](#_Toc425177869)

[SingleAxisThetaSlaveParametersSet 193](#_Toc425177870)

[SpindleSlaveModeEnable 194](#_Toc425177871)

[SpindleSlaveModeDisable 194](#_Toc425177872)

[SpindleSlaveParametersSet 194](#_Toc425177873)

[SpindleSlaveParametersGet 195](#_Toc425177874)

[GroupSpinParametersSet 195](#_Toc425177875)

[GroupSpinParametersGet 196](#_Toc425177876)

[GroupSpinCurrentGet 196](#_Toc425177877)

[GroupSpinModeStop 197](#_Toc425177878)

[PositionerFeedforwardAccSet 197](#_Toc425177879)

[PositionerFeedforwardAccGet 198](#_Toc425177880)

[PositionerFeedforwardAccEnable 199](#_Toc425177881)

[PositionerFeedforwardAccDisable 199](#_Toc425177882)

[PositionerFeedforwardPositionSet 200](#_Toc425177883)

[PositionerFeedforwardPositionGet 200](#_Toc425177884)

[PositionerFeedforwardPositionEnable 201](#_Toc425177885)

[PositionerFeedforwardPositionDisable 201](#_Toc425177886)

[XYZGroupPositionCorrectedProfilerGet 202](#_Toc425177887)

[XYZGroupPositionPCORawEncoderGet 202](#_Toc425177888)

[TZMotorDecouplingMatrixGet 203](#_Toc425177889)

[TZMotorDecouplingMatrixSet 204](#_Toc425177890)

[TZFocusModeEnable 204](#_Toc425177891)

[TZFocusModeDisable 205](#_Toc425177892)

[TZTrackingUserMaximumZZZTargetDifferenceGet 205](#_Toc425177893)

[TZTrackingUserMaximumZZZTargetDifferenceSet 206](#_Toc425177894)

[FocusProcessSocketReserve 206](#_Toc425177895)

[FocusProcessSocketFree 207](#_Toc425177896)

[LoginCheck 207](#_Toc425177897)

[GroupAllPositionTrace 208](#_Toc425177898)

[GroupMotorMatrixTrace 208](#_Toc425177899)

[GroupMotorMatrixInverseTrace 209](#_Toc425177900)

[GroupPositionCurrentRawGet 209](#_Toc425177901)

[PositionerMotorOutputOffsetGet 210](#_Toc425177902)

[PositionerMotorOutputOffsetSet 210](#_Toc425177903)

[SingleAxisThetaPositionRawGet 211](#_Toc425177904)

[EEPROMCIESet 211](#_Toc425177905)

[EEPROMDACOffsetCIESet 212](#_Toc425177906)

[EEPROMDriverSet 212](#_Toc425177907)

[EEPROMINTSet 213](#_Toc425177908)

[CPUCoreAndBoardSupplyVoltagesGet 213](#_Toc425177909)

[CPUTemperatureAndFanSpeedGet 214](#_Toc425177910)

[ActionListGet 215](#_Toc425177911)

[ActionExtendedListGet 215](#_Toc425177912)

[APIExtendedListGet 216](#_Toc425177913)

[APIListGet 216](#_Toc425177914)

[APIListStandardGet 217](#_Toc425177915)

[APIListAMATGet 217](#_Toc425177916)

[ControllerStatusListGet 218](#_Toc425177917)

[ErrorListGet 218](#_Toc425177918)

[EventListGet 219](#_Toc425177919)

[GatheringListGet 219](#_Toc425177920)

[GatheringExtendedListGet 220](#_Toc425177921)

[GatheringExternalListGet 220](#_Toc425177922)

[GroupStatusListGet 221](#_Toc425177923)

[HardwareInternalListGet 221](#_Toc425177924)

[ObjectsListGet 222](#_Toc425177925)

[PositionerErrorListGet 222](#_Toc425177926)

[PositionerHardwareStatusListGet 223](#_Toc425177927)

[PositionerDriverStatusListGet 223](#_Toc425177928)

[ReferencingActionListGet 224](#_Toc425177929)

[ReferencingSensorListGet 224](#_Toc425177930)

[SystemIniParameterGet 225](#_Toc425177931)

[SystemIniParameterSet 225](#_Toc425177932)

[SystemRefParameterGet 226](#_Toc425177933)

[SystemRefParameterSet 226](#_Toc425177934)

[GatheringUserDatasGet 227](#_Toc425177935)

[ControllerMotionKernelMinMaxTimeLoadGet 227](#_Toc425177936)

[ControllerMotionKernelMinMaxTimeLoadReset 228](#_Toc425177937)

[ControllerMotionKernelPeriodMinMaxGet 229](#_Toc425177938)

[ControllerMotionKernelPeriodMinMaxReset 229](#_Toc425177939)

[SocketsStatusGet 230](#_Toc425177940)

[TestTCP 230](#_Toc425177941)

[ISRCorrectorCompensateOverrunNumberGet 231](#_Toc425177942)

[ISRCorrectorCompensateOverrunNumberReset 231](#_Toc425177943)

[OptionalModuleExecute 232](#_Toc425177944)

[OptionalModuleKill 232](#_Toc425177945)

[CIERegister32ValueGet 233](#_Toc425177946)

[CIERegister64ValueGet 233](#_Toc425177947)

[CIERegisterValueSet 234](#_Toc425177948)

[DebugTraceCommunicationReset 234](#_Toc425177949)

[DebugTraceCommunicationSave 235](#_Toc425177950)

[RunTraceloggerProcessWithTimeSetting 235](#_Toc425177951)

[RunTraceloggerProcessWithRollingBuffer 236](#_Toc425177952)

[CreateQNXEvent 236](#_Toc425177953)

[StartEventsAcqusition 237](#_Toc425177954)

[StopEventsAcqusition 237](#_Toc425177955)

[EventTriggerSet 237](#_Toc425177956)

[DebugCorrectorTimeUsageGet 238](#_Toc425177957)

[DebugProfilerTimeUsageGet 239](#_Toc425177958)

[DebugServitudesTimeUsageGet 239](#_Toc425177959)

## XPS Unified Command Interface

### GetVerCommand

#### Syntax

##### C# prototype

int GetVerCommand(out string Version, out string errstring)

##### Python prototype

[Version, errstring] GetVerCommand ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Version: Version

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetVerCommand command which is used to Return firmware version. Refer to the XPS Programmer's manual to get the command description.

### SetGantryMode

#### Syntax

##### C# prototype

int SetGantryMode(string Option, out string errstring)

##### Python prototype

[errstring] SetGantryMode (Option)

#### Parameters

##### Input parameters

(string) Option: Option

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetGantryMode command which is used to Set Gantry mode . Refer to the XPS Programmer's manual to get the command description.

### GetGantryMode

#### Syntax

##### C# prototype

int GetGantryMode(out string Option, out string errstring)

##### Python prototype

[Option, errstring] GetGantryMode ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Option: Option

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetGantryMode command which is used to Set Gantry mode . Refer to the XPS Programmer's manual to get the command description.

### InitializeAndHomeXY

#### Syntax

##### C# prototype

int InitializeAndHomeXY(string Option, out string errstring)

##### Python prototype

[errstring] InitializeAndHomeXY (Option)

#### Parameters

##### Input parameters

(string) Option: Option

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous InitializeAndHomeXY command which is used to Do a Home search on X and after Y axis. Refer to the XPS Programmer's manual to get the command description.

### InitializeAndHomeY

#### Syntax

##### C# prototype

int InitializeAndHomeY( out string errstring)

##### Python prototype

[errstring] InitializeAndHomeY ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous InitializeAndHomeY command which is used to Do a Home search on Y axis. It's always the scale. Refer to the XPS Programmer's manual to get the command description.

### InitializeAndHomeX

#### Syntax

##### C# prototype

int InitializeAndHomeX(string Option, out string errstring)

##### Python prototype

[errstring] InitializeAndHomeX (Option)

#### Parameters

##### Input parameters

(string) Option: Option

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous InitializeAndHomeX command which is used to Do a Home search on X axis. Refer to the XPS Programmer's manual to get the command description.

### MoveAbsolute

#### Syntax

##### C# prototype

int MoveAbsolute(double PositonAbsoluteX\_um, double PositionAbsoluteY\_um, out string errstring)

##### Python prototype

[errstring] MoveAbsolute (PositonAbsoluteX\_um, PositionAbsoluteY\_um)

#### Parameters

##### Input parameters

(double) PositonAbsoluteX\_um: PositonAbsoluteX\_um

(double) PositionAbsoluteY\_um: PositionAbsoluteY\_um

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MoveAbsolute command which is used to Moves the stage to the end position. The positions are defined in um. Refer to the XPS Programmer's manual to get the command description.

### MoveSlice

#### Syntax

##### C# prototype

int MoveSlice(double y\_end\_um, double x\_end\_um, double scan\_angle\_urad, out string errstring)

##### Python prototype

[errstring] MoveSlice (y\_end\_um, x\_end\_um, scan\_angle\_urad)

#### Parameters

##### Input parameters

(double) y\_end\_um: y\_end\_um

(double) x\_end\_um: x\_end\_um

(double) scan\_angle\_urad: scan\_angle\_urad

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MoveSlice command which is used to Moves the stage to the end position. Begin a linear interpolation to move to . Refer to the XPS Programmer's manual to get the command description.

### WaitMotionEnd

#### Syntax

##### C# prototype

int WaitMotionEnd(double time\_out\_ms, double y\_end\_um, double x\_end\_um, out string errstring)

##### Python prototype

[errstring] WaitMotionEnd (time\_out\_ms, y\_end\_um, x\_end\_um)

#### Parameters

##### Input parameters

(double) time\_out\_ms: time\_out\_ms

(double) y\_end\_um: y\_end\_um

(double) x\_end\_um: x\_end\_um

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous WaitMotionEnd command which is used to Waits the end of motion and checks the target positions. Refer to the XPS Programmer's manual to get the command description.

### AbortMove

#### Syntax

##### C# prototype

int AbortMove( out string errstring)

##### Python prototype

[errstring] AbortMove ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous AbortMove command which is used to Abort motion. Refer to the XPS Programmer's manual to get the command description.

### GetAccParams

#### Syntax

##### C# prototype

int GetAccParams(out int x\_acc\_time\_ms, out int x\_smooth\_factor\_ms, out int y\_acc\_time\_ms, out int y\_smooth\_factor\_ms, out string errstring)

##### Python prototype

[x\_acc\_time\_ms, x\_smooth\_factor\_ms, y\_acc\_time\_ms, y\_smooth\_factor\_ms, errstring] GetAccParams ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) x\_acc\_time\_ms: x\_acc\_time\_ms

(int) x\_smooth\_factor\_ms: x\_smooth\_factor\_ms

(int) y\_acc\_time\_ms: y\_acc\_time\_ms

(int) y\_smooth\_factor\_ms: y\_smooth\_factor\_ms

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetAccParams command which is used to Read acceleration parameters. Smooth factor = jerk time and all parameters unit in msec. Refer to the XPS Programmer's manual to get the command description.

### SetAccParams

#### Syntax

##### C# prototype

int SetAccParams(int x\_acc\_time\_ms, int x\_smooth\_factor\_ms, int y\_acc\_time\_ms, int y\_smooth\_factor\_ms, out string errstring)

##### Python prototype

[errstring] SetAccParams (x\_acc\_time\_ms, x\_smooth\_factor\_ms, y\_acc\_time\_ms, y\_smooth\_factor\_ms)

#### Parameters

##### Input parameters

(int) x\_acc\_time\_ms: x\_acc\_time\_ms

(int) x\_smooth\_factor\_ms: x\_smooth\_factor\_ms

(int) y\_acc\_time\_ms: y\_acc\_time\_ms

(int) y\_smooth\_factor\_ms: y\_smooth\_factor\_ms

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetAccParams command which is used to Update acceleration parameters. Smooth factor = jerk time and all parameters unit in msec. Refer to the XPS Programmer's manual to get the command description.

### GetVelParams

#### Syntax

##### C# prototype

int GetVelParams(out double vel\_x, out double vel\_y, out string errstring)

##### Python prototype

[vel\_x, vel\_y, errstring] GetVelParams ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) vel\_x: vel\_x

(double) vel\_y: vel\_y

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetVelParams command which is used to Read velocity parameters. All parameters unit in microns/sec. Refer to the XPS Programmer's manual to get the command description.

### SetVelParams

#### Syntax

##### C# prototype

int SetVelParams(double vel\_x, double vel\_y, out string errstring)

##### Python prototype

[errstring] SetVelParams (vel\_x, vel\_y)

#### Parameters

##### Input parameters

(double) vel\_x: vel\_x

(double) vel\_y: vel\_y

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetVelParams command which is used to Update velocity parameters. All parameters unit in microns/sec. Refer to the XPS Programmer's manual to get the command description.

### GetXVelParams

#### Syntax

##### C# prototype

int GetXVelParams(out double vel\_x, out string errstring)

##### Python prototype

[vel\_x, errstring] GetXVelParams ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) vel\_x: vel\_x

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetXVelParams command which is used to Read velocity parameter for X axis. Parameter unit in microns/sec. Refer to the XPS Programmer's manual to get the command description.

### SetXVelParams

#### Syntax

##### C# prototype

int SetXVelParams(double vel\_x, out string errstring)

##### Python prototype

[errstring] SetXVelParams (vel\_x)

#### Parameters

##### Input parameters

(double) vel\_x: vel\_x

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetXVelParams command which is used to Update velocity parameter for X axis. Parameter unit in microns/sec. Refer to the XPS Programmer's manual to get the command description.

### GetYVelParams

#### Syntax

##### C# prototype

int GetYVelParams(out double vel\_y, out string errstring)

##### Python prototype

[vel\_y, errstring] GetYVelParams ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) vel\_y: vel\_y

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetYVelParams command which is used to Read velocity parameter for Y axis. Parameter unit in microns/sec. Refer to the XPS Programmer's manual to get the command description.

### SetYVelParams

#### Syntax

##### C# prototype

int SetYVelParams(double vel\_y, out string errstring)

##### Python prototype

[errstring] SetYVelParams (vel\_y)

#### Parameters

##### Input parameters

(double) vel\_y: vel\_y

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetYVelParams command which is used to Update velocity parameter for Y axis. Parameter unit in microns/sec. Refer to the XPS Programmer's manual to get the command description.

### SetJogVelocity

#### Syntax

##### C# prototype

int SetJogVelocity(double vx, double vy, int joystickAck\_Timeout\_ms, out string errstring)

##### Python prototype

[errstring] SetJogVelocity (vx, vy, joystickAck\_Timeout\_ms)

#### Parameters

##### Input parameters

(double) vx: vx

(double) vy: vy

(int) joystickAck\_Timeout\_ms: joystickAck\_Timeout\_ms

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetJogVelocity command which is used to Modify Jog velocity. Refer to the XPS Programmer's manual to get the command description.

### GetJogVelocity

#### Syntax

##### C# prototype

int GetJogVelocity(out double vx, out double vy, out int joystickAck\_Timeout\_ms, out string errstring)

##### Python prototype

[vx, vy, joystickAck\_Timeout\_ms, errstring] GetJogVelocity ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) vx: vx

(double) vy: vy

(int) joystickAck\_Timeout\_ms: joystickAck\_Timeout\_ms

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetJogVelocity command which is used to Get Jog velocity. Refer to the XPS Programmer's manual to get the command description.

### SetJogAcceleration

#### Syntax

##### C# prototype

int SetJogAcceleration(int x\_acceleration\_Time\_ms, int x\_smooth\_factor\_ms, int y\_acceleration\_Time\_ms, int y\_smooth\_factor\_ms, out string errstring)

##### Python prototype

[errstring] SetJogAcceleration (x\_acceleration\_Time\_ms, x\_smooth\_factor\_ms, y\_acceleration\_Time\_ms, y\_smooth\_factor\_ms)

#### Parameters

##### Input parameters

(int) x\_acceleration\_Time\_ms: x\_acceleration\_Time\_ms

(int) x\_smooth\_factor\_ms: x\_smooth\_factor\_ms

(int) y\_acceleration\_Time\_ms: y\_acceleration\_Time\_ms

(int) y\_smooth\_factor\_ms: y\_smooth\_factor\_ms

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetJogAcceleration command which is used to Modify Jog acceleration. Refer to the XPS Programmer's manual to get the command description.

### GetJogAcceleration

#### Syntax

##### C# prototype

int GetJogAcceleration(out int x\_acceleration\_Time\_ms, out int x\_smooth\_factor\_ms, out int y\_acceleration\_Time\_ms, out int y\_smooth\_factor\_ms, out string errstring)

##### Python prototype

[x\_acceleration\_Time\_ms, x\_smooth\_factor\_ms, y\_acceleration\_Time\_ms, y\_smooth\_factor\_ms, errstring] GetJogAcceleration ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) x\_acceleration\_Time\_ms: x\_acceleration\_Time\_ms

(int) x\_smooth\_factor\_ms: x\_smooth\_factor\_ms

(int) y\_acceleration\_Time\_ms: y\_acceleration\_Time\_ms

(int) y\_smooth\_factor\_ms: y\_smooth\_factor\_ms

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetJogAcceleration command which is used to Get Jog acceleration. Refer to the XPS Programmer's manual to get the command description.

### StartJog

#### Syntax

##### C# prototype

int StartJog( out string errstring)

##### Python prototype

[errstring] StartJog ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous StartJog command which is used to Enable Jog mode. Refer to the XPS Programmer's manual to get the command description.

### EndJog

#### Syntax

##### C# prototype

int EndJog( out string errstring)

##### Python prototype

[errstring] EndJog ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EndJog command which is used to Disable Jog mode. Refer to the XPS Programmer's manual to get the command description.

### GetCurrentPosition

#### Syntax

##### C# prototype

int GetCurrentPosition(out double y\_position\_um, out double x\_position\_um, out int x1\_position\_cnts, out int x2\_position\_cnts, out int y\_laser\_position\_cnts, out int x\_laser\_position\_cnts, out string errstring)

##### Python prototype

[y\_position\_um, x\_position\_um, x1\_position\_cnts, x2\_position\_cnts, y\_laser\_position\_cnts, x\_laser\_position\_cnts, errstring] GetCurrentPosition ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) y\_position\_um: y\_position\_um

(double) x\_position\_um: x\_position\_um

(int) x1\_position\_cnts: x1\_position\_cnts

(int) x2\_position\_cnts: x2\_position\_cnts

(int) y\_laser\_position\_cnts: y\_laser\_position\_cnts

(int) x\_laser\_position\_cnts: x\_laser\_position\_cnts

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetCurrentPosition command which is used to Read current positions. Refer to the XPS Programmer's manual to get the command description.

### SetZone

#### Syntax

##### C# prototype

int SetZone(double x\_center\_um, double y\_center\_um, double radius\_um, double hysteresis\_um, out string errstring)

##### Python prototype

[errstring] SetZone (x\_center\_um, y\_center\_um, radius\_um, hysteresis\_um)

#### Parameters

##### Input parameters

(double) x\_center\_um: x\_center\_um

(double) y\_center\_um: y\_center\_um

(double) radius\_um: radius\_um

(double) hysteresis\_um: hysteresis\_um

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetZone command which is used to Set Zone - output signal when inside a circle. Refer to the XPS Programmer's manual to get the command description.

### GetZone

#### Syntax

##### C# prototype

int GetZone(out double x\_center\_um, out double y\_center\_um, out double radius\_um, out double hysteresis\_um, out string errstring)

##### Python prototype

[x\_center\_um, y\_center\_um, radius\_um, hysteresis\_um, errstring] GetZone ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) x\_center\_um: x\_center\_um

(double) y\_center\_um: y\_center\_um

(double) radius\_um: radius\_um

(double) hysteresis\_um: hysteresis\_um

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetZone command which is used to Get Zone parameters. Refer to the XPS Programmer's manual to get the command description.

### SetVarX

#### Syntax

##### C# prototype

int SetVarX(string NameParameter, double value, out string errstring)

##### Python prototype

[errstring] SetVarX (NameParameter, value)

#### Parameters

##### Input parameters

(string) NameParameter: NameParameter

(double) value: value

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetVarX command which is used to Set var parameter for X. Refer to the XPS Programmer's manual to get the command description.

### SetVarXSecondary

#### Syntax

##### C# prototype

int SetVarXSecondary(string NameParameter, double value, out string errstring)

##### Python prototype

[errstring] SetVarXSecondary (NameParameter, value)

#### Parameters

##### Input parameters

(string) NameParameter: NameParameter

(double) value: value

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetVarXSecondary command which is used to Set var parameter X secondary. Refer to the XPS Programmer's manual to get the command description.

### SetVarY

#### Syntax

##### C# prototype

int SetVarY(string NameParameter, double value, out string errstring)

##### Python prototype

[errstring] SetVarY (NameParameter, value)

#### Parameters

##### Input parameters

(string) NameParameter: NameParameter

(double) value: value

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetVarY command which is used to Set var parameter for Y. Refer to the XPS Programmer's manual to get the command description.

### GetVarX

#### Syntax

##### C# prototype

int GetVarX(string NameParameter, out double value, out string errstring)

##### Python prototype

[value, errstring] GetVarX (NameParameter)

#### Parameters

##### Input parameters

(string) NameParameter: NameParameter

##### Output parameters

(double) value: value

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetVarX command which is used to Get var parameter for X. Refer to the XPS Programmer's manual to get the command description.

### GetVarXSecondary

#### Syntax

##### C# prototype

int GetVarXSecondary(string NameParameter, out double value, out string errstring)

##### Python prototype

[value, errstring] GetVarXSecondary (NameParameter)

#### Parameters

##### Input parameters

(string) NameParameter: NameParameter

##### Output parameters

(double) value: value

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetVarXSecondary command which is used to Get var parameter X secondary. Refer to the XPS Programmer's manual to get the command description.

### GetVarY

#### Syntax

##### C# prototype

int GetVarY(string NameParameter, out double value, out string errstring)

##### Python prototype

[value, errstring] GetVarY (NameParameter)

#### Parameters

##### Input parameters

(string) NameParameter: NameParameter

##### Output parameters

(double) value: value

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetVarY command which is used to Get var parameter for Y. Refer to the XPS Programmer's manual to get the command description.

### SetPiston

#### Syntax

##### C# prototype

int SetPiston(int Command, out string errstring)

##### Python prototype

[errstring] SetPiston (Command)

#### Parameters

##### Input parameters

(int) Command: Command

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetPiston command which is used to Set Piston command . Refer to the XPS Programmer's manual to get the command description.

### GetPistonState

#### Syntax

##### C# prototype

int GetPistonState(out int CommandState, out int isUp, out int isDown, out int LiftPinUPInterlock, out string errstring)

##### Python prototype

[CommandState, isUp, isDown, LiftPinUPInterlock, errstring] GetPistonState ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) CommandState: CommandState

(int) isUp: isUp

(int) isDown: isDown

(int) LiftPinUPInterlock: LiftPinUPInterlock

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetPistonState command which is used to Get Piston State. Refer to the XPS Programmer's manual to get the command description.

### SetBrake

#### Syntax

##### C# prototype

int SetBrake(int Command, out string errstring)

##### Python prototype

[errstring] SetBrake (Command)

#### Parameters

##### Input parameters

(int) Command: Command

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SetBrake command which is used to Set Brake command . Refer to the XPS Programmer's manual to get the command description.

### GetBrakeState

#### Syntax

##### C# prototype

int GetBrakeState(out int CommandState, out string errstring)

##### Python prototype

[CommandState, errstring] GetBrakeState ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) CommandState: CommandState

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GetBrakeState command which is used to Get Brake State . Refer to the XPS Programmer's manual to get the command description.

### ZygoStartBoardP2

#### Syntax

##### C# prototype

int ZygoStartBoardP2(int Kv, int Kp, bool ReverseDirectionSense, int DataAgeAdjust, out string errstring)

##### Python prototype

[errstring] ZygoStartBoardP2 (Kv, Kp, ReverseDirectionSense, DataAgeAdjust)

#### Parameters

##### Input parameters

(int) Kv: Kv

(int) Kp: Kp

(bool) ReverseDirectionSense: ReverseDirectionSense

(int) DataAgeAdjust: DataAgeAdjust

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoStartBoardP2 command which is used to Get the Zygo Board P2 up. Refer to the XPS Programmer's manual to get the command description.

### ZygoPositionGet

#### Syntax

##### C# prototype

int ZygoPositionGet(out long PositionY, out long PositionX, out string errstring)

##### Python prototype

[PositionY, PositionX, errstring] ZygoPositionGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(long) PositionY: PositionY

(long) PositionX: PositionX

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoPositionGet command which is used to Return Zygo Interferometer positions. Refer to the XPS Programmer's manual to get the command description.

### ZygoErrorStatusStringGet

#### Syntax

##### C# prototype

int ZygoErrorStatusStringGet(int axis, out string ErrorStatus, out string errstring)

##### Python prototype

[ErrorStatus, errstring] ZygoErrorStatusStringGet (axis)

#### Parameters

##### Input parameters

(int) axis: axis

##### Output parameters

(string) ErrorStatus: ErrorStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoErrorStatusStringGet command which is used to Return Zygo Interferometer Error Status description. Refer to the XPS Programmer's manual to get the command description.

### ZygoErrorStatusGet

#### Syntax

##### C# prototype

int ZygoErrorStatusGet(int axis, out string ErrorStatus, out string errstring)

##### Python prototype

[ErrorStatus, errstring] ZygoErrorStatusGet (axis)

#### Parameters

##### Input parameters

(int) axis: axis

##### Output parameters

(string) ErrorStatus: ErrorStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoErrorStatusGet command which is used to Return Zygo Interferometer Error Status code. Refer to the XPS Programmer's manual to get the command description.

### ZygoStatusStringGet

#### Syntax

##### C# prototype

int ZygoStatusStringGet(int axis, out string ErrorStatus, out string errstring)

##### Python prototype

[ErrorStatus, errstring] ZygoStatusStringGet (axis)

#### Parameters

##### Input parameters

(int) axis: axis

##### Output parameters

(string) ErrorStatus: ErrorStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoStatusStringGet command which is used to Return Zygo Interferometer Status description. Refer to the XPS Programmer's manual to get the command description.

### ZygoStatusGet

#### Syntax

##### C# prototype

int ZygoStatusGet(int axis, out string ErrorStatus, out string errstring)

##### Python prototype

[ErrorStatus, errstring] ZygoStatusGet (axis)

#### Parameters

##### Input parameters

(int) axis: axis

##### Output parameters

(string) ErrorStatus: ErrorStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoStatusGet command which is used to Return Zygo Interferometer Status code. Refer to the XPS Programmer's manual to get the command description.

### ZygoGetVerInterfero

#### Syntax

##### C# prototype

int ZygoGetVerInterfero(out string Version, out string errstring)

##### Python prototype

[Version, errstring] ZygoGetVerInterfero ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Version: Version

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoGetVerInterfero command which is used to Return Interferometer firmware version. Refer to the XPS Programmer's manual to get the command description.

### ZygoResetX

#### Syntax

##### C# prototype

int ZygoResetX( out string errstring)

##### Python prototype

[errstring] ZygoResetX ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoResetX command which is used to Reset interferometer axis relates to axis X. Refer to the XPS Programmer's manual to get the command description.

### ZygoResetY

#### Syntax

##### C# prototype

int ZygoResetY( out string errstring)

##### Python prototype

[errstring] ZygoResetY ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoResetY command which is used to Reset interferometer axis relates to axis Y. Refer to the XPS Programmer's manual to get the command description.

### ZygoSetOffsetX

#### Syntax

##### C# prototype

int ZygoSetOffsetX(long offset\_x, out string errstring)

##### Python prototype

[errstring] ZygoSetOffsetX (offset\_x)

#### Parameters

##### Input parameters

(long) offset\_x: offset\_x

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoSetOffsetX command which is used to Write the offset register of interferometer relates to axis X. Refer to the XPS Programmer's manual to get the command description.

### ZygoSetOffsetY

#### Syntax

##### C# prototype

int ZygoSetOffsetY(long offset\_y, out string errstring)

##### Python prototype

[errstring] ZygoSetOffsetY (offset\_y)

#### Parameters

##### Input parameters

(long) offset\_y: offset\_y

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoSetOffsetY command which is used to Write the offset register of interferometer relates to axis Y. Refer to the XPS Programmer's manual to get the command description.

### ZygoSetPEGParams

#### Syntax

##### C# prototype

int ZygoSetPEGParams(long P1, long P2, uint delta1, long K1, uint delta2, long K2, int ControlWord, out string errstring)

##### Python prototype

[errstring] ZygoSetPEGParams (P1, P2, delta1, K1, delta2, K2, ControlWord)

#### Parameters

##### Input parameters

(long) P1: P1

(long) P2: P2

(uint) delta1: delta1

(long) K1: K1

(uint) delta2: delta2

(long) K2: K2

(int) ControlWord: ControlWord

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoSetPEGParams command which is used to Write these parameters from Zygo interferometer. Refer to the XPS Programmer's manual to get the command description.

### ZygoGetPEGLastCommunicationTime

#### Syntax

##### C# prototype

int ZygoGetPEGLastCommunicationTime(out double LastCommunicationTime, out string errstring)

##### Python prototype

[LastCommunicationTime, errstring] ZygoGetPEGLastCommunicationTime ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) LastCommunicationTime: LastCommunicationTime

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoGetPEGLastCommunicationTime command which is used to Get the last communication time to configure Zygo PEG. Refer to the XPS Programmer's manual to get the command description.

### FirmwareVersionGet

#### Syntax

##### C# prototype

int FirmwareVersionGet(out string Version, out string errstring)

##### Python prototype

[Version, errstring] FirmwareVersionGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Version: Version

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous FirmwareVersionGet command which is used to Return firmware version. Refer to the XPS Programmer's manual to get the command description.

### InstallerVersionGet

#### Syntax

##### C# prototype

int InstallerVersionGet(out string Version, out string errstring)

##### Python prototype

[Version, errstring] InstallerVersionGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Version: Version

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous InstallerVersionGet command which is used to Return installer pack version. Refer to the XPS Programmer's manual to get the command description.

### Reboot

#### Syntax

##### C# prototype

int Reboot( out string errstring)

##### Python prototype

[errstring] Reboot ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous Reboot command which is used to Reboot the controller. Refer to the XPS Programmer's manual to get the command description.

### RestartApplication

#### Syntax

##### C# prototype

int RestartApplication( out string errstring)

##### Python prototype

[errstring] RestartApplication ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous RestartApplication command which is used to Restart the Controller. Refer to the XPS Programmer's manual to get the command description.

### CIEHeaderGet

#### Syntax

##### C# prototype

int CIEHeaderGet(int CIEBoardIndex, out string HeaderString, out string errstring)

##### Python prototype

[HeaderString, errstring] CIEHeaderGet (CIEBoardIndex)

#### Parameters

##### Input parameters

(int) CIEBoardIndex: CIEBoardIndex

##### Output parameters

(string) HeaderString: HeaderString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CIEHeaderGet command which is used to Return the CIE board header string. Refer to the XPS Programmer's manual to get the command description.

### CIEReset

#### Syntax

##### C# prototype

int CIEReset( out string errstring)

##### Python prototype

[errstring] CIEReset ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CIEReset command which is used to Stop and reset all CIE boards. Refer to the XPS Programmer's manual to get the command description.

### ControllerMotionKernelTimeLoadGet

#### Syntax

##### C# prototype

int ControllerMotionKernelTimeLoadGet(out double CPUTotalLoadRatio, out double CPUCorrectorLoadRatio, out double CPUProfilerLoadRatio, out double CPUServitudesLoadRatio, out string errstring)

##### Python prototype

[CPUTotalLoadRatio, CPUCorrectorLoadRatio, CPUProfilerLoadRatio, CPUServitudesLoadRatio, errstring] ControllerMotionKernelTimeLoadGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) CPUTotalLoadRatio: CPUTotalLoadRatio

(double) CPUCorrectorLoadRatio: CPUCorrectorLoadRatio

(double) CPUProfilerLoadRatio: CPUProfilerLoadRatio

(double) CPUServitudesLoadRatio: CPUServitudesLoadRatio

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerMotionKernelTimeLoadGet command which is used to Get controller motion kernel time load. Refer to the XPS Programmer's manual to get the command description.

### ControllerRTTimeGet

#### Syntax

##### C# prototype

int ControllerRTTimeGet(out double CurrentRTPeriod, out double CurrentRTUsage, out string errstring)

##### Python prototype

[CurrentRTPeriod, CurrentRTUsage, errstring] ControllerRTTimeGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) CurrentRTPeriod: CurrentRTPeriod

(double) CurrentRTUsage: CurrentRTUsage

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerRTTimeGet command which is used to Get controller corrector period and calculation time. Refer to the XPS Programmer's manual to get the command description.

### ControllerSlaveStatusGet

#### Syntax

##### C# prototype

int ControllerSlaveStatusGet(out int SlaveControllerStatus, out string errstring)

##### Python prototype

[SlaveControllerStatus, errstring] ControllerSlaveStatusGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) SlaveControllerStatus: SlaveControllerStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerSlaveStatusGet command which is used to Read slave controller status. Refer to the XPS Programmer's manual to get the command description.

### ControllerSlaveStatusStringGet

#### Syntax

##### C# prototype

int ControllerSlaveStatusStringGet(int SlaveControllerStatusCode, out string SlaveControllerStatusString, out string errstring)

##### Python prototype

[SlaveControllerStatusString, errstring] ControllerSlaveStatusStringGet (SlaveControllerStatusCode)

#### Parameters

##### Input parameters

(int) SlaveControllerStatusCode: SlaveControllerStatusCode

##### Output parameters

(string) SlaveControllerStatusString: SlaveControllerStatusString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerSlaveStatusStringGet command which is used to Return the slave controller status string. Refer to the XPS Programmer's manual to get the command description.

### ControllerSynchronizeCorrectorISR

#### Syntax

##### C# prototype

int ControllerSynchronizeCorrectorISR(string ModeString, out string errstring)

##### Python prototype

[errstring] ControllerSynchronizeCorrectorISR (ModeString)

#### Parameters

##### Input parameters

(string) ModeString: ModeString

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerSynchronizeCorrectorISR command which is used to Synchronize controller corrector ISR. Refer to the XPS Programmer's manual to get the command description.

### ControllerStatusGet

#### Syntax

##### C# prototype

int ControllerStatusGet(out int ControllerStatus, out string errstring)

##### Python prototype

[ControllerStatus, errstring] ControllerStatusGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) ControllerStatus: ControllerStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerStatusGet command which is used to Get controller current status and reset the status. Refer to the XPS Programmer's manual to get the command description.

### ControllerStatusRead

#### Syntax

##### C# prototype

int ControllerStatusRead(out int ControllerStatus, out string errstring)

##### Python prototype

[ControllerStatus, errstring] ControllerStatusRead ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) ControllerStatus: ControllerStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerStatusRead command which is used to Read controller current status. Refer to the XPS Programmer's manual to get the command description.

### ControllerStatusStringGet

#### Syntax

##### C# prototype

int ControllerStatusStringGet(int ControllerStatusCode, out string ControllerStatusString, out string errstring)

##### Python prototype

[ControllerStatusString, errstring] ControllerStatusStringGet (ControllerStatusCode)

#### Parameters

##### Input parameters

(int) ControllerStatusCode: ControllerStatusCode

##### Output parameters

(string) ControllerStatusString: ControllerStatusString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerStatusStringGet command which is used to Return the controller status string. Refer to the XPS Programmer's manual to get the command description.

### ElapsedTimeGet

#### Syntax

##### C# prototype

int ElapsedTimeGet(out double ElapsedTime, out string errstring)

##### Python prototype

[ElapsedTime, errstring] ElapsedTimeGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) ElapsedTime: ElapsedTime

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ElapsedTimeGet command which is used to Return elapsed time from controller power on. Refer to the XPS Programmer's manual to get the command description.

### ErrorStringGet

#### Syntax

##### C# prototype

int ErrorStringGet(int ErrorCode, out string ErrorString, out string errstring)

##### Python prototype

[ErrorString, errstring] ErrorStringGet (ErrorCode)

#### Parameters

##### Input parameters

(int) ErrorCode: ErrorCode

##### Output parameters

(string) ErrorString: ErrorString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ErrorStringGet command which is used to Return the error string corresponding to the error code. Refer to the XPS Programmer's manual to get the command description.

### Login

#### Syntax

##### C# prototype

int Login(string Name, string Password, out string errstring)

##### Python prototype

[errstring] Login (Name, Password)

#### Parameters

##### Input parameters

(string) Name: Name

(string) Password: Password

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous Login command which is used to Log in. Refer to the XPS Programmer's manual to get the command description.

### CloseAllOtherSockets

#### Syntax

##### C# prototype

int CloseAllOtherSockets( out string errstring)

##### Python prototype

[errstring] CloseAllOtherSockets ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CloseAllOtherSockets command which is used to Close all socket beside the one used to send this command. Refer to the XPS Programmer's manual to get the command description.

### HardwareDriverAndStageGet

#### Syntax

##### C# prototype

int HardwareDriverAndStageGet(int PlugNumber, out string DriverName, out string StageName, out string errstring)

##### Python prototype

[DriverName, StageName, errstring] HardwareDriverAndStageGet (PlugNumber)

#### Parameters

##### Input parameters

(int) PlugNumber: PlugNumber

##### Output parameters

(string) DriverName: DriverName

(string) StageName: StageName

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous HardwareDriverAndStageGet command which is used to Smart hardware. Refer to the XPS Programmer's manual to get the command description.

### HardwareDateAndTimeGet

#### Syntax

##### C# prototype

int HardwareDateAndTimeGet(out string DateAndTime, out string errstring)

##### Python prototype

[DateAndTime, errstring] HardwareDateAndTimeGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) DateAndTime: DateAndTime

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous HardwareDateAndTimeGet command which is used to Return hardware date and time. Refer to the XPS Programmer's manual to get the command description.

### HardwareDateAndTimeSet

#### Syntax

##### C# prototype

int HardwareDateAndTimeSet(string DateAndTime, out string errstring)

##### Python prototype

[errstring] HardwareDateAndTimeSet (DateAndTime)

#### Parameters

##### Input parameters

(string) DateAndTime: DateAndTime

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous HardwareDateAndTimeSet command which is used to Set hardware date and time. Refer to the XPS Programmer's manual to get the command description.

### FileScriptHistoryRename

#### Syntax

##### C# prototype

int FileScriptHistoryRename(string TCLFileName, out string errstring)

##### Python prototype

[errstring] FileScriptHistoryRename (TCLFileName)

#### Parameters

##### Input parameters

(string) TCLFileName: TCLFileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous FileScriptHistoryRename command which is used to Rename History.tcl. Refer to the XPS Programmer's manual to get the command description.

### FileGatheringRename

#### Syntax

##### C# prototype

int FileGatheringRename(string FileName, out string errstring)

##### Python prototype

[errstring] FileGatheringRename (FileName)

#### Parameters

##### Input parameters

(string) FileName: FileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous FileGatheringRename command which is used to Rename Gathering.dat. Refer to the XPS Programmer's manual to get the command description.

### INTServitudesStatusGet

#### Syntax

##### C# prototype

int INTServitudesStatusGet(out short INTServitudesStatus, out string errstring)

##### Python prototype

[INTServitudesStatus, errstring] INTServitudesStatusGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(short) INTServitudesStatus: INTServitudesStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous INTServitudesStatusGet command which is used to Read INT servitudes status. Refer to the XPS Programmer's manual to get the command description.

### INTServitudesCommandGet

#### Syntax

##### C# prototype

int INTServitudesCommandGet(out short INTServitudesCommand, out string errstring)

##### Python prototype

[INTServitudesCommand, errstring] INTServitudesCommandGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(short) INTServitudesCommand: INTServitudesCommand

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous INTServitudesCommandGet command which is used to Read INT servitudes command. Refer to the XPS Programmer's manual to get the command description.

### TCLScriptExecute

#### Syntax

##### C# prototype

int TCLScriptExecute(string TCLFileName, string TaskName, string ParametersList, out string errstring)

##### Python prototype

[errstring] TCLScriptExecute (TCLFileName, TaskName, ParametersList)

#### Parameters

##### Input parameters

(string) TCLFileName: TCLFileName

(string) TaskName: TaskName

(string) ParametersList: ParametersList

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TCLScriptExecute command which is used to Execute a TCL script from a TCL file. Refer to the XPS Programmer's manual to get the command description.

### TCLScriptExecuteAndWait

#### Syntax

##### C# prototype

int TCLScriptExecuteAndWait(string TCLFileName, string TaskName, string InputParametersList, out string OutputParametersList, out string errstring)

##### Python prototype

[OutputParametersList, errstring] TCLScriptExecuteAndWait (TCLFileName, TaskName, InputParametersList)

#### Parameters

##### Input parameters

(string) TCLFileName: TCLFileName

(string) TaskName: TaskName

(string) InputParametersList: InputParametersList

##### Output parameters

(string) OutputParametersList: OutputParametersList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TCLScriptExecuteAndWait command which is used to Execute a TCL script from a TCL file and wait the end of execution to return. Refer to the XPS Programmer's manual to get the command description.

### TCLScriptExecuteWithPriority

#### Syntax

##### C# prototype

int TCLScriptExecuteWithPriority(string TCLFileName, string TaskName, string TaskPriorityLevel, string ParametersList, out string errstring)

##### Python prototype

[errstring] TCLScriptExecuteWithPriority (TCLFileName, TaskName, TaskPriorityLevel, ParametersList)

#### Parameters

##### Input parameters

(string) TCLFileName: TCLFileName

(string) TaskName: TaskName

(string) TaskPriorityLevel: TaskPriorityLevel

(string) ParametersList: ParametersList

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TCLScriptExecuteWithPriority command which is used to Execute a TCL script with defined priority. Refer to the XPS Programmer's manual to get the command description.

### TCLScriptKill

#### Syntax

##### C# prototype

int TCLScriptKill(string TaskName, out string errstring)

##### Python prototype

[errstring] TCLScriptKill (TaskName)

#### Parameters

##### Input parameters

(string) TaskName: TaskName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TCLScriptKill command which is used to Kill TCL Task. Refer to the XPS Programmer's manual to get the command description.

### TCLScriptKillAll

#### Syntax

##### C# prototype

int TCLScriptKillAll( out string errstring)

##### Python prototype

[errstring] TCLScriptKillAll ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TCLScriptKillAll command which is used to Kill all TCL Tasks. Refer to the XPS Programmer's manual to get the command description.

### TCLScriptRunningListGet

#### Syntax

##### C# prototype

int TCLScriptRunningListGet(out string TCLTaskList, out string errstring)

##### Python prototype

[TCLTaskList, errstring] TCLScriptRunningListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) TCLTaskList: TCLTaskList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TCLScriptRunningListGet command which is used to TCL task running list . Refer to the XPS Programmer's manual to get the command description.

### TimerGet

#### Syntax

##### C# prototype

int TimerGet(string TimerName, out int FrequencyTicks, out string errstring)

##### Python prototype

[FrequencyTicks, errstring] TimerGet (TimerName)

#### Parameters

##### Input parameters

(string) TimerName: TimerName

##### Output parameters

(int) FrequencyTicks: FrequencyTicks

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TimerGet command which is used to Get a timer. Refer to the XPS Programmer's manual to get the command description.

### TimerSet

#### Syntax

##### C# prototype

int TimerSet(string TimerName, int FrequencyTicks, out string errstring)

##### Python prototype

[errstring] TimerSet (TimerName, FrequencyTicks)

#### Parameters

##### Input parameters

(string) TimerName: TimerName

(int) FrequencyTicks: FrequencyTicks

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TimerSet command which is used to Set a timer. Refer to the XPS Programmer's manual to get the command description.

### GlobalArrayGet

#### Syntax

##### C# prototype

int GlobalArrayGet(int Number, out string ValueString, out string errstring)

##### Python prototype

[ValueString, errstring] GlobalArrayGet (Number)

#### Parameters

##### Input parameters

(int) Number: Number

##### Output parameters

(string) ValueString: ValueString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GlobalArrayGet command which is used to Get global array value. Refer to the XPS Programmer's manual to get the command description.

### GlobalArraySet

#### Syntax

##### C# prototype

int GlobalArraySet(int Number, string ValueString, out string errstring)

##### Python prototype

[errstring] GlobalArraySet (Number, ValueString)

#### Parameters

##### Input parameters

(int) Number: Number

(string) ValueString: ValueString

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GlobalArraySet command which is used to Set global array value. Refer to the XPS Programmer's manual to get the command description.

### DoubleGlobalArrayGet

#### Syntax

##### C# prototype

int DoubleGlobalArrayGet(int Number, out double DoubleValue, out string errstring)

##### Python prototype

[DoubleValue, errstring] DoubleGlobalArrayGet (Number)

#### Parameters

##### Input parameters

(int) Number: Number

##### Output parameters

(double) DoubleValue: DoubleValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous DoubleGlobalArrayGet command which is used to Get double global array value. Refer to the XPS Programmer's manual to get the command description.

### DoubleGlobalArraySet

#### Syntax

##### C# prototype

int DoubleGlobalArraySet(int Number, double DoubleValue, out string errstring)

##### Python prototype

[errstring] DoubleGlobalArraySet (Number, DoubleValue)

#### Parameters

##### Input parameters

(int) Number: Number

(double) DoubleValue: DoubleValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous DoubleGlobalArraySet command which is used to Set double global array value. Refer to the XPS Programmer's manual to get the command description.

### PositionerMagneticTrackPositionAtHomeGet

#### Syntax

##### C# prototype

int PositionerMagneticTrackPositionAtHomeGet(string PositionerName, out double MagneticTrackPosition, out double RefPos, out double CorrectedEncoderPos, out double PhaseBeforeCalculation, out double PhaseAfterCalculation, out double MagnetPeriod, out double ResMod2Pi, out string errstring)

##### Python prototype

[MagneticTrackPosition, RefPos, CorrectedEncoderPos, PhaseBeforeCalculation, PhaseAfterCalculation, MagnetPeriod, ResMod2Pi, errstring] PositionerMagneticTrackPositionAtHomeGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) MagneticTrackPosition: MagneticTrackPosition

(double) RefPos: RefPos

(double) CorrectedEncoderPos: CorrectedEncoderPos

(double) PhaseBeforeCalculation: PhaseBeforeCalculation

(double) PhaseAfterCalculation: PhaseAfterCalculation

(double) MagnetPeriod: MagnetPeriod

(double) ResMod2Pi: ResMod2Pi

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerMagneticTrackPositionAtHomeGet command which is used to Get magnetic track position at home in units, RefPos=Round. Refer to the XPS Programmer's manual to get the command description.

### ZygoConnectToServer

#### Syntax

##### C# prototype

int ZygoConnectToServer( out string errstring)

##### Python prototype

[errstring] ZygoConnectToServer ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoConnectToServer command which is used to Connect to Zygo TCP server. Refer to the XPS Programmer's manual to get the command description.

### ZygoReset

#### Syntax

##### C# prototype

int ZygoReset( out string errstring)

##### Python prototype

[errstring] ZygoReset ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoReset command which is used to Reset all Zygo axes. Refer to the XPS Programmer's manual to get the command description.

### ZygoReadWord

#### Syntax

##### C# prototype

int ZygoReadWord(string AxisNum, string Register, out long RegisterValue, out string Response, out string errstring)

##### Python prototype

[RegisterValue, Response, errstring] ZygoReadWord (AxisNum, Register)

#### Parameters

##### Input parameters

(string) AxisNum: AxisNum

(string) Register: Register

##### Output parameters

(long) RegisterValue: RegisterValue

(string) Response: Response

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoReadWord command which is used to Read word size from Zygo register,enter AXIS1|2|3|4 and register offset. Refer to the XPS Programmer's manual to get the command description.

### ZygoReadLong

#### Syntax

##### C# prototype

int ZygoReadLong(string AxisNum, string Register, out string Response, out string errstring)

##### Python prototype

[Response, errstring] ZygoReadLong (AxisNum, Register)

#### Parameters

##### Input parameters

(string) AxisNum: AxisNum

(string) Register: Register

##### Output parameters

(string) Response: Response

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoReadLong command which is used to Read long size from Zygo register,enter AXIS1|2|3|4 and register offset. Refer to the XPS Programmer's manual to get the command description.

### ZygoWriteWord

#### Syntax

##### C# prototype

int ZygoWriteWord(string AxisNum, string Register, string Data, out string Response, out string errstring)

##### Python prototype

[Response, errstring] ZygoWriteWord (AxisNum, Register, Data)

#### Parameters

##### Input parameters

(string) AxisNum: AxisNum

(string) Register: Register

(string) Data: Data

##### Output parameters

(string) Response: Response

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoWriteWord command which is used to Read word size from Zygo register,enter AXIS1|2|3|4, register offset and data value. Refer to the XPS Programmer's manual to get the command description.

### ZygoWriteLong

#### Syntax

##### C# prototype

int ZygoWriteLong(string AxisNum, string Register, string Data, out string Response, out string errstring)

##### Python prototype

[Response, errstring] ZygoWriteLong (AxisNum, Register, Data)

#### Parameters

##### Input parameters

(string) AxisNum: AxisNum

(string) Register: Register

(string) Data: Data

##### Output parameters

(string) Response: Response

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoWriteLong command which is used to Read long size from Zygo register,enter AXIS1|2|3|4, register offset and data value. Refer to the XPS Programmer's manual to get the command description.

### ZygoSendAndReceive

#### Syntax

##### C# prototype

int ZygoSendAndReceive(string Command, out string Response, out string errstring)

##### Python prototype

[Response, errstring] ZygoSendAndReceive (Command)

#### Parameters

##### Input parameters

(string) Command: Command

##### Output parameters

(string) Response: Response

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoSendAndReceive command which is used to Send command to Zygo borad and receive response. Refer to the XPS Programmer's manual to get the command description.

### ZygoDisconnectFromServer

#### Syntax

##### C# prototype

int ZygoDisconnectFromServer( out string errstring)

##### Python prototype

[errstring] ZygoDisconnectFromServer ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoDisconnectFromServer command which is used to Disconnect from Zygo server. Refer to the XPS Programmer's manual to get the command description.

### ZygoRegisterSet

#### Syntax

##### C# prototype

int ZygoRegisterSet(string PositionerName, int Register, int Value, out string errstring)

##### Python prototype

[errstring] ZygoRegisterSet (PositionerName, Register, Value)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) Register: Register

(int) Value: Value

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoRegisterSet command which is used to . Refer to the XPS Programmer's manual to get the command description.

### ZygoRegisterGet

#### Syntax

##### C# prototype

int ZygoRegisterGet(string PositionerName, int Register, out int Value, out string errstring)

##### Python prototype

[Value, errstring] ZygoRegisterGet (PositionerName, Register)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) Register: Register

##### Output parameters

(int) Value: Value

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ZygoRegisterGet command which is used to . Refer to the XPS Programmer's manual to get the command description.

### EventAdd

#### Syntax

##### C# prototype

int EventAdd(string PositionerName, string EventName, string EventParameter, string ActionName, string ActionParameter1, string ActionParameter2, string ActionParameter3, out string errstring)

##### Python prototype

[errstring] EventAdd (PositionerName, EventName, EventParameter, ActionName, ActionParameter1, ActionParameter2, ActionParameter3)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) EventName: EventName

(string) EventParameter: EventParameter

(string) ActionName: ActionName

(string) ActionParameter1: ActionParameter1

(string) ActionParameter2: ActionParameter2

(string) ActionParameter3: ActionParameter3

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventAdd command which is used to \*\* OBSOLETE \*\* Add an event. Refer to the XPS Programmer's manual to get the command description.

### EventGet

#### Syntax

##### C# prototype

int EventGet(string PositionerName, out string EventsAndActionsList, out string errstring)

##### Python prototype

[EventsAndActionsList, errstring] EventGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) EventsAndActionsList: EventsAndActionsList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventGet command which is used to \*\* OBSOLETE \*\* Read events and actions list. Refer to the XPS Programmer's manual to get the command description.

### EventRemove

#### Syntax

##### C# prototype

int EventRemove(string PositionerName, string EventName, string EventParameter, out string errstring)

##### Python prototype

[errstring] EventRemove (PositionerName, EventName, EventParameter)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) EventName: EventName

(string) EventParameter: EventParameter

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventRemove command which is used to \*\* OBSOLETE \*\* Delete an event. Refer to the XPS Programmer's manual to get the command description.

### EventWait

#### Syntax

##### C# prototype

int EventWait(string PositionerName, string EventName, string EventParameter, out string errstring)

##### Python prototype

[errstring] EventWait (PositionerName, EventName, EventParameter)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) EventName: EventName

(string) EventParameter: EventParameter

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventWait command which is used to \*\* OBSOLETE \*\* Wait an event. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedConfigurationTriggerSet

#### Syntax

##### C# prototype

int EventExtendedConfigurationTriggerSet(string[] ExtendedEventName, string[] EventParameter1, string[] EventParameter2, string[] EventParameter3, string[] EventParameter4, out string errstring)

##### Python prototype

[errstring] EventExtendedConfigurationTriggerSet (ExtendedEventName, EventParameter1, EventParameter2, EventParameter3, EventParameter4)

#### Parameters

##### Input parameters

(string[]) ExtendedEventName: ExtendedEventName

(string[]) EventParameter1: EventParameter1

(string[]) EventParameter2: EventParameter2

(string[]) EventParameter3: EventParameter3

(string[]) EventParameter4: EventParameter4

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedConfigurationTriggerSet command which is used to Configure one or several events. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedConfigurationTriggerGet

#### Syntax

##### C# prototype

int EventExtendedConfigurationTriggerGet(out string EventTriggerConfiguration, out string errstring)

##### Python prototype

[EventTriggerConfiguration, errstring] EventExtendedConfigurationTriggerGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) EventTriggerConfiguration: EventTriggerConfiguration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedConfigurationTriggerGet command which is used to Read the event configuration. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedConfigurationActionSet

#### Syntax

##### C# prototype

int EventExtendedConfigurationActionSet(string[] ExtendedActionName, string[] ActionParameter1, string[] ActionParameter2, string[] ActionParameter3, string[] ActionParameter4, out string errstring)

##### Python prototype

[errstring] EventExtendedConfigurationActionSet (ExtendedActionName, ActionParameter1, ActionParameter2, ActionParameter3, ActionParameter4)

#### Parameters

##### Input parameters

(string[]) ExtendedActionName: ExtendedActionName

(string[]) ActionParameter1: ActionParameter1

(string[]) ActionParameter2: ActionParameter2

(string[]) ActionParameter3: ActionParameter3

(string[]) ActionParameter4: ActionParameter4

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedConfigurationActionSet command which is used to Configure one or several actions. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedConfigurationActionGet

#### Syntax

##### C# prototype

int EventExtendedConfigurationActionGet(out string ActionConfiguration, out string errstring)

##### Python prototype

[ActionConfiguration, errstring] EventExtendedConfigurationActionGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) ActionConfiguration: ActionConfiguration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedConfigurationActionGet command which is used to Read the action configuration. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedStart

#### Syntax

##### C# prototype

int EventExtendedStart(out int ID, out string errstring)

##### Python prototype

[ID, errstring] EventExtendedStart ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) ID: ID

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedStart command which is used to Launch the last event and action configuration and return an ID. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedAllGet

#### Syntax

##### C# prototype

int EventExtendedAllGet(out string EventActionConfigurations, out string errstring)

##### Python prototype

[EventActionConfigurations, errstring] EventExtendedAllGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) EventActionConfigurations: EventActionConfigurations

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedAllGet command which is used to Read all event and action configurations. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedGet

#### Syntax

##### C# prototype

int EventExtendedGet(int ID, out string EventTriggerConfiguration, out string ActionConfiguration, out string errstring)

##### Python prototype

[EventTriggerConfiguration, ActionConfiguration, errstring] EventExtendedGet (ID)

#### Parameters

##### Input parameters

(int) ID: ID

##### Output parameters

(string) EventTriggerConfiguration: EventTriggerConfiguration

(string) ActionConfiguration: ActionConfiguration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedGet command which is used to Read the event and action configuration defined by ID. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedRemove

#### Syntax

##### C# prototype

int EventExtendedRemove(int ID, out string errstring)

##### Python prototype

[errstring] EventExtendedRemove (ID)

#### Parameters

##### Input parameters

(int) ID: ID

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedRemove command which is used to Remove the event and action configuration defined by ID. Refer to the XPS Programmer's manual to get the command description.

### EventExtendedWait

#### Syntax

##### C# prototype

int EventExtendedWait( out string errstring)

##### Python prototype

[errstring] EventExtendedWait ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventExtendedWait command which is used to Wait events from the last event configuration. Refer to the XPS Programmer's manual to get the command description.

### GatheringConfigurationGet

#### Syntax

##### C# prototype

int GatheringConfigurationGet(out string Type, out string errstring)

##### Python prototype

[Type, errstring] GatheringConfigurationGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Type: Type

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringConfigurationGet command which is used to Read different mnemonique type. Refer to the XPS Programmer's manual to get the command description.

### GatheringConfigurationSet

#### Syntax

##### C# prototype

int GatheringConfigurationSet(string[] Type, out string errstring)

##### Python prototype

[errstring] GatheringConfigurationSet (Type)

#### Parameters

##### Input parameters

(string[]) Type: Type

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringConfigurationSet command which is used to Configuration acquisition. Refer to the XPS Programmer's manual to get the command description.

### GatheringCurrentNumberGet

#### Syntax

##### C# prototype

int GatheringCurrentNumberGet(out int CurrentNumber, out int MaximumSamplesNumber, out string errstring)

##### Python prototype

[CurrentNumber, MaximumSamplesNumber, errstring] GatheringCurrentNumberGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) CurrentNumber: CurrentNumber

(int) MaximumSamplesNumber: MaximumSamplesNumber

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringCurrentNumberGet command which is used to Maximum number of samples and current number during acquisition. Refer to the XPS Programmer's manual to get the command description.

### GatheringStopAndSave

#### Syntax

##### C# prototype

int GatheringStopAndSave( out string errstring)

##### Python prototype

[errstring] GatheringStopAndSave ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringStopAndSave command which is used to Stop acquisition and save data. Refer to the XPS Programmer's manual to get the command description.

### GatheringDataAcquire

#### Syntax

##### C# prototype

int GatheringDataAcquire( out string errstring)

##### Python prototype

[errstring] GatheringDataAcquire ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringDataAcquire command which is used to Acquire a configured data. Refer to the XPS Programmer's manual to get the command description.

### GatheringDataGet

#### Syntax

##### C# prototype

int GatheringDataGet(int IndexPoint, out string DataBufferLine, out string errstring)

##### Python prototype

[DataBufferLine, errstring] GatheringDataGet (IndexPoint)

#### Parameters

##### Input parameters

(int) IndexPoint: IndexPoint

##### Output parameters

(string) DataBufferLine: DataBufferLine

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringDataGet command which is used to Get a data line from gathering buffer. Refer to the XPS Programmer's manual to get the command description.

### GatheringDataMultipleLinesGet

#### Syntax

##### C# prototype

int GatheringDataMultipleLinesGet(int IndexPoint, int NumberOfLines, out string DataBufferLine, out string errstring)

##### Python prototype

[DataBufferLine, errstring] GatheringDataMultipleLinesGet (IndexPoint, NumberOfLines)

#### Parameters

##### Input parameters

(int) IndexPoint: IndexPoint

(int) NumberOfLines: NumberOfLines

##### Output parameters

(string) DataBufferLine: DataBufferLine

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringDataMultipleLinesGet command which is used to Get multiple data lines from gathering buffer. Refer to the XPS Programmer's manual to get the command description.

### GatheringReset

#### Syntax

##### C# prototype

int GatheringReset( out string errstring)

##### Python prototype

[errstring] GatheringReset ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringReset command which is used to Empty the gathered data in memory to start new gathering from scratch. Refer to the XPS Programmer's manual to get the command description.

### GatheringRun

#### Syntax

##### C# prototype

int GatheringRun(int DataNumber, int Divisor, out string errstring)

##### Python prototype

[errstring] GatheringRun (DataNumber, Divisor)

#### Parameters

##### Input parameters

(int) DataNumber: DataNumber

(int) Divisor: Divisor

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringRun command which is used to Start a new gathering. Refer to the XPS Programmer's manual to get the command description.

### GatheringRunAppend

#### Syntax

##### C# prototype

int GatheringRunAppend( out string errstring)

##### Python prototype

[errstring] GatheringRunAppend ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringRunAppend command which is used to Re-start the stopped gathering to add new data. Refer to the XPS Programmer's manual to get the command description.

### GatheringStop

#### Syntax

##### C# prototype

int GatheringStop( out string errstring)

##### Python prototype

[errstring] GatheringStop ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringStop command which is used to Stop the data gathering . Refer to the XPS Programmer's manual to get the command description.

### GatheringExternalConfigurationSet

#### Syntax

##### C# prototype

int GatheringExternalConfigurationSet(string[] Type, out string errstring)

##### Python prototype

[errstring] GatheringExternalConfigurationSet (Type)

#### Parameters

##### Input parameters

(string[]) Type: Type

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringExternalConfigurationSet command which is used to Configuration acquisition. Refer to the XPS Programmer's manual to get the command description.

### GatheringExternalConfigurationGet

#### Syntax

##### C# prototype

int GatheringExternalConfigurationGet(out string Type, out string errstring)

##### Python prototype

[Type, errstring] GatheringExternalConfigurationGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Type: Type

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringExternalConfigurationGet command which is used to Read different mnemonique type. Refer to the XPS Programmer's manual to get the command description.

### GatheringExternalCurrentNumberGet

#### Syntax

##### C# prototype

int GatheringExternalCurrentNumberGet(out int CurrentNumber, out int MaximumSamplesNumber, out string errstring)

##### Python prototype

[CurrentNumber, MaximumSamplesNumber, errstring] GatheringExternalCurrentNumberGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) CurrentNumber: CurrentNumber

(int) MaximumSamplesNumber: MaximumSamplesNumber

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringExternalCurrentNumberGet command which is used to Maximum number of samples and current number during acquisition. Refer to the XPS Programmer's manual to get the command description.

### GatheringExternalDataGet

#### Syntax

##### C# prototype

int GatheringExternalDataGet(int IndexPoint, out string DataBufferLine, out string errstring)

##### Python prototype

[DataBufferLine, errstring] GatheringExternalDataGet (IndexPoint)

#### Parameters

##### Input parameters

(int) IndexPoint: IndexPoint

##### Output parameters

(string) DataBufferLine: DataBufferLine

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringExternalDataGet command which is used to Get a data line from external gathering buffer. Refer to the XPS Programmer's manual to get the command description.

### GatheringExternalStopAndSave

#### Syntax

##### C# prototype

int GatheringExternalStopAndSave( out string errstring)

##### Python prototype

[errstring] GatheringExternalStopAndSave ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringExternalStopAndSave command which is used to Stop acquisition and save data. Refer to the XPS Programmer's manual to get the command description.

### GPIOAnalogGet

#### Syntax

##### C# prototype

int GPIOAnalogGet(string[] GPIOName, out double[] AnalogValue, out string errstring)

##### Python prototype

[AnalogValue, errstring] GPIOAnalogGet (GPIOName)

#### Parameters

##### Input parameters

(string[]) GPIOName: GPIOName

##### Output parameters

(double[]) AnalogValue: AnalogValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GPIOAnalogGet command which is used to Read analog input or analog output for one or few input. Refer to the XPS Programmer's manual to get the command description.

### GPIOAnalogSet

#### Syntax

##### C# prototype

int GPIOAnalogSet(string[] GPIOName, double[] AnalogOutputValue, out string errstring)

##### Python prototype

[errstring] GPIOAnalogSet (GPIOName, AnalogOutputValue)

#### Parameters

##### Input parameters

(string[]) GPIOName: GPIOName

(double[]) AnalogOutputValue: AnalogOutputValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GPIOAnalogSet command which is used to Set analog output for one or few output. Refer to the XPS Programmer's manual to get the command description.

### GPIOAnalogGainGet

#### Syntax

##### C# prototype

int GPIOAnalogGainGet(string[] GPIOName, out int[] AnalogInputGainValue, out string errstring)

##### Python prototype

[AnalogInputGainValue, errstring] GPIOAnalogGainGet (GPIOName)

#### Parameters

##### Input parameters

(string[]) GPIOName: GPIOName

##### Output parameters

(int[]) AnalogInputGainValue: AnalogInputGainValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GPIOAnalogGainGet command which is used to Read analog input gain 1, 2, 4 or 8, for one or few input. Refer to the XPS Programmer's manual to get the command description.

### GPIOAnalogGainSet

#### Syntax

##### C# prototype

int GPIOAnalogGainSet(string[] GPIOName, int[] AnalogInputGainValue, out string errstring)

##### Python prototype

[errstring] GPIOAnalogGainSet (GPIOName, AnalogInputGainValue)

#### Parameters

##### Input parameters

(string[]) GPIOName: GPIOName

(int[]) AnalogInputGainValue: AnalogInputGainValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GPIOAnalogGainSet command which is used to Set analog input gain 1, 2, 4 or 8, for one or few input. Refer to the XPS Programmer's manual to get the command description.

### GPIOAnalogConfigurationGet

#### Syntax

##### C# prototype

int GPIOAnalogConfigurationGet(out ulong DACConfiguration, out string errstring)

##### Python prototype

[DACConfiguration, errstring] GPIOAnalogConfigurationGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(ulong) DACConfiguration: DACConfiguration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GPIOAnalogConfigurationGet command which is used to Get analog output configuration. Refer to the XPS Programmer's manual to get the command description.

### GPIOAnalogConfigurationSet

#### Syntax

##### C# prototype

int GPIOAnalogConfigurationSet(uint DAC1\_4Configuration, uint DAC5\_8Configuration, out string errstring)

##### Python prototype

[errstring] GPIOAnalogConfigurationSet (DAC1\_4Configuration, DAC5\_8Configuration)

#### Parameters

##### Input parameters

(uint) DAC1\_4Configuration: DAC1\_4Configuration

(uint) DAC5\_8Configuration: DAC5\_8Configuration

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GPIOAnalogConfigurationSet command which is used to Set FAR DAC configuration for DAC1-4 and DAC5-8 . Refer to the XPS Programmer's manual to get the command description.

### GPIODigitalGet

#### Syntax

##### C# prototype

int GPIODigitalGet(string GPIOName, out ushort DigitalValue, out string errstring)

##### Python prototype

[DigitalValue, errstring] GPIODigitalGet (GPIOName)

#### Parameters

##### Input parameters

(string) GPIOName: GPIOName

##### Output parameters

(ushort) DigitalValue: DigitalValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GPIODigitalGet command which is used to Read digital output or digital input . Refer to the XPS Programmer's manual to get the command description.

### GPIODigitalSet

#### Syntax

##### C# prototype

int GPIODigitalSet(string GPIOName, ushort Mask, ushort DigitalOutputValue, out string errstring)

##### Python prototype

[errstring] GPIODigitalSet (GPIOName, Mask, DigitalOutputValue)

#### Parameters

##### Input parameters

(string) GPIOName: GPIOName

(ushort) Mask: Mask

(ushort) DigitalOutputValue: DigitalOutputValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GPIODigitalSet command which is used to Set Digital Output for one or few output TTL. Refer to the XPS Programmer's manual to get the command description.

### KillAll

#### Syntax

##### C# prototype

int KillAll( out string errstring)

##### Python prototype

[errstring] KillAll ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous KillAll command which is used to Put all groups in 'Not initialized' state. Refer to the XPS Programmer's manual to get the command description.

### GroupAccelerationSetpointGet

#### Syntax

##### C# prototype

int GroupAccelerationSetpointGet(string GroupName, out double[] SetpointAcceleration, int nbItems, out string errstring)

##### Python prototype

[SetpointAcceleration, errstring] GroupAccelerationSetpointGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) SetpointAcceleration: SetpointAcceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupAccelerationSetpointGet command which is used to Return setpoint accelerations. Refer to the XPS Programmer's manual to get the command description.

### GroupAnalogTrackingModeEnable

#### Syntax

##### C# prototype

int GroupAnalogTrackingModeEnable(string GroupName, string Type, out string errstring)

##### Python prototype

[errstring] GroupAnalogTrackingModeEnable (GroupName, Type)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) Type: Type

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupAnalogTrackingModeEnable command which is used to Enable Analog Tracking mode on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupAnalogTrackingModeDisable

#### Syntax

##### C# prototype

int GroupAnalogTrackingModeDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupAnalogTrackingModeDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupAnalogTrackingModeDisable command which is used to Disable Analog Tracking mode on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupCorrectorOutputGet

#### Syntax

##### C# prototype

int GroupCorrectorOutputGet(string GroupName, out double[] CorrectorOutput, int nbItems, out string errstring)

##### Python prototype

[CorrectorOutput, errstring] GroupCorrectorOutputGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) CorrectorOutput: CorrectorOutput

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupCorrectorOutputGet command which is used to Return corrector outputs. Refer to the XPS Programmer's manual to get the command description.

### GroupCurrentFollowingErrorGet

#### Syntax

##### C# prototype

int GroupCurrentFollowingErrorGet(string GroupName, out double[] CurrentFollowingError, int nbItems, out string errstring)

##### Python prototype

[CurrentFollowingError, errstring] GroupCurrentFollowingErrorGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) CurrentFollowingError: CurrentFollowingError

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupCurrentFollowingErrorGet command which is used to Return current following errors. Refer to the XPS Programmer's manual to get the command description.

### GroupHomeSearch

#### Syntax

##### C# prototype

int GroupHomeSearch(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupHomeSearch (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupHomeSearch command which is used to Start home search sequence. Refer to the XPS Programmer's manual to get the command description.

### GroupHomeSearchAndRelativeMove

#### Syntax

##### C# prototype

int GroupHomeSearchAndRelativeMove(string GroupName, double[] TargetDisplacement, int nbItems, out string errstring)

##### Python prototype

[errstring] GroupHomeSearchAndRelativeMove (GroupName, TargetDisplacement, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double[]) TargetDisplacement: TargetDisplacement

(int) nbItems: nbItems

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupHomeSearchAndRelativeMove command which is used to Start home search sequence and execute a displacement. Refer to the XPS Programmer's manual to get the command description.

### GroupInitialize

#### Syntax

##### C# prototype

int GroupInitialize(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupInitialize (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupInitialize command which is used to Start the initialization. Refer to the XPS Programmer's manual to get the command description.

### GroupInitializeNoEncoderReset

#### Syntax

##### C# prototype

int GroupInitializeNoEncoderReset(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupInitializeNoEncoderReset (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupInitializeNoEncoderReset command which is used to Group initialization with no encoder reset. Refer to the XPS Programmer's manual to get the command description.

### GroupInitializeWithEncoderCalibration

#### Syntax

##### C# prototype

int GroupInitializeWithEncoderCalibration(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupInitializeWithEncoderCalibration (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupInitializeWithEncoderCalibration command which is used to Group initialization with encoder calibration. Refer to the XPS Programmer's manual to get the command description.

### GroupInterlockDisable

#### Syntax

##### C# prototype

int GroupInterlockDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupInterlockDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupInterlockDisable command which is used to Set group interlock disable. Refer to the XPS Programmer's manual to get the command description.

### GroupInterlockEnable

#### Syntax

##### C# prototype

int GroupInterlockEnable(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupInterlockEnable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupInterlockEnable command which is used to Set group interlock enable. Refer to the XPS Programmer's manual to get the command description.

### GroupJogParametersSet

#### Syntax

##### C# prototype

int GroupJogParametersSet(string GroupName, double[] Velocity, double[] Acceleration, int nbItems, out string errstring)

##### Python prototype

[errstring] GroupJogParametersSet (GroupName, Velocity, Acceleration, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double[]) Velocity: Velocity

(double[]) Acceleration: Acceleration

(int) nbItems: nbItems

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupJogParametersSet command which is used to Modify Jog parameters on selected group and activate the continuous move. Refer to the XPS Programmer's manual to get the command description.

### GroupJogParametersGet

#### Syntax

##### C# prototype

int GroupJogParametersGet(string GroupName, out double[] Velocity, out double[] Acceleration, int nbItems, out string errstring)

##### Python prototype

[Velocity, Acceleration, errstring] GroupJogParametersGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) Velocity: Velocity

(double[]) Acceleration: Acceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupJogParametersGet command which is used to Get Jog parameters on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupJogCurrentGet

#### Syntax

##### C# prototype

int GroupJogCurrentGet(string GroupName, out double[] Velocity, out double[] Acceleration, int nbItems, out string errstring)

##### Python prototype

[Velocity, Acceleration, errstring] GroupJogCurrentGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) Velocity: Velocity

(double[]) Acceleration: Acceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupJogCurrentGet command which is used to Get Jog current on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupJogModeEnable

#### Syntax

##### C# prototype

int GroupJogModeEnable(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupJogModeEnable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupJogModeEnable command which is used to Enable Jog mode on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupJogModeDisable

#### Syntax

##### C# prototype

int GroupJogModeDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupJogModeDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupJogModeDisable command which is used to Disable Jog mode on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupKill

#### Syntax

##### C# prototype

int GroupKill(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupKill (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupKill command which is used to Kill the group. Refer to the XPS Programmer's manual to get the command description.

### GroupMotionDisable

#### Syntax

##### C# prototype

int GroupMotionDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupMotionDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMotionDisable command which is used to Set Motion disable on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupMotionEnable

#### Syntax

##### C# prototype

int GroupMotionEnable(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupMotionEnable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMotionEnable command which is used to Set Motion enable on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupMotionStatusGet

#### Syntax

##### C# prototype

int GroupMotionStatusGet(string GroupName, out int[] Status, int nbItems, out string errstring)

##### Python prototype

[Status, errstring] GroupMotionStatusGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(int[]) Status: Status

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMotionStatusGet command which is used to Return group or positioner status. Refer to the XPS Programmer's manual to get the command description.

### GroupMoveAbort

#### Syntax

##### C# prototype

int GroupMoveAbort(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupMoveAbort (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMoveAbort command which is used to Abort a move. Refer to the XPS Programmer's manual to get the command description.

### GroupMoveAbortFast

#### Syntax

##### C# prototype

int GroupMoveAbortFast(string GroupName, int AccelerationMultiplier, out string errstring)

##### Python prototype

[errstring] GroupMoveAbortFast (GroupName, AccelerationMultiplier)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) AccelerationMultiplier: AccelerationMultiplier

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMoveAbortFast command which is used to Abort quickly a move. Refer to the XPS Programmer's manual to get the command description.

### GroupMoveAbsolute

#### Syntax

##### C# prototype

int GroupMoveAbsolute(string GroupName, double[] TargetPosition, int nbItems, out string errstring)

##### Python prototype

[errstring] GroupMoveAbsolute (GroupName, TargetPosition, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double[]) TargetPosition: TargetPosition

(int) nbItems: nbItems

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMoveAbsolute command which is used to Do an absolute move. Refer to the XPS Programmer's manual to get the command description.

### GroupMoveRelative

#### Syntax

##### C# prototype

int GroupMoveRelative(string GroupName, double[] TargetDisplacement, int nbItems, out string errstring)

##### Python prototype

[errstring] GroupMoveRelative (GroupName, TargetDisplacement, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double[]) TargetDisplacement: TargetDisplacement

(int) nbItems: nbItems

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMoveRelative command which is used to Do a relative move. Refer to the XPS Programmer's manual to get the command description.

### GroupPositionCorrectedProfilerGet

#### Syntax

##### C# prototype

int GroupPositionCorrectedProfilerGet(string GroupName, double PositionX, double PositionY, out double CorrectedProfilerPositionX, out double CorrectedProfilerPositionY, out string errstring)

##### Python prototype

[CorrectedProfilerPositionX, CorrectedProfilerPositionY, errstring] GroupPositionCorrectedProfilerGet (GroupName, PositionX, PositionY)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) PositionX: PositionX

(double) PositionY: PositionY

##### Output parameters

(double) CorrectedProfilerPositionX: CorrectedProfilerPositionX

(double) CorrectedProfilerPositionY: CorrectedProfilerPositionY

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupPositionCorrectedProfilerGet command which is used to Return corrected profiler positions. Refer to the XPS Programmer's manual to get the command description.

### GroupPositionCurrentGet

#### Syntax

##### C# prototype

int GroupPositionCurrentGet(string GroupName, out double[] CurrentEncoderPosition, int nbItems, out string errstring)

##### Python prototype

[CurrentEncoderPosition, errstring] GroupPositionCurrentGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) CurrentEncoderPosition: CurrentEncoderPosition

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupPositionCurrentGet command which is used to Return current positions. Refer to the XPS Programmer's manual to get the command description.

### GroupPositionPCORawEncoderGet

#### Syntax

##### C# prototype

int GroupPositionPCORawEncoderGet(string GroupName, double PositionX, double PositionY, out double PCORawPositionX, out double PCORawPositionY, out string errstring)

##### Python prototype

[PCORawPositionX, PCORawPositionY, errstring] GroupPositionPCORawEncoderGet (GroupName, PositionX, PositionY)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) PositionX: PositionX

(double) PositionY: PositionY

##### Output parameters

(double) PCORawPositionX: PCORawPositionX

(double) PCORawPositionY: PCORawPositionY

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupPositionPCORawEncoderGet command which is used to Return PCO raw encoder positions. Refer to the XPS Programmer's manual to get the command description.

### GroupPositionSetpointGet

#### Syntax

##### C# prototype

int GroupPositionSetpointGet(string GroupName, out double[] SetPointPosition, int nbItems, out string errstring)

##### Python prototype

[SetPointPosition, errstring] GroupPositionSetpointGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) SetPointPosition: SetPointPosition

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupPositionSetpointGet command which is used to Return setpoint positions. Refer to the XPS Programmer's manual to get the command description.

### GroupPositionTargetGet

#### Syntax

##### C# prototype

int GroupPositionTargetGet(string GroupName, out double[] TargetPosition, int nbItems, out string errstring)

##### Python prototype

[TargetPosition, errstring] GroupPositionTargetGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) TargetPosition: TargetPosition

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupPositionTargetGet command which is used to Return target positions. Refer to the XPS Programmer's manual to get the command description.

### GroupReferencingActionExecute

#### Syntax

##### C# prototype

int GroupReferencingActionExecute(string PositionerName, string ReferencingAction, string ReferencingSensor, double ReferencingParameter, out string errstring)

##### Python prototype

[errstring] GroupReferencingActionExecute (PositionerName, ReferencingAction, ReferencingSensor, ReferencingParameter)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) ReferencingAction: ReferencingAction

(string) ReferencingSensor: ReferencingSensor

(double) ReferencingParameter: ReferencingParameter

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupReferencingActionExecute command which is used to Execute an action in referencing mode. Refer to the XPS Programmer's manual to get the command description.

### GroupReferencingStart

#### Syntax

##### C# prototype

int GroupReferencingStart(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupReferencingStart (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupReferencingStart command which is used to Enter referencing mode. Refer to the XPS Programmer's manual to get the command description.

### GroupReferencingStop

#### Syntax

##### C# prototype

int GroupReferencingStop(string GroupName, out string errstring)

##### Python prototype

[errstring] GroupReferencingStop (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupReferencingStop command which is used to Exit referencing mode. Refer to the XPS Programmer's manual to get the command description.

### GroupStatusGet

#### Syntax

##### C# prototype

int GroupStatusGet(string GroupName, out int Status, out string errstring)

##### Python prototype

[Status, errstring] GroupStatusGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(int) Status: Status

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupStatusGet command which is used to Return group status. Refer to the XPS Programmer's manual to get the command description.

### GroupStatusStringGet

#### Syntax

##### C# prototype

int GroupStatusStringGet(int GroupStatusCode, out string GroupStatusString, out string errstring)

##### Python prototype

[GroupStatusString, errstring] GroupStatusStringGet (GroupStatusCode)

#### Parameters

##### Input parameters

(int) GroupStatusCode: GroupStatusCode

##### Output parameters

(string) GroupStatusString: GroupStatusString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupStatusStringGet command which is used to Return the group status string corresponding to the group status code. Refer to the XPS Programmer's manual to get the command description.

### GroupVelocityCurrentGet

#### Syntax

##### C# prototype

int GroupVelocityCurrentGet(string GroupName, out double[] CurrentVelocity, int nbItems, out string errstring)

##### Python prototype

[CurrentVelocity, errstring] GroupVelocityCurrentGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) CurrentVelocity: CurrentVelocity

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupVelocityCurrentGet command which is used to Return current velocities. Refer to the XPS Programmer's manual to get the command description.

### PositionerDriverStatusGet

#### Syntax

##### C# prototype

int PositionerDriverStatusGet(string PositionerName, out int DriverStatus, out string errstring)

##### Python prototype

[DriverStatus, errstring] PositionerDriverStatusGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(int) DriverStatus: DriverStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerDriverStatusGet command which is used to Read positioner driver status. Refer to the XPS Programmer's manual to get the command description.

### PositionerDriverStatusStringGet

#### Syntax

##### C# prototype

int PositionerDriverStatusStringGet(int PositionerDriverStatus, out string PositionerDriverStatusString, out string errstring)

##### Python prototype

[PositionerDriverStatusString, errstring] PositionerDriverStatusStringGet (PositionerDriverStatus)

#### Parameters

##### Input parameters

(int) PositionerDriverStatus: PositionerDriverStatus

##### Output parameters

(string) PositionerDriverStatusString: PositionerDriverStatusString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerDriverStatusStringGet command which is used to Return the positioner driver status string corresponding to the positioner error code. Refer to the XPS Programmer's manual to get the command description.

### PositionerErrorGet

#### Syntax

##### C# prototype

int PositionerErrorGet(string PositionerName, out int ErrorCode, out string errstring)

##### Python prototype

[ErrorCode, errstring] PositionerErrorGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(int) ErrorCode: ErrorCode

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerErrorGet command which is used to Read and clear positioner error code. Refer to the XPS Programmer's manual to get the command description.

### PositionerErrorRead

#### Syntax

##### C# prototype

int PositionerErrorRead(string PositionerName, out int ErrorCode, out string errstring)

##### Python prototype

[ErrorCode, errstring] PositionerErrorRead (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(int) ErrorCode: ErrorCode

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerErrorRead command which is used to Read only positioner error code without clear it. Refer to the XPS Programmer's manual to get the command description.

### PositionerErrorStringGet

#### Syntax

##### C# prototype

int PositionerErrorStringGet(int PositionerErrorCode, out string PositionerErrorString, out string errstring)

##### Python prototype

[PositionerErrorString, errstring] PositionerErrorStringGet (PositionerErrorCode)

#### Parameters

##### Input parameters

(int) PositionerErrorCode: PositionerErrorCode

##### Output parameters

(string) PositionerErrorString: PositionerErrorString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerErrorStringGet command which is used to Return the positioner status string corresponding to the positioner error code. Refer to the XPS Programmer's manual to get the command description.

### PositionerHardwareStatusGet

#### Syntax

##### C# prototype

int PositionerHardwareStatusGet(string PositionerName, out int HardwareStatus, out string errstring)

##### Python prototype

[HardwareStatus, errstring] PositionerHardwareStatusGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(int) HardwareStatus: HardwareStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerHardwareStatusGet command which is used to Read positioner hardware status. Refer to the XPS Programmer's manual to get the command description.

### PositionerHardwareStatusStringGet

#### Syntax

##### C# prototype

int PositionerHardwareStatusStringGet(int PositionerHardwareStatus, out string PositionerHardwareStatusString, out string errstring)

##### Python prototype

[PositionerHardwareStatusString, errstring] PositionerHardwareStatusStringGet (PositionerHardwareStatus)

#### Parameters

##### Input parameters

(int) PositionerHardwareStatus: PositionerHardwareStatus

##### Output parameters

(string) PositionerHardwareStatusString: PositionerHardwareStatusString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerHardwareStatusStringGet command which is used to Return the positioner hardware status string corresponding to the positioner error code. Refer to the XPS Programmer's manual to get the command description.

### PositionersEncoderIndexDifferenceGet

#### Syntax

##### C# prototype

int PositionersEncoderIndexDifferenceGet(string PositionerName, out double distance, out string errstring)

##### Python prototype

[distance, errstring] PositionersEncoderIndexDifferenceGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) distance: distance

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionersEncoderIndexDifferenceGet command which is used to Return the difference between index of primary axis and secondary axis . Refer to the XPS Programmer's manual to get the command description.

### PositionerGantryEndReferencingPositionGet

#### Syntax

##### C# prototype

int PositionerGantryEndReferencingPositionGet(string PositionerName, out double Position, out string errstring)

##### Python prototype

[Position, errstring] PositionerGantryEndReferencingPositionGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) Position: Position

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerGantryEndReferencingPositionGet command which is used to Return the secondary axis position at the end of home. Refer to the XPS Programmer's manual to get the command description.

### PositionerStageParameterGet

#### Syntax

##### C# prototype

int PositionerStageParameterGet(string PositionerName, string ParameterName, out string ParameterValue, out string errstring)

##### Python prototype

[ParameterValue, errstring] PositionerStageParameterGet (PositionerName, ParameterName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) ParameterName: ParameterName

##### Output parameters

(string) ParameterValue: ParameterValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerStageParameterGet command which is used to Return the stage parameter. Refer to the XPS Programmer's manual to get the command description.

### PositionerStageParameterSet

#### Syntax

##### C# prototype

int PositionerStageParameterSet(string PositionerName, string ParameterName, string ParameterValue, out string errstring)

##### Python prototype

[errstring] PositionerStageParameterSet (PositionerName, ParameterName, ParameterValue)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) ParameterName: ParameterName

(string) ParameterValue: ParameterValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerStageParameterSet command which is used to Save the stage parameter. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorTypeGet

#### Syntax

##### C# prototype

int PositionerCorrectorTypeGet(string PositionerName, out string CorrectorType, out string errstring)

##### Python prototype

[CorrectorType, errstring] PositionerCorrectorTypeGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) CorrectorType: CorrectorType

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorTypeGet command which is used to Read corrector type. Refer to the XPS Programmer's manual to get the command description.

### PositionerWarningFollowingErrorSet

#### Syntax

##### C# prototype

int PositionerWarningFollowingErrorSet(string PositionerName, double WarningFollowingError, out string errstring)

##### Python prototype

[errstring] PositionerWarningFollowingErrorSet (PositionerName, WarningFollowingError)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) WarningFollowingError: WarningFollowingError

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerWarningFollowingErrorSet command which is used to Set positioner warning following error limit. Refer to the XPS Programmer's manual to get the command description.

### PositionerWarningFollowingErrorGet

#### Syntax

##### C# prototype

int PositionerWarningFollowingErrorGet(string PositionerName, out double WarningFollowingError, out string errstring)

##### Python prototype

[WarningFollowingError, errstring] PositionerWarningFollowingErrorGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) WarningFollowingError: WarningFollowingError

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerWarningFollowingErrorGet command which is used to Get positioner warning following error limit. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPositionFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPositionFilterSet(string PositionerName, int FilterNumber, double Frequency, double DampingFactor, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPositionFilterSet (PositionerName, FilterNumber, Frequency, DampingFactor)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) FilterNumber: FilterNumber

(double) Frequency: Frequency

(double) DampingFactor: DampingFactor

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPositionFilterSet command which is used to Set and update position filter parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPositionFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPositionFilterGet(string PositionerName, int FilterNumber, out double Frequency, out double DampingFactor, out string errstring)

##### Python prototype

[Frequency, DampingFactor, errstring] PositionerCompensationPositionFilterGet (PositionerName, FilterNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) FilterNumber: FilterNumber

##### Output parameters

(double) Frequency: Frequency

(double) DampingFactor: DampingFactor

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPositionFilterGet command which is used to Get position filter parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorAutoTuning

#### Syntax

##### C# prototype

int PositionerCorrectorAutoTuning(string PositionerName, int TuningMode, out double KP, out double KI, out double KD, out string errstring)

##### Python prototype

[KP, KI, KD, errstring] PositionerCorrectorAutoTuning (PositionerName, TuningMode)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) TuningMode: TuningMode

##### Output parameters

(double) KP: KP

(double) KI: KI

(double) KD: KD

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorAutoTuning command which is used to Astrom and Hagglund based auto-tuning. Refer to the XPS Programmer's manual to get the command description.

### PositionerAccelerationAutoScaling

#### Syntax

##### C# prototype

int PositionerAccelerationAutoScaling(string PositionerName, out double Scaling, out string errstring)

##### Python prototype

[Scaling, errstring] PositionerAccelerationAutoScaling (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) Scaling: Scaling

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerAccelerationAutoScaling command which is used to Astrom and Hagglund based auto-scaling. Refer to the XPS Programmer's manual to get the command description.

### PositionerExcitationSignalGet

#### Syntax

##### C# prototype

int PositionerExcitationSignalGet(string PositionerName, out int Mode, out double Frequency, out double Amplitude, out double Time, out string errstring)

##### Python prototype

[Mode, Frequency, Amplitude, Time, errstring] PositionerExcitationSignalGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(int) Mode: Mode

(double) Frequency: Frequency

(double) Amplitude: Amplitude

(double) Time: Time

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerExcitationSignalGet command which is used to Get excitation signal mode. Refer to the XPS Programmer's manual to get the command description.

### PositionerExcitationSignalSet

#### Syntax

##### C# prototype

int PositionerExcitationSignalSet(string PositionerName, int Mode, double Frequency, double Amplitude, double Time, out string errstring)

##### Python prototype

[errstring] PositionerExcitationSignalSet (PositionerName, Mode, Frequency, Amplitude, Time)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) Mode: Mode

(double) Frequency: Frequency

(double) Amplitude: Amplitude

(double) Time: Time

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerExcitationSignalSet command which is used to Set excitation signal mode. Refer to the XPS Programmer's manual to get the command description.

### PositionerCurrentVelocityAccelerationFiltersSet

#### Syntax

##### C# prototype

int PositionerCurrentVelocityAccelerationFiltersSet(string PositionerName, double CurrentVelocityCutOffFrequency, double CurrentAccelerationCutOffFrequency, out string errstring)

##### Python prototype

[errstring] PositionerCurrentVelocityAccelerationFiltersSet (PositionerName, CurrentVelocityCutOffFrequency, CurrentAccelerationCutOffFrequency)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) CurrentVelocityCutOffFrequency: CurrentVelocityCutOffFrequency

(double) CurrentAccelerationCutOffFrequency: CurrentAccelerationCutOffFrequency

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCurrentVelocityAccelerationFiltersSet command which is used to Set current velocity and acceleration cut off frequencies. Refer to the XPS Programmer's manual to get the command description.

### PositionerCurrentVelocityAccelerationFiltersGet

#### Syntax

##### C# prototype

int PositionerCurrentVelocityAccelerationFiltersGet(string PositionerName, out double CurrentVelocityCutOffFrequency, out double CurrentAccelerationCutOffFrequency, out string errstring)

##### Python prototype

[CurrentVelocityCutOffFrequency, CurrentAccelerationCutOffFrequency, errstring] PositionerCurrentVelocityAccelerationFiltersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) CurrentVelocityCutOffFrequency: CurrentVelocityCutOffFrequency

(double) CurrentAccelerationCutOffFrequency: CurrentAccelerationCutOffFrequency

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCurrentVelocityAccelerationFiltersGet command which is used to Get current velocity and acceleration cut off frequencies. Refer to the XPS Programmer's manual to get the command description.

### PositionerEncoderAmplitudeValuesGet

#### Syntax

##### C# prototype

int PositionerEncoderAmplitudeValuesGet(string PositionerName, out double CalibrationSinusAmplitude, out double CurrentSinusAmplitude, out double CalibrationCosinusAmplitude, out double CurrentCosinusAmplitude, out string errstring)

##### Python prototype

[CalibrationSinusAmplitude, CurrentSinusAmplitude, CalibrationCosinusAmplitude, CurrentCosinusAmplitude, errstring] PositionerEncoderAmplitudeValuesGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) CalibrationSinusAmplitude: CalibrationSinusAmplitude

(double) CurrentSinusAmplitude: CurrentSinusAmplitude

(double) CalibrationCosinusAmplitude: CalibrationCosinusAmplitude

(double) CurrentCosinusAmplitude: CurrentCosinusAmplitude

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerEncoderAmplitudeValuesGet command which is used to Read analog interpolated encoder amplitude values. Refer to the XPS Programmer's manual to get the command description.

### PositionerEncoderCalibrationParametersGet

#### Syntax

##### C# prototype

int PositionerEncoderCalibrationParametersGet(string PositionerName, out double SinusOffset, out double CosinusOffset, out double DifferentialGain, out double PhaseCompensation, out string errstring)

##### Python prototype

[SinusOffset, CosinusOffset, DifferentialGain, PhaseCompensation, errstring] PositionerEncoderCalibrationParametersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) SinusOffset: SinusOffset

(double) CosinusOffset: CosinusOffset

(double) DifferentialGain: DifferentialGain

(double) PhaseCompensation: PhaseCompensation

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerEncoderCalibrationParametersGet command which is used to Read analog interpolated encoder calibration parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerRawEncoderPositionGet

#### Syntax

##### C# prototype

int PositionerRawEncoderPositionGet(string PositionerName, double UserEncoderPosition, out double RawEncoderPosition, out string errstring)

##### Python prototype

[RawEncoderPosition, errstring] PositionerRawEncoderPositionGet (PositionerName, UserEncoderPosition)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) UserEncoderPosition: UserEncoderPosition

##### Output parameters

(double) RawEncoderPosition: RawEncoderPosition

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerRawEncoderPositionGet command which is used to Get the raw encoder position. Refer to the XPS Programmer's manual to get the command description.

### PositionerBacklashSet

#### Syntax

##### C# prototype

int PositionerBacklashSet(string PositionerName, double BacklashValue, out string errstring)

##### Python prototype

[errstring] PositionerBacklashSet (PositionerName, BacklashValue)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) BacklashValue: BacklashValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerBacklashSet command which is used to Set backlash value. Refer to the XPS Programmer's manual to get the command description.

### PositionerBacklashGet

#### Syntax

##### C# prototype

int PositionerBacklashGet(string PositionerName, out double BacklashValue, out string BacklaskStatus, out string errstring)

##### Python prototype

[BacklashValue, BacklaskStatus, errstring] PositionerBacklashGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) BacklashValue: BacklashValue

(string) BacklaskStatus: BacklaskStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerBacklashGet command which is used to Read backlash value and status. Refer to the XPS Programmer's manual to get the command description.

### PositionerBacklashEnable

#### Syntax

##### C# prototype

int PositionerBacklashEnable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerBacklashEnable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerBacklashEnable command which is used to Enable the backlash. Refer to the XPS Programmer's manual to get the command description.

### PositionerBacklashDisable

#### Syntax

##### C# prototype

int PositionerBacklashDisable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerBacklashDisable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerBacklashDisable command which is used to Disable the backlash. Refer to the XPS Programmer's manual to get the command description.

### PositionerPreCorrectorExcitationSignalGet

#### Syntax

##### C# prototype

int PositionerPreCorrectorExcitationSignalGet(string PositionerName, out double Frequency, out double Amplitude, out double Time, out string errstring)

##### Python prototype

[Frequency, Amplitude, Time, errstring] PositionerPreCorrectorExcitationSignalGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) Frequency: Frequency

(double) Amplitude: Amplitude

(double) Time: Time

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPreCorrectorExcitationSignalGet command which is used to Get pre-corrector excitation signal mode. Refer to the XPS Programmer's manual to get the command description.

### PositionerPreCorrectorExcitationSignalSet

#### Syntax

##### C# prototype

int PositionerPreCorrectorExcitationSignalSet(string PositionerName, double Frequency, double Amplitude, double Time, out string errstring)

##### Python prototype

[errstring] PositionerPreCorrectorExcitationSignalSet (PositionerName, Frequency, Amplitude, Time)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) Frequency: Frequency

(double) Amplitude: Amplitude

(double) Time: Time

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPreCorrectorExcitationSignalSet command which is used to Set pre-corrector excitation signal mode. Refer to the XPS Programmer's manual to get the command description.

### PositionerMotionDoneGet

#### Syntax

##### C# prototype

int PositionerMotionDoneGet(string PositionerName, out double PositionWindow, out double VelocityWindow, out double CheckingTime, out double MeanPeriod, out double TimeOut, out string errstring)

##### Python prototype

[PositionWindow, VelocityWindow, CheckingTime, MeanPeriod, TimeOut, errstring] PositionerMotionDoneGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) PositionWindow: PositionWindow

(double) VelocityWindow: VelocityWindow

(double) CheckingTime: CheckingTime

(double) MeanPeriod: MeanPeriod

(double) TimeOut: TimeOut

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerMotionDoneGet command which is used to Read motion done parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerMotionDoneSet

#### Syntax

##### C# prototype

int PositionerMotionDoneSet(string PositionerName, double PositionWindow, double VelocityWindow, double CheckingTime, double MeanPeriod, double TimeOut, out string errstring)

##### Python prototype

[errstring] PositionerMotionDoneSet (PositionerName, PositionWindow, VelocityWindow, CheckingTime, MeanPeriod, TimeOut)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) PositionWindow: PositionWindow

(double) VelocityWindow: VelocityWindow

(double) CheckingTime: CheckingTime

(double) MeanPeriod: MeanPeriod

(double) TimeOut: TimeOut

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerMotionDoneSet command which is used to Update motion done parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerHardInterpolatorFactorGet

#### Syntax

##### C# prototype

int PositionerHardInterpolatorFactorGet(string PositionerName, out int InterpolationFactor, out string errstring)

##### Python prototype

[InterpolationFactor, errstring] PositionerHardInterpolatorFactorGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(int) InterpolationFactor: InterpolationFactor

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerHardInterpolatorFactorGet command which is used to Get hard interpolator parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerHardInterpolatorFactorSet

#### Syntax

##### C# prototype

int PositionerHardInterpolatorFactorSet(string PositionerName, int InterpolationFactor, out string errstring)

##### Python prototype

[errstring] PositionerHardInterpolatorFactorSet (PositionerName, InterpolationFactor)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) InterpolationFactor: InterpolationFactor

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerHardInterpolatorFactorSet command which is used to Set hard interpolator parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerHardInterpolatorPositionGet

#### Syntax

##### C# prototype

int PositionerHardInterpolatorPositionGet(string PositionerName, out double Position, out string errstring)

##### Python prototype

[Position, errstring] PositionerHardInterpolatorPositionGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) Position: Position

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerHardInterpolatorPositionGet command which is used to Read external latch position. Refer to the XPS Programmer's manual to get the command description.

### PositionerDriverFiltersGet

#### Syntax

##### C# prototype

int PositionerDriverFiltersGet(string PositionerName, out double KI, out double NotchFrequency, out double NotchBandwidth, out double NotchGain, out double LowpassFrequency, out string errstring)

##### Python prototype

[KI, NotchFrequency, NotchBandwidth, NotchGain, LowpassFrequency, errstring] PositionerDriverFiltersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) KI: KI

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

(double) LowpassFrequency: LowpassFrequency

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerDriverFiltersGet command which is used to Get driver filters parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerDriverFiltersSet

#### Syntax

##### C# prototype

int PositionerDriverFiltersSet(string PositionerName, double KI, double NotchFrequency, double NotchBandwidth, double NotchGain, double LowpassFrequency, out string errstring)

##### Python prototype

[errstring] PositionerDriverFiltersSet (PositionerName, KI, NotchFrequency, NotchBandwidth, NotchGain, LowpassFrequency)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) KI: KI

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

(double) LowpassFrequency: LowpassFrequency

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerDriverFiltersSet command which is used to Set driver filters parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerDriverPositionOffsetsGet

#### Syntax

##### C# prototype

int PositionerDriverPositionOffsetsGet(string PositionerName, out double StagePositionOffset, out double GagePositionOffset, out string errstring)

##### Python prototype

[StagePositionOffset, GagePositionOffset, errstring] PositionerDriverPositionOffsetsGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) StagePositionOffset: StagePositionOffset

(double) GagePositionOffset: GagePositionOffset

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerDriverPositionOffsetsGet command which is used to Get driver stage and gage position offset. Refer to the XPS Programmer's manual to get the command description.

### PositionerPositionCompareGet

#### Syntax

##### C# prototype

int PositionerPositionCompareGet(string PositionerName, out double MinimumPosition, out double MaximumPosition, out double PositionStep, out bool EnableState, out string errstring)

##### Python prototype

[MinimumPosition, MaximumPosition, PositionStep, EnableState, errstring] PositionerPositionCompareGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) MinimumPosition: MinimumPosition

(double) MaximumPosition: MaximumPosition

(double) PositionStep: PositionStep

(bool) EnableState: EnableState

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPositionCompareGet command which is used to Read position compare parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerPositionCompareSet

#### Syntax

##### C# prototype

int PositionerPositionCompareSet(string PositionerName, double MinimumPosition, double MaximumPosition, double PositionStep, out string errstring)

##### Python prototype

[errstring] PositionerPositionCompareSet (PositionerName, MinimumPosition, MaximumPosition, PositionStep)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) MinimumPosition: MinimumPosition

(double) MaximumPosition: MaximumPosition

(double) PositionStep: PositionStep

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPositionCompareSet command which is used to Set position compare parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerPositionCompareEnable

#### Syntax

##### C# prototype

int PositionerPositionCompareEnable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerPositionCompareEnable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPositionCompareEnable command which is used to Enable position compare. Refer to the XPS Programmer's manual to get the command description.

### PositionerPositionCompareDisable

#### Syntax

##### C# prototype

int PositionerPositionCompareDisable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerPositionCompareDisable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPositionCompareDisable command which is used to Disable position compare. Refer to the XPS Programmer's manual to get the command description.

### PositionerPositionComparePulseParametersGet

#### Syntax

##### C# prototype

int PositionerPositionComparePulseParametersGet(string PositionerName, out double PCOPulseWidth, out double EncoderSettlingTime, out string errstring)

##### Python prototype

[PCOPulseWidth, EncoderSettlingTime, errstring] PositionerPositionComparePulseParametersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) PCOPulseWidth: PCOPulseWidth

(double) EncoderSettlingTime: EncoderSettlingTime

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPositionComparePulseParametersGet command which is used to Get position compare PCO pulse parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerPositionComparePulseParametersSet

#### Syntax

##### C# prototype

int PositionerPositionComparePulseParametersSet(string PositionerName, double PCOPulseWidth, double EncoderSettlingTime, out string errstring)

##### Python prototype

[errstring] PositionerPositionComparePulseParametersSet (PositionerName, PCOPulseWidth, EncoderSettlingTime)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) PCOPulseWidth: PCOPulseWidth

(double) EncoderSettlingTime: EncoderSettlingTime

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPositionComparePulseParametersSet command which is used to Set position compare PCO pulse parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOAbort

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOAbort(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerCompensatedFastPCOAbort (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOAbort command which is used to Abort fast compensated PCO mode. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOCurrentStatusGet

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOCurrentStatusGet(string PositionerName, out int Status, out string errstring)

##### Python prototype

[Status, errstring] PositionerCompensatedFastPCOCurrentStatusGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(int) Status: Status

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOCurrentStatusGet command which is used to Get current status of fast compensated PCO mode. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOEnable

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOEnable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerCompensatedFastPCOEnable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOEnable command which is used to Enable fast compensated PCO mode execution. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOFromFile

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOFromFile(string PositionerName, string DataFileName, out string errstring)

##### Python prototype

[errstring] PositionerCompensatedFastPCOFromFile (PositionerName, DataFileName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) DataFileName: DataFileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOFromFile command which is used to Load file to fast compensated PCO data buffer. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOLoadToMemory

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOLoadToMemory(string PositionerName, string DataLines, out string errstring)

##### Python prototype

[errstring] PositionerCompensatedFastPCOLoadToMemory (PositionerName, DataLines)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) DataLines: DataLines

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOLoadToMemory command which is used to Load data lines to fast compensated PCO data buffer. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOMemoryReset

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOMemoryReset(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerCompensatedFastPCOMemoryReset (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOMemoryReset command which is used to Reset fast compensated PCO data buffer. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOPrepare

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOPrepare(string PositionerName, int ScanDirection, double[] StartPosition, int nbItems, out string errstring)

##### Python prototype

[errstring] PositionerCompensatedFastPCOPrepare (PositionerName, ScanDirection, StartPosition, nbItems)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) ScanDirection: ScanDirection

(double[]) StartPosition: StartPosition

(int) nbItems: nbItems

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOPrepare command which is used to Prepare data for fast compensated PCO mode. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOSet

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOSet(string PositionerName, double Start, double Stop, double Step, out string errstring)

##### Python prototype

[errstring] PositionerCompensatedFastPCOSet (PositionerName, Start, Stop, Step)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) Start: Start

(double) Stop: Stop

(double) Step: Step

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOSet command which is used to Set data to compensated PCO data buffer. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOPulseParametersGet

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOPulseParametersGet(string PositionerName, out double PulseWidth, out int PulsePolarity, out bool PulseToggle, out string errstring)

##### Python prototype

[PulseWidth, PulsePolarity, PulseToggle, errstring] PositionerCompensatedFastPCOPulseParametersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) PulseWidth: PulseWidth

(int) PulsePolarity: PulsePolarity

(bool) PulseToggle: PulseToggle

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOPulseParametersGet command which is used to Get pulse configuration to compensated PCO. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensatedFastPCOPulseParametersSet

#### Syntax

##### C# prototype

int PositionerCompensatedFastPCOPulseParametersSet(string PositionerName, double PulseWidth, int PulsePolarity, bool PulseToggle, out string errstring)

##### Python prototype

[errstring] PositionerCompensatedFastPCOPulseParametersSet (PositionerName, PulseWidth, PulsePolarity, PulseToggle)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) PulseWidth: PulseWidth

(int) PulsePolarity: PulsePolarity

(bool) PulseToggle: PulseToggle

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensatedFastPCOPulseParametersSet command which is used to Set pulse configuration to compensated PCO. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDFFAccelerationSet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDFFAccelerationSet(string PositionerName, bool ClosedLoopStatus, double KP, double KI, double KD, double KS, double IntegrationTime, double DerivativeFilterCutOffFrequency, double GKP, double GKI, double GKD, double KForm, double KFeedForwardAcceleration, double KFeedForwardJerk, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorPIDFFAccelerationSet (PositionerName, ClosedLoopStatus, KP, KI, KD, KS, IntegrationTime, DerivativeFilterCutOffFrequency, GKP, GKI, GKD, KForm, KFeedForwardAcceleration, KFeedForwardJerk)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KD: KD

(double) KS: KS

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) GKP: GKP

(double) GKI: GKI

(double) GKD: GKD

(double) KForm: KForm

(double) KFeedForwardAcceleration: KFeedForwardAcceleration

(double) KFeedForwardJerk: KFeedForwardJerk

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDFFAccelerationSet command which is used to Update corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDFFAccelerationGet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDFFAccelerationGet(string PositionerName, out bool ClosedLoopStatus, out double KP, out double KI, out double KD, out double KS, out double IntegrationTime, out double DerivativeFilterCutOffFrequency, out double GKP, out double GKI, out double GKD, out double KForm, out double KFeedForwardAcceleration, out double KFeedForwardJerk, out string errstring)

##### Python prototype

[ClosedLoopStatus, KP, KI, KD, KS, IntegrationTime, DerivativeFilterCutOffFrequency, GKP, GKI, GKD, KForm, KFeedForwardAcceleration, KFeedForwardJerk, errstring] PositionerCorrectorPIDFFAccelerationGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KD: KD

(double) KS: KS

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) GKP: GKP

(double) GKI: GKI

(double) GKD: GKD

(double) KForm: KForm

(double) KFeedForwardAcceleration: KFeedForwardAcceleration

(double) KFeedForwardJerk: KFeedForwardJerk

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDFFAccelerationGet command which is used to Read corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDAccelerationFilterSet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDAccelerationFilterSet(string PositionerName, bool FilterControlStatus, double KD, double DerivativeFilterCutOffFrequency, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorPIDAccelerationFilterSet (PositionerName, FilterControlStatus, KD, DerivativeFilterCutOffFrequency)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(bool) FilterControlStatus: FilterControlStatus

(double) KD: KD

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDAccelerationFilterSet command which is used to Update PID acceleration corrector filter parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDAccelerationFilterGet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDAccelerationFilterGet(string PositionerName, out bool FilterControlStatus, out double KD, out double DerivativeFilterCutOffFrequency, out string errstring)

##### Python prototype

[FilterControlStatus, KD, DerivativeFilterCutOffFrequency, errstring] PositionerCorrectorPIDAccelerationFilterGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(bool) FilterControlStatus: FilterControlStatus

(double) KD: KD

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDAccelerationFilterGet command which is used to Read PID acceleration corrector filter parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorDamperFilterSet

#### Syntax

##### C# prototype

int PositionerCorrectorDamperFilterSet(string PositionerName, double CutOffFrequency, double DampingFactor, double Gain, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorDamperFilterSet (PositionerName, CutOffFrequency, DampingFactor, Gain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) CutOffFrequency: CutOffFrequency

(double) DampingFactor: DampingFactor

(double) Gain: Gain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorDamperFilterSet command which is used to Update corrector damper filter parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorDamperFilterGet

#### Syntax

##### C# prototype

int PositionerCorrectorDamperFilterGet(string PositionerName, out double CutOffFrequency, out double DampingFactor, out double Gain, out string errstring)

##### Python prototype

[CutOffFrequency, DampingFactor, Gain, errstring] PositionerCorrectorDamperFilterGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) CutOffFrequency: CutOffFrequency

(double) DampingFactor: DampingFactor

(double) Gain: Gain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorDamperFilterGet command which is used to Read corrector damper filter parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPostFFSet

#### Syntax

##### C# prototype

int PositionerCorrectorPostFFSet(string PositionerName, double PostKFeedForwardAcceleration, double PostKFeedForwardJerk, double PostKFeedForwardSlope, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorPostFFSet (PositionerName, PostKFeedForwardAcceleration, PostKFeedForwardJerk, PostKFeedForwardSlope)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) PostKFeedForwardAcceleration: PostKFeedForwardAcceleration

(double) PostKFeedForwardJerk: PostKFeedForwardJerk

(double) PostKFeedForwardSlope: PostKFeedForwardSlope

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPostFFSet command which is used to Update post feedforward parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPostFFGet

#### Syntax

##### C# prototype

int PositionerCorrectorPostFFGet(string PositionerName, out double PostKFeedForwardAcceleration, out double PostKFeedForwardJerk, out double PostKFeedForwardSlope, out string errstring)

##### Python prototype

[PostKFeedForwardAcceleration, PostKFeedForwardJerk, PostKFeedForwardSlope, errstring] PositionerCorrectorPostFFGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) PostKFeedForwardAcceleration: PostKFeedForwardAcceleration

(double) PostKFeedForwardJerk: PostKFeedForwardJerk

(double) PostKFeedForwardSlope: PostKFeedForwardSlope

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPostFFGet command which is used to Read post feedforward parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDBaseSet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDBaseSet(string PositionerName, double MovingMass, double StaticMass, double Viscosity, double Stiffness, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorPIDBaseSet (PositionerName, MovingMass, StaticMass, Viscosity, Stiffness)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) MovingMass: MovingMass

(double) StaticMass: StaticMass

(double) Viscosity: Viscosity

(double) Stiffness: Stiffness

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDBaseSet command which is used to Update PIDBase parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDBaseGet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDBaseGet(string PositionerName, out double MovingMass, out double StaticMass, out double Viscosity, out double Stiffness, out string errstring)

##### Python prototype

[MovingMass, StaticMass, Viscosity, Stiffness, errstring] PositionerCorrectorPIDBaseGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) MovingMass: MovingMass

(double) StaticMass: StaticMass

(double) Viscosity: Viscosity

(double) Stiffness: Stiffness

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDBaseGet command which is used to Read PIDBase parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerPositionCompareScanAccelerationLimitGet

#### Syntax

##### C# prototype

int PositionerPositionCompareScanAccelerationLimitGet(string PositionerName, out double ScanAccelerationLimit, out string errstring)

##### Python prototype

[ScanAccelerationLimit, errstring] PositionerPositionCompareScanAccelerationLimitGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) ScanAccelerationLimit: ScanAccelerationLimit

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPositionCompareScanAccelerationLimitGet command which is used to Get position compare scan acceleration limit. Refer to the XPS Programmer's manual to get the command description.

### PositionerPositionCompareScanAccelerationLimitSet

#### Syntax

##### C# prototype

int PositionerPositionCompareScanAccelerationLimitSet(string PositionerName, double ScanAccelerationLimit, out string errstring)

##### Python prototype

[errstring] PositionerPositionCompareScanAccelerationLimitSet (PositionerName, ScanAccelerationLimit)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) ScanAccelerationLimit: ScanAccelerationLimit

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerPositionCompareScanAccelerationLimitSet command which is used to Set position compare scan acceleration limit. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorP2IDFFAccelerationSet

#### Syntax

##### C# prototype

int PositionerCorrectorP2IDFFAccelerationSet(string PositionerName, bool ClosedLoopStatus, double KP, double KI, double KI2, double KD, double KS, double IntegrationTime, double DerivativeFilterCutOffFrequency, double GKP, double GKI, double GKD, double KForm, double KFeedForwardAcceleration, double KFeedForwardJerk, double SetpointPositionDelay, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorP2IDFFAccelerationSet (PositionerName, ClosedLoopStatus, KP, KI, KI2, KD, KS, IntegrationTime, DerivativeFilterCutOffFrequency, GKP, GKI, GKD, KForm, KFeedForwardAcceleration, KFeedForwardJerk, SetpointPositionDelay)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KI2: KI2

(double) KD: KD

(double) KS: KS

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) GKP: GKP

(double) GKI: GKI

(double) GKD: GKD

(double) KForm: KForm

(double) KFeedForwardAcceleration: KFeedForwardAcceleration

(double) KFeedForwardJerk: KFeedForwardJerk

(double) SetpointPositionDelay: SetpointPositionDelay

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorP2IDFFAccelerationSet command which is used to Update corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorP2IDFFAccelerationGet

#### Syntax

##### C# prototype

int PositionerCorrectorP2IDFFAccelerationGet(string PositionerName, out bool ClosedLoopStatus, out double KP, out double KI, out double KI2, out double KD, out double KS, out double IntegrationTime, out double DerivativeFilterCutOffFrequency, out double GKP, out double GKI, out double GKD, out double KForm, out double KFeedForwardAcceleration, out double KFeedForwardJerk, out double SetpointPositionDelay, out string errstring)

##### Python prototype

[ClosedLoopStatus, KP, KI, KI2, KD, KS, IntegrationTime, DerivativeFilterCutOffFrequency, GKP, GKI, GKD, KForm, KFeedForwardAcceleration, KFeedForwardJerk, SetpointPositionDelay, errstring] PositionerCorrectorP2IDFFAccelerationGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KI2: KI2

(double) KD: KD

(double) KS: KS

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) GKP: GKP

(double) GKI: GKI

(double) GKD: GKD

(double) KForm: KForm

(double) KFeedForwardAcceleration: KFeedForwardAcceleration

(double) KFeedForwardJerk: KFeedForwardJerk

(double) SetpointPositionDelay: SetpointPositionDelay

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorP2IDFFAccelerationGet command which is used to Read corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDDualFFVoltageSet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDDualFFVoltageSet(string PositionerName, bool ClosedLoopStatus, double KP, double KI, double KD, double KS, double IntegrationTime, double DerivativeFilterCutOffFrequency, double GKP, double GKI, double GKD, double KForm, double KFeedForwardVelocity, double KFeedForwardAcceleration, double Friction, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorPIDDualFFVoltageSet (PositionerName, ClosedLoopStatus, KP, KI, KD, KS, IntegrationTime, DerivativeFilterCutOffFrequency, GKP, GKI, GKD, KForm, KFeedForwardVelocity, KFeedForwardAcceleration, Friction)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KD: KD

(double) KS: KS

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) GKP: GKP

(double) GKI: GKI

(double) GKD: GKD

(double) KForm: KForm

(double) KFeedForwardVelocity: KFeedForwardVelocity

(double) KFeedForwardAcceleration: KFeedForwardAcceleration

(double) Friction: Friction

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDDualFFVoltageSet command which is used to Update corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDDualFFVoltageGet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDDualFFVoltageGet(string PositionerName, out bool ClosedLoopStatus, out double KP, out double KI, out double KD, out double KS, out double IntegrationTime, out double DerivativeFilterCutOffFrequency, out double GKP, out double GKI, out double GKD, out double KForm, out double KFeedForwardVelocity, out double KFeedForwardAcceleration, out double Friction, out string errstring)

##### Python prototype

[ClosedLoopStatus, KP, KI, KD, KS, IntegrationTime, DerivativeFilterCutOffFrequency, GKP, GKI, GKD, KForm, KFeedForwardVelocity, KFeedForwardAcceleration, Friction, errstring] PositionerCorrectorPIDDualFFVoltageGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KD: KD

(double) KS: KS

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) GKP: GKP

(double) GKI: GKI

(double) GKD: GKD

(double) KForm: KForm

(double) KFeedForwardVelocity: KFeedForwardVelocity

(double) KFeedForwardAcceleration: KFeedForwardAcceleration

(double) Friction: Friction

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDDualFFVoltageGet command which is used to Read corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDFFVelocitySet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDFFVelocitySet(string PositionerName, bool ClosedLoopStatus, double KP, double KI, double KD, double KS, double IntegrationTime, double DerivativeFilterCutOffFrequency, double GKP, double GKI, double GKD, double KForm, double KFeedForwardVelocity, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorPIDFFVelocitySet (PositionerName, ClosedLoopStatus, KP, KI, KD, KS, IntegrationTime, DerivativeFilterCutOffFrequency, GKP, GKI, GKD, KForm, KFeedForwardVelocity)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KD: KD

(double) KS: KS

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) GKP: GKP

(double) GKI: GKI

(double) GKD: GKD

(double) KForm: KForm

(double) KFeedForwardVelocity: KFeedForwardVelocity

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDFFVelocitySet command which is used to Update corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIDFFVelocityGet

#### Syntax

##### C# prototype

int PositionerCorrectorPIDFFVelocityGet(string PositionerName, out bool ClosedLoopStatus, out double KP, out double KI, out double KD, out double KS, out double IntegrationTime, out double DerivativeFilterCutOffFrequency, out double GKP, out double GKI, out double GKD, out double KForm, out double KFeedForwardVelocity, out string errstring)

##### Python prototype

[ClosedLoopStatus, KP, KI, KD, KS, IntegrationTime, DerivativeFilterCutOffFrequency, GKP, GKI, GKD, KForm, KFeedForwardVelocity, errstring] PositionerCorrectorPIDFFVelocityGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KD: KD

(double) KS: KS

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) GKP: GKP

(double) GKI: GKI

(double) GKD: GKD

(double) KForm: KForm

(double) KFeedForwardVelocity: KFeedForwardVelocity

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIDFFVelocityGet command which is used to Read corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIPositionSet

#### Syntax

##### C# prototype

int PositionerCorrectorPIPositionSet(string PositionerName, bool ClosedLoopStatus, double KP, double KI, double IntegrationTime, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorPIPositionSet (PositionerName, ClosedLoopStatus, KP, KI, IntegrationTime)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) IntegrationTime: IntegrationTime

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIPositionSet command which is used to Update corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorPIPositionGet

#### Syntax

##### C# prototype

int PositionerCorrectorPIPositionGet(string PositionerName, out bool ClosedLoopStatus, out double KP, out double KI, out double IntegrationTime, out string errstring)

##### Python prototype

[ClosedLoopStatus, KP, KI, IntegrationTime, errstring] PositionerCorrectorPIPositionGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) IntegrationTime: IntegrationTime

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorPIPositionGet command which is used to Read corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorSR1AccelerationSet

#### Syntax

##### C# prototype

int PositionerCorrectorSR1AccelerationSet(string PositionerName, bool ClosedLoopStatus, double KP, double KI, double KV, double ObserverFrequency, double CompensationGainVelocity, double CompensationGainAcceleration, double CompensationGainJerk, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorSR1AccelerationSet (PositionerName, ClosedLoopStatus, KP, KI, KV, ObserverFrequency, CompensationGainVelocity, CompensationGainAcceleration, CompensationGainJerk)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KV: KV

(double) ObserverFrequency: ObserverFrequency

(double) CompensationGainVelocity: CompensationGainVelocity

(double) CompensationGainAcceleration: CompensationGainAcceleration

(double) CompensationGainJerk: CompensationGainJerk

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorSR1AccelerationSet command which is used to Update corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorSR1AccelerationGet

#### Syntax

##### C# prototype

int PositionerCorrectorSR1AccelerationGet(string PositionerName, out bool ClosedLoopStatus, out double KP, out double KI, out double KV, out double ObserverFrequency, out double CompensationGainVelocity, out double CompensationGainAcceleration, out double CompensationGainJerk, out string errstring)

##### Python prototype

[ClosedLoopStatus, KP, KI, KV, ObserverFrequency, CompensationGainVelocity, CompensationGainAcceleration, CompensationGainJerk, errstring] PositionerCorrectorSR1AccelerationGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KV: KV

(double) ObserverFrequency: ObserverFrequency

(double) CompensationGainVelocity: CompensationGainVelocity

(double) CompensationGainAcceleration: CompensationGainAcceleration

(double) CompensationGainJerk: CompensationGainJerk

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorSR1AccelerationGet command which is used to Read corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorSR1ObserverAccelerationSet

#### Syntax

##### C# prototype

int PositionerCorrectorSR1ObserverAccelerationSet(string PositionerName, double ParameterA, double ParameterB, double ParameterC, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorSR1ObserverAccelerationSet (PositionerName, ParameterA, ParameterB, ParameterC)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) ParameterA: ParameterA

(double) ParameterB: ParameterB

(double) ParameterC: ParameterC

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorSR1ObserverAccelerationSet command which is used to Update SR1 corrector observer parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorSR1ObserverAccelerationGet

#### Syntax

##### C# prototype

int PositionerCorrectorSR1ObserverAccelerationGet(string PositionerName, out double ParameterA, out double ParameterB, out double ParameterC, out string errstring)

##### Python prototype

[ParameterA, ParameterB, ParameterC, errstring] PositionerCorrectorSR1ObserverAccelerationGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) ParameterA: ParameterA

(double) ParameterB: ParameterB

(double) ParameterC: ParameterC

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorSR1ObserverAccelerationGet command which is used to Read SR1 corrector observer parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorSR1OffsetAccelerationSet

#### Syntax

##### C# prototype

int PositionerCorrectorSR1OffsetAccelerationSet(string PositionerName, double AccelerationOffset, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorSR1OffsetAccelerationSet (PositionerName, AccelerationOffset)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) AccelerationOffset: AccelerationOffset

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorSR1OffsetAccelerationSet command which is used to Update SR1 corrector output acceleration offset. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorSR1OffsetAccelerationGet

#### Syntax

##### C# prototype

int PositionerCorrectorSR1OffsetAccelerationGet(string PositionerName, out double AccelerationOffset, out string errstring)

##### Python prototype

[AccelerationOffset, errstring] PositionerCorrectorSR1OffsetAccelerationGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) AccelerationOffset: AccelerationOffset

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorSR1OffsetAccelerationGet command which is used to Read SR1 corrector output acceleration offset. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorDualSet

#### Syntax

##### C# prototype

int PositionerCorrectorDualSet(string PositionerName, bool ClosedLoopStatus, double KP, double KI, double KD, double IntegrationTime, double DerivativeFilterCutOffFrequency, double KFeedForwardAcceleration, double KFeedForwardJerk, double AntiWindUpTime, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorDualSet (PositionerName, ClosedLoopStatus, KP, KI, KD, IntegrationTime, DerivativeFilterCutOffFrequency, KFeedForwardAcceleration, KFeedForwardJerk, AntiWindUpTime)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KD: KD

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) KFeedForwardAcceleration: KFeedForwardAcceleration

(double) KFeedForwardJerk: KFeedForwardJerk

(double) AntiWindUpTime: AntiWindUpTime

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorDualSet command which is used to Update dual corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorDualGet

#### Syntax

##### C# prototype

int PositionerCorrectorDualGet(string PositionerName, out bool ClosedLoopStatus, out double KP, out double KI, out double KD, out double IntegrationTime, out double DerivativeFilterCutOffFrequency, out double KFeedForwardAcceleration, out double KFeedForwardJerk, out double AntiWindUpTime, out string errstring)

##### Python prototype

[ClosedLoopStatus, KP, KI, KD, IntegrationTime, DerivativeFilterCutOffFrequency, KFeedForwardAcceleration, KFeedForwardJerk, AntiWindUpTime, errstring] PositionerCorrectorDualGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(bool) ClosedLoopStatus: ClosedLoopStatus

(double) KP: KP

(double) KI: KI

(double) KD: KD

(double) IntegrationTime: IntegrationTime

(double) DerivativeFilterCutOffFrequency: DerivativeFilterCutOffFrequency

(double) KFeedForwardAcceleration: KFeedForwardAcceleration

(double) KFeedForwardJerk: KFeedForwardJerk

(double) AntiWindUpTime: AntiWindUpTime

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorDualGet command which is used to Read dual corrector parameters. Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationDualLoopNotchFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationDualLoopNotchFilterSet(string PositionerName, int NotchNumber, double NotchFrequency, double NotchBandwidth, double NotchGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationDualLoopNotchFilterSet (PositionerName, NotchNumber, NotchFrequency, NotchBandwidth, NotchGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchNumber: NotchNumber

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationDualLoopNotchFilterSet command which is used to Update Notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationDualLoopNotchFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationDualLoopNotchFilterGet(string PositionerName, int NotchNumber, out double NotchFrequency, out double NotchBandwidth, out double NotchGain, out string errstring)

##### Python prototype

[NotchFrequency, NotchBandwidth, NotchGain, errstring] PositionerCompensationDualLoopNotchFilterGet (PositionerName, NotchNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchNumber: NotchNumber

##### Output parameters

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationDualLoopNotchFilterGet command which is used to Read Notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationDualLoopPhaseCorrectionFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationDualLoopPhaseCorrectionFilterSet(string PositionerName, int PhaseCorrectionFilterNumber, double PhaseCorrectionFn, double PhaseCorrectionFd, double PhaseCorrectionGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationDualLoopPhaseCorrectionFilterSet (PositionerName, PhaseCorrectionFilterNumber, PhaseCorrectionFn, PhaseCorrectionFd, PhaseCorrectionGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) PhaseCorrectionFilterNumber: PhaseCorrectionFilterNumber

(double) PhaseCorrectionFn: PhaseCorrectionFn

(double) PhaseCorrectionFd: PhaseCorrectionFd

(double) PhaseCorrectionGain: PhaseCorrectionGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationDualLoopPhaseCorrectionFilterSet command which is used to Update one phase correction filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationDualLoopPhaseCorrectionFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationDualLoopPhaseCorrectionFilterGet(string PositionerName, int PhaseCorrectionFilterNumber, out double PhaseCorrectionFn, out double PhaseCorrectionFd, out double PhaseCorrectionGain, out string errstring)

##### Python prototype

[PhaseCorrectionFn, PhaseCorrectionFd, PhaseCorrectionGain, errstring] PositionerCompensationDualLoopPhaseCorrectionFilterGet (PositionerName, PhaseCorrectionFilterNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) PhaseCorrectionFilterNumber: PhaseCorrectionFilterNumber

##### Output parameters

(double) PhaseCorrectionFn: PhaseCorrectionFn

(double) PhaseCorrectionFd: PhaseCorrectionFd

(double) PhaseCorrectionGain: PhaseCorrectionGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationDualLoopPhaseCorrectionFilterGet command which is used to Read one phase correction filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationNotchFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationNotchFilterSet(string PositionerName, int NotchNumber, double NotchFrequency, double NotchBandwidth, double NotchGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationNotchFilterSet (PositionerName, NotchNumber, NotchFrequency, NotchBandwidth, NotchGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchNumber: NotchNumber

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationNotchFilterSet command which is used to Update Notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationNotchFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationNotchFilterGet(string PositionerName, int NotchNumber, out double NotchFrequency, out double NotchBandwidth, out double NotchGain, out string errstring)

##### Python prototype

[NotchFrequency, NotchBandwidth, NotchGain, errstring] PositionerCompensationNotchFilterGet (PositionerName, NotchNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchNumber: NotchNumber

##### Output parameters

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationNotchFilterGet command which is used to Read Notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPhaseCorrectionFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPhaseCorrectionFilterSet(string PositionerName, int PhaseCorrectionFilterNumber, double PhaseCorrectionFn, double PhaseCorrectionFd, double PhaseCorrectionGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPhaseCorrectionFilterSet (PositionerName, PhaseCorrectionFilterNumber, PhaseCorrectionFn, PhaseCorrectionFd, PhaseCorrectionGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) PhaseCorrectionFilterNumber: PhaseCorrectionFilterNumber

(double) PhaseCorrectionFn: PhaseCorrectionFn

(double) PhaseCorrectionFd: PhaseCorrectionFd

(double) PhaseCorrectionGain: PhaseCorrectionGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPhaseCorrectionFilterSet command which is used to Update one phase correction filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPhaseCorrectionFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPhaseCorrectionFilterGet(string PositionerName, int PhaseCorrectionFilterNumber, out double PhaseCorrectionFn, out double PhaseCorrectionFd, out double PhaseCorrectionGain, out string errstring)

##### Python prototype

[PhaseCorrectionFn, PhaseCorrectionFd, PhaseCorrectionGain, errstring] PositionerCompensationPhaseCorrectionFilterGet (PositionerName, PhaseCorrectionFilterNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) PhaseCorrectionFilterNumber: PhaseCorrectionFilterNumber

##### Output parameters

(double) PhaseCorrectionFn: PhaseCorrectionFn

(double) PhaseCorrectionFd: PhaseCorrectionFd

(double) PhaseCorrectionGain: PhaseCorrectionGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPhaseCorrectionFilterGet command which is used to Read one phase correction filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorNotchFiltersSet

#### Syntax

##### C# prototype

int PositionerCorrectorNotchFiltersSet(string PositionerName, double NotchFrequency1, double NotchBandwidth1, double NotchGain1, double NotchFrequency2, double NotchBandwidth2, double NotchGain2, out string errstring)

##### Python prototype

[errstring] PositionerCorrectorNotchFiltersSet (PositionerName, NotchFrequency1, NotchBandwidth1, NotchGain1, NotchFrequency2, NotchBandwidth2, NotchGain2)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) NotchFrequency1: NotchFrequency1

(double) NotchBandwidth1: NotchBandwidth1

(double) NotchGain1: NotchGain1

(double) NotchFrequency2: NotchFrequency2

(double) NotchBandwidth2: NotchBandwidth2

(double) NotchGain2: NotchGain2

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorNotchFiltersSet command which is used to Update filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCorrectorNotchFiltersGet

#### Syntax

##### C# prototype

int PositionerCorrectorNotchFiltersGet(string PositionerName, out double NotchFrequency1, out double NotchBandwidth1, out double NotchGain1, out double NotchFrequency2, out double NotchBandwidth2, out double NotchGain2, out string errstring)

##### Python prototype

[NotchFrequency1, NotchBandwidth1, NotchGain1, NotchFrequency2, NotchBandwidth2, NotchGain2, errstring] PositionerCorrectorNotchFiltersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) NotchFrequency1: NotchFrequency1

(double) NotchBandwidth1: NotchBandwidth1

(double) NotchGain1: NotchGain1

(double) NotchFrequency2: NotchFrequency2

(double) NotchBandwidth2: NotchBandwidth2

(double) NotchGain2: NotchGain2

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCorrectorNotchFiltersGet command which is used to Read filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPreFeedForwardFrequencyNotchFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPreFeedForwardFrequencyNotchFilterGet(string PositionerName, int NotchFrequencyNumber, out double NotchFrequency, out double NotchBandwidth, out double NotchGain, out string errstring)

##### Python prototype

[NotchFrequency, NotchBandwidth, NotchGain, errstring] PositionerCompensationPreFeedForwardFrequencyNotchFilterGet (PositionerName, NotchFrequencyNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchFrequencyNumber: NotchFrequencyNumber

##### Output parameters

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPreFeedForwardFrequencyNotchFilterGet command which is used to Read one frequency compensation notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPreFeedForwardFrequencyNotchFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPreFeedForwardFrequencyNotchFilterSet(string PositionerName, int NotchFrequencyNumber, double NotchFrequency, double NotchBandwidth, double NotchGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPreFeedForwardFrequencyNotchFilterSet (PositionerName, NotchFrequencyNumber, NotchFrequency, NotchBandwidth, NotchGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchFrequencyNumber: NotchFrequencyNumber

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPreFeedForwardFrequencyNotchFilterSet command which is used to Update one frequency compensation notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPreFeedForwardSpatialNotchFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPreFeedForwardSpatialNotchFilterGet(string PositionerName, int SpatialNotchNumber, out double SpatialNotchStep, out double SpatialNotchBandwidth, out double SpatialNotchGain, out string errstring)

##### Python prototype

[SpatialNotchStep, SpatialNotchBandwidth, SpatialNotchGain, errstring] PositionerCompensationPreFeedForwardSpatialNotchFilterGet (PositionerName, SpatialNotchNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) SpatialNotchNumber: SpatialNotchNumber

##### Output parameters

(double) SpatialNotchStep: SpatialNotchStep

(double) SpatialNotchBandwidth: SpatialNotchBandwidth

(double) SpatialNotchGain: SpatialNotchGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPreFeedForwardSpatialNotchFilterGet command which is used to Read one spatial periodic compensation notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPreFeedForwardSpatialNotchFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPreFeedForwardSpatialNotchFilterSet(string PositionerName, int SpatialNotchNumber, double SpatialNotchStep, double SpatialNotchBandwidth, double SpatialNotchGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPreFeedForwardSpatialNotchFilterSet (PositionerName, SpatialNotchNumber, SpatialNotchStep, SpatialNotchBandwidth, SpatialNotchGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) SpatialNotchNumber: SpatialNotchNumber

(double) SpatialNotchStep: SpatialNotchStep

(double) SpatialNotchBandwidth: SpatialNotchBandwidth

(double) SpatialNotchGain: SpatialNotchGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPreFeedForwardSpatialNotchFilterSet command which is used to Update one spatial periodic compensation notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPreFeedForwardPhaseCorrectionFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPreFeedForwardPhaseCorrectionFilterGet(string PositionerName, int PhaseCorrectionFilterNumber, out double PhaseCorrectionFn, out double PhaseCorrectionFd, out double PhaseCorrectionGain, out string errstring)

##### Python prototype

[PhaseCorrectionFn, PhaseCorrectionFd, PhaseCorrectionGain, errstring] PositionerCompensationPreFeedForwardPhaseCorrectionFilterGet (PositionerName, PhaseCorrectionFilterNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) PhaseCorrectionFilterNumber: PhaseCorrectionFilterNumber

##### Output parameters

(double) PhaseCorrectionFn: PhaseCorrectionFn

(double) PhaseCorrectionFd: PhaseCorrectionFd

(double) PhaseCorrectionGain: PhaseCorrectionGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPreFeedForwardPhaseCorrectionFilterGet command which is used to Read one phase correction filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPreFeedForwardPhaseCorrectionFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPreFeedForwardPhaseCorrectionFilterSet(string PositionerName, int PhaseCorrectionFilterNumber, double PhaseCorrectionFn, double PhaseCorrectionFd, double PhaseCorrectionGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPreFeedForwardPhaseCorrectionFilterSet (PositionerName, PhaseCorrectionFilterNumber, PhaseCorrectionFn, PhaseCorrectionFd, PhaseCorrectionGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) PhaseCorrectionFilterNumber: PhaseCorrectionFilterNumber

(double) PhaseCorrectionFn: PhaseCorrectionFn

(double) PhaseCorrectionFd: PhaseCorrectionFd

(double) PhaseCorrectionGain: PhaseCorrectionGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPreFeedForwardPhaseCorrectionFilterSet command which is used to Update one phase correction filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationFrequencyNotchsGet

#### Syntax

##### C# prototype

int PositionerCompensationFrequencyNotchsGet(string PositionerName, out double NotchFrequency1, out double NotchBandwidth1, out double NotchGain1, out double NotchFrequency2, out double NotchBandwidth2, out double NotchGain2, out double NotchFrequency3, out double NotchBandwidth3, out double NotchGain3, out string errstring)

##### Python prototype

[NotchFrequency1, NotchBandwidth1, NotchGain1, NotchFrequency2, NotchBandwidth2, NotchGain2, NotchFrequency3, NotchBandwidth3, NotchGain3, errstring] PositionerCompensationFrequencyNotchsGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) NotchFrequency1: NotchFrequency1

(double) NotchBandwidth1: NotchBandwidth1

(double) NotchGain1: NotchGain1

(double) NotchFrequency2: NotchFrequency2

(double) NotchBandwidth2: NotchBandwidth2

(double) NotchGain2: NotchGain2

(double) NotchFrequency3: NotchFrequency3

(double) NotchBandwidth3: NotchBandwidth3

(double) NotchGain3: NotchGain3

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationFrequencyNotchsGet command which is used to Read frequency compensation notch filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationFrequencyNotchsSet

#### Syntax

##### C# prototype

int PositionerCompensationFrequencyNotchsSet(string PositionerName, double NotchFrequency1, double NotchBandwidth1, double NotchGain1, double NotchFrequency2, double NotchBandwidth2, double NotchGain2, double NotchFrequency3, double NotchBandwidth3, double NotchGain3, out string errstring)

##### Python prototype

[errstring] PositionerCompensationFrequencyNotchsSet (PositionerName, NotchFrequency1, NotchBandwidth1, NotchGain1, NotchFrequency2, NotchBandwidth2, NotchGain2, NotchFrequency3, NotchBandwidth3, NotchGain3)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) NotchFrequency1: NotchFrequency1

(double) NotchBandwidth1: NotchBandwidth1

(double) NotchGain1: NotchGain1

(double) NotchFrequency2: NotchFrequency2

(double) NotchBandwidth2: NotchBandwidth2

(double) NotchGain2: NotchGain2

(double) NotchFrequency3: NotchFrequency3

(double) NotchBandwidth3: NotchBandwidth3

(double) NotchGain3: NotchGain3

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationFrequencyNotchsSet command which is used to Update frequency compensation notch filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationSpatialPeriodicNotchsGet

#### Syntax

##### C# prototype

int PositionerCompensationSpatialPeriodicNotchsGet(string PositionerName, out double SpatialNotchStep1, out double SpatialNotchBandwidth1, out double SpatialNotchGain1, out double SpatialNotchStep2, out double SpatialNotchBandwidth2, out double SpatialNotchGain2, out double SpatialNotchStep3, out double SpatialNotchBandwidth3, out double SpatialNotchGain3, out string errstring)

##### Python prototype

[SpatialNotchStep1, SpatialNotchBandwidth1, SpatialNotchGain1, SpatialNotchStep2, SpatialNotchBandwidth2, SpatialNotchGain2, SpatialNotchStep3, SpatialNotchBandwidth3, SpatialNotchGain3, errstring] PositionerCompensationSpatialPeriodicNotchsGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) SpatialNotchStep1: SpatialNotchStep1

(double) SpatialNotchBandwidth1: SpatialNotchBandwidth1

(double) SpatialNotchGain1: SpatialNotchGain1

(double) SpatialNotchStep2: SpatialNotchStep2

(double) SpatialNotchBandwidth2: SpatialNotchBandwidth2

(double) SpatialNotchGain2: SpatialNotchGain2

(double) SpatialNotchStep3: SpatialNotchStep3

(double) SpatialNotchBandwidth3: SpatialNotchBandwidth3

(double) SpatialNotchGain3: SpatialNotchGain3

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationSpatialPeriodicNotchsGet command which is used to Read spatial compensation notch filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationSpatialPeriodicNotchsSet

#### Syntax

##### C# prototype

int PositionerCompensationSpatialPeriodicNotchsSet(string PositionerName, double SpatialNotchStep1, double SpatialNotchBandwidth1, double SpatialNotchGain1, double SpatialNotchStep2, double SpatialNotchBandwidth2, double SpatialNotchGain2, double SpatialNotchStep3, double SpatialNotchBandwidth3, double SpatialNotchGain3, out string errstring)

##### Python prototype

[errstring] PositionerCompensationSpatialPeriodicNotchsSet (PositionerName, SpatialNotchStep1, SpatialNotchBandwidth1, SpatialNotchGain1, SpatialNotchStep2, SpatialNotchBandwidth2, SpatialNotchGain2, SpatialNotchStep3, SpatialNotchBandwidth3, SpatialNotchGain3)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) SpatialNotchStep1: SpatialNotchStep1

(double) SpatialNotchBandwidth1: SpatialNotchBandwidth1

(double) SpatialNotchGain1: SpatialNotchGain1

(double) SpatialNotchStep2: SpatialNotchStep2

(double) SpatialNotchBandwidth2: SpatialNotchBandwidth2

(double) SpatialNotchGain2: SpatialNotchGain2

(double) SpatialNotchStep3: SpatialNotchStep3

(double) SpatialNotchBandwidth3: SpatialNotchBandwidth3

(double) SpatialNotchGain3: SpatialNotchGain3

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationSpatialPeriodicNotchsSet command which is used to Update spatial compensation notch filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPostExcitationLowPassFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPostExcitationLowPassFilterGet(string PositionerName, out double CutOffFrequency, out string errstring)

##### Python prototype

[CutOffFrequency, errstring] PositionerCompensationPostExcitationLowPassFilterGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) CutOffFrequency: CutOffFrequency

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPostExcitationLowPassFilterGet command which is used to Read second order low-pass filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPostExcitationLowPassFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPostExcitationLowPassFilterSet(string PositionerName, double CutOffFrequency, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPostExcitationLowPassFilterSet (PositionerName, CutOffFrequency)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) CutOffFrequency: CutOffFrequency

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPostExcitationLowPassFilterSet command which is used to Update second order low-pass filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPostExcitationFrequencyNotchFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPostExcitationFrequencyNotchFilterGet(string PositionerName, int NotchFrequencyNumber, out double NotchFrequency, out double NotchBandwidth, out double NotchGain, out string errstring)

##### Python prototype

[NotchFrequency, NotchBandwidth, NotchGain, errstring] PositionerCompensationPostExcitationFrequencyNotchFilterGet (PositionerName, NotchFrequencyNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchFrequencyNumber: NotchFrequencyNumber

##### Output parameters

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPostExcitationFrequencyNotchFilterGet command which is used to Read one frequency compensation notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPostExcitationFrequencyNotchFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPostExcitationFrequencyNotchFilterSet(string PositionerName, int NotchFrequencyNumber, double NotchFrequency, double NotchBandwidth, double NotchGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPostExcitationFrequencyNotchFilterSet (PositionerName, NotchFrequencyNumber, NotchFrequency, NotchBandwidth, NotchGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchFrequencyNumber: NotchFrequencyNumber

(double) NotchFrequency: NotchFrequency

(double) NotchBandwidth: NotchBandwidth

(double) NotchGain: NotchGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPostExcitationFrequencyNotchFilterSet command which is used to Update one frequency compensation notch filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPostExcitationNotchModeFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPostExcitationNotchModeFilterGet(string PositionerName, int NotchModeNumber, out double NotchModeFr, out double NotchModeFa, out double NotchModeZr, out double NotchModeZa, out string errstring)

##### Python prototype

[NotchModeFr, NotchModeFa, NotchModeZr, NotchModeZa, errstring] PositionerCompensationPostExcitationNotchModeFilterGet (PositionerName, NotchModeNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchModeNumber: NotchModeNumber

##### Output parameters

(double) NotchModeFr: NotchModeFr

(double) NotchModeFa: NotchModeFa

(double) NotchModeZr: NotchModeZr

(double) NotchModeZa: NotchModeZa

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPostExcitationNotchModeFilterGet command which is used to Read a notch mode filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPostExcitationNotchModeFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPostExcitationNotchModeFilterSet(string PositionerName, int NotchModeNumber, double NotchModeFr, double NotchModeFa, double NotchModeZr, double NotchModeZa, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPostExcitationNotchModeFilterSet (PositionerName, NotchModeNumber, NotchModeFr, NotchModeFa, NotchModeZr, NotchModeZa)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) NotchModeNumber: NotchModeNumber

(double) NotchModeFr: NotchModeFr

(double) NotchModeFa: NotchModeFa

(double) NotchModeZr: NotchModeZr

(double) NotchModeZa: NotchModeZa

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPostExcitationNotchModeFilterSet command which is used to Update a notch mode filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPostExcitationPhaseCorrectionFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationPostExcitationPhaseCorrectionFilterGet(string PositionerName, int PhaseCorrectionFilterNumber, out double PhaseCorrectionFn, out double PhaseCorrectionFd, out double PhaseCorrectionGain, out string errstring)

##### Python prototype

[PhaseCorrectionFn, PhaseCorrectionFd, PhaseCorrectionGain, errstring] PositionerCompensationPostExcitationPhaseCorrectionFilterGet (PositionerName, PhaseCorrectionFilterNumber)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) PhaseCorrectionFilterNumber: PhaseCorrectionFilterNumber

##### Output parameters

(double) PhaseCorrectionFn: PhaseCorrectionFn

(double) PhaseCorrectionFd: PhaseCorrectionFd

(double) PhaseCorrectionGain: PhaseCorrectionGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPostExcitationPhaseCorrectionFilterGet command which is used to Read one phase correction filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPostExcitationPhaseCorrectionFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationPostExcitationPhaseCorrectionFilterSet(string PositionerName, int PhaseCorrectionFilterNumber, double PhaseCorrectionFn, double PhaseCorrectionFd, double PhaseCorrectionGain, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPostExcitationPhaseCorrectionFilterSet (PositionerName, PhaseCorrectionFilterNumber, PhaseCorrectionFn, PhaseCorrectionFd, PhaseCorrectionGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(int) PhaseCorrectionFilterNumber: PhaseCorrectionFilterNumber

(double) PhaseCorrectionFn: PhaseCorrectionFn

(double) PhaseCorrectionFd: PhaseCorrectionFd

(double) PhaseCorrectionGain: PhaseCorrectionGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPostExcitationPhaseCorrectionFilterSet command which is used to Update one phase correction filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationLowPassTwoFilterGet

#### Syntax

##### C# prototype

int PositionerCompensationLowPassTwoFilterGet(string PositionerName, out double CutOffFrequency, out string errstring)

##### Python prototype

[CutOffFrequency, errstring] PositionerCompensationLowPassTwoFilterGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) CutOffFrequency: CutOffFrequency

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationLowPassTwoFilterGet command which is used to Read second order low-pass filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationLowPassTwoFilterSet

#### Syntax

##### C# prototype

int PositionerCompensationLowPassTwoFilterSet(string PositionerName, double CutOffFrequency, out string errstring)

##### Python prototype

[errstring] PositionerCompensationLowPassTwoFilterSet (PositionerName, CutOffFrequency)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) CutOffFrequency: CutOffFrequency

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationLowPassTwoFilterSet command which is used to Update second order low-pass filter parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationNotchModeFiltersGet

#### Syntax

##### C# prototype

int PositionerCompensationNotchModeFiltersGet(string PositionerName, out double NotchModeFr1, out double NotchModeFa1, out double NotchModeZr1, out double NotchModeZa1, out double NotchModeFr2, out double NotchModeFa2, out double NotchModeZr2, out double NotchModeZa2, out string errstring)

##### Python prototype

[NotchModeFr1, NotchModeFa1, NotchModeZr1, NotchModeZa1, NotchModeFr2, NotchModeFa2, NotchModeZr2, NotchModeZa2, errstring] PositionerCompensationNotchModeFiltersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) NotchModeFr1: NotchModeFr1

(double) NotchModeFa1: NotchModeFa1

(double) NotchModeZr1: NotchModeZr1

(double) NotchModeZa1: NotchModeZa1

(double) NotchModeFr2: NotchModeFr2

(double) NotchModeFa2: NotchModeFa2

(double) NotchModeZr2: NotchModeZr2

(double) NotchModeZa2: NotchModeZa2

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationNotchModeFiltersGet command which is used to Read notch mode filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationNotchModeFiltersSet

#### Syntax

##### C# prototype

int PositionerCompensationNotchModeFiltersSet(string PositionerName, double NotchModeFr1, double NotchModeFa1, double NotchModeZr1, double NotchModeZa1, double NotchModeFr2, double NotchModeFa2, double NotchModeZr2, double NotchModeZa2, out string errstring)

##### Python prototype

[errstring] PositionerCompensationNotchModeFiltersSet (PositionerName, NotchModeFr1, NotchModeFa1, NotchModeZr1, NotchModeZa1, NotchModeFr2, NotchModeFa2, NotchModeZr2, NotchModeZa2)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) NotchModeFr1: NotchModeFr1

(double) NotchModeFa1: NotchModeFa1

(double) NotchModeZr1: NotchModeZr1

(double) NotchModeZa1: NotchModeZa1

(double) NotchModeFr2: NotchModeFr2

(double) NotchModeFa2: NotchModeFa2

(double) NotchModeZr2: NotchModeZr2

(double) NotchModeZa2: NotchModeZa2

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationNotchModeFiltersSet command which is used to Update notch mode filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPhaseCorrectionFiltersGet

#### Syntax

##### C# prototype

int PositionerCompensationPhaseCorrectionFiltersGet(string PositionerName, out double PhaseCorrectionFn1, out double PhaseCorrectionFd1, out double PhaseCorrectionGain1, out double PhaseCorrectionFn2, out double PhaseCorrectionFd2, out double PhaseCorrectionGain2, out string errstring)

##### Python prototype

[PhaseCorrectionFn1, PhaseCorrectionFd1, PhaseCorrectionGain1, PhaseCorrectionFn2, PhaseCorrectionFd2, PhaseCorrectionGain2, errstring] PositionerCompensationPhaseCorrectionFiltersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) PhaseCorrectionFn1: PhaseCorrectionFn1

(double) PhaseCorrectionFd1: PhaseCorrectionFd1

(double) PhaseCorrectionGain1: PhaseCorrectionGain1

(double) PhaseCorrectionFn2: PhaseCorrectionFn2

(double) PhaseCorrectionFd2: PhaseCorrectionFd2

(double) PhaseCorrectionGain2: PhaseCorrectionGain2

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPhaseCorrectionFiltersGet command which is used to Read phase correction filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerCompensationPhaseCorrectionFiltersSet

#### Syntax

##### C# prototype

int PositionerCompensationPhaseCorrectionFiltersSet(string PositionerName, double PhaseCorrectionFn1, double PhaseCorrectionFd1, double PhaseCorrectionGain1, double PhaseCorrectionFn2, double PhaseCorrectionFd2, double PhaseCorrectionGain2, out string errstring)

##### Python prototype

[errstring] PositionerCompensationPhaseCorrectionFiltersSet (PositionerName, PhaseCorrectionFn1, PhaseCorrectionFd1, PhaseCorrectionGain1, PhaseCorrectionFn2, PhaseCorrectionFd2, PhaseCorrectionGain2)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) PhaseCorrectionFn1: PhaseCorrectionFn1

(double) PhaseCorrectionFd1: PhaseCorrectionFd1

(double) PhaseCorrectionGain1: PhaseCorrectionGain1

(double) PhaseCorrectionFn2: PhaseCorrectionFn2

(double) PhaseCorrectionFd2: PhaseCorrectionFd2

(double) PhaseCorrectionGain2: PhaseCorrectionGain2

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerCompensationPhaseCorrectionFiltersSet command which is used to Update phase correction filters parameters . Refer to the XPS Programmer's manual to get the command description.

### PositionerAnalogTrackingPositionParametersGet

#### Syntax

##### C# prototype

int PositionerAnalogTrackingPositionParametersGet(string PositionerName, out string GPIOName, out double Offset, out double Scale, out double Velocity, out double Acceleration, out string errstring)

##### Python prototype

[GPIOName, Offset, Scale, Velocity, Acceleration, errstring] PositionerAnalogTrackingPositionParametersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) GPIOName: GPIOName

(double) Offset: Offset

(double) Scale: Scale

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerAnalogTrackingPositionParametersGet command which is used to Read dynamic parameters for one axe of a group for a future analog tracking position. Refer to the XPS Programmer's manual to get the command description.

### PositionerAnalogTrackingPositionParametersSet

#### Syntax

##### C# prototype

int PositionerAnalogTrackingPositionParametersSet(string PositionerName, string GPIOName, double Offset, double Scale, double Velocity, double Acceleration, out string errstring)

##### Python prototype

[errstring] PositionerAnalogTrackingPositionParametersSet (PositionerName, GPIOName, Offset, Scale, Velocity, Acceleration)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) GPIOName: GPIOName

(double) Offset: Offset

(double) Scale: Scale

(double) Velocity: Velocity

(double) Acceleration: Acceleration

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerAnalogTrackingPositionParametersSet command which is used to Update dynamic parameters for one axe of a group for a future analog tracking position. Refer to the XPS Programmer's manual to get the command description.

### PositionerAnalogTrackingVelocityParametersGet

#### Syntax

##### C# prototype

int PositionerAnalogTrackingVelocityParametersGet(string PositionerName, out string GPIOName, out double Offset, out double Scale, out double DeadBandThreshold, out int Order, out double Velocity, out double Acceleration, out string errstring)

##### Python prototype

[GPIOName, Offset, Scale, DeadBandThreshold, Order, Velocity, Acceleration, errstring] PositionerAnalogTrackingVelocityParametersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) GPIOName: GPIOName

(double) Offset: Offset

(double) Scale: Scale

(double) DeadBandThreshold: DeadBandThreshold

(int) Order: Order

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerAnalogTrackingVelocityParametersGet command which is used to Read dynamic parameters for one axe of a group for a future analog tracking velocity. Refer to the XPS Programmer's manual to get the command description.

### PositionerAnalogTrackingVelocityParametersSet

#### Syntax

##### C# prototype

int PositionerAnalogTrackingVelocityParametersSet(string PositionerName, string GPIOName, double Offset, double Scale, double DeadBandThreshold, int Order, double Velocity, double Acceleration, out string errstring)

##### Python prototype

[errstring] PositionerAnalogTrackingVelocityParametersSet (PositionerName, GPIOName, Offset, Scale, DeadBandThreshold, Order, Velocity, Acceleration)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) GPIOName: GPIOName

(double) Offset: Offset

(double) Scale: Scale

(double) DeadBandThreshold: DeadBandThreshold

(int) Order: Order

(double) Velocity: Velocity

(double) Acceleration: Acceleration

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerAnalogTrackingVelocityParametersSet command which is used to Update dynamic parameters for one axe of a group for a future analog tracking velocity. Refer to the XPS Programmer's manual to get the command description.

### PositionerJogMaximumVelocityAndAccelerationGet

#### Syntax

##### C# prototype

int PositionerJogMaximumVelocityAndAccelerationGet(string PositionerName, out double MaximumVelocity, out double MaximumAcceleration, out string errstring)

##### Python prototype

[MaximumVelocity, MaximumAcceleration, errstring] PositionerJogMaximumVelocityAndAccelerationGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) MaximumVelocity: MaximumVelocity

(double) MaximumAcceleration: MaximumAcceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerJogMaximumVelocityAndAccelerationGet command which is used to Return jog maximum velocity and acceleration of the positioner. Refer to the XPS Programmer's manual to get the command description.

### PositionerMaximumVelocityAndAccelerationGet

#### Syntax

##### C# prototype

int PositionerMaximumVelocityAndAccelerationGet(string PositionerName, out double MaximumVelocity, out double MaximumAcceleration, out string errstring)

##### Python prototype

[MaximumVelocity, MaximumAcceleration, errstring] PositionerMaximumVelocityAndAccelerationGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) MaximumVelocity: MaximumVelocity

(double) MaximumAcceleration: MaximumAcceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerMaximumVelocityAndAccelerationGet command which is used to Return maximum velocity and acceleration of the positioner. Refer to the XPS Programmer's manual to get the command description.

### PositionerUserTravelLimitsGet

#### Syntax

##### C# prototype

int PositionerUserTravelLimitsGet(string PositionerName, out double UserMinimumTarget, out double UserMaximumTarget, out string errstring)

##### Python prototype

[UserMinimumTarget, UserMaximumTarget, errstring] PositionerUserTravelLimitsGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) UserMinimumTarget: UserMinimumTarget

(double) UserMaximumTarget: UserMaximumTarget

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerUserTravelLimitsGet command which is used to Read UserMinimumTarget and UserMaximumTarget. Refer to the XPS Programmer's manual to get the command description.

### PositionerUserTravelLimitsSet

#### Syntax

##### C# prototype

int PositionerUserTravelLimitsSet(string PositionerName, double UserMinimumTarget, double UserMaximumTarget, out string errstring)

##### Python prototype

[errstring] PositionerUserTravelLimitsSet (PositionerName, UserMinimumTarget, UserMaximumTarget)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) UserMinimumTarget: UserMinimumTarget

(double) UserMaximumTarget: UserMaximumTarget

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerUserTravelLimitsSet command which is used to Update UserMinimumTarget and UserMaximumTarget. Refer to the XPS Programmer's manual to get the command description.

### PositionerSGammaExactVelocityAjustedDisplacementGet

#### Syntax

##### C# prototype

int PositionerSGammaExactVelocityAjustedDisplacementGet(string PositionerName, double DesiredDisplacement, out double AdjustedDisplacement, out string errstring)

##### Python prototype

[AdjustedDisplacement, errstring] PositionerSGammaExactVelocityAjustedDisplacementGet (PositionerName, DesiredDisplacement)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) DesiredDisplacement: DesiredDisplacement

##### Output parameters

(double) AdjustedDisplacement: AdjustedDisplacement

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerSGammaExactVelocityAjustedDisplacementGet command which is used to Return adjusted displacement to get exact velocity. Refer to the XPS Programmer's manual to get the command description.

### PositionerSGammaParametersGet

#### Syntax

##### C# prototype

int PositionerSGammaParametersGet(string PositionerName, out double Velocity, out double Acceleration, out double MinimumTjerkTime, out double MaximumTjerkTime, out string errstring)

##### Python prototype

[Velocity, Acceleration, MinimumTjerkTime, MaximumTjerkTime, errstring] PositionerSGammaParametersGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(double) MinimumTjerkTime: MinimumTjerkTime

(double) MaximumTjerkTime: MaximumTjerkTime

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerSGammaParametersGet command which is used to Read dynamic parameters for one axe of a group for a future displacement . Refer to the XPS Programmer's manual to get the command description.

### PositionerSGammaParametersSet

#### Syntax

##### C# prototype

int PositionerSGammaParametersSet(string PositionerName, double Velocity, double Acceleration, double MinimumTjerkTime, double MaximumTjerkTime, out string errstring)

##### Python prototype

[errstring] PositionerSGammaParametersSet (PositionerName, Velocity, Acceleration, MinimumTjerkTime, MaximumTjerkTime)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(double) MinimumTjerkTime: MinimumTjerkTime

(double) MaximumTjerkTime: MaximumTjerkTime

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerSGammaParametersSet command which is used to Update dynamic parameters for one axe of a group for a future displacement. Refer to the XPS Programmer's manual to get the command description.

### PositionerSGammaVelocityAndAccelerationSet

#### Syntax

##### C# prototype

int PositionerSGammaVelocityAndAccelerationSet(string PositionerName, double Velocity, double Acceleration, out string errstring)

##### Python prototype

[errstring] PositionerSGammaVelocityAndAccelerationSet (PositionerName, Velocity, Acceleration)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) Velocity: Velocity

(double) Acceleration: Acceleration

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerSGammaVelocityAndAccelerationSet command which is used to Update dynamic velocity and acceleration parameters for one axe of a group for a future displacement. Refer to the XPS Programmer's manual to get the command description.

### PositionerSGammaPreviousMotionTimesGet

#### Syntax

##### C# prototype

int PositionerSGammaPreviousMotionTimesGet(string PositionerName, out double SettingTime, out double SettlingTime, out string errstring)

##### Python prototype

[SettingTime, SettlingTime, errstring] PositionerSGammaPreviousMotionTimesGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) SettingTime: SettingTime

(double) SettlingTime: SettlingTime

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerSGammaPreviousMotionTimesGet command which is used to Read SettingTime and SettlingTime. Refer to the XPS Programmer's manual to get the command description.

### MultipleAxesPVTVerification

#### Syntax

##### C# prototype

int MultipleAxesPVTVerification(string GroupName, string TrajectoryFileName, out string errstring)

##### Python prototype

[errstring] MultipleAxesPVTVerification (GroupName, TrajectoryFileName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MultipleAxesPVTVerification command which is used to Multiple axes PVT trajectory verification. Refer to the XPS Programmer's manual to get the command description.

### MultipleAxesPVTVerificationResultGet

#### Syntax

##### C# prototype

int MultipleAxesPVTVerificationResultGet(string PositionerName, out string FileName, out double MinimumPosition, out double MaximumPosition, out double MaximumVelocity, out double MaximumAcceleration, out string errstring)

##### Python prototype

[FileName, MinimumPosition, MaximumPosition, MaximumVelocity, MaximumAcceleration, errstring] MultipleAxesPVTVerificationResultGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) FileName: FileName

(double) MinimumPosition: MinimumPosition

(double) MaximumPosition: MaximumPosition

(double) MaximumVelocity: MaximumVelocity

(double) MaximumAcceleration: MaximumAcceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MultipleAxesPVTVerificationResultGet command which is used to Multiple axes PVT trajectory verification result get. Refer to the XPS Programmer's manual to get the command description.

### MultipleAxesPVTExecution

#### Syntax

##### C# prototype

int MultipleAxesPVTExecution(string GroupName, string TrajectoryFileName, int ExecutionNumber, out string errstring)

##### Python prototype

[errstring] MultipleAxesPVTExecution (GroupName, TrajectoryFileName, ExecutionNumber)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

(int) ExecutionNumber: ExecutionNumber

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MultipleAxesPVTExecution command which is used to Multiple axes PVT trajectory execution. Refer to the XPS Programmer's manual to get the command description.

### MultipleAxesPVTParametersGet

#### Syntax

##### C# prototype

int MultipleAxesPVTParametersGet(string GroupName, out string FileName, out int CurrentElementNumber, out string errstring)

##### Python prototype

[FileName, CurrentElementNumber, errstring] MultipleAxesPVTParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) FileName: FileName

(int) CurrentElementNumber: CurrentElementNumber

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MultipleAxesPVTParametersGet command which is used to Multiple axes PVT trajectory get parameters. Refer to the XPS Programmer's manual to get the command description.

### MultipleAxesPVTPulseOutputSet

#### Syntax

##### C# prototype

int MultipleAxesPVTPulseOutputSet(string GroupName, int StartElement, int EndElement, double TimeInterval, out string errstring)

##### Python prototype

[errstring] MultipleAxesPVTPulseOutputSet (GroupName, StartElement, EndElement, TimeInterval)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) StartElement: StartElement

(int) EndElement: EndElement

(double) TimeInterval: TimeInterval

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MultipleAxesPVTPulseOutputSet command which is used to Configure pulse output on trajectory. Refer to the XPS Programmer's manual to get the command description.

### MultipleAxesPVTPulseOutputGet

#### Syntax

##### C# prototype

int MultipleAxesPVTPulseOutputGet(string GroupName, out int StartElement, out int EndElement, out double TimeInterval, out string errstring)

##### Python prototype

[StartElement, EndElement, TimeInterval, errstring] MultipleAxesPVTPulseOutputGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(int) StartElement: StartElement

(int) EndElement: EndElement

(double) TimeInterval: TimeInterval

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MultipleAxesPVTPulseOutputGet command which is used to Get pulse output on trajectory configuration. Refer to the XPS Programmer's manual to get the command description.

### MultipleAxesPVTLoadToMemory

#### Syntax

##### C# prototype

int MultipleAxesPVTLoadToMemory(string GroupName, string TrajectoryPart, out string errstring)

##### Python prototype

[errstring] MultipleAxesPVTLoadToMemory (GroupName, TrajectoryPart)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryPart: TrajectoryPart

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MultipleAxesPVTLoadToMemory command which is used to Multiple Axes Load PVT trajectory through function. Refer to the XPS Programmer's manual to get the command description.

### MultipleAxesPVTResetInMemory

#### Syntax

##### C# prototype

int MultipleAxesPVTResetInMemory(string GroupName, out string errstring)

##### Python prototype

[errstring] MultipleAxesPVTResetInMemory (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous MultipleAxesPVTResetInMemory command which is used to Multiple Axes PVT trajectory reset in memory. Refer to the XPS Programmer's manual to get the command description.

### XYPVTVerification

#### Syntax

##### C# prototype

int XYPVTVerification(string GroupName, string TrajectoryFileName, out string errstring)

##### Python prototype

[errstring] XYPVTVerification (GroupName, TrajectoryFileName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYPVTVerification command which is used to XY PVT trajectory verification. Refer to the XPS Programmer's manual to get the command description.

### XYPVTVerificationResultGet

#### Syntax

##### C# prototype

int XYPVTVerificationResultGet(string PositionerName, out string FileName, out double MinimumPosition, out double MaximumPosition, out double MaximumVelocity, out double MaximumAcceleration, out string errstring)

##### Python prototype

[FileName, MinimumPosition, MaximumPosition, MaximumVelocity, MaximumAcceleration, errstring] XYPVTVerificationResultGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) FileName: FileName

(double) MinimumPosition: MinimumPosition

(double) MaximumPosition: MaximumPosition

(double) MaximumVelocity: MaximumVelocity

(double) MaximumAcceleration: MaximumAcceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYPVTVerificationResultGet command which is used to XY PVT trajectory verification result get. Refer to the XPS Programmer's manual to get the command description.

### XYPVTExecution

#### Syntax

##### C# prototype

int XYPVTExecution(string GroupName, string TrajectoryFileName, int ExecutionNumber, out string errstring)

##### Python prototype

[errstring] XYPVTExecution (GroupName, TrajectoryFileName, ExecutionNumber)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

(int) ExecutionNumber: ExecutionNumber

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYPVTExecution command which is used to XY PVT trajectory execution. Refer to the XPS Programmer's manual to get the command description.

### XYPVTParametersGet

#### Syntax

##### C# prototype

int XYPVTParametersGet(string GroupName, out string FileName, out int CurrentElementNumber, out string errstring)

##### Python prototype

[FileName, CurrentElementNumber, errstring] XYPVTParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) FileName: FileName

(int) CurrentElementNumber: CurrentElementNumber

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYPVTParametersGet command which is used to XY PVT trajectory get parameters. Refer to the XPS Programmer's manual to get the command description.

### XYPVTPulseOutputSet

#### Syntax

##### C# prototype

int XYPVTPulseOutputSet(string GroupName, int StartElement, int EndElement, double TimeInterval, out string errstring)

##### Python prototype

[errstring] XYPVTPulseOutputSet (GroupName, StartElement, EndElement, TimeInterval)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) StartElement: StartElement

(int) EndElement: EndElement

(double) TimeInterval: TimeInterval

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYPVTPulseOutputSet command which is used to Configure pulse output on trajectory. Refer to the XPS Programmer's manual to get the command description.

### XYPVTPulseOutputGet

#### Syntax

##### C# prototype

int XYPVTPulseOutputGet(string GroupName, out int StartElement, out int EndElement, out double TimeInterval, out string errstring)

##### Python prototype

[StartElement, EndElement, TimeInterval, errstring] XYPVTPulseOutputGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(int) StartElement: StartElement

(int) EndElement: EndElement

(double) TimeInterval: TimeInterval

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYPVTPulseOutputGet command which is used to Get pulse output on trajectory configuration. Refer to the XPS Programmer's manual to get the command description.

### XYPVTLoadToMemory

#### Syntax

##### C# prototype

int XYPVTLoadToMemory(string GroupName, string TrajectoryPart, out string errstring)

##### Python prototype

[errstring] XYPVTLoadToMemory (GroupName, TrajectoryPart)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryPart: TrajectoryPart

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYPVTLoadToMemory command which is used to XY Load PVT trajectory through function. Refer to the XPS Programmer's manual to get the command description.

### XYPVTResetInMemory

#### Syntax

##### C# prototype

int XYPVTResetInMemory(string GroupName, out string errstring)

##### Python prototype

[errstring] XYPVTResetInMemory (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYPVTResetInMemory command which is used to XY PVT trajectory reset in memory. Refer to the XPS Programmer's manual to get the command description.

### TZPVTVerification

#### Syntax

##### C# prototype

int TZPVTVerification(string GroupName, string TrajectoryFileName, out string errstring)

##### Python prototype

[errstring] TZPVTVerification (GroupName, TrajectoryFileName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZPVTVerification command which is used to TZ PVT trajectory verification. Refer to the XPS Programmer's manual to get the command description.

### TZPVTVerificationResultGet

#### Syntax

##### C# prototype

int TZPVTVerificationResultGet(string PositionerName, out string FileName, out double MinimumPosition, out double MaximumPosition, out double MaximumVelocity, out double MaximumAcceleration, out string errstring)

##### Python prototype

[FileName, MinimumPosition, MaximumPosition, MaximumVelocity, MaximumAcceleration, errstring] TZPVTVerificationResultGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) FileName: FileName

(double) MinimumPosition: MinimumPosition

(double) MaximumPosition: MaximumPosition

(double) MaximumVelocity: MaximumVelocity

(double) MaximumAcceleration: MaximumAcceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZPVTVerificationResultGet command which is used to TZ PVT trajectory verification result get. Refer to the XPS Programmer's manual to get the command description.

### TZPVTExecution

#### Syntax

##### C# prototype

int TZPVTExecution(string GroupName, string TrajectoryFileName, int ExecutionNumber, out string errstring)

##### Python prototype

[errstring] TZPVTExecution (GroupName, TrajectoryFileName, ExecutionNumber)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

(int) ExecutionNumber: ExecutionNumber

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZPVTExecution command which is used to TZ PVT trajectory execution. Refer to the XPS Programmer's manual to get the command description.

### TZPVTParametersGet

#### Syntax

##### C# prototype

int TZPVTParametersGet(string GroupName, out string FileName, out int CurrentElementNumber, out string errstring)

##### Python prototype

[FileName, CurrentElementNumber, errstring] TZPVTParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) FileName: FileName

(int) CurrentElementNumber: CurrentElementNumber

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZPVTParametersGet command which is used to TZ PVT trajectory get parameters. Refer to the XPS Programmer's manual to get the command description.

### TZPVTPulseOutputSet

#### Syntax

##### C# prototype

int TZPVTPulseOutputSet(string GroupName, int StartElement, int EndElement, double TimeInterval, out string errstring)

##### Python prototype

[errstring] TZPVTPulseOutputSet (GroupName, StartElement, EndElement, TimeInterval)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) StartElement: StartElement

(int) EndElement: EndElement

(double) TimeInterval: TimeInterval

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZPVTPulseOutputSet command which is used to Configure pulse output on trajectory. Refer to the XPS Programmer's manual to get the command description.

### TZPVTPulseOutputGet

#### Syntax

##### C# prototype

int TZPVTPulseOutputGet(string GroupName, out int StartElement, out int EndElement, out double TimeInterval, out string errstring)

##### Python prototype

[StartElement, EndElement, TimeInterval, errstring] TZPVTPulseOutputGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(int) StartElement: StartElement

(int) EndElement: EndElement

(double) TimeInterval: TimeInterval

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZPVTPulseOutputGet command which is used to Get pulse output on trajectory configuration. Refer to the XPS Programmer's manual to get the command description.

### TZPVTLoadToMemory

#### Syntax

##### C# prototype

int TZPVTLoadToMemory(string GroupName, string TrajectoryPart, out string errstring)

##### Python prototype

[errstring] TZPVTLoadToMemory (GroupName, TrajectoryPart)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryPart: TrajectoryPart

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZPVTLoadToMemory command which is used to TZ Load PVT trajectory through function. Refer to the XPS Programmer's manual to get the command description.

### TZPVTResetInMemory

#### Syntax

##### C# prototype

int TZPVTResetInMemory(string GroupName, out string errstring)

##### Python prototype

[errstring] TZPVTResetInMemory (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZPVTResetInMemory command which is used to TZ PVT trajectory reset in memory. Refer to the XPS Programmer's manual to get the command description.

### XYLineArcVerification

#### Syntax

##### C# prototype

int XYLineArcVerification(string GroupName, string TrajectoryFileName, out string errstring)

##### Python prototype

[errstring] XYLineArcVerification (GroupName, TrajectoryFileName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYLineArcVerification command which is used to XY trajectory verification. Refer to the XPS Programmer's manual to get the command description.

### XYLineArcVerificationResultGet

#### Syntax

##### C# prototype

int XYLineArcVerificationResultGet(string PositionerName, out string FileName, out double MinimumPosition, out double MaximumPosition, out double MaximumVelocity, out double MaximumAcceleration, out string errstring)

##### Python prototype

[FileName, MinimumPosition, MaximumPosition, MaximumVelocity, MaximumAcceleration, errstring] XYLineArcVerificationResultGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) FileName: FileName

(double) MinimumPosition: MinimumPosition

(double) MaximumPosition: MaximumPosition

(double) MaximumVelocity: MaximumVelocity

(double) MaximumAcceleration: MaximumAcceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYLineArcVerificationResultGet command which is used to XY trajectory verification result get. Refer to the XPS Programmer's manual to get the command description.

### XYLineArcExecution

#### Syntax

##### C# prototype

int XYLineArcExecution(string GroupName, string TrajectoryFileName, double Velocity, double Acceleration, int ExecutionNumber, out string errstring)

##### Python prototype

[errstring] XYLineArcExecution (GroupName, TrajectoryFileName, Velocity, Acceleration, ExecutionNumber)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(int) ExecutionNumber: ExecutionNumber

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYLineArcExecution command which is used to XY trajectory execution. Refer to the XPS Programmer's manual to get the command description.

### XYLineArcParametersGet

#### Syntax

##### C# prototype

int XYLineArcParametersGet(string GroupName, out string FileName, out double Velocity, out double Acceleration, out int CurrentElementNumber, out string errstring)

##### Python prototype

[FileName, Velocity, Acceleration, CurrentElementNumber, errstring] XYLineArcParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) FileName: FileName

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(int) CurrentElementNumber: CurrentElementNumber

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYLineArcParametersGet command which is used to XY trajectory get parameters. Refer to the XPS Programmer's manual to get the command description.

### XYLineArcPulseOutputSet

#### Syntax

##### C# prototype

int XYLineArcPulseOutputSet(string GroupName, double StartLength, double EndLength, double PathLengthInterval, out string errstring)

##### Python prototype

[errstring] XYLineArcPulseOutputSet (GroupName, StartLength, EndLength, PathLengthInterval)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) StartLength: StartLength

(double) EndLength: EndLength

(double) PathLengthInterval: PathLengthInterval

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYLineArcPulseOutputSet command which is used to Configure pulse output on trajectory. Refer to the XPS Programmer's manual to get the command description.

### XYLineArcPulseOutputGet

#### Syntax

##### C# prototype

int XYLineArcPulseOutputGet(string GroupName, out double StartLength, out double EndLength, out double PathLengthInterval, out string errstring)

##### Python prototype

[StartLength, EndLength, PathLengthInterval, errstring] XYLineArcPulseOutputGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(double) StartLength: StartLength

(double) EndLength: EndLength

(double) PathLengthInterval: PathLengthInterval

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYLineArcPulseOutputGet command which is used to Get pulse output on trajectory configuration. Refer to the XPS Programmer's manual to get the command description.

### XYZSplineVerification

#### Syntax

##### C# prototype

int XYZSplineVerification(string GroupName, string TrajectoryFileName, out string errstring)

##### Python prototype

[errstring] XYZSplineVerification (GroupName, TrajectoryFileName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYZSplineVerification command which is used to XYZ trajectory verifivation. Refer to the XPS Programmer's manual to get the command description.

### XYZSplineVerificationResultGet

#### Syntax

##### C# prototype

int XYZSplineVerificationResultGet(string PositionerName, out string FileName, out double MinimumPosition, out double MaximumPosition, out double MaximumVelocity, out double MaximumAcceleration, out string errstring)

##### Python prototype

[FileName, MinimumPosition, MaximumPosition, MaximumVelocity, MaximumAcceleration, errstring] XYZSplineVerificationResultGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) FileName: FileName

(double) MinimumPosition: MinimumPosition

(double) MaximumPosition: MaximumPosition

(double) MaximumVelocity: MaximumVelocity

(double) MaximumAcceleration: MaximumAcceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYZSplineVerificationResultGet command which is used to XYZ trajectory verification result get. Refer to the XPS Programmer's manual to get the command description.

### XYZSplineExecution

#### Syntax

##### C# prototype

int XYZSplineExecution(string GroupName, string TrajectoryFileName, double Velocity, double Acceleration, out string errstring)

##### Python prototype

[errstring] XYZSplineExecution (GroupName, TrajectoryFileName, Velocity, Acceleration)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TrajectoryFileName: TrajectoryFileName

(double) Velocity: Velocity

(double) Acceleration: Acceleration

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYZSplineExecution command which is used to XYZ trajectory execution. Refer to the XPS Programmer's manual to get the command description.

### XYZSplineParametersGet

#### Syntax

##### C# prototype

int XYZSplineParametersGet(string GroupName, out string FileName, out double Velocity, out double Acceleration, out int CurrentElementNumber, out string errstring)

##### Python prototype

[FileName, Velocity, Acceleration, CurrentElementNumber, errstring] XYZSplineParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) FileName: FileName

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(int) CurrentElementNumber: CurrentElementNumber

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYZSplineParametersGet command which is used to XYZ trajectory get parameters. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisSlaveModeEnable

#### Syntax

##### C# prototype

int SingleAxisSlaveModeEnable(string GroupName, out string errstring)

##### Python prototype

[errstring] SingleAxisSlaveModeEnable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisSlaveModeEnable command which is used to Enable the slave mode. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisSlaveModeDisable

#### Syntax

##### C# prototype

int SingleAxisSlaveModeDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] SingleAxisSlaveModeDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisSlaveModeDisable command which is used to Disable the slave mode. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisSlaveParametersSet

#### Syntax

##### C# prototype

int SingleAxisSlaveParametersSet(string GroupName, string PositionerName, double Ratio, out string errstring)

##### Python prototype

[errstring] SingleAxisSlaveParametersSet (GroupName, PositionerName, Ratio)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) PositionerName: PositionerName

(double) Ratio: Ratio

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisSlaveParametersSet command which is used to Set slave parameters. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisSlaveParametersGet

#### Syntax

##### C# prototype

int SingleAxisSlaveParametersGet(string GroupName, out string PositionerName, out double Ratio, out string errstring)

##### Python prototype

[PositionerName, Ratio, errstring] SingleAxisSlaveParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) PositionerName: PositionerName

(double) Ratio: Ratio

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisSlaveParametersGet command which is used to Get slave parameters. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaClampDisable

#### Syntax

##### C# prototype

int SingleAxisThetaClampDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] SingleAxisThetaClampDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaClampDisable command which is used to Set clamping disable on selected group. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaClampEnable

#### Syntax

##### C# prototype

int SingleAxisThetaClampEnable(string GroupName, out string errstring)

##### Python prototype

[errstring] SingleAxisThetaClampEnable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaClampEnable command which is used to Set clamping enable on selected group. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaFeedforwardParametersGet

#### Syntax

##### C# prototype

int SingleAxisThetaFeedforwardParametersGet(string GroupName, out double KFeedforwardX, out double KFeedforwardY, out string errstring)

##### Python prototype

[KFeedforwardX, KFeedforwardY, errstring] SingleAxisThetaFeedforwardParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(double) KFeedforwardX: KFeedforwardX

(double) KFeedforwardY: KFeedforwardY

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaFeedforwardParametersGet command which is used to Get XY to Theta feedforward gains. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaFeedforwardParametersSet

#### Syntax

##### C# prototype

int SingleAxisThetaFeedforwardParametersSet(string GroupName, double KFeedforwardX, double KFeedforwardY, out string errstring)

##### Python prototype

[errstring] SingleAxisThetaFeedforwardParametersSet (GroupName, KFeedforwardX, KFeedforwardY)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) KFeedforwardX: KFeedforwardX

(double) KFeedforwardY: KFeedforwardY

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaFeedforwardParametersSet command which is used to Set XY to Theta feedforward gains. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaSlaveModeEnable

#### Syntax

##### C# prototype

int SingleAxisThetaSlaveModeEnable(string GroupName, out string errstring)

##### Python prototype

[errstring] SingleAxisThetaSlaveModeEnable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaSlaveModeEnable command which is used to Enable the slave mode. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaSlaveModeDisable

#### Syntax

##### C# prototype

int SingleAxisThetaSlaveModeDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] SingleAxisThetaSlaveModeDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaSlaveModeDisable command which is used to Disable the slave mode. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaSlaveParametersGet

#### Syntax

##### C# prototype

int SingleAxisThetaSlaveParametersGet(string GroupName, out string PositionerName, out double Ratio, out string errstring)

##### Python prototype

[PositionerName, Ratio, errstring] SingleAxisThetaSlaveParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) PositionerName: PositionerName

(double) Ratio: Ratio

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaSlaveParametersGet command which is used to Get slave parameters. Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaSlaveParametersSet

#### Syntax

##### C# prototype

int SingleAxisThetaSlaveParametersSet(string GroupName, string PositionerName, double Ratio, out string errstring)

##### Python prototype

[errstring] SingleAxisThetaSlaveParametersSet (GroupName, PositionerName, Ratio)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) PositionerName: PositionerName

(double) Ratio: Ratio

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaSlaveParametersSet command which is used to Set slave parameters. Refer to the XPS Programmer's manual to get the command description.

### SpindleSlaveModeEnable

#### Syntax

##### C# prototype

int SpindleSlaveModeEnable(string GroupName, out string errstring)

##### Python prototype

[errstring] SpindleSlaveModeEnable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SpindleSlaveModeEnable command which is used to Enable the slave mode. Refer to the XPS Programmer's manual to get the command description.

### SpindleSlaveModeDisable

#### Syntax

##### C# prototype

int SpindleSlaveModeDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] SpindleSlaveModeDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SpindleSlaveModeDisable command which is used to Disable the slave mode. Refer to the XPS Programmer's manual to get the command description.

### SpindleSlaveParametersSet

#### Syntax

##### C# prototype

int SpindleSlaveParametersSet(string GroupName, string PositionerName, double Ratio, out string errstring)

##### Python prototype

[errstring] SpindleSlaveParametersSet (GroupName, PositionerName, Ratio)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) PositionerName: PositionerName

(double) Ratio: Ratio

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SpindleSlaveParametersSet command which is used to Set slave parameters. Refer to the XPS Programmer's manual to get the command description.

### SpindleSlaveParametersGet

#### Syntax

##### C# prototype

int SpindleSlaveParametersGet(string GroupName, out string PositionerName, out double Ratio, out string errstring)

##### Python prototype

[PositionerName, Ratio, errstring] SpindleSlaveParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) PositionerName: PositionerName

(double) Ratio: Ratio

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SpindleSlaveParametersGet command which is used to Get slave parameters. Refer to the XPS Programmer's manual to get the command description.

### GroupSpinParametersSet

#### Syntax

##### C# prototype

int GroupSpinParametersSet(string GroupName, double Velocity, double Acceleration, out string errstring)

##### Python prototype

[errstring] GroupSpinParametersSet (GroupName, Velocity, Acceleration)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) Velocity: Velocity

(double) Acceleration: Acceleration

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupSpinParametersSet command which is used to Modify Spin parameters on selected group and activate the continuous move. Refer to the XPS Programmer's manual to get the command description.

### GroupSpinParametersGet

#### Syntax

##### C# prototype

int GroupSpinParametersGet(string GroupName, out double Velocity, out double Acceleration, out string errstring)

##### Python prototype

[Velocity, Acceleration, errstring] GroupSpinParametersGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupSpinParametersGet command which is used to Get Spin parameters on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupSpinCurrentGet

#### Syntax

##### C# prototype

int GroupSpinCurrentGet(string GroupName, out double Velocity, out double Acceleration, out string errstring)

##### Python prototype

[Velocity, Acceleration, errstring] GroupSpinCurrentGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(double) Velocity: Velocity

(double) Acceleration: Acceleration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupSpinCurrentGet command which is used to Get Spin current on selected group. Refer to the XPS Programmer's manual to get the command description.

### GroupSpinModeStop

#### Syntax

##### C# prototype

int GroupSpinModeStop(string GroupName, double Acceleration, out string errstring)

##### Python prototype

[errstring] GroupSpinModeStop (GroupName, Acceleration)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) Acceleration: Acceleration

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupSpinModeStop command which is used to Stop Spin mode on selected group with specified acceleration. Refer to the XPS Programmer's manual to get the command description.

### PositionerFeedforwardAccSet

#### Syntax

##### C# prototype

int PositionerFeedforwardAccSet(string PositionerName, string OutputName1, double Scale1, string OutputName2, double Scale2, out string errstring)

##### Python prototype

[errstring] PositionerFeedforwardAccSet (PositionerName, OutputName1, Scale1, OutputName2, Scale2)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) OutputName1: OutputName1

(double) Scale1: Scale1

(string) OutputName2: OutputName2

(double) Scale2: Scale2

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerFeedforwardAccSet command which is used to Set Feed forward Acc. Refer to the XPS Programmer's manual to get the command description.

### PositionerFeedforwardAccGet

#### Syntax

##### C# prototype

int PositionerFeedforwardAccGet(string PositionerName, out string OutputName1, out double Scale1, out string OutputName2, out double Scale2, out string errstring)

##### Python prototype

[OutputName1, Scale1, OutputName2, Scale2, errstring] PositionerFeedforwardAccGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) OutputName1: OutputName1

(double) Scale1: Scale1

(string) OutputName2: OutputName2

(double) Scale2: Scale2

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerFeedforwardAccGet command which is used to Set Feed forward Acc. Refer to the XPS Programmer's manual to get the command description.

### PositionerFeedforwardAccEnable

#### Syntax

##### C# prototype

int PositionerFeedforwardAccEnable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerFeedforwardAccEnable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerFeedforwardAccEnable command which is used to Enable Feed forward Acc. Refer to the XPS Programmer's manual to get the command description.

### PositionerFeedforwardAccDisable

#### Syntax

##### C# prototype

int PositionerFeedforwardAccDisable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerFeedforwardAccDisable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerFeedforwardAccDisable command which is used to Disable Feed forward Acc. Refer to the XPS Programmer's manual to get the command description.

### PositionerFeedforwardPositionSet

#### Syntax

##### C# prototype

int PositionerFeedforwardPositionSet(string PositionerName, string OutputName, double Scale, double Offset, out string errstring)

##### Python prototype

[errstring] PositionerFeedforwardPositionSet (PositionerName, OutputName, Scale, Offset)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(string) OutputName: OutputName

(double) Scale: Scale

(double) Offset: Offset

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerFeedforwardPositionSet command which is used to Set Feed forward Position. Refer to the XPS Programmer's manual to get the command description.

### PositionerFeedforwardPositionGet

#### Syntax

##### C# prototype

int PositionerFeedforwardPositionGet(string PositionerName, out string OutputName, out double Scale, out double Offset, out string errstring)

##### Python prototype

[OutputName, Scale, Offset, errstring] PositionerFeedforwardPositionGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) OutputName: OutputName

(double) Scale: Scale

(double) Offset: Offset

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerFeedforwardPositionGet command which is used to Get Feed forward Position. Refer to the XPS Programmer's manual to get the command description.

### PositionerFeedforwardPositionEnable

#### Syntax

##### C# prototype

int PositionerFeedforwardPositionEnable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerFeedforwardPositionEnable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerFeedforwardPositionEnable command which is used to Enable Feed forward Position. Refer to the XPS Programmer's manual to get the command description.

### PositionerFeedforwardPositionDisable

#### Syntax

##### C# prototype

int PositionerFeedforwardPositionDisable(string PositionerName, out string errstring)

##### Python prototype

[errstring] PositionerFeedforwardPositionDisable (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerFeedforwardPositionDisable command which is used to Disable Feed forward Position. Refer to the XPS Programmer's manual to get the command description.

### XYZGroupPositionCorrectedProfilerGet

#### Syntax

##### C# prototype

int XYZGroupPositionCorrectedProfilerGet(string GroupName, double PositionX, double PositionY, double PositionZ, out double CorrectedProfilerPositionX, out double CorrectedProfilerPositionY, out double CorrectedProfilerPositionZ, out string errstring)

##### Python prototype

[CorrectedProfilerPositionX, CorrectedProfilerPositionY, CorrectedProfilerPositionZ, errstring] XYZGroupPositionCorrectedProfilerGet (GroupName, PositionX, PositionY, PositionZ)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) PositionX: PositionX

(double) PositionY: PositionY

(double) PositionZ: PositionZ

##### Output parameters

(double) CorrectedProfilerPositionX: CorrectedProfilerPositionX

(double) CorrectedProfilerPositionY: CorrectedProfilerPositionY

(double) CorrectedProfilerPositionZ: CorrectedProfilerPositionZ

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYZGroupPositionCorrectedProfilerGet command which is used to Return corrected profiler positions. Refer to the XPS Programmer's manual to get the command description.

### XYZGroupPositionPCORawEncoderGet

#### Syntax

##### C# prototype

int XYZGroupPositionPCORawEncoderGet(string GroupName, double PositionX, double PositionY, double PositionZ, out double PCORawPositionX, out double PCORawPositionY, out double PCORawPositionZ, out string errstring)

##### Python prototype

[PCORawPositionX, PCORawPositionY, PCORawPositionZ, errstring] XYZGroupPositionPCORawEncoderGet (GroupName, PositionX, PositionY, PositionZ)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) PositionX: PositionX

(double) PositionY: PositionY

(double) PositionZ: PositionZ

##### Output parameters

(double) PCORawPositionX: PCORawPositionX

(double) PCORawPositionY: PCORawPositionY

(double) PCORawPositionZ: PCORawPositionZ

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous XYZGroupPositionPCORawEncoderGet command which is used to Return PCO raw encoder positions. Refer to the XPS Programmer's manual to get the command description.

### TZMotorDecouplingMatrixGet

#### Syntax

##### C# prototype

int TZMotorDecouplingMatrixGet(string GroupName, out double Value11, out double Value12, out double Value13, out double Value21, out double Value22, out double Value23, out double Value31, out double Value32, out double Value33, out string errstring)

##### Python prototype

[Value11, Value12, Value13, Value21, Value22, Value23, Value31, Value32, Value33, errstring] TZMotorDecouplingMatrixGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(double) Value11: Value11

(double) Value12: Value12

(double) Value13: Value13

(double) Value21: Value21

(double) Value22: Value22

(double) Value23: Value23

(double) Value31: Value31

(double) Value32: Value32

(double) Value33: Value33

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZMotorDecouplingMatrixGet command which is used to Get TZ motor decoupling matrix. Refer to the XPS Programmer's manual to get the command description.

### TZMotorDecouplingMatrixSet

#### Syntax

##### C# prototype

int TZMotorDecouplingMatrixSet(string GroupName, double Value11, double Value12, double Value13, double Value21, double Value22, double Value23, double Value31, double Value32, double Value33, out string errstring)

##### Python prototype

[errstring] TZMotorDecouplingMatrixSet (GroupName, Value11, Value12, Value13, Value21, Value22, Value23, Value31, Value32, Value33)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) Value11: Value11

(double) Value12: Value12

(double) Value13: Value13

(double) Value21: Value21

(double) Value22: Value22

(double) Value23: Value23

(double) Value31: Value31

(double) Value32: Value32

(double) Value33: Value33

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZMotorDecouplingMatrixSet command which is used to Set TZ motor decoupling matrix. Refer to the XPS Programmer's manual to get the command description.

### TZFocusModeEnable

#### Syntax

##### C# prototype

int TZFocusModeEnable(string GroupName, out string errstring)

##### Python prototype

[errstring] TZFocusModeEnable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZFocusModeEnable command which is used to Enable the focus mode. Refer to the XPS Programmer's manual to get the command description.

### TZFocusModeDisable

#### Syntax

##### C# prototype

int TZFocusModeDisable(string GroupName, out string errstring)

##### Python prototype

[errstring] TZFocusModeDisable (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZFocusModeDisable command which is used to Disable the focus mode. Refer to the XPS Programmer's manual to get the command description.

### TZTrackingUserMaximumZZZTargetDifferenceGet

#### Syntax

##### C# prototype

int TZTrackingUserMaximumZZZTargetDifferenceGet(string GroupName, out double UserMaximumZZZTargetDifference, out string errstring)

##### Python prototype

[UserMaximumZZZTargetDifference, errstring] TZTrackingUserMaximumZZZTargetDifferenceGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(double) UserMaximumZZZTargetDifference: UserMaximumZZZTargetDifference

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZTrackingUserMaximumZZZTargetDifferenceGet command which is used to Get user maximum ZZZ target difference for tracking control. Refer to the XPS Programmer's manual to get the command description.

### TZTrackingUserMaximumZZZTargetDifferenceSet

#### Syntax

##### C# prototype

int TZTrackingUserMaximumZZZTargetDifferenceSet(string GroupName, double UserMaximumZZZTargetDifference, out string errstring)

##### Python prototype

[errstring] TZTrackingUserMaximumZZZTargetDifferenceSet (GroupName, UserMaximumZZZTargetDifference)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(double) UserMaximumZZZTargetDifference: UserMaximumZZZTargetDifference

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TZTrackingUserMaximumZZZTargetDifferenceSet command which is used to Set user maximum ZZZ target difference for tracking control. Refer to the XPS Programmer's manual to get the command description.

### FocusProcessSocketReserve

#### Syntax

##### C# prototype

int FocusProcessSocketReserve( out string errstring)

##### Python prototype

[errstring] FocusProcessSocketReserve ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous FocusProcessSocketReserve command which is used to . Refer to the XPS Programmer's manual to get the command description.

### FocusProcessSocketFree

#### Syntax

##### C# prototype

int FocusProcessSocketFree( out string errstring)

##### Python prototype

[errstring] FocusProcessSocketFree ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous FocusProcessSocketFree command which is used to . Refer to the XPS Programmer's manual to get the command description.

### LoginCheck

#### Syntax

##### C# prototype

int LoginCheck(string Name, string Password, out string errstring)

##### Python prototype

[errstring] LoginCheck (Name, Password)

#### Parameters

##### Input parameters

(string) Name: Name

(string) Password: Password

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous LoginCheck command which is used to Check the Login Parameters. Refer to the XPS Programmer's manual to get the command description.

### GroupAllPositionTrace

#### Syntax

##### C# prototype

int GroupAllPositionTrace(string GroupName, string TitleTrace, out string errstring)

##### Python prototype

[errstring] GroupAllPositionTrace (GroupName, TitleTrace)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TitleTrace: TitleTrace

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupAllPositionTrace command which is used to Trace all position in TraceAllPositions.log. Refer to the XPS Programmer's manual to get the command description.

### GroupMotorMatrixTrace

#### Syntax

##### C# prototype

int GroupMotorMatrixTrace(string GroupName, string TitleTrace, out string errstring)

##### Python prototype

[errstring] GroupMotorMatrixTrace (GroupName, TitleTrace)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TitleTrace: TitleTrace

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMotorMatrixTrace command which is used to Trace all position in TraceMotorMatrix.log. Refer to the XPS Programmer's manual to get the command description.

### GroupMotorMatrixInverseTrace

#### Syntax

##### C# prototype

int GroupMotorMatrixInverseTrace(string GroupName, string TitleTrace, out string errstring)

##### Python prototype

[errstring] GroupMotorMatrixInverseTrace (GroupName, TitleTrace)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(string) TitleTrace: TitleTrace

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupMotorMatrixInverseTrace command which is used to Trace all position in TraceMotorMatrixInverse.log. Refer to the XPS Programmer's manual to get the command description.

### GroupPositionCurrentRawGet

#### Syntax

##### C# prototype

int GroupPositionCurrentRawGet(string GroupName, out double[] RawCurrentEncoderPosition, int nbItems, out string errstring)

##### Python prototype

[RawCurrentEncoderPosition, errstring] GroupPositionCurrentRawGet (GroupName, nbItems)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

(int) nbItems: nbItems

##### Output parameters

(double[]) RawCurrentEncoderPosition: RawCurrentEncoderPosition

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupPositionCurrentRawGet command which is used to Return raw current positions. Refer to the XPS Programmer's manual to get the command description.

### PositionerMotorOutputOffsetGet

#### Syntax

##### C# prototype

int PositionerMotorOutputOffsetGet(string PositionerName, out double PrimaryDAC1, out double PrimaryDAC2, out double PrimaryDACDifferentialGain, out double SecondaryDAC1, out double SecondaryDAC2, out double SecondaryDACDifferentialGain, out string errstring)

##### Python prototype

[PrimaryDAC1, PrimaryDAC2, PrimaryDACDifferentialGain, SecondaryDAC1, SecondaryDAC2, SecondaryDACDifferentialGain, errstring] PositionerMotorOutputOffsetGet (PositionerName)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

##### Output parameters

(double) PrimaryDAC1: PrimaryDAC1

(double) PrimaryDAC2: PrimaryDAC2

(double) PrimaryDACDifferentialGain: PrimaryDACDifferentialGain

(double) SecondaryDAC1: SecondaryDAC1

(double) SecondaryDAC2: SecondaryDAC2

(double) SecondaryDACDifferentialGain: SecondaryDACDifferentialGain

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerMotorOutputOffsetGet command which is used to Get soft . Refer to the XPS Programmer's manual to get the command description.

### PositionerMotorOutputOffsetSet

#### Syntax

##### C# prototype

int PositionerMotorOutputOffsetSet(string PositionerName, double PrimaryDAC1, double PrimaryDAC2, double PrimaryDACDifferentialGain, double SecondaryDAC1, double SecondaryDAC2, double SecondaryDACDifferentialGain, out string errstring)

##### Python prototype

[errstring] PositionerMotorOutputOffsetSet (PositionerName, PrimaryDAC1, PrimaryDAC2, PrimaryDACDifferentialGain, SecondaryDAC1, SecondaryDAC2, SecondaryDACDifferentialGain)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(double) PrimaryDAC1: PrimaryDAC1

(double) PrimaryDAC2: PrimaryDAC2

(double) PrimaryDACDifferentialGain: PrimaryDACDifferentialGain

(double) SecondaryDAC1: SecondaryDAC1

(double) SecondaryDAC2: SecondaryDAC2

(double) SecondaryDACDifferentialGain: SecondaryDACDifferentialGain

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerMotorOutputOffsetSet command which is used to Set soft . Refer to the XPS Programmer's manual to get the command description.

### SingleAxisThetaPositionRawGet

#### Syntax

##### C# prototype

int SingleAxisThetaPositionRawGet(string GroupName, out double RawEncoderPosition1, out double RawEncoderPosition2, out double RawEncoderPosition3, out string errstring)

##### Python prototype

[RawEncoderPosition1, RawEncoderPosition2, RawEncoderPosition3, errstring] SingleAxisThetaPositionRawGet (GroupName)

#### Parameters

##### Input parameters

(string) GroupName: GroupName

##### Output parameters

(double) RawEncoderPosition1: RawEncoderPosition1

(double) RawEncoderPosition2: RawEncoderPosition2

(double) RawEncoderPosition3: RawEncoderPosition3

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SingleAxisThetaPositionRawGet command which is used to Get raw encoder positions for single axis theta encoder. Refer to the XPS Programmer's manual to get the command description.

### EEPROMCIESet

#### Syntax

##### C# prototype

int EEPROMCIESet(int CardNumber, string ReferenceString, out string errstring)

##### Python prototype

[errstring] EEPROMCIESet (CardNumber, ReferenceString)

#### Parameters

##### Input parameters

(int) CardNumber: CardNumber

(string) ReferenceString: ReferenceString

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EEPROMCIESet command which is used to . Refer to the XPS Programmer's manual to get the command description.

### EEPROMDACOffsetCIESet

#### Syntax

##### C# prototype

int EEPROMDACOffsetCIESet(int PlugNumber, double DAC1Offset, double DAC2Offset, out string errstring)

##### Python prototype

[errstring] EEPROMDACOffsetCIESet (PlugNumber, DAC1Offset, DAC2Offset)

#### Parameters

##### Input parameters

(int) PlugNumber: PlugNumber

(double) DAC1Offset: DAC1Offset

(double) DAC2Offset: DAC2Offset

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EEPROMDACOffsetCIESet command which is used to . Refer to the XPS Programmer's manual to get the command description.

### EEPROMDriverSet

#### Syntax

##### C# prototype

int EEPROMDriverSet(int PlugNumber, string ReferenceString, out string errstring)

##### Python prototype

[errstring] EEPROMDriverSet (PlugNumber, ReferenceString)

#### Parameters

##### Input parameters

(int) PlugNumber: PlugNumber

(string) ReferenceString: ReferenceString

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EEPROMDriverSet command which is used to . Refer to the XPS Programmer's manual to get the command description.

### EEPROMINTSet

#### Syntax

##### C# prototype

int EEPROMINTSet(int CardNumber, string ReferenceString, out string errstring)

##### Python prototype

[errstring] EEPROMINTSet (CardNumber, ReferenceString)

#### Parameters

##### Input parameters

(int) CardNumber: CardNumber

(string) ReferenceString: ReferenceString

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EEPROMINTSet command which is used to . Refer to the XPS Programmer's manual to get the command description.

### CPUCoreAndBoardSupplyVoltagesGet

#### Syntax

##### C# prototype

int CPUCoreAndBoardSupplyVoltagesGet(out double VoltageCPUCore, out double SupplyVoltage1P5V, out double SupplyVoltage3P3V, out double SupplyVoltage5V, out double SupplyVoltage12V, out double SupplyVoltageM12V, out double SupplyVoltageM5V, out double SupplyVoltage5VSB, out string errstring)

##### Python prototype

[VoltageCPUCore, SupplyVoltage1P5V, SupplyVoltage3P3V, SupplyVoltage5V, SupplyVoltage12V, SupplyVoltageM12V, SupplyVoltageM5V, SupplyVoltage5VSB, errstring] CPUCoreAndBoardSupplyVoltagesGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) VoltageCPUCore: VoltageCPUCore

(double) SupplyVoltage1P5V: SupplyVoltage1P5V

(double) SupplyVoltage3P3V: SupplyVoltage3P3V

(double) SupplyVoltage5V: SupplyVoltage5V

(double) SupplyVoltage12V: SupplyVoltage12V

(double) SupplyVoltageM12V: SupplyVoltageM12V

(double) SupplyVoltageM5V: SupplyVoltageM5V

(double) SupplyVoltage5VSB: SupplyVoltage5VSB

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CPUCoreAndBoardSupplyVoltagesGet command which is used to . Refer to the XPS Programmer's manual to get the command description.

### CPUTemperatureAndFanSpeedGet

#### Syntax

##### C# prototype

int CPUTemperatureAndFanSpeedGet(out double CPUTemperature, out double CPUFanSpeed, out string errstring)

##### Python prototype

[CPUTemperature, CPUFanSpeed, errstring] CPUTemperatureAndFanSpeedGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) CPUTemperature: CPUTemperature

(double) CPUFanSpeed: CPUFanSpeed

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CPUTemperatureAndFanSpeedGet command which is used to . Refer to the XPS Programmer's manual to get the command description.

### ActionListGet

#### Syntax

##### C# prototype

int ActionListGet(out string ActionList, out string errstring)

##### Python prototype

[ActionList, errstring] ActionListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) ActionList: ActionList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ActionListGet command which is used to Action list. Refer to the XPS Programmer's manual to get the command description.

### ActionExtendedListGet

#### Syntax

##### C# prototype

int ActionExtendedListGet(out string ActionList, out string errstring)

##### Python prototype

[ActionList, errstring] ActionExtendedListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) ActionList: ActionList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ActionExtendedListGet command which is used to Action extended list. Refer to the XPS Programmer's manual to get the command description.

### APIExtendedListGet

#### Syntax

##### C# prototype

int APIExtendedListGet(out string Method, out string errstring)

##### Python prototype

[Method, errstring] APIExtendedListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Method: Method

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous APIExtendedListGet command which is used to API method list. Refer to the XPS Programmer's manual to get the command description.

### APIListGet

#### Syntax

##### C# prototype

int APIListGet(out string Method, out string errstring)

##### Python prototype

[Method, errstring] APIListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Method: Method

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous APIListGet command which is used to User API method list without extended API. Refer to the XPS Programmer's manual to get the command description.

### APIListStandardGet

#### Syntax

##### C# prototype

int APIListStandardGet(out string Method, out string errstring)

##### Python prototype

[Method, errstring] APIListStandardGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Method: Method

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous APIListStandardGet command which is used to Standard API method list without extended API. Refer to the XPS Programmer's manual to get the command description.

### APIListAMATGet

#### Syntax

##### C# prototype

int APIListAMATGet(out string Method, out string errstring)

##### Python prototype

[Method, errstring] APIListAMATGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) Method: Method

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous APIListAMATGet command which is used to AMAT API method list without extended API. Refer to the XPS Programmer's manual to get the command description.

### ControllerStatusListGet

#### Syntax

##### C# prototype

int ControllerStatusListGet(out string ControllerStatusList, out string errstring)

##### Python prototype

[ControllerStatusList, errstring] ControllerStatusListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) ControllerStatusList: ControllerStatusList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerStatusListGet command which is used to Controller status list. Refer to the XPS Programmer's manual to get the command description.

### ErrorListGet

#### Syntax

##### C# prototype

int ErrorListGet(out string ErrorsList, out string errstring)

##### Python prototype

[ErrorsList, errstring] ErrorListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) ErrorsList: ErrorsList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ErrorListGet command which is used to Error list. Refer to the XPS Programmer's manual to get the command description.

### EventListGet

#### Syntax

##### C# prototype

int EventListGet(out string EventList, out string errstring)

##### Python prototype

[EventList, errstring] EventListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) EventList: EventList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventListGet command which is used to General event list. Refer to the XPS Programmer's manual to get the command description.

### GatheringListGet

#### Syntax

##### C# prototype

int GatheringListGet(out string list, out string errstring)

##### Python prototype

[list, errstring] GatheringListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) list: list

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringListGet command which is used to Gathering type list. Refer to the XPS Programmer's manual to get the command description.

### GatheringExtendedListGet

#### Syntax

##### C# prototype

int GatheringExtendedListGet(out string list, out string errstring)

##### Python prototype

[list, errstring] GatheringExtendedListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) list: list

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringExtendedListGet command which is used to Gathering type extended list. Refer to the XPS Programmer's manual to get the command description.

### GatheringExternalListGet

#### Syntax

##### C# prototype

int GatheringExternalListGet(out string list, out string errstring)

##### Python prototype

[list, errstring] GatheringExternalListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) list: list

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringExternalListGet command which is used to External Gathering type list. Refer to the XPS Programmer's manual to get the command description.

### GroupStatusListGet

#### Syntax

##### C# prototype

int GroupStatusListGet(out string GroupStatusList, out string errstring)

##### Python prototype

[GroupStatusList, errstring] GroupStatusListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) GroupStatusList: GroupStatusList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GroupStatusListGet command which is used to Group status list. Refer to the XPS Programmer's manual to get the command description.

### HardwareInternalListGet

#### Syntax

##### C# prototype

int HardwareInternalListGet(out string InternalHardwareList, out string errstring)

##### Python prototype

[InternalHardwareList, errstring] HardwareInternalListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) InternalHardwareList: InternalHardwareList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous HardwareInternalListGet command which is used to Internal hardware list. Refer to the XPS Programmer's manual to get the command description.

### ObjectsListGet

#### Syntax

##### C# prototype

int ObjectsListGet(out string ObjectsList, out string errstring)

##### Python prototype

[ObjectsList, errstring] ObjectsListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) ObjectsList: ObjectsList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ObjectsListGet command which is used to Group name and positioner name. Refer to the XPS Programmer's manual to get the command description.

### PositionerErrorListGet

#### Syntax

##### C# prototype

int PositionerErrorListGet(out string PositionerErrorList, out string errstring)

##### Python prototype

[PositionerErrorList, errstring] PositionerErrorListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) PositionerErrorList: PositionerErrorList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerErrorListGet command which is used to Positioner error list. Refer to the XPS Programmer's manual to get the command description.

### PositionerHardwareStatusListGet

#### Syntax

##### C# prototype

int PositionerHardwareStatusListGet(out string PositionerHardwareStatusList, out string errstring)

##### Python prototype

[PositionerHardwareStatusList, errstring] PositionerHardwareStatusListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) PositionerHardwareStatusList: PositionerHardwareStatusList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerHardwareStatusListGet command which is used to Positioner hardware status list. Refer to the XPS Programmer's manual to get the command description.

### PositionerDriverStatusListGet

#### Syntax

##### C# prototype

int PositionerDriverStatusListGet(out string PositionerDriverStatusList, out string errstring)

##### Python prototype

[PositionerDriverStatusList, errstring] PositionerDriverStatusListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) PositionerDriverStatusList: PositionerDriverStatusList

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous PositionerDriverStatusListGet command which is used to Positioner driver status list. Refer to the XPS Programmer's manual to get the command description.

### ReferencingActionListGet

#### Syntax

##### C# prototype

int ReferencingActionListGet(out string list, out string errstring)

##### Python prototype

[list, errstring] ReferencingActionListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) list: list

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ReferencingActionListGet command which is used to Get referencing action list. Refer to the XPS Programmer's manual to get the command description.

### ReferencingSensorListGet

#### Syntax

##### C# prototype

int ReferencingSensorListGet(out string list, out string errstring)

##### Python prototype

[list, errstring] ReferencingSensorListGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) list: list

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ReferencingSensorListGet command which is used to Get referencing sensor list. Refer to the XPS Programmer's manual to get the command description.

### SystemIniParameterGet

#### Syntax

##### C# prototype

int SystemIniParameterGet(string SectionName, string ParameterName, out string ParameterValue, out string errstring)

##### Python prototype

[ParameterValue, errstring] SystemIniParameterGet (SectionName, ParameterName)

#### Parameters

##### Input parameters

(string) SectionName: SectionName

(string) ParameterName: ParameterName

##### Output parameters

(string) ParameterValue: ParameterValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SystemIniParameterGet command which is used to Return the system.ini parameter. Refer to the XPS Programmer's manual to get the command description.

### SystemIniParameterSet

#### Syntax

##### C# prototype

int SystemIniParameterSet(string SectionName, string ParameterName, string ParameterValue, out string errstring)

##### Python prototype

[errstring] SystemIniParameterSet (SectionName, ParameterName, ParameterValue)

#### Parameters

##### Input parameters

(string) SectionName: SectionName

(string) ParameterName: ParameterName

(string) ParameterValue: ParameterValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SystemIniParameterSet command which is used to Save the system.ini parameter. Refer to the XPS Programmer's manual to get the command description.

### SystemRefParameterGet

#### Syntax

##### C# prototype

int SystemRefParameterGet(string SectionName, string ParameterName, out string ParameterValue, out string errstring)

##### Python prototype

[ParameterValue, errstring] SystemRefParameterGet (SectionName, ParameterName)

#### Parameters

##### Input parameters

(string) SectionName: SectionName

(string) ParameterName: ParameterName

##### Output parameters

(string) ParameterValue: ParameterValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SystemRefParameterGet command which is used to Return the system.ref parameter. Refer to the XPS Programmer's manual to get the command description.

### SystemRefParameterSet

#### Syntax

##### C# prototype

int SystemRefParameterSet(string SectionName, string ParameterName, string ParameterValue, out string errstring)

##### Python prototype

[errstring] SystemRefParameterSet (SectionName, ParameterName, ParameterValue)

#### Parameters

##### Input parameters

(string) SectionName: SectionName

(string) ParameterName: ParameterName

(string) ParameterValue: ParameterValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SystemRefParameterSet command which is used to Save the system.ref parameter. Refer to the XPS Programmer's manual to get the command description.

### GatheringUserDatasGet

#### Syntax

##### C# prototype

int GatheringUserDatasGet(out double UserData1, out double UserData2, out double UserData3, out double UserData4, out double UserData5, out double UserData6, out double UserData7, out double UserData8, out double UserData9, out double UserData10, out double UserData11, out double UserData12, out double UserData13, out double UserData14, out double UserData15, out double UserData16, out string errstring)

##### Python prototype

[UserData1, UserData2, UserData3, UserData4, UserData5, UserData6, UserData7, UserData8, UserData9, UserData10, UserData11, UserData12, UserData13, UserData14, UserData15, UserData16, errstring] GatheringUserDatasGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) UserData1: UserData1

(double) UserData2: UserData2

(double) UserData3: UserData3

(double) UserData4: UserData4

(double) UserData5: UserData5

(double) UserData6: UserData6

(double) UserData7: UserData7

(double) UserData8: UserData8

(double) UserData9: UserData9

(double) UserData10: UserData10

(double) UserData11: UserData11

(double) UserData12: UserData12

(double) UserData13: UserData13

(double) UserData14: UserData14

(double) UserData15: UserData15

(double) UserData16: UserData16

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous GatheringUserDatasGet command which is used to Return UserDatas values. Refer to the XPS Programmer's manual to get the command description.

### ControllerMotionKernelMinMaxTimeLoadGet

#### Syntax

##### C# prototype

int ControllerMotionKernelMinMaxTimeLoadGet(out double MinimumCPUTotalLoadRatio, out double MaximumCPUTotalLoadRatio, out double MinimumCPUCorrectorLoadRatio, out double MaximumCPUCorrectorLoadRatio, out double MinimumCPUProfilerLoadRatio, out double MaximumCPUProfilerLoadRatio, out double MinimumCPUServitudesLoadRatio, out double MaximumCPUServitudesLoadRatio, out string errstring)

##### Python prototype

[MinimumCPUTotalLoadRatio, MaximumCPUTotalLoadRatio, MinimumCPUCorrectorLoadRatio, MaximumCPUCorrectorLoadRatio, MinimumCPUProfilerLoadRatio, MaximumCPUProfilerLoadRatio, MinimumCPUServitudesLoadRatio, MaximumCPUServitudesLoadRatio, errstring] ControllerMotionKernelMinMaxTimeLoadGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) MinimumCPUTotalLoadRatio: MinimumCPUTotalLoadRatio

(double) MaximumCPUTotalLoadRatio: MaximumCPUTotalLoadRatio

(double) MinimumCPUCorrectorLoadRatio: MinimumCPUCorrectorLoadRatio

(double) MaximumCPUCorrectorLoadRatio: MaximumCPUCorrectorLoadRatio

(double) MinimumCPUProfilerLoadRatio: MinimumCPUProfilerLoadRatio

(double) MaximumCPUProfilerLoadRatio: MaximumCPUProfilerLoadRatio

(double) MinimumCPUServitudesLoadRatio: MinimumCPUServitudesLoadRatio

(double) MaximumCPUServitudesLoadRatio: MaximumCPUServitudesLoadRatio

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerMotionKernelMinMaxTimeLoadGet command which is used to Get controller motion kernel minimum and maximum time load. Refer to the XPS Programmer's manual to get the command description.

### ControllerMotionKernelMinMaxTimeLoadReset

#### Syntax

##### C# prototype

int ControllerMotionKernelMinMaxTimeLoadReset( out string errstring)

##### Python prototype

[errstring] ControllerMotionKernelMinMaxTimeLoadReset ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerMotionKernelMinMaxTimeLoadReset command which is used to Reset controller motion kernel min/max time load. Refer to the XPS Programmer's manual to get the command description.

### ControllerMotionKernelPeriodMinMaxGet

#### Syntax

##### C# prototype

int ControllerMotionKernelPeriodMinMaxGet(out double MinimumCorrectorPeriod, out double MaximumCorrectorPeriod, out double MinimumProfilerPeriod, out double MaximumProfilerPeriod, out double MinimumServitudesPeriod, out double MaximumServitudesPeriod, out string errstring)

##### Python prototype

[MinimumCorrectorPeriod, MaximumCorrectorPeriod, MinimumProfilerPeriod, MaximumProfilerPeriod, MinimumServitudesPeriod, MaximumServitudesPeriod, errstring] ControllerMotionKernelPeriodMinMaxGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) MinimumCorrectorPeriod: MinimumCorrectorPeriod

(double) MaximumCorrectorPeriod: MaximumCorrectorPeriod

(double) MinimumProfilerPeriod: MinimumProfilerPeriod

(double) MaximumProfilerPeriod: MaximumProfilerPeriod

(double) MinimumServitudesPeriod: MinimumServitudesPeriod

(double) MaximumServitudesPeriod: MaximumServitudesPeriod

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerMotionKernelPeriodMinMaxGet command which is used to Get controller motion kernel min/max periods. Refer to the XPS Programmer's manual to get the command description.

### ControllerMotionKernelPeriodMinMaxReset

#### Syntax

##### C# prototype

int ControllerMotionKernelPeriodMinMaxReset( out string errstring)

##### Python prototype

[errstring] ControllerMotionKernelPeriodMinMaxReset ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ControllerMotionKernelPeriodMinMaxReset command which is used to Reset controller motion kernel min/max periods. Refer to the XPS Programmer's manual to get the command description.

### SocketsStatusGet

#### Syntax

##### C# prototype

int SocketsStatusGet(out string SocketsStatus, out string errstring)

##### Python prototype

[SocketsStatus, errstring] SocketsStatusGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) SocketsStatus: SocketsStatus

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous SocketsStatusGet command which is used to Get sockets current status. Refer to the XPS Programmer's manual to get the command description.

### TestTCP

#### Syntax

##### C# prototype

int TestTCP(string InputString, out string ReturnString, out string errstring)

##### Python prototype

[ReturnString, errstring] TestTCP (InputString)

#### Parameters

##### Input parameters

(string) InputString: InputString

##### Output parameters

(string) ReturnString: ReturnString

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous TestTCP command which is used to Test TCP/IP transfert. Refer to the XPS Programmer's manual to get the command description.

### ISRCorrectorCompensateOverrunNumberGet

#### Syntax

##### C# prototype

int ISRCorrectorCompensateOverrunNumberGet(out int CorrectorOverrunCompensationNumber, out int CorrectorOverrunCompensationMissNumber, out string errstring)

##### Python prototype

[CorrectorOverrunCompensationNumber, CorrectorOverrunCompensationMissNumber, errstring] ISRCorrectorCompensateOverrunNumberGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(int) CorrectorOverrunCompensationNumber: CorrectorOverrunCompensationNumber

(int) CorrectorOverrunCompensationMissNumber: CorrectorOverrunCompensationMissNumber

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ISRCorrectorCompensateOverrunNumberGet command which is used to Get ISR Corrector Compensate Overrun Number. Refer to the XPS Programmer's manual to get the command description.

### ISRCorrectorCompensateOverrunNumberReset

#### Syntax

##### C# prototype

int ISRCorrectorCompensateOverrunNumberReset( out string errstring)

##### Python prototype

[errstring] ISRCorrectorCompensateOverrunNumberReset ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous ISRCorrectorCompensateOverrunNumberReset command which is used to Reset ISR Corrector Compensate Overrun Number. Refer to the XPS Programmer's manual to get the command description.

### OptionalModuleExecute

#### Syntax

##### C# prototype

int OptionalModuleExecute(string ModuleFileName, out string errstring)

##### Python prototype

[errstring] OptionalModuleExecute (ModuleFileName)

#### Parameters

##### Input parameters

(string) ModuleFileName: ModuleFileName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous OptionalModuleExecute command which is used to Execute an optional module. Refer to the XPS Programmer's manual to get the command description.

### OptionalModuleKill

#### Syntax

##### C# prototype

int OptionalModuleKill(string TaskName, out string errstring)

##### Python prototype

[errstring] OptionalModuleKill (TaskName)

#### Parameters

##### Input parameters

(string) TaskName: TaskName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous OptionalModuleKill command which is used to Kill an optional module. Refer to the XPS Programmer's manual to get the command description.

### CIERegister32ValueGet

#### Syntax

##### C# prototype

int CIERegister32ValueGet(string PositionerName, short BoardOffset, out int RegisterValue, out string errstring)

##### Python prototype

[RegisterValue, errstring] CIERegister32ValueGet (PositionerName, BoardOffset)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(short) BoardOffset: BoardOffset

##### Output parameters

(int) RegisterValue: RegisterValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CIERegister32ValueGet command which is used to Get register value of the CIE board. Refer to the XPS Programmer's manual to get the command description.

### CIERegister64ValueGet

#### Syntax

##### C# prototype

int CIERegister64ValueGet(string PositionerName, short BoardOffset, out int RegisterValue, out string errstring)

##### Python prototype

[RegisterValue, errstring] CIERegister64ValueGet (PositionerName, BoardOffset)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(short) BoardOffset: BoardOffset

##### Output parameters

(int) RegisterValue: RegisterValue

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CIERegister64ValueGet command which is used to Get register value of the CIE board. Refer to the XPS Programmer's manual to get the command description.

### CIERegisterValueSet

#### Syntax

##### C# prototype

int CIERegisterValueSet(string PositionerName, short BoardOffset, int RegisterValue, out string errstring)

##### Python prototype

[errstring] CIERegisterValueSet (PositionerName, BoardOffset, RegisterValue)

#### Parameters

##### Input parameters

(string) PositionerName: PositionerName

(short) BoardOffset: BoardOffset

(int) RegisterValue: RegisterValue

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CIERegisterValueSet command which is used to Set register value of the CIE board. Refer to the XPS Programmer's manual to get the command description.

### DebugTraceCommunicationReset

#### Syntax

##### C# prototype

int DebugTraceCommunicationReset( out string errstring)

##### Python prototype

[errstring] DebugTraceCommunicationReset ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous DebugTraceCommunicationReset command which is used to Reset rolling buffer. Refer to the XPS Programmer's manual to get the command description.

### DebugTraceCommunicationSave

#### Syntax

##### C# prototype

int DebugTraceCommunicationSave( out string errstring)

##### Python prototype

[errstring] DebugTraceCommunicationSave ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous DebugTraceCommunicationSave command which is used to Save rolling buffer. Refer to the XPS Programmer's manual to get the command description.

### RunTraceloggerProcessWithTimeSetting

#### Syntax

##### C# prototype

int RunTraceloggerProcessWithTimeSetting(int TraceloggerTime, out string errstring)

##### Python prototype

[errstring] RunTraceloggerProcessWithTimeSetting (TraceloggerTime)

#### Parameters

##### Input parameters

(int) TraceloggerTime: TraceloggerTime

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous RunTraceloggerProcessWithTimeSetting command which is used to Run Tracelogger Process With Time Setting. Refer to the XPS Programmer's manual to get the command description.

### RunTraceloggerProcessWithRollingBuffer

#### Syntax

##### C# prototype

int RunTraceloggerProcessWithRollingBuffer( out string errstring)

##### Python prototype

[errstring] RunTraceloggerProcessWithRollingBuffer ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous RunTraceloggerProcessWithRollingBuffer command which is used to Run Tracelogger Process With Rolling Buffer. Refer to the XPS Programmer's manual to get the command description.

### CreateQNXEvent

#### Syntax

##### C# prototype

int CreateQNXEvent(int EventNumber, string EventName, out string errstring)

##### Python prototype

[errstring] CreateQNXEvent (EventNumber, EventName)

#### Parameters

##### Input parameters

(int) EventNumber: EventNumber

(string) EventName: EventName

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous CreateQNXEvent command which is used to Run Create Event. Refer to the XPS Programmer's manual to get the command description.

### StartEventsAcqusition

#### Syntax

##### C# prototype

int StartEventsAcqusition( out string errstring)

##### Python prototype

[errstring] StartEventsAcqusition ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous StartEventsAcqusition command which is used to Start Events Acqusition. Refer to the XPS Programmer's manual to get the command description.

### StopEventsAcqusition

#### Syntax

##### C# prototype

int StopEventsAcqusition( out string errstring)

##### Python prototype

[errstring] StopEventsAcqusition ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous StopEventsAcqusition command which is used to Stop Events Acqusition. Refer to the XPS Programmer's manual to get the command description.

### EventTriggerSet

#### Syntax

##### C# prototype

int EventTriggerSet( out string errstring)

##### Python prototype

[errstring] EventTriggerSet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous EventTriggerSet command which is used to Generate internal event. Refer to the XPS Programmer's manual to get the command description.

### DebugCorrectorTimeUsageGet

#### Syntax

##### C# prototype

int DebugCorrectorTimeUsageGet(out double ISRInitParam0Duration, out double ISRInitParam1Duration, out double ISRTimerDuration, out double ISRGPIODuration, out double ISRGroupsDuration, out double ISREventDuration, out double ISRGeneralInhibDuration, out double ISRGatheringDuration, out double ISRInitParam2Duration, out string errstring)

##### Python prototype

[ISRInitParam0Duration, ISRInitParam1Duration, ISRTimerDuration, ISRGPIODuration, ISRGroupsDuration, ISREventDuration, ISRGeneralInhibDuration, ISRGatheringDuration, ISRInitParam2Duration, errstring] DebugCorrectorTimeUsageGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) ISRInitParam0Duration: ISRInitParam0Duration

(double) ISRInitParam1Duration: ISRInitParam1Duration

(double) ISRTimerDuration: ISRTimerDuration

(double) ISRGPIODuration: ISRGPIODuration

(double) ISRGroupsDuration: ISRGroupsDuration

(double) ISREventDuration: ISREventDuration

(double) ISRGeneralInhibDuration: ISRGeneralInhibDuration

(double) ISRGatheringDuration: ISRGatheringDuration

(double) ISRInitParam2Duration: ISRInitParam2Duration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous DebugCorrectorTimeUsageGet command which is used to Get Corrector segmentation durations. Refer to the XPS Programmer's manual to get the command description.

### DebugProfilerTimeUsageGet

#### Syntax

##### C# prototype

int DebugProfilerTimeUsageGet(out double ISRCorrectProfilDelta, out double ISRGroupsDuration, out double ISRInitParamDuration, out string errstring)

##### Python prototype

[ISRCorrectProfilDelta, ISRGroupsDuration, ISRInitParamDuration, errstring] DebugProfilerTimeUsageGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) ISRCorrectProfilDelta: ISRCorrectProfilDelta

(double) ISRGroupsDuration: ISRGroupsDuration

(double) ISRInitParamDuration: ISRInitParamDuration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous DebugProfilerTimeUsageGet command which is used to Get profiler segmentation durations. Refer to the XPS Programmer's manual to get the command description.

### DebugServitudesTimeUsageGet

#### Syntax

##### C# prototype

int DebugServitudesTimeUsageGet(out double ISRProfilServDelta, out double ISRErrorCheckDuration, out double ISRGroupsDuration, out double ISRInitParamDuration, out string errstring)

##### Python prototype

[ISRProfilServDelta, ISRErrorCheckDuration, ISRGroupsDuration, ISRInitParamDuration, errstring] DebugServitudesTimeUsageGet ()

#### Parameters

##### Input parameters

None

##### Output parameters

(double) ISRProfilServDelta: ISRProfilServDelta

(double) ISRErrorCheckDuration: ISRErrorCheckDuration

(double) ISRGroupsDuration: ISRGroupsDuration

(double) ISRInitParamDuration: ISRInitParamDuration

(string) errString: The failure reason

##### Return

(int) error code: 0 in success and -1 on failure

#### Description

This function is used to process synchrounous DebugServitudesTimeUsageGet command which is used to Get Servitudes segmentation durations. Refer to the XPS Programmer's manual to get the command description.