**Equipment Driver Interface**

|  |  |  |  |
| --- | --- | --- | --- |
| Rev. | Author(s) | Date | Comments |
| 1A | James.xu | 2014-05-24 | Initial release |
| 1B | jessica | 2014-6-03 | Update as phase 2 code |
| 1C | jessica | 2014-6-10 | Update as combination code |
| 1D | jessica | 2014-11-25 | Check cmd,change for CGR4… |
|  |  |  |  |

目录

[概要： 3](#_Toc388697704)

[FrameWork 3](#_Toc388697705)

[DutStruct 3](#_Toc388697706)

[EquipmentStruct 4](#_Toc388697707)

[EquipmentBase 5](#_Toc388697708)

[Attennuator 5](#_Toc388697709)

[ElecSwitch 7](#_Toc388697710)

[ErrorDetector 8](#_Toc388697711)

[OpticalSwitch 9](#_Toc388697712)

[PowerMeter 10](#_Toc388697713)

[Powersupply 11](#_Toc388697714)

[PPG 12](#_Toc388697715)

[Scope 15](#_Toc388697716)

[Thermocontroller 21](#_Toc388697717)

[Driver 22](#_Toc388697718)

[AQ2011OpticalSwitch 22](#_Toc388697719)

[AQ2011PowerMeter 23](#_Toc388697720)

[AQ2211Atten 25](#_Toc388697721)

[D86100 26](#_Toc388697722)

[E3631 31](#_Toc388697723)

[ElectricalSwitch 32](#_Toc388697724)

[FLEX86100 33](#_Toc388697725)

[MP1800PPG 41](#_Toc388697726)

[MP1800ED 44](#_Toc388697727)

[N490XPPG 47](#_Toc388697728)

[N490XED 49](#_Toc388697729)

[TPO4300 50](#_Toc388697730)

# 概要：

|  |  |  |  |
| --- | --- | --- | --- |
| **FrameWork Class** | **Driver Class** | | |
| EquipmentBase |  |  |  |
| Attennuator | AQ2211Atten |  |  |
| PowerMeter | AQ2211PowerMeter |  |  |
| OpticalSwitch | AQ2211OpticalSwitch |  |  |
| ElecSwitch | ElectricalSwitch |  |  |
| DUT | QSFPDUT | SFPDUT | XFPDUT |
| PPG | N490XPPG | MP1800PPG |  |
| ErrorDetector | N490XED | MP1800ED |  |
| Powersupply | E3631 |  |  |
| Scope | D86100 | FLEX86100 |  |
| Thermocontroller | TPO4300 |  |  |

# FrameWork

## DutStruct

1. 类名：public struct DutStruct
2. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | FiledName |  |
| public | byte | Channel |  |
| public | int | SlaveAdress | 删除不用 |
| public | byte | EngPage |  |
| public | int | StartAddress |  |
| public | byte | Length |  |
| public | byte | TempSelect | 删除不用 |
| public | byte | VccSelect | 删除不用 |
| public | byte | DebugStartAddress | 删除不用 |
| public | byte | Format |  |
| public | byte | ChipLine | 删除不用 |

## DriverStruct

1. 类名：public struct DutStruct
2. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | FiledName |  |
| public | byte | MoudleLine |  |
| public | int | RegisterAddress |  |
| public | byte | Length |  |
| public | byte | DriverType |  |
| public | byte | ChipLine |  |

## EquipmentStruct

1. 类名：public struct EquipmentStruct
2. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | FiledName | 属性名称 |
| public | byte | DefaultValue | 属性值 |

## EquipmentBase

1. 类名：public struct EquipmentBase
2. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | IOPort | MyIO | 仪器通信IO口 |
| public | bool | EquipmentConnectflag | 仪器是否连接成功 |
| public | bool | EquipmentConfigflag | 仪器是否已配置 |
| public | IOPort | USBIO | DUT通信IO口 |
| public | string | CurrentChannel | 切换通道时，用来记录当前通道 |
| public | string | log | 输出log信息，用于调试，或者信息输出 |
| public | SortedList | offsetlist | 收集页面传值，校准仪器使用 |
| public | int | Referenced\_Times | 仪器被使用的次数 |
| public | bool | bReusable | 仪器可否被复用 |
| public | bool | EquipmentErrorflag | 仪器指令出错标志 |
| public | bool | bReady | 仪器是否可被testmodel使用 |

3.方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | bool | Configure |  | 配置仪器 |
| public virtual | bool | Initialize | TestModeEquipmentParameters[]  LIST | 初始化仪器字段 |
| public virtual | bool | Initialize | DutStruct[] LIST | 初始化DUT字段 |
| public virtual | bool | Initialize | DutStruct[] DutList, DriverStruct[] DriverList,string AuxAttribles | 初始化DUT字段 |
| public virtual | bool | Initialize | DutStruct[] DutList, DriverStruct[] DriverList,DriverInitializeStruct[] DriverinitList, string AuxAttribles | 初始化DUT字段 |
| public virtual | bool | Connect |  | 连接仪器 |
| public virtual | bool | ChangeChannel | string channel | 切换通道 |
| public | String | GetLogInfo |  | 获取log信息 |
| public virtual | bool | configoffset | string channel，string offset | 配置不同模块通道对应的offset |
| public | Void | IncreaseReferencedTimes |  | 仪器被使用次数加一 |
| public | Void | DecreaseReferencedTimes |  | 释放仪器时使用次数减一 |
| public virtual | bool | Switch | bool Switch（衰减器输出开关）, int syn = 0(缺省参数，0异步，1同步) | 开关 |

## Attennuator

1. 类名：public class Attennuator : EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：public Attennuator()
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | AttChannel | 衰减器所处通道 |
| public | string | AttSlot | 衰减器所处槽位 |
| public | string | AttValue | 衰减值 |
| public | string | TotalChannel | 模块通道数 |
| public | string | Wavelength | 衰减器波长组合 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |
| public | Int | opendelay | 打开开关后延时 |
| public | Int | closedelay | 关闭开关后延时 |
| public | Int | setattdelay | 设置衰减后延时 |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | bool | ConfigWavelength | int syn = 0  （缺省参数） | 设置衰减器的波长 |
| public virtual | bool | AttnValue | string value（衰减量），int syn = 0  （缺省参数，0异步，1同步） | 设置衰减器的衰减量 |
| public virtual | bool | AddCalFactor | string CalFactor（衰减器修正因子） | 设置衰减器的修正因子 |
| public virtual | double | GetAtten |  | 读取衰减器的衰减量 |

## ElecSwitch

1. 类名：public class ElecSwitch : EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | ElecSwitchChannel | 电开关切换的目标通道 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | bool | ChangeElecSwitchChannel |  | 切换到目标通道 |

## ErrorDetector

1. 类名：public class ErrorDetector : EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |
| public | bool | CDRSwitch | CDR开关 |
| public | string | CDRFreq | CDR频率，例如10.3125 |
| public | string | PRBS | PRBS长度，例如7,31 |
| MP1800 add |  |  |  |
| public | byte | slot | ED所处槽位 |
| public | byte | totalChannels | ED共有几个通道 |
| public | byte | currentChannel | ED的当前通道 |
| public | byte | dataInputInterface | ED数据输入接口类型0=SingleEnd,1=Diff 50ohm,2=Diff 100ohm |
| public | byte | prbsLength | PRBS码型长度 |
| public | byte | patternType | ED码型选择0=PRBS,1=Zero Subsitution,2=Data,3=Alternate,4=Mixed Data,5=Sequense |
| public | byte | errorResultZoom | ED误码信息显示类型0=ZoomIn(显示详细误码信息),1=ZoomOut(只显示误码率和误码数) |
| public | byte | edGatingMode | ED误码累积模式0=REPeat,1=SINGle,2=UNTimed |
| public | byte | edGatingUnit | ED误码累积单位0=TIME,1=CLOCk,2=ERRor,3=BLOCk |
| public | int | edGatingTime | ED误码累积时间 |
| public | string | patternfile | 用户定义码型文件地址 |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | double | GetErrorRate | int syn = 0  （缺省参数，0异步，1同步） | 读取误码率 |
| public virtual | bool | ConfigureCDRONOFF | bool Switch（CDR的开关） | 设置CDR开关 |
| public virtual | bool | ConfigureCDRFreq |  | 设置CDR频率 |
| public virtual | bool | AutoAlaign | bool becenter（是否处于数据中间） | 自动搜寻误码判决点 |
| public virtual | bool | ConfigurePRBS |  | 设置PRBS码型 |
| MP1800 add |  |  |  |  |
| public virtual | bool | AutoAlaign |  | 自动搜寻误码判决点 |

## OpticalSwitch

1. 类名：public class OpticalSwitch : EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：public OpticalSwitch();
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | OpticalSwitchSlot | 光开关所处槽位 |
| public | string | SwitchChannel | 光开关中哪一列 |
| public | string | ToChannel | 切换到目标通道数 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |

1. 方法

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | bool | Switchchannel | int syn = 0  （缺省参数，0异步，1同步） | 切换光开关通道 |

## PowerMeter

1. 类名：public class PowerMeter : EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：public PowerMeter();
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |
| public | string | PowerMeterSlot | 光功率计所处槽位 |
| public | string | PowerMeterWavelength | 光功率计波长 |
| public | string | PowerMeterChannel | 光功率计通道 |
| public | string | UnitType | 光功率单位，0=dBm,1=W |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | bool | ConfigWavelength | int syn = 0  （缺省参数，0异步，1同步） | 设置光功率计波长 |
| public virtual | double | ReadPower |  | 读取光功率值 |
| public virtual | bool | Selectunit | int syn = 0  （缺省参数，0异步，1同步） | 设置光功率的单位 |

## Powersupply

1. 类名：public class Powersupply: EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：public Powersupply ();
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |
| public | string | DutChannel | DUT电源所处通道 |
| public | string | OptSourceChannel | 光源电源所处通道 |
| public | string | DutVoltage | DUT的电源电压值 |
| public | string | DutCurrent | DUT的电流值 |
| public | string | OptVoltage | 光源的电源电压值 |
| public | string | OptCurrent | 光源的电流值 |
| public | string | voltageoffset | 电源电压补偿 |
| public | string | currentoffset | 电流补偿 |
| public | Int | opendelay | 开关打开延时 |
| public | int | closedelay | 开关关闭延时 |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | bool | ConfigVoltageCurrent | string voltage（电压值，单位V） | 设置电源的电压和限流 |
| public virtual | double | GetCurrent |  | 读取电流值，返回值单位为mA |
| public virtual | double | GetVoltage |  | 读取电压值，返回值单位为V |

## PPG

1. 类名：public class PPG: EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：public PPG ();
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |
| public | string | TriggerDRatio | 触发信号是时钟的几分频 |
| public | string | TriggerMode | 触发信号的模式"DCL""PATT" |
| public | string | ClockHigVoltage | 时钟信号的高电位电压值，单位V |
| public | string | ClockLowVoltage | 时钟信号的低电位电压值，单位V |
| public | string | DataHigVoltage | 数据信号的高电位电压值，单位V |
| public | string | DataLowVoltage | 数据信号的低电位电压值，单位V |
| public | string | BertDataRate | 数据信号的比特率，单位bps |
| public | string | PRBS | PRBS码型长度 |
| MP1800 ADD |  |  |  |
| public | string | configFilePath | 设备中初始化文件地址 |
| public | byte | slot | 所处槽位 |
| public | byte | clockSource | 参考时钟选择内部还是外部时钟,0=InternalClock |
| public | byte | auxOutputClkDiv | Auxoutput输出时钟信号是数据传输速率的几分频 |
| public | byte | totalChannels | 通道总数 |
| public | byte | prbsLength | PRBS码型长度 |
| public | byte | patternType | PRBS码型选择0=PRBS,1=Zero Subsitution,2=Data,3=Alternate,4=Mixed Data,5=Sequense |
| public | byte | dataSwitch | 数据输出开关 |
| public | byte | dataTrackingSwitch | 数据DATA和/DATA跟踪开关 |
| public | byte | dataLevelGuardSwitch | 输出幅度保护开关 |
| public | byte | dataAcModeSwitch | 输出交流模式开关 |
| public | byte | dataLevelMode | 输出电平类型0=VARiable,1=NECL,2=PCML,3=NCML,4=SCFL,5=LVPecl,6=LVDS200,7=LVDS400 |
| public | byte | clockSwitch | 时钟输出开关 |
| public | byte | outputSwitch | 总体信号开关 |
| public | string | dataRate | 比特率，单位为Gbps |
| public | double | dataLevelGuardAmpMax | 单端电信号幅度最大值单位是mV |
| public | double | dataLevelGuardOffsetMax | 单端电信号最大偏移量单位是mV |
| public | double | dataLevelGuardOffsetMin | 单端电信号最小偏移量单位是mV |
| public | double | dataAmplitude | 单端电信号输出幅度单位mV |
| public | double | dataCrossPoint | 电信号交点单位% |
| public | string | patternfile | 用户自定义码型文件地址 |

5.方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | bool | ConfigureDataRate |  | 设置PPG速率单位bps |
| public virtual | bool | ConfigurePRBS |  | 设置PPG PRBS长度 |
| public virtual | bool | ConfigureDataVoltage |  | 设置数据的电平幅度 |
| public virtual | bool | ConfigureClockVoltage |  | 设置时钟的电平幅度 |
| public virtual | bool | ConfigureTriggerMode |  | 设置触发模式 |
| public virtual | bool | ConfigureTriggerDRatio |  | 设置触发分频量 |

## Scope

1. 类名：public class Scope: EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：public Scope ();
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |
| public | string | OptChannel | 光通道 |
| public | string | ElecChannel | 电通道 |
| public | string | Scale | 屏幕纵向显示比例 |
| public | string | Offset | 屏幕纵向显示偏移量 |
| public | string | MaskName | 模板名称 |
| public | string | DcaAtt | 光口补偿值 |
| public | string | FilterFreq | 滤波器频率 |
| public | byte | Percentage | 模板余量百分比 |
| public | string | DcaThreshold | RiseTime FallTime阈值点 |
| public | string | TriggerBwlimit | 触发信号带宽选择 |
| public | string | DcaWavelength | 示波器的波长 |
| public | string | DcaDataRate | 示波器的速率 |
| public | int | WaveformCount | 波形数 |
| public | int | setscaledelay | 设置scale后延时 |
| FLEX86100 |  |  |  |
| public | string | configFilePath | 配置文件地址 |
| public | byte | FilterSwitch | 滤波器开关 |
| public | byte | triggerSource | 触发源选择0=FrontPanel,1=FreeRun |
| public | byte | opticalSlot | 光信号所处模组的槽位 |
| public | byte | elecSlot | 电信号所处模组的槽位 |
| public | byte | opticalAttSwitch | 光口补偿开关 |
| public | byte | erFactorSwitch | ER补偿值开关 |
| public | double | erFactor | ER补偿值单位% |
| public | byte | FlexOptChannel | 光信号所处通道 |
| public | byte | FlexElecChannel | 电信号所处通道 |
| public | byte | precisionTimebaseModuleSlot | 精准时基模组所处槽位 |
| public | byte | precisionTimebaseSynchMethod | 精准时基同步方式选择0=OLIN,1=FAST |
| public | byte | rapidEyeSwitch | 快速眼图开关 |
| public | byte | Threshold | RiseTime，FallTime阈值点0=80%50%20%,1=90%50%10% |
| public | byte | reference | RiseTime，FallTime参考点0=OneZero,1=VtopVbase |
| public | byte | marginType | 模板余量测试方法false=Manual,true=Auto |
| public | byte | marginHitType | 模板余量碰撞类型0=HitCount,1=HitRatio |
| public | byte | acqLimitType | 眼图累积限制方式0=Waveforms,1=Samples |
| public | double | precisionTimebaseRefClk | 精准时基参考时钟，例如10.3125e9 |
| public | double | marginHitRatio | 模板余量碰撞比例，例如5E-6 |
| public | string | opticalMaskName | 光眼图模板名称 |
| public | string | elecMaskName | 电眼图模板名称 |
| public | string | opticalEyeSavePath | 光眼图存储地址 |
| public | string | elecEyeSavePath | 电眼图存储地址 |
| public | int | marginHitCount | 模板余量碰撞数量 |
| public | int | acqLimitNumber | 眼图累积限制数量 |
| public | byte | FlexTriggerBwlimit | 触发源带宽选择0=FilterdEdge,1=StandardEdge,2=Clock/Divided |
| public | byte | FlexDcaWavelength | 波长选择0=850,1=1310,2=1550 |
| public | double | FlexDcaDataRate | 信号速率 |
| public | double | FlexFilterFreq | 滤波器速率，例如10.3125e9 |
| public | double | FlexDcaAtt | 光口补偿值 |
| public | double | FlexScale | 屏幕纵向显示比例 |
| public | double | FlexOffset | 屏幕纵向显示偏移量 |
| public | int | flexsetscaledelay | 设置scale后延时 |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | bool | ClearDisplay |  | 刷新屏幕 |
| public virtual | bool | OpenOpticalChannel | bool Switch，int syn = 0  （缺省参数，0异步，1同步） | 切换光电通道 |
| public virtual | bool | SetMode | byte Mode(0=”EYE”;1=”OSC”;2=”TDR” )，int syn = 0  （缺省参数，0异步，1同步） | 设置显示模式 |
| public virtual | bool | AutoScale | int syn = 0  （缺省参数，0异步，1同步） | 自动调整眼图显示 |
| public virtual | bool | RunStop | bool run | 采样启动和停止 |
| public virtual | double | GetAveragePowerWatt |  | 读取平均光功率单位W |
| public virtual | double | GetAveragePowerdbm |  | 读取平均光功率单位dBm |
| public virtual | bool | MaskONOFF | bool MaskOn（模板开关），int syn = 0  （缺省参数，0异步，1同步） | 模板开关 |
| public virtual | bool | SetscaleOffset |  | 设置屏幕显示比例 |
| public virtual | bool | SetMaskAlignMethod | byte method(0=DISPLAY,1=EWINDOW)，int syn = 0  （缺省参数，0异步，1同步） | 设置模板对齐方式 |
| public virtual | bool | SetRunTilOff | int syn = 0  （缺省参数，0异步，1同步） | 眼图累积限制开关 |
| public virtual | double[] | OpticalEyeTest | int syn = 0  （缺省参数，0异步，1同步） | 光眼图测试 |
| public virtual | double[] | ElecEyeTest | int syn = 0  （缺省参数，0异步，1同步） | 电眼图测试 |

## Thermocontroller

1. 类名：public class Thermocontroller: EquipmentBase
2. 父类：EquipmentBase
3. 实例化函数：public Thermocontroller ();
4. 字段：

|  |  |  |  |
| --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 描述 |
| public | string | IOType | IO口类型，例如"GPIB" |
| public | string | Addr | IO口地址 |
| public | string | Name | 设备名称 |
| public | bool | Reset | 设备是否需要复位 |
| public | string | FLSE | 气流量，默认15 |
| public | string | ULIM | 气流温度上限，单位C |
| public | string | LLIM | 气流温度下限，单位C |
| public | string | Sensor | 0=No Sensor,1=T,2=k,3=rtd,4=diode |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public virtual | string | ReadCurrentTemp |  | 读取当前温度 |
| public virtual | bool | AirFlowSetting |  | 设置气流速度 |
| public virtual | bool | AirTempsUpperlimit | int syn = 0  （缺省参数，0异步，1同步） | 设置气流温度上限 |
| public virtual | bool | AirTempslowerlimit | int syn = 0  （缺省参数，0异步，1同步） | 设置气流温度下限 |
| public virtual | bool | SensorType | int syn = 0  （缺省参数，0异步，1同步） | 设置温度探头种类 |
| public virtual | bool | DUTControlModeOnOFF | byte Switch（1=DUT Control,0=Air Control) int syn = 0  （缺省参数，0异步，1同步） | 设置温度控制源 |
| public virtual | bool | lockHeadPosition | byte Switch，int syn = 0  （缺省参数，0异步，1同步） | 1 UP,0 DOWN |
| public virtual | string | ReadSetpoint |  | 读取罩子位置 |
| public virtual | string | ReadSetpointTemp |  | 读取目标温度值 |
| public virtual | bool | SetPositionUPDown | string position(1=down,0=up)，int syn = 0  （缺省参数，0异步，1同步） | 设置罩子位置 |
| public virtual | bool | SetPoint | string Point(0=HOT,1=AMB,2=COLD) | 设置目标温度区间 |
| public virtual | bool | SetPointTemp | double Temp(目标温度，单位C），int syn = 0（缺省参数，0异步，1同步） | 设置目标温度 |

# Driver

## AQ2211OpticalSwitch

1. 类名：public class AQ2011OpticalSwitch: OpticalSwitch
2. 父类：OpticalSwitch
3. 实例化函数：
4. 字段：
5. 属性列表:

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 20 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | AQ2211OpticalSwitch | 仪器名称 |
| string | OpticalSwitchSlot | 1 | 光开关所处槽位 |
| string | SwitchChannel | 1 | 光开关需要切换的通道 |
| string | ToChannel | 1 | 光开关切换的目标通道 |

1. 方法:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] AQ2211OpticalSwitchstruct | 初始化光开关的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| protected | bool | Switchchannel | string To\_Channel（目标通道），int syn = 0（缺省参数，0异步，1同步） | 切换通道 |
| public override | bool | Switchchannel | int syn = 0（缺省参数，0异步，1同步） | 切换通道 |
| public override | bool | ChangeChannel | String Channel（模块通道），int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset，int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时，收集OFFSET |

## AQ2211PowerMeter

1. 类名：public class AQ2011PowerMeter: PowerMeter
2. 父类：PowerMeter
3. 实例化函数：
4. 字段：
5. 属性列表:

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 20 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | AQ2211PowerMeter | 仪器名称 |
| string | PowerMeterSlot | 3 | 光功率计所处槽位 |
| string | PowerMeterWavelength | 1310 | 光功率计的波长 |
| string | PowerMeterChannel | 1 | 光功率计的通道 |
| string | UnitType | dBm | 功功率值的单位(dBm or W) |

1. 方法:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] AQ2211PowerMeterstruct  （属性参数的结构体数组） | 初始化光功率计的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| public override | bool | ConfigWavelength | int syn = 0（缺省参数，0异步，1同步） | 设置波长 |
| public override | double | ReadPower |  | 读取光功率 |
| public override | bool | Selectunit | int syn = 0（缺省参数，0异步，1同步） | 设置光功率单位 |
| public override | bool | ChangeChannel | String Channel（模块通道），int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset，int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时，收集OFFSET |

## AQ2211Atten

1. 类名：public class AQ2211Atten: Attennuator
2. 父类：Attennuator
3. 实例化函数：
4. 字段：
5. 属性列表:

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 20 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | AQ2211Atten | 仪器名称 |
| string | AttSlot | 2 | 衰减器所处槽位 |
| string | AttValue | 0 | 衰减值 |
| string | Wavelength | 850,1270,1310… | 衰减器波长组合 |
| string | AttChannel | 1 | 衰减器所处通道 |
| string | TotalChannel | 4 | 模块通道总数 |
| Int | opendelay | 1000 | 打开延时 |
| int | closedelay | 500 | 关闭延时 |
| int | setattdelay | 1000 | 设置延时 |

1. 方法:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] AQ2211list  （属性参数的结构体数组） | 初始化衰减器的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| protected | bool | ConfigWavelength | string dutcurrentchannel int syn = 0（缺省参数，0异步，1同步） | 设置波长 |
| public override | bool | ConfigWavelength | int syn = 0（缺省参数，0异步，1同步） | 设置波长 |
| public override | double | AttnValue | string Value（衰减量，dB) | 设置衰减量 |
| public override | bool | Switch | bool Swith（开关）int syn = 0（缺省参数，0异步，1同步） | 输出开关 |
| public override | bool | AddCalFactor | string CalFactor（修正因子） | 设置修正因子 |
| public override | double | GetAtten |  | 读取衰减量 |
| public override | bool | ChangeChannel | String Channel（模块通道）int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时收集 OFFSET |
| protected | bool | SetOffset | string offset int syn = 0（缺省参数，0异步，1同步） | 配置OFFSET |

## D86100

1. 类名：public class D86100: Scope
2. 父类：Scope
3. 实例化函数：public D86100();
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 7 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | D86100 | 仪器名称 |
| string | OptChannel | 1 | 光口通道 |
| string | ElecChannel | 2 | 电口通道 |
| string | Scale | 0 | 屏幕显示比例（单位W/div） |
| string | Offset | 0 | 屏幕显示偏移（单位W） |
| string | DcaAtt | 0 | 光口补偿值（单位dB) |
| string | opticalMaskName | c:\scope\masks\10GBE\_10\_3125\_MAY02.MSK | 模板名称（地址） |
| String | elecMaskName |  |  |
| string | FilterFreq | 10.3125 | 滤波器频率（单位Gbps) |
| byte | Percentage | 10 | 初始模板余量 |
| string | DcaThreshold | 80,50,20 | RiseFallTime阈值点 |
| string | TriggerBwlimit | 2 | 触发信号带宽（0=HIGH 1=LOW 2=DIV） |
| string | DcaWavelength | 2 | 波长选择（1=850,2=1310,3=1550） |
| string | DcaDataRate | 10.3125E9 | 数据速率（单位bps） |
| int | WaveformCount | 700 | 波形数 |
| int | setscaledelay | 1000 | 设置延时 |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] D86100Struct  （属性参数的结构体数组） | 初始化示波器的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| public override | bool | ClearDisplay |  | 刷新屏幕 |
| public override | bool | SetMode | byte Mode(0="EYE";1="OSC";2="TDR" )，int syn = 0（缺省参数，0异步，1同步） | 设置显示模式 |
| public override | bool | AutoScale | int syn = 0（缺省参数，0异步，1同步） | 自动调整眼图显示 |
| public override | bool | RunStop | bool run | 采样启动和停止 |
| private | bool | WavelengthSelect | string channel，string Wavelength（1=850,2=1310,3=1550）int syn = 0（缺省参数，0异步，1同步） | 选择波长 |
| private | double | GetCrossing |  | 读取Crossing |
| public override | double | GetEratio |  | 读取ER |
| private | double | GetJitterPP |  | 读取JitterPP |
| private | double | GetJitterRMS |  | 读取JitterRMS |
| private | double | GetFalltime |  | 读取FallTime |
| private | double | GetRisetime |  | 读取RiseTime |
| public override | double | GetAveragePowerWatt |  | 读取平均光功率单位W |
| public override | double | GetAveragePowerdbm |  | 读取平均光功率单位dBm |
| public override | bool | FileterSwitch | bool Switch | 滤波器开关 |
| Private | bool | FileterSwitch | string channel, bool Switch | 滤波器开关 |
| Private | bool | FileterSelect | string channel | 滤波器选择 |
| string FilterFreq（滤波器速率，单位Gb/s） int syn = 0（缺省参数，0异步，1同步） |
| Private | bool | ClearMeasurements |  | 清除所有显示项目 |
| Private | bool | SelectChannel | int syn = 0（缺省参数，0异步，1同步） | 选择通道 |
| Private | byte | MaskTest | string maskname,string savePath, int syn = 0（缺省参数，0异步，1同步） | 模板测试 |
| private | bool | SetMaskMargin | bool Swith（模板余量开关） | 设置模板余量 |
| private | bool | LoadMask | string maskname | 装载模板 |
| private | bool | MaskAlign |  | 模板对齐 |
| public override | bool | MaskONOFF | bool MaskOn（模板开关）int syn = 0（缺省参数，0异步，1同步） | 模板开关 |
| private | bool | MaskTestStart |  | 模板测试开始 |
| private | double | MaskGetWaveform |  | 读取模板测试失败采样点数 |
| private | bool | SetAttenuation | string channel | 设置光口补偿量 |
| string DCAAttenuation（光口补偿量，单位dB) int syn = 0（缺省参数，0异步，1同步） |
| private | bool | SetOffset | string channel | 设置屏幕显示偏移量 |
| string offsetValue(屏幕显示偏移量，单位W） int syn = 0（缺省参数，0异步，1同步） |
| private | bool | Setscale | string channel | 设置屏幕显示比例 |
| string Scale（屏幕显示比例，单位W/div） |
| string DataRate（数据速率，单位bps) int syn = 0（缺省参数，0异步，1同步） |
| public override | bool | SetScaleOffset | int syn = 0（缺省参数，0异步，1同步） | 设置屏幕显示比例 |
| private | bool | SetTriggerBwlimit | int syn = 0（缺省参数，0异步，1同步） | 设置触发源带宽 |
| private | bool | DCACalibration |  | 校验示波器 |
| public override | bool | SetMaskAlignMethod | Byte method(0=DISPLAY,1=EWINDOW) int syn = 0（缺省参数，0异步，1同步） | 设置模板对齐方式 |
| public override | bool | SetRunTilOff | int syn = 0（缺省参数，0异步，1同步） | 眼图累积限制开关 |
| private | bool | SetChannel | string channel | 通道开关控制 |
| bool on\_off int syn = 0（缺省参数，0异步，1同步） |
| private | bool | CenterEye | int syn = 0（缺省参数，0异步，1同步） | 眼图调整到中间 |
| private | bool | SetRefPos | byte pos(参考点0=left,1=center)  int syn = 0（缺省参数，0异步，1同步） | 设置眼图参考点 |
| private | bool | SetPos | double pos（偏移量，单位s) int syn = 0（缺省参数，0异步，1同步） | 设置眼图偏移量 |
| private | bool | SetHistMod | bool on\_off(Hist模式开关） int syn = 0（缺省参数，0异步，1同步） | Hist模式开关 |
| private | bool | SetHistSource | int syn = 0（缺省参数，0异步，1同步） | Hist模式测试信号源选择 |
| private | bool | SetHistWindow | double x1（Hist窗口横坐标第一点） | 设置Hist窗口大小以及位置 |
| double x2（Hist窗口横坐标第二点） |
| double y1（Hist窗口纵坐标第一点） |
| double y2（Hist窗口纵坐标第二点）, int syn = 0（缺省参数，0异步，1同步） |
| private | bool | GetHistMedian | out double median（Hist窗口中点）, int syn = 0（缺省参数，0异步，1同步） | 读取Hist窗口中点位置 |
| private | bool | SetTimeDelay | double delayTime | 设置延迟时间 |
| private | bool | GetTimeDelay | out double delayTime | 读取延迟时间 |
| private | bool | SetruntilMod | int waveformcount | 设置眼图累积方式 |
| byte off\_waveform\_sample(0="OFF" 1="WAV" 2="SAMP") int syn = 0（缺省参数，0异步，1同步） |
| public override | bool | DisplayER |  | 显示ER |
| private | bool | DisplayEyeAmplitude |  | 显示OMA |
| private | double | ReadEyeAmplitude |  | 读取OMA |
| private | bool | DisplayCrossing |  | 显示Crossing |
| private | bool | WaitUntilComplete |  | 等待眼图累积完毕 |
| private | int | Gettotalmaskerrors |  | 读取模板测试失败采样点数 |
| private | int | Getregionerrors | string region（测试模板余量区域） | 读取模板相应区域失败点数 |
| private | byte | GetTestMargin | byte iniMargin（初始余量） | 读取模板余量 |
| private | bool | ClearMesures |  | 清除所有显示项目 |
| private | bool | SaveEyeDiagram | string savePath | 保存眼图 |
| private | bool | OpenCurrentChannel | string currentChannel int syn = 0（缺省参数，0异步，1同步） | 打开当前通道 |
| public override | bool | OpenOpticalChannel | bool Switch int syn = 0（缺省参数，0异步，1同步） | 打开光通道 |
| private | bool | SetDCAThreshold | int syn = 0（缺省参数，0异步，1同步） | 设置Risetime Falltime测试的阈值点 |
| public override | bool | ChangeChannel | String Channel（模块通道）int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时收集 OFFSET |

## E3631

1. 类名：public class E3631: Powersupply
2. 父类：Powersupply
3. 实例化函数：public E3631 ();
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 5 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | E3631 | 仪器名称 |
| string | DutChannel | 1 | DUT电源的通道（1=P6V，2=P25V） |
| string | OptSourceChannel | 2 | 光源电源的通道（单位V） |
| string | DutVoltage | 3.3 | DUT的电压（单位V） |
| string | DutCurrent | 1 | DUT电源的限流值（单位A） |
| string | OptVoltage | 3.3 | 光源的电压（单位V） |
| string | OptCurrent | 1 | 光源电源的限流值（单位A） |
| string | voltageoffset | 0.2 | 电压补偿 |
| string | currentoffset | 0 | 电流补偿 |
| int | opendelay | 1000 | 打开延时 |
| int | closedelay | 500 | 关闭延时 |

1. 方法：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 作用范围 | 类型 | | 名称 | | 形参 | 描述 | | |
| public override | bool | | Initialize | | TestModeEquipmentParameters[] PSStruct  （属性参数的结构体数组） | 初始化电源的属性参数 | | |
| public override | bool | | Connect | |  | 连接设备 | | |
| public | bool | | ReSet | |  | 复位设备 | | |
| public override | bool | | Configure | | int syn = 0（缺省参数，0异步，1同步） | 配置设备 | | |
| public override | bool | | Switch | | bool Switch int syn = 0（缺省参数，0异步，1同步） | 电源输出开关 | | |
| protected | bool | | ConfigVoltageCurrent | | string channel（通道） | 设置电压和限流 | | |
| string voltage（电压，V） |
| string current（限流，A） |
| public override | bool | | ConfigVoltageCurrent | | string voltage（电压，V） | 配置电压 | | |
| public override | double | | GetCurrent | |  | 读取电流，mA | | |
| public override | double | | GetVoltage | |  | 读取电压，V | | |
| public override | bool | ChangeChannel | | String Channel（模块通道）int syn = 0（缺省参数，0异步，1同步） | | | 切换模块通道 |
| public override | bool | configoffset | | string channel, string offset int syn = 0（缺省参数，0异步，1同步） | | | 切换模块通道时收集 OFFSET |

## ElectricalSwitch

1. 类名：public class ElectricalSwitch: ElecSwitch
2. 父类：ElecSwitch
3. 实例化函数：public ElectricalSwitch ();
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 0 | 地址 |
| string | IOType | USB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | ElectricalSwitch | 名字 |
| string | ElecSwitchChannel | 1 | 电开关切换的目标通道（1,2,3,4） |

1. 方法：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 作用范围 | 类型 | | 名称 | | 形参 | 描述 | |
| public override | bool | | Initialize | | EquipmentStruct[] list（属性参数的结构体数组） | 初始化电开关属性参数 | |
| public override | bool | | Connect | |  | 连接设备 | |
| public | bool | | ReSet | |  | 复位设备 | |
| public override | bool | | Configure | |  | 配置设备 | |
| protected | bool | | ChangeElecSwitchChannel | | String channel（通道） | 切换电开关通道 | |
| public override | bool | | ChangeElecSwitchChannel | |  | 切换电开关通道 | |
| public override | bool | | ChangeChannel | | String Channel（模块通道） | 切换模块通道 | |
| public override | bool | configoffset | | string channel, string offset | | | 切换模块通道时收集 OFFSET |

## FLEX86100

1. 类名：public class FLEX86100: Scope
2. 父类：Scope
3. 实例化函数：public FLEX86100 ();
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 7 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | 86100 | 仪器名称 |
| string | configFilePath | 1 | 仪器内部配置文件地址 |
| double | FlexDcaDataRate | 10.3125E9 | 示波器速率 |
| byte | FilterSwitch | 1 | 滤波器开关 |
| double | FlexFilterFreq | 10.3125E9 | 滤波器速率 |
| byte | triggerSource | 0 | 触发源选择（0=FrontPannel,1=FreeRun) |
| byte | FlexTriggerBwlimit | 2 | 触发信号带宽（0=FILTered，1=EDGE，2=CLOCK） |
| byte | opticalSlot | 1 | 光口所处槽位 |
| byte | elecSlot | 2 | 电口所处槽位 |
| byte | FlexOptChannel | 1 | 光口通道（1=A,2=B,3=C,4=D） |
| byte | FlexElecChannel | 1 | 电口通道（1=A,2=B,3=C,4=D） |
| byte | FlexDcaWavelength | 2 | 波长选择（1=850,2=1310,3=1550） |
| byte | opticalAttSwitch | 1 | 光口补偿开关 |
| double | FlexDcaAtt | 0 | 光口补偿值（单位dB) |
| double | erFactor | 0 | ER修正值（单位%） |
| double | FlexScale | 300 | 屏幕显示比例(单位uW/div） |
| double | FlexOffset | 300 | 屏幕显示偏移(单位uW) |
| byte | Threshold | 0 | RiseFallTime阈值点（0=80,50,20，1=90,50,10） |
| byte | reference | 0 | RiseFallTime参考点（0=OneZero，1=TopBase） |
| byte | precisionTimebaseModuleSlot | 3 | 精准时基单元槽位 |
| byte | precisionTimebaseSynchMethod | 1 | 精准时基同步方式（0=OLIN，1=FAST） |
| double | precisionTimebaseRefClk | 10.3125E9 | 精准时基单元参考时钟（单位bps） |
| byte | rapidEyeSwitch | 1 | 快速眼图模式开关 |
| byte | marginType | 1 | 模板余量测试自动手动选择（0=手动，1=自动） |
| byte | marginHitType | 0 | 自动模板余量测试判决方式选择（0=碰撞点数，1=碰撞比例） |
| double | marginHitRatio | 5E-6 | 模板余量测试自动碰撞比例 |
| int | marginHitCount | 0 | 模板余量测试自动碰撞数 |
| byte | acqLimitType | 0 | 眼图累积方式选择（0=wavefors,1=samples) |
| int | acqLimitNumber | 100 | 眼图累积数量 |
| string | opticalMaskName | c:\scope\masks\10GBE\_10\_3125\_MAY02.MSK | 光眼图模板名称 |
| string | elecMaskName |  | 电眼图模板名称 |
| string | opticalEyeSavePath |  | 光眼图保存地址 |
| string | elecEyeSavePath |  | 电眼图保存地址 |
| int | flexsetscaledelay | 1000 | 设置延时 |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] FLEX86100Struct  （属性参数的结构体数组） | 初始化示波器属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| public override | bool | ConfigFromFile |  | 按照配置文件对设备进行配置 |
| public override | double | ReadPower |  | 读取光功率，单位为uW |
| public override | double | ReadEr |  | 读取ER，单位为dB |
| public override | double[] | OpticalEyeTest | int syn = 0（缺省参数，0异步，1同步） | 测试光眼图，测量结果以数组形式返回，依次为AP,ER,OMA,Crossing,JitterPP,JitterRMS,RisTime,FallTime,EMM |
| public override | double[] | ElecEyeTest | int syn = 0（缺省参数，0异步，1同步） | 测试电眼图，测量结果以数组形式返回，依次为OMA,Crossing,JitterPP,JitterRMS,RisTime,FallTime,EMM |
| public override | bool | AutoScale | int syn = 0（缺省参数，0异步，1同步） | 自动眼图调整 |
| private | bool | WriteOpc | string command （指令字符串） | 带检查是否执行完成的IO输入指令(指令+OPC) |
| double readTimeDelay （单次读取等待时间(s)） |
| UInt32totalWaitTime（总共等待时间(s)） |
| private | string | AssembleChannelString | byte slot（槽位1~4） | 组装槽位和通道，输出标准通道字符串 |
| byte channel（通道1~4） |
| private | bool | isModuleNeedCalibrate | byte slot（模组所处槽位） | 检查模组是否需要校验 |
| private | bool | TriggerSetup | byte triggerSource (0=FrontPanel,1=FreeRun) | 触发信号设置 |
| byte triggerBandwidth (0=FilterdEdge,1=StandardEdge,2=Clock/Divided) |
| int syn = 0（缺省参数，0异步，1同步） |
| private | bool | CalibrateModel | byte slot | 校验模组 |
| private | bool | WavelengthSelect | string channel （通道，例如"1A"） | 波长选择 |
| byte waveLength（0=850,1=1310,2=1550） |
| int syn = 0（缺省参数，0异步，1同步） |
| private | bool | FileterSelect | string channel （通道，例如"1A"） | 滤波器速率选择 |
| double bitRate （速率选择，例如10.3125E9） |
| private | bool | FileterSwitch | string channel | 滤波器开关 |
| byte filterSwith（1=ON,0=OFF） |
| int syn = 0（缺省参数，0异步，1同步） |
| private | bool | SetAttenuation | string channel | 光功率补偿值设置 |
| double attenuation （补偿值单位dB） |
| byte attSwitch (1=ON,0=OFF) |
| int syn = 0（缺省参数，0异步，1同步） |
| private | bool | SetERCorrectFactor | string channel | 修正因子设置 |
| double erFactorPercent (修正量单位%) |
| byte erFactorSwitch |
| int syn = 0（缺省参数，0异步，1同步） |
| private | bool | PrecisionTimebaseSetup | byte slot | 精准时基单元的设置  86107A |
| double refClkFrequency (参考时钟例如10.3125E9) |
| byte Switch |
| byte synchMethod (0=OLIN,1=FAST) |
| int syn = 0（缺省参数，0异步，1同步） |
| private | bool | PTBSetup | double refClkFrequency (参考时钟例如10.3125E9) byte synchMethod (0=OLIN,1=FAST  int syn = 0（缺省参数，0异步，1同步）  byte Switch | 精准时基单元的设置 |
| private | bool | RapidEyeSetup | double bitRate | 快速眼图测试的设置 |
| int syn = 0（缺省参数，0异步，1同步） |
| byte rapidEyeSwitch |
| private | bool | EyeTuningDisplay | byte Switch  int syn = 0（缺省参数，0异步，1同步） | 眼图实时刷新显示方式开关 |
| private | bool | MessureThresholdSetup | byte threshold （0=80%50%20%,1=90%50%10%） | RiseTime FallTime测试的阈值和参考点设置 |
| int syn = 0（缺省参数，0异步，1同步） |
| byte reference （0=OneZero,1=VtopVbase） |
| private | bool | ChannelDisplaySwitch | string channel | 通道显示开关 |
| int syn = 0（缺省参数，0异步，1同步） |
| byte Switch |
| private | bool | EyeScaleOffset | string channel | 设置示波器显示的Scale和Offset |
| int syn = 0（缺省参数，0异步，1同步） |
| double scale （单位为uW/div或者uV/div） |
| double offset （单位为uW或者uV） |
| private | bool | LoadMask | string Channel | 装载模板 |
| int syn = 0（缺省参数，0异步，1同步） |
| string MaskName(模板路径） |
| private | bool | MaskONOFF | byte MaskON  int syn = 0（缺省参数，0异步，1同步） | 模板测试开关 |
| private | bool | MaskTestMarginSetup | byte marginOnOff | 模板余量测试方法设置 |
| byte marginAutoManul （1=自动，0=手动） |
| int syn = 0（缺省参数，0异步，1同步） |
| int manualMarginPercent （手动测试余量百分比） |
| byte autoMarginType （1=HitRatio,0=HitCount） |
| double hitRatio（例如1E-6） |
| int hitCount(例如0) |
| private | bool | AcquisitionControl | byte control （0=Run,1=Stop,2=Clear） | 示波器运行控制指令 |
| private | bool | CenterEye | double bitRate | 使眼图调整到中间 |
| private | bool | SetMode | byte Mode（0=EyeMode,1=OscMode,2=JitterMode）  int syn = 0（缺省参数，0异步，1同步） | 示波器模式选择 |
| private | bool | DisplayRiseFallTime | byte Rftime （0=RiseTime,1=FallTime） | 显示RisTime或者FallTime |
| private | bool | DisplayJitter | byte jitterFormat （0=JitterPP,1=JitterRMS） | 显示JitterPP或者JitterRMS |
| private | bool | DisplayCrossing |  | 显示Crossing |
| private | bool | DisplayOther | byte measureItem (0=BitRate,1=Amplitude,2=CrossingTime,3=SNR,4=OneLevel,5=ZeroLevel) | 显示其他指标 |
| private | bool | DisplayER | byte erUnit (0=Decibel,1=Percent,2=Ratio) | 显示ER |
| private | bool | DisplayPower | byte powerUnits (0=Watt,1=dBm) | 显示平均光功率 |
| private | bool | DisplayClearAlllist |  | 清除所有显示项 |
| private | bool | AcquisitionLimitTestSetup | byte limitCondition （0=Waveforms,1=Samples） | 眼图累积限制方式选择 |
| int syn = 0（缺省参数，0异步，1同步） |
| int limitNumber(累计数） |
| private | bool | AcquisitionLimitTestSwitch | byte Switch  int syn = 0（缺省参数，0异步，1同步） | 眼图累积限制开关 |
| private | bool | SaveEyeDiagram | string savePath（眼图保存完整路径） | 保存眼图到电脑指定文件 |
| byte picColor （图片颜色（0=彩色,1=黑白）） |
| byte bgColor （背景色（0=黑色，1=白色）） |
| private | double | ReadPower | byte powerUnit （0=uW,1=dBm） | 读取平均光功率 |
| private | double | ReadER | byte erUnits （0=Decibel,1=Percent,2=Ratio） | 读取ER |
| private | double | ReadCrossing |  | 读取Crossing |
| private | double | ReadAMPLitude |  | 读取眼图OMA，返回值为uW或者uV |
| private | double | ReadJitter | byte jitterformat (0=JitterPP,1=JitterRMS) | 读取JitterPP或者JitterRMS |
| private | double | ReadRiseFallTime | byte Rftime （0=RiseTime,1=FallTime） | 读取眼图上升沿或者下降沿时间 |
| private | double | ReadMaskMargin |  | 读取模板余量 |
| public override | bool | ChangeChannel | String Channel（模块通道）int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset  int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时收集 OFFSET |
| public override | bool | OpenOpticalChannel | bool Switch int syn = 0（缺省参数，0异步，1同步） | 打开光通道 |

## MP1800PPG

1. 类名：public class MP1800PPG: PPG
2. 父类：PPG
3. 实例化函数：public MP1800PPG ();
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 1 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | MP1800 | 仪器名称 |
| string | dataRate | 25.78125 | PPG速率（Gbps) |
| double | dataLevelGuardAmpMax | 1000 | 输出保护幅度最大值（单位mV） |
| double | dataLevelGuardOffsetMax | 1000 | 输出保护最大偏移量（单位mV） |
| double | dataLevelGuardOffsetMin | -1000 | 输出保护最小偏移量（单位mV） |
| byte | dataLevelGuardSwitch | 1 | 输出保护开关 |
| double | dataAmplitude | 500 | 输出单端幅度（单位mV） |
| double | dataCrossPoint | 50 | 输出数据信号交叉点 |
| string | configFilePath |  | 仪器中配置文件的地址 |
| byte | slot | 1 | PPG所处槽位 |
| byte | clockSource | 0 | PPG时钟源选择（0=内部时钟） |
| byte | auxOutputClkDiv | 8 | 辅助输出是时钟信号的几分频 |
| byte | TOTALCHANNELS | 4 | 设备通道数 |
| byte | prbsLength | 31 | PRBS码型长度 |
| byte | patternType | 0 | PPG码型选择（0=PRBS,1=Zero Subsitution,2=Data,3=Alternate,4=Mixed Data,5=Sequense） |
| byte | dataSwitch | 1 | 数据输出开关 |
| byte | dataTrackingSwitch | 1 | DATA /DATA跟踪开关 |
| byte | dataAcModeSwitch | 1 | 输出模式选择(0=DC，1=AC） |
| byte | dataLevelMode | 0 | 输出电平模式选择（0=VARiable,1=NECL,2=PCML,3=NCML,4=SCFL,5=LVPecl,6=LVDS200,7=LVDS400） |
| byte | clockSwitch | 1 | 时钟输出开关 |
| byte | outputSwitch | 1 | 总输出开关 |
| String | patternfile |  | 用户定义码型地址 @"C:\02" |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] MP1800PPGStruct | 初始化PPG的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| public | bool | ConfigureFromFile |  | 按照文件配置设备 |
| public override | bool | ConfigureDataRate | string datarate（比特率，单位Gbps) ，int syn = 0（缺省参数，0异步，1同步） | 设置比特率 |
| private | bool | ConfigureSlot | byte slot(PPG所处槽位），int syn = 0（缺省参数，0异步，1同步） | 设置槽位 |
| private | bool | ConfigureClockSource | byte clkSource（0=IntCLK，1=ExtCLK），int syn = 0（缺省参数，0异步，1同步） | 时钟源选择 |
| private | bool | ConfigureAuxOutputClkDiv | byte clkDiv（分频数），int syn = 0（缺省参数，0异步，1同步） | 配置辅助输出是时钟的几分频 |
| private | bool | ConfigureChannel | byte channel ，int syn = 0（缺省参数，0异步，1同步） | 配置通道 |
| private | bool | ConfigurePatternType | byte patternType（0=PRBS,1=ZSUBstitution,2=DATA,3=ALT,4=MIXData,5=MIXalt,6=SEQuence） ，int syn = 0（缺省参数，0异步，1同步） | 配置PPG数据码型 |
| private | bool | ConfigurePrbsLength | byte prbsLength，int syn = 0（缺省参数，0异步，1同步） | 配置PRBS码型长度 |
| private | bool | ConfigureDataSwitch | byte dataSwitch，int syn = 0（缺省参数，0异步，1同步） | 数据输出开关 |
| private | bool | ConfigureDataTracking | byte dataTrackingSwitch，int syn = 0（缺省参数，0异步，1同步） | DATA /DATA跟踪开关 |
| private | bool | ConfigureDataLevelGuardAmpMax | double ampMax，int syn = 0（缺省参数，0异步，1同步） | 配置输出保护单端幅度最大值 |
| private | bool | ConfigureDataLevelGuardOffset | double offsetMax | 配置输出保护单端最大偏移量 |
| int syn = 0（缺省参数，0异步，1同步） |  |
| double offsetMin | 配置输出保护单端最小偏移量 |
| private | bool | ConfigureDataLevelGuardSwitch | byte lvGuardSwitch，int syn = 0（缺省参数，0异步，1同步） | 输出保护开关 |
| private | bool | ConfigureDataAcModeSwitch | byte acSwitch（1=AC,0=DC)，int syn = 0（缺省参数，0异步，1同步） | 交流直流输出开关 |
| private | bool | ConfigureDataLevelMode | byte elecLevelMode（0=VARiable,1=NECL,2=PCML,3=NCML,4=SCFL,5=LVPecl,6=LVDS200,7=LVDS400），int syn = 0（缺省参数，0异步，1同步） | 输出电平类型选择 |
| private | bool | ConfigureDataAmplitude | double amplitude，int syn = 0（缺省参数，0异步，1同步） | 配置单端输出幅度 |
| private | bool | ConfigureDataCrossPoint | double crossPoint，int syn = 0（缺省参数，0异步，1同步） | 配置信号交点 |
| private | bool | ConfigureClockSwitch | byte clkSwitch，int syn = 0（缺省参数，0异步，1同步） | 时钟输出开关 |
| private | bool | ConfigureOutputSwitch | byte outSwitch，int syn = 0（缺省参数，0异步，1同步） | 输出总开关 |
| public override | bool | ChangeChannel | String Channel（模块通道）int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset  int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时收集 OFFSET |
| private | bool | configureusertype | string filename | patternType=2,选择DATA,user pattern，用户定义码型 |

## MP1800ED

1. 类名：public class MP1800ED: ErrorDetector
2. 父类：ErrorDetector
3. 实例化函数：public MP1800ED ();
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 1 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | MP1800 | 仪器名称 |
| byte | slot | 3 | ED所处槽位 |
| byte | totalChannels | 4 | ED总通道数 |
| byte | currentChannel | 1 | ED当前通道 |
| byte | dataInputInterface | 2 | 数据输入接口类型 |
| byte | prbsLength | 31 | PRBS码型长度 |
| byte | errorResultZoom | 1 | 0=ZoomIn(显示详细误码信息),1=ZoomOut(只显示误码率和误码数) |
| byte | edGatingMode | 1 | 累积模式（0=REPeat,1=SINGle,2=UNTimed） |
| byte | edGatingUnit | 0 | 累积单位（0=TIME,1=CLOCk,2=ERRor,3=BLOCk） |
| int | edGatingTime | 5 | 累积数量 |
| String | patternfile |  | 用户定义码型地址如：@"C:\02" |

1. 方法:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | EquipmentStruct[] EDlist（属性参数的结构体数组） | 初始化ED的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| public override | bool | AutoAlaign |  | 自动搜寻判决点 |
| public override | bool | GetErrorRate | int syn = 0（缺省参数，0异步，1同步） | 读取误码率 |
| private | bool | ConfigureSlot | byte slot(ED所处槽位），int syn = 0（缺省参数，0异步，1同步） | 设置槽位 |
| private | bool | ConfigureChannel | byte channel(ED所处通道），int syn = 0（缺省参数，0异步，1同步） | 配置通道 |
| private | bool | ConfigureDataInputInterface | byte dataInterface （0=SingleEnd,1=Diff 50ohm,2=Diff 100ohm），int syn = 0（缺省参数，0异步，1同步） | 配置数据输入接口类型 |
| private | bool | ConfigurePatternType | byte patternType （0=PRBS,1=Zero Subsitution,2=Data,3=Alternate,4=Mixed Data,5=Sequense），int syn = 0（缺省参数，0异步，1同步） | 配置ED码型 |
| private | bool | configureusertype | string filename | patternType=2,选择DATA,user pattern，用户定义码型 |
| private | bool | ConfigurePrbsLength | byte prbsLength，int syn = 0（缺省参数，0异步，1同步） | 配置PRBS码型长度 |
| private | bool | ConfigureErrorResultZoomSwitch | byte ZoomSwitch （1=ZoomOut,0=ZoomIn)，int syn = 0（缺省参数，0异步，1同步） | 配置误码结果显示方式 |
| private | bool | EdAutoSearchSetAll |  | 配置为自动搜寻所有通道 |
| private | bool | EdAutoSearchStart |  | 自动搜寻开始 |
| private | string | EdAutoSearchState |  | 自动搜寻状态读取 |
| private | bool | ConfigureEdGatingMode | byte gatingMode (0=REPeat,1=UNTimed,2=SINGle)，int syn = 0（缺省参数，0异步，1同步） | 设置误码扫描方式 |
| private | bool | ConfigureEdGatingUnit | byte gatingUnit （0=TIME,1=BLOCk,2=CLOCk,3=ERRor），int syn = 0（缺省参数，0异步，1同步） | 设置扫描累积单位 |
| private | bool | ConfigureEdGatingPeriod | int gatingTime （扫描累积数量），int syn = 0（缺省参数，0异步，1同步） | 设置扫描累积数量 |
| private | bool | EdGatingStart |  | 扫描开始 |
| private | double | QureyEdErrorRatio |  | 查询误码率 |
| public override | bool | ChangeChannel | String Channel（模块通道）int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset  int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时收集 OFFSET |

## N490XPPG

1. 类名：public class N490XPPG: PPG
2. 父类：PPG
3. 实例化函数：
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 1 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | N490XPPG | 仪器名称 |
| string | TriggerDRatio | 4 | 触发信号是信号速率的几分频 |
| string | TriggerMode | DCL | 触发信号模式（DCL or PATT） |
| string | ClockLowVoltage | 0 | 时钟信号低电压（单位V） |
| string | ClockHigVoltage | 0.5 | 时钟信号高电压（单位V） |
| string | DataLowVoltage | -0.5 | 数据信号低电压（单位V） |
| string | DataHigVoltage | 0 | 数据信号高电压（单位V） |
| string | BertDataRate | 10.3125E9 | PPG速率（bps) |
| string | PRBS | 31 | PRBS码型长度 |

1. 方法：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] N490XPPGStruct  （属性参数的结构体数组） | 初始化PPG的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| private | bool | ConfigureDataRate | int syn = 0（缺省参数，0异步，1同步） | 配置速率 |
| private | bool | ConfigurePRBS | int syn = 0（缺省参数，0异步，1同步） | 配置PRBS |
| private | bool | ConfigureDataVoltage | string DataHigVoltage, string DataLowVoltage, int syn = 0（缺省参数，0异步，1同步） | 配置数据电平电压值 |
| private | bool | ConfigureClockVoltage | string ClockHigVoltage, string ClockLowVoltage  int syn = 0（缺省参数，0异步，1同步） | 配置时钟电平电压值 |
| private | bool | ConfigureTriggerMode | string PGTriggerMode  int syn = 0（缺省参数，0异步，1同步） | 配置触发信号的模式 |
| private | bool | ConfigureTriggerDRatio | string TriggerDRatio  int syn = 0（缺省参数，0异步，1同步） | 配置触发信号的分频数 |
| public override | bool | ChangeChannel | String Channel（模块通道），int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset，int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时收集 OFFSET |

## N490XED

1. 类名：public class N490XED: ErrorDetector
2. 父类：ErrorDetector
3. 实例化函数：
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 1 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | N490XED | 仪器名称 |
| bool | CDRSwitch | TRUE | CDR开关 |
| string | CDRFreq | 10.3125E9 | CDR时钟选择 |
| string | PRBS | 31 | PRBS码型长度 |

1. 方法:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] N490XEDStruct  （属性参数的结构体数组） | 初始化ED的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| private | bool | ConfigurePRBS | int syn = 0（缺省参数，0异步，1同步）string PRBS | 配置PRBS |
| public override | bool | AutoAlaign | bool becenter（是否数据中心） | 自动搜寻判决点 |
| private | bool | ConfigureCDRONOFF | bool Switch，int syn = 0（缺省参数，0异步，1同步） | 配置CDR开关 |
| private | bool | ConfigureCDRFreq | string CDRFreq  int syn = 0（缺省参数，0异步，1同步） | 配置CDR速率 |
| public override | double | GetErrorRate | int syn = 0（缺省参数，0异步，1同步） | 读取误码率 |
| public override | bool | ChangeChannel | String Channel（模块通道），int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset，int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时收集 OFFSET |

## TPO4300

1. 类名：public class TPO4300: Thermocontroller
2. 父类：Thermocontroller
3. 实例化函数：
4. 字段：
5. 属性列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | Initialize用到的属性 | 默认值 | 描述 |
| string | Addr | 23 | 地址 |
| string | IOType | GPIB | 接口类型 |
| bool | Reset | FALSE | 是否需要复位 |
| string | Name | TPO4300 | 仪器名称 |
| string | FLSE | 15 | 气流速度 |
| string | ULIM | 90 | 气流温度最大值 |
| string | LLIM | -20 | 气流温度最小值 |
| string | Sensor | 1 | 探头类型（0=No Sensor,1=T,2=k,3=rtd,4=diode） |

1. 方法:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 作用范围 | 类型 | 名称 | 形参 | 描述 |
| public override | bool | Initialize | TestModeEquipmentParameters[] TPO4300Struct  （属性参数的结构体数组） | 初始化热流仪的属性参数 |
| public override | bool | Connect |  | 连接设备 |
| public | bool | ReSet |  | 复位设备 |
| public override | bool | Configure | int syn = 0（缺省参数，0异步，1同步） | 配置设备 |
| public override | bool | DUTControlModeOnOFF | byte Switch（1=DUT控制，0=气流控制），int syn = 0（缺省参数，0异步，1同步） | 温度控制模式选择 |
| public override | string | ReadCurrentTemp |  | 读取当前温度 |
| public override | bool | AirFlowSetting |  | 设置气流速度 |
| public override | bool | AirTempsUpperlimit | int syn = 0（缺省参数，0异步，1同步） | 设置气流温度上限 |
| public override | bool | AirTempslowerlimit | int syn = 0（缺省参数，0异步，1同步） | 设置气流温度下限 |
| public override | bool | SensorType | int syn = 0（缺省参数，0异步，1同步） | 设置探头类型 |
| public override | string | ReadSetpoint |  | 读取温度点 |
| public override | string | ReadSetpointTemp |  | 读取温度值 |
| public override | bool | SetPositionUPDown | string position（1=down,0=up)，int syn = 0（缺省参数，0异步，1同步） | 设置罩子外置 |
| public override | bool | SetPoint | string Point（0=HOT,1=AMB,2=COLD），int syn = 0（缺省参数，0异步，1同步） | 设置温度区间 |
| public override | bool | SetPointTemp | double Temp（单位C），int syn = 0（缺省参数，0异步，1同步） | 设置目标温度 |
| public override | bool | ChangeChannel | String Channel（模块通道），int syn = 0（缺省参数，0异步，1同步） | 切换模块通道 |
| public override | bool | configoffset | string channel, string offset，int syn = 0（缺省参数，0异步，1同步） | 切换模块通道时收集 OFFSET |