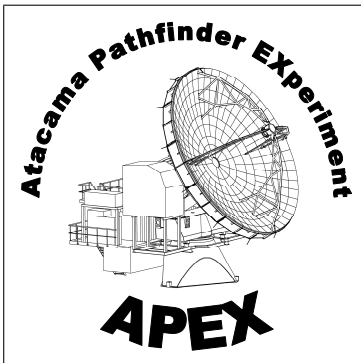


Atacama Pathfinder EXperiment

Interface Control Document

APEX-MPI-ICD-0004
Revision: 4.0
Release: February 26, 2019
Category: 4
Author: D. Muders

APEX Instruments Generic CORBA IDL Interfaces

D. Muders

Keywords: APEX Instruments CORBA IDL Interfaces, IDL

Author Signature: D. Muders	Date: February 26, 2019
Approved by: L.-Å. Nyman Institute: APEX	Signature: L.-Å. Nyman Date: February 26, 2019
Released by: L.-Å. Nyman Institute: APEX	Signature: L.-Å. Nyman Date: February 26, 2019

Change Record

Revision	Date	Author	Section/ Page affected	Remarks
1.0-1.7	2002 to Nov. 2006	D. Muders	All	APECS Development
1.8	21 January 2007	D. Muders	HFE IDL	Added dewar rotator state
1.9	3 April 2007	D. Muders	HFE, Cal units, SZ	Interface iterations after CHAMP+ and SZ installations at APEX
2.0-3.2	10 October 2016	D. Muders	HFE, CFE, Cal units, ACU, VLBI	Interface iterations up to APECS 3.2
3.3	13 February 2017	D. Muders	Third C cabin optics	Combined Artemis and holography optics
3.4	13 July 2018	D. Muders	ACU, WIU, wobbler, SBE, optical frame grabber	APEX-II; SBE sub-bands; new optical setup
4.0	26 February 2019	D. Muders	ACU, IF chain	Shutter ICD update; IF chain parameters

Contents

Chapter 1

Introduction

The interfaces described in this document are used at the APEX telescope. They are generic and can be applied to any instrument of a given class. They provide a set of building blocks out of which we can construct any instrument such as a receiver. All instruments will have the *same* high-level interface. This makes the setup for the high-level observing software very simple because ones just adds a new name but one does not have to worry about adding new features at that level.
The actual detailed setup of e.g. a given receiver differs, of course, from one to another. This is represented by a naming hierarchy of objects for the components in the receiver. This way we can find all relevant parts belonging to one instrument and we can see its structure by looking at the naming hierarchy. This hierarchy is stored in the ACS Configuration Data Base (CDB) which consists of directories defining the name space in which the generic interface components are started and XML files describing the IDLs of the individual sub-devices like local oscillators, backend bands etc.
One can see that we need only a few IDLs to describe a complex device like a heterodyne front-end HFE : apexHFE, apexHFE Mixer, apexHFE ColdAmp, apexHFE_LO, apexHFE_Mixer, apexHFE_Multi and apexCalUnit. We reuse the same IDL and even the same component library many times. A new device is merely a new CDB entry. We do not even have to compile any new code unless one adds something that is not covered by the existing IDL's.

Chapter 2

Compound Index

2.1 Class List

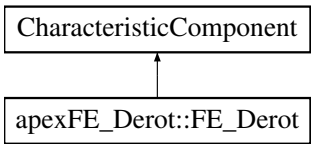
Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

Frontends

3.1 apexFE_Derot::FE_Derot Interface Reference

`import"apexFE_Derot.idl";`
Inheritance diagram for apexFE_Derot::FE_Derot:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string dewarAngle_DESCRIPTION = ”Dewar Angle”
- const string **dewarAngle_UNITS** = ”deg”
- const string **dewarAngle_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **dewarAngle**
- const string cmdDewarAngle_DESCRIPTION = ”Commanded Dewar Angle”
- const string **cmdDewarAngle_UNITS** = ”deg”
- readonly attribute ACS::RWdouble **cmdDewarAngle**
- const string derotatorState_DESCRIPTION = ”Derotator state”
- const string **derotatorState_UNITS** = ””
- const string **derotatorState_SAMPLEINTERVAL** = ”60”
- readonly attribute ROderotatorStatus **derotatorState**
- const string dewarTrack_DESCRIPTION = ”Dewar Track Mode”
- const string **dewarTrack_UNITS** = ””
- const string **dewarTrack_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **dewarTrack**
- const string cmdDewarTrack_DESCRIPTION = ”Commanded Dewar Track Mode”

- const string **cmdDewarTrack_UNITS** = ””
- readonly attribute RWlogical **cmdDewarTrack**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**

3.1.1 Detailed Description

This interface belongs to the abstract Front End device. It will be the standard interface for APEX Front End Units.

3.1.2 Member Function Documentation

void apexFE_Derot::FE_Derot::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexFE_Derot::FE_Derot::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexFE_Derot::FE_Derot::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.1.3 Member Data Documentation

const string apexFE_Derot::FE_Derot::cmdDewarAngle_DESCRIPTION = ”Commanded Dewar Angle”

Commanded dewar rotation angle (degrees)

const string apexFE_Derot::FE_Derot::cmdDewarTrack_DESCRIPTION = ”Commanded Dewar Track Mode”

Commanded dewar tracking (OFF, ON)

const string apexFE_Derot::FE_Derot::derotatorState_DESCRIPTION = ”Derotator state”

Derotator state (ONPOSITION, MOVING, UNDEFINED)

const string apexFE_Derot::FE_Derot::dewarAngle_DESCRIPTION = ”Dewar Angle”

Dewar rotation angle (degrees)

const string apexFE_Derot::FE_Derot::dewarTrack_DESCRIPTION = ”Dewar Track Mode”

Dewar tracking (OFF, ON)

const string apexFE_Derot::FE_Derot::state_DESCRIPTION = ”State”

Actual State of the Frontend Derotator

const string apexFE_Derot::FE_Derot::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

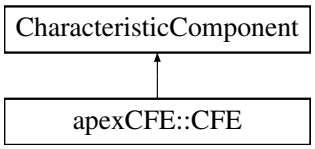
- apexFE_Derot.idl

3.2 Continuum Frontends

3.3 apexCFE::CFE Interface Reference

```
import"apexCFE.idl";
```

Inheritance diagram for apexCFE::CFE:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void configure (in string mode, in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string elements_DESCRIPTION = ”Pixel Usage”
- const string **elements_UNITS** = ””
- readonly attribute ACS::ROlongSeq **elements**
- const string cmdElements_DESCRIPTION = ”Commanded Pixel Usage”
- const string **cmdElements_UNITS** = ””
- readonly attribute ACS::RWlongSeq **cmdElements**
- const string configModified_DESCRIPTION = ”Configuration Modified”
- const string **configModified_UNITS** = ””
- const string **configModified_SAMPLEINTERVAL** = ”60”
- readonly attribute ROyesNo **configModified**
- const string dewarTemp_DESCRIPTION = ”Dewar Temperature”
- const string **dewarTemp_UNITS** = ”K”
- const string **dewarTemp_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **dewarTemp**
- const string dewarVacuum_DESCRIPTION = ”Dewar Vacuum”
- const string **dewarVacuum_UNITS** = ”mbar”
- const string **dewarVacuum_FORMAT** = ”%.7f”
- const string **dewarVacuum_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **dewarVacuum**
- const string dewarPump_DESCRIPTION = ”Dewar Pump State”
- const string **dewarPump_UNITS** = ””
- const string **dewarPump_FORMAT** = ”%.7f”

- const string **dewarPump_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **dewarPump**
- const string cmdDewarPump_DESCRIPTION = ”Commanded Dewar Pump State”
- const string **cmdDewarPump_UNITS** = ””
- readonly attribute RWlogical **cmdDewarPump**
- const string dewarPumpSpeed_DESCRIPTION = ”Dewar Pump Speed”
- const string **dewarPumpSpeed_UNITS** = ”rpm”
- const string **dewarPumpSpeed_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **dewarPumpSpeed**
- const string cmdDewarPumpSpeed_DESCRIPTION = ”Commanded Dewar Pump Speed”
- const string **cmdDewarPumpSpeed_UNITS** = ”rpm”
- readonly attribute ACS::RWlong **cmdDewarPumpSpeed**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**

3.3.1 Detailed Description

This interface belongs to the abstract Continuum Front End device. It will be the standard interface for APEX Continuum Front End Units.

3.3.2 Member Function Documentation

void apexCFE::CFE::configure (in string *mode,* in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Set up the continuum frontend according to the high-level properties that are currently defined. The embedded system must keep track of those values.
Parameters

mode	configuration mode (optional, can be left blank)
cb	callback when completed
desc	description

void apexCFE::CFE::off (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexCFE::CFE::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexCFE::CFE::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.3.3 Member Data Documentation

const string apexCFE::CFE::cmdDewarPump_DESCRIPTION = ”Commanded Dewar Pump State”

Commanded dewar pump state

const string apexCFE::CFE::cmdDewarPumpSpeed_DESCRIPTION = ”Commanded Dewar Pump Speed”

Commanded dewar pump speed (rpm)

const string apexCFE::CFE::cmdElements_DESCRIPTION = ”Commanded Pixel Usage”

Commanded List of elements (0: not used, 1: used)

const string apexCFE::CFE::configModified_DESCRIPTION = ”Configuration Modified”

Flag to indicate that the configuration was changed by the receiver control system after a configure command. (YES / NO)

const string apexCFE::CFE::dewarPump_DESCRIPTION = ”Dewar Pump State”

Dewar pump state

const string apexCFE::CFE::dewarPumpSpeed_DESCRIPTION = ”Dewar Pump Speed”

Dewar pump speed (rpm)

const string apexCFE::CFE::dewarTemp_DESCRIPTION = ”Dewar Temperature”

Dewar temperature (K)

const string apexCFE::CFE::dewarVacuum_DESCRIPTION = ”Dewar Vacuum”

Dewar vacuum (mbar)

const string apexCFE::CFE::elements_DESCRIPTION = ”Pixel Usage”

List of pixels to be used (0: not used, 1: used)

const string apexCFE::CFE::state_DESCRIPTION = ”State”

Actual State of the continuum frontend

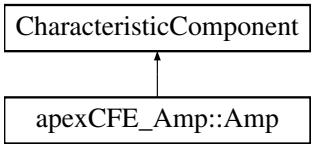
const string apexCFE::CFE::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

- apexCFE.idl

3.4 apexCFE_Amp::Amp Interface Reference

`import"apexCFE_Amp.idl";`
Inheritance diagram for apexCFE_Amp::Amp:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long removeOffsets () raises (ACSErr::ACSException)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = "State"
- const string **state_UNITS** = ""
- const string **state_ALARMON** = "4"
- const string **state_SAMPLEINTERVAL** = "60"
- readonly attribute ROstatus **state**
- const string ACDC_DESCRIPTION = "ACDC Mode"
- const string **ACDC_UNITS** = ""
- readonly attribute ACS::ROlongSeq **ACDC**
- const string cmdACDC_DESCRIPTION = "Commanded ACDC Mode"
- const string **cmdACDC_UNITS** = ""
- readonly attribute ACS::RWlongSeq **cmdACDC**
- const string gain_DESCRIPTION = "Gain Multiplier"
- const string **gain_UNITS** = ""
- const string **gain_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROlong **gain**
- const string cmdGain_DESCRIPTION = "Commanded Gain Multiplier"
- const string **cmdGain_UNITS** = ""
- readonly attribute ACS::RWlong **cmdGain**
- const string offsets_DESCRIPTION = "Amplifier Offsets"
- const string **offsets_UNITS** = "counts"
- readonly attribute ACS::ROdoubleSeq **offsets**
- const string version_DESCRIPTION = "Version"
- const string **version_UNITS** = ""
- readonly attribute ACS::ROstring **version**

3.4.1 Detailed Description

This interface belongs to the abstract Continuum Front End device. It will be the standard interface for APEX Continuum Front End Units.

3.4.2 Member Function Documentation

void apexCFE_Amp::Amp::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.

Parameters

cb	callback when completed
desc	description

void apexCFE_Amp::Amp::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexCFE_Amp::Amp::removeOffsets ( ) raises ACSErr::ACSException)

Remove amplifier offsets and store them in the offsets property for later use by the FitsWriter

long apexCFE_Amp::Amp::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.4.3 Member Data Documentation

const string apexCFE_Amp::Amp::ACDC_DESCRIPTION = "ACDC Mode"

AC/DC mode (0: AC, 1: DC)

const string apexCFE_Amp::Amp::cmdACDC_DESCRIPTION = "Commanded ACDC Mode"

Commanded AC/DC mode (0: AC, 1: DC)

const string apexCFE_Amp::Amp::cmdGain_DESCRIPTION = "Commanded Gain Multiplier"

Commanded gain multiplier (typical values: 1,2,4,8)

const string apexCFE_Amp::Amp::gain_DESCRIPTION = "Gain Multiplier"

Gain multiplier (typical values: 1,2,4,8)

const string apexCFE_Amp::Amp::offsets_DESCRIPTION = "Amplifier Offsets"

Apmlifier offsets

const string apexCFE_Amp::Amp::state_DESCRIPTION = "State"

Actual State of the continuum frontend amplifier

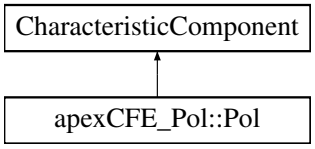
const string apexCFE_Amp::Amp::version_DESCRIPTION = "Version"

Software version
The documentation for this interface was generated from the following file:

- apexCFE_Amp.idl

3.5 apexCFE_Pol::Pol Interface Reference

`import"apexCFE_Pol.idl";`
Inheritance diagram for apexCFE_Pol::Pol:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string filterWheel_DESCRIPTION = ”Filter Wheel Configuration”
- const string **filterWheel_SAMPLEINTERVAL** = ”60”
- const string **filterWheel_UNITS** = ””
- readonly attribute ROfilterPos **filterWheel**
- const string cmdFilterWheel_DESCRIPTION = ”Commanded Filter Wheel Configuration”
- const string **cmdFilterWheel_UNITS** = ””
- readonly attribute RWfilterPos **cmdFilterWheel**
- const string frequency_DESCRIPTION = ”Rotation Frequency”
- const string **frequency_UNITS** = ”Hz”
- const string **frequency_FORMAT** = ”%.3f”
- const string **frequency_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::RODouble **frequency**
- const string cmdFrequency_DESCRIPTION = ”Commanded Rotation Frequency”
- const string **cmdFrequency_UNITS** = ”Hz”
- const string **cmdFrequency_FORMAT** = ”%.3f”
- readonly attribute ACS::RWdouble **cmdFrequency**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**

3.5.1 Detailed Description

This interface belongs to the abstract Continuum Frontend device. It will be the standard interface for APEX Continuum Frontend Polarimeter Units.

3.5.2 Member Function Documentation

void apexCFE_Pol::Pol::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.

Parameters

cb	callback when completed
desc	description

void apexCFE_Pol::Pol::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexCFE_Pol::Pol::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.5.3 Member Data Documentation

const string apexCFE_Pol::Pol::cmdFilterWheel_DESCRIPTION = "Commanded Filter Wheel Configuration"

Commanded Filter Wheel Configuration (HORIZONTAL, VERTICAL)

const string apexCFE_Pol::Pol::cmdFrequency_DESCRIPTION = "Commanded Rotation Frequency"

Commanded Polarimeter HWP Rotation Frequency

const string apexCFE_Pol::Pol::filterWheel_DESCRIPTION = "Filter Wheel Configuration"

Filter Wheel Configuration (HORIZONTAL, VERTICAL)

const string apexCFE_Pol::Pol::frequency_DESCRIPTION = "Rotation Frequency"

Polarimeter Half-Wave-Plate (HWP) Rotation Frequency

const string apexCFE_Pol::Pol::state_DESCRIPTION = "State"

Actual State of the continuum frontend polarimeter unit

const string apexCFE_Pol::Pol::version_DESCRIPTION = "Version"

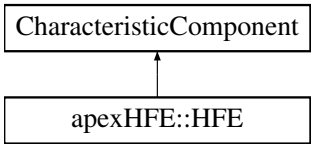
Software version
The documentation for this interface was generated from the following file:

- apexCFE_Pol.idl

3.6 Heterodyne Frontends

3.7 apexHFE::HFE Interface Reference

`import"apexHFE.idl";`
Inheritance diagram for apexHFE::HFE:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void tune (in string mode, in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void magFieldScan (in double minBField, in double maxBField, in long numPoints, in long feedNumber, in ACS::CBdoubleSeq cb, in ACS::CBDescIn desc)
- void IVScan (in double minBias, in double maxBias, in long numPoints, in long feedNumber, in ACS::CBdoubleSeq cb, in ACS::CBDescIn desc)
- void TPScan (in double minBias, in double maxBias, in long numPoints, in long feedNumber, in ACS::CBdoubleSeq cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string skyFrequency_DESCRIPTION = ”Sky Frequency”
- const string **skyFrequency_UNITS** = ”GHz”
- const string **skyFrequency_FORMAT** = ”%.9f”
- const string **skyFrequency_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **skyFrequency**
- const string cmdSkyFrequency_DESCRIPTION = ”Commanded Sky Frequency”
- const string **cmdSkyFrequency_UNITS** = ”GHz”
- readonly attribute ACS::RWdouble **cmdSkyFrequency**
- const string sideBand_DESCRIPTION = ”Sideband”
- const string **sideBand_UNITS** = ””
- const string **sideBand_SAMPLEINTERVAL** = ”60”
- readonly attribute ROsideBand **sideBand**
- const string cmdSideBand_DESCRIPTION = ”Commanded Sideband”
- const string **cmdSideBand_UNITS** = ””
- readonly attribute RWsideBand **cmdSideBand**
- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string polarisations_DESCRIPTION = ”Polarisation Usage”
- const string **polarisations_UNITS** = ””
- readonly attribute ACS::ROlongSeq **polarisations**
- const string cmdPolarisations_DESCRIPTION = ”Commanded Polarisation Usage”
- const string **cmdPolarisations_UNITS** = ””
- readonly attribute ACS::RWlongSeq **cmdPolarisations**
- const string pixels_DESCRIPTION = ”Pixel Usage”

- const string **pixels_UNITS** = ""
- readonly attribute ACS::ROlongSeq **pixels**
- const string cmdPixels_DESCRIPTION = "Commanded Pixel Usage"
- const string **cmdPixels_UNITS** = ""
- readonly attribute ACS::RWlongSeq **cmdPixels**
- const string freqSwitchMode_DESCRIPTION = "Frequency Switch Mode"
- const string **freqSwitchMode_UNITS** = ""
- const string **freqSwitchMode_SAMPLEINTERVAL** = "60"
- readonly attribute ROlogical **freqSwitchMode**
- const string cmdFreqSwitchMode_DESCRIPTION = "Commanded Frequency Switch Mode"
- const string **cmdFreqSwitchMode_UNITS** = ""
- readonly attribute RWlogical **cmdFreqSwitchMode**
- const string freqSwitchThrows_DESCRIPTION = "Frequency Switch Throws"
- const string **freqSwitchThrows_UNITS** = "MHz"
- const string **freqSwitchThrows_FORMAT** = "%.6f"
- readonly attribute ACS::ROdoubleSeq **freqSwitchThrows**
- const string cmdFreqSwitchThrows_DESCRIPTION = "Commanded Frequency Switch Throws"
- const string **cmdFreqSwitchThrows_UNITS** = "MHz"
- readonly attribute ACS::RWdoubleSeq **cmdFreqSwitchThrows**
- const string tuningModified_DESCRIPTION = "Tuning Modified"
- const string **tuningModified_UNITS** = ""
- const string **tuningModified_SAMPLEINTERVAL** = "60"
- readonly attribute ROyesNo **tuningModified**
- const string dewarTemp_DESCRIPTION = "Dewar Temperature"
- const string **dewarTemp_UNITS** = "K"
- const string **dewarTemp_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROdouble **dewarTemp**
- const string dewarVacuum_DESCRIPTION = "Dewar Vacuum"
- const string **dewarVacuum_UNITS** = "mbar"
- const string **dewarVacuum_FORMAT** = "%.7f"
- const string **dewarVacuum_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROdouble **dewarVacuum**
- const string version_DESCRIPTION = "Version"
- const string **version_UNITS** = ""
- readonly attribute ACS::ROstring **version**

3.7.1 Detailed Description

This interface belongs to the abstract Heterodyne Front End device. It will be the standard interface for APEX Heterodyne Front End Units.

3.7.2 Member Function Documentation

void apexHFE::HFE::IVScan (in double *minBias*, in double *maxBias*, in long *numPoints*, in long *feedNumber*, in ACS::CBdoubleSeq *cb*, in ACS::CBDescIn *desc*)

Measure IV curve.
Parameters

minBias	minimum bias voltage (mV)
maxBias	maximum bias voltage (mV)

numPoints	number of points to be sampled (max. 100)
feedNumber	number of the feed to be examined
cb	callback when completed
desc	description

void apexHFE::HFE::magFieldScan (in double *minBField,* in double *maxBField,* in long *numPoints,* in long *feedNumber,* in ACS::CBdoubleSeq *cb,* in ACS::CBDescIn *desc*)

Magnetic field scan.
Parameters

minBField	minimum magnetic field strength (% (-100 to 100))
maxBField	maximum magnetic field strength (% (-100 to 100))
numPoints	number of points to be sampled (max. 100)
feedNumber	number of the feed to be examined
cb	callback when completed
desc	description

void apexHFE::HFE::off (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexHFE::HFE::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexHFE::HFE::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

void apexHFE::HFE::TPScan (in double *minBias,* in double *maxBias,* in long *numPoints,* in long *feedNumber,* in ACS::CBdoubleSeq *cb,* in ACS::CBDescIn *desc*)

Measure total power / conversion curve
Parameters

minBias	minimum bias voltage (mV)
maxBias	maximum bias voltage (mV)
numPoints	number of points to be sampled (max. 100)
feedNumber	number of the feed to be examined
cb	callback when completed
desc	description

void apexHFE::HFE::tune (in string *mode,* in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Tune the receiver according to the high-level properties that are currently defined. The embedded system must keep track of those values.

Parameters

mode	tuning mode (optional, can be left blank)
cb	callback when completed
desc	description

3.7.3 Member Data Documentation

const string apexHFE::HFE::cmdFreqSwitchMode_DESCRIPTION = "Commanded Frequency Switch Mode"

Commanded frequency switching mode (OFF, ON)

const string apexHFE::HFE::cmdFreqSwitchThrows_DESCRIPTION = "Commanded Frequency Switch Throws"

Commanded frequency switching throws (MHz)

const string apexHFE::HFE::cmdPixels_DESCRIPTION = "Commanded Pixel Usage"

Commanded list of elements to be used (0: Not used, 1: Used)

const string apexHFE::HFE::cmdPolarisations_DESCRIPTION = "Commanded Polarisation Usage"

Commanded list of polarisations (0: Horizontal, 1: Vertical)

const string apexHFE::HFE::cmdSideBand_DESCRIPTION = "Commanded Sideband"

Commanded sideband (LSB, USB)

const string apexHFE::HFE::cmdSkyFrequency_DESCRIPTION = "Commanded Sky Frequency"

Commanded sky frequency (GHz)

const string apexHFE::HFE::dewarTemp_DESCRIPTION = "Dewar Temperature"

Dewar temperature (K)

const string apexHFE::HFE::dewarVacuum_DESCRIPTION = "Dewar Vacuum"

Dewar vacuum (mbar)

const string apexHFE::HFE::freqSwitchMode_DESCRIPTION = "Frequency Switch Mode"

Frequency switching mode selection (OFF, ON)

const string apexHFE::HFE::freqSwitchThrows_DESCRIPTION = "Frequency Switch Throws"

Frequency switching throws (MHz)

const string apexHFE::HFE::pixels_DESCRIPTION = "Pixel Usage"

List of pixels to be used (0: Not used, 1: Used)

const string apexHFE::HFE::polarisations_DESCRIPTION = "Polarisation Usage"

List of polarisations (0: Horizontal, 1: Vertical)

const string apexHFE::HFE::sideBand_DESCRIPTION = "Sideband"

Sideband (LSB, USB)

const string apexHFE::HFE::skyFrequency_DESCRIPTION = ”Sky Frequency”

Sky frequency (GHz)

const string apexHFE::HFE::state_DESCRIPTION = ”State”

Actual State of the Heterodyne Frontend

const string apexHFE::HFE::tuningModified_DESCRIPTION = ”Tuning Modified”

Flag to indicate that the tuning was changed by the receiver control system after a tune command. (YES / NO)

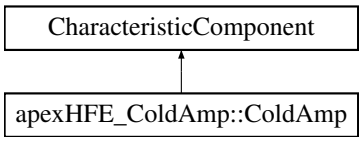
const string apexHFE::HFE::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

- apexHFE.idl

3.8 apexHFE_ColdAmp::ColdAmp Interface Reference

`import"apexHFE_ColdAmp.idl";`
Inheritance diagram for apexHFE_ColdAmp::ColdAmp:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**
- const string biasMode_DESCRIPTION = ”Bias Mode”
- const string **biasMode_UNITS** = ””
- const string **biasMode_SAMPLEINTERVAL** = ”60”
- readonly attribute RObiasMode **biasMode**
- const string cmsBiasMode_DESCRIPTION = ”Commanded Bias Mode”
- const string **cmsBiasMode_UNITS** = ””
- readonly attribute RWbiasMode **cmsBiasMode**
- const string bias1_DESCRIPTION = ”Bias Voltage #1”
- const string **bias1_UNITS** = ”mV”
- const string **bias1_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **bias1**
- const string cmdBias1_DESCRIPTION = ”Commanded Bias Voltage #1”
- const string **cmdBias1_UNITS** = ”mV”
- readonly attribute ACS::RWdouble **cmdBias1**
- const string current1_DESCRIPTION = ”Current #1”
- const string **current1_UNITS** = ”muA”
- const string **current1_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **current1**
- const string cmdCurrent1_DESCRIPTION = ”Commanded Current #1”
- const string **cmdCurrent1_UNITS** = ”muA”
- readonly attribute ACS::RWdouble **cmdCurrent1**
- const string bias2_DESCRIPTION = ”Bias Voltage #2”
- const string **bias2_UNITS** = ”mV”
- const string **bias2_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **bias2**
- const string cmdBias2_DESCRIPTION = ”Commanded Bias Voltage #2”
- const string **cmdBias2_UNITS** = ”mV”
- const string **cmdBias2_SAMPLEINTERVAL** = ”60”

- readonly attribute ACS::RWdouble **cmdBias2**
- const string current2_DESCRIPTION = ”Current #2”
- const string **current2_UNITS** = ”muA”
- const string **current2_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **current2**
- const string cmdCurrent2_DESCRIPTION = ”Commanded Current #2”
- const string **cmdCurrent2_UNITS** = ”muA”
- readonly attribute ACS::RWdouble **cmdCurrent2**
- const string bias3_DESCRIPTION = ”Bias Voltage #3”
- const string **bias3_UNITS** = ”mV”
- const string **bias3_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **bias3**
- const string cmdBias3_DESCRIPTION = ”Commanded Bias Voltage #3”
- const string **cmdBias3_UNITS** = ”mV”
- readonly attribute ACS::RWdouble **cmdBias3**
- const string current3_DESCRIPTION = ”Current #3”
- const string **current3_UNITS** = ”muA”
- const string **current3_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **current3**
- const string cmdCurrent3_DESCRIPTION = ”Commanded Current #3”
- const string **cmdCurrent3_UNITS** = ”muA”
- readonly attribute ACS::RWdouble **cmdCurrent3**
- const string IFAtten_DESCRIPTION = ”IF Attenuation”
- const string **IFAtten_UNITS** = ”dB”
- const string **IFAtten_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **IFAtten**
- const string cmdIFAtten_DESCRIPTION = ”Commanded IF Attenuation”
- const string **cmdIFAtten_UNITS** = ”dB”
- readonly attribute ACS::RWdouble **cmdIFAtten**
- const string IFLevel_DESCRIPTION = ”IF Output Level”
- const string **IFLevel_UNITS** = ”dBm”
- const string **IFLevel_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **IFLevel**
- const string cmdIFLevel_DESCRIPTION = ”Commanded IF Output Level”
- const string **cmdIFLevel_UNITS** = ”dBm”
- readonly attribute ACS::RWdouble **cmdIFLevel**

3.8.1 Detailed Description

This interface belongs to the abstract Heterodyne Front End device. It will be the standard interface for APEX Heterodyne Front End Units.

3.8.2 Member Function Documentation

void apexHFE_ColdAmp::ColdAmp::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexHFE_ColdAmp::ColdAmp::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.

Parameters

cb	callback when completed
desc	description

long apexHFE_ColdAmp::ColdAmp::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.8.3 Member Data Documentation

const string apexHFE_ColdAmp::ColdAmp::bias1_DESCRIPTION = "Bias Voltage #1"

Cold amplifier bias voltage 1 (mV)

const string apexHFE_ColdAmp::ColdAmp::bias2_DESCRIPTION = "Bias Voltage #2"

Cold amplifier bias voltage 2 (mV)

const string apexHFE_ColdAmp::ColdAmp::bias3_DESCRIPTION = "Bias Voltage #3"

Cold amplifier bias voltage 3 (mV)

const string apexHFE_ColdAmp::ColdAmp::biasMode_DESCRIPTION = "Bias Mode"

Cold amplifier bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_ColdAmp::ColdAmp::cmdBias1_DESCRIPTION = "Commanded Bias Voltage #1"

Commanded cold amplifier bias voltage 1 (mV)

const string apexHFE_ColdAmp::ColdAmp::cmdBias2_DESCRIPTION = "Commanded Bias Voltage #2"

Commanded cold amplifier bias voltage 2 (mV)

const string apexHFE_ColdAmp::ColdAmp::cmdBias3_DESCRIPTION = "Commanded Bias Voltage #3"

Commanded cold amplifier bias voltage 3 (mV)

const string apexHFE_ColdAmp::ColdAmp::cmdCurrent1_DESCRIPTION = "Commanded Current #1"

Commanded cold amplifier current 1 (micro A)

const string apexHFE_ColdAmp::ColdAmp::cmdCurrent2_DESCRIPTION = "Commanded Current #2"

Commanded cold amplifier current 2 (micro A)

const string apexHFE_ColdAmp::ColdAmp::cmdCurrent3_DESCRIPTION = "Commanded Current #3"

Commanded cold amplifier current 3 (micro A)

const string apexHFE_ColdAmp::ColdAmp::cmdIFAtten_DESCRIPTION = "Commanded IF Attenuation"

Commanded IF attenuation

const string apexHFE_ColdAmp::ColdAmp::cmdIFLevel_DESCRIPTION = "Commanded IF Output Level"

Commanded IF level

const string apexHFE_ColdAmp::ColdAmp::cmsBiasMode_DESCRIPTION = ”Commanded Bias Mode”

Commanded cold amplifier bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_ColdAmp::ColdAmp::current1_DESCRIPTION = ”Current #1”

Cold amplifier current 1 (micro A)

const string apexHFE_ColdAmp::ColdAmp::current2_DESCRIPTION = ”Current #2”

Cold amplifier current 2 (micro A)

const string apexHFE_ColdAmp::ColdAmp::current3_DESCRIPTION = ”Current #3”

Cold amplifier current 3 (micro A)

const string apexHFE_ColdAmp::ColdAmp::IFAtten_DESCRIPTION = ”IF Attenuation”

IF attenuation

const string apexHFE_ColdAmp::ColdAmp::IFLevel_DESCRIPTION = ”IF Output Level”

IF level

const string apexHFE_ColdAmp::ColdAmp::state_DESCRIPTION = ”State”

Actual State of the cold amplifier

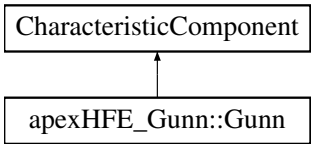
const string apexHFE_ColdAmp::ColdAmp::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

- apexHFE_ColdAmp.idl

3.9 apexHFE_Gunn::Gunn Interface Reference

`import"apexHFE_Gunn.idl";`
Inheritance diagram for apexHFE_Gunn::Gunn:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**
- const string frequency_DESCRIPTION = ”Frequency”
- const string **frequency_UNITS** = ”GHz”
- const string **frequency_FORMAT** = ”%.9f”
- const string **frequency_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **frequency**
- const string cmdFrequency_DESCRIPTION = ”Commanded Frequency”
- const string **cmdFrequency_UNITS** = ”GHz”
- readonly attribute ACS::RWdouble **cmdFrequency**
- const string biasMode_DESCRIPTION = ”Bias Mode”
- const string **biasMode_UNITS** = ””
- const string **biasMode_SAMPLEINTERVAL** = ”60”
- readonly attribute RObiasMode **biasMode**
- const string cmdBiasMode_DESCRIPTION = ”Commanded Bias Mode”
- const string **cmdBiasMode_UNITS** = ””
- readonly attribute RWbiasMode **cmdBiasMode**
- const string bias_DESCRIPTION = ”Bias Voltage”
- const string **bias_UNITS** = ”mV”
- const string **bias_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **bias**
- const string cmdBias_DESCRIPTION = ”Commanded Bias Voltage”
- const string **cmdBias_UNITS** = ”mV”
- readonly attribute ACS::RWdouble **cmdBias**
- const string current_DESCRIPTION = ”Current”
- const string **current_UNITS** = ”muA”
- const string **current_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **current**
- const string cmdCurrent_DESCRIPTION = ”Commanded Current”
- const string **cmdCurrent_UNITS** = ”muA”

- readonly attribute ACS::RWdouble **cmdCurrent**
- const string tuner_DESCRIPTION = "Tuner"
- const string **tuner_UNITS** = "counts"
- const string **tuner_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROdouble **tuner**
- const string cmdTuner_DESCRIPTION = "Commanded Tuner"
- const string **cmdTuner_UNITS** = "counts"
- readonly attribute ACS::RWdouble **cmdTuner**
- const string BShort_DESCRIPTION = "Backshort"
- const string **BShort_UNITS** = "counts"
- const string **BShort_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROdouble **BShort**
- const string cmdBShort_DESCRIPTION = "Commanded Backshort"
- const string **cmdBShort_UNITS** = "counts"
- readonly attribute ACS::RWdouble **cmdBShort**

3.9.1 Detailed Description

This interface belongs to the abstract Heterodyne Front End device. It will be the standard interface for APEX Heterodyne Front End Units.

3.9.2 Member Function Documentation

void apexHFE_Gunn::Gunn::off (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexHFE_Gunn::Gunn::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

void apexHFE_Gunn::Gunn::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.9.3 Member Data Documentation

const string apexHFE_Gunn::Gunn::bias_DESCRIPTION = "Bias Voltage"

Gunn bias voltage (mV)

const string apexHFE_Gunn::Gunn::biasMode_DESCRIPTION = "Bias Mode"

Gunn bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_Gunn::Gunn::BShort_DESCRIPTION = "Backshort"

Gunn backshort setting (counts)

const string apexHFE_Gunn::Gunn::cmdBias_DESCRIPTION = ”Commanded Bias Voltage”

Commanded Gunn bias voltage (mV)

const string apexHFE_Gunn::Gunn::cmdBiasMode_DESCRIPTION = ”Commanded Bias Mode”

Commanded Gunn bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_Gunn::Gunn::cmdBShort_DESCRIPTION = ”Commanded Backshort”

Commanded Gunn backshort setting (counts)

const string apexHFE_Gunn::Gunn::cmdCurrent_DESCRIPTION = ”Commanded Current”

Commanded Gunn current (micro A)

const string apexHFE_Gunn::Gunn::cmdFrequency_DESCRIPTION = ”Commanded Frequency”

Commanded Gunn frequency (GHz)

const string apexHFE_Gunn::Gunn::cmdTuner_DESCRIPTION = ”Commanded Tuner”

Commanded Gunn tuner setting (counts)

const string apexHFE_Gunn::Gunn::current_DESCRIPTION = ”Current”

Gunn current (micro A)

const string apexHFE_Gunn::Gunn::frequency_DESCRIPTION = ”Frequency”

Gunn frequency (GHz)

const string apexHFE_Gunn::Gunn::state_DESCRIPTION = ”State”

Actual State of the Gunn

const string apexHFE_Gunn::Gunn::tuner_DESCRIPTION = ”Tuner”

Gunn tuner setting (counts)

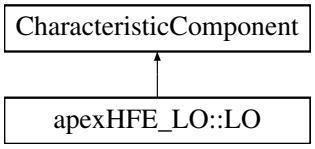
const string apexHFE_Gunn::Gunn::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

- apexHFE_Gunn.idl

3.10 apexHFE_LO::LO Interface Reference

`import"apexHFE_LO.idl";`
Inheritance diagram for apexHFE_LO::LO:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string frequency_DESCRIPTION = ”Frequency”
- const string **frequency_UNITS** = ”GHz”
- const string **frequency_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **frequency**
- const string cmdFrequency_DESCRIPTION = ”Commanded Frequency”
- const string **cmdFrequency_UNITS** = ”GHz”
- readonly attribute ACS::RWdouble **cmdFrequency**
- const string powerLevel_DESCRIPTION = ”Power Level”
- const string **powerLevel_UNITS** = ”dBm”
- const string **powerLevel_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **powerLevel**
- const string cmdPowerLevel_DESCRIPTION = ”Commanded Power Level”
- const string **cmdPowerLevel_UNITS** = ”dBm”
- readonly attribute ACS::RWdouble **cmdPowerLevel**
- const string refSynthFreq_DESCRIPTION = ”Reference Frequency”
- const string **refSynthFreq_UNITS** = ”GHz”
- const string **refSynthFreq_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **refSynthFreq**
- const string cmdRefSynthFreq_DESCRIPTION = ”Commanded Reference Frequency”
- const string **cmdRefSynthFreq_UNITS** = ”GHz”
- readonly attribute ACS::RWdouble **cmdRefSynthFreq**
- const string attenuator1_DESCRIPTION = ”Attenuator #1”
- const string **attenuator1_UNITS** = ”dB”
- const string **attenuator1_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **attenuator1**
- const string cmdAttenuator1_DESCRIPTION = ”Commanded Attenuator #1”
- const string **cmdAttenuator1_UNITS** = ”dB”
- readonly attribute ACS::RWdouble **cmdAttenuator1**
- const string attenuator2_DESCRIPTION = ”Attenuator #2”
- const string **attenuator2_UNITS** = ”dB”
- const string **attenuator2_SAMPLEINTERVAL** = ”60”

- readonly attribute ACS::ROdouble **attenuator2**
- const string cmdAttenuator2_DESCRIPTION = ”Commanded Attenuator #2”
- const string **cmdAttenuator2_UNITS** = ”dB”
- readonly attribute ACS::RWdouble **cmdAttenuator2**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**

3.10.1 Detailed Description

This interface belongs to the abstract Heterodyne Front End device. It will be the standard interface for APEX Heterodyne Front End Units.

3.10.2 Member Function Documentation

void apexHFE_LO::LO::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexHFE_LO::LO::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

void apexHFE_LO::LO::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.10.3 Member Data Documentation

const string apexHFE_LO::LO::attenuator1_DESCRIPTION = ”Attenuator #1”

LO attenuator 1 setting (dB)

const string apexHFE_LO::LO::attenuator2_DESCRIPTION = ”Attenuator #2”

LO attenuator 2 setting (dB)

const string apexHFE_LO::LO::cmdAttenuator1_DESCRIPTION = ”Commanded Attenuator #1”

Commanded LO attenuator 1 setting (dB)

const string apexHFE_LO::LO::cmdAttenuator2_DESCRIPTION = ”Commanded Attenuator #2”

Commanded LO attenuator 2 setting (dB)

const string apexHFE_LO::LO::cmdFrequency_DESCRIPTION = ”Commanded Frequency”

Commanded LO frequency (GHz)

const string apexHFE_LO::LO::cmdPowerLevel_DESCRIPTION = ”Commanded Power Level”

Commanded LO frequency (dBm)

const string apexHFE_LO::LO::cmdRefSynthFreq_DESCRIPTION = ”Commanded Reference Frequency”

Commanded LO reference frequency (GHz)

const string apexHFE_LO::LO::frequency_DESCRIPTION = ”Frequency”

LO frequency (GHz)

const string apexHFE_LO::LO::powerLevel_DESCRIPTION = ”Power Level”

LO power level (dBm)

const string apexHFE_LO::LO::refSynthFreq_DESCRIPTION = ”Reference Frequency”

LO reference frequency. The synthesizer (typically 2-20 GHz range) is set via a different DO. This is just for information for the embedded system. (GHz)

const string apexHFE_LO::LO::state_DESCRIPTION = ”State”

Actual State of the Heterodyne Frontend Local Oscillator

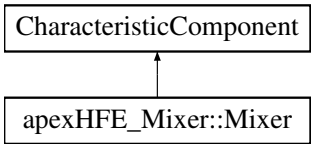
const string apexHFE_LO::LO::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

- apexHFE_LO.idl

3.11 apexHFE_Mixer::Mixer Interface Reference

`import"apexHFE_Mixer.idl";`
Inheritance diagram for apexHFE_Mixer::Mixer:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**
- const string imageMode_DESCRIPTION = ”Image Mode”
- const string **imageMode_UNITS** = ””
- const string **imageMode_SAMPLEINTERVAL** = ”60”
- readonly attribute ROimageMode **imageMode**
- const string cmdImageMode_DESCRIPTION = ”Commanded Image Mode”
- const string **cmdImageMode_UNITS** = ””
- readonly attribute RWimageMode **cmdImageMode**
- const string biasMode_DESCRIPTION = ”Bias Mode”
- const string **biasMode_UNITS** = ””
- const string **biasMode_SAMPLEINTERVAL** = ”60”
- readonly attribute RObiasMode **biasMode**
- const string cmdBiasMode_DESCRIPTION = ”Commanded Bias Mode”
- const string **cmdBiasMode_UNITS** = ””
- readonly attribute RWbiasMode **cmdBiasMode**
- const string bias_DESCRIPTION = ”Bias Voltage”
- const string **bias_UNITS** = ”mV”
- const string **bias_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **bias**
- const string cmdBias_DESCRIPTION = ”Commanded Bias Voltage”
- const string **cmdBias_UNITS** = ”mV”
- readonly attribute ACS::RWdouble **cmdBias**
- const string current_DESCRIPTION = ”Current”
- const string **current_UNITS** = ”muA”
- const string **current_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **current**
- const string cmdCurrent_DESCRIPTION = ”Commanded Current”
- const string **cmdCurrent_UNITS** = ”muA”
- readonly attribute ACS::RWdouble **cmdCurrent**

- const string BShort_DESCRIPTION = ”Backshort”
- const string **BShort_UNITS** = ”counts”
- const string **BShort_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **BShort**
- const string cmdBShort_DESCRIPTION = ”Commanded Backshort”
- const string **cmdBShort_UNITS** = ”counts”
- readonly attribute ACS::RWdouble **cmdBShort**
- const string EPlane_DESCRIPTION = ”E-Plane”
- const string **EPlane_UNITS** = ”counts”
- const string **EPlane_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **EPlane**
- const string cmdEPlane_DESCRIPTION = ”Commanded E-Plane”
- const string **cmdEPlane_UNITS** = ”counts”
- readonly attribute ACS::RWdouble **cmdEPlane**
- const string magFieldCurrent_DESCRIPTION = ”Magnetic Field Current”
- const string **magFieldCurrent_UNITS** = ”mA”
- const string **magFieldCurrent_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **magFieldCurrent**
- const string cmdMagFieldCurrent_DESCRIPTION = ”Commanded Magnetic Field Current”
- const string **cmdMagFieldCurrent_UNITS** = ”mA”
- readonly attribute ACS::RWdouble **cmdMagFieldCurrent**
- const string heater_DESCRIPTION = ”Heater Status”
- const string **heater_UNITS** = ””
- const string **heater_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **heater**
- const string cmdHeater_DESCRIPTION = ”Commanded Heater Status”
- const string **cmdHeater_UNITS** = ””
- readonly attribute RWlogical **cmdHeater**

3.11.1 Detailed Description

This interface belongs to the abstract Heterodyne Front End device. It will be the standard interface for APEX Heterodyne Front End Units.

3.11.2 Member Function Documentation

void apexHFE_Mixer::Mixer::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexHFE_Mixer::Mixer::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

void apexHFE_Mixer::Mixer::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.11.3 Member Data Documentation

const string apexHFE_Mixer::Mixer::bias_DESCRIPTION = "Bias Voltage"

Mixer bias voltage (mV)

const string apexHFE_Mixer::Mixer::biasMode_DESCRIPTION = "Bias Mode"

Mixer bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_Mixer::Mixer::BShort_DESCRIPTION = "Backshort"

Mixer backshort setting (counts)

const string apexHFE_Mixer::Mixer::cmdBias_DESCRIPTION = "Commanded Bias Voltage"

Commanded mixer bias voltage (mV)

const string apexHFE_Mixer::Mixer::cmdBiasMode_DESCRIPTION = "Commanded Bias Mode"

Commanded mixer bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_Mixer::Mixer::cmdBShort_DESCRIPTION = "Commanded Backshort"

Commanded mixer backshort setting (counts)

const string apexHFE_Mixer::Mixer::cmdCurrent_DESCRIPTION = "Commanded Current"

Commanded mixer current (micro A)

const string apexHFE_Mixer::Mixer::cmdEPlane_DESCRIPTION = "Commanded E-Plane"

Commanded mixer E-plane setting (counts)

const string apexHFE_Mixer::Mixer::cmdHeater_DESCRIPTION = "Commanded Heater Status"

Toggle mixer heater on/off (ON, OFF)

const string apexHFE_Mixer::Mixer::cmdImageMode_DESCRIPTION = "Commanded Image Mode"

Commanded mixer image mode (SSB, DSB)

const string apexHFE_Mixer::Mixer::cmdMagFieldCurrent_DESCRIPTION = "Commanded Magnetic Field Current"

Commanded mixer magnetic field current (mA). The sign defines the polarity.

const string apexHFE_Mixer::Mixer::current_DESCRIPTION = "Current"

Mixer current (micro A)

const string apexHFE_Mixer::Mixer::EPlane_DESCRIPTION = "E-Plane"

Mixer E-plane setting (counts)

const string apexHFE_Mixer::Mixer::heater_DESCRIPTION = "Heater Status"

Mixer heater status (ON, OFF)

const string apexHFE_Mixer::Mixer::imageMode_DESCRIPTION = "Image Mode"

Mixer image mode (SSB, DSB)

const string apexHFE_Mixer::Mixer::magFieldCurrent_DESCRIPTION = ”Magnetic Field Current”

Mixer magnetic field current (mA)

const string apexHFE_Mixer::Mixer::state_DESCRIPTION = ”State”

Actual State of the mixer

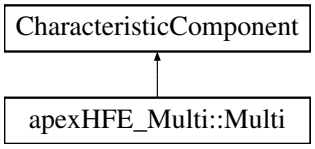
const string apexHFE_Mixer::Mixer::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

- apexHFE_Mixer.idl

3.12 apexHFE_Multi::Multi Interface Reference

`import"apexHFE_Multi.idl";`
Inheritance diagram for apexHFE_Multi::Multi:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**
- const string biasMode_DESCRIPTION = ”Bias Mode”
- const string **biasMode_UNITS** = ””
- const string **biasMode_SAMPLEINTERVAL** = ”60”
- readonly attribute RObiasMode **biasMode**
- const string cmdBiasMode_DESCRIPTION = ”Commanded Bias Mode”
- const string **cmdBiasMode_UNITS** = ””
- readonly attribute ACS::RWdouble **cmdBiasMode**
- const string bias_DESCRIPTION = ”Bias Voltage”
- const string **bias_UNITS** = ”mV”
- const string **bias_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **bias**
- const string cmdBias_DESCRIPTION = ”Commanded Bias Voltage”
- const string **cmdBias_UNITS** = ”mV”
- readonly attribute ACS::RWdouble **cmdBias**
- const string current_DESCRIPTION = ”Current”
- const string **current_UNITS** = ”muA”
- const string **current_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **current**
- const string cmdCurrent_DESCRIPTION = ”Commanded Current”
- const string **cmdCurrent_UNITS** = ”muA”
- readonly attribute ACS::RWdouble **cmdCurrent**
- const string BShort1_DESCRIPTION = ”Backshort #1”
- const string **BShort1_UNITS** = ”counts”
- const string **BShort1_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **BShort1**
- const string cmdBShort1_DESCRIPTION = ”Commanded Backshort #1”
- const string **cmdBShort1_UNITS** = ”counts”
- readonly attribute ACS::RWdouble **cmdBShort1**

- const string BShort2_DESCRIPTION = ”Backshort #2”
- const string **BShort2_UNITS** = ”counts”
- const string **BShort2_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **BShort2**
- const string cmdBShort2_DESCRIPTION = ”Commanded Backshort #2”
- const string **cmdBShort2_UNITS** = ”counts”
- readonly attribute ACS::RWdouble **cmdBShort2**

3.12.1 Detailed Description

This interface belongs to the abstract Heterodyne Front End device. It will be the standard interface for APEX Heterodyne Front End Units.

3.12.2 Member Function Documentation

void apexHFE_Multi::Multi::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexHFE_Multi::Multi::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

void apexHFE_Multi::Multi::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.12.3 Member Data Documentation

const string apexHFE_Multi::Multi::bias_DESCRIPTION = ”Bias Voltage”

Multiplier bias voltage (mV)

const string apexHFE_Multi::Multi::biasMode_DESCRIPTION = ”Bias Mode”

Multiplier bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_Multi::Multi::BShort1_DESCRIPTION = ”Backshort #1”

Multiplier backshort 1 setting (counts)

const string apexHFE_Multi::Multi::BShort2_DESCRIPTION = ”Backshort #2”

Multiplier backshort 2 setting (counts)

const string apexHFE_Multi::Multi::cmdBias_DESCRIPTION = ”Commanded Bias Voltage”

Commanded multiplier bias voltage (mV)

const string apexHFE_Multi::Multi::cmdBiasMode_DESCRIPTION = ”Commanded Bias Mode”

Commanded multiplier bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_Multi::Multi::cmdBShort1_DESCRIPTION = ”Commanded Backshort #1”

Commanded multiplier backshort 1 setting (counts)

const string apexHFE_Multi::Multi::cmdBShort2_DESCRIPTION = ”Commanded Backshort #2”

Commanded multiplier backshort 2 setting (counts)

const string apexHFE_Multi::Multi::cmdCurrent_DESCRIPTION = ”Commanded Current”

Commanded multiplier current (micro A)

const string apexHFE_Multi::Multi::current_DESCRIPTION = ”Current”

Multiplier current (micro A)

const string apexHFE_Multi::Multi::state_DESCRIPTION = ”State”

Actual State of the multiplier

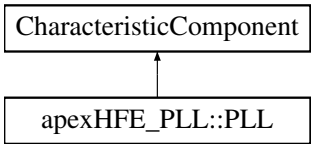
const string apexHFE_Multi::Multi::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

- apexHFE_Multi.idl

3.13 apexHFE_PLL::PLL Interface Reference

`import"apexHFE_PLL.idl";`
Inheritance diagram for apexHFE_PLL::PLL:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**
- const string synthFreq_DESCRIPTION = ”Synthesizer Frequency”
- const string **synthFreq_UNITS** = ”MHz”
- readonly attribute ACS::ROdouble **synthFreq**
- const string cmdSynthFreq_DESCRIPTION = ”Commanded Synthesizer Frequency”
- const string **cmdSynthFreq_UNITS** = ”MHz”
- readonly attribute ACS::RWdouble **cmdSynthFreq**
- const string harmNum_DESCRIPTION = ”Harmonic Number”
- const string **harmNum_UNITS** = ””
- const string **harmNum_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **harmNum**
- const string cmdHarmNum_DESCRIPTION = ”Commanded Harmonic Number”
- const string **cmdHarmNum_UNITS** = ””
- readonly attribute ACS::RWlong **cmdHarmNum**
- const string recHarmNum_DESCRIPTION = ”Recommended Harmonic Number”
- const string **recHarmNum_UNITS** = ””
- const string **recHarmNum_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **recHarmNum**
- const string HMixBiasMode_DESCRIPTION = ”Harmonic Mixer Bias Mode”
- const string **HMixBiasMode_UNITS** = ””
- const string **HMixBiasMode_SAMPLEINTERVAL** = ”60”
- readonly attribute RObiasMode **HMixBiasMode**
- const string cmdHMixBiasMode_DESCRIPTION = ”Commanded Harmonic Mixer Bias Mode”
- const string **cmdHMixBiasMode_UNITS** = ””
- readonly attribute RWbiasMode **cmdHMixBiasMode**
- const string HMixBias_DESCRIPTION = ”Harmonic Mixer Bias Voltage”
- const string **HMixBias_UNITS** = ”mV”
- const string **HMixBias_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **HMixBias**

- const string cmdHMixBias_DESCRIPTION = ”Harmonic Mixer Bias Voltage”
- const string **cmdHMixBias_UNITS** = ”mV”
- readonly attribute ACS::RWdouble **cmdHMixBias**
- const string HMixCurrent_DESCRIPTION = ”Harmonic Mixer Current”
- const string **HMixCurrent_UNITS** = ”muA”
- const string **HMixCurrent_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **HMixCurrent**
- const string cmdHMixCurrent_DESCRIPTION = ”Harmonic Mixer Current”
- const string **cmdHMixCurrent_UNITS** = ”muA”
- readonly attribute ACS::RWdouble **cmdHMixCurrent**
- const string freqLoop_DESCRIPTION = ”Frequency Loop State”
- const string **freqLoop_UNITS** = ””
- const string **freqLoop_ALARMON** = ”0”
- const string **freqLoop_SAMPLEINTERVAL** = ”60”
- readonly attribute ROloopState **freqLoop**
- const string cmdFreqLoop_DESCRIPTION = ”Commanded Frequency Loop State”
- const string **cmdFreqLoop_UNITS** = ””
- readonly attribute RWloopState **cmdFreqLoop**
- const string phaseLoop_DESCRIPTION = ”Phase Loop State”
- const string **phaseLoop_UNITS** = ””
- const string **phaseLoop_ALARMON** = ”0”
- const string **phaseLoop_SAMPLEINTERVAL** = ”60”
- readonly attribute ROloopState **phaseLoop**
- const string cmdPhaseLoop_DESCRIPTION = ”Commanded Phase Loop State”
- const string **cmdPhaseLoop_UNITS** = ””
- readonly attribute RWloopState **cmdPhaseLoop**
- const string lock_DESCRIPTION = ”Lock Status”
- const string **lock_UNITS** = ””
- const string **lock_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **lock**
- const string gain_DESCRIPTION = ”Gain”
- const string **gain_UNITS** = ””
- const string **gain_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **gain**
- const string cmdGain_DESCRIPTION = ”Commanded Gain”
- const string **cmdGain_UNITS** = ””
- readonly attribute ACS::RWdouble **cmdGain**
- const string holdRange_DESCRIPTION = ”Hold Range”
- const string **holdRange_UNITS** = ”%”
- const string **holdRange_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **holdRange**

3.13.1 Detailed Description

This interface belongs to the abstract Heterodyne Front End device. It will be the standard interface for APEX Heterodyne Front End Units.

3.13.2 Member Function Documentation

void apexHFE_PLL::PLL::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.

Parameters

cb	callback when completed
desc	description

void apexHFE_PLL::PLL::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

void apexHFE_PLL::PLL::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

3.13.3 Member Data Documentation

const string apexHFE_PLL::PLL::cmdFreqLoop_DESCRIPTION = "Commanded Frequency Loop State"

Commanded PLL frequency loop state

const string apexHFE_PLL::PLL::cmdGain_DESCRIPTION = "Commanded Gain"

Commanded PLL gain (-)

const string apexHFE_PLL::PLL::cmdHarmNum_DESCRIPTION = "Commanded Harmonic Number"

Commanded PLL harmonic number (-)

const string apexHFE_PLL::PLL::cmdHMixBias_DESCRIPTION = "Harmonic Mixer Bias Voltage"

Commanded PLL harmonic mixer bias voltage (mV)

const string apexHFE_PLL::PLL::cmdHMixBiasMode_DESCRIPTION = "Commanded Harmonic Mixer Bias Mode"

Commanded PLL harmonic mixer bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_PLL::PLL::cmdHMixCurrent_DESCRIPTION = "Harmonic Mixer Current"

Commanded harmonic mixer PLL bias current (micro A)

const string apexHFE_PLL::PLL::cmdPhaseLoop_DESCRIPTION = "Commanded Phase Loop State"

Commanded PLL phase loop state

const string apexHFE_PLL::PLL::cmdSynthFreq_DESCRIPTION = "Commanded Synthesizer Frequency"

Commanded PLL synthesizer frequency (MHz)

const string apexHFE_PLL::PLL::freqLoop_DESCRIPTION = "Frequency Loop State"

PLL frequency loop state (OPEN, CLOSED)

const string apexHFE_PLL::PLL::gain_DESCRIPTION = "Gain"

PLL gain (-)

const string apexHFE_PLL::PLL::harmNum_DESCRIPTION = "Harmonic Number"

PLL harmonic number (-)

const string apexHFE_PLL::PLL::HMixBias_DESCRIPTION = "Harmonic Mixer Bias Voltage"

PLL harmonic mixer bias voltage (mV)

const string apexHFE_PLL::PLL::HMixBiasMode_DESCRIPTION = "Harmonic Mixer Bias Mode"

PLL harmonic mixer bias mode (VOLTAGE, CURRENT, BOTH, PROTECTED)

const string apexHFE_PLL::PLL::HMixCurrent_DESCRIPTION = "Harmonic Mixer Current"

PLL harmonic mixer bias current (micro A)

const string apexHFE_PLL::PLL::holdRange_DESCRIPTION = "Hold Range"

PLL hold range(%)

const string apexHFE_PLL::PLL::lock_DESCRIPTION = "Lock Status"

PLL lock status

const string apexHFE_PLL::PLL::phaseLoop_DESCRIPTION = "Phase Loop State"

PLL phase loop state (OPEN, CLOSED)

const string apexHFE_PLL::PLL::recHarmNum_DESCRIPTION = "Recommended Harmonic Number"

Recommended PLL harmonic number (-)
Some receivers (e.g. the HET345) require certain harmonic numbers for different sky frequencies to get optimum results. The receiver control system may thus set this number upon receiving a new cmdSkyFrequency to let APECS know before setting up the reference synthesizers.

const string apexHFE_PLL::PLL::state_DESCRIPTION = "State"

Actual State of the PLL

const string apexHFE_PLL::PLL::synthFreq_DESCRIPTION = "Synthesizer Frequency"

PLL synthesizer frequency (MHz). Typically 100 MHz via 0.01-1 GHz Rohde & Schwarz synthesizer. The frequency is set via another DO (see apexSynthesizer.idl). This property is meant just for information for the embedded system. (MHz)

const string apexHFE_PLL::PLL::version_DESCRIPTION = "Version"

Software version
The documentation for this interface was generated from the following file:

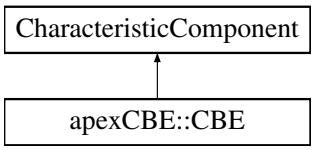
- apexHFE_PLL.idl

Chapter 4

Continuum Backends

4.1 apexCBE::CBE Interface Reference

`import"apexCBE.idl";`
Inheritance diagram for apexCBE::CBE:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void configure (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long start () raises (ACSErr::ACSException)
- long stop () raises (ACSErr::ACSException)
- long abort () raises (ACSErr::ACSException)
- void reset () raises (ACSErr::ACSException)

Public Attributes

- const string integrationTime_DESCRIPTION = ”Dump Time”
- const string **integrationTime_UNITS** = ”ms”
- const string **integrationTime_DISPLAYUNITS** = ”s”
- const string **integrationTime_FORMAT** = ”%.3f”
- const string **integrationTime_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **integrationTime**
- const string cmdIntegrationTime_DESCRIPTION = ”Commanded Dump Time”
- const string **cmdIntegrationTime_UNITS** = ”ms”
- const string **cmdIntegrationTime_DISPLAYUNITS** = ”s”
- const string **cmdIntegrationTime_FORMAT** = ”%.3f”
- readonly attribute ACS::RWlong **cmdIntegrationTime**
- const string usedChannels_DESCRIPTION = ”Used Channels”
- const string **usedChannels_UNITS** = ””
- readonly attribute ACS::ROlongSeq **usedChannels**
- const string cmdUsedChannels_DESCRIPTION = ”Commanded Used Channels”
- const string **cmdUsedChannels_UNITS** = ””
- readonly attribute ACS::RWlongSeq **cmdUsedChannels**
- const string syncTime_DESCRIPTION = ”Sync Time”

- const string **syncTime_UNITS** = ”mus”
- const string **syncTime_DISPLAYUNITS** = ”ms”
- const string **syncTime_FORMAT** = ”%.3f”
- const string **syncTime_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **syncTime**
- const string cmdSyncTime_DESCRIPTION = ”Commanded Sync Time”
- const string **cmdSyncTime_UNITS** = ”mus”
- const string **cmdSyncTime_DISPLAYUNITS** = ”ms”
- const string **cmdSyncTime_FORMAT** = ”%.3f”
- readonly attribute ACS::RWlong **cmdSyncTime**
- const string blankTime_DESCRIPTION = ”Blank Time”
- const string **blankTime_UNITS** = ”mus”
- const string **blankTime_DISPLAYUNITS** = ”ms”
- const string **blankTime_FORMAT** = ”%.3f”
- const string **blankTime_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **blankTime**
- const string cmdBlankTime_DESCRIPTION = ”Commanded Blank Time”
- const string **cmdBlankTime_UNITS** = ”mus”
- const string **cmdBlankTime_DISPLAYUNITS** = ”ms”
- const string **cmdBlankTime_FORMAT** = ”%.3f”
- readonly attribute ACS::RWlong **cmdBlankTime**
- const string numPhases_DESCRIPTION = ”Number of Phases”
- const string **numPhases_UNITS** = ””
- const string **numPhases_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **numPhases**
- const string cmdNumPhases_DESCRIPTION = ”Commanded Number of Phases”
- const string **cmdNumPhases_UNITS** = ””
- readonly attribute ACS::RWlong **cmdNumPhases**
- const string mode_DESCRIPTION = ”Blank / Sync Mode”
- const string **mode_UNITS** = ””
- const string **mode_SAMPLEINTERVAL** = ”60”
- readonly attribute ROmode **mode**
- const string cmdMode_DESCRIPTION = ”Commanded Blank / Sync Mode”
- const string **cmdMode_UNITS** = ””
- readonly attribute RWmode **cmdMode**
- const string gain_DESCRIPTION = ”Gain”
- const string **gain_UNITS** = ””
- const string **gain_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **gain**
- const string cmdGain_DESCRIPTION = ”Commanded Gain”
- const string **cmdGain_UNITS** = ””
- readonly attribute ACS::RWlong **cmdGain**
- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**

4.1.1 Detailed Description

This interface belongs to the abstract Continuum Back End device. It will be the standard interface for APEX Continuum Back End Units.

4.1.2 Member Function Documentation

long apexCBE::CBE::abort ( ) raises ACSErr::ACSException)

Abort taking data immediately.

void apexCBE::CBE::configure (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Set up the continuum backend according to the high-level properties that are currently defined. The embedded system must keep track of those values.
Parameters

cb	callback when completed
desc	description

void apexCBE::CBE::off (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexCBE::CBE::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

void apexCBE::CBE::reset ( ) raises ACSErr::ACSException)

Hardware reset.

long apexCBE::CBE::start ( ) raises ACSErr::ACSException)

Synchronize with signal from central timing generator and start taking data. This command must not block, i.e. the embedded system returns a status immediately and keeps taking data.

long apexCBE::CBE::stop ( ) raises ACSErr::ACSException)

Stop taking data. If possible, finish the observing cycle.

4.1.3 Member Data Documentation

const string apexCBE::CBE::blankTime_DESCRIPTION = "Blank Time"

Blanking time (micro seconds)

const string apexCBE::CBE::cmdBlankTime_DESCRIPTION = "Commanded Blank Time"

Commanded blanking time (micro seconds)
This is used by some backends (e.g. the ATNF / SEST correlator) for technical reasons to generate the blank/sync pulses rather than using the TSGEN pulses. The other backends should be able to measure this using the hardware input signals.

const string apexCBE::CBE::cmdGain_DESCRIPTION = "Commanded Gain"

Commanded gain
This is used by some backends to scale the input signal by 1/cmdGain to enhance the dynamic range.

const string apexCBE::CBE::cmdIntegrationTime_DESCRIPTION = "Commanded Dump Time"

Commanded intgeration time per dump (milli-seconds)

const string apexCBE::CBE::cmdMode_DESCRIPTION = "Commanded Blank / Sync Mode"

Commanded blank / sync mode.

const string apexCBE::CBE::cmdNumPhases_DESCRIPTION = "Commanded Number of Phases"

Commanded number of phases
This is used by some backends (e.g. the ATNF / SEST correlator) for technical reasons to generate the blank/sync pulses rather than using the TSGEN pulses. The other backends should be able to measure this using the hardware input signals.

const string apexCBE::CBE::cmdSyncTime_DESCRIPTION = "Commanded Sync Time"

Commanded sync time (micro seconds)
This is used by some backends (e.g. the ATNF / SEST correlator) for technical reasons to generate the blank/sync pulses rather than using the TSGEN pulses. The other backends should be able to measure this using the hardware input signals.

const string apexCBE::CBE::cmdUsedChannels_DESCRIPTION = "Commanded Used Channels"

Commanded list of continuum backend channels currently in use

const string apexCBE::CBE::gain_DESCRIPTION = "Gain"

Gain

const string apexCBE::CBE::integrationTime_DESCRIPTION = "Dump Time"

Integration time per dump (milli-seconds)

const string apexCBE::CBE::mode_DESCRIPTION = "Blank / Sync Mode"

Blank / sync mode. EXTERNAL means that the backend receives the hardware blank / sync signal from the timing signal generator. INTERNAL means that it generates blank / sync itself according to the commanded values (this is the case for the ASC correlator). The backend must, however, still obey the possibly delayed blank signal from the timing signal generator to blank out e.g. the wobbler transition phase.

const string apexCBE::CBE::numPhases_DESCRIPTION = "Number of Phases"

Number of phases

const string apexCBE::CBE::state_DESCRIPTION = "State"

Actual State of the continuum backend

const string apexCBE::CBE::syncTime_DESCRIPTION = "Sync Time"

Sync time (micro seconds)

const string apexCBE::CBE::usedChannels_DESCRIPTION = "Used Channels"

List of continuum backend channels currently in use (1: used channel, 0: unused channel (this should be a logical-Seq but this type does not exist yet)). The number of sequence elements must match the maximum number of continuum channels.

const string apexCBE::CBE::version_DESCRIPTION = "Version"

Hardware/software version
The documentation for this interface was generated from the following file:

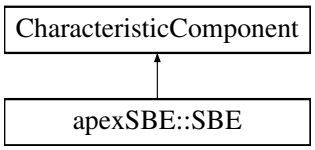
- apexCBE.idl

Chapter 5

Spectral Backends

5.1 apexSBE::SBE Interface Reference

`import"apexSBE.idl";`
Inheritance diagram for apexSBE::SBE:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void configure (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void stabilityTest (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long start () raises (ACSErr::ACSException)
- long stop () raises (ACSErr::ACSException)
- long abort () raises (ACSErr::ACSException)
- void reset () raises (ACSErr::ACSException)
- long comb () raises (ACSErr::ACSException)
- long noise () raises (ACSErr::ACSException)
- string status () raises (ACSErr::ACSException)

Public Attributes

- const string integrationTime_DESCRIPTION = ”Integration Time”
- const string **integrationTime_UNITS** = ”ms”
- const string **integrationTime_DISPLAYUNITS** = ”s”
- const string **integrationTime_FORMAT** = ”%.3f”
- const string **integrationTime_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **integrationTime**
- const string cmdIntegrationTime_DESCRIPTION = ”Commanded Integration Time”
- const string **cmdIntegrationTime_UNITS** = ”ms”
- const string **cmdIntegrationTime_DISPLAYUNITS** = ”s”
- const string **cmdIntegrationTime_FORMAT** = ”%.3f”
- readonly attribute ACS::RWlong **cmdIntegrationTime**
- const string usedSections_DESCRIPTION = ”Used Sections”
- const string **usedSections_UNITS** = ””
- readonly attribute ACS::ROlongSeq **usedSections**

- const string cmdUsedSections_DESCRIPTION = ”Commanded Used Sections”
- const string **cmdUsedSections_UNITS** = ””
- readonly attribute ACS::RWlongSeq **cmdUsedSections**
- const string blankTime_DESCRIPTION = ”Blank Time”
- const string **blankTime_UNITS** = ”mus”
- const string **blankTime_DISPLAYUNITS** = ”ms”
- const string **blankTime_FORMAT** = ”%.3f”
- const string **blankTime_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **blankTime**
- const string cmdBlankTime_DESCRIPTION = ”Commanded Blank Time”
- const string **cmdBlankTime_UNITS** = ”mus”
- const string **cmdBlankTime_DISPLAYUNITS** = ”ms”
- const string **cmdBlankTime_FORMAT** = ”%.3f”
- readonly attribute ACS::RWlong **cmdBlankTime**
- const string syncTime_DESCRIPTION = ”Sync Time”
- const string **syncTime_UNITS** = ”mus”
- const string **syncTime_DISPLAYUNITS** = ”ms”
- const string **syncTime_FORMAT** = ”%.3f”
- const string **syncTime_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **syncTime**
- const string cmdSyncTime_DESCRIPTION = ”Commanded Sync Time”
- const string **cmdSyncTime_UNITS** = ”mus”
- const string **cmdSyncTime_DISPLAYUNITS** = ”ms”
- const string **cmdSyncTime_FORMAT** = ”%.3f”
- readonly attribute ACS::RWlong **cmdSyncTime**
- const string numPhases_DESCRIPTION = ”Number of Phases”
- const string **numPhases_UNITS** = ””
- const string **numPhases_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **numPhases**
- const string cmdNumPhases_DESCRIPTION = ”Commanded Number of Phases”
- const string **cmdNumPhases_UNITS** = ””
- readonly attribute ACS::RWlong **cmdNumPhases**
- const string mode_DESCRIPTION = ”Blank / Sync Mode”
- const string **mode_UNITS** = ””
- const string **mode_SAMPLEINTERVAL** = ”60”
- readonly attribute ROmode **mode**
- const string cmdMode_DESCRIPTION = ”Commanded Blank / Sync Mode”
- const string **cmdMode_UNITS** = ””
- readonly attribute RWmode **cmdMode**
- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**

5.1.1 Detailed Description

This interface belongs to the abstract Spectrometer Back End device. It will be the standard interface for APEX Spectrometer Back End Units.

5.1.2 Member Function Documentation

long apexSBE::SBE::abort ( ) raises ACSErr::ACSException)

Abort taking data immediately.

long apexSBE::SBE::comb ( ) raises ACSErr::ACSException)

Generate a comb spectrum for frequency calibration.

void apexSBE::SBE::configure (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Set up the spectral backend according to the high-level properties that are currently defined. The embedded system must keep track of those values.
Parameters

cb	callback when completed
desc	description

long apexSBE::SBE::noise ( ) raises ACSErr::ACSException)

Measure noise generator spectrum for test purposes.

void apexSBE::SBE::off (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexSBE::SBE::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

void apexSBE::SBE::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

void apexSBE::SBE::stabilityTest (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Start a stability test using the noise diode.
Parameters

cb	callback when completed
desc	description

long apexSBE::SBE::start ( ) raises ACSErr::ACSException)

Synchronise with signal from central timing generator, and start taking data and sending it via the TCP output stream.

string apexSBE::SBE::status ( ) raises ACSErr::ACSException)

Status of observation / instrument - what it is currently doing. May include information that is not available via properties.

long apexSBE::SBE::stop ( ) raises ACSErr::ACSException)

Stop taking/sending data. If possible, finish the observing cycle.

5.1.3 Member Data Documentation

const string apexSBE::SBE::blankTime_DESCRIPTION = "Blank Time"

Blanking time (micro seconds)

const string apexSBE::SBE::cmdBlankTime_DESCRIPTION = "Commanded Blank Time"

Commanded blanking time (micro seconds)
This is used by some backends (e.g. the ATNF / SEST correlator) for technical reasons to generate the blank/sync pulses rather than using the TSGEN pulses. The other backends should be able to measure this using the hardware input signals.

const string apexSBE::SBE::cmdIntegrationTime_DESCRIPTION = "Commanded Integration Time"

Commanded intgeration time per dump (milli-seconds)

const string apexSBE::SBE::cmdMode_DESCRIPTION = "Commanded Blank / Sync Mode"

Commanded blank / sync mode.

const string apexSBE::SBE::cmdNumPhases_DESCRIPTION = "Commanded Number of Phases"

Commanded number of phases.
This is used by some backends (e.g. the ATNF / SEST correlator) for technical reasons to generate the blank/sync pulses rather than using the TSGEN pulses. The other backends should be able to measure this using the hardware input signals.

const string apexSBE::SBE::cmdSyncTime_DESCRIPTION = "Commanded Sync Time"

Commanded sync time (micro seconds)
This is used by some backends (e.g. the ATNF / SEST correlator) for technical reasons to generate the blank/sync pulses rather than using the TSGEN pulses. The other backends should be able to measure this using the hardware input signals.

const string apexSBE::SBE::cmdUsedSections_DESCRIPTION = "Commanded Used Sections"

Commanded list of spectral backend sections currently in use

const string apexSBE::SBE::integrationTime_DESCRIPTION = "Integration Time"

Integration time per dump (milli-seconds)

const string apexSBE::SBE::mode_DESCRIPTION = "Blank / Sync Mode"

Blank / sync mode. EXTERNAL means that the backend receives the hardware blank / sync signal from the timing signal generator. INTERNAL means that it generates blank / sync itself according to the commanded values (this is the case for the ASC correlator). The backend must, however, still obey the possibly delayed blank signal from the timing signal generator to blank out e.g. the wobbler transition phase.

const string apexSBE::SBE::numPhases_DESCRIPTION = "Number of Phases"

Number of phases

const string apexSBE::SBE::state_DESCRIPTION = "State"

Actual State of the spectral SBE

const string apexSBE::SBE::syncTime_DESCRIPTION = "Sync Time"

Sync time (micro seconds)

const string apexSBE::SBE::usedSections_DESCRIPTION = "Used Sections"

List of spectral backend sections currently in use (1: used section, 0: unused section (this should be a logicalSeq but this type does not exist yet)). The number of sequence elements must match the number of spectral backend sections.

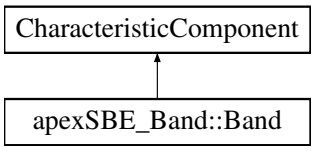
const string apexSBE::SBE::version_DESCRIPTION = "Version"

Hardware/software version
The documentation for this interface was generated from the following file:

- apexSBE.idl

5.2 apexSBE_Band::Band Interface Reference

`import"apexSBE_Band.idl";`
Inheritance diagram for apexSBE_Band::Band:



Public Attributes

- const string numSpecChan_DESCRIPTION = ”Number of Spectral Channels”
- const string **numSpecChan_UNITS** = ””
- const string **numSpecChan_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **numSpecChan**
- const string cmdNumSpecChan_DESCRIPTION = ”Commanded Number of Spectral Channels”
- const string **cmdNumSpecChan_UNITS** = ””
- readonly attribute ACS::RWlong **cmdNumSpecChan**
- const string maxNumSpecChan_DESCRIPTION = ”Maximum number of Spectral Channels”
- const string **maxNumSpecChan_UNITS** = ””
- readonly attribute ACS::ROlong **maxNumSpecChan**
- const string startChan_DESCRIPTION = ”Start Channel”
- const string **startChan_UNITS** = ””
- const string **startChan_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **startChan**
- const string cmdStartChan_DESCRIPTION = ”Commanded Start Channel”
- const string **cmdStartChan_UNITS** = ””
- readonly attribute ACS::RWlong **cmdStartChan**
- const string stopChan_DESCRIPTION = ”Stop Channel”
- const string **stopChan_UNITS** = ””
- const string **stopChan_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **stopChan**
- const string cmdStopChan_DESCRIPTION = ”Commanded Stop Channel”
- const string **cmdStopChan_UNITS** = ””
- readonly attribute ACS::RWlong **cmdStopChan**
- const string bandWidth_DESCRIPTION = ”Bandwidth”
- const string **bandWidth_UNITS** = ”MHz”
- const string **bandWidth_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **bandWidth**
- const string cmdBandWidth_DESCRIPTION = ”Commanded Bandwidth”
- const string **cmdBandWidth_UNITS** = ”MHz”
- readonly attribute ACS::RWdouble **cmdBandWidth**
- const string chanWidth_DESCRIPTION = ”Channel Width”
- const string **chanWidth_UNITS** = ”MHz”
- readonly attribute ACS::ROdouble **chanWidth**
- const string IFAtten_DESCRIPTION = ”IF Attenuation”
- const string **IFAtten_UNITS** = ”dB”
- const string **IFAtten_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **IFAtten**
- const string cmdIFAtten_DESCRIPTION = ”Commanded IF Attenuation”
- const string **cmdIFAtten_UNITS** = ”dB”
- readonly attribute ACS::RWlong **cmdIFAtten**
- const string IFLevel_DESCRIPTION = ”IF Power Level”

- const string **IFLevel_UNITS** = ”dBm”
- const string **IFLevel_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **IFLevel**
- const string darkCurrent_DESCRIPTION = ”Dark current”
- const string **darkCurrent_UNITS** = ”Counts”
- readonly attribute ACS::RODoubleSeq **darkCurrent**

5.2.1 Detailed Description

This interface belongs to the abstract Spectrometer Back End device. It will be the standard interface for APEX Spectrometer Back End Units.

5.2.2 Member Data Documentation

const string apexSBE_Band::Band::bandWidth_DESCRIPTION = ”Bandwidth”

Bandwidth (MHz)

const string apexSBE_Band::Band::chanWidth_DESCRIPTION = ”Channel Width”

Channel width (MHz)

const string apexSBE_Band::Band::cmdBandWidth_DESCRIPTION = ”Commanded Bandwidth”

Commanded bandwidth (MHz)

const string apexSBE_Band::Band::cmdIFAtten_DESCRIPTION = ”Commanded IF Attenuation”

Commanded IF attenuation (dB)

const string apexSBE_Band::Band::cmdNumSpecChan_DESCRIPTION = ”Commanded Number of Spectral Channels”

Commanded number of spectral channels

const string apexSBE_Band::Band::cmdStartChan_DESCRIPTION = ”Commanded Start Channel”

Commanded start channel (1-based in the frame of the maximum number of channels)

const string apexSBE_Band::Band::cmdStopChan_DESCRIPTION = ”Commanded Stop Channel”

Commanded stop channel (1-based in the frame of the maximum number of channels)

const string apexSBE_Band::Band::darkCurrent_DESCRIPTION = ”Dark current”

Dark current / zero offset

const string apexSBE_Band::Band::IFAtten_DESCRIPTION = ”IF Attenuation”

IF attenuation (dB)

const string apexSBE_Band::Band::IFLevel_DESCRIPTION = ”IF Power Level”

Current IF power level (dBm)

const string apexSBE_Band::Band::maxNumSpecChan_DESCRIPTION = ”Maximum number of Spectral Channels”

Maximum number of spectral channels

const string apexSBE_Band::Band::numSpecChan_DESCRIPTION = ”Number of Spectral Channels”

Number of spectral channels

const string apexSBE_Band::Band::startChan_DESCRIPTION = ”Start Channel”

Start channel (1-based in the frame of the maximum number of channels)

const string apexSBE_Band::Band::stopChan_DESCRIPTION = ”Stop Channel”

Stop channel (1-based in the frame of the maximum number of channels)
The documentation for this interface was generated from the following file:

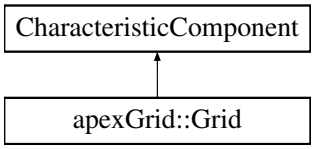
- apexSBE_Band.idl

Chapter 6

Auxiliary Devices

6.1 apexGrid::Grid Interface Reference

Inheritance diagram for apexGrid::Grid:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string angle_DESCRIPTION = ”Angle”
- const string **angle_UNITS** = ”deg”
- const string **angle_FORMAT** = ”%.3f”
- const string **angle_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **angle**

6.1.1 Member Data Documentation

const string apexGrid::Grid::angle_DESCRIPTION = ”Angle”

Grid angle

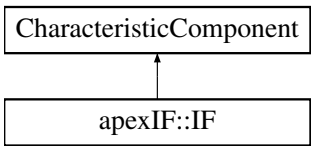
const string apexGrid::Grid::state_DESCRIPTION = ”State”

Actual State of the continuum frontend polarimeter unit
The documentation for this interface was generated from the following file:

- apexGrid.idl

6.2 apexIF::IF Interface Reference

`import"apexIF.idl";`
Inheritance diagram for apexIF::IF:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void configure (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**

6.2.1 Detailed Description

APEX IF Processor Interface

6.2.2 Member Function Documentation

void apexIF::IF::configure (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Set up the IF processor according to the high-level properties that are currently defined. The embedded system must keep track of those values.
Parameters

cb	callback when completed
desc	description

void apexIF::IF::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexIF::IF::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexIF::IF::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

6.2.3 Member Data Documentation

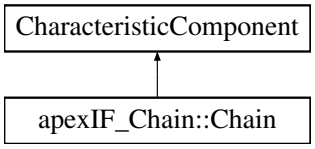
const string apexIF::IF::state_DESCRIPTION = "State"

Actual State of the IF
The documentation for this interface was generated from the following file:

- apexIF.idl

6.3 apexIF_Chain::Chain Interface Reference

`import"apexIF_Chain.idl";`
Inheritance diagram for apexIF_Chain::Chain:



Public Member Functions

- void autoLevel (in ACS::CBvoid cb, in ACS::CBDescIn desc)

Public Attributes

- const string input_DESCRIPTION = ”Input Channel”
- const string **input_UNITS** = ””
- const string **input_SAMPLEINTERVAL** = ”60”
- readonly attribute ROinputChannel **input**
- const string cmdInput_DESCRIPTION = ”Commanded Input Channel”
- const string **cmdInput_UNITS** = ””
- readonly attribute RWinputChannel **cmdInput**
- const string IFInput_DESCRIPTION = ”IF Chain Input”
- const string **IFInput_UNITS** = ””
- const string **IFInput_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **IFInput**
- const string cmdIFInput_DESCRIPTION = ”Commanded IF Chain Input”
- const string **cmdIFInput_UNITS** = ””
- readonly attribute ACS::RWlong **cmdIFInput**
- const string centerFreq_DESCRIPTION = ”Center Frequency”
- const string **centerFreq_UNITS** = ”GHz”
- const string **centerFreq_FORMAT** = ”%.9f”
- const string **centerFreq_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **centerFreq**
- const string cmdCenterFreq_DESCRIPTION = ”Commanded Center Frequency”
- const string **cmdCenterFreq_UNITS** = ”GHz”
- readonly attribute ACS::RWdouble **cmdCenterFreq**
- const string lock_DESCRIPTION = ”Lock Status”
- const string **lock_UNITS** = ””
- const string **lock_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **lock**
- const string bandWidth_DESCRIPTION = ”Bandwidth”
- const string **bandWidth_UNITS** = ”GHz”
- const string **bandWidth_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **bandWidth**
- const string cmdBandWidth_DESCRIPTION = ”Commanded Bandwidth”
- const string **cmdBandWidth_UNITS** = ”GHz”
- readonly attribute ACS::RWdouble **cmdBandWidth**
- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**

- const string level_DESCRIPTION = ”Output Power Level”
- const string **level_UNITS** = ”dBm”
- const string **level_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::RODouble **level**
- const string atten_DESCRIPTION = ”Input Attenuation”
- const string **atten_UNITS** = ”dB”
- const string **atten_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::RODouble **atten**
- const string cmdAtten_DESCRIPTION = ”Commanded Input Attenuation”
- const string **cmdAtten_UNITS** = ”dB”
- readonly attribute ACS::RWdouble **cmdAtten**
- const string freqAxisFlipped_DESCRIPTION = ”Frequency Axis Flipped”
- const string **freqAxisFlipped_UNITS** = ””
- const string **freqAxisFlipped_SAMPLEINTERVAL** = ”60”
- readonly attribute ROyesNo **freqAxisFlipped**

6.3.1 Detailed Description

APEX IF Chain Interface

6.3.2 Member Function Documentation

void apexIF_Chain::Chain::autoLevel (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Set the levels automatically to the optimum value. For some IF chains this can take longer than a few seconds, thus the method is asynchronous.

Parameters

cb	callback when completed
desc	description

6.3.3 Member Data Documentation

const string apexIF_Chain::Chain::atten_DESCRIPTION = ”Input Attenuation”

IF attenuation (dB) (Sequence denotes feed inputs)

const string apexIF_Chain::Chain::bandWidth_DESCRIPTION = ”Bandwidth”

IF bandwidth (GHz)

const string apexIF_Chain::Chain::centerFreq_DESCRIPTION = ”Center Frequency”

IF center frequency (GHz)

const string apexIF_Chain::Chain::cmdAtten_DESCRIPTION = ”Commanded Input Attenuation”

Commanded IF attenuation (dB) (Sequence denotes feed inputs)

const string apexIF_Chain::Chain::cmdBandWidth_DESCRIPTION = ”Commanded Bandwidth”

Commanded IF bandwidth (GHz)

const string apexIF_Chain::Chain::cmdCenterFreq_DESCRIPTION = ”Commanded Center Frequency”

Commanded IF center frequency (GHz)

const string apexIF_Chain::Chain::cmdIFInput_DESCRIPTION = ”Commanded IF Chain Input”

Commanded IF chain input (1-based) Used to switch the input past the filters and first amplifiers to the opposite IF input to be able to use all backend sections simultaneously for one receiver.

const string apexIF_Chain::Chain::cmdInput_DESCRIPTION = ”Commanded Input Channel”

Commanded input channel

const string apexIF_Chain::Chain::freqAxisFlipped_DESCRIPTION = ”Frequency Axis Flipped”

Flag to indicate whether the IF processor flips the frequency axis. This is used to determine the correct axis description when writing the raw data.

const string apexIF_Chain::Chain::IFInput_DESCRIPTION = ”IF Chain Input”

IF chain input (1-based)

const string apexIF_Chain::Chain::input_DESCRIPTION = ”Input Channel”

Input channel

const string apexIF_Chain::Chain::level_DESCRIPTION = ”Output Power Level”

IF output power level (dBm) (Sequence denotes feed inputs)

const string apexIF_Chain::Chain::lock_DESCRIPTION = ”Lock Status”

IF Synthesizer lock status

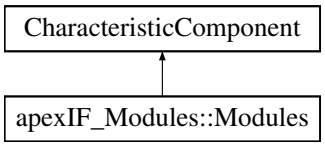
const string apexIF_Chain::Chain::state_DESCRIPTION = ”State”

Actual State of the IF
The documentation for this interface was generated from the following file:

- apexIF_Chain.idl

6.4 apexIF_Modules::Modules Interface Reference

`import"apexIF_Modules.idl";`
Inheritance diagram for apexIF_Modules::Modules:



Public Attributes

- const string usedModules_DESCRIPTION = ”Used Modules”
- const string **usedModules_UNITS** = ””
- readonly attribute ACS::ROlongSeq **usedModules**
- const string **cmdUsedModules_DESCRIPTION** = ”Commanded Used Modules”
- const string **cmdUsedModules_UNITS** = ””
- readonly attribute ACS::RWlongSeq **cmdUsedModules**

6.4.1 Detailed Description

APEX IF Processor Modules Interface

6.4.2 Member Data Documentation

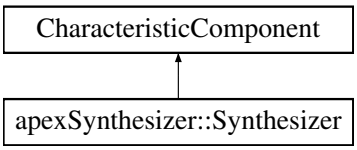
const string apexIF_Modules::Modules::usedModules_DESCRIPTION = ”Used Modules”

Used IF Modules (1-based sequence of active module numbers)
The documentation for this interface was generated from the following file:

- apexIF_Modules.idl

6.5 apexSynthesizer::Synthesizer Interface Reference

`import"apexSynthesizer.idl";`
Inheritance diagram for apexSynthesizer::Synthesizer:



Public Member Functions

- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string frequency_DESCRIPTION = ”Frequency”
- const string **frequency_UNITS** = ”GHz”
- const string **frequency_FORMAT** = ”%.9f”
- const string **frequency_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **frequency**
- const string cmdFrequency_DESCRIPTION = ”Commanded Frequency”
- const string **cmdFrequency_UNITS** = ”GHz”
- readonly attribute ACS::RWdouble **cmdFrequency**
- const string RFPowerLevel_DESCRIPTION = ”RF Power Level”
- const string **RFPowerLevel_UNITS** = ”dBm”
- const string **RFPowerLevel_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **RFPowerLevel**
- const string cmdRFPowerLevel_DESCRIPTION = ”Commanded RF Power Level”
- const string **cmdRFPowerLevel_UNITS** = ”dBm”
- readonly attribute ACS::RWdouble **cmdRFPowerLevel**
- const string RFPowerSwitch_DESCRIPTION = ”RF Power Switch”
- const string **RFPowerSwitch_UNITS** = ””
- const string **RFPowerSwitch_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **RFPowerSwitch**
- const string cmdRFPowerSwitch_DESCRIPTION = ”Commanded RF Power Switch”
- const string **cmdRFPowerSwitch_UNITS** = ””
- readonly attribute RWlogical **cmdRFPowerSwitch**

6.5.1 Detailed Description

Generic APEX Synthesizer Interface

6.5.2 Member Function Documentation

long apexSynthesizer::Synthesizer::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

6.5.3 Member Data Documentation

const string apexSynthesizer::Synthesizer::cmdFrequency_DESCRIPTION = "Commanded Frequency"

Commanded synthesizer frequency (GHz)

const string apexSynthesizer::Synthesizer::cmdRFPowerLevel_DESCRIPTION = "Commanded RF Power Level"

Commanded synthesizer RF power level (dBm)

const string apexSynthesizer::Synthesizer::cmdRFPowerSwitch_DESCRIPTION = "Commanded RF Power Switch"

Commanded synthesizer RF switch

const string apexSynthesizer::Synthesizer::frequency_DESCRIPTION = "Frequency"

Synthesizer frequency (GHz) for normal operation

const string apexSynthesizer::Synthesizer::RFPowerLevel_DESCRIPTION = "RF Power Level"

Synthesizer RF power level (dBm)

const string apexSynthesizer::Synthesizer::RFPowerSwitch_DESCRIPTION = "RF Power Switch"

Synthesizer RF power switch

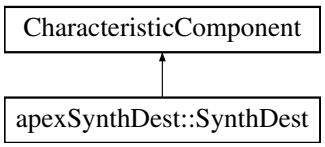
const string apexSynthesizer::Synthesizer::state_DESCRIPTION = "State"

Actual State of the synthesizer
The documentation for this interface was generated from the following file:

- apexSynthesizer.idl

6.6 apexSynthDest::SynthDest Interface Reference

`import"apexSynthDest.idl";`
Inheritance diagram for apexSynthDest::SynthDest:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string cabin_DESCRIPTION = ”Destination Cabin”
- const string **cabin_UNITS** = ””
- const string **cabin_SAMPLEINTERVAL** = ”60”
- readonly attribute ROcabinName **cabin**
- const string cmdCabin_DESCRIPTION = ”Commanded Destination Cabin”
- const string **cmdCabin_UNITS** = ””
- readonly attribute RWcabinName **cmdCabin**
- const string port_DESCRIPTION = ”Destination Port”
- const string **port_UNITS** = ””
- const string **port_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **_port**
- const string cmdPort_DESCRIPTION = ”Commanded Destination Port”
- const string **cmdPort_UNITS** = ””
- readonly attribute ACS::RWlong **cmdPort**

6.6.1 Detailed Description

Generic APEX Synthesizer Destination Interface

6.6.2 Member Data Documentation

const string apexSynthDest::SynthDest::cabin_DESCRIPTION = ”Destination Cabin”

Destination cabin (Facility receivers in A-cabin or CHAMP in B-cabin)

const string apexSynthDest::SynthDest::cmdCabin_DESCRIPTION = ”Commanded Destination Cabin”

Commanded destination cabin

const string apexSynthDest::SynthDest::cmdPort_DESCRIPTION = ”Commanded Destination Port”

Commanded destination port

const string apexSynthDest::SynthDest::port_DESCRIPTION = ”Destination Port”

Destination port

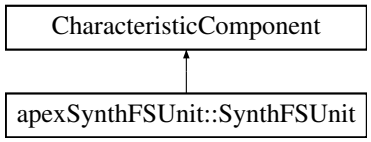
const string apexSynthDest::SynthDest::state_DESCRIPTION = ”State”

Actual State of the synthesizer destination switch box
The documentation for this interface was generated from the following file:

- apexSynthDest.idl

6.7 apexSynthFSUnit::SynthFSUnit Interface Reference

`import"apexSynthFSUnit.idl";`
Inheritance diagram for apexSynthFSUnit::SynthFSUnit:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string frequencyList_DESCRIPTION = ”Frequency List”
- const string **frequencyList_UNITS** = ”GHz”
- const string **frequencyList_FORMAT** = ”%.9f”
- readonly attribute ACS::RODoubleSeq **frequencyList**
- const string cmdFrequencyList_DESCRIPTION = ”Commanded Frequency List”
- const string **cmdFrequencyList_UNITS** = ”GHz”
- readonly attribute ACS::RWdoubleSeq **cmdFrequencyList**
- const string freqSwitchMode_DESCRIPTION = ”Frequency Switch Mode”
- const string **freqSwitchMode_UNITS** = ””
- const string **freqSwitchMode_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **freqSwitchMode**
- const string cmdFreqSwitchMode_DESCRIPTION = ”Commanded Frequency Switch Mode”
- const string **cmdFreqSwitchMode_UNITS** = ””
- readonly attribute RWlogical **cmdFreqSwitchMode**
- const string RFPowerList_DESCRIPTION = ”RF Power List”
- const string **RFPowerList_UNITS** = ”dBm”
- readonly attribute ACS::RODoubleSeq **RFPowerList**
- const string cmdRFPowerList_DESCRIPTION = ”Commanded RF Power List”
- const string **cmdRFPowerList_UNITS** = ”dBm”
- readonly attribute ACS::RWdoubleSeq **cmdRFPowerList**

6.7.1 Detailed Description

Generic APEX Synthesizer Frequency Switching Unit Interface

6.7.2 Member Data Documentation

const string apexSynthFSUnit::SynthFSUnit::cmdFreqSwitchMode_DESCRIPTION = ”Commanded Frequency Switch Mode”

Commanded frequency switching mode

const string apexSynthFSUnit::SynthFSUnit::cmdFrequencyList_DESCRIPTION = ”Commanded Frequency List”

Commanded synthesizer frequency (GHz) list for frequency switching

const string apexSynthFSUnit::SynthFSUnit::cmdRFPowerList_DESCRIPTION = ”Commanded RF Power List”

Commanded synthesizer RF power list for frequency switching (GHz)

const string apexSynthFSUnit::SynthFSUnit::freqSwitchMode_DESCRIPTION = ”Frequency Switch Mode”

Frequency switching mode.
Switching ”ON” causes the synthesizer to use the external hardware blank/sync signals to switch through the commanded list of frequencies and corresponding output power levels.
Switching ”OFF” goes back to single frequency mode determined by ”frequency” and ”RFPowerLevel”.

const string apexSynthFSUnit::SynthFSUnit::frequencyList_DESCRIPTION = ”Frequency List”

Synthesizer frequency list for frequency (GHz) switching

const string apexSynthFSUnit::SynthFSUnit::RFPowerList_DESCRIPTION = ”RF Power List”

Synthesizer RF power list for frequency switching (GHz)

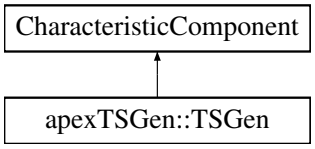
const string apexSynthFSUnit::SynthFSUnit::state_DESCRIPTION = ”State”

Actual State of the synthesizer frequency switching unit
The documentation for this interface was generated from the following file:

- apexSynthFSUnit.idl

6.8 apexTSGen::TSGen Interface Reference

`import"apexTSGen.idl";`
Inheritance diagram for apexTSGen::TSGen:



Public Member Functions

- void configure (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long clearHistory () raises (ACSErr::ACSException)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string syncTime_DESCRIPTION = "Sync Time"
- const string **syncTime_UNITS** = "mus"
- const string **syncTime_DISPLAYUNITS** = "ms"
- const string **syncTime_FORMAT** = "%.3f"
- const string **syncTime_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROlong **syncTime**
- const string cmdSyncTime_DESCRIPTION = "Commanded Sync Time"
- const string **cmdSyncTime_UNITS** = "mus"
- const string **cmdSyncTime_DISPLAYUNITS** = "ms"
- const string **cmdSyncTime_FORMAT** = "%.3f"
- readonly attribute ACS::RWlong **cmdSyncTime**
- const string numPhases_DESCRIPTION = "Number of Phases"
- const string **numPhases_UNITS** = ""
- const string **numPhases_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROlong **numPhases**
- const string cmdNumPhases_DESCRIPTION = "Commanded Number of Phases"
- const string **cmdNumPhases_UNITS** = ""
- readonly attribute ACS::RWlong **cmdNumPhases**
- const string blankTime_DESCRIPTION = "Blank Time"
- const string **blankTime_UNITS** = "mus"
- const string **blankTime_DISPLAYUNITS** = "ms"
- const string **blankTime_FORMAT** = "%.3f"
- const string **blankTime_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROlong **blankTime**
- const string cmdBlankTime_DESCRIPTION = "Commanded Blank Time"
- const string **cmdBlankTime_UNITS** = "mus"
- const string **cmdBlankTime_DISPLAYUNITS** = "ms"
- const string **cmdBlankTime_FORMAT** = "%.3f"
- readonly attribute ACS::RWlong **cmdBlankTime**
- const string offsetSyncTime_DESCRIPTION = "Offset Sync Time"
- const string **offsetSyncTime_UNITS** = "mus"
- const string **offsetSyncTime_DISPLAYUNITS** = "ms"
- const string **offsetSyncTime_FORMAT** = "%.3f"
- const string **offsetSyncTime_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROlong **offsetSyncTime**
- const string cmdOffsetSyncTime_DESCRIPTION = "Commanded Offset Sync Time"

- const string **cmdOffsetSyncTime_UNITS** = ”mus”
- const string **cmdOffsetSyncTime_DISPLAYUNITS** = ”ms”
- const string **cmdOffsetSyncTime_FORMAT** = ”%.3f”
- readonly attribute ACS::RWlong **cmdOffsetSyncTime**
- const string mode_DESCRIPTION = ”Mode”
- const string **mode_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **mode**
- const string cmdMode_DESCRIPTION = ”Commanded Mode”
- readonly attribute RWstatus **cmdMode**
- const string delayBlankPorts_DESCRIPTION = ”Delay Blank Port Status”
- const string **delayBlankPorts_UNITS** = ””
- readonly attribute ACS::ROlongSeq **delayBlankPorts**
- const string cmdDelayBlankPorts_DESCRIPTION = ”Commanded Delay Blank Port Status”
- const string **cmdDelayBlankPorts_UNITS** = ””
- readonly attribute ACS::RWlongSeq **cmdDelayBlankPorts**
- const string TSGBlankStatus_DESCRIPTION = ”Blank Signal Status”
- const string **TSGBlankStatus_UNITS** = ””
- const string **TSGBlankStatus_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **TSGBlankStatus**
- const string TSG10MHzStatus_DESCRIPTION = ”10MHz Signal Status”
- const string **TSG10MHzStatus_UNITS** = ””
- const string **TSG10MHzStatus_ALARMINTERVAL** = ”60”
- const string **TSG10MHzStatus_ALARMON** = ”0”
- const string **TSG10MHzStatus_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **TSG10MHzStatus**
- const string TSG1ppsStatus_DESCRIPTION = ”1PPS Signal Status”
- const string **TSG1ppsStatus_UNITS** = ””
- const string **TSG1ppsStatus_ALARMINTERVAL** = ”60”
- const string **TSG1ppsStatus_ALARMON** = ”0”
- const string **TSG1ppsStatus_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **TSG1ppsStatus**
- const string TSG6secStatus_DESCRIPTION = ”6sec Signal Status”
- const string **TSG6secStatus_UNITS** = ””
- const string **TSG6secStatus_ALARMINTERVAL** = ”60”
- const string **TSG6secStatus_ALARMON** = ”0”
- const string **TSG6secStatus_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **TSG6secStatus**
- const string TSGBlankHistory_DESCRIPTION = ”Blank Signal Status History”
- const string **TSGBlankHistory_UNITS** = ””
- readonly attribute ROlogical **TSGBlankHistory**
- const string TSG10MHzHistory_DESCRIPTION = ”10MHz Signal Status History”
- const string **TSG10MHzHistory_UNITS** = ””
- readonly attribute ROlogical **TSG10MHzHistory**
- const string TSG1ppsHistory_DESCRIPTION = ”1PPS Signal Status History”
- const string **TSG1ppsHistory_UNITS** = ””
- readonly attribute ROlogical **TSG1ppsHistory**
- const string TSG6secHistory_DESCRIPTION = ”6sec Signal Status History”
- const string **TSG6secHistory_UNITS** = ””
- readonly attribute ROlogical **TSG6secHistory**

6.8.1 Detailed Description

APEX Timing Signal Generator Interface

6.8.2 Member Function Documentation

long apexTSGen::TSGen::clearHistory ( ) raises ACSErr::ACSException)

Clears the History of the used Signals (Blank, 1pps, 10MHz, 6sec)

void apexTSGen::TSGen::configure (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Set up the timing signal generator according to the high-level properties that are currently defined. The embedded system must keep track of those values.

Parameters

cb	callback when completed
desc	description

long apexTSGen::TSGen::reset ( ) raises ACSErr::ACSException)

Resets the APEX Timing Signal Generator Interface

6.8.3 Member Data Documentation

const string apexTSGen::TSGen::blankTime_DESCRIPTION = "Blank Time"

Generated blanking time (micro seconds)

const string apexTSGen::TSGen::cmdBlankTime_DESCRIPTION = "Commanded Blank Time"

Commanded generated blanking time (micro seconds)

const string apexTSGen::TSGen::cmdDelayBlankPorts_DESCRIPTION = "Commanded Delay Blank Port Status"

Commanded DelayBlankOutput

const string apexTSGen::TSGen::cmdMode_DESCRIPTION = "Commanded Mode"

Commanded Blank/Sync Generator Mode

const string apexTSGen::TSGen::cmdNumPhases_DESCRIPTION = "Commanded Number of Phases"

Commanded generated number of phases

const string apexTSGen::TSGen::cmdOffsetSyncTime_DESCRIPTION = "Commanded Offset Sync Time"

Commanded generated offset sync time (micro seconds)

const string apexTSGen::TSGen::cmdSyncTime_DESCRIPTION = "Commanded Sync Time"

Commanded generated sync time (micro seconds)

const string apexTSGen::TSGen::delayBlankPorts_DESCRIPTION = "Delay Blank Port Status"

State of DelayBlankOutputPorts (8 ports; 0 = disabled, 1 = enabled)

const string apexTSGen::TSGen::mode_DESCRIPTION = "Mode"

Blank/Sync Generator Mode

const string apexTSGen::TSGen::numPhases_DESCRIPTION = "Number of Phases"

Measured number of phases

const string apexTSGen::TSGen::offsetSyncTime_DESCRIPTION = "Offset Sync Time"

Measured offset sync time (micro seconds).
This is the shift relative to the 48ms TE. It may be useful if the sync time is a multiple of 48ms so that the blank/sync signal is synchronized to the 48ms ticks.

const string apexTSGen::TSGen::syncTime_DESCRIPTION = "Sync Time"

Measured sync time (micro seconds)

const string apexTSGen::TSGen::TSG10MHzHistory_DESCRIPTION = "10MHz Signal Status History"

History of the TSGen 10MHz Signal

const string apexTSGen::TSGen::TSG10MHzStatus_DESCRIPTION = "10MHz Signal Status"

Status of the TSGen 10MHz Signal

const string apexTSGen::TSGen::TSG1ppsHistory_DESCRIPTION = "1PPS Signal Status History"

History of the TSGen 1 PPS Signal

const string apexTSGen::TSGen::TSG1ppsStatus_DESCRIPTION = "1PPS Signal Status"

Status of the TSGen 1 PPS Signal

const string apexTSGen::TSGen::TSG6secHistory_DESCRIPTION = "6sec Signal Status History"

History of the TSGen 6 sec Signal

const string apexTSGen::TSGen::TSG6secStatus_DESCRIPTION = "6sec Signal Status"

Status of the TSGen 6 sec Signal

const string apexTSGen::TSGen::TSGBlankHistory_DESCRIPTION = "Blank Signal Status History"

History of the TSGen blank Signal

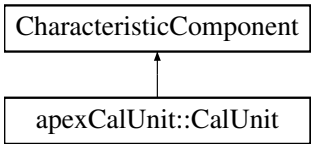
const string apexTSGen::TSGen::TSGBlankStatus_DESCRIPTION = "Blank Signal Status"

Status of the TSGen blank Signal
The documentation for this interface was generated from the following file:

- apexTSGen.idl

6.9 apexCalUnit::CalUnit Interface Reference

`import"apexCalUnit.idl";`
Inheritance diagram for apexCalUnit::CalUnit:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)
- double coldLoss (in double frequency) raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = "State"
- const string **state_UNITS** = ""
- const string **state_ALARMON** = "4"
- const string **state_SAMPLEINTERVAL** = "60"
- readonly attribute ROstatus **state**
- const string hotLoadTemp_DESCRIPTION = "Hot Load Temperature"
- const string **hotLoadTemp_UNITS** = "K"
- const string **hotLoadTemp_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROdouble **hotLoadTemp**
- const string coldLoadTemp_DESCRIPTION = "Cold Load Temperature"
- const string **coldLoadTemp_UNITS** = "K"
- const string **coldLoadTemp_SAMPLEINTERVAL** = "60"
- readonly attribute ACS::ROdouble **coldLoadTemp**
- const string chopMode_DESCRIPTION = "Chopper Mode"
- const string **chopMode_UNITS** = ""
- const string **chopMode_ALARMON** = "5"
- const string **chopMode_SAMPLEINTERVAL** = "60"
- readonly attribute ROchopState **chopMode**
- const string cmdChopMode_DESCRIPTION = "Commanded Chopper Mode"
- const string **cmdChopMode_UNITS** = ""
- readonly attribute RWchopState **cmdChopMode**
- const string version_DESCRIPTION = "Version"
- const string **version_UNITS** = ""
- readonly attribute ACS::ROstring **version**

6.9.1 Detailed Description

This interface belongs to the abstract calibration unit device. It will be the standard interface for APEX Calibration Units.

6.9.2 Member Function Documentation

double apexCalUnit::CalUnit::coldLoss (in double *frequency*) raises ACSErr::ACSException)

Read cold load loss depending on the frequency in GHz. Returns loss as number between 0.0 and 1.0.

void apexCalUnit::CalUnit::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.

Parameters

cb	callback when completed
desc	description

void apexCalUnit::CalUnit::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexCalUnit::CalUnit::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

6.9.3 Member Data Documentation

const string apexCalUnit::CalUnit::chopMode_DESCRIPTION = "Chopper Mode"

Chopper mode (HOT, COLD, LINE_INJECTOR, SKY)

const string apexCalUnit::CalUnit::cmdChopMode_DESCRIPTION = "Commanded Chopper Mode"

Commanded chopper mode (HOT, COLD, LINE_INJECTOR, SKY)

const string apexCalUnit::CalUnit::coldLoadTemp_DESCRIPTION = "Cold Load Temperature"

Cold load temperature (K)

const string apexCalUnit::CalUnit::hotLoadTemp_DESCRIPTION = "Hot Load Temperature"

Hot load temperature (K)

const string apexCalUnit::CalUnit::state_DESCRIPTION = "State"

Actual State of the Calibration Unit

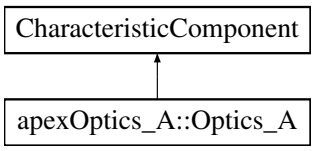
const string apexCalUnit::CalUnit::version_DESCRIPTION = "Version"

Software version
The documentation for this interface was generated from the following file:

- apexCalUnit.idl

6.10 apexOptics_A::Optics_A Interface Reference

`import"apexOptics_A.idl";`
Inheritance diagram for apexOptics_A::Optics_A:



Public Member Functions

- void **calibrate** (in ACS::CBvoid cb, in ACS::CBDescIn desc)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string position_DESCRIPTION = ”Mirror Position”
- const string **position_UNITS** = ””
- const string **position_ALARMON** = ”4”
- const string **position_SAMPLEINTERVAL** = ”60”
- readonly attribute ROposition **position**
- const string cmdPosition_DESCRIPTION = ”Commanded Mirror Position”
- const string **cmdPosition_UNITS** = ””
- readonly attribute RWposition **cmdPosition**

6.10.1 Detailed Description

APEX Cabin A Optics Interface

6.10.2 Member Data Documentation

const string apexOptics_A::Optics_A::cmdPosition_DESCRIPTION = ”Commanded Mirror Position”

Commanded mirror position (FACILITY, PI1, PI2)

const string apexOptics_A::Optics_A::position_DESCRIPTION = ”Mirror Position”

Mirror position (FACILITY, PI1, PI2, MOVING, UNDEFINED)

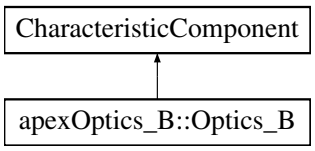
const string apexOptics_A::Optics_A::state_DESCRIPTION = ”State”

Actual State of the Cabin A Optics (INITIALIZE, ENABLED, DISABLED, SHUTDOWN, FAULTED)
The documentation for this interface was generated from the following file:

- apexOptics_A.idl

6.11 apexOptics_B::Optics_B Interface Reference

`import"apexOptics_B.idl";`
Inheritance diagram for apexOptics_B::Optics_B:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string position_DESCRIPTION = ”Mirror Position”
- const string **position_UNITS** = ””
- const string **position_ALARMON** = ”4”
- const string **position_SAMPLEINTERVAL** = ”60”
- readonly attribute ROposition **position**
- const string cmdPosition_DESCRIPTION = ”Commanded Mirror Position”
- const string **cmdPosition_UNITS** = ””
- readonly attribute RWposition **cmdPosition**

6.11.1 Detailed Description

APEX Cabin B Optics Interface

6.11.2 Member Data Documentation

const string apexOptics_B::Optics_B::cmdPosition_DESCRIPTION = ”Commanded Mirror Position”

Commanded mirror position (PI1, PI2, PI3, PI4)

const string apexOptics_B::Optics_B::position_DESCRIPTION = ”Mirror Position”

Mirror position (PI1, PI2, PI3, PI4, MOVING, UNDEFINED)

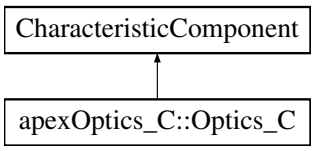
const string apexOptics_B::Optics_B::state_DESCRIPTION = ”State”

Actual State of the Cabin B Optics (INITIALIZE, ENABLED, DISABLED, SHUTDOWN, FAULTED)
The documentation for this interface was generated from the following file:

- apexOptics_B.idl

6.12 apexOptics_C::Optics_C Interface Reference

`import"apexOptics_C.idl";`
Inheritance diagram for apexOptics_C::Optics_C:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string position_DESCRIPTION = ”Mirror Position”
- const string **position_UNITS** = ””
- const string **position_ALARMON** = ”4”
- const string **position_SAMPLEINTERVAL** = ”60”
- readonly attribute ROposition **position**
- const string cmdPosition_DESCRIPTION = ”Commanded Mirror Position”
- const string **cmdPosition_UNITS** = ””
- readonly attribute RWposition **cmdPosition**

6.12.1 Detailed Description

APEX Cabin C Optics Interface

6.12.2 Member Data Documentation

const string apexOptics_C::Optics_C::cmdPosition_DESCRIPTION = ”Commanded Mirror Position”

Commanded mirror position (A, B, C)

const string apexOptics_C::Optics_C::position_DESCRIPTION = ”Mirror Position”

Mirror position (A, B, C, MOVING, UNDEFINED)

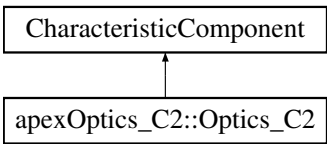
const string apexOptics_C::Optics_C::state_DESCRIPTION = ”State”

Actual State of the Cabin C Optics (INITIALIZE, ENABLED, DISABLED, SHUTDOWN, FAULTED)
The documentation for this interface was generated from the following file:

- apexOptics_C.idl

6.13 apexOptics_C2::Optics_C2 Interface Reference

`import"apexOptics_C2.idl";`
Inheritance diagram for apexOptics_C2::Optics_C2:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string position_DESCRIPTION = ”Mirror Position”
- const string **position_UNITS** = ””
- const string **position_ALARMON** = ”3”
- const string **position_SAMPLEINTERVAL** = ”60”
- readonly attribute ROposition **position**
- const string cmdPosition_DESCRIPTION = ”Commanded Mirror Position”
- const string **cmdPosition_UNITS** = ””
- readonly attribute RWposition **cmdPosition**

6.13.1 Detailed Description

APEX Cabin C2 Optics Interface

6.13.2 Member Data Documentation

const string apexOptics_C2::Optics_C2::cmdPosition_DESCRIPTION = ”Commanded Mirror Position”

Commanded mirror position (FACILITY, PI)

const string apexOptics_C2::Optics_C2::position_DESCRIPTION = ”Mirror Position”

Mirror position (FACILITY, PI, MOVING, UNDEFINED)

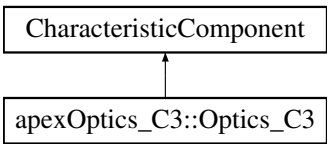
const string apexOptics_C2::Optics_C2::state_DESCRIPTION = ”State”

Actual State of the Cabin C2 Optics (INITIALIZE, ENABLED, DISABLED, SHUTDOWN, FAULTED)
The documentation for this interface was generated from the following file:

- apexOptics_C2.idl

6.14 apexOptics_C3::Optics_C3 Interface Reference

`import"apexOptics_C3.idl";`
Inheritance diagram for apexOptics_C3::Optics_C3:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string position_DESCRIPTION = ”Mirror Position”
- const string **position_UNITS** = ””
- const string **position_ALARMON** = ”4”
- const string **position_SAMPLEINTERVAL** = ”60”
- readonly attribute ROposition **position**
- const string cmdPosition_DESCRIPTION = ”Commanded Mirror Position”
- const string **cmdPosition_UNITS** = ””
- readonly attribute RWposition **cmdPosition**

6.14.1 Detailed Description

APEX Cabin C3 Optics Interface

6.14.2 Member Data Documentation

const string apexOptics_C3::Optics_C3::cmdPosition_DESCRIPTION = ”Commanded Mirror Position”

Commanded mirror position (STOWED, RX1, RX2)

const string apexOptics_C3::Optics_C3::position_DESCRIPTION = ”Mirror Position”

Mirror position (STOWED, RX1, RX2, MOVING, UNDEFINED)

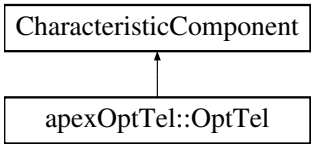
const string apexOptics_C3::Optics_C3::state_DESCRIPTION = ”State”

Actual State of the Cabin C3 Optics (INITIALIZE, ENABLED, DISABLED, SHUTDOWN, FAULTED)
The documentation for this interface was generated from the following file:

- apexOptics_C3.idl

6.15 apexOptTel::OptTel Interface Reference

`import"apexOptTel.idl";`
Inheritance diagram for apexOptTel::OptTel:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string lid_DESCRIPTION = ”Lid State”
- const string **lid_UNITS** = ””
- const string **lid_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlidStates **lid**
- const string cmdLid_DESCRIPTION = ”Commanded Lid State”
- const string **cmdLid_UNITS** = ””
- readonly attribute RWlidStates **cmdLid**
- const string filterWheel_DESCRIPTION = ”Filter Wheel State”
- const string **filterWheel_UNITS** = ””
- const string **filterWheel_SAMPLEINTERVAL** = ”60”
- readonly attribute ROfilterWheelStates **filterWheel**
- const string cmdFilterWheel_DESCRIPTION = ”Commanded Filter Wheel State”
- const string **cmdFilterWheel_UNITS** = ””
- readonly attribute RWfilterWheelStates **cmdFilterWheel**
- const string focus_DESCRIPTION = ”Focus Position”
- const string **focus_UNITS** = ”mm”
- const string **focus_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **focus**
- const string cmdFocus_DESCRIPTION = ”Commanded Focus Position”
- const string **cmdFocus_UNITS** = ”mm”
- readonly attribute ACS::RWdouble **cmdFocus**

6.15.1 Detailed Description

APEX Optical Pointing Telescope Interface

6.15.2 Member Function Documentation

void apexOptTel::OptTel::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.

Parameters

cb	callback when completed
desc	description

void apexOptTel::OptTel::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

6.15.3 Member Data Documentation

const string apexOptTel::OptTel::cmdFilterWheel_DESCRIPTION = "Commanded Filter Wheel State"

Filter wheel (R: (625 nm red) I: (770 nm extreme red) Z: (950 nm infrared) FREE: (no filter)

const string apexOptTel::OptTel::cmdFocus_DESCRIPTION = "Commanded Focus Position"

Focus position in mm

const string apexOptTel::OptTel::cmdLid_DESCRIPTION = "Commanded Lid State"

Lid drive (CLOSE, OPEN)

const string apexOptTel::OptTel::filterWheel_DESCRIPTION = "Filter Wheel State"

Filter wheel (R: (625 nm red) I: (770 nm extreme red) Z: (950 nm infrared) FREE: (no filter)

const string apexOptTel::OptTel::focus_DESCRIPTION = "Focus Position"

Focus position in mm

const string apexOptTel::OptTel::lid_DESCRIPTION = "Lid State"

Lid drive (CLOSE, OPEN)

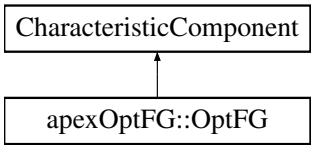
const string apexOptTel::OptTel::state_DESCRIPTION = "State"

Actual State of the Optical Pointing Telescope
The documentation for this interface was generated from the following file:

- apexOptTel.idl

6.16 apexOptFG::OptFG Interface Reference

`import"apexOptFG.idl";`
Inheritance diagram for apexOptFG::OptFG:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void takeFrame (in ACS::CBvoid cb, in ACS::CBDescIn desc)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string status_DESCRIPTION = ”Status”
- const string **status_UNITS** = ””
- const string **status_SAMPLEINTERVAL** = ”60”
- const string **status_BITDESCRIPTION** = ”Idle, Waiting for trigger, Exposing, Downloading, Line ready, Image ready, Flushing BIR, Error”
- const string **status_WHENCLEARED** = ”N,N,N,N,N,N,N,N”
- const string **status_WHENSET** = ”S,S,S,S,S,S,S,F”
- readonly attribute ACS::ROpattern **status**
- const string integrationTime_DESCRIPTION = ”Integration Time”
- const string **integrationTime_UNITS** = ”s”
- const string **integrationTime_FORMAT** = ”%.3f”
- const string **integrationTime_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **integrationTime**
- const string cmdIntegrationTime_DESCRIPTION = ”Commanded Integration Time”
- const string **cmdIntegrationTime_UNITS** = ”s”
- const string **cmdIntegrationTime_FORMAT** = ”%.3f”
- readonly attribute ACS::RWdouble **cmdIntegrationTime**
- const string shutter_DESCRIPTION = ”Camera Shutter State”
- const string **shutter_UNITS** = ””
- const string **shutter_SAMPLEINTERVAL** = ”60”
- readonly attribute ROshutterStates **shutter**
- const string cmdShutter_DESCRIPTION = ”Commanded Camera Shutter State”
- const string **cmdShutter_UNITS** = ””
- readonly attribute RWshutterStates **cmdShutter**
- const string sizeX_DESCRIPTION = ”Size X”
- const string **sizeX_UNITS** = ””
- readonly attribute ACS::ROlong **sizeX**
- const string sizeY_DESCRIPTION = ”Size Y”
- const string **sizeY_UNITS** = ””
- readonly attribute ACS::ROlong **sizeY**
- const string scaleX_DESCRIPTION = ”Scale X”

- const string **scaleX_UNITS** = ”arcsec”
- readonly attribute ACS::ROdouble **scaleX**
- const string scaleY_DESCRIPTION = ”Scale Y”
- const string **scaleY_UNITS** = ”arcsec”
- readonly attribute ACS::ROdouble **scaleY**
- const string angle_DESCRIPTION = ”Frame Grabber Angle”
- const string **angle_UNITS** = ”deg”
- readonly attribute ACS::ROdouble **angle**
- const string CCDTemperature_DESCRIPTION = ”CCD Temperature”
- const string **CCDTemperature_UNITS** = ”degC”
- const string **CCDTemperature_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **CCDTemperature**

6.16.1 Detailed Description

APEX Optical Pointing Frame Grabber Interface

6.16.2 Member Function Documentation

void apexOptFG::OptFG::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexOptFG::OptFG::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

void apexOptFG::OptFG::takeFrame (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Take a frame with the given integration time and send it out via TCP.
Parameters

cb	callback when completed
desc	description

6.16.3 Member Data Documentation

const string apexOptFG::OptFG::angle_DESCRIPTION = ”Frame Grabber Angle”

Rotation angle of frame grabber relative to the horizon

const string apexOptFG::OptFG::CCDTemperature_DESCRIPTION = ”CCD Temperature”

CCD temperature [degC]

const string apexOptFG::OptFG::cmdIntegrationTime_DESCRIPTION = ”Commanded Integration Time”

Commanded intgeration time per frame (seconds)

const string apexOptFG::OptFG::cmdShutter_DESCRIPTION = ”Commanded Camera Shutter State”

Camera shutter (CLOSE, OPEN)

const string apexOptFG::OptFG::integrationTime_DESCRIPTION = ”Integration Time”

Integration time per frame (seconds)

const string apexOptFG::OptFG::scaleX_DESCRIPTION = ”Scale X”

Size of one pixel on sky in x direction

const string apexOptFG::OptFG::scaleY_DESCRIPTION = ”Scale Y”

Size of one pixel on sky in y direction

const string apexOptFG::OptFG::shutter_DESCRIPTION = ”Camera Shutter State”

Camera shutter (CLOSE, OPEN)

const string apexOptFG::OptFG::sizeX_DESCRIPTION = ”Size X”

Number of pixels in x direction

const string apexOptFG::OptFG::sizeY_DESCRIPTION = ”Size Y”

Number of pixels in y direction

const string apexOptFG::OptFG::state_DESCRIPTION = ”State”

Actual State of the Optical Pointing Frame Grabber

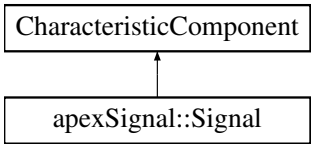
const string apexOptFG::OptFG::status_DESCRIPTION = ”Status”

Frame grabber state machine
The documentation for this interface was generated from the following file:

- apexOptFG.idl

6.17 apexSignal::Signal Interface Reference

`import"apexSignal.idl";`
Inheritance diagram for apexSignal::Signal:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”30”
- readonly attribute ROstatus **state**
- const string red_DESCRIPTION = ”Red Light Status”
- const string **red_UNITS** = ””
- const string **red_SAMPLEINTERVAL** = ”30”
- readonly attribute ROlightMode **red**
- const string cmdRed_DESCRIPTION = ”Commanded Red Light Status”
- const string **cmdRed_UNITS** = ””
- readonly attribute RWlightMode **cmdRed**
- const string yellow_DESCRIPTION = ”Yellow Light Status”
- const string **yellowred_UNITS** = ””
- const string **yellow_SAMPLEINTERVAL** = ”30”
- readonly attribute ROlightMode **yellow**
- const string cmdYellow_DESCRIPTION = ”Commanded Yellow Light Status”
- const string **cmdYellow_UNITS** = ””
- readonly attribute RWlightMode **cmdYellow**
- const string green_DESCRIPTION = ”Green Light Status”
- const string **green_UNITS** = ””
- const string **green_SAMPLEINTERVAL** = ”30”
- readonly attribute ROlightMode **green**
- const string cmdGreen_DESCRIPTION = ”Commanded Green Light Status”
- const string **cmdGreen_UNITS** = ””
- readonly attribute RWlightMode **cmdGreen**
- const string buzzer_DESCRIPTION = ”Buzzer Status”
- const string **buzzer_UNITS** = ””
- const string **buzzer_SAMPLEINTERVAL** = ”30”
- readonly attribute RObuzzerMode **buzzer**
- const string cmdBuzzer_DESCRIPTION = ”Commanded Buzzer Status”
- const string **cmdBuzzer_UNITS** = ””
- readonly attribute RWbuzzerMode **cmdBuzzer**

6.17.1 Detailed Description

APEX Signal Component Interface

6.17.2 Member Data Documentation

const string apexSignal::Signal::buzzer_DESCRIPTION = ”Buzzer Status”

Buzzer status

const string apexSignal::Signal::cmdBuzzer_DESCRIPTION = ”Commanded Buzzer Status”

Commanded buzzer status

const string apexSignal::Signal::cmdGreen_DESCRIPTION = ”Commanded Green Light Status”

Commanded green light status

const string apexSignal::Signal::cmdRed_DESCRIPTION = ”Commanded Red Light Status”

Commanded red light status

const string apexSignal::Signal::cmdYellow_DESCRIPTION = ”Commanded Yellow Light Status”

Commanded yellow light status

const string apexSignal::Signal::green_DESCRIPTION = ”Green Light Status”

Green light status

const string apexSignal::Signal::red_DESCRIPTION = ”Red Light Status”

Red light status

const string apexSignal::Signal::state_DESCRIPTION = ”State”

Actual State of the Signal Component (INITIALIZE, ENABLED, DISABLED, SHUTDOWN, FAULTED)

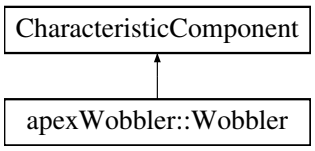
const string apexSignal::Signal::yellow_DESCRIPTION = ”Yellow Light Status”

Yellow light status
The documentation for this interface was generated from the following file:

- apexSignal.idl

6.18 apexWobbler::Wobbler Interface Reference

`import"apexWobbler.idl";`
Inheritance diagram for apexWobbler::Wobbler:



Public Member Functions

- void configure (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long start () raises (ACSErr::ACSException)
- long stop () raises (ACSErr::ACSException)
- long stow () raises (ACSErr::ACSException)
- long unstow () raises (ACSErr::ACSException)
- long reset () raises (ACSErr::ACSException)
- long clear_fault () raises (ACSErr::ACSException)
- long set_fans (in long fan_state) raises (ACSErr::ACSException)

Public Attributes

- readonly attribute ROstatus state
- readonly attribute ACS::ROpattern statusSet1a
- readonly attribute ACS::ROpattern statusSet1b
- readonly attribute ACS::ROpattern statusSet2a
- readonly attribute ACS::ROlongSeq sdErrorID
- readonly attribute ACS::ROdouble amplitude
- readonly attribute ACS::RWdouble cmdAmplitude
- readonly attribute ACS::ROdouble zeroPosition
- readonly attribute ACS::RWdouble cmdZeroPosition
- readonly attribute ACS::ROdouble tolerance
- readonly attribute ACS::RWdouble cmdTolerance
- readonly attribute ACS::ROdouble positionTE0
- readonly attribute ACS::ROdouble positionTE4
- readonly attribute ACS::ROdouble positionTE8
- readonly attribute ACS::ROdouble positionTE12
- readonly attribute ACS::ROdouble positionTE16
- readonly attribute ACS::ROdouble positionTE20
- readonly attribute ACS::ROdouble positionTE24
- readonly attribute ACS::ROdouble positionTE28
- readonly attribute ACS::ROdouble positionTE32
- readonly attribute ACS::ROdouble positionTE36
- readonly attribute ACS::ROdouble positionTE40
- readonly attribute ACS::ROdouble positionTE44
- readonly attribute ACS::ROdouble SRMotorTemp
- readonly attribute ACS::ROdouble CWMotorTemp

6.18.1 Detailed Description

An ACS Component providing direct access to the wobbler motion control and monitoring. The access is achieved through the antenna vendor's Wobbler Interface Unit (WIU) via the CAN bus.

6.18.2 Member Function Documentation

long apexWobbler::Wobbler::clear_fault ( ) raises ACSErr::ACSException)

Clear all latched faults

void apexWobbler::Wobbler::configure (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Configures the wobbler, set the values for frequency, amplitude and zeroOffset acording to the values stored in cmd-Amplitude, cmdFrequency and cmdZeroPosition. THE WOBBLER HAS TO BE IN STOP/SHUTDOWN mode
Parameters

cb	callback when completed
desc	description

long apexWobbler::Wobbler::reset ( ) raises ACSErr::ACSException)

Reset the wobbler

long apexWobbler::Wobbler::set_fans (in long *fan_state*) raises ACSErr::ACSException)

Control fans

long apexWobbler::Wobbler::start ( ) raises ACSErr::ACSException)

Start the wobbler program.

long apexWobbler::Wobbler::stop ( ) raises ACSErr::ACSException)

Stop the wobbler program.

long apexWobbler::Wobbler::stow ( ) raises ACSErr::ACSException)

Stow the wobbler.

long apexWobbler::Wobbler::unstow ( ) raises ACSErr::ACSException)

Unstow the wobbler.

6.18.3 Member Data Documentation

readonly attribute ACS::ROdouble apexWobbler::Wobbler::amplitude

Amplitude (arcsec)

readonly attribute ACS::RWdouble apexWobbler::Wobbler::cmdAmplitude

Commanded amplitude (arcsec)

readonly attribute ACS::RWdouble apexWobbler::Wobbler::cmdTolerance

Commanded tolerance for blank signal generation (arcsec)

readonly attribute ACS::RWdouble apexWobbler::Wobbler::cmdZeroPosition

Commanded wobbler zero position (arcsec)

readonly attribute ACS::ROdouble apexWobbler::Wobbler::CWMotorTemp

Counterweight motor temperature (degC)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE0

wobbler position at last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE12

wobbler position 12ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE16

wobbler position 16ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE20

wobbler position 20ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE24

wobbler position 24ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE28

wobbler position 28ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE32

wobbler position 32ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE36

wobbler position 36ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE4

wobbler position 4ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE40

wobbler position 40ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE44

wobbler position 44ms before last TE (arcsec)

readonly attribute ACS::RODouble apexWobbler::Wobbler::positionTE8

wobbler position 8ms before last TE (arcsec)

readonly attribute ACS::ROlongSeq apexWobbler::Wobbler::sdErrorID

Servo drive error ID

readonly attribute ACS::RODouble apexWobbler::Wobbler::SRMotorTemp

Subreflector motor temperature (degC)

readonly attribute ROstatus apexWobbler::Wobbler::state

Actual State of the Wobbler

readonly attribute ACS::ROpattern apexWobbler::Wobbler::statusSet1a

Wobbler status set 1a

readonly attribute ACS::ROpattern apexWobbler::Wobbler::statusSet1b

Wobbler status set 1b

readonly attribute ACS::ROpattern apexWobbler::Wobbler::statusSet2a

Wobbler status set 2a

readonly attribute ACS::ROdouble apexWobbler::Wobbler::tolerance

Tolerance for blank signal generation (arcsec)

readonly attribute ACS::ROdouble apexWobbler::Wobbler::zeroPosition

Wobbler zero position (arcsec)
The documentation for this interface was generated from the following file:

- apexWobbler.idl

6.19 apexWobbler::PositionMessage Struct Reference

```
import"apexWobbler.idl";
```

Public Attributes

- double **wobblerPosTE0**
- double **wobblerPosTE4**
- double **wobblerPosTE8**
- double **wobblerPosTE12**
- double **wobblerPosTE16**
- double **wobblerPosTE20**
- double **wobblerPosTE24**
- double **wobblerPosTE28**
- double **wobblerPosTE32**
- double **wobblerPosTE36**
- double **wobblerPosTE40**
- double **wobblerPosTE44**
- ACS::Time **timeStamp**

6.19.1 Detailed Description

This structure holds a set of wobbler positions. A filled structure is sent via the CORBA Notification Channel during each Timing Event (TE) by the apexWobbler component.
The documentation for this struct was generated from the following file:

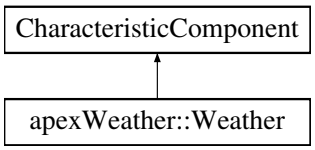
- apexWobbler.idl

Chapter 7

Environmental Devices

7.1 apexWeather::Weather Interface Reference

`import"apexWeather.idl";`
Inheritance diagram for apexWeather::Weather:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_SAMPLEINTERVAL** = ”60”
- const string **state_ALARMINTERVAL** = ”60”
- const string **state_ALARMON** = ”4”
- readonly attribute ROstatus **state**
- const string temperature_DESCRIPTION = ”Temperature”
- const string **temperature_UNITS** = ”degC”
- const string **temperature_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **temperature**
- const string dewPoint_DESCRIPTION = ”Dewpoint”
- const string **dewPoint_UNITS** = ”degC”
- const string **dewPoint_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **dewPoint**
- const string humidity_DESCRIPTION = ”Humidity”
- const string **humidity_UNITS** = ”percent”
- const string **humidity_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **humidity**
- const string pressure_DESCRIPTION = ”Pressure”
- const string **pressure_UNITS** = ”hPa”
- const string **pressure_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **pressure**

- const string windSpeed_DESCRIPTION = ”Wind Speed”
- const string **windSpeed_UNITS** = ”m/s”
- const string **windSpeed_SAMPLEINTERVAL** = ”60”
- const string **windSpeed_ALARMHIGH** = ”20.0”
- readonly attribute ACS::ROdouble **windSpeed**
- const string windDirection_DESCRIPTION = ”Wind Direction”
- const string **windDirection_UNITS** = ”deg”
- const string **windDirection_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **windDirection**
- const string controlBoxTemperature_DESCRIPTION = ”Control Box Temperature”
- const string **controlBoxTemperature_UNITS** = ”degC”
- const string **controlBoxTemperature_SAMPLEINTERVAL** = ”60”
- const string **controlBoxTemperature_ALARMINTERVAL** = ”60”
- const string **controlBoxTemperature_ALARMLOWON** = ”8.0”
- const string **controlBoxTemperature_ALARMLOWOFF** = ”9.0”
- const string **controlBoxTemperature_ALARMHIGHON** = ”30.0”
- const string **controlBoxTemperature_ALARMHIGHOFF** = ”29.0”
- readonly attribute ACS::ROdouble **controlBoxTemperature**

7.1.1 Detailed Description

APEX Weather Station Interface

7.1.2 Member Function Documentation

void apexWeather::Weather::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexWeather::Weather::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexWeather::Weather::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

7.1.3 Member Data Documentation

const string apexWeather::Weather::controlBoxTemperature_DESCRIPTION = ”Control Box Temperature”

Control box temperature (degrees Celsius) For equipment monitoring purposes.

const string apexWeather::Weather::dewPoint_DESCRIPTION = ”Dewpoint”

Dewpoint (degrees Celsius)

const string apexWeather::Weather::humidity_DESCRIPTION = ”Humidity”

Relative humidity (%)

const string apexWeather::Weather::pressure_DESCRIPTION = ”Pressure”

Pressure (hPa)

const string apexWeather::Weather::state_DESCRIPTION = ”State”

Actual State of the Weather Station

const string apexWeather::Weather::temperature_DESCRIPTION = ”Temperature”

Temperature (degrees Celsius)

const string apexWeather::Weather::windDirection_DESCRIPTION = ”Wind Direction”

Wind direction (degrees)

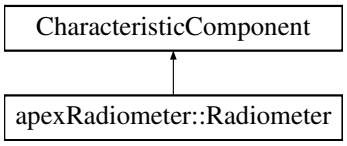
const string apexWeather::Weather::windSpeed_DESCRIPTION = ”Wind Speed”

Wind speed (m/s)
The documentation for this interface was generated from the following file:

- apexWeather.idl

7.2 apexRadiometer::Radiometer Interface Reference

`import"apexRadiometer.idl";`
Inheritance diagram for apexRadiometer::Radiometer:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)

Public Attributes

- const string state_DESCRIPTION = ”Radiometer State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string skyTB06_DESCRIPTION = ”TB @ 0.6 GHz”
- const string **skyTB06_UNITS** = ”K”
- const string **skyTB06_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **skyTB06**
- const string skyTB15_DESCRIPTION = ”TB @ 1.5 GHz”
- const string **skyTB15_UNITS** = ”K”
- const string **skyTB15_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **skyTB15**
- const string skyTB25_DESCRIPTION = ”TB @ 2.5 GHz”
- const string **skyTB25_UNITS** = ”K”
- const string **skyTB25_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **skyTB25**
- const string skyTB35_DESCRIPTION = ”TB @ 3.5 GHz”
- const string **skyTB35_UNITS** = ”K”
- const string **skyTB35_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **skyTB35**
- const string skyTB50_DESCRIPTION = ”TB @ 5.0 GHz”
- const string **skyTB50_UNITS** = ”K”
- const string **skyTB50_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **skyTB50**
- const string skyTB75_DESCRIPTION = ”TB @ 7.5 GHz”
- const string **skyTB75_UNITS** = ”K”
- const string **skyTB75_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **skyTB75**
- const string skyTB_DESCRIPTION = ”Brightness Temperatures”
- const string **skyTB_UNITS** = ”K”
- readonly attribute ACS::ROdoubleSeq **skyTB**

7.2.1 Detailed Description

APEX Radiometer Interface

7.2.2 Member Function Documentation

void apexRadiometer::Radiometer::off (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Move external mirror out of the beam.

Parameters

cb	callback when completed
desc	description

void apexRadiometer::Radiometer::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Move external mirror into beam.
Parameters

cb	callback when completed
desc	description

7.2.3 Member Data Documentation

const string apexRadiometer::Radiometer::skyTB06_DESCRIPTION = "TB @ 0.6 GHz"

Brightness temperatures the 0.6 GHz IF band (K)

const string apexRadiometer::Radiometer::skyTB15_DESCRIPTION = "TB @ 1.5 GHz"

Brightness temperatures the 1.5 GHz IF band (K)

const string apexRadiometer::Radiometer::skyTB25_DESCRIPTION = "TB @ 2.5 GHz"

Brightness temperatures the 2.5 GHz IF band (K)

const string apexRadiometer::Radiometer::skyTB35_DESCRIPTION = "TB @ 3.5 GHz"

Brightness temperatures the 3.5 GHz IF band (K)

const string apexRadiometer::Radiometer::skyTB50_DESCRIPTION = "TB @ 5.0 GHz"

Brightness temperatures the 5.0 GHz IF band (K)

const string apexRadiometer::Radiometer::skyTB75_DESCRIPTION = "TB @ 7.5 GHz"

Brightness temperatures the 7.5 GHz IF band (K)

const string apexRadiometer::Radiometer::skyTB_DESCRIPTION = "Brightness Temperatures"

Brightness temperatures in the 6 IF bands as vector (K)

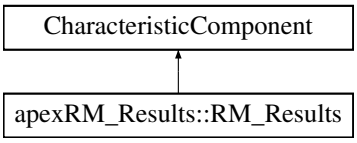
const string apexRadiometer::Radiometer::state_DESCRIPTION = "Radiometer State"

Actual State of the Radiometer
The documentation for this interface was generated from the following file:

- apexRadiometer.idl

7.3 apexRM_Results::RM_Results Interface Reference

```
import"apexRM_Results.idl";
```
Inheritance diagram for apexRM_Results::RM_Results:



Public Attributes

- const string tau183_DESCRIPTION = ”Tau @ 183 GHz”
- const string **tau183_UNITS** = ””
- const string **tau183_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **tau183**
- const string pwv_DESCRIPTION = ”Precipitable Water Vapor”
- const string **pwv_UNITS** = ”mm”
- const string **pwv_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **pwv**

7.3.1 Detailed Description

APEX Radiometer Results Interface

7.3.2 Member Data Documentation

const string apexRM_Results::RM_Results::pwv_DESCRIPTION = ”Precipitable Water Vapor”

Atmospheric Water Vapor Content

const string apexRM_Results::RM_Results::tau183_DESCRIPTION = ”Tau @ 183 GHz”

Atmospheric Opacity at 183 GHz
The documentation for this interface was generated from the following file:

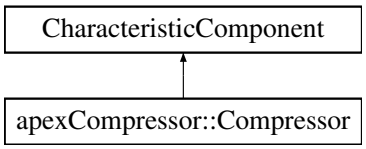
- apexRM_Results.idl

Chapter 8

Infrastructure Devices

8.1 apexCompressor::Compressor Interface Reference

`import"apexCompressor.idl";`
Inheritance diagram for apexCompressor::Compressor:



Public Member Functions

- long on () raises (ACSErr::ACSException)
- long off () raises (ACSErr::ACSException)
- long coldHeadOn () raises (ACSErr::ACSException)
- long coldHeadOff () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string supplyPressure_DESCRIPTION = ”Helium Supply Pressure”
- const string **supplyPressure_UNITS** = ”bar”
- const string **supplyPressure_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **supplyPressure**
- const string returnPressure_DESCRIPTION = ”Helium Return Pressure”
- const string **returnPressure_UNITS** = ”bar”
- const string **returnPressure_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **returnPressure**
- const string supplyJTPressure_DESCRIPTION = ”Helium Supply JT-Pressure”
- const string **supplyJTPressure_UNITS** = ”bar”
- const string **supplyJTPressure_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **supplyJTPressure**
- const string returnJTPressure_DESCRIPTION = ”Helium Return JT-Pressure”
- const string **returnJTPressure_UNITS** = ”bar”
- const string **returnJTPressure_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **returnJTPressure**

8.1.1 Detailed Description

APEX Compressor Interface

8.1.2 Member Function Documentation

long apexCompressor::Compressor::coldHeadOff ( ) raises ACSErr::ACSException)

Turn cold head off.

long apexCompressor::Compressor::coldHeadOn ( ) raises ACSErr::ACSException)

Turn cold head on.

long apexCompressor::Compressor::off ( ) raises ACSErr::ACSException)

Turn compressor off.

long apexCompressor::Compressor::on ( ) raises ACSErr::ACSException)

Turn compressor on.

8.1.3 Member Data Documentation

const string apexCompressor::Compressor::returnJTPressure_DESCRIPTION = ”Helium Return JT-Pressure”

Helium return JT pressure

const string apexCompressor::Compressor::returnPressure_DESCRIPTION = ”Helium Return Pressure”

Helium return pressure

const string apexCompressor::Compressor::state_DESCRIPTION = ”State”

Actual State of the compressor

const string apexCompressor::Compressor::supplyJTPressure_DESCRIPTION = ”Helium Supply JT-Pressure”

Helium supply JT pressure

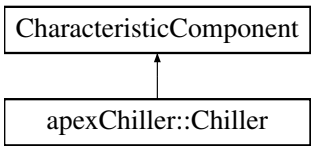
const string apexCompressor::Compressor::supplyPressure_DESCRIPTION = ”Helium Supply Pressure”

Helium supply pressure
The documentation for this interface was generated from the following file:

- apexCompressor.idl

8.2 apexChiller::Chiller Interface Reference

`import"apexChiller.idl";`
Inheritance diagram for apexChiller::Chiller:



Public Member Functions

- long on () raises (ACSErr::ACSException)
- long off () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string supplyTemp_DESCRIPTION = ”Water Supply Temperature”
- const string **supplyTemp_UNITS** = ”degC”
- const string **supplyTemp_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **supplyTemp**
- const string returnTemp_DESCRIPTION = ”Water Return Temperature”
- const string **returnTemp_UNITS** = ”degC”
- const string **returnTemp_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **returnTemp**
- const string waterFlow_DESCRIPTION = ”Water Flow Velocity”
- const string **waterFlow_UNITS** = ”l/min”
- const string **waterFlow_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **waterFlow**
- const string waterPressure_DESCRIPTION = ”Water Pressure”
- const string **waterPressure_UNITS** = ”bar”
- const string **waterPressure_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **waterPressure**
- const string waterLevel_DESCRIPTION = ”Water Level”
- const string **waterLevel_UNITS** = ”percent”
- const string **waterLevel_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **waterLevel**

8.2.1 Detailed Description

APEX Chiller Interface

8.2.2 Member Function Documentation

long apexChiller::Chiller::off ( ) raises ACSErr::ACSException)

Turn off any parts of the instrument for which power is under software control.

long apexChiller::Chiller::on ( ) raises ACSErr::ACSException)

Turn on any parts of the instrument for which power is under software control.

8.2.3 Member Data Documentation

const string apexChiller::Chiller::returnTemp_DESCRIPTION = "Water Return Temperature"

Water return temperature

const string apexChiller::Chiller::state_DESCRIPTION = "State"

State of the chiller system

const string apexChiller::Chiller::supplyTemp_DESCRIPTION = "Water Supply Temperature"

Water supply temperature

const string apexChiller::Chiller::waterFlow_DESCRIPTION = "Water Flow Velocity"

Water flow velocity

const string apexChiller::Chiller::waterLevel_DESCRIPTION = "Water Level"

Water level

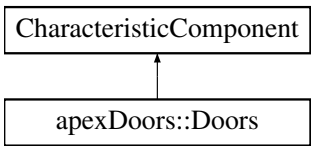
const string apexChiller::Chiller::waterPressure_DESCRIPTION = "Water Pressure"

Water pressure
The documentation for this interface was generated from the following file:

- apexChiller.idl

8.3 apexDoors::Doors Interface Reference

```
import"apexDoors.idl";
```
Inheritance diagram for apexDoors::Doors:



Public Attributes

- const string cabinADoor_DESCRIPTION = ”Cabin A Door”
- const string **cabinADoor_UNITS** = ””
- const string **cabinADoor_ALARMON** = ”1”
- const string **cabinADoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **cabinADoor**
- const string cabinBDoor_DESCRIPTION = ”Cabin B Door”
- const string **cabinBDoor_UNITS** = ””
- const string **cabinBDoor_ALARMON** = ”1”
- const string **cabinBDoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **cabinBDoor**
- const string cabinCDoor_DESCRIPTION = ”Cabin C Door”
- const string **cabinCDoor_UNITS** = ””
- const string **cabinCDoor_ALARMON** = ”1”
- const string **cabinCDoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **cabinCDoor**
- const string containerFrontDoor_DESCRIPTION = ”Container Front Door”
- const string **containerFrontDoor_UNITS** = ””
- const string **containerFrontDoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **containerFrontDoor**
- const string containerRearDoor_DESCRIPTION = ”Container Rear Door”
- const string **containerRearDoor_UNITS** = ””
- const string **containerRearDoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **containerRearDoor**
- const string containerRightDoor_DESCRIPTION = ”Main Container Door”
- const string **containerRightDoor_UNITS** = ””
- const string **containerRightDoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **containerRightDoor**
- const string rightPlatformDoor_DESCRIPTION = ”Right Platform Door”
- const string **rightPlatformDoor_UNITS** = ””
- const string **rightPlatformDoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **rightPlatformDoor**
- const string leftPlatformDoor_DESCRIPTION = ”Left Platform Door”
- const string **leftPlatformDoor_UNITS** = ””
- const string **leftPlatformDoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **leftPlatformDoor**
- const string groundDoor_DESCRIPTION = ”Pedestal Door”
- const string **groundDoor_UNITS** = ””
- const string **groundDoor_ALARMON** = ”1”
- const string **groundDoor_SAMPLEINTERVAL** = ”180”
- readonly attribute RODoorStatus **groundDoor**

8.3.1 Detailed Description

APEX Doors State Interface

8.3.2 Member Data Documentation

const string apexDoors::Doors::cabinADoor_DESCRIPTION = ”Cabin A Door”

Cabin A door status

const string apexDoors::Doors::cabinBDoor_DESCRIPTION = ”Cabin B Door”

Cabin B door status

const string apexDoors::Doors::cabinCDoor_DESCRIPTION = ”Cabin C Door”

Cabin C door status

const string apexDoors::Doors::containerFrontDoor_DESCRIPTION = ”Container Front Door”

Container front door status

const string apexDoors::Doors::containerRearDoor_DESCRIPTION = ”Container Rear Door”

Container rear door status

const string apexDoors::Doors::containerRightDoor_DESCRIPTION = ”Main Container Door”

Container right door status

const string apexDoors::Doors::groundDoor_DESCRIPTION = ”Pedestal Door”

Ground door status

const string apexDoors::Doors::leftPlatformDoor_DESCRIPTION = ”Left Platform Door”

Left platform door status

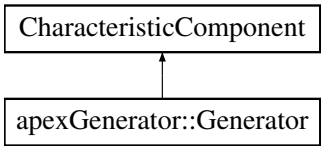
const string apexDoors::Doors::rightPlatformDoor_DESCRIPTION = ”Right Platform Door”

Right platform door status
The documentation for this interface was generated from the following file:

- apexDoors.idl

8.4 apexGenerator::Generator Interface Reference

`import"apexGenerator.idl";`
Inheritance diagram for apexGenerator::Generator:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**

8.4.1 Detailed Description

APEX Main Generator Interface

8.4.2 Member Function Documentation

void apexGenerator::Generator::off (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexGenerator::Generator::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexGenerator::Generator::reset ( ) raises ACSErr::ACSException)

Set up the continuum backend according to the high-level properties that are currently defined. The embedded system must keep track of those values.
Parameters

cb	callback when completed

desc	description Hardware reset (restart system) and set defaults.

8.4.3 Member Data Documentation

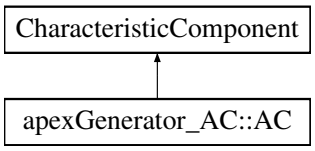
const string apexGenerator::Generator::state_DESCRIPTION = ”State”

Actual State of the IF
The documentation for this interface was generated from the following file:

- apexGenerator.idl

8.5 apexGenerator_AC::AC Interface Reference

```
import"apexGenerator_AC.idl";
```
Inheritance diagram for apexGenerator_AC::AC:



Public Attributes

- const string freq_DESCRIPTION = ”Frequency”
- const string **freq_UNITS** = ”Hz”
- const string **freq_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **freq**
- const string pf_DESCRIPTION = ”Power Factor”
- const string **pf_UNITS** = ”PF”
- const string **pf_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **pf**
- const string kva_DESCRIPTION = ”Apparent Power”
- const string **kva_UNITS** = ”KVA”
- const string **kva_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **kva**
- const string kw_DESCRIPTION = ”Real Power”
- const string **kw_UNITS** = ”KW”
- const string **kw_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **kw**
- const string kvar_DESCRIPTION = ”Reactive Power”
- const string **kvar_UNITS** = ”KVAR”
- const string **kvar_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **kvar**
- const string voltageAB_DESCRIPTION = ”AB line to line voltage”
- const string **voltageAB_UNITS** = ”V”
- const string **voltageAB_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **voltageAB**
- const string voltageBC_DESCRIPTION = ”BC line to line voltage”
- const string **voltageBC_UNITS** = ”V”
- const string **voltageBC_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **voltageBC**
- const string voltageCA_DESCRIPTION = ”CA line to line voltage”
- const string **voltageCA_UNITS** = ”V”
- const string **voltageCA_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **voltageCA**
- const string voltageA_DESCRIPTION = ”phase A line voltage”
- const string **voltageA_UNITS** = ”V”
- const string **voltageA_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **voltageA**
- const string voltageB_DESCRIPTION = ”phase B line voltage”
- const string **voltageB_UNITS** = ”V”
- const string **voltageB_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **voltageB**
- const string voltageC_DESCRIPTION = ”phase C line voltage”
- const string **voltageC_UNITS** = ”V”

- const string **voltageC_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **voltageC**
- const string currentA_DESCRIPTION = ”phase A line current”
- const string **currentA_UNITS** = ”A”
- const string **currentA_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **currentA**
- const string currentB_DESCRIPTION = ”phase B line current”
- const string **currentB_UNITS** = ”A”
- const string **currentB_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **currentB**
- const string currentC_DESCRIPTION = ”phase C line current”
- const string **currentC_UNITS** = ”A”
- const string **currentC_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **currentC**
- const string percentCurrentA_DESCRIPTION = ”phase A line percent current”
- const string **percentCurrentA_UNITS** = ”%”
- const string **percentCurrentA_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **percentCurrentA**
- const string percentCurrentB_DESCRIPTION = ”phase B line percent current”
- const string **percentCurrentB_UNITS** = ”%”
- const string **percentCurrentB_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **percentCurrentB**
- const string percentCurrentC_DESCRIPTION = ”phase C line percent current”
- const string **percentCurrentC_UNITS** = ”%”
- const string **percentCurrentC_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **percentCurrentC**

8.5.1 Detailed Description

APEX Main Generators AC Data Interface

8.5.2 Member Data Documentation

const string apexGenerator_AC::AC::currentA_DESCRIPTION = ”phase A line current”

phase-A current

const string apexGenerator_AC::AC::currentB_DESCRIPTION = ”phase B line current”

phase-B current

const string apexGenerator_AC::AC::currentC_DESCRIPTION = ”phase C line current”

phase-C current

const string apexGenerator_AC::AC::freq_DESCRIPTION = ”Frequency”

Frequency

const string apexGenerator_AC::AC::kva_DESCRIPTION = ”Apparent Power”

Apparent Power

const string apexGenerator_AC::AC::kvar_DESCRIPTION = ”Reactive Power”

Reactive Power

const string apexGenerator_AC::AC::kw_DESCRIPTION = ”Real Power”

Real Power

const string apexGenerator_AC::AC::percentCurrentA_DESCRIPTION = ”phase A line percent current”

phase-A percent current

const string apexGenerator_AC::AC::percentCurrentB_DESCRIPTION = ”phase B line percent current”

phase-B percent current

const string apexGenerator_AC::AC::percentCurrentC_DESCRIPTION = ”phase C line percent current”

phase-C percent current

const string apexGenerator_AC::AC::pf_DESCRIPTION = ”Power Factor”

Power factor

const string apexGenerator_AC::AC::voltageA_DESCRIPTION = ”phase A line voltage”

phase-A line voltage

const string apexGenerator_AC::AC::voltageAB_DESCRIPTION = ”AB line to line voltage”

AB line to line voltage

const string apexGenerator_AC::AC::voltageB_DESCRIPTION = ”phase B line voltage”

phase-B line voltage

const string apexGenerator_AC::AC::voltageBC_DESCRIPTION = ”BC line to line voltage”

BC line to line voltage

const string apexGenerator_AC::AC::voltageC_DESCRIPTION = ”phase C line voltage”

phase-C line voltage

const string apexGenerator_AC::AC::voltageCA_DESCRIPTION = ”CA line to line voltage”

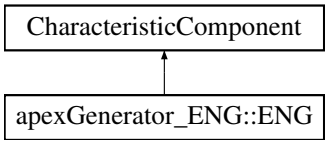
CA line to line voltage
The documentation for this interface was generated from the following file:

- apexGenerator_AC.idl

8.6 apexGenerator_ENG::ENG Interface Reference

```
import"apexGenerator_ENG.idl";
```

Inheritance diagram for apexGenerator_ENG::ENG:



Public Attributes

- const string batteryVoltage_DESCRIPTION = ”Battery Voltage”
- const string **batteryVoltage_UNITS** = ”V”
- const string **batteryVoltage_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **batteryVoltage**
- const string oilPressure_DESCRIPTION = ”Oil Pressure”
- const string **oilPressure_UNITS** = ”KPA”
- const string **oilPressure_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **oilPressure**
- const string oilTemp_DESCRIPTION = ”Oil Temperature”
- const string **oilTemp_UNITS** = ”K”
- const string **oilTemp_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **oilTemp**
- const string coolantTemp_DESCRIPTION = ”Coolant Temperature”
- const string **coolantTemp_UNITS** = ”K”
- const string **coolantTemp_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **coolantTemp**
- const string miscTemp1_DESCRIPTION = ”Misc Temperature 1”
- const string **miscTemp1_UNITS** = ”K”
- const string **miscTemp1_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **miscTemp1**
- const string miscTemp2_DESCRIPTION = ”Misc Temperature 2”
- const string **miscTemp2_UNITS** = ”K”
- const string **miscTemp2_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **miscTemp2**
- const string fuelRate_DESCRIPTION = ”Fuel Rate”
- const string **fuelRate_UNITS** = ”GPH”
- const string **fuelRate_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **fuelRate**
- const string engineRPM_DESCRIPTION = ”Engine RPM”
- const string **engineRPM_UNITS** = ”RPM”
- const string **engineRPM_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **engineRPM**
- const string engineStarts_DESCRIPTION = ”Engine starts”
- const string **engineStarts_UNITS** = ”RPM”
- const string **engineStarts_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **engineStarts**
- const string engineRunTime_DESCRIPTION = ”Engine runtime”
- const string **engineRunTime_UNITS** = ”S”
- const string **engineRunTime_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **engineRunTime**
- const string totalKWH_DESCRIPTION = ”Total KWH”
- const string **totalKWH_UNITS** = ”KWH”
- const string **totalKWH_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **totalKWH**

8.6.1 Detailed Description

APEX Main Generators AC Data Interface

8.6.2 Member Data Documentation

const string apexGenerator_ENG::ENG::batteryVoltage_DESCRIPTION = ”Battery Voltage”

Battery Voltage

const string apexGenerator_ENG::ENG::coolantTemp_DESCRIPTION = ”Coolant Temperature”

Coolant Temperature

const string apexGenerator_ENG::ENG::engineRPM_DESCRIPTION = ”Engine RPM”

Engine RPM

const string apexGenerator_ENG::ENG::engineRunTime_DESCRIPTION = ”Engine runtime”

Engine RunTime

const string apexGenerator_ENG::ENG::engineStarts_DESCRIPTION = ”Engine starts”

Engine starts

const string apexGenerator_ENG::ENG::fuelRate_DESCRIPTION = ”Fuel Rate”

Fuel Rate

const string apexGenerator_ENG::ENG::miscTemp1_DESCRIPTION = ”Misc Temperature 1”

Misc Temperature 1

const string apexGenerator_ENG::ENG::miscTemp2_DESCRIPTION = ”Misc Temperature 2”

Misc Temperature 2

const string apexGenerator_ENG::ENG::oilPressure_DESCRIPTION = ”Oil Pressure”

Oil Pressure

const string apexGenerator_ENG::ENG::oilTemp_DESCRIPTION = ”Oil Temperature”

Oil Temperature

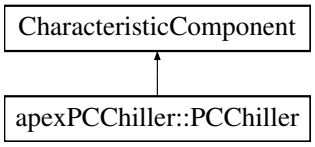
const string apexGenerator_ENG::ENG::totalKWH_DESCRIPTION = ”Total KWH”

Total KWH
The documentation for this interface was generated from the following file:

- apexGenerator_ENG.idl

8.7 apexPCChiller::PCChiller Interface Reference

`import"apexPCChiller.idl";`
Inheritance diagram for apexPCChiller::PCChiller:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string waterLevel_DESCRIPTION = ”Water Level”
- const string **waterLevel_UNITS** = ”percent”
- const string **waterLevel_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **waterLevel**

8.7.1 Detailed Description

APEX PC Chiller Interface

8.7.2 Member Data Documentation

const string apexPCChiller::PCChiller::state_DESCRIPTION = ”State”

State of the chiller system

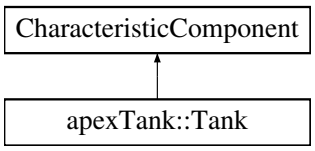
const string apexPCChiller::PCChiller::waterLevel_DESCRIPTION = ”Water Level”

Water level
The documentation for this interface was generated from the following file:

- apexPCChiller.idl

8.8 apexTank::Tank Interface Reference

`import"apexTank.idl";`
Inheritance diagram for apexTank::Tank:



Public Attributes

- const string waterLevel_DESCRIPTION = ”Water Level”
- const string **waterLevel_UNITS** = ”percent”
- const string **waterLevel_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **waterLevel**
- const string pumpState_DESCRIPTION = ”Pump State”
- const string **pumpState_UNITS** = ””
- const string **pumpState_SAMPLEINTERVAL** = ”60”
- readonly attribute ROlogical **pumpState**

8.8.1 Detailed Description

APEX Water Tank Interface

8.8.2 Member Data Documentation

const string apexTank::Tank::pumpState_DESCRIPTION = ”Pump State”

Pump state

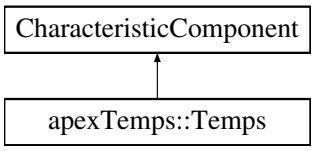
const string apexTank::Tank::waterLevel_DESCRIPTION = ”Water Level”

Water level
The documentation for this interface was generated from the following file:

- apexTank.idl

8.9 apexTemps::Temps Interface Reference

`import"apexTemps.idl";`
Inheritance diagram for apexTemps::Temps:



Public Attributes

- const string rack1_DESCRIPTION = ”Rack 1 Temperature”
- const string **rack1_UNITS** = ”degC”
- const string **rack1_SAMPLEINTERVAL** = ”60”
- const string **rack1_ALARMINTERVAL** = ”60”
- const string **rack1_ALARMLOWON** = ”15.0”
- const string **rack1_ALARMLOWOFF** = ”16.0”
- const string **rack1_ALARMHIGHON** = ”21.0”
- const string **rack1_ALARMHIGHOFF** = ”20.0”
- readonly attribute ACS::ROdouble **rack1**
- const string rack2_DESCRIPTION = ”Rack 2 Temperature”
- const string **rack2_UNITS** = ”degC”
- const string **rack2_SAMPLEINTERVAL** = ”60”
- const string **rack2_ALARMINTERVAL** = ”60”
- const string **rack2_ALARMLOWON** = ”15.0”
- const string **rack2_ALARMLOWOFF** = ”16.0”
- const string **rack2_ALARMHIGHON** = ”21.0”
- const string **rack2_ALARMHIGHOFF** = ”20.0”
- readonly attribute ACS::ROdouble **rack2**
- const string rack3_DESCRIPTION = ”Rack 3 Temperature”
- const string **rack3_UNITS** = ”degC”
- const string **rack3_SAMPLEINTERVAL** = ”60”
- const string **rack3_ALARMINTERVAL** = ”60”
- const string **rack3_ALARMLOWON** = ”15.0”
- const string **rack3_ALARMLOWOFF** = ”16.0”
- const string **rack3_ALARMHIGHON** = ”21.0”
- const string **rack3_ALARMHIGHOFF** = ”20.0”
- readonly attribute ACS::ROdouble **rack3**
- const string rack4_DESCRIPTION = ”Rack 4 Temperature”
- const string **rack4_UNITS** = ”degC”
- const string **rack4_SAMPLEINTERVAL** = ”60”
- const string **rack4_ALARMLOWON** = ”15.0”
- const string **rack4_ALARMLOWOFF** = ”16.0”
- const string **rack4_ALARMHIGHON** = ”21.0”
- const string **rack4_ALARMHIGHOFF** = ”20.0”
- readonly attribute ACS::ROdouble **rack4**
- const string rack5_DESCRIPTION = ”Rack 5 Temperature”
- const string **rack5_UNITS** = ”degC”
- const string **rack5_SAMPLEINTERVAL** = ”60”
- const string **rack5_ALARMLOWON** = ”15.0”
- const string **rack5_ALARMLOWOFF** = ”16.0”
- const string **rack5_ALARMHIGHON** = ”21.0”
- const string **rack5_ALARMHIGHOFF** = ”20.0”

- readonly attribute ACS::ROdouble **rack5**
- const string rack6_DESCRIPTION = ”Rack 6 Temperature”
- const string **rack6_UNITS** = ”degC”
- const string **rack6_SAMPLEINTERVAL** = ”60”
- const string **rack6_ALARMLOWON** = ”15.0”
- const string **rack6_ALARMLOWOFF** = ”16.0”
- const string **rack6_ALARMHIGHON** = ”21.0”
- const string **rack6_ALARMHIGHOFF** = ”20.0”
- readonly attribute ACS::ROdouble **rack6**
- const string macsRack1_DESCRIPTION = ”MACS Rack 1 Temperature”
- const string **macsRack1_UNITS** = ”degC”
- const string **macsRack1_SAMPLEINTERVAL** = ”60”
- const string **macsRack1_ALARMLOWON** = ”15.0”
- const string **macsRack1_ALARMLOWOFF** = ”16.0”
- const string **macsRack1_ALARMHIGHON** = ”21.0”
- const string **macsRack1_ALARMHIGHOFF** = ”20.0”
- readonly attribute ACS::ROdouble **macsRack1**
- const string macsRack2_DESCRIPTION = ”MACS Rack 2 Temperature”
- const string **macsRack2_UNITS** = ”degC”
- const string **macsRack2_SAMPLEINTERVAL** = ”60”
- const string **macsRack2_ALARMLOWON** = ”15.0”
- const string **macsRack2_ALARMLOWOFF** = ”16.0”
- const string **macsRack2_ALARMHIGHON** = ”21.0”
- const string **macsRack2_ALARMHIGHOFF** = ”20.0”
- readonly attribute ACS::ROdouble **macsRack2**
- const string macsRack3_DESCRIPTION = ”MACS Rack 3 Temperature”
- const string **macsRack3_UNITS** = ”degC”
- const string **macsRack3_SAMPLEINTERVAL** = ”60”
- const string **macsRack3_ALARMLOWON** = ”15.0”
- const string **macsRack3_ALARMLOWOFF** = ”16.0”
- const string **macsRack3_ALARMHIGHON** = ”21.0”
- const string **macsRack3_ALARMHIGHOFF** = ”20.0”
- readonly attribute ACS::ROdouble **macsRack3**
- const string cabinARack1_DESCRIPTION = ”Cabin A Rack 1 Temperature”
- const string **cabinARack1_UNITS** = ”degC”
- const string **cabinARack1_SAMPLEINTERVAL** = ”60”
- const string **cabinARack1_ALARMINTERVAL** = ”60”
- const string **cabinARack1_ALARMLOWON** = ”11.0”
- const string **cabinARack1_ALARMLOWOFF** = ”12.0”
- const string **cabinARack1_ALARMHIGHON** = ”15.0”
- const string **cabinARack1_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinARack1**
- const string cabinARack2_DESCRIPTION = ”Cabin A Rack 2 Temperature”
- const string **cabinARack2_UNITS** = ”degC”
- const string **cabinARack2_SAMPLEINTERVAL** = ”60”
- const string **cabinARack2_ALARMLOWON** = ”11.0”
- const string **cabinARack2_ALARMLOWOFF** = ”12.0”
- const string **cabinARack2_ALARMHIGHON** = ”15.0”
- const string **cabinARack2_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinARack2**
- const string cabinA3_DESCRIPTION = ”Cabin A Temperature Sensor 3”
- const string **cabinA3_UNITS** = ”degC”
- const string **cabinA3_SAMPLEINTERVAL** = ”60”
- const string **cabinA3_ALARMINTERVAL** = ”60”

- const string **cabinA3_ALARMLOWON** = ”11.0”
- const string **cabinA3_ALARMLOWOFF** = ”12.0”
- const string **cabinA3_ALARMHIGHON** = ”15.0”
- const string **cabinA3_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinA3**
- const string cabinA4_DESCRIPTION = ”Cabin A Temperature Sensor 4”
- const string **cabinA4_UNITS** = ”degC”
- const string **cabinA4_SAMPLEINTERVAL** = ”60”
- const string **cabinA4_ALARMLOWON** = ”11.0”
- const string **cabinA4_ALARMLOWOFF** = ”12.0”
- const string **cabinA4_ALARMHIGHON** = ”15.0”
- const string **cabinA4_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinA4**
- const string cabinA5_DESCRIPTION = ”Cabin A Temperature Sensor 5”
- const string **cabinA5_UNITS** = ”degC”
- const string **cabinA5_SAMPLEINTERVAL** = ”60”
- const string **cabinA5_ALARMLOWON** = ”11.0”
- const string **cabinA5_ALARMLOWOFF** = ”12.0”
- const string **cabinA5_ALARMHIGHON** = ”15.0”
- const string **cabinA5_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinA5**
- const string cabinA6_DESCRIPTION = ”Cabin A Temperature Sensor 6”
- const string **cabinA6_UNITS** = ”degC”
- const string **cabinA6_SAMPLEINTERVAL** = ”60”
- const string **cabinA6_ALARMLOWON** = ”11.0”
- const string **cabinA6_ALARMLOWOFF** = ”12.0”
- const string **cabinA6_ALARMHIGHON** = ”15.0”
- const string **cabinA6_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinA6**
- const string cabinBIFRack_DESCRIPTION = ”Cabin B IF Rack Temperature”
- const string **cabinBIFRack_UNITS** = ”degC”
- const string **cabinBIFRack_SAMPLEINTERVAL** = ”60”
- const string **cabinBIFRack_ALARMLOWON** = ”11.0”
- const string **cabinBIFRack_ALARMLOWOFF** = ”12.0”
- const string **cabinBIFRack_ALARMHIGHON** = ”15.0”
- const string **cabinBIFRack_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinBIFRack**
- const string cabinB2_DESCRIPTION = ”Cabin B Temperature Sensor 2”
- const string **cabinB2_UNITS** = ”degC”
- const string **cabinB2_SAMPLEINTERVAL** = ”60”
- const string **cabinB2_ALARMINTERVAL** = ”60”
- const string **cabinB2_ALARMLOWON** = ”11.0”
- const string **cabinB2_ALARMLOWOFF** = ”12.0”
- const string **cabinB2_ALARMHIGHON** = ”15.0”
- const string **cabinB2_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinB2**
- const string cabinB3_DESCRIPTION = ”Cabin B Temperature Sensor 3”
- const string **cabinB3_UNITS** = ”degC”
- const string **cabinB3_SAMPLEINTERVAL** = ”60”
- const string **cabinB3_ALARMLOWON** = ”11.0”
- const string **cabinB3_ALARMLOWOFF** = ”12.0”
- const string **cabinB3_ALARMHIGHON** = ”15.0”
- const string **cabinB3_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinB3**

- const string cabinB4_DESCRIPTION = ”Cabin B Temperature Sensor 4”
- const string **cabinB4_UNITS** = ”degC”
- const string **cabinB4_SAMPLEINTERVAL** = ”60”
- const string **cabinB4_ALARMLOWON** = ”11.0”
- const string **cabinB4_ALARMLOWOFF** = ”12.0”
- const string **cabinB4_ALARMHIGHON** = ”15.0”
- const string **cabinB4_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinB4**
- const string cabinB5_DESCRIPTION = ”Cabin B Temperature Sensor 5”
- const string **cabinB5_UNITS** = ”degC”
- const string **cabinB5_SAMPLEINTERVAL** = ”60”
- const string **cabinB5_ALARMLOWON** = ”11.0”
- const string **cabinB5_ALARMLOWOFF** = ”12.0”
- const string **cabinB5_ALARMHIGHON** = ”15.0”
- const string **cabinB5_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinB5**
- const string cabinB6_DESCRIPTION = ”Cabin B Temperature Sensor 6”
- const string **cabinB6_UNITS** = ”degC”
- const string **cabinB6_SAMPLEINTERVAL** = ”60”
- const string **cabinB6_ALARMLOWON** = ”11.0”
- const string **cabinB6_ALARMLOWOFF** = ”12.0”
- const string **cabinB6_ALARMHIGHON** = ”15.0”
- const string **cabinB6_ALARMHIGHOFF** = ”14.0”
- readonly attribute ACS::ROdouble **cabinB6**
- const string cabinC1_DESCRIPTION = ”Cabin C Temperature Sensor 1”
- const string **cabinC1_UNITS** = ”degC”
- const string **cabinC1_SAMPLEINTERVAL** = ”60”
- const string **cabinC1_ALARMINTERVAL** = ”60”
- const string **cabinC1_ALARMLOWON** = ”8.0”
- const string **cabinC1_ALARMLOWOFF** = ”9.0”
- const string **cabinC1_ALARMHIGHON** = ”12.0”
- const string **cabinC1_ALARMHIGHOFF** = ”11.0”
- readonly attribute ACS::ROdouble **cabinC1**
- const string cabinC2_DESCRIPTION = ”Cabin C Temperature Sensor 2”
- const string **cabinC2_UNITS** = ”degC”
- const string **cabinC2_SAMPLEINTERVAL** = ”60”
- const string **cabinC2_ALARMINTERVAL** = ”60”
- const string **cabinC2_ALARMLOWON** = ”8.0”
- const string **cabinC2_ALARMLOWOFF** = ”9.0”
- const string **cabinC2_ALARMHIGHON** = ”12.0”
- const string **cabinC2_ALARMHIGHOFF** = ”11.0”
- readonly attribute ACS::ROdouble **cabinC2**
- const string cabinC3_DESCRIPTION = ”Cabin C Temperature Sensor 3”
- const string **cabinC3_UNITS** = ”degC”
- const string **cabinC3_SAMPLEINTERVAL** = ”60”
- const string **cabinC3_ALARMINTERVAL** = ”60”
- const string **cabinC3_ALARMLOWON** = ”8.0”
- const string **cabinC3_ALARMLOWOFF** = ”9.0”
- const string **cabinC3_ALARMHIGHON** = ”12.0”
- const string **cabinC3_ALARMHIGHOFF** = ”11.0”
- readonly attribute ACS::ROdouble **cabinC3**
- const string cabinC4_DESCRIPTION = ”Cabin C Temperature Sensor 4”
- const string **cabinC4_UNITS** = ”degC”
- const string **cabinC4_SAMPLEINTERVAL** = ”60”

- const string **cabinC4_ALARMINTERVAL** = ”60”
- const string **cabinC4_ALARMLOWON** = ”8.0”
- const string **cabinC4_ALARMLOWOFF** = ”9.0”
- const string **cabinC4_ALARMHIGHON** = ”12.0”
- const string **cabinC4_ALARMHIGHOFF** = ”11.0”
- readonly attribute ACS::ROdouble **cabinC4**
- const string cabinC5_DESCRIPTION = ”Cabin C Temperature Sensor 5”
- const string **cabinC5_UNITS** = ”degC”
- const string **cabinC5_SAMPLEINTERVAL** = ”60”
- const string **cabinC5_ALARMINTERVAL** = ”60”
- const string **cabinC5_ALARMLOWON** = ”8.0”
- const string **cabinC5_ALARMLOWOFF** = ”9.0”
- const string **cabinC5_ALARMHIGHON** = ”12.0”
- const string **cabinC5_ALARMHIGHOFF** = ”11.0”
- readonly attribute ACS::ROdouble **cabinC5**
- const string cabinC6_DESCRIPTION = ”Cabin C Temperature Sensor 6”
- const string **cabinC6_UNITS** = ”degC”
- const string **cabinC6_SAMPLEINTERVAL** = ”60”
- const string **cabinC6_ALARMINTERVAL** = ”60”
- const string **cabinC6_ALARMLOWON** = ”8.0”
- const string **cabinC6_ALARMLOWOFF** = ”9.0”
- const string **cabinC6_ALARMHIGHON** = ”12.0”
- const string **cabinC6_ALARMHIGHOFF** = ”11.0”
- readonly attribute ACS::ROdouble **cabinC6**
- const string container_DESCRIPTION = ”Container Temperature”
- const string **container_UNITS** = ”degC”
- const string **container_SAMPLEINTERVAL** = ”60”
- const string **container_ALARMINTERVAL** = ”60”
- const string **container_ALARMLOWON** = ”5.0”
- const string **container_ALARMLOWOFF** = ”7.0”
- const string **container_ALARMHIGHON** = ”25.0”
- const string **container_ALARMHIGHOFF** = ”23.0”
- readonly attribute ACS::ROdouble **container**
- const string platform_DESCRIPTION = ”Platform Temperature”
- const string **platform_UNITS** = ”degC”
- const string **platform_SAMPLEINTERVAL** = ”60”
- const string **platform_ALARMINTERVAL** = ”60”
- const string **platform_ALARMLOWON** = ”0.0”
- const string **platform_ALARMLOWOFF** = ”2.0”
- const string **platform_ALARMHIGHON** = ”30.0”
- const string **platform_ALARMHIGHOFF** = ”28.0”
- readonly attribute ACS::ROdouble **platform**

8.9.1 Detailed Description

APEX Cabin and Rack Temperature Interface

8.9.2 Member Data Documentation

const string apexTemps::Temps::cabinA3_DESCRIPTION = ”Cabin A Temperature Sensor 3”

Cabin A temperature sensor 3

const string apexTemps::Temps::cabinA4_DESCRIPTION = ”Cabin A Temperature Sensor 4”

Cabin A temperature sensor 4

const string apexTemps::Temps::cabinA5_DESCRIPTION = ”Cabin A Temperature Sensor 5”

Cabin A temperature sensor 5

const string apexTemps::Temps::cabinA6_DESCRIPTION = ”Cabin A Temperature Sensor 6”

Cabin A temperature sensor 6

const string apexTemps::Temps::cabinARack1_DESCRIPTION = ”Cabin A Rack 1 Temperature”

Cabin A rack 1 temperature

const string apexTemps::Temps::cabinARack2_DESCRIPTION = ”Cabin A Rack 2 Temperature”

Cabin A rack 2 temperature

const string apexTemps::Temps::cabinB2_DESCRIPTION = ”Cabin B Temperature Sensor 2”

Cabin B temperature sensor 2

const string apexTemps::Temps::cabinB3_DESCRIPTION = ”Cabin B Temperature Sensor 3”

Cabin B temperature sensor 3

const string apexTemps::Temps::cabinB4_DESCRIPTION = ”Cabin B Temperature Sensor 4”

Cabin B temperature sensor 4

const string apexTemps::Temps::cabinB5_DESCRIPTION = ”Cabin B Temperature Sensor 5”

Cabin B temperature sensor 5

const string apexTemps::Temps::cabinB6_DESCRIPTION = ”Cabin B Temperature Sensor 6”

Cabin B temperature sensor 6

const string apexTemps::Temps::cabinBIFRack_DESCRIPTION = ”Cabin B IF Rack Temperature”

Cabin B IF rack temperature

const string apexTemps::Temps::cabinC1_DESCRIPTION = ”Cabin C Temperature Sensor 1”

Cabin C temperature sensor 1

const string apexTemps::Temps::cabinC2_DESCRIPTION = ”Cabin C Temperature Sensor 2”

Cabin C temperature sensor 2

const string apexTemps::Temps::cabinC3_DESCRIPTION = ”Cabin C Temperature Sensor 3”

Cabin C temperature sensor 3

const string apexTemps::Temps::cabinC4_DESCRIPTION = ”Cabin C Temperature Sensor 4”

Cabin C temperature sensor 4

const string apexTemps::Temps::cabinC5_DESCRIPTION = ”Cabin C Temperature Sensor 5”

Cabin C temperature sensor 5

const string apexTemps::Temps::cabinC6_DESCRIPTION = ”Cabin C Temperature Sensor 6”

Cabin C temperature sensor 6

const string apexTemps::Temps::container_DESCRIPTION = ”Container Temperature”

Container temperature

const string apexTemps::Temps::macsRack1_DESCRIPTION = ”MACS Rack 1 Temperature”

MACS rack 1 temperature

const string apexTemps::Temps::macsRack2_DESCRIPTION = ”MACS Rack 2 Temperature”

MACS rack 2 temperature

const string apexTemps::Temps::macsRack3_DESCRIPTION = ”MACS Rack 3 Temperature”

MACS rack 3 temperature

const string apexTemps::Temps::platform_DESCRIPTION = ”Platform Temperature”

Platform temperature

const string apexTemps::Temps::rack1_DESCRIPTION = ”Rack 1 Temperature”

Instrument container rack 1 temperature

const string apexTemps::Temps::rack2_DESCRIPTION = ”Rack 2 Temperature”

Instrument container rack 2 temperature

const string apexTemps::Temps::rack3_DESCRIPTION = ”Rack 3 Temperature”

Instrument container rack 3 temperature

const string apexTemps::Temps::rack4_DESCRIPTION = ”Rack 4 Temperature”

Instrument container rack 4 temperature

const string apexTemps::Temps::rack5_DESCRIPTION = ”Rack 5 Temperature”

Instrument container rack 5 temperature

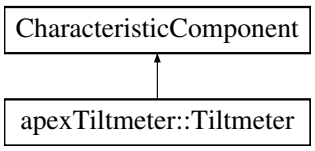
const string apexTemps::Temps::rack6_DESCRIPTION = ”Rack 6 Temperature”

Instrument container rack 6 temperature
The documentation for this interface was generated from the following file:

- apexTemps.idl

8.10 apexTiltmeter::Tiltmeter Interface Reference

`import"apexTiltmeter.idl";`
Inheritance diagram for apexTiltmeter::Tiltmeter:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string tiltX_DESCRIPTION = ”x-Amplitude”
- const string **tiltX_UNITS** = ”arcsec”
- const string **tiltX_SAMPLEINTERVAL** = ”5”
- readonly attribute ACS::ROdouble **tiltX**
- const string tiltY_DESCRIPTION = ”y-Amplitude”
- const string **tiltY_UNITS** = ”arcsec”
- const string **tiltY_SAMPLEINTERVAL** = ”5”
- readonly attribute ACS::ROdouble **tiltY**
- const string temperature_DESCRIPTION = ”Temperature”
- const string **temperature_UNITS** = ”degC”
- const string **temperature_SAMPLEINTERVAL** = ”5”
- readonly attribute ACS::ROdouble **temperature**

8.10.1 Detailed Description

APEX Tiltmeter Interface

8.10.2 Member Data Documentation

const string apexTiltmeter::Tiltmeter::state_DESCRIPTION = ”State”

State of the tiltmeter

const string apexTiltmeter::Tiltmeter::temperature_DESCRIPTION = ”Temperature”

Tiltmeter temperature

const string apexTiltmeter::Tiltmeter::tiltX_DESCRIPTION = ”x-Amplitude”

Tiltmeter x-amplitude

const string apexTiltmeter::Tiltmeter::tiltY_DESCRIPTION = ”y-Amplitude”

Tiltmeter y-amplitude
The documentation for this interface was generated from the following file:

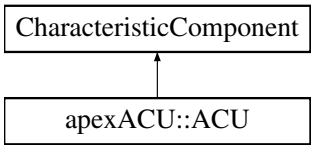
- apexTiltmeter.idl

Chapter 9

Dummy ACU Devices

9.1 apexACU::ACU Interface Reference

`import"apexACU.idl";`
Inheritance diagram for apexACU::ACU:



Public Attributes

- const string **SYSTEM_STATUS1_DESCRIPTION** = ”System Status 1”
- const string **SYSTEM_STATUS1_BITDESCRIPTION** = ”E-stop, Stairway interlock, Handling interlock, Smoke alarm, ACU boot failure, Stop due to missing commands after idle time, Stop due to missing timing pulse after idle time, Timing pulse missing, Safe switch, Power failure, 24V failure, Breaker failure, ACU fan failure, Communication error ACU - PLC, Cabinet over-temperature, Reserved, Reserved, Hoist interlock, Reserved, Ramp to receiver cabin not tilted up, Platform 2 handrails not lowered, Access interlock, Pedestal door open, Receiver cabin C door open, Cabinet e-stop, Az drives / equipment container e-stop, Stairs to platform 1 e-stop, Platform 2 / receiver cabin C e-stop, Nasmyth cabins A/B e-stop, Portable Control Unit e-stop, Reserved, Inside antenna base / control container e-stop”
- const string **SYSTEM_STATUS1_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N
- const string **SYSTEM_STATUS1_WHENSET** = ”F,F,F,F,F,F,F,F,S,F,F,F,F,F,F,N,N,S,N,S,S,S,S,S,F,F,F,F,F,F,N,F”
- const string **SYSTEM_STATUS1_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **SYSTEM_STATUS1**
- const string **SYSTEM_STATUS2_DESCRIPTION** = ”System Status 2”
- const string **SYSTEM_STATUS2_BITDESCRIPTION** = ”Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Communication error ACU - UPS1, Communication error ACU - UPS2, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved”
- const string **SYSTEM_STATUS2_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **SYSTEM_STATUS2_WHENSET** = ”N,N,N,N,N,N,N,N,S,S,N,N,N,N,N,N”
- const string **SYSTEM_STATUS2_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **SYSTEM_STATUS2**
- const string **SAS_STATUS1_DESCRIPTION** = ”SAS Status 1”
- const string **SAS_STATUS1_BITDESCRIPTION** = ”Sun avoidance interlock enabled, Sun avoidance interlock has triggered, Antenna moving due to sun avoidance, Sun avoidance interlock triggered but antenna cannot move, PLC time not yet confirmed after PLC reboot, SAS safe switch, Reserved, Reserved, Inaccurate time, GPS failure, Network connection from APEX to PLC failed, Reserved, Reserved, Reserved, Reserved, Reserved, Antenna power issues, Loss of network connection to base camp, Temperature sensors, Manual SAS activation, Reserved, Reserved, Reserved, Reserved, Inaccurate time input, GPS failure input, Network connection from APEX to PLC failed input, Reserved, Reserved, Reserved, Reserved, Reserved”

- const string **SAS_STATUS1_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,O-FF,OFF,OFF,N,N,N,N,N”
- const string **SAS_STATUS1_WHENSET** = ”S,F,S,F,S,S,N,N,S,S,S,N,N,N,N,N,F,F,F,F,N,N,N,N,ON,ON,O-N,N,N,N,N,N”
- const string **SAS_STATUS1_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **SAS_STATUS1**
- const string **SAS_STATUS2_DESCRIPTION** = ”SAS Status 2”
- const string **SAS_STATUS2_BITDESCRIPTION** = ”Antenna power issues input, Loss of network connection to base camp input, Temperature sensors input, Manual SAS activation input, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved”
- const string **SAS_STATUS2_WHENCLEARED** = ”OFF,OFF,OFF,OFF,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **SAS_STATUS2_WHENSET** = ”ON,ON,ON,ON,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **SAS_STATUS2_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **SAS_STATUS2**
- const string **AZ_STATUS1_DESCRIPTION** = ”Azimuth Status 1”
- const string **AZ_STATUS1_BITDESCRIPTION** = ”Prelimit cw, Prelimit ccw, Limit cw, Limit ccw, Emergency limit cw, Emergency limit ccw, 2nd emergency limit cw, 2nd emergency limit ccw, Servo failure, Overspeed, Immobile, Speed zero, Stow position, Encoder failure, Insane velocity feedback, DC bus overvoltage, Brake failure, Power Cycle Interlock, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Amplifier 1 failure, Amplifier 2 failure, Reserved, Reserved, CAN communication error to servo amplifiers, Breaker fault, Cam switch failure, Reserved”
- const string **AZ_STATUS1_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N
- const string **AZ_STATUS1_WHENSET** = ”S,S,S,S,S,S,S,S,S,F,F,F,S,S,F,F,F,F,S,N,N,N,N,N,N,F,F,N,N,F,F,F,N”
- const string **AZ_STATUS1_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **AZ_STATUS1**
- const string **AZ_STATUS2_DESCRIPTION** = ”Azimuth Status 2”
- const string **AZ_STATUS2_BITDESCRIPTION** = ”Motor 1 over temperature, Motor 2 over temperature, Reserved, Reserved, Regeneration resistor over temperature, Servo oscillation, Auxiliary encoder failure, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, AUX - motor 1 only, AUX - motor 2 only, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Computer disabled, Axis disabled, Handheld control unit (PCU) operation, Axis in stop, Flip-flop buffer position incorrect, Reserved, Reserved, Reserved”
- const string **AZ_STATUS2_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N
- const string **AZ_STATUS2_WHENSET** = ”F,F,N,N,F,F,S,N,N,N,N,N,N,N,N,N,S,S,N,N,N,N,N,N,S,S,S,S,S,N,N,N”
- const string **AZ_STATUS2_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **AZ_STATUS2**
- const string **EL_STATUS1_DESCRIPTION** = ”Elevation Status 1”
- const string **EL_STATUS1_BITDESCRIPTION** = ”Prelimit up, Prelimit down, Limit up, Limit down, Emergency limit up, Emergency limit down, Reserved, Reserved, Servo failure, Overspeed, Immobile, Speed zero, Stow position, Encoder failure, Insane velocity feedback, DC bus over-voltage, Brake failure, Power Cycle Interlock, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Amplifier 1 failure, Amplifier 2 failure, Amplifier 3 failure, Amplifier 4 failure, CAN communication error to servo amplifiers, Breaker fault, Cam switch failure, Reserved”
- const string **EL_STATUS1_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N
- const string **EL_STATUS1_WHENSET** = ”S,S,S,S,S,S,N,N,F,F,F,S,S,F,F,F,F,S,N,N,N,N,N,N,F,F,F,F,F,F,F,N”
- const string **EL_STATUS1_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **EL_STATUS1**
- const string **EL_STATUS2_DESCRIPTION** = ”Elevation Status 2”
- const string **EL_STATUS2_BITDESCRIPTION** = ”Motor 1 over temperature, Motor 2 over temperature, Motor 3 over temperature, Motor 4 over temperature, Regeneration resistor over temperature, Servo oscillation, Auxiliary encoder failure, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, AUX - motors 1 and 2 only, AUX - motors 3 and 4 only, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Computer disabled, Axis disabled, Handheld control unit (PCU) operation, Axis in stop, Reserved, Reserved, Reserved, Reserved”
- const string **EL_STATUS2_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N
- const string **EL_STATUS2_WHENSET** = ”F,F,F,F,F,F,F,N,N,N,N,N,N,N,N,N,S,S,N,N,N,N,N,N,S,S,S,S,N,N,N,N”

- const string **EL_STATUS2_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **EL_STATUS2**
- const string **AC_STATUS_DESCRIPTION** = ”A/C Status”
- const string **AC_STATUS_BITDESCRIPTION** = ”Air conditioning system 1, Air conditioning system 2, Air conditioning system automatic, Air conditioning system in local, Air conditioning system 1 alarm, Air conditioning system 2 alarm, Reserved, Reserved”
- const string **AC_STATUS_WHENCLEARED** = ”OFF,OFF,N,N,N,N,N,N”
- const string **AC_STATUS_WHENSET** = ”ON,ON,S,S,F,F,N,N”
- const string **AC_STATUS_UPDATEINT** = ”30”
- readonly attribute ACS::ROpattern **AC_STATUS**
- const string **OTHER_STATUS_DESCRIPTION** = ”Other Status”
- const string **OTHER_STATUS_BITDESCRIPTION** = ”Smoke alarm computer rack, Smoke alarm receiver cabin C, Smoke alarm antenna tower, Smoke alarm UPS cabinet / equipment platform, Over-temperature receiver cabin C, Reserved, Reserved, Reserved”
- const string **OTHER_STATUS_WHENCLEARED** = ”N,N,N,N,N,N,N,N”
- const string **OTHER_STATUS_WHENSET** = ”F,F,F,F,S,N,N,N”
- const string **OTHER_STATUS_UPDATEINT** = ”30”
- readonly attribute ACS::ROpattern **OTHER_STATUS**
- const string **POWER_STATUS_DESCRIPTION** = ”Power Status”
- const string **POWER_STATUS_BITDESCRIPTION** = ”Reserved, Main switch: Antenna base power supply, Main switch: Transportable power supply, Reserved, Reserved, Lightning arrestor tripped, Reserved, Reserved, Single phase interlock, Reverse phase protection released, Circuit breaker critical electronic bus (to UPS), Circuit breaker equipment container power, Reserved, Reserved, Reserved, Reserved”
- const string **POWER_STATUS_WHENCLEARED** = ”N,OFF,OFF,N,N,N,N,N,N,N,OFF,OFF,N,N,N,N”
- const string **POWER_STATUS_WHENSET** = ”N,ON,ON,N,N,S,N,N,F,F,ON,ON,N,N,N,N”
- const string **POWER_STATUS_UPDATEINT** = ”30”
- readonly attribute ACS::ROpattern **POWER_STATUS**
- const string **SHUTTER_STATUS_DESCRIPTION** = ”Shutter Status”
- const string **SHUTTER_STATUS_BITDESCRIPTION** = ”Open, Close, Motor shutter, Local system error, Shutter motion timeout, Opening possible without heating, Heating period active, Heater fault”
- const string **SHUTTER_STATUS_WHENCLEARED** = ”N,N,OFF,N,N,N,N,N”
- const string **SHUTTER_STATUS_WHENSET** = ”S,S,ON,F,F,S,S,F”
- const string **SHUTTER_STATUS_UPDATEINT** = ”60”
- readonly attribute ACS::ROpattern **SHUTTER_STATUS**
- const string **SHUTTER_TEMP1_DESCRIPTION** = ”Shutter Temperature 1”
- const string **SHUTTER_TEMP1_UNITS** = ”degC”
- const string **SHUTTER_TEMP1_UPDATEINT** = ”60”
- readonly attribute ACS::ROdouble **SHUTTER_TEMP1**
- const string **SHUTTER_TEMP2_DESCRIPTION** = ”Shutter Temperature 2”
- const string **SHUTTER_TEMP2_UNITS** = ”degC”
- const string **SHUTTER_TEMP2_UPDATEINT** = ”60”
- readonly attribute ACS::ROdouble **SHUTTER_TEMP2**

9.1.1 Detailed Description

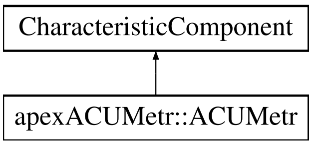
APEX dummy ACU Interface
The documentation for this interface was generated from the following file:

- apexACU.idl

9.2 apexACUMetr::ACUMetr Interface Reference

`import"apexACUMetr.idl";`

Inheritance diagram for apexACUMetr::ACUMetr:



Public Attributes

- const string **MODE_DESCRIPTION** = ”Metrology Mode”
- const string **MODE_BITDESCRIPTION** = ”Standard pointing error model enabled, Tiltmeter compensation enabled, Tiltmeter temperature offset model enabled, Delta path length temperature model enabled, Automatic subreflector position correction enabled (Nasmyth cabin), Automatic subreflector position correction enabled (Cassegrain cabin), Correction based on encoder mount displacement sensors enabled, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved”
- const string **MODE_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **MODE_WHENSET** = ”S,S,S,S,S,S,S,N,N,N,N,N,N,N,N,N”
- const string **MODE_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **MODE**
- const string **EQUIP_STATUS_DESCRIPTION** = ”Equipment Status”
- const string **EQUIP_STATUS_BITDESCRIPTION** = ”Local bus error ACU - metrology equipment, Tiltmeter 1 readout error, Tiltmeter 2 readout error, Tiltmeter 3 readout error, Reserved, Reserved, Temperature sensor readout error, Displacement sensor readout error, Reserved, Communication error ACU - MMS, Metrology data is not valid”
- const string **EQUIP_STATUS_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,
- const string **EQUIP_STATUS_WHENSET** = ”F,F,F,F,N,N,F,F,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,F,S”
- const string **EQUIP_STATUS_UPDATEINT** = ”30”
- readonly attribute ACS::ROpattern **EQUIP_STATUS**

9.2.1 Detailed Description

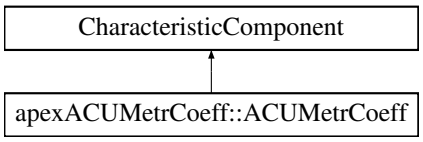
APEX dummy ACU Metrology Interface

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

- apexACUMetr.idl

9.3 apexACUMetrCoeff::ACUMetrCoeff Interface Reference

`import"apexACUMetrCoeff.idl";`
Inheritance diagram for apexACUMetrCoeff::ACUMetrCoeff:



Public Attributes

- const string **AN0_DESCRIPTION** = ”AN0”
- const string **AN0_UNITS** = ”arcsec”
- const string **AN0_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **AN0**
- const string **AW0_DESCRIPTION** = ”AW0”
- const string **AW0_UNITS** = ”arcsec”
- const string **AW0_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **AW0**

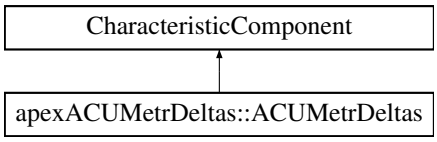
9.3.1 Detailed Description

APEX dummy ACU Metrology Coefficient Interface
The documentation for this interface was generated from the following file:

- apexACUMetrCoeff.idl

9.4 apexACUMetrDeltas::ACUMetrDeltas Interface Reference

`import"apexACUMetrDeltas.idl";`
Inheritance diagram for apexACUMetrDeltas::ACUMetrDeltas:



Public Attributes

- const string **DAZ_SPEM_DESCRIPTION** = ”Pointing model az corr.”
- const string **DAZ_SPEM_UNITS** = ”arcsec”
- const string **DAZ_SPEM_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DAZ_SPEM**
- const string **DEL_SPEM_DESCRIPTION** = ”Pointing model el corr.”
- const string **DEL_SPEM_UNITS** = ”arcsec”
- const string **DEL_SPEM_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DEL_SPEM**
- const string **DAZ_TEMP_DESCRIPTION** = ”Temp. model az corr.”
- const string **DAZ_TEMP_UNITS** = ”arcsec”
- const string **DAZ_TEMP_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DAZ_TEMP**
- const string **DEL_TEMP_DESCRIPTION** = ”Temp. model el corr.”
- const string **DEL_TEMP_UNITS** = ”arcsec”
- const string **DEL_TEMP_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DEL_TEMP**
- const string **DAZ_TILT_DESCRIPTION** = ”Tiltmeter az corr.”
- const string **DAZ_TILT_UNITS** = ”arcsec”
- const string **DAZ_TILT_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DAZ_TILT**
- const string **DEL_TILT_DESCRIPTION** = ”Tiltmeter el corr.”
- const string **DEL_TILT_UNITS** = ”arcsec”
- const string **DEL_TILT_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DEL_TILT**
- const string **DAZ_TILTTEMP_DESCRIPTION** = ”Tilt. temp. offset az corr.”
- const string **DAZ_TILTTEMP_UNITS** = ”arcsec”
- const string **DAZ_TILTTEMP_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DAZ_TILTTEMP**
- const string **DEL_TILTTEMP_DESCRIPTION** = ”Tilt. temp. offset el corr.”
- const string **DEL_TILTTEMP_UNITS** = ”arcsec”
- const string **DEL_TILTTEMP_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DEL_TILTTEMP**
- const string **DAZ_DISP_DESCRIPTION** = ”Displ. sensor az corr.”
- const string **DAZ_DISP_UNITS** = ”arcsec”
- const string **DAZ_DISP_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DAZ_DISP**
- const string **DEL_DISP_DESCRIPTION** = ”Displ. sensor el corr.”
- const string **DEL_DISP_UNITS** = ”arcsec”
- const string **DEL_DISP_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DEL_DISP**
- const string **DAZ_TOTAL_DESCRIPTION** = ”Applied az correction”
- const string **DAZ_TOTAL_UNITS** = ”arcsec”

- const string **DAZ_TOTAL_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DAZ_TOTAL**
- const string **DEL_TOTAL_DESCRIPTION** = ”Applied el correction”
- const string **DEL_TOTAL_UNITS** = ”arcsec”
- const string **DEL_TOTAL_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DEL_TOTAL**
- const string **DELTAPATH_DESCRIPTION** = ”Z-focus temperature correction”
- const string **DELTAPATH_UNITS** = ”mm”
- const string **DELTAPATH_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **DELTAPATH**

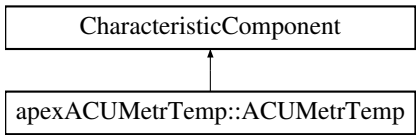
9.4.1 Detailed Description

APEX dummy ACU Metrology Deltas Interface
The documentation for this interface was generated from the following file:

- apexACUMetrDeltas.idl

9.5 apexACUMetrTemp::ACUMetrTemp Interface Reference

`import"apexACUMetrTemp.idl";`
Inheritance diagram for apexACUMetrTemp::ACUMetrTemp:



Public Attributes

- const string **TEMP01_DESCRIPTION** = ”Temperature Sensor 1”
- const string **TEMP01_UNITS** = ”degC”
- const string **TEMP01_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP01**
- const string **TEMP02_DESCRIPTION** = ”Temperature Sensor 2”
- const string **TEMP02_UNITS** = ”degC”
- const string **TEMP02_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP02**
- const string **TEMP03_DESCRIPTION** = ”Temperature Sensor 3”
- const string **TEMP03_UNITS** = ”degC”
- const string **TEMP03_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP03**
- const string **TEMP04_DESCRIPTION** = ”Temperature Sensor 4”
- const string **TEMP04_UNITS** = ”degC”
- const string **TEMP04_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP04**
- const string **TEMP05_DESCRIPTION** = ”Temperature Sensor 5”
- const string **TEMP05_UNITS** = ”degC”
- const string **TEMP05_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP05**
- const string **TEMP06_DESCRIPTION** = ”Temperature Sensor 6”
- const string **TEMP06_UNITS** = ”degC”
- const string **TEMP06_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP06**
- const string **TEMP07_DESCRIPTION** = ”Temperature Sensor 7”
- const string **TEMP07_UNITS** = ”degC”
- const string **TEMP07_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP07**
- const string **TEMP08_DESCRIPTION** = ”Temperature Sensor 8”
- const string **TEMP08_UNITS** = ”degC”
- const string **TEMP08_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP08**
- const string **TEMP09_DESCRIPTION** = ”Temperature Sensor 9”
- const string **TEMP09_UNITS** = ”degC”
- const string **TEMP09_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP09**
- const string **TEMP10_DESCRIPTION** = ”Temperature Sensor 10”
- const string **TEMP10_UNITS** = ”degC”
- const string **TEMP10_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP10**
- const string **TEMP11_DESCRIPTION** = ”Temperature Sensor 11”
- const string **TEMP11_UNITS** = ”degC”

- const string **TEMP11_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP11**
- const string **TEMP12_DESCRIPTION** = ”Temperature Sensor 12”
- const string **TEMP12_UNITS** = ”degC”
- const string **TEMP12_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP12**
- const string **TEMP13_DESCRIPTION** = ”Temperature Sensor 13”
- const string **TEMP13_UNITS** = ”degC”
- const string **TEMP13_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP13**
- const string **TEMP14_DESCRIPTION** = ”Temperature Sensor 14”
- const string **TEMP14_UNITS** = ”degC”
- const string **TEMP14_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP14**
- const string **TEMP15_DESCRIPTION** = ”Temperature Sensor 15”
- const string **TEMP15_UNITS** = ”degC”
- const string **TEMP15_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP15**
- const string **TEMP16_DESCRIPTION** = ”Temperature Sensor 16”
- const string **TEMP16_UNITS** = ”degC”
- const string **TEMP16_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP16**
- const string **TEMP17_DESCRIPTION** = ”Temperature Sensor 17”
- const string **TEMP17_UNITS** = ”degC”
- const string **TEMP17_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP17**
- const string **TEMP18_DESCRIPTION** = ”Temperature Sensor 18”
- const string **TEMP18_UNITS** = ”degC”
- const string **TEMP18_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP18**
- const string **TEMP19_DESCRIPTION** = ”Temperature Sensor 19”
- const string **TEMP19_UNITS** = ”degC”
- const string **TEMP19_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP19**
- const string **TEMP20_DESCRIPTION** = ”Temperature Sensor 20”
- const string **TEMP20_UNITS** = ”degC”
- const string **TEMP20_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP20**
- const string **TEMP21_DESCRIPTION** = ”Temperature Sensor 21”
- const string **TEMP21_UNITS** = ”degC”
- const string **TEMP21_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP21**
- const string **TEMP22_DESCRIPTION** = ”Temperature Sensor 22”
- const string **TEMP22_UNITS** = ”degC”
- const string **TEMP22_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP22**
- const string **TEMP23_DESCRIPTION** = ”Temperature Sensor 23”
- const string **TEMP23_UNITS** = ”degC”
- const string **TEMP23_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP23**
- const string **TEMP24_DESCRIPTION** = ”Temperature Sensor 24”
- const string **TEMP24_UNITS** = ”degC”
- const string **TEMP24_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP24**

- const string **TEMP25_DESCRIPTION** = ”Temperature Sensor 25”
- const string **TEMP25_UNITS** = ”degC”
- const string **TEMP25_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP25**
- const string **TEMP26_DESCRIPTION** = ”Temperature Sensor 26”
- const string **TEMP26_UNITS** = ”degC”
- const string **TEMP26_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP26**
- const string **TEMP27_DESCRIPTION** = ”Temperature Sensor 27”
- const string **TEMP27_UNITS** = ”degC”
- const string **TEMP27_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP27**
- const string **TEMP28_DESCRIPTION** = ”Temperature Sensor 28”
- const string **TEMP28_UNITS** = ”degC”
- const string **TEMP28_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **TEMP28**

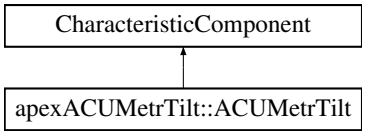
9.5.1 Detailed Description

APEX dummy ACU Metrology Temperature Interface
The documentation for this interface was generated from the following file:

- apexACUMetrTemp.idl

9.6 apexACUMetrTilt::ACUMetrTilt Interface Reference

`import"apexACUMetrTilt.idl";`
Inheritance diagram for apexACUMetrTilt::ACUMetrTilt:



Public Attributes

- const string **TILT1X_DESCRIPTION** = ”Tiltmeter 1 x-Amplitude”
- const string **TILT1X_UNITS** = ”arcsec”
- const string **TILT1X_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT1X**
- const string **TILT1Y_DESCRIPTION** = ”Tiltmeter 1 y-Amplitude”
- const string **TILT1Y_UNITS** = ”arcsec”
- const string **TILT1Y_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT1Y**
- const string **TILT1T_DESCRIPTION** = ”Tiltmeter 1 Temperature”
- const string **TILT1T_UNITS** = ”degC”
- const string **TILT1T_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT1T**
- const string **TILT2X_DESCRIPTION** = ”Tiltmeter 2 x-Amplitude”
- const string **TILT2X_UNITS** = ”arcsec”
- const string **TILT2X_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT2X**
- const string **TILT2Y_DESCRIPTION** = ”Tiltmeter 2 y-Amplitude”
- const string **TILT2Y_UNITS** = ”arcsec”
- const string **TILT2Y_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT2Y**
- const string **TILT2T_DESCRIPTION** = ”Tiltmeter 2 Temperature”
- const string **TILT2T_UNITS** = ”degC”
- const string **TILT2T_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT2T**
- const string **TILT3X_DESCRIPTION** = ”Tiltmeter 3 x-Amplitude”
- const string **TILT3X_UNITS** = ”arcsec”
- const string **TILT3X_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT3X**
- const string **TILT3Y_DESCRIPTION** = ”Tiltmeter 3 y-Amplitude”
- const string **TILT3Y_UNITS** = ”arcsec”
- const string **TILT3Y_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT3Y**
- const string **TILT3T_DESCRIPTION** = ”Tiltmeter 3 Temperature”
- const string **TILT3T_UNITS** = ”degC”
- const string **TILT3T_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **TILT3T**
- const string **DISP_ABS1_DESCRIPTION** = ”Displacement Sensor 1”
- const string **DISP_ABS1_UNITS** = ”mum”
- const string **DISP_ABS1_DISPLAYUNITS** = ”mm”
- const string **DISP_ABS1_FORMAT** = ”%.4f”
- const string **DISP_ABS1_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DISP_ABS1**

- const string **DISP_ABS2_DESCRIPTION** = ”Displacement Sensor 2”
- const string **DISP_ABS2_UNITS** = ”mum”
- const string **DISP_ABS2_DISPLAYUNITS** = ”mm”
- const string **DISP_ABS2_FORMAT** = ”%.4f”
- const string **DISP_ABS2_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DISP_ABS2**
- const string **DISP_ABS3_DESCRIPTION** = ”Displacement Sensor 3”
- const string **DISP_ABS3_UNITS** = ”mum”
- const string **DISP_ABS3_DISPLAYUNITS** = ”mm”
- const string **DISP_ABS3_FORMAT** = ”%.4f”
- const string **DISP_ABS3_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DISP_ABS3**
- const string **DISP_ABS4_DESCRIPTION** = ”Displacement Sensor 4”
- const string **DISP_ABS4_UNITS** = ”mum”
- const string **DISP_ABS4_DISPLAYUNITS** = ”mm”
- const string **DISP_ABS4_FORMAT** = ”%.4f”
- const string **DISP_ABS4_UPDATEINT** = ”5”
- readonly attribute ACS::ROdouble **DISP_ABS4**

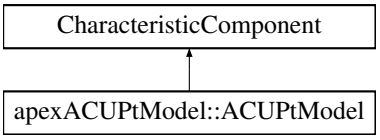
9.6.1 Detailed Description

APEX dummy ACU Metrology Tiltmeter Interface
The documentation for this interface was generated from the following file:

- apexACUMetrTilt.idl

9.7 apexACUPtModel::ACUPtModel Interface Reference

`import"apexACUPtModel.idl";`
Inheritance diagram for apexACUPtModel::ACUPtModel:



Public Attributes

- const string **IA_DESCRIPTION** = ”IA”
- const string **IA_UNITS** = ”arcsec”
- const string **IA_UPDATEINT** = ”180”
- readonly attribute ACS::ROdouble **IA**
- const string **CA_DESCRIPTION** = ”CA”
- const string **CA_UNITS** = ”arcsec”
- const string **CA_UPDATEINT** = ”180”
- readonly attribute ACS::ROdouble **CA**
- const string **NPAE_DESCRIPTION** = ”NPAE”
- const string **NPAE_UNITS** = ”arcsec”
- const string **NPAE_UPDATEINT** = ”180”
- readonly attribute ACS::ROdouble **NPAE**
- const string **AN_DESCRIPTION** = ”AN”
- const string **AN_UNITS** = ”arcsec”
- const string **AN_UPDATEINT** = ”180”
- readonly attribute ACS::ROdouble **AN**
- const string **AW_DESCRIPTION** = ”AW”
- const string **AW_UNITS** = ”arcsec”
- const string **AW_UPDATEINT** = ”180”
- readonly attribute ACS::ROdouble **AW**
- const string **IE_DESCRIPTION** = ”IE”
- const string **IE_UNITS** = ”arcsec”
- const string **IE_UPDATEINT** = ”180”
- readonly attribute ACS::ROdouble **IE**
- const string **ECEC_DESCRIPTION** = ”ECEC”
- const string **ECEC_UNITS** = ”arcsec”
- const string **ECEC_UPDATEINT** = ”180”
- readonly attribute ACS::ROdouble **ECEC**
- const string **ZFLX_DESCRIPTION** = ”ZFLX”
- const string **ZFLX_UNITS** = ”arcsec”
- const string **ZFLX_UPDATEINT** = ”180”
- readonly attribute ACS::ROdouble **ZFLX**

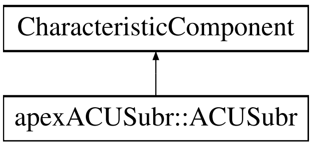
9.7.1 Detailed Description

APEX dummy ACU Pointing Model Interface
The documentation for this interface was generated from the following file:

- apexACUPtModel.idl

9.8 apexACUSubr::ACUSubr Interface Reference

`import"apexACUSubr.idl";`
Inheritance diagram for apexACUSubr::ACUSubr:



Public Attributes

- const string **LIMIT1_DESCRIPTION** = ”Hexapod Limits 1”
- const string **LIMIT1_BITDESCRIPTION** = ”Actuator 1 negative limit, Actuator 2 negative limit, Actuator 3 negative limit, Actuator 4 negative limit, Actuator 5 negative limit, Actuator 6 negative limit, Reserved, Reserved, Actuator 1 positive limit, Actuator 2 positive limit, Actuator 3 positive limit, Actuator 4 positive limit, Actuator 5 positive limit, Actuator 6 positive limit, Reserved, Reserved, Hexapod negative x-limit, Hexapod positive x-limit, Hexapod negative y-limit, Hexapod positive y-limit, Hexapod negative z-limit, Hexapod positive z-limit, Reserved, Reserved, Hexapod negative theta-x-limit, Hexapod positive theta-x-limit, Hexapod negative theta-y-limit, Hexapod positive theta-y-limit, Hexapod negative theta-z-limit, Hexapod positive theta-z-limit, Reserved, Reserved”
- const string **LIMIT1_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **LIMIT1_WHENSET** = ”S,S,S,S,S,S,N,N,S,S,S,S,S,S,N,N,S,S,S,S,S,S,N,N,S,S,S,S,S,S,N,N”
- const string **LIMIT1_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **LIMIT1**
- const string **LIMIT2_DESCRIPTION** = ”Hexapod Limits 2”
- const string **LIMIT2_BITDESCRIPTION** = ”Hexapod vertical collision switch 1, Hexapod vertical collision switch 2, Hexapod vertical collision switch 3, Hexapod vertical collision switch 4, Hexapod vertical collision switch 5, Hexapod vertical collision switch 6, Hexapod vertical collision switch 7, Hexapod vertical collision switch 8, Hexapod horizontal collision switch 1, Hexapod horizontal collision switch 2, Hexapod horizontal collision switch 3, Hexapod horizontal collision switch 4, Hexapod horizontal collision switch 5, Hexapod horizontal collision switch 6, Hexapod horizontal collision switch 7, Hexapod horizontal collision switch 8”
- const string **LIMIT2_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **LIMIT2_WHENSET** = ”F,F,F,F,F,F,F,F,F,F,F,F,F,F,F,F”
- const string **LIMIT2_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **LIMIT2**
- const string **STATUS1_DESCRIPTION** = ”Hexapod Status 1”
- const string **STATUS1_BITDESCRIPTION** = ”Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Amplifier actuator 1 over-temperature, Amplifier actuator 2 over-temperature, Amplifier actuator 3 over-temperature, Amplifier actuator 4 over-temperature, Amplifier actuator 5 over-temperature, Amplifier actuator 6 over-temperature, Reserved, Reserved, Actuator 1 does not move, Actuator 2 does not move, Actuator 3 does not move, Actuator 4 does not move, Actuator 5 does not move, Actuator 6 does not move, Reserved, Reserved, Actuator 1 not initialized, Actuator 2 not initialized, Actuator 3 not initialized, Actuator 4 not initialized, Actuator 5 not initialized, Actuator 6 not initialized, Hexapod in initializing mode, Reserved”
- const string **STATUS1_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **STATUS1_WHENSET** = ”N,N,N,N,N,N,N,N,F,F,F,F,F,F,N,N,F,F,F,F,F,F,N,N,S,S,S,S,S,S,S,N”
- const string **STATUS1_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **STATUS1**
- const string **STATUS2_DESCRIPTION** = ”Hexapod Status 2”
- const string **STATUS2_BITDESCRIPTION** = ”Subreflector drive power supply failure, ACU - HPC communication error, Temperature of drive electronics at apex too low, Hexapod at commanded position, Reserved, Reserved, Reserved, Reserved, Circuit breaker infeed power hexapod tripped, Circuit breaker 24 V power supply hexapod tripped, 24 V circuit breaker hexapod PWM 1 tripped, 24 V circuit breaker hexapod PWM 2 tripped, 24 V circuit breaker hexapod PWM 3 tripped, 24 V circuit breaker HPC tripped, Hexapod PWM power off, 24 V power supply hexapod fault”

- const string **STATUS2_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **STATUS2_WHENSET** = ”F,F,S,S,N,N,N,N,F,F,F,F,F,F,S,F”
- const string **STATUS2_UPDATEINT** = ”10”
- readonly attribute ACS::ROpattern **STATUS2**

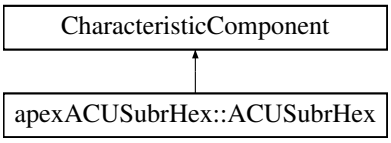
9.8.1 Detailed Description

APEX dummy ACU Subreflector Status Interface
The documentation for this interface was generated from the following file:

- apexACUSubr.idl

9.9 apexACUSubrHex::ACUSubrHex Interface Reference

```
import"apexACUSubrHex.idl";
```
Inheritance diagram for apexACUSubrHex::ACUSubrHex:



Public Attributes

- const string **POSITIONX_DESCRIPTION** = ”X-Position”
- const string **POSITIONX_UNITS** = ”mum”
- const string **POSITIONX_DISPLAYUNITS** = ”mm”
- const string **POSITIONX_FORMAT** = ”%.3f”
- const string **POSITIONX_UPDATEINT** = ”10”
- readonly attribute ACS::ROdouble **POSITIONX**
- const string **POSITIONY_DESCRIPTION** = ”Y-Position”
- const string **POSITIONY_UNITS** = ”mum”
- const string **POSITIONY_DISPLAYUNITS** = ”mm”
- const string **POSITIONY_FORMAT** = ”%.3f”
- const string **POSITIONY_UPDATEINT** = ”10”
- readonly attribute ACS::ROdouble **POSITIONY**
- const string **POSITIONZ_DESCRIPTION** = ”Z-Position”
- const string **POSITIONZ_UNITS** = ”mum”
- const string **POSITIONZ_DISPLAYUNITS** = ”mm”
- const string **POSITIONZ_FORMAT** = ”%.3f”
- const string **POSITIONZ_UPDATEINT** = ”10”
- readonly attribute ACS::ROdouble **POSITIONZ**
- const string **ROTATIONX_DESCRIPTION** = ”X-Rotation”
- const string **ROTATIONX_UNITS** = ”mdeg”
- const string **ROTATIONX_DISPLAYUNITS** = ”arcsec”
- const string **ROTATIONX_UPDATEINT** = ”10”
- readonly attribute ACS::ROdouble **ROTATIONX**
- const string **ROTATIONY_DESCRIPTION** = ”Y-Rotation”
- const string **ROTATIONY_UNITS** = ”mdeg”
- const string **ROTATIONY_DISPLAYUNITS** = ”arcsec”
- const string **ROTATIONY_UPDATEINT** = ”10”
- readonly attribute ACS::ROdouble **ROTATIONY**
- const string **ROTATIONZ_DESCRIPTION** = ”Z-Rotation”
- const string **ROTATIONZ_UNITS** = ”mdeg”
- const string **ROTATIONZ_DISPLAYUNITS** = ”arcsec”
- const string **ROTATIONZ_UPDATEINT** = ”10”
- readonly attribute ACS::ROdouble **ROTATIONZ**
- const string **DELTA_POSNX_DESCRIPTION** = ”Delta X-Position”
- const string **DELTA_POSNX_UNITS** = ”mum”
- const string **DELTA_POSNX_DISPLAYUNITS** = ”mm”
- const string **DELTA_POSNX_FORMAT** = ”%.3f”
- const string **DELTA_POSNX_UPDATEINT** = ”10”
- readonly attribute ACS::ROdouble **DELTA_POSNX**
- const string **DELTA_POSNY_DESCRIPTION** = ”Delta Y-Position”
- const string **DELTA_POSNY_UNITS** = ”mum”
- const string **DELTA_POSNY_DISPLAYUNITS** = ”mm”

- const string **DELTA_POSNY_FORMAT** = "%.3f"
- const string **DELTA_POSNY_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **DELTA_POSNY**
- const string **DELTA_POSNZ_DESCRIPTION** = "Delta Z-Position"
- const string **DELTA_POSNZ_UNITS** = "mum"
- const string **DELTA_POSNZ_DISPLAYUNITS** = "mm"
- const string **DELTA_POSNZ_FORMAT** = "%.3f"
- const string **DELTA_POSNZ_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **DELTA_POSNZ**
- const string **DELTA_ROTNX_DESCRIPTION** = "Delta X-Rotation"
- const string **DELTA_ROTNX_UNITS** = "mdeg"
- const string **DELTA_ROTNX_DISPLAYUNITS** = "arcsec"
- const string **DELTA_ROTNX_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **DELTA_ROTNX**
- const string **DELTA_ROTNY_DESCRIPTION** = "Delta Y-Rotation"
- const string **DELTA_ROTNY_UNITS** = "mdeg"
- const string **DELTA_ROTNY_DISPLAYUNITS** = "arcsec"
- const string **DELTA_ROTNY_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **DELTA_ROTNY**
- const string **DELTA_ROTNZ_DESCRIPTION** = "Delta Z-Rotation"
- const string **DELTA_ROTNZ_UNITS** = "mdeg"
- const string **DELTA_ROTNZ_DISPLAYUNITS** = "arcsec"
- const string **DELTA_ROTNZ_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **DELTA_ROTNZ**
- const string **TEMP1_DESCRIPTION** = "Actuator 1 Motor Temp."
- const string **TEMP1_UNITS** = "degC"
- const string **TEMP1_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **TEMP1**
- const string **TEMP2_DESCRIPTION** = "Actuator 2 Motor Temp."
- const string **TEMP2_UNITS** = "degC"
- const string **TEMP2_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **TEMP2**
- const string **TEMP3_DESCRIPTION** = "Actuator 3 Motor Temp."
- const string **TEMP3_UNITS** = "degC"
- const string **TEMP3_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **TEMP3**
- const string **TEMP4_DESCRIPTION** = "Actuator 4 Motor Temp."
- const string **TEMP4_UNITS** = "degC"
- const string **TEMP4_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **TEMP4**
- const string **TEMP5_DESCRIPTION** = "Actuator 5 Motor Temp."
- const string **TEMP5_UNITS** = "degC"
- const string **TEMP5_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **TEMP5**
- const string **TEMP6_DESCRIPTION** = "Actuator 6 Motor Temp."
- const string **TEMP6_UNITS** = "degC"
- const string **TEMP6_UPDATEINT** = "10"
- readonly attribute ACS::ROdouble **TEMP6**

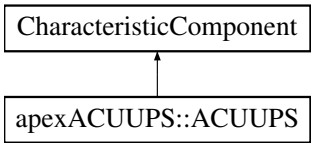
9.9.1 Detailed Description

APEX dummy ACU Subreflector Hexapod Interface
The documentation for this interface was generated from the following file:

- apexACUSubrHex.idl

9.10 apexACUUPS::ACUUPS Interface Reference

`import"apexACUUPS.idl";`
Inheritance diagram for apexACUUPS::ACUUPS:



Public Attributes

- const string **STATUS_DESCRIPTION** = ”Status”
- const string **STATUS_BITDESCRIPTION** = ”DC voltage within tolerance, Battery has normal capacity, Inverter within tolerance, Inverter feeds load, Inverter synchrony, Bypass within tolerance, Reserved, Reserved”
- const string **STATUS_WHENCLEARED** = ”N,N,N,N,N,N,N,N”
- const string **STATUS_WHENSET** = ”S,S,S,S,S,S,N,N”
- const string **STATUS_UPDATEINT** = ”30”
- readonly attribute ACS::ROpattern **STATUS**
- const string **ALARM_DESCRIPTION** = ”Alarm”
- const string **ALARM_BITDESCRIPTION** = ”Mains failure, Rectifier failure, Rectifier fuse blown, Thermal image, Rectifier output voltage out of tolerance, UPS phase sequenece not correct, Battery switch open, Battery is already discharged, Battery capacity close to zero, Battery fault, Performing battery test, PLL fault, Inverter voltage out of tolerance, Overload, Emergency bypass is not available, Load supplied by bypass, Retransfer between bypass and inverter is blocked, Manual bypass switch is closed, Output circuit breaker is open, Reserved, High temperature, Bypass switch is in position BYPASS, Electronics power off, Inverter bridge is blocked due to current exceeding limit, Short circuit, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Communication error ACU - UPS”
- const string **ALARM_WHENCLEARED** = ”N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N,N”
- const string **ALARM_WHENSET** = ”F,F,F,S,S,S,S,S,S,F,S,F,S,S,S,S,F,S,S,S,S,S,S,F,S,N,N,N,N,N,N,F”
- const string **ALARM_UPDATEINT** = ”30”
- readonly attribute ACS::ROpattern **ALARM**
- const string **BATTERY_VOLTS_DESCRIPTION** = ”Battery Voltage”
- const string **BATTERY_VOLTS_UNITS** = ”V”
- const string **BATTERY_VOLTS_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **BATTERY_VOLTS**
- const string **BATTERY_AMPS_DESCRIPTION** = ”Battery Current”
- const string **BATTERY_AMPS_UNITS** = ”A”
- const string **BATTERY_AMPS_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **BATTERY_AMPS**
- const string **BATTERY_MIN_DESCRIPTION** = ”Battery Time”
- const string **BATTERY_MIN_UNITS** = ”min”
- const string **BATTERY_MIN_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **BATTERY_MIN**
- const string **BATTERY_PERCENT_DESCRIPTION** = ”Battery Capacity”
- const string **BATTERY_PERCENT_UNITS** = ”percent”
- const string **BATTERY_PERCENT_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **BATTERY_PERCENT**
- const string **BYPASS_VOLTS1_DESCRIPTION** = ”Bypass Phase 1 Voltage”
- const string **BYPASS_VOLTS1_UNITS** = ”V”
- const string **BYPASS_VOLTS1_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **BYPASS_VOLTS1**
- const string **BYPASS_VOLTS2_DESCRIPTION** = ”Bypass Phase 2 Voltage”
- const string **BYPASS_VOLTS2_UNITS** = ”V”

- const string **BYPASS_VOLTS2_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **BYPASS_VOLTS2**
- const string **BYPASS_VOLTS3_DESCRIPTION** = ”Bypass Phase 3 Voltage”
- const string **BYPASS_VOLTS3_UNITS** = ”V”
- const string **BYPASS_VOLTS3_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **BYPASS_VOLTS3**
- const string **FREQB_DESCRIPTION** = ”Bypass Frequency”
- const string **FREQB_UNITS** = ”Hz”
- const string **FREQB_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **FREQB**
- const string **FREQI_DESCRIPTION** = ”Inverter Frequency”
- const string **FREQI_UNITS** = ”Hz”
- const string **FREQI_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **FREQI**
- const string **INVERTER_SW_DESCRIPTION** = ”Inverter Switch”
- const string **INVERTER_SW_BITDESCRIPTION** = ”Switch, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved”
- const string **INVERTER_SW_WHENCLEARED** = ”OFF,N,N,N,N,N,N,N”
- const string **INVERTER_SW_WHENSET** = ”ON,N,N,N,N,N,N,N”
- const string **INVERTER_SW_UPDATEINT** = ”30”
- readonly attribute ACS::ROpattern **INVERTER_SW**
- const string **INVERTER_VOLTS1_DESCRIPTION** = ”Inverter Phase 1 Voltage”
- const string **INVERTER_VOLTS1_UNITS** = ”V”
- const string **INVERTER_VOLTS1_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **INVERTER_VOLTS1**
- const string **INVERTER_VOLTS2_DESCRIPTION** = ”Inverter Phase 2 Voltage”
- const string **INVERTER_VOLTS2_UNITS** = ”V”
- const string **INVERTER_VOLTS2_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **INVERTER_VOLTS2**
- const string **INVERTER_VOLTS3_DESCRIPTION** = ”Inverter Phase 3 Voltage”
- const string **INVERTER_VOLTS3_UNITS** = ”V”
- const string **INVERTER_VOLTS3_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **INVERTER_VOLTS3**
- const string **OUTPUT_AMP1_DESCRIPTION** = ”Phase 1 Output Current”
- const string **OUTPUT_AMP1_UNITS** = ”A”
- const string **OUTPUT_AMP1_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **OUTPUT_AMP1**
- const string **OUTPUT_AMP2_DESCRIPTION** = ”Phase 2 Output Current”
- const string **OUTPUT_AMP2_UNITS** = ”A”
- const string **OUTPUT_AMP2_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **OUTPUT_AMP2**
- const string **OUTPUT_AMP3_DESCRIPTION** = ”Phase 3 Output Current”
- const string **OUTPUT_AMP3_UNITS** = ”A”
- const string **OUTPUT_AMP3_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **OUTPUT_AMP3**
- const string **OUTPUT_VOLT1_DESCRIPTION** = ”Phase 1 Output Voltage”
- const string **OUTPUT_VOLT1_UNITS** = ”V”
- const string **OUTPUT_VOLT1_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **OUTPUT_VOLT1**
- const string **OUTPUT_VOLT2_DESCRIPTION** = ”Phase 2 Output Voltage”
- const string **OUTPUT_VOLT2_UNITS** = ”V”
- const string **OUTPUT_VOLT2_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **OUTPUT_VOLT2**
- const string **OUTPUT_VOLT3_DESCRIPTION** = ”Phase 3 Output Voltage”
- const string **OUTPUT_VOLT3_UNITS** = ”V”
- const string **OUTPUT_VOLT3_UPDATEINT** = ”30”
- readonly attribute ACS::ROdouble **OUTPUT_VOLT3**

9.10.1 Detailed Description

APEX dummy ACU UPS Interface
The documentation for this interface was generated from the following file:

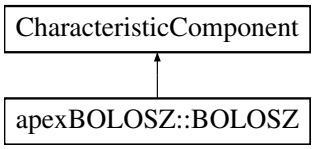
- apexACUUPS.idl

Chapter 10

Specific Devices

10.1 apexBOLOSZ::BOLOSZ Interface Reference

`import"apexBOLOSZ.idl";`
Inheritance diagram for apexBOLOSZ::BOLOSZ:



Public Attributes

- const string **nullingCurrent_DESCRIPTION** = ”Nulling current”
- const string **nullingCurrent_UNITS** = ”A”
- readonly attribute ACS::ROdoubleSeq **nullingCurrent**
- const string **signalToCurrent_DESCRIPTION** = ”Conversion factor to signal amplitude to bolometer current”
- const string **signalToCurrent_UNITS** = ”A”
- readonly attribute ACS::ROdoubleSeq **signalToCurrent**
- const string **currentToPower_DESCRIPTION** = ”Conversion factor to convert bolometer current to observed power”
- const string **currentToPower_UNITS** = ”V”
- readonly attribute ACS::ROdoubleSeq **currentToPower**
- const string **totalPower_DESCRIPTION** = ”Total Power at TES”
- const string **totalPower_UNITS** = ”W”
- readonly attribute ACS::ROdoubleSeq **totalPower**
- const string **biasFrequency_DESCRIPTION** = ”AM carrier bias Frequency”
- const string **biasFrequency_UNITS** = ”Hz”
- readonly attribute ACS::ROdoubleSeq **biasFrequency**
- const string **biasAmplitude_DESCRIPTION** = ”Carrier bias amplitude”
- const string **biasAmplitude_UNITS** = ””
- readonly attribute ACS::ROdoubleSeq **biasAmplitude**
- const string **biasPotSetting_DESCRIPTION** = ”Carrier potentiometer setting--0 minimum, 255 maximum”
- const string **biasPotSetting_UNITS** = ””
- readonly attribute ACS::ROlongSeq **biasPotSetting**
- const string **biasAttenSetting_DESCRIPTION** = ”Carrier attenuation setting”
- const string **biasAttenSetting_UNITS** = ””
- readonly attribute ACS::ROlongSeq **biasAttenSetting**
- const string **QDACamplitude_DESCRIPTION** = ”Quadrature Nulling amplitude”
- const string **QDACamplitude_UNITS** = ””

- readonly attribute ACS::RODoubleSeq **QDACamplitude**
- const string **nullPotSetting_DESCRIPTION** = "Nulling Potentometer setting--0 minimum, 255 maximum"
- const string **nullPotSetting_UNITS** = ""
- readonly attribute ACS::ROlongSeq **nullPotSetting**
- const string **nullAttenSetting_DESCRIPTION** = "Nuller attenuation setting"
- const string **nullAttenSetting_UNITS** = ""
- readonly attribute ACS::ROlongSeq **nullAttenSetting**
- const string **SQUIDBias_DESCRIPTION** = "Amplitude of SQUID Bias Current at Controller DAC"
- const string **SQUIDBias_UNITS** = "V"
- readonly attribute ACS::RODoubleSeq **SQUIDBias**
- const string **SQUIDFluxBias_DESCRIPTION** = "Amplitude of SQUID Bias Flux at Controller DAC"
- const string **SQUIDFluxBias_UNITS** = "V"
- readonly attribute ACS::RODoubleSeq **SQUIDFluxBias**
- const string **SQUIDOffset_DESCRIPTION** = "Amplitude of SQUID Voltage Offset at Controller DAC"
- const string **SQUIDOffset_UNITS** = "V"
- readonly attribute ACS::RODoubleSeq **SQUIDOffset**
- const string **SQUIDHeater_DESCRIPTION** = "Amplitude of SQUID Heater Voltage Offset at Controller DAC"
- const string **SQUIDHeater_UNITS** = "V"
- readonly attribute ACS::RODoubleSeq **SQUIDHeater**
- const string **SQUIDEnabled_DESCRIPTION** = "Enable state of SQUID Amplifier"
- const string **SQUIDEnabled_UNITS** = ""
- readonly attribute ACS::ROlongSeq **SQUIDEnabled**
- const string **SQUIDFeedback_DESCRIPTION** = "Value of SQUID Feeback resistance"
- const string **SQUIDFeedback_UNITS** = "Ohm"
- readonly attribute ACS::RODoubleSeq **SQUIDFeedback**
- const string **demodAtten_DESCRIPTION** = "Demodulator Attenuation"
- const string **demodAtten_UNITS** = ""
- readonly attribute ACS::ROlongSeq **demodAtten**
- const string **demodACCoupling_DESCRIPTION** = "AC Coupling setting for the Demodulator"
- const string **demodACCoupling_UNITS** = ""
- readonly attribute ACS::ROlongSeq **demodACCoupling**

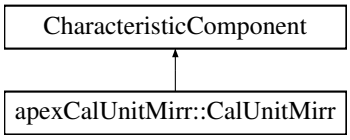
10.1.1 Detailed Description

APEX SZ Receiver Maintenance Interface
Contains all conversion factors required to convert the raw data to physical units.
All attributes are sequencetypes containing one element per frontend Channel
The first four attributes are straightforward conversion factors: nullingCurrent signalToCurrent currentToPower totalPower
The remaining attributes give a detailed description of all software tunable bolometer parameters. In principle the conversion factors above can be generated from this information, though an accurate estimate of each factor requires a detailed model of each channel, including many physical instrument calibration factors.
The conversion factors are therefore provided for convenience.
Using the conversion factors to find changes in optical power:
Delta(P) = Delta(ADC) $*$ signalToCurrent $*$ currentToPower P: optical power received at bolometer ADC: ADC counts sent to FITSWriter
Using the conversion factors to find total optical power:
Total(I) = (-ADC $*$ signalToCurrent) + nullingCurrent Optical(P) = TotalPower - Total(I)$*$currentToPower
The documentation for this interface was generated from the following file:

- apexBOLOSZ.idl

10.2 apexCalUnitMirr::CalUnitMirr Interface Reference

`import"apexCalUnitMirr.idl";`
Inheritance diagram for apexCalUnitMirr::CalUnitMirr:



Public Attributes

- const string rotMirPos_DESCRIPTION = ”Rotating Mirror Position”
- const string **rotMirPos_UNITS** = ”percent”
- const string **rotMirPos_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **rotMirPos**
- const string rotMirStat_DESCRIPTION = ”Rotating Mirror Status”
- const string **rotMirStat_UNITS** = ””
- const string **rotMirStat_ALARMON** = ”6”
- const string **rotMirStat_SAMPLEINTERVAL** = ”60”
- readonly attribute ROMirrStat **rotMirStat**
- const string linMirPos_DESCRIPTION = ”Linear Mirror Position”
- const string **linMirPos_UNITS** = ”percent”
- const string **linMirPos_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **linMirPos**
- const string linMirStat_DESCRIPTION = ”Linear Mirror Status”
- const string **linMirStat_UNITS** = ””
- const string **linMirStat_ALARMON** = ”6”
- const string **linMirStat_SAMPLEINTERVAL** = ”60”
- readonly attribute ROMirrStat **linMirStat**

10.2.1 Detailed Description

This interface belongs to the abstract calibration unit device. It will be the standard interface for APEX Calibration Units.

10.2.2 Member Data Documentation

const string apexCalUnitMirr::CalUnitMirr::linMirPos_DESCRIPTION = ”Linear Mirror Position”

Linear mirror position (percent) 0% = cold position; 100% = hot position

const string apexCalUnitMirr::CalUnitMirr::linMirStat_DESCRIPTION = ”Linear Mirror Status”

Linear mirror status (LOAD, SKY, ERROR, MOVING, UNDEFINED)

const string apexCalUnitMirr::CalUnitMirr::rotMirPos_DESCRIPTION = ”Rotating Mirror Position”

Rotating mirror position (percent) 0% = cold position; 100% = hot position

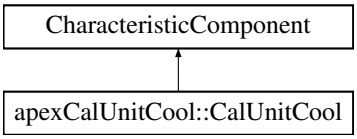
const string apexCalUnitMirr::CalUnitMirr::rotMirStat_DESCRIPTION = ”Rotating Mirror Status”

Rotating mirror status (HOT, COLD, ERROR, MOVING, UNDEFINED)
The documentation for this interface was generated from the following file:

- apexCalUnitMirr.idl

10.3 apexCalUnitCool::CalUnitCool Interface Reference

`import"apexCalUnitCool.idl";`
Inheritance diagram for apexCalUnitCool::CalUnitCool:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)

Public Attributes

- const string status_DESCRIPTION = ”Cooler Status”
- const string **status_UNITS** = ””
- const string **status_ALARMON** = ”2”
- const string **status_SAMPLEINTERVAL** = ”60”
- readonly attribute ROcoolerStatus **status**
- const string coldHeadTemp_DESCRIPTION = ”Cold Head Temperature”
- const string **coldHeadTemp_UNITS** = ”K”
- const string **coldHeadTemp_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **coldHeadTemp**
- const string rejectTemp_DESCRIPTION = ”Reject Temperature”
- const string **rejectTemp_UNITS** = ”degC”
- const string **rejectTemp_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **rejectTemp**
- const string circuitTemp_DESCRIPTION = ”Circuit Temperature”
- const string **circuitTemp_UNITS** = ”degC”
- const string **circuitTemp_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **circuitTemp**
- const string vacuum_DESCRIPTION = ”Vacuum”
- const string **vacuum_UNITS** = ”mbar”
- const string **vacuum_FORMAT** = ”%.7f”
- const string **vacuum_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **vacuum**
- const string current_DESCRIPTION = ”Current”
- const string **current_UNITS** = ”A”
- const string **current_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **current**
- const string voltage_DESCRIPTION = ”Voltage”
- const string **voltage_UNITS** = ”V”
- const string **voltage_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **voltage**
- const string hours_DESCRIPTION = ”Cooler Run Time”
- const string **hours_UNITS** = ”h”
- const string **hours_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **hours**
- const string errorCount_DESCRIPTION = ”Error Count”
- const string **errorCount_UNITS** = ””
- const string **errorCount_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **errorCount**

10.3.1 Detailed Description

This interface belongs to the abstract calibration unit device. It will be the standard interface for APEX Calibration Units.

10.3.2 Member Function Documentation

void apexCalUnitCool::CalUnitCool::off (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn off cooler

void apexCalUnitCool::CalUnitCool::on (in ACS::CBvoid *cb,* in ACS::CBDescIn *desc*)

Turn on cooler

10.3.3 Member Data Documentation

const string apexCalUnitCool::CalUnitCool::circuitTemp_DESCRIPTION = "Circuit Temperature"

Circuit temperature (degrees Celsius)

const string apexCalUnitCool::CalUnitCool::coldHeadTemp_DESCRIPTION = "Cold Head Temperature"

Cold head temperature (K)

const string apexCalUnitCool::CalUnitCool::current_DESCRIPTION = "Current"

Current (A)

const string apexCalUnitCool::CalUnitCool::errorCount_DESCRIPTION = "Error Count"

Error Count ()

const string apexCalUnitCool::CalUnitCool::hours_DESCRIPTION = "Cooler Run Time"

Cooler Run Time (h)

const string apexCalUnitCool::CalUnitCool::rejectTemp_DESCRIPTION = "Reject Temperature"

Reject temperature (K)

const string apexCalUnitCool::CalUnitCool::status_DESCRIPTION = "Cooler Status"

Status of the Calibration Unit Cooler

const string apexCalUnitCool::CalUnitCool::vacuum_DESCRIPTION = "Vacuum"

Vacuum (mbar)

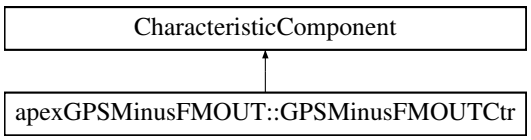
const string apexCalUnitCool::CalUnitCool::voltage_DESCRIPTION = "Voltage"

Voltage (V)
The documentation for this interface was generated from the following file:

- apexCalUnitCool.idl

10.4 apexGPSMinusFMOUT::GPSMinusFMOUTCtr Interface Reference

`import"apexGPSMinusFMOUT.idl";`
Inheritance diagram for apexGPSMinusFMOUT::GPSMinusFMOUTCtr:



Public Attributes

- const string GPSMinusFMOUT_DESCRIPTION = ”GPS-FMOUT”
- const string **GPSMinusFMOUT_UNITS** = ”ns”
- const string **GPSMinusFMOUT_SAMPLEINTERVAL** = ”60”
- const string **GPSMinusFMOUT_ALARMLOWON** = ”-2147483648”
- const string **GPSMinusFMOUT_ALARMLOWOFF** = ”-2147483647”
- const string **GPSMinusFMOUT_ALARMHIGHON** = ”1000”
- const string **GPSMinusFMOUT_ALARMHIGHOFF** = ”950”
- readonly attribute ACS::ROlong **GPSMinusFMOUT**

10.4.1 Detailed Description

APEX GPSMinusFMOUT Counter Interface

10.4.2 Member Data Documentation

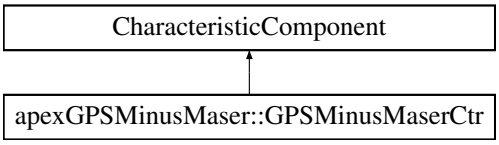
const string apexGPSMinusFMOUT::GPSMinusFMOUTCtr::GPSMinusFMOUT_DESCRIPTION = ”GPS-FMOUT”

GPS - FMOUT
The documentation for this interface was generated from the following file:

- apexGPSMinusFMOUT.idl

10.5 apexGPSMinusMaser::GPSMinusMaserCtr Interface Reference

`import"apexGPSMinusMaser.idl";`
Inheritance diagram for apexGPSMinusMaser::GPSMinusMaserCtr:



Public Attributes

- const string GPSMinusMaser_DESCRIPTION = ”GPS-Maser”
- const string **GPSMinusMaser_UNITS** = ”ns”
- const string **GPSMinusMaser_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROlong **GPSMinusMaser**

10.5.1 Detailed Description

APEX GPS minus Maser Counter Interface

10.5.2 Member Data Documentation

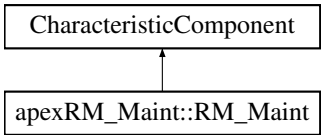
const string apexGPSMinusMaser::GPSMinusMaserCtr::GPSMinusMaser_DESCRIPTION = ”GPS-Maser”

GPS Minus Maser counter
The documentation for this interface was generated from the following file:

- apexGPSMinusMaser.idl

10.6 apexRM_Maint::RM_Maint Interface Reference

`import"apexRM_Maint.idl";`
Inheritance diagram for apexRM_Maint::RM_Maint:



Public Member Functions

- void absCal (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void gainCal (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void noiseCal (in ACS::CBvoid cb, in ACS::CBDescIn desc)

Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- readonly attribute ROstatus **state**
- const string calibMirror_DESCRIPTION = ”Calibration Mirror Position”
- const string **calibMirror_UNITS** = ””
- readonly attribute ROcalMirrorPos **calibMirror**
- const string cmdCalibMirror_DESCRIPTION = ”Commanded Calibration Mirror Position”
- const string **cmdCalibMirror_UNITS** = ””
- readonly attribute RWcalMirrorPos **cmdCalibMirror**
- const string counts_DESCRIPTION = ”Detector Voltages in the 6 IF Bands”
- const string **counts_UNITS** = ”mV”
- readonly attribute ACS::RODoubleSeq **counts**
- const string noiseTB_DESCRIPTION = ”Noise Diode Brightness Temperatures”
- const string **noiseTB_UNITS** = ”K”
- readonly attribute ACS::RODouble **noiseTB**
- const string Trec_DESCRIPTION = ”Receiver Board Temperature”
- const string **Trec_UNITS** = ”degC”
- readonly attribute ACS::RODouble **Trec**
- const string Tamb_DESCRIPTION = ”Ambient Target Temperature”
- const string **Tamb_UNITS** = ”K”
- readonly attribute ACS::RODouble **Tamb**
- const string pressure_DESCRIPTION = ”Atmospheric Pressure”
- const string **pressure_UNITS** = ”hPa”
- readonly attribute ACS::RODouble **pressure**
- const string mode_DESCRIPTION = ”Observation Mode”
- const string **mode_UNITS** = ””
- readonly attribute ROobsMode **mode**
- const string calibMode_DESCRIPTION = ”Calibration Mode”
- const string **calibMode_UNITS** = ””
- readonly attribute ROcalibMode **calibMode**
- const string cmdCalibMode_DESCRIPTION = ”Commanded Calibration Mode”
- const string **cmdCalibMode_UNITS** = ””
- readonly attribute RWcalibMode **cmdCalibMode**
- const string relCalInt_DESCRIPTION = ”Relative Calibration Interval”
- const string **relCalInt_UNITS** = ”s”

- readonly attribute ACS::ROlong **relCalInt**
- const string cmdRelCalInt_DESCRIPTION = ”Commanded Relative Calibration Interval”
- const string **cmdRelCalInt_UNITS** = ”s”
- readonly attribute ACS::RWlong **cmdRelCalInt**
- const string noiseCalInt_DESCRIPTION = ”Noise Calibration Interval”
- const string **noiseCalInt_UNITS** = ”s”
- readonly attribute ACS::ROlong **noiseCalInt**
- const string cmdNoiseCalInt_DESCRIPTION = ”Commanded Noise Calibration Interval”
- const string **cmdNoiseCalInt_UNITS** = ”s”
- readonly attribute ACS::RWlong **cmdNoiseCalInt**
- const string lastCalData_DESCRIPTION = ”Last Calibration Data”
- const string **lastCalData_UNITS** = ””
- readonly attribute ACS::ROdoubleSeq **lastCalData**

10.6.1 Detailed Description

APEX Radiometer Maintenance Interface

10.6.2 Member Function Documentation

void apexRM_Maint::RM_Maint::absCal (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

performs an absolute calibration.
Parameters

cb	callback when completed
desc	description

void apexRM_Maint::RM_Maint::gainCal (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

performs a gain calibration.
Parameters

cb	callback when completed
desc	description

void apexRM_Maint::RM_Maint::noiseCal (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

performs a noise calibration.
Parameters

cb	callback when completed
desc	description

10.6.3 Member Data Documentation

const string apexRM_Maint::RM_Maint::calibMirror_DESCRIPTION = ”Calibration Mirror Position”

actual position of the radiometer Calibration Mirror (LOAD, COLD, MOVING, SKY)

const string apexRM_Maint::RM_Maint::calibMode_DESCRIPTION = ”Calibration Mode”

Actual calibration mode of the Radiometer {AUTO,MANUAL}

const string apexRM_Maint::RM_Maint::cmdCalibMirror_DESCRIPTION = ”Commanded Calibration Mirror Position”

Comanded position of the radiometer Calibration Mirror (LOAD, COLD, SKY, INITIALIZE)

const string apexRM_Maint::RM_Maint::cmdCalibMode_DESCRIPTION = "Commanded Calibration Mode"

comanded calibration mode of the Radiometer {AUTO,MANUAL}

const string apexRM_Maint::RM_Maint::cmdNoiseCalInt_DESCRIPTION = "Commanded Noise Calibration Interval"

comanded interval between two noise calibrations in AUTO mode [s]

const string apexRM_Maint::RM_Maint::cmdRelCalInt_DESCRIPTION = "Commanded Relative Calibration Interval"

comanded interval between two relative calibrations in AUTO mode [s]

const string apexRM_Maint::RM_Maint::counts_DESCRIPTION = "Detector Voltages in the 6 IF Bands"

detector Voltages in the 6 IF bands (mV)

const string apexRM_Maint::RM_Maint::lastCalData_DESCRIPTION = "Last Calibration Data"

Values of the last absolute calibration of the Radiometer gain, systemTemperature, timestamp of last calibration

const string apexRM_Maint::RM_Maint::mode_DESCRIPTION = "Observation Mode"

Actual observation mode of the Radiometer

const string apexRM_Maint::RM_Maint::noiseCalInt_DESCRIPTION = "Noise Calibration Interval"

interval between two noise calibrations in AUTO mode [s]

const string apexRM_Maint::RM_Maint::noiseTB_DESCRIPTION = "Noise Diode Brightness Temperatures"

Brightness temperatures of the noise diode (K)

const string apexRM_Maint::RM_Maint::pressure_DESCRIPTION = "Atmospheric Pressure"

atmospheric Pressure (mbar)

const string apexRM_Maint::RM_Maint::relCalInt_DESCRIPTION = "Relative Calibration Interval"

interval between two relative calibrations in AUTO mode [s]

const string apexRM_Maint::RM_Maint::state_DESCRIPTION = "State"

Actual State of the Radiometer

const string apexRM_Maint::RM_Maint::Tamb_DESCRIPTION = "Ambient Target Temperature"

Temperature of the ambient Target (K)

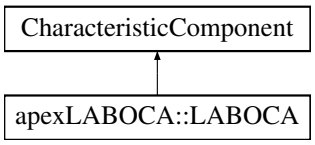
const string apexRM_Maint::RM_Maint::Trec_DESCRIPTION = "Receiver Board Temperature"

temperature of the receiver board (degree Celcius)
The documentation for this interface was generated from the following file:

- apexRM_Maint.idl

10.7 apexLABOCA::LABOCA Interface Reference

`import"apexLABOCA.idl";`
Inheritance diagram for apexLABOCA::LABOCA:



Public Member Functions

- void on (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void off (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- void configure (in ACS::CBvoid cb, in ACS::CBDescIn desc)
- long startRecycle () raises (ACSErr::ACSException)
- long stopRecycle () raises (ACSErr::ACSException)
- long reset () raises (ACSErr::ACSException)

Public Attributes

- const string he4Switch_DESCRIPTION = ”Thermal Switch He4 Pump”
- const string **he4Switch_UNITS** = ””
- const string **he4Switch_SAMPLEINTERVAL** = ”180”
- readonly attribute ROlogical **he4Switch**
- const string cmdHe4Switch_DESCRIPTION = ”Commanded Thermal Switch He4 Pump”
- const string **cmdHe4Switch_UNITS** = ””
- readonly attribute RWlogical **cmdHe4Switch**
- const string he3Switch_DESCRIPTION = ”Thermal Switch He3 Pump”
- const string **he3Switch_UNITS** = ””
- const string **he3Switch_SAMPLEINTERVAL** = ”180”
- readonly attribute ROlogical **he3Switch**
- const string cmdHe3Switch_DESCRIPTION = ”Commanded Thermal Switch He3 Pump”
- const string **cmdHe3Switch_UNITS** = ””
- readonly attribute RWlogical **cmdHe3Switch**
- const string he4Heater_DESCRIPTION = ”Heater He4 Pump”
- const string **he4Heater_UNITS** = ””
- const string **he4Heater_SAMPLEINTERVAL** = ”180”
- readonly attribute ROlogical **he4Heater**
- const string cmdHe4Heater_DESCRIPTION = ”Commanded Heater He4 Pump”
- const string **cmdHe4Heater_UNITS** = ””
- readonly attribute RWlogical **cmdHe4Heater**
- const string he3Heater_DESCRIPTION = ”Heater He3 Pump”
- const string **he3Heater_UNITS** = ””
- const string **he3Heater_SAMPLEINTERVAL** = ”180”
- readonly attribute ROlogical **he3Heater**
- const string cmdHe3Heater_DESCRIPTION = ”Commanded Heater He3 Pump”
- const string **cmdHe3Heater_UNITS** = ””
- readonly attribute RWlogical **cmdHe3Heater**
- const string he4PumpTempLimit_DESCRIPTION = ”Limit Temperature of He4 Pump”
- const string **he4PumpTempLimit_UNITS** = ”K”
- const string **he4PumpTempLimit_FORMAT** = ”%.3f”
- const string **he4PumpTempLimit_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **he4PumpTempLimit**

- const string cmdHe4PumpTempLimit_DESCRIPTION = ”Commanded Limit Temperature of He4 Pump”
- const string **cmdHe4PumpTempLimit_UNITS** = ”K”
- const string **cmdHe4PumpTempLimit_FORMAT** = ”%.3f”
- readonly attribute ACS::ROdouble **cmdHe4PumpTempLimit**
- const string he3PumpTempLimit_DESCRIPTION = ”Limit Temperature of He3 Pump”
- const string **he3PumpTempLimit_UNITS** = ”K”
- const string **he3PumpTempLimit_FORMAT** = ”%.3f”
- const string **he3PumpTempLimit_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **he3PumpTempLimit**
- const string cmdHe3PumpTempLimit_DESCRIPTION = ”Commanded Limit Temperature of He3 Pump”
- const string **cmdHe3PumpTempLimit_UNITS** = ”K”
- const string **cmdHe3PumpTempLimit_FORMAT** = ”%.3f”
- readonly attribute ACS::ROdouble **cmdHe3PumpTempLimit**
- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”180”
- readonly attribute ROstatus **state**
- const string he4Temp_DESCRIPTION = ”Helium 4 Temperature”
- const string **he4Temp_UNITS** = ”K”
- const string **he4Temp_FORMAT** = ”%.3f”
- const string **he4Temp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **he4Temp**
- const string he3Temp_DESCRIPTION = ”Helium 3 Temperature”
- const string **he3Temp_UNITS** = ”K”
- const string **he3Temp_FORMAT** = ”%.3f”
- const string **he3Temp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **he3Temp**
- const string he4PumpTemp_DESCRIPTION = ”Sorption Pump He4 Temperature”
- const string **he4PumpTemp_UNITS** = ”K”
- const string **he4PumpTemp_FORMAT** = ”%.3f”
- const string **he4PumpTemp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **he4PumpTemp**
- const string he3PumpTemp_DESCRIPTION = ”Sorption Pump He3 Temperature”
- const string **he3PumpTemp_UNITS** = ”K”
- const string **he3PumpTemp_FORMAT** = ”%.3f”
- const string **he3PumpTemp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **he3PumpTemp**
- const string he4SwitchTemp_DESCRIPTION = ”Thermal Switch He4 Pump Temperature”
- const string **he4SwitchTemp_UNITS** = ”K”
- const string **he4SwitchTemp_FORMAT** = ”%.3f”
- const string **he4SwitchTemp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **he4SwitchTemp**
- const string he3SwitchTemp_DESCRIPTION = ”Thermal Switch He3 Pump Temperature”
- const string **he3SwitchTemp_UNITS** = ”K”
- const string **he3SwitchTemp_FORMAT** = ”%.3f”
- const string **he3SwitchTemp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **he3SwitchTemp**
- const string pTubeFirstTemp_DESCRIPTION = ”Pulse Tube First Stage Temperature”
- const string **pTubeFirstTemp_UNITS** = ”K”
- const string **pTubeFirstTemp_FORMAT** = ”%.3f”
- const string **pTubeFirstTemp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::ROdouble **pTubeFirstTemp**
- const string pTubeSecondTemp_DESCRIPTION = ”Pulse Tube Second Stage Temperature”

- const string **pTubeSecondTemp_UNITS** = ”K”
- const string **pTubeSecondTemp_FORMAT** = ”%.3f”
- const string **pTubeSecondTemp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::RODouble **pTubeSecondTemp**
- const string boloFirstTemp_DESCRIPTION = ”Bolometer Mount First Stage Temperature”
- const string **boloFirstTemp_UNITS** = ”K”
- const string **boloFirstTemp_FORMAT** = ”%.3f”
- const string **boloFirstTemp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::RODouble **boloFirstTemp**
- const string boloSecondTemp_DESCRIPTION = ”Bolometer Mount Second Stage Temperature”
- const string **boloSecondTemp_UNITS** = ”K”
- const string **boloSecondTemp_FORMAT** = ”%.3f”
- const string **boloSecondTemp_SAMPLEINTERVAL** = ”180”
- readonly attribute ACS::RODouble **boloSecondTemp**
- const string version_DESCRIPTION = ”Version”
- const string **version_UNITS** = ””
- readonly attribute ACS::ROstring **version**

10.7.1 Detailed Description

This interface belongs to the abstract Continuum Front End device. It will be the standard interface for APEX Continuum Front End Units.

10.7.2 Member Function Documentation

void apexLABOCA::LABOCA::configure (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Set up the continuum frontend according to the high-level properties that are currently defined. The embedded system must keep track of those values.
Parameters

cb	callback when completed
desc	description

void apexLABOCA::LABOCA::off (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn off any parts of the instrument for which power is under software control.
Parameters

cb	callback when completed
desc	description

void apexLABOCA::LABOCA::on (in ACS::CBvoid *cb*, in ACS::CBDescIn *desc*)

Turn on any parts of the instrument for which power is under software control, and initialise by setting defaults.
Parameters

cb	callback when completed
desc	description

long apexLABOCA::LABOCA::reset ( ) raises ACSErr::ACSException)

Hardware reset (restart system) and set defaults.

long apexLABOCA::LABOCA::startRecycle ( ) raises ACSErr::ACSException)

Start the automatic Recycle of the He4-He3 cooler

long apexLABOCA::LABOCA::stopRecycle ( ) raises ACSErr::ACSException)

Stop the automatic Recycle of the He4-He3 cooler

10.7.3 Member Data Documentation

const string apexLABOCA::LABOCA::boloFirstTemp_DESCRIPTION = "Bolometer Mount First Stage Temperature"

Bolometer Mount - First Stage Temperature (K)

const string apexLABOCA::LABOCA::boloSecondTemp_DESCRIPTION = "Bolometer Mount Second Stage Temperature"

Bolometer Mount - Second Stage Temperature (K)

const string apexLABOCA::LABOCA::cmdHe3Heater_DESCRIPTION = "Commanded Heater He3 Pump"

Heater of the Helium-3 Pump - Commanded Status (On/Off)

const string apexLABOCA::LABOCA::cmdHe3PumpTempLimit_DESCRIPTION = "Commanded Limit Temperature of He3 Pump"

Commanded Limit of the Helium-3 Pump Temperature (K)

const string apexLABOCA::LABOCA::cmdHe3Switch_DESCRIPTION = "Commanded Thermal Switch He3 Pump"

Commanded Thermal Switch of the Helium-3 Pump (On/Off)

const string apexLABOCA::LABOCA::cmdHe4Heater_DESCRIPTION = "Commanded Heater He4 Pump"

Commanded Heater of the Helium-4 Pump (On/Off)

const string apexLABOCA::LABOCA::cmdHe4PumpTempLimit_DESCRIPTION = "Commanded Limit Temperature of He4 Pump"

Commanded Limit of the Helium-4 Pump Temperature (K)

const string apexLABOCA::LABOCA::cmdHe4Switch_DESCRIPTION = "Commanded Thermal Switch He4 Pump"

Commanded Thermal Switch of the Helium-4 Pump (On/Off)

const string apexLABOCA::LABOCA::he3Heater_DESCRIPTION = "Heater He3 Pump"

Heater of the Helium-3 Pump - Status (On/Off)

const string apexLABOCA::LABOCA::he3PumpTemp_DESCRIPTION = "Sorption Pump He3 Temperature"

Sorption Pump of the Helium-3 Temperature (K)

const string apexLABOCA::LABOCA::he3PumpTempLimit_DESCRIPTION = "Limit Temperature of He3 Pump"

Limit of the Helium-3 Pump Temperature (K)

const string apexLABOCA::LABOCA::he3Switch_DESCRIPTION = ”Thermal Switch He3 Pump”

Thermal Switch of the Helium-3 Pump (On/Off)

const string apexLABOCA::LABOCA::he3SwitchTemp_DESCRIPTION = ”Thermal Switch He3 Pump Temperature”

Thermal Switch of the Helium-3 Pump Temperature (K)

const string apexLABOCA::LABOCA::he3Temp_DESCRIPTION = ”Helium 3 Temperature”

Evaporator of the Helium-3 Temperature (K)

const string apexLABOCA::LABOCA::he4Heater_DESCRIPTION = ”Heater He4 Pump”

Heater of the Helium-4 Pump (On/Off)

const string apexLABOCA::LABOCA::he4PumpTemp_DESCRIPTION = ”Sorption Pump He4 Temperature”

Sorption Pump of the Helium-4 Temperature (K)

const string apexLABOCA::LABOCA::he4PumpTempLimit_DESCRIPTION = ”Limit Temperature of He4 Pump”

Limit of the Helium-4 Pump Temperature (K)

const string apexLABOCA::LABOCA::he4Switch_DESCRIPTION = ”Thermal Switch He4 Pump”

Thermal Switch of the Helium-4 Pump (On/Off)

const string apexLABOCA::LABOCA::he4SwitchTemp_DESCRIPTION = ”Thermal Switch He4 Pump Temperature”

Thermal Switch of the Helium-4 Pump Temperature (K)

const string apexLABOCA::LABOCA::he4Temp_DESCRIPTION = ”Helium 4 Temperature”

Evaporator of the Helium-4 Temperature (K)

const string apexLABOCA::LABOCA::pTubeFirstTemp_DESCRIPTION = ”Pulse Tube First Stage Temperature”

Pulse Tube Cooler - First Stage Temperature (K)

const string apexLABOCA::LABOCA::pTubeSecondTemp_DESCRIPTION = ”Pulse Tube Second Stage Temperature”

Pulse Tube Cooler - Second Stage Temperature (K)

const string apexLABOCA::LABOCA::state_DESCRIPTION = ”State”

Actual State

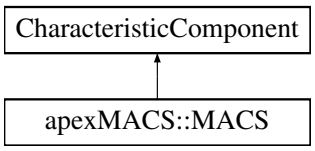
const string apexLABOCA::LABOCA::version_DESCRIPTION = ”Version”

Software version
The documentation for this interface was generated from the following file:

- apexLABOCA.idl

10.8 apexMACS::MACS Interface Reference

`import"apexMACS.idl";`
Inheritance diagram for apexMACS::MACS:



Public Attributes

- const string samplerControl_DESCRIPTION = ”Sampler Control”
- const string **samplerControl_UNITS** = ””
- const string **samplerControl_SAMPLEINTERVAL** = ”180”
- readonly attribute ROlogical **samplerControl**
- const string cmdSamplerControl_DESCRIPTION = ”Commanded Sampler Control”
- const string **cmdSamplerControl_UNITS** = ””
- readonly attribute RWlogical **cmdSamplerControl**

10.8.1 Detailed Description

Special interface to monitor the state of the MACS correlator.

10.8.2 Member Data Documentation

const string apexMACS::MACS::cmdSamplerControl_DESCRIPTION = ”Commanded Sampler Control”

Commanded sampler control setting

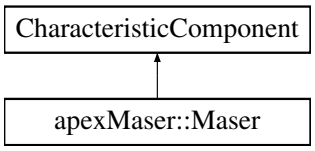
const string apexMACS::MACS::samplerControl_DESCRIPTION = ”Sampler Control”

Sampler control setting
The documentation for this interface was generated from the following file:

- apexMACS.idl

10.9 apexMaser::Maser Interface Reference

`import"apexMaser.idl";`
Inheritance diagram for apexMaser::Maser:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”300”
- readonly attribute ROstatus **state**
- const string U405_DESCRIPTION = ”1420 MHz Output Power”
- const string **U405_UNITS** = ””
- const string **U405_SAMPLEINTERVAL** = ”300”
- readonly attribute ACS::ROdouble **U405**
- const string lock_DESCRIPTION = ”PLL Lock Status”
- const string **lock_UNITS** = ””
- const string **lock_SAMPLEINTERVAL** = ”300”
- readonly attribute ROlogical **lock**
- const string boxTemperature_DESCRIPTION = ”Temperature”
- const string **boxTemperature_UNITS** = ”degC”
- const string **boxTemperature_SAMPLEINTERVAL** = ”300”
- readonly attribute ACS::ROdouble **boxTemperature**
- const string ambientTemperature_DESCRIPTION = ”Temperature”
- const string **ambientTemperature_UNITS** = ”degC”
- const string **ambientTemperature_SAMPLEINTERVAL** = ”300”
- readonly attribute ACS::ROdouble **ambientTemperature**

10.9.1 Detailed Description

APEX Maser Interface

10.9.2 Member Data Documentation

const string apexMaser::Maser::ambientTemperature_DESCRIPTION = ”Temperature”

Ambient temperature (degrees Celsius)

const string apexMaser::Maser::boxTemperature_DESCRIPTION = ”Temperature”

Box temperature (degrees Celsius)

const string apexMaser::Maser::lock_DESCRIPTION = ”PLL Lock Status”

PLL lock status

const string apexMaser::Maser::state_DESCRIPTION = ”State”

Actual State of the Maser

const string apexMaser::Maser::U405_DESCRIPTION = ”1420 MHz Output Power”

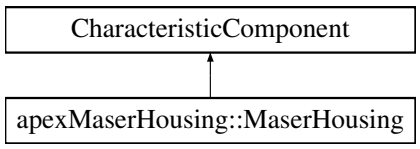
1420 MHz output power as health check. Range: 0.0 to 10.0 (?)
The documentation for this interface was generated from the following file:

- apexMaser.idl

10.10 apexMaserHousing::MaserHousing Interface Reference

```
import"apexMaserHousing.idl";
```

Inheritance diagram for apexMaserHousing::MaserHousing:



Public Attributes

- const string state_DESCRIPTION = ”State”
- const string **state_UNITS** = ””
- const string **state_ALARMON** = ”4”
- const string **state_SAMPLEINTERVAL** = ”60”
- readonly attribute ROstatus **state**
- const string subState_DESCRIPTION = ”Substate”
- const string **subState_UNITS** = ””
- const string **subState_ALARMON** = ”2”
- const string **subState_SAMPLEINTERVAL** = ”60”
- readonly attribute ROsubStatus **subState**
- const string status_DESCRIPTION = ”Status”
- const string **status_BITDESCRIPTION** = ”Alarm 1: Temperature too low, Alarm 2: Humidity too low, Alarm 3: Temperature too high, Alarm 4: Humidity too high, Alarm 5: Open control loop, Alarm 6: Low water, Alarm 7: System fault, Reserved”
- const string **status_WHENCLEARED** = ”N,N,N,N,N,N,N,N”
- const string **status_WHENSET** = ”F,F,F,F,F,F,F,N”
- const string **status_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROpattern **status**
- const string temperature_DESCRIPTION = ”Temperature”
- const string **temperature_UNITS** = ”degC”
- const string **temperature_SAMPLEINTERVAL** = ”60”
- const string **temperature_ALARMLOWON** = ”16.0”
- const string **temperature_ALARMLOWOFF** = ”16.5”
- const string **temperature_ALARMHIGHON** = ”30.0”
- const string **temperature_ALARMHIGHOFF** = ”29.5”
- readonly attribute ACS::ROdouble **temperature**
- const string cmdTemperature_DESCRIPTION = ”Commanded Temperature”
- const string **cmdTemperature_UNITS** = ”degC”
- readonly attribute ACS::RWdouble **cmdTemperature**
- const string tempDutyCycle_DESCRIPTION = ”Temperature Duty Cycle”
- const string **tempDutyCycle_UNITS** = ”%”
- const string **tempDutyCycle_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **tempDutyCycle**
- const string humidity_DESCRIPTION = ”Humidity”
- const string **humidity_UNITS** = ”percent”
- const string **humidity_SAMPLEINTERVAL** = ”60”
- const string **humidity_ALARMLOWON** = ”8.0”
- const string **humidity_ALARMLOWOFF** = ”9.0”
- const string **humidity_ALARMHIGHON** = ”90.0”
- const string **humidity_ALARMHIGHOFF** = ”88.0”
- readonly attribute ACS::ROdouble **humidity**
- const string cmdHumidity_DESCRIPTION = ”Commanded Humidity”

- const string **cmdHumidity_UNITS** = ”degC”
- readonly attribute ACS::RWdouble **cmdHumidity**
- const string humidDutyCycle_DESCRIPTION = ”Humidity Duty Cycle”
- const string **humidDutyCycle_UNITS** = ”%”
- const string **humidDutyCycle_SAMPLEINTERVAL** = ”60”
- readonly attribute ACS::ROdouble **humidDutyCycle**

10.10.1 Detailed Description

APEX Maser Housing Interface

10.10.2 Member Data Documentation

const string apexMaserHousing::MaserHousing::cmdHumidity_DESCRIPTION = ”Commanded Humidity”

Commanded relative humidity (%)

const string apexMaserHousing::MaserHousing::cmdTemperature_DESCRIPTION = ”Commanded Temperature”

Commanded temperature (degrees Celsius)

const string apexMaserHousing::MaserHousing::humidDutyCycle_DESCRIPTION = ”Humidity Duty Cycle”

Relative humidity duty cycle (%)

const string apexMaserHousing::MaserHousing::humidity_DESCRIPTION = ”Humidity”

Relative humidity (%)

const string apexMaserHousing::MaserHousing::state_DESCRIPTION = ”State”

Actual State of the Maser Housing

const string apexMaserHousing::MaserHousing::status_DESCRIPTION = ”Status”

Status bits

const string apexMaserHousing::MaserHousing::subState_DESCRIPTION = ”Substate”

Actual Substate of the Maser Housing

const string apexMaserHousing::MaserHousing::tempDutyCycle_DESCRIPTION = ”Temperature Duty Cycle”

Temperature duty cycle (%)

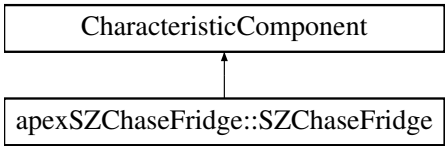
const string apexMaserHousing::MaserHousing::temperature_DESCRIPTION = ”Temperature”

Temperature (degrees Celsius)
The documentation for this interface was generated from the following file:

- apexMaserHousing.idl

10.11 apexSZChaseFridge::SZChaseFridge Interface Reference

`import"apexSZChaseFridge.idl";`
Inheritance diagram for apexSZChaseFridge::SZChaseFridge:



Public Attributes

- const string **uStageTemp_DESCRIPTION** = ”Temperature of the U-stage”
- const string **uStageTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **uStageTemp**
- const string **iStageTemp_DESCRIPTION** = ”Temperature of the I-stage”
- const string **iStageTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **iStageTemp**
- const string **Plate4KTemp_DESCRIPTION** = ”Temperature of the 4K plate”
- const string **Plate4KTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **Plate4KTemp**
- const string **Plate50KTemp_DESCRIPTION** = ”Temperature of the 50K plate”
- const string **Plate50KTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **Plate50KTemp**
- const string **PTCHead4KTemp_DESCRIPTION** = ”Temperature of the 4K PTC head”
- const string **PTCHead4KTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **PTCHead4KTemp**
- const string **PTCHead50KTemp_DESCRIPTION** = ”Temperature of the 50K PTC head”
- const string **PTCHead50KTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **PTCHead50KTemp**
- const string **uHeadTemp_DESCRIPTION** = ”Temperature of the U-head”
- const string **uHeadTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **uHeadTemp**
- const string **uPumpTemp_DESCRIPTION** = ”Temperature of the U-head Pump”
- const string **uPumpTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **uPumpTemp**
- const string **uPumpCurrent_DESCRIPTION** = ”Current in the U-head pump”
- const string **uPumpCurrent_UNITS** = ”A”
- readonly attribute ACS::ROdouble **uPumpCurrent**
- const string **uSwitchState_DESCRIPTION** = ”State of the heat switch for the U-head pump”
- const string **uSwitchState_UNITS** = ””
- readonly attribute ACS::ROlong **uSwitchState**
- const string **uSwitchTemp_DESCRIPTION** = ”Temperature of the U-head heat switch”
- const string **uSwitchTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **uSwitchTemp**
- const string **iHeadTemp_DESCRIPTION** = ”Temperature of the I-head”
- const string **iHeadTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **iHeadTemp**
- const string **iPumpTemp_DESCRIPTION** = ”Temperature of the I-head Pump”
- const string **iPumpTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **iPumpTemp**
- const string **iPumpCurrent_DESCRIPTION** = ”Current in the I-head pump”
- const string **iPumpCurrent_UNITS** = ”A”
- readonly attribute ACS::ROdouble **iPumpCurrent**

- const string **iSwitchState_DESCRIPTION** = ”State of the heat switch for the I-head pump”
- const string **iSwitchState_UNITS** = ””
- readonly attribute ACS::ROlong **iSwitchState**
- const string **iSwitchTemp_DESCRIPTION** = ”Temperature of the I-head heat switch”
- const string **iSwitchTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **iSwitchTemp**
- const string **hExTemp_DESCRIPTION** = ”Temperature of the Heat Exchanger, or \”hex\””
- const string **hExTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **he4ExTemp**
- const string **he4PumpTemp_DESCRIPTION** = ”Temperature of the I-head Pump”
- const string **he4PumpTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **he4PumpTemp**
- const string **he4PumpCurrent_DESCRIPTION** = ”Current in the I-head pump”
- const string **he4PumpCurrent_UNITS** = ”A”
- readonly attribute ACS::ROdouble **he4PumpCurrent**
- const string **he4SwitchState_DESCRIPTION** = ”State of the heat switch for the U-head pump”
- const string **he4SwitchState_UNITS** = ””
- readonly attribute ACS::ROlong **he4SwitchState**
- const string **he4SwitchTemp_DESCRIPTION** = ”Temperature of the I-head heat switch”
- const string **he4SwitchTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **he4SwitchTemp**
- const string **mainplateTemp_DESCRIPTION** = ”Mainplate temperature ”
- const string **mainplateTemp_UNITS** = ”K”
- readonly attribute ACS::ROdouble **mainplateTemp**

10.11.1 Detailed Description

APEX SZ Receiver: Simon Chase 3-stage Sorption Fridge interface
The SZ-receiver is cooled by a 3-stage sorption fridge, manufactured by Chase Research. Each stage cools by maintaining a bath of LHe at a reduced pressure, using charcoal pumps. Of the three stages, one is a 4He stage, the other two are 3He stages. The colder of the two 3He stages is called the ”Ultra”-stage (or u-stage), while the other 3He stage is referred to as the ”Inter”-stage (or i-stage).
For each stage we must monitor 5 quantities:

- The temperature of the cooling head
- The temperature of charcoal pump
- The current being applied to heat the pump
- The state of the heat switch used to cool the pump. When the heatswitch is on the pump cools to the mainplate temperature.
- The temperature of the heat switch, for diagnostic purposes.

There is one other quantity which we measure: the temperature of the ”main plate”, which provides a bath for cooling the pumps.
Typically the mainplate operates near 4K.
NOTE: The actual temperature of the bolometers is not recorded here, it is recorded as ”dewarTemp” under the apexCFE interface
The documentation for this interface was generated from the following file:

- apexSZChaseFridge.idl