E U R O P E A N S O U T H E R N O B S E R V A T ORY

Organisation Européenne pour des Recherches Astronomiques dans l'Hémisphère Austral

Europäische Organisation für astronomische Forschung in der südlichen Hemisphäre

VLT PROGRAMME

VERY LARGE TELESCOPE

VLTI
PRIMA Metrology Control Software
Detailed Design Description

Doc. No.: VLT-SPE-ESO-15736-3384

Issue: 3.0

Date: 08/12/2010

	Name	Date	Signature
Prepared:	R. Frahm		
Approved:	S. Leveque/F. Delplancke		
Released:	R. Gilmozzi		

CHANGE RECORD

ISSUE	DATE	SECTION/PAGE AFFECTED	REASON/INITIATION DOCUMENTS/REMARKS
1.0 prep. 1	2004-07-21	All	First Preparation
1.0	2004-08-09	p. 6: list of AD's	Changes due to comment list from FDR
2.0	2008-04-29	All	Completely rewritten, to reflect the current status of PRIMET for the PRIMA PAE.
3.0	2010-12-08	All	Updated with all changes during PRIMA commissionings 1 to 13.

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

1 INTRODUCTION

1.1 Purpose

This document describes the detailed design and architecture of the PRIMA Metrology Control Software (PMCS) running on the PRIMA Workstation and the Metrology Alignment and Measurement LCUs. PMCS shall implement the architecture outlined in the PMCS System Design Description [AD 07] in accordance with the VLT and VLTI software requirement specifications [AD 01, AD 02].

The present document identifies and describes the major packages and modules, describes general design considerations and explains special design features.

1.2 Scope

The document is released for the VLT archive at specified project milestones, with a release number and a release date, to document the history of the project and for reference. The UML diagrams in this document were generated with Rational Rose. The Rose model is available within the module `primadoc` from the CMM archive.

1.3 Applicable Documents

The following documents, of the issue shown if specified, form part of this manual to the extent specified herein. In the event of conflict between this document and those referenced, the content of this document shall be considered as a superseding requirement unless explicitly stated otherwise herein.

Ref	Document Number	Issue	Date	Title
[AD 01]	VLT-SPE-ESO-10000-0011	2.0	1992-09-30	VLT Software Requirements Specification
[AD 02]	VLT-MAN-ESO-17210-0667	1.1	2001-02-20	Guidelines for the Development of VLT Application Software
[AD 03]	VLT-PRO-ESO-10000-0228	1.0	1993-03-10	VLT Software Programming Standards
[AD 04]	(Internal Document)			VLTI PRIMA Software Project Plan
[AD 05]	VLT-MAN-ESO-17200-0908	1.6	2002-10-24	Tools for Automated Testing User Manual
[AD 06]	VLT-SPE-ESO-15400-0886	2.0	1996-12-18	VLTI Software - Requirements Specification
[AD 07]	VLT-SPE-ESO-15736-2998	1.0	2003-09-16	VLTI PRIMA Metrology Control Software System Design Description
[AD 08]	VLT-SPE-ESO-15736-2953	1.0	2003-09-11	VLTI PRIMA Metrology Control Software Requirements Specification
[AD 09]	GH047	05.97		Newport OMEGA CN77000 Handbuch
[AD 10]		2.6	10/2003	Model SR844 RF Lock-In Amplifier User's Manual
[AD 11]	330044	Rev. A		EO Modulator Driver and Source, Models 3363-A, 3363-B, and 3363-C User's Guide
[AD 12]	VLT-MAN-ESO-17210-2970	2.0	2004-03-21	VLT Common Software – Tools for Advanced Control User Manual

[AD 13]	VLT-TRE-ESO-15735-2963	1.0	2004-01-12	Design of the control electronics of the PRIMA Metrology System
[AD 14]	VLT-ICD-ESO-15736-3060	4.2	2008-08-18	PRIMA Supervisor Software Interface Control Document
[AD 15]	VLT-ICD-ESO-15730-2999	1.1	2003-09-15	PRIMA Interface Control Document between the electro-mechanical Hardware and the Software of the Metrology System
[AD 16]	VLT-TRE-ESO-15736-3034	1.0	2003-04-07	PRIMA Metrology Control Software – Prototypes and Performance Analysis
[AD 17]	GEN-SPE-ESO-00000-0266	1.0	1993-05-10	ESO Graphical User Interface Common Conventions
[AD 18]	VLT-MAN-ESO-17210-2252	3.0	2003-10-15	LCU Common Software – LCU Server Framework User Manual
[AD 19]			1997-03-04	LIGHTWAVE Electronics, Diode-Pumped, Fiber-Coupled, Non-Planar Ring Laser, User’s Manual
[AD 20]	VLT-SPE-ESO-15410-1957	4	2004-05-24	VLTI, Final Lay-out of VLTI Control LANs
[AD 21]	VLT-TRE-ESO-15730-3000	1.0	2003-11-21	Design Description of the PRIMA Metrology System.
[AD 22]	VLT-SPE-ESO-15736-3243	3.0	2008-04-29	PRIMA FSU – Software Design Description
[AD 23]	VLT-SPE-ESO-15736-3186	2.12	2010-04-12	VLTI PRIMA Star Separator Control Software System Design Specification
[AD 24]	VLT-TRE-IMT-15734-3002	3	2003-08-28	Design of the Phase Meter of the PRIMA Metrology System
[AD 25]	VLT-TRE-IMT-15731-3154	4	2003-10-08	Design of the Laser Assembly of the PRIMA Metrology System
[AD 26]	VLT-SPE-ESO-15732-4087	1.0	2008-04-02	Design of the Pupil Tracker for PRIMET
[AD 27]	VLT-SPE-ESO-15736-3899	1.0	2008-04-14	Specifications for the PRIMA Metrology data files, data logging and algorithms
[AD 28]	VLT-SPE-ESO-15735-4544	1.0	2008-04-01	Design of the PRIMA Metrology Laser Interlock System Technical Manual

Table 1 – Applicable documents

1.4 Reference Documents

The reference documents contain background information required to fully understand the structure of the PRIMET documents, the terminology used, the software environment in which the PMCS shall be integrated and the interface characteristics to the external systems.

Ref	Title	Author	Edition
[RD 01]	The Unified Modeling Language User Guide, Addison Wesley	I. Jacobson, G. Booch, J. Rumbaugh	1998
[RD 02]	The Elements of UML Style	S. W. Ambler	2003
[RD 03]	Design Patterns: Elements of Reusable Object-Oriented Software	E. Gamma, R. Helm, R. Johnson, J. Vlissides	March 2002

Table 2 – Reference documents

1.5 Abbreviations and Acronyms

This document employs several abbreviations and acronyms to refer concisely to an item, after it has been introduced. The following list is aimed to help the reader in recalling the extended meaning of each short expression:

λ_0	Working wavelength
AO	Adaptive Optics
ARAL	Artificial sources and Alignment Toolkit
CCS	Central Control Software
DDL	Differential Delay Line
DL	Delay Line
DLCS	Delay Line Control Software
dOPDC	Differential Optical Path Difference Controller
ESO	European Southern Observatory
FSU	Fringe Sensor Unit
GD	Group Delay
GDT	Group Delay Tracking
GUI	Graphical User Interface
IEE	Institute of Electrical Engineer
IEEE	Institute of Electrical and Electronic Engineers
ISS	Interferometer Supervisor Software
LCC	LCU Common Software
LCU	Local Control Unit
LEO	Leonardo: VLTI artificial star
NTP	Network Time Protocol
OLDB	Online Database
OPD	Optical Path Difference
OPDC	Optical Path Difference Controller
OPL	Optical Path Length
OSLX	Objective SLX
PDCS	PRIMA Differential delay line Control Software
PMCS	PRIMA Metrology Control Software
PRICS	PRIMA Control Software
PRIMA	Phase Referenced Imaging and Microarcsecond Astrometry facility
PRIMET	PRIMA Metrology sub-system
PSS	PRIMA Supervisor Software
RMN	Reflective Memory Network
SLX	Setup Files and operations Logs Handling
TIM	Time Interface Module
UML	Unified Modeling Language
UT	Unit Telescope
UTC	Universal Time Coordinates
VLT	Very Large Telescope
VLTI	Very Large Telescope Interferometer
VLTICS	Very Large Telescope Interferometer Control Software
VME	VERSA Module Euro card
WS	Workstation

1.6 Glossary

This glossary defines those terms (single words or phrases) concerning Software Engineering, Telescope Control and Interferometry, mentioned in this document, but firstly introduced and extensively described in other documents. The meaning of each term is carefully explained, focusing on its usage in the context of this specific document. Words, belonging to a term, which are enclosed in brackets, are implicit when no ambiguity can arise. Those words contained in the definition of a term and included also in the glossary, which are used with a specific technical meaning, are printed in italics. Entries are ordered alphabetically.

- **Actor**: An actor is a role of an entity external to the system. Actors can be humans, machines, or devices. One physical object may play several roles and therefore be modeled by several actor. A primary actor is one having a goal requiring the assistance of the system. A secondary actor is one from which the system needs assistance to satisfy its goal.
- **Beam combiner**: an optical system which combines the light beams coming from different telescopes used for interferometry.
- **Channel**: a channel consists of two light beams of the same object from two telescopes. There are two Fringe Sensor Units, each handling one channel: channel B for the first object and channel A for the second object.
- **Database attribute**: A location, uniquely identified by a name, within the WS or the LCU real-time database, where a value is stored.**Dual feed**: an optical system, located at the telescope focus, capable to select two narrow field-of-view beams.
- **Exposure time**: the time during which the photons are accumulated in an exposure.
- **Function**: A defined objective or characteristic action of a system or component.
- **Functional requirement**: A requirement that specifies a function that a system or component must be able to perform.
- **Functional specification**: A document that specifies the functions that a system or component must perform.
- **Mode**: A condition of existence that a system, subsystem, or component, may be in. Normally it comprises a set of possible states.
- **Module**: See software module.
- **Optical path difference**: is the sum of the external optical path difference and the internal optical path difference.
  - External optical path difference: the geometric difference in the optical path length from the observed object to the telescope. It depends on the object position and on the baseline vector, and varies in time as the Earth rotates (sidereal motion). It is computed by using the formula: $OPD_{ext} = S\ B$ (where S is the unit vector pointing to the object, and B the baseline vector.
  - Internal optical path difference: the difference in the optical path length, due to the interferometer layout, from the telescope to the detector. It is also called delay offset or delay constant.
- **Operational state**: The state of a system, subsystem, or component that is installed in its intended environment.
- **Package**: See software package.
- **Process**: See software process.
- **Software device driver**: A collection of subroutines and data that constitutes the software interface to an I/O device.
- **Software life cycle**: The period of time that begins when a software product is conceived and ends when the software is no longer available for use. The software life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and check-out phase, operation and maintenance phase, and, sometimes, retirement phase. These phases may overlap or be performed iteratively.

- **(Software) module**: A relatively large subdivision of the implementation items in a software package. In most cases a software package correspond to one single software module, but complex software packages can be split across more software modules. A software module is handled as a single configuration control unit and follows a standardized directory structure. Software modules correspond to *components* in UML terminology.
- **(Software) package**: A major subdivision of a software project that collects a set of correlated functions that are designed, developed and tested all together and independently from other packages. Software packages can be recursively defined as containing other software packages.
- **(Software) process**: A program in execution. It consists of the executable program, the program's data and stack, the support data stored in the database, its program counter, stack pointer and other registers, and all the other information needed to run the program.
- **State**: The value assumed at a given instant by the variable used by the control software to represent the condition of a system, subsystem, or component. Normally it is a finer specification within a given mode.
- **Status**: The set of values of all the parameters (state, numeric read-outs, flags,...) that define the condition of a system, subsystem, or component.
- **Stereotype**: UML term for "a new kind of model element defined within the model based on an existing kind of model element. Stereotypes may extend the semantics but not the structure of pre-existing metamodel classes."
- **Subsystem**: A secondary or subordinate system within a larger system. It usually refers to a device equipped with the control electronics and low level software.
- **System**: A collection of components organized to accomplish a specific function or a set of functions. When no further characterized, it is generally used to refer to the whole of a complex equipment made up by heterogeneous parts.
- **Use case**: "A specific way of using the system by performing some part of the functionality. Each Use Case constitutes a complete course of action initiated by an actor, and it specifies the interaction that takes place between an actor and the system.... The collected use cases specify all the existing ways of using the system" [RD 01].
- **Working wavelength (λ_0)**: is the effective wavelength of the FSU.

1.7 Stylistic Conventions

The following styles are used:

- **bold**
  in the text, for commands, file names, pre/suffixes as they have to be typed.
- *italic*
  in the text, for parts that have to be substituted with the real content before typing.
- `teletype`
  for examples.
- <name>
  in the examples, for parts that have to be substituted with the real content before typing.
  bold and italic are also used to highlight words.

2 Detailed Design

2.1 Overview

The intention of this document is to detail the software design of the PRIMA metrology system (PRIMET), outlined in [AD 07]. A good knowledge of [AD 07] is essential to understand the contents of this document, since background information required to understand the system will not be provided.

PRIMET architecture and detailed design are described using the Unified Modeling Language (UML) graphical notation defined in "The Unified Modeling Language User Guide" [RD 01]. UML diagrams in this document are presented according to the guidelines described in "The Elements of UML Style" [RD 02].

Physically, the PRIMA Metrology system consists of four LCUs for three different subsystems. Two LCUs are identical: The "Phase Meter" LCUs run the same SW on identical HW. While the first measures the differential delta L = FSUB – FSUA, the second unit measures one single channel, -FSU-B. Knowing these two measurements, the remaining channel can easily be deducted.

The HW deployment of the PRIMA MET system is shown in Figure 1: PRIMET HW Deployment.

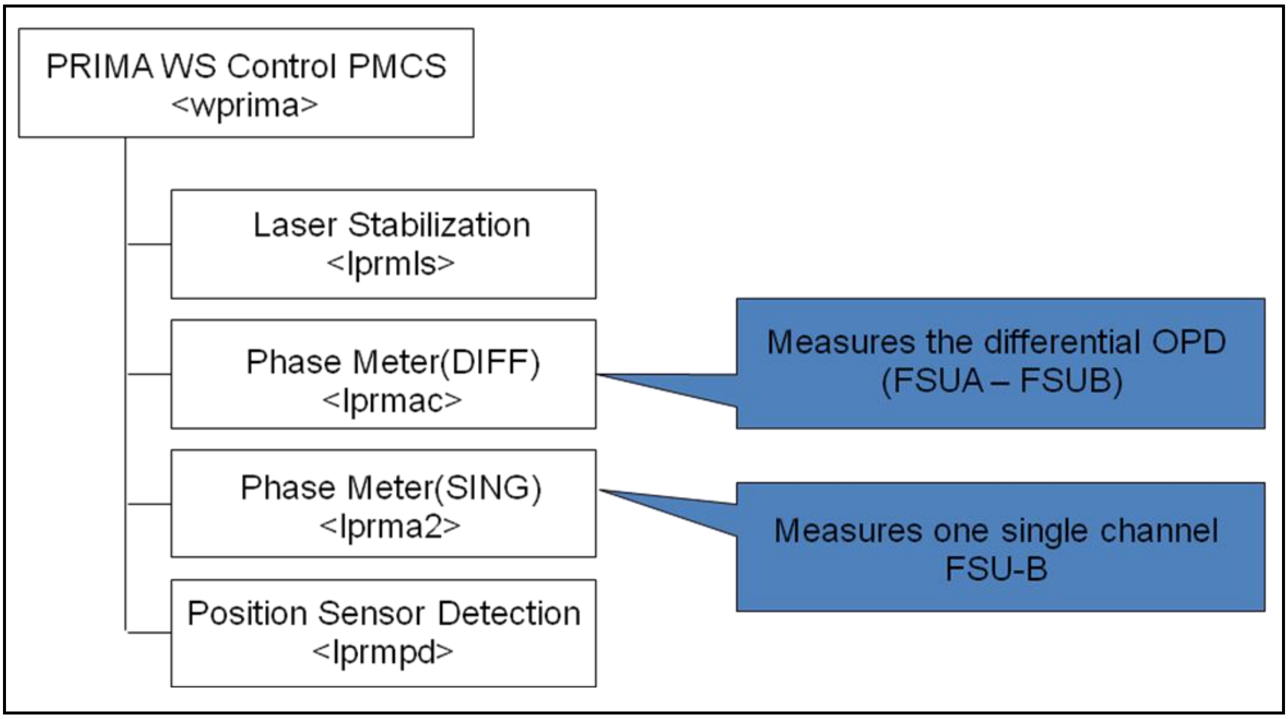

Figure 1: PRIMET HW Deployment

The functionality of the PRIMA metrology system is logically split into four modules, which can be used independently of each other. An overview of the SW packages is given in Figure 2: PRIMET SW Packages.

- PRIMET Laser Stabilization SW: PMLSS
- PRIMET Acquisition Unit (=Phase Meter) SW: PMACQ
- PRIMET Position Sensor Detection SW: PMPSD
- PRIMET WS Control SW: PMCS

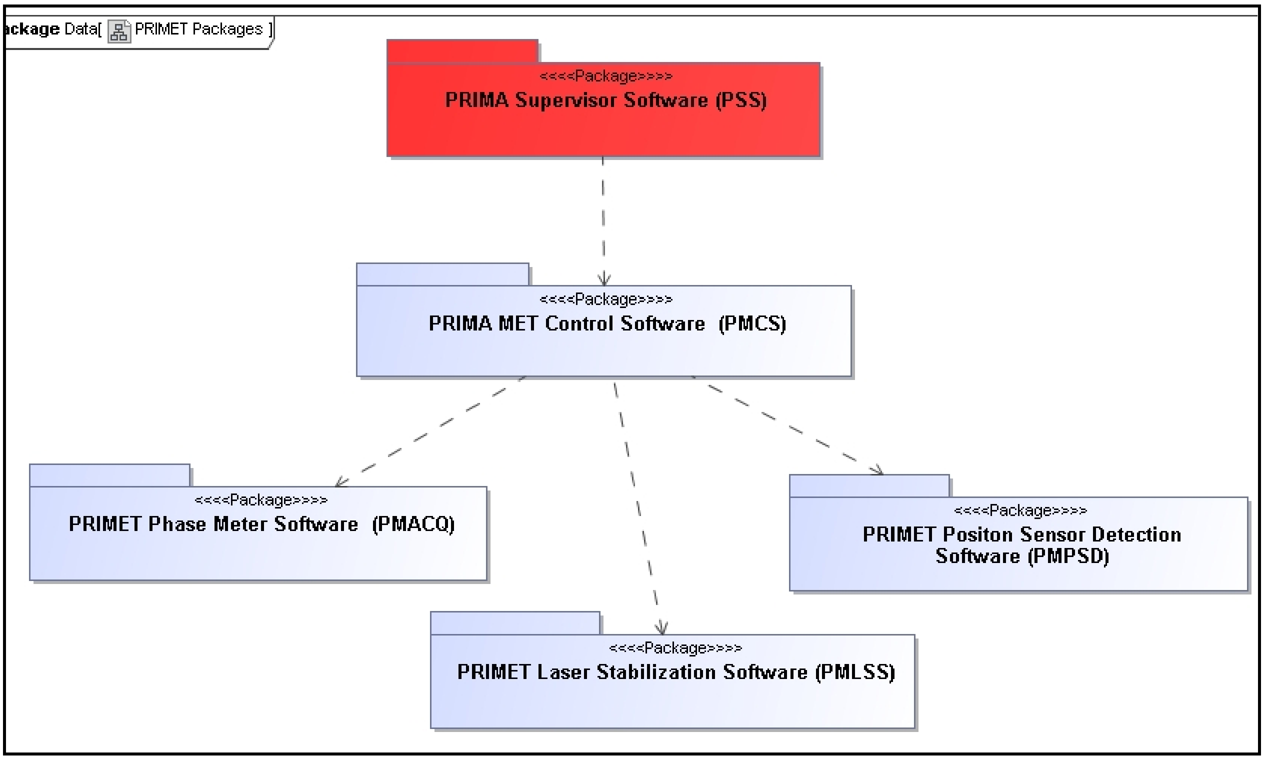

Figure 2: PRIMET SW Packages

The installation of all metrology modules, as well as the creation of the required CCS environments, is done by a dedicated “primaBUILD” module. Two user accounts shall be created on the PRIMA WS: “prima” and “primamgr”. The “primamgr” account shall be used to install the SW:

```
cmmCopy primaBUILD
pkginBuild primaBUILD
```

Please note that this procedure will also install the PRIMA FSU and the IRACE DCS SW.

As soon as the CCS environments have been created successfully, the processes shown in Figure 3: PRIMET SW Processes are running in their corresponding environments.

For the PRIMET WS control SW, the CCS environment “wprima” is utilized, which is however not exclusively used by PRIMET, but shared among all PRIMA sub systems. For each of the PRIMET modules, one engineering GUI is available which can be started in this environment:

- `pmlssGui`: Maintenance panel for the PRIMET “Laser Stabilization” sub system
- `pmacqGui`: Maintenance panel for the PRIMET “Phase Meter” sub system
- `pmpsdGui`: Maintenance panel for the PRIMET “Position Sensor Detection” sub system
- `pmcsGui`: Maintenance panel for the PRIMET WS control SW
- `pmcsILGui`: Maintenance panel for the PRIMET interlock interface

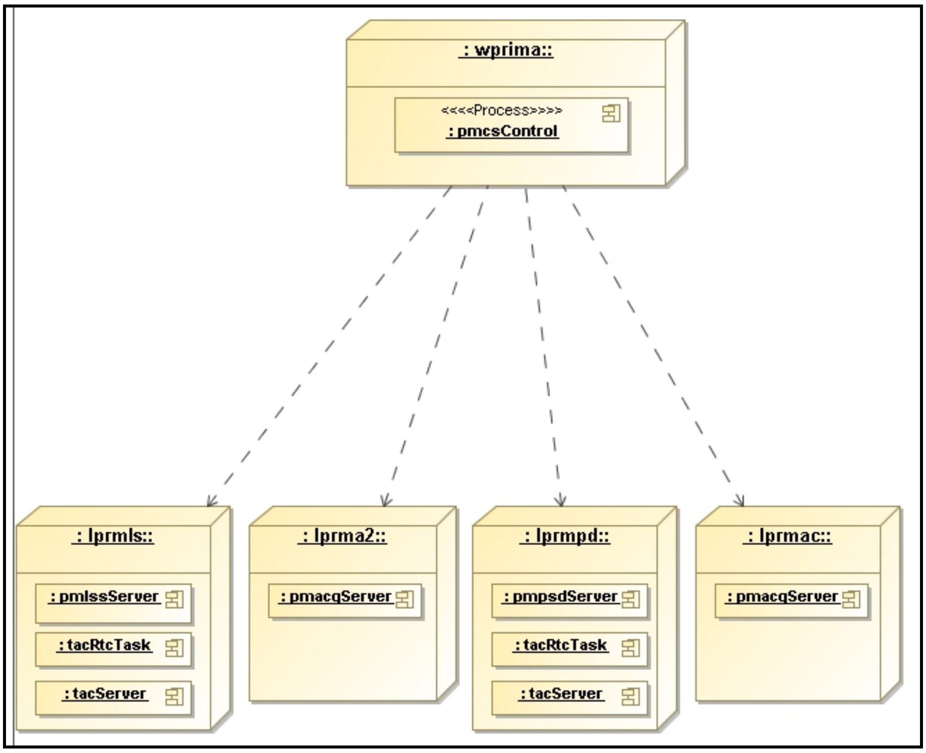

Figure 3: PRIMET SW Processes

PRIMET in the PRIMA Context

PRIMET is part of the PRIMA Control Software (PRICS) that includes the following software components:

- PRIMA Supervisor Software (PSS): responsible for coordinating the other PRICS components. PSS is also the only interface between the Interferometer Supervisor Software (ISS) and PRICS.
- PRIMA Metrology Control Software (PMCS): responsible for controlling the Metrology sub-system.
- PRIMA Fringe sensor unit Control Software (PFCS): responsible for controlling one Fringe Sensor Unit. Two instances of this software are used to control FSU-A and FSU-B
- PRIMA Differential delay line Control Software (DDLCS): responsible for controlling the Differential Delay Lines and the differential OPD Controller (dOPDC). DDLCS implements the same interface of DLCS.
- PRIMA Star separator Control Software (PSCS): responsible for controlling one Star Separator. If the STS is installed on UT, PSCS is considered to be a subsystem of Coude' Control Software. If the STS is installed on AT, PSCS is considered to be a subsystem of AT Control Software (ATCS).

On the PRIMA workstation, PMCS, as well as all other PRIMA subsystems are running, controlled by the PRIMA Supervisor Software PSS. The interface between PMCS and PSS consists of a set of commands and a local database. The required interface is specified in the PSS Interface Control Document [AD 14].

While generally the PRIMA subsystems do not know about each other, since PSS is the only software controlling the interactions between these subsystems, there is one exception to this rule. The translation of the beams in lateral position (pupil optimization) is performed by one STS VCM LCU, whereas PRIMET as a sensor computes the required corrections. For performance reasons, PRIMET directly sends these corrections to the STS VCM LCUs, bypassing PSS. This process and the corresponding interface is described in [AD 23].

2.2 PRIMET Laser Stablization

2.2.1 Overview

The PRIMET "Laser Stabilization" subsystem consists of a laser head together with its frequency stabilization hardware [AD 25]. A model SR844 lock-in amplifier generates the (analog) error signal X, which is read by a VMIVME-3123 analog input board. On a dedicated LCU, a TAC control loop running at 1 kHz calculates two correction signals, which are transmitted to the (analog) frequency tuning inputs of the LightWave model 125 (alternatively InnoLight MIR500NE-FC) microprocessor-based power supply via a MPV-955 analog output board:

- The first tuning signal SLOW FREQUENCY BNC controls the temperature of the laser resonator. This signal allows for a large tuning range, with a slow time constant (normally 1 Hz)
- The second tuning signal FAST FREQUENCY BNC applies a voltage to a piezo mounted on the crystal resonator, which results in a strain that varies the frequency. Tuning this signal can be done up to a rate of 100 kHz.

The functionality of the "Laser Stabilization" package of PMCS is contained in the separate CMM module `pmlss`. PMLSS can be used as a stand-alone application, it does not depend on other PRIMA metrology modules. A maintenance panel is provided to send commands to and display the status of PMLSS.

The following LCU architecture has been selected for the PMLSS application:

- 1 Motorola MVME-6100 PPC CPU board
- 1 VMIVME-3123 analog input board
- 1 ESO Time Interface Module (TIM) board.
- 1 MPV-955 analog output board
- 1 ISER-12 serial interface board
- 1 MEN carrier board, equipped with 2xM36 (analog in) and 1xM58 (digital out) modules for diagnostic/reset of the Innolight laser
- 1 MEN carrier board, equipped with 1xM28 (digital out) and 1xM31 (digital in) to control two flip mirror devices

A schematic of the HW layout of the PMLSS LCU and all cabling is given in Figure 4: PMLSS LCU. Several external devices are required, which are mainly connected to the LCU via serial cables.

Originally, the LightWave model 125-1319-200 was foreseen to be used as the PRIMET light source. Due to problems with the manufacturer, it was replaced by a InnoLight model MIR 500NE-FC later. The SW is configurable to use either of them, since the LightWave laser has already been procured and shall be used as a PRIMET spare part.

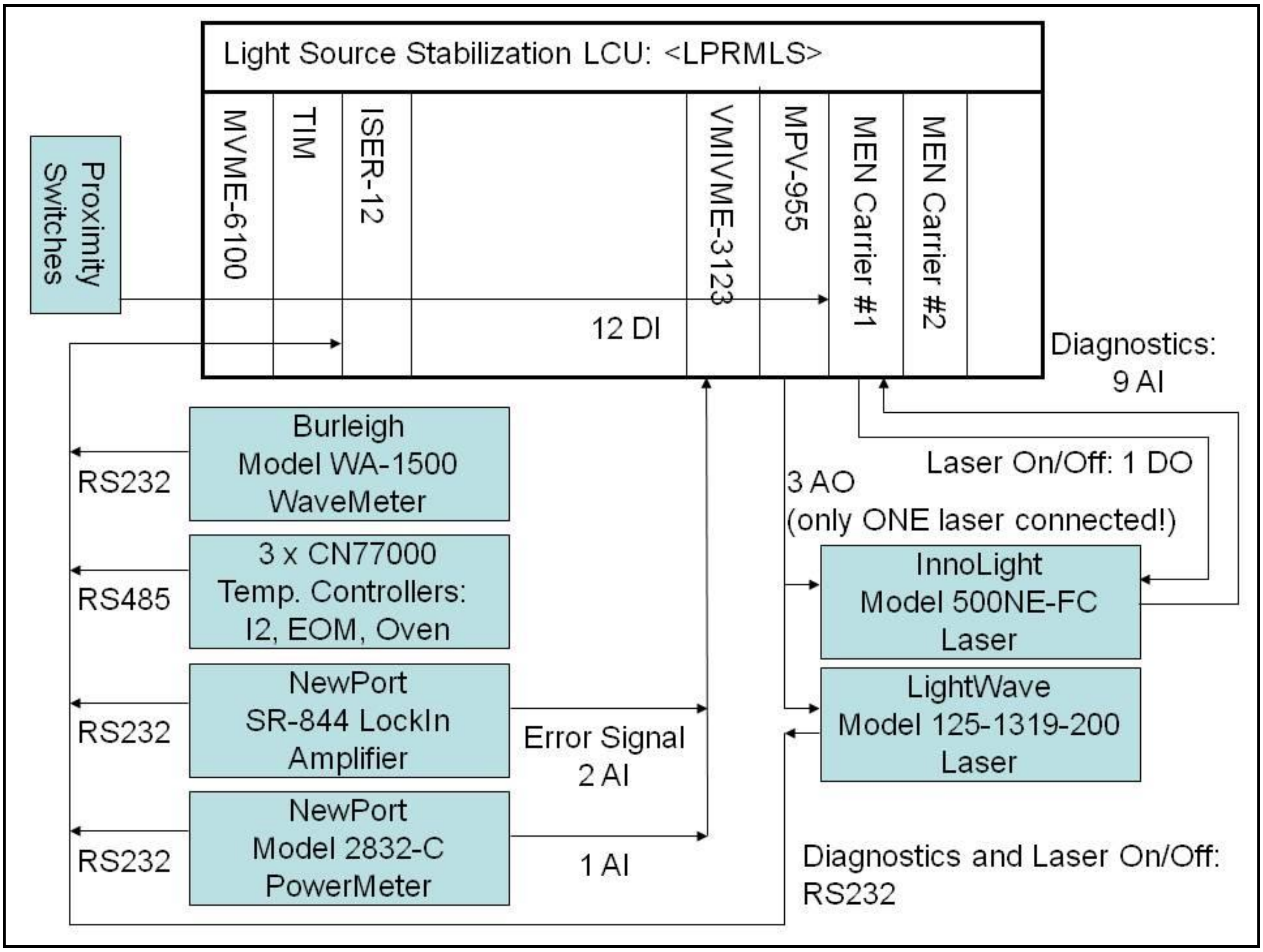

Figure 4: PMLSS LCU

The “Light Source Control” package is deployed on the dedicated PMLSS LCU. It consists of a LSF application [AD 18] that provides the database and command handling, and implements a ESO standard state machine. The control loop to stabilize the light source shall be implemented by TAC [AD 12], which shall use only standard blocks from the TAC library. For these TAC blocks, no device drivers are required.

Since the ISER-12 and the AVME-9481 boards are simply used for initialization and monitoring, they shall be controlled by the LSF application, not by TAC. For these two devices, the corresponding device drivers must be loaded in the bootscript of the LCU.

Of the ISER-12, five serial connections shall be used, which are created as separate LSF serial devices:

- RS232 connection to the microprocessor based power supply of the LightWave model 125 laser.
- RS485 connection to the NewPort CN77000 temperature controllers for the iodine cell, the EOM driver, and the OVEN.
- RS232 connection to the Burleigh WA-1500 WaveMeter.
- RS232 connection to the NewPort 2832-C PowerMeter.
- RS232 connection to the NewPort SR-844 lock-in amplifier.

A interlock and interface circuitry shall switch off the laser when activated. Once a interlock occured, a manual or software reset is necessary before the laser can be switched on again. The status of this circuit shall be monitored by PMLSS only indirectly via the “interlock status” flag within the laser status.

The state diagram of PMLSS is shown in Figure 5: PMLSS State Chart. As soon as PMLSS is commanded to ONLINE, the TAC control loop is running, however in open loop. A dedicated command CLSLOOP is used to automatically close the loop of the laser stabilization. The steps required to close the loop are:

1. Check with the WA-1500 Wavemeter if 1319.1774 nm < lambda(nm) < 1319.1786 nm. If OK move to step (4), otherwise calculate:
   for Lightwave laser:
   AOOffset = (1319.1786 – lambda(nm)) * 100
   for Innolight laser:
   AOOffset = -1.0 * (1319.1786 – lambda(nm)) * 100
2. Wait 15 sec
3. if X (Lock-In) < 0 move to step (4), otherwise send the "autophase" command to the SR-844 lock-in amplifier
4. After 1 sec store the first averaged value from the Powermeter as TestTACConst3
5. Set the parameter of the "TestTacSwitch" block to 2
6. Wait 1 sec
7. Set the parameter of the "SwitchTemp" block to 2
8. Wait 5 sec
9. Set the parameter of the "SwitchPiezo" block to 2

The frequency stabilization loop is considered as successfully closed if the following conditions are met [AD 27]:

- Laser OK (see below).
- Temperature range of the CN77000 temperature controllers NOT exceeded.
- The wavemeter indicates a wavelength of 1319.1771 nm ± 0.0001 nm (query using RS-232).
- The reference signal is detected by the Lock-in (query using RS-232).
- The mean error signal (X output of the lock-in) computed over 1 sec is < 0.05 V (see), corresponding to a laser frequency shift of 2MHz (read analog input).
- The standard deviation of the error signal (X output of the lock-in) computed over 1 sec is < 0.05 V , corresponding to a laser frequency shift of 2MHz (read analog input).

The "Laser Stable" attribute in the OLDB visualizes if the above conditions are fulfilled. During ONLINE, PMLSS constantly monitors the status of the laser. For each of the two laser models, different conditions must be met for the laser head to be OK or NOT OK:

Conditions necessary to generate a "laser OK" for the Lightwave laser model 125-1319-200 [AD 27]:

- Laser Diode On
- Laser Power 200 mW ± 10% (TBC)
- Interlock status: no interlock
- Fault status: No Fault

Conditions necessary to generate a "laser OK" for the Innolight laser model MIR 500NE-FC [AD 27]:

- Diode laser 1 power monitor 1.2 V± 10% (150mW± 10%) TBC
- Diode laser 2 power monitor 1.2 V± 10% (150mW± 10%) TBC
- Diode laser 1, Temperature guard (temperature control loop OK)
- Diode laser 2, Temperature guard (temperature control loop OK)
- Noise eater monitor ON
- Interlock status: no interlock

Additionally, PMLSS monitors the status of the proximity switches of the phasemeter and the acusto-optic modulator (AOM). Three instruments can be configured for PRIMA: FSU-A, AMBER and MIDI. The fibres of the instrument to be used for the observation must be connected manually by an operator. However, the SW can detect via the proximity switches which instrument is currently connected.

For maintenance purposes, two commands STRTENG/STOPENG are available which allow the user to create engineering files. The format of these files is defined in [AD 27].

Diagnostic information from the frequency stabilization HW shall be read at low frequency:

- CN77000 temperature measurements of the I2 cell, the oven, and the EOM.
- NewPort PowerMeter Channel B reading.
- Wavelength as measured by the Burleigh WA-1500 WaveMeter.
- Newport SR-844 Lock-in Amplifier sensitivity, time constant, filter slope, and detection frequency.
- In case of the LightWave model 125-1319-200 laser: fault status, interlock status, diode power, laser temperature.
- In case of the InnoLight model MIR 500NE-FC laser: D1 and D2 power, D1 and D2 TEC error, crystal TEC error.

At a fixed frequency of 1/30 Hz, PMLSS logs statistical information to the CCS logging system. Please refer to [AD 27] for the keywords generated periodically by PMLSS. Additionally, most commands and their results, as well as “exceptional” events (Laser not OK, temperature range exceeded) are logged, and can therefore be retrieved at the end of the night via the OPS log database.

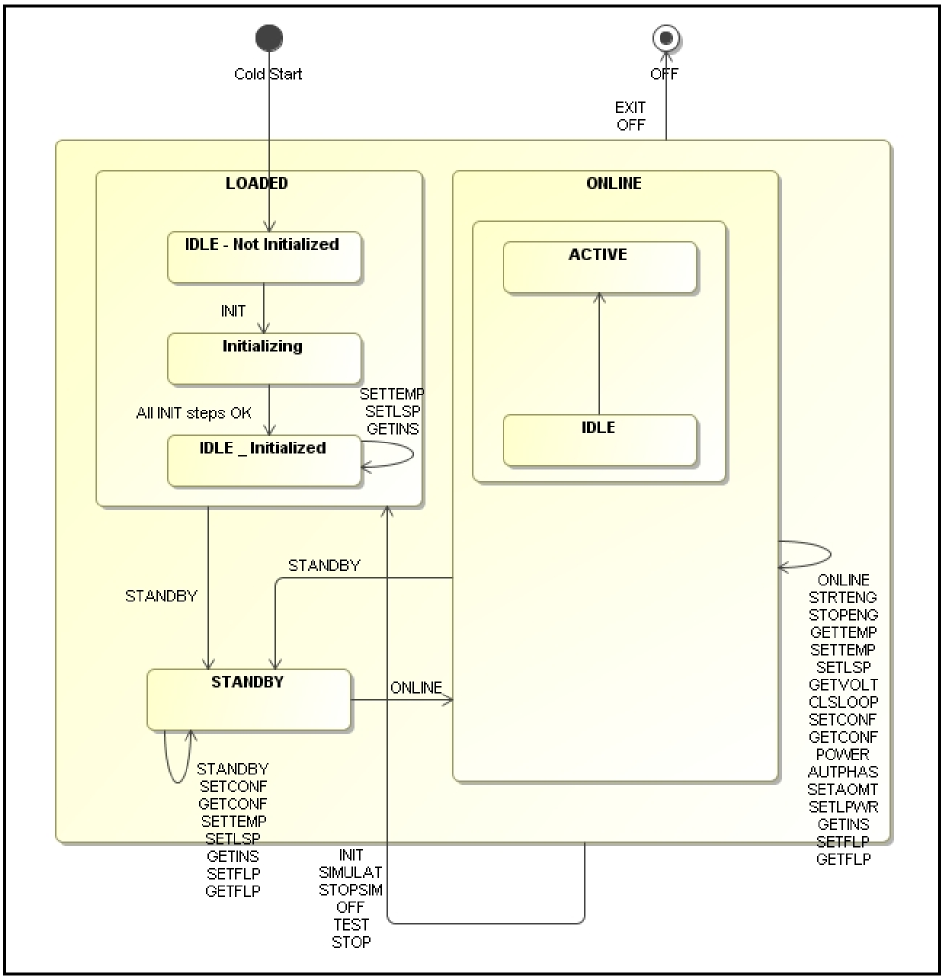

Figure 5: PMLSS State Chart

2.2.2 Command Description

Command	SETLSP
Parameters	<LSP> (REAL)
Reply	OK/ERROR
Description	The SETLSP command is used to modify laser stabilization parameter that qualifies the laser to be "stabilized". The mean value of the error signal measured by the lock-in amplifier must be below this value.

Command	SETTEMP
Parameters	<Device Name> (STRING : « I2 », « EOM » or « OVEN ») <Temperature> (REAL)
Reply	OK/ERROR
Description	The SETTEMP command is used to modify the destination temperature of the CN77000 temperature controller used for either the I2, EOM or OVEN.

Command	GETTEMP
Parameters	<Device Name> (STRING : « I2 », « EOM » or « OVEN »)
Reply	<Temperature> (REAL)
Description	The SETTEMP command is used to read the destination temperature of the CN77000 temperature controller used for either the I2, EOM or OVEN.

Command	GETVOLT
Parameters	None
Reply	<Voltage Piezeo> (REAL) <Voltage Temp.> (REAL)
Description	The GETVOLT command is used to read the voltages applied to the FAST FREQUENCY BNC and SLOW FREQUENCY BNC inputs of the laser’s power supply unit done by TAC during ONLINE.

Command	CLSLOOP
Parameters	<On/Off Flag> (STRING)
Reply	OK/ERROR
Description	The CLSLOOP command is used to open or close the light source stabilization control loop.

Command	SETCONF
Parameters	<Temperature I2> (REAL) <Temperature DC> (REAL) <Temperature OVEN> (REAL)
Reply	OK/ERROR
Description	The SETCONF command sets the temperature setpoints of the CN77000 temperature controllers user for I2, DC and OVEN.

Command	GETCONF
Parameters	None
Reply	<Temperature I2> (REAL) <Temperature DC> (REAL) <Temperature OVEN> (REAL)
Description	The GETCONF command reads the temperature setpoints of the CN77000 temperature controllers user for I2, DC and OVEN.

Command	POWER
Parameters	<On/Off Flag> (STRING)
Reply	OK/ERROR
Description	The POWER command is used to switch the laser diode ON or OFF. Please note that the laser diode can only be turned on if the key on the front plate of the LightWave model 125 laser/InnoLight model MIR500 has been turned manually in advance.

Command	RESILCK
Parameters	None
Reply	OK/ERROR
Description	The RESILCK command is used to reset the interlock circuitry with a digital signal.

Command	AUTPHAS
Parameters	None
Reply	OK/ERROR
Description	The AUTPHAS command is used to perform a manual autophase function of the SR-844 Lock-In amplifier. Normally this is done automatically when going ONLINE with pmlss.

Command	STRTENG
Parameters	<Number of Samples> (INTEGER)
Reply	OK/ERROR
Description	The STRTENG command is used to start recording data coming from the stabilization loop into a engineering file.

Command	STOPENG
Parameters	None
Reply	OK/ERROR
Description	The STRTENG command is used to stop recording data coming from the stabilization loop into a engineering file.

Command	SETLPWR
Parameters	<Laser Power> (REAL)
Reply	OK/ERROR
Description	The SETLPWR command is used to set the laser power of the Innolight/Lighwave laser in [mW].

Command	SETAOMT
Parameters	<AOM ID> (STRING) <Transmission> (REAL)
Reply	OK/ERROR
Description	The SETAOMT command is used to set the transmission coefficient of one of the four AOMs (in percent).

Command	GETINS
Parameters	None
Reply	<Instrument Name> (STRING)
Description	Returns the name of the instrument configured via the metrology switchyard. The return value is among the values FSUA, AMBER, MIDI, or UNDEF.

Command	GETFLP
Parameters	<Flip ID> (STRING)

Reply	<Flip OD Position> (STRING)
Description	Returns the position of the FLIP MIRROR device "PRIMETA" or "PRIMETB".

Command	SETFLP
Parameters	<Flip ID> (STRING) <In/Out Flag> (STRING>
Reply	OK/ERROR
Description	Sets the position of the FLIP MIRROR device "PRIMETA" or "PRIMETB" to either "IN" or "OUT".

2.2.3 Task Description

Task Name	**pmlssMonitor**
Description	pmlssMonitor serves as a background monitor running at low frequency (1 Hz), which is used to update the status of the OLDB with: • The current temperature measurements of the CN77000 controllers for the iodine cell, the EOM driver, and the OVEN. • The laser temperature setting, the status of the laser diode (ON/OFF), the interlock status and the fault status of the LightWave model 125 laser (if configured) OR analog diagnostic information from the InnoLight model MIR 500 laser. • The measurements of the SR844 Lock-In Amplifier, i.e. the error signals X and Y, the phase, and the reference phase. • The status of the interlock circuitry, i.e. the status of the emergency stop button, the interlock origin, the latch status of the manual reset button, and the status of the 10 lamps mounted next to the entrance doors.

2.2.4 RS232 connection to the power supply of the LightWave model 125 laser

The LightWave model 125/126 microprocessor-based power supply provides a RS232 interface to monitor and control the laser by software [AD 19]. However, most of the parameters can only be read, but not be changed through this interface.

This serial interface uses only the signals RxD, TxD and GND, and must be operated with the following parameters:

- Baud Rate: 9600 baud
- No Parity
- 8 Data Bits, 1 Stop Bit
- Full Duplex

The setting for the baud rate must be configured manually through a DIP switch on the back of the power supply. That switch must be set to 9600 baud in advance to assure that the device will interface correctly with PMLSS.

Only the following functionalities of the LightWave model 125/126 microprocessor-based power supply are used by PMLSS, and implemented in the specified procedure name within pmlssSerialInterface.c:

Procedure	**pmlssLW125GetParam**
Parameters	<vltINT32: Parameter ID>
Returns	<vltBYTES16: Parameter Value>

Description	Reads the specific parameter given by <Parameter ID> from the LightWave model 125/126 microprocessor-based power supply. Supported <Parameter ID>'s are: ▪ ASCII "0": Model Number and SW Date ▪ ASCII "2": Chassis Serial Number ▪ ASCII "6": Diode ON/OFF ▪ ASCII "7": LASER Power ▪ ASCII "8": Interlock Status ▪ ASCII "9": Fault Status ▪ ASCII "A": Diode Current Adjust ▪ ASCII "C": Diode Power Setting ▪ ASCII "D": Laser Temperature Setting As a result, the value of the parameter is supplied in ASCII format. The format of the reply message is dependent on the parameter ID, the interpretation of the string is left to the calling function: ▪ ASCII "0": Format "125 dd MMM yy" ▪ ASCII "2": Format "nnnnn" ▪ ASCII "6": Format "DIODE ON" or "DIODE OFF" ▪ ASCII "7": Format "Pwr nn mW" ▪ ASCII "8": Format "ILOCK OPEN" or "ILOCK CLOSED" ▪ ASCII "9": ASCII String, with the following meaning: - "No Fault": No Error Condition - "ILCKOPEN": Interlock condition detected - "LASER ON": Laser diode is switched ON - "CABLE ?": Check cable to laser - "HD STDBY": "LASE" interlock line set to low - "HD FAULT": Laser head or diode laser too hot ▪ ASCII "A": Format "ADJ +/-nn" ▪ ASCII "C": Format "nn.nnnn" ▪ ASCII "D": Format "nn.nnnn" For completeness, this procedure supports the reading of all available parameters from the LightWave model 125/126 power supply. A background monitor within PMLSS shall query periodically the parameters "Laser Temperature Setting", "Diode Power Setting", "Interlock Status", and "Fault Status" to reflect the current status of the input signals connected to the BNC inputs of the power supply, and update the error status of the device.
Basic Course	1. PMLSS shall check if the desired parameter ID is among the above ASCII values. 2. PMLSS shall send the following bytes: ▪ Byte 0: ESC ▪ Byte 1: ASCII "0" (Note: Laser Number, always 0) ▪ Byte 2: ASCII "4" (Note: Command "Read Parameter") ▪ Byte 3: <Parameter ID> ▪ Byte 4: LF ▪ Byte 5: CR

Procedure	**pmlssLW125SwitchDiode**
Parameters	<vltLOGICAL: diodeOn>
Returns	SUCCESS/FAILURE
Description	The Laser Diode can be turned ON or OFF via the RS232 interface of the LightWave model 125/126 microprocessor-based power supply. As a precondition, the laser must already be at

	least in STANDBY mode, which is achieved manually by turning a key on the front plate of the power supply.
Basic Course	PMLSS shall send the following bytes: ▪ Byte 0: ESC ▪ Byte 1: ASCII “0” (Note: Laser Number, always 0) ▪ Byte 2: - if diodeOn == ccsTRUE: ASCII “2” (Note: Command “Turn Parameter ON”) - if diodeOn == ccsFALSE: ASCII “3” (Note: Command “Turn Parameter OFF”) ▪ Byte 3: ASCII “6” (Note: Parameter ID “Diode ON/OFF”) ▪ Byte 4: LF ▪ Byte 5: CR

2.2.5 RS485 connection to the NewPort CN77000 Temperature Controllers

Three identical NewPort CN77000 temperature controllers are used: to control the temperatures of the iodine cell, the EOM driver, and the OVEN. This serial interface for these devices uses only the signals RxD, TxD and GND, and must be operated with the following parameters:

- Baud Rate: 9600 baud
- No Parity
- 8 Data Bits
- 1 Stop Bit
- Full Duplex

Note: The parameters for the serial interface must be configured manually via the service menu, accessible via the front panel of the CN77000. In detail, the following parameters must be set by the user on the device beforehand in order to assure that the temperature controller correctly interfaces with PMLSS:

Parameter	Value	Description
BAUD	9600	Baudrate
PRTY	No	Parity
DATA	8	Number of Data Bits
STOP	1	Number of Stop Bits
STND	C4	Selects between RS232 and RS485
ADDR	01-03	Set the device ID to 01-03

Table 3: CN77000 Manual Interface Settings

The initialization of the serial interface on the LCU is done by LSF. For the procedures listed below, PMLSS shall evaluate the reply message of the temperature controller, which in case of an error consists of the following values:

Reply Message	Description
?43	CN77000 received invalid command code.
?45	CN77000 could not write data into ist EEPROM.
?46	The number of bytes for a command is not correct.
?48	Checksum error (not used)
?56	CN77000 received a invalid device ID with the command.

Table 4: CN77000 Error Messages

Only the following functionalities of the CN77000 temperature controllers shall be used by PMLSS, and implemented within the specified procedure name.

Procedure	**pmlssCNConfigure**
Parameters	<vltINT32: Temperature Controller ID>
Returns	SUCCESS/FAILURE
Description	This procedure sets various configuration parameters for the serial interface and the format of the status messages to the default values which PMLSS expects. The commands to the CN77000 shall start with a "*", followed by the device ID set manually through the front panel beforehand. The command "W" instructs the temperature controller to "write" a parameter into its local EEPROM. These settings are not lost after a controller reset.
Basic Course	PMLSS sends the following commands to the CN77000: * 01 W 1F 38 * 01 W 20 02 Note 1: The first command sets the parameter "bus format" to the following values: ▪ Do not add a checksum byte to command and status messages. ▪ Do not add a "LF" character after each status message, send a single "CR" instead. ▪ Do not send back a echo of the command in the status message. ▪ Set the bus interface to RS485. ▪ Do not send continuous status messages, but send status messages only upon command. ▪ Separate data fields with a "SPC" character. Note 2: The second command sets the parameter "data format" to the following values: ▪ Do not send the alarm status in the status message. ▪ Send the measured value in the status message. ▪ Send the maximum measurement value in the status message. ▪ Send the minimum measurement value in the status message. ▪ Do not include units in the status message.

Procedure	**pmlssCNReadTemp**
Parameters	<vltINT32: Temperature Controller ID>
Returns	<vltDOUBLE: Current Temperature Measurement> [Celsius]
Description	This procedure shall read the current temperature measurement of the specified CN77000 unit, in the format selected via the procedure pmlssCNConfigure. Therefore, the precision of the measurement is one digit after the decimal point. Commands to the CN77000 shall start with a "*", followed by the device ID that was set manually through the front panel beforehand. The command "V" instructs the temperature controller to deliver all data values that were requested through the setup with command "data format". In the case of PMLSS, this is only the current temperature measurement, unitless.
Basic Course	1. PMLSS sends the following commands to the CN77000: * 01 V <Controller ID> 2. PMLSS scans the reply message of the temperature controller, and returns a vltDOUBLE with the current measurement.

Procedure	**pmlssCNSetTemp**
Parameters	<vltINT32: Temperature Controller ID> <vltDOUBLE: Temperature> [Celsius]
Returns	SUCCESS/FAILURE
Description	This procedure sets the temperature setpoint for the given temperature controller. It is assumed that the parameter for the temperature setpoint is given with a precision of one digit after the decimal point. Parameters supplied to this procedure with a higher resolution are rounded to one digit after the decimal point. Commands to the CN77000 shall start with a "*", followed by the device ID that was set manually through the front panel beforehand. The command "P" instructs the temperature controller to "put" a parameter into its local RAM. These settings are lost after a controller reset.
Basic Course	PMLSS sends the following commands to the CN77000: * 01 P 01 <Temperature Setpoint> Note: The data values for the temperature setpoint consists of three bytes that must comply with the following format: ▪ Bit 23 is the sign bit: "1" for negative values, "0" for positive values. ▪ Bits 22-20 must always be set to "010", which selects a precision of one digit after the decimal point. ▪ Bits 19-0 contain the actual temperature setpoint.

2.2.6 RS232 connection to the Stanford Research SR-844 Lock-In Amplifier

The interface uses the signals RxD, TxD, DTR, CTS and GND, and must be operated with the following parameters:

- Baud Rate: 9600 baud
- No Parity
- 8 Data Bits
- 1 Stop Bit
- Full Duplex

All above communication parameters must be set through the front panel of the SR-844 lock-in amplifier before trying to operate this unit together with PMLSS. Please note that the SR-844 supports both a RS232 and a GBIP interface. Via the front panel, the RS232 interface must be selected. Additional settings are required, like the time constant, the low pass filter, the sensitivity setting, the input signal impedance and the configuration of the two output channels. It is not foreseen to set these values through PMLSS, they shall be set manually by the user.

However, as soon as they have been configured, it is possible to store the current setup in the EEPROM of the SR-844. For this purpose, the lock-in amplifier provides two useful commands ("SSET" and "RSET") to either save the actual settings of all parameters into one of ten available buffers of a local EEPROM, or to restore these settings from one of these buffers.

The initialization of the serial interface is done by LSF, however the above parameters must be provided in the DB configuration file of PMLSS. Only the following functionalities of by the SR-844 lock-in amplifier are used by PMLSS, and implemented within the specified procedure name:

Procedure	**pmlssSR844Init**
Parameters	None
Returns	SUCCESS/FAILURE
Description	This procedure sets various configuration parameters for the serial interface to those default values that PMLSS expects.
Basic Course	PMLSS sends the following commands to the SR844: "*CLS" "*IDN?" Note 1: The first command clears the error buffer of the SR-844 device. Note 2: The second command queries the identification of the SR844, used to test if the interface to the device is working. The reply message consists of a string of the following format: "Standford_Research_Systems,SR844,s/n00111,ver1.0".

Procedure	**pmlssSR844AutoPhase**
Parameters	None
Returns	SUCCESS/FAILURE
Description	This procedure performs the auto phase function of the SR844, which adjusts the reference phase such that the current measurement has a Y value of zero, and a X value equal to the signal magnitude (R).
Basic Course	PMLSS shall send the following command to the SR844: "APHS\n". The "CR" character indicates the end of the command.

Procedure	**pmlssSR844Update**
Parameters	None
Returns	SUCCESS/FAILURE
Description	This procedure shall query the current settings of the SR844 device for: Second Harmonic, Detection Frequency, Sensitivity, Time Constant, Filter Constant. The "SNAP?" command of the Lock-In Amplifier shall be used to query multiple values measured at at most two different timestamps. This is due to the fact that X and Y are always recorded together, as are R and the phase, but internally both are recorded at different frequencies. Therefore, between the recording of the two pairs of values, a delay of 84 µs might be present.
Basic Course	1. PMLSS shall sends the following command to the SR844: "SNAP? 4,5\n" to query the "Second Harmonic" and "Detection Frequency" settings of the SR-844 device. 2. PMLSS shall sends the following command to the SR844: "SENS?\n" to query the "Sensitivity" setting of the SR-844 device. 3. PMLSS shall sends the following command to the SR844: "OFLT?\n" to query the "Time Constant" setting of the SR-844 device. 4. PMLSS shall sends the following command to the SR844: "OFSL?\n" to query the "Filter Constant" setting of the SR-844 device.

2.2.7 RS232 connection to the NewPort 2832-C PowerMeter

This serial interface uses only the signals RxD, TxD and GND, and must be operated with the following parameters:

- Baud Rate: 9.600 baud
- No Parity
- 8 Data Bits
- 1 Stop Bit
- No Flow Control

The initialization of the serial interface on the LCU is done by LSF, however all above parameters must be provided in the DB configuration file of PMLSS. The following functionalities of the NewPort 2832-C PowerMeter shall be used by PMLSS, and implemented within the specified procedure name:

Procedure	**pmlssPMInit**
Parameters	None
Returns	SUCCESS/FAILURE
Description	This procedure shall initialize the communication with the NewPort 2832-C device.
Basic Course	1. PMLSS shall send the command "*CLS" to clear the error buffer of the NewPort 2832-C device. 2. PMLSS shall send the command "*IDN" to query the device ID of the NewPort 2832-C device.

Procedure	**pmlssPMUpdate**
Parameters	<Channel ID>
Returns	None
Description	This procedure shall read the current measurement of the selected channel ("A" or "B") of the NewPort 2832-C device.
Basic Course	1. PMLSS shall send the command "R_A?\r\n" (in case of channel "A") or "R_B?\r\n" (in case of channel "B") to the NewPort 2832-C device, and return the delivered value.

2.2.8 TAC Light Source Stabilization Control Loop

The control loop to implement the actual light source stabilization shall be implemented with TAC. The structure of the TAC algorithm is shown in Figure 6: PMLSS TAC Control Loop.

The error signal "X" (actually a ±10 V analog voltage proportional to "X") delivered by the SR-844 lock-in amplifier shall be read by the analog I/O board VMIVME-3123 at a fixed sampling frequency (1 kHz). The availability of a new error signal triggers the TAC algorithm, which in turn uses two standard PID2 control blocks from the TAC library to calculate the corrections signals.

The output of the first PID2 controller drives the SLOW FREQUENCY BNC input of the LightWave model 125/126 power supply (alternatively: InnoLight model MIR 500), while the output of the second PID2 controller is used as a direct input to the FAST FREQUENCY BNC connector. Since both inputs on the LightWave model 125/126 power supply (alternatively: InnoLight model MIR 500) require analog signals, a MPV-955 TAC block is used to convert the digital outputs of the PID controller to ±10 V analog voltages.

Finally, one TAC monitor block is used to update the database with the input signal coming from the VMIVME-3123 block, and the two output signals generated by the MPV-955 block. Since the database values are only used for monitoring but not required by the control loop, the monitor block runs at low frequency (1 Hz). A "scope" block, running at the RT algorithm's speed (1 kHz), is however used for real-time data display via RTDScope. The following information can be displayed in realtime by connecting to the "Scope" block of the TAC RT algorithm:

- Error signal "X", delivered by the SR844 Lock-in Amplifier.
- Piezo command, calculated by TAC.
- Temperature command, calculated by TAC.
- PowerMeter analog-out channel #B.
- Theta signal, delivered by the SR844 Lock-in Amplifier.

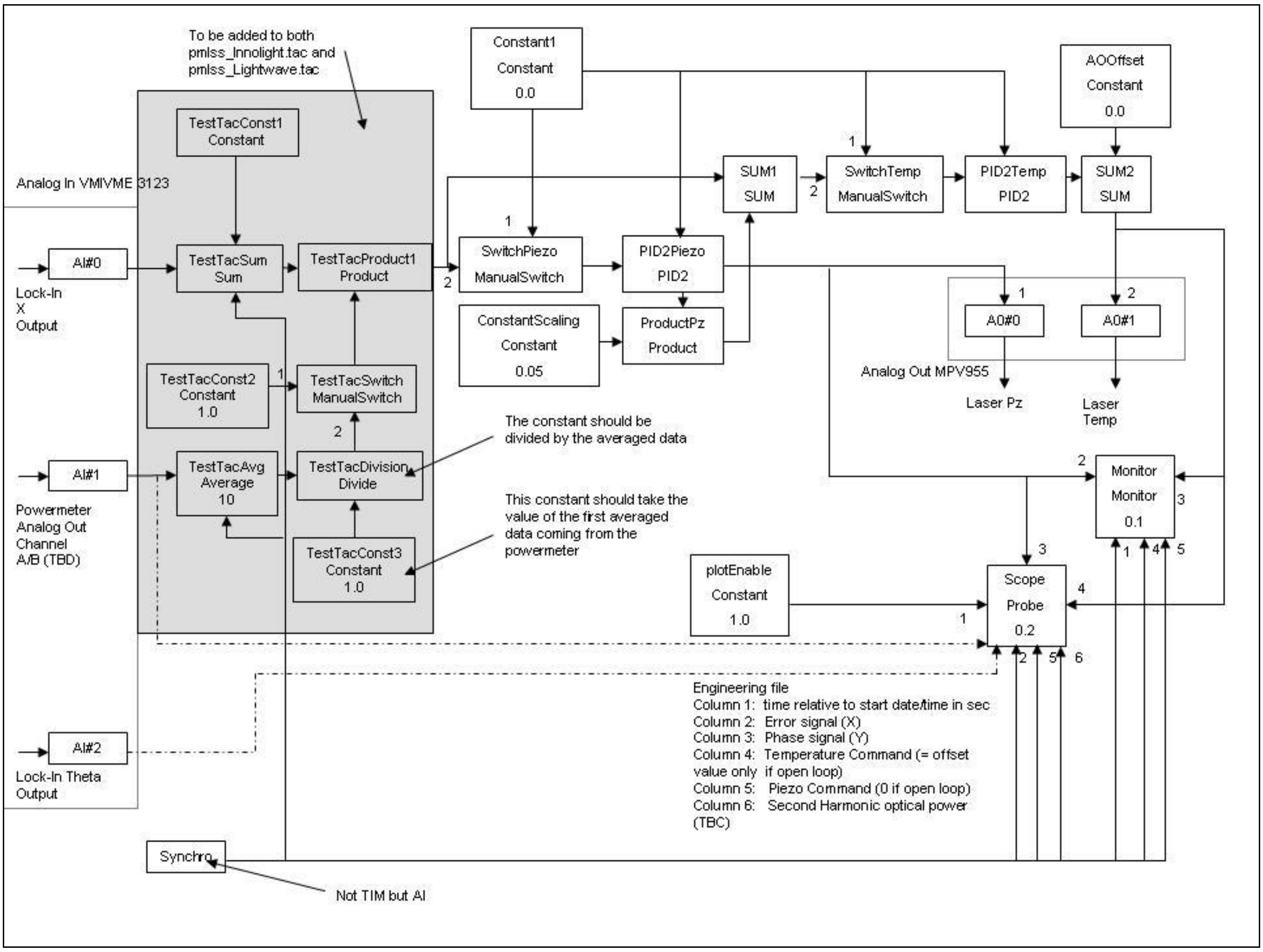

Figure 6: PMLSS TAC Control Loop

2.2.9 Database Description

The structure of the PMLSS database is outlined in Figure 7: PMLSS OLDB Structure.

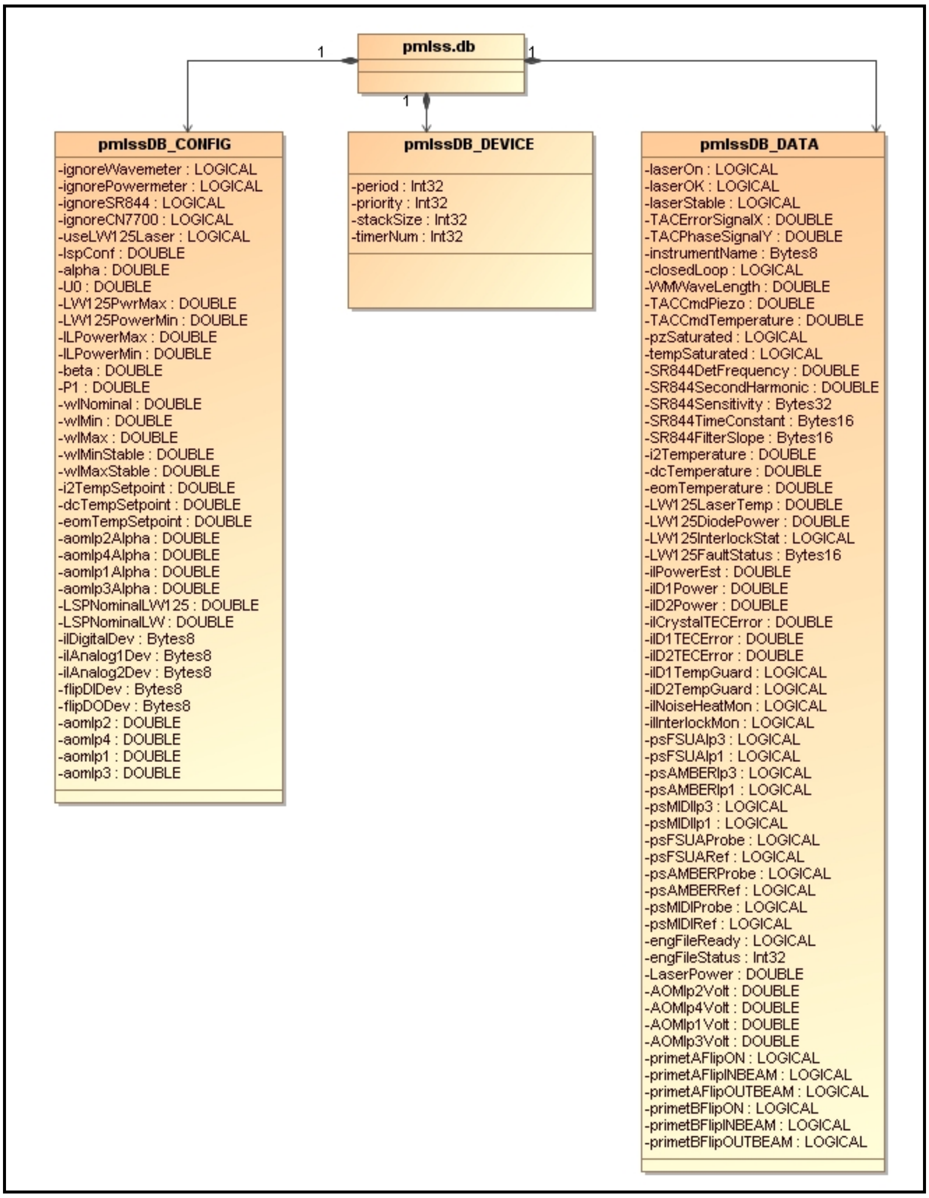

Figure 7: PMLSS OLDB Structure

PMLSS configuration data is stored relative to the database point “<alias>pmlss:config”:

Attribute	Type	Description
ignoreWavemeter ignorPowerMeter ignoreSR844 ignoreCN77000	LOGICAL	In case one or more of the external (serial) devices required for the laser stabilization are missing, the user has to possibility to explicitly “ignore” them. In this case, it is not possible to close the stabilization loop!
useLW125Laser	LOGICAL	Indicates if the LightWave Model 125 laser shall be used. If FALSE, the InnoLight Model MIR 500 laser shall be selected.
lspConf alpha U0 beta P1	vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE	Laser stabilization parameters.
lw125PwrMin	vltDOUBLE	Min./Max. values for the SETLPWR command [mW].

lw125PwrMax ilPwrMin ilPwrMax		
wlNominal wlMin wlMax wlMinStable wlMaxStable	vltDOUBLE	Wavelength measurements of the WaveMeter for “nominal” and “stable” (i.e. during closed loop operation) conditions.
i2TempSetpoint dcTempSetpoint eomTempSetpoint	vltDOUBLE vltDOUBLE vltDOUBLE	Temperature setpoint for the NewPort CN77000 temperature controller of the iodine cell. Temperature setpoint for the NewPort CN77000 temperature controller of the OVEN. Temperature setpoint for the NewPort CN77000 temperature controller of the EOM.
aomIp2Alpha aomIp4Alpha aomIp1Alpha aomIp3Alpha aomIp2 aomIp4 aomIp1 aomIp3	vltDOUBLE vltDOUBLE	Parameters for the AOMs. Initial settings of the transmission coefficients of the four AOMs (in percent).
lspNominalLW lspNominalIL	vltDOUBLE	Configuration of the initial (nominal) laser power setting.
ilDigitalDev ilAnalog1Dev ilAnalog2Dev flipDIDev flipDODev	vltBYTES8	Device names of the MEN modules used for: - Innolight digital signals - Innolight analog signals (1) - Innolight analog signals (2) - Flip mirror devices digital in signals - Flip mirror devices digital out signals

Table 5: PMLSS Control Database Point

PMLSS runtime data is stored relative to the database point “<alias>pmlss:data”:

Attribute	Type	Description
laserOn	LOGICAL	Flag to indicate that the laser is “switched on”.
laserOK	LOGICAL	Flag to indicate that the laser status is OK.
laserStable	LOGICAL	Flag to indicate that the criteria to stabilize the laser is fulfilled.
WMWaveLength	vltDOUBLE	Wavelength measured by the WaveMeter.
closedLoop	LOGICAL	Indicator if the stabilization loop is closed.
instrumentName	vltBYTES8	Name of the instrument selected via the switchyard, and detected via proximity switches.
TACCmdTemperature	vltDOUBLE	Slow correction signal, as applied to the LightWave model 125 power supply (input SLOW FREQUENCY BNC), calculated by the TAC control loop.
tempSaturated	LOGICAL	Indicates if the upper limit of the MPV-955 range is reached, i.e. the command is saturated at 10 V.
TACCmdPiezo	vltDOUBLE	Fast correction signal, as applied to the LightWave model 125 power supply (input FAST FREQUENCY BNC), calculated by the TAC control loop.
pzSaturated	LOGICAL	Indicates if the upper limit of the MPV-955 range is reached,

		i.e. the command is saturated at 10 V.
TACErrorSignalX	vltDOUBLE	Error Signal “X” delivered by SR-844 lock-in amplifier
TACErrorSignalY	vltDOUBLE	Error Signal “Y” delivered by SR-844 lock-in amplifier
SR844DetFrequency SR844SecondHarmonic SR844Sensitivity SR844TimeConstant SR844FilterSlope	vltDOUBLE vltDOUBLE vltBYTES32 vltBYTES16 vltBYTES16	Measurements delivered by the SR-844 lock-in amplifier.
emergencyStop	vltLOGICAL	Status of the digital signal “ilEmergencyStop” as read by the AVME-9481 ACROMAG board
emStopOrigin	vltLOGICAL	Status of the digital signal “ilEmergencyStopOrigin” as read by the AVME-9481 ACROMAG board
emStopLatch	vltLOGICAL	Status of the digital signal “ilEmergencyStopLatch” as read by the AVME-9481 ACROMAG board
lampStatus	vltBYTES32	Status of the digital signal “ilEntranceDoor” as read by the AVME-9481 ACROMAG board. This is not a single signal, but the status of the 10 lamps mounted next to the entrance doors.
eomTemperature i2Temperature dcTemperature	vltDOUBLE vltDOUBLE vltDOUBLE	Temperature measurement of the NewPort CN77000 Temperature Controller used for the New Focus Model 3363-B EO Modulator. Temperature measurement of the NewPort CN77000 Temperature Controller used for the I2 cell. Temperature measurement of the NewPort CN77000 Temperature Controller used for the OVEN.
PMChannelA PMChannelB	vltDOUBLE vltDOUBLE	PowerMeter measurement for channel #A PowerMeter measurement for channel #B
LW125LaserTemp LW125DiodePower LW125InterlockStat LW125FaultStatus	vltDOUBLE vltDOUBLE LOGICAL vltBYTES16	Status of the LightWave Model 125 laser, if used (see CONFIG section).
ilPowerEst ilD1Power ilD2Power ilCrystalTECError ilD1TECError ilD2TECError ilD1TempGuard ilD2TempGuard ilNoiseHeatMon ilInterlockMon	vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE LOGICAL LOGICAL LOGICAL LOGICAL	Status of the InnoLight Model MIR 500 laser, if used (see CONFIG section).
psFSUAIp3 psFSUAIp1 psAMBERIp3 psAMBERIp1 psMIDIIp3 psMIDIIp1 psFSUAProbe psFSUARef psAMBERProbe psAMBERRef psMIDIProbe psMIDIRef	LOGICAL	Status of the proximity switches.

engFileReady engFileStatus	LOGICAL INT32	Flag to indicate that the requested number of samples has been recorded. Percentage of completion of the current engineering file.
AOMIp2Volt AOMIp4Volt AOMIp1Volt AOMIp3Volt	vltDOUBLE	Status of the AOM transmission settings.
LaserPower	vltDOUBLE	Laser power setting (see SETLPWR command) [mW].
primetAFlipON primetAFlipINBEAM primetAFlipOUTBEAM primetBFlipON primetBFlipINBEAM primetBFlipOUTBEAM	vltLOGICAL	Flags to indicate for each of the two FLIP mirror devices: - Device is powered ON - Device is IN the beam - Device is OUT of the beam

Table 6: PMLSS Data Database Point

2.2.10 Test and Simulation Support

The following files provide TAT tests for `pmlss`:

- `pmlssTestStdCmds.tcl` for testing the standard commands (TCL/TK script file).
- `pmlssTestCommands.tcl` for testing all PMLSS specific commands (TCL/TK script file).

Simulation support is enabled via the SIMULAT command. In that case, the laser stabilization control loop is synchronized by the TACLIB AuxClock block, at a rate of 100 Hz.

2.3 Phase Meter

2.3.1 Overview

All functionality of the "Phase Meter" package of PMCS shall be contained in the separate CMM module `pmacq`. PMACQ can be used as a stand-alone application, it does not depend on other PRIMA metrology modules. A maintenance panel is provided to send commands and display the status of PMACQ.

The following LCU architecture has been selected for the PMACQ application:

- 1 Motorola MVME-6100 PPC CPU board.
- 1 ESO Time Interface Module (TIM) board.
- 1 HPDI-32 digital-input board (PMC connector).
- 1 PMC-5565 RMN board (PMC connector).
- 1 MEN carrier board, equipped with 1xM58 (digital out) module.

A schematic of the HW layout of the PMACQ LCU and all cabling is given in Figure 12: PMACQ LCU.

The PRIMA metrology system consists of two heterodyne Michelson interferometers which are operated simultaneously, and have common optical paths with both observed stars through the VLTI optical train. The disturbance to be measured (ΔL) corresponds to the difference between the path variations recorded by the two Michelson interferometers. The phase difference between the two channels of the PRIMA metrology system is measured by a stand-alone device, the "Phase Meter" [AD 08].

Technically, in each metrology channel, the beams are first superimposed to form a reference signal which monitors the OPL variations from the laser to the Beam Launcher/Combiner. Then the beams are launched separately in the stallar paths. After a round-trip through the VLTI, the beams are combined again to form a probe signal. Both the reference and the probe signal are input to the Phase Meter, where they are detected and processed to compute the differential optical phase variations.

A TIM board is used to trigger the measurements of the “Phase Meter” with a high frequency (up to 8 kHz). The main task of PMACQ is to read the data generated by the “Phase Meter” at the same frequency. For that purpose, a dedicated high-speed digital input board HPDI-32 directly mounted to one of the two available PCI slots of the MVME-6100 CPU board is used. The HPDI-32 shall generate an interrupt when new data is available. The format of the data is shown in Figure 8: Phase Meter Data Format [AD 24].

	31 – 28	27 – 24	23 – 20	19 – 16	15 – 12	11 – 8	7 – 4	3 – 0
Word #0	Packet#	SOT	Undef.	Undef.	*Phase sum MSB*			
Word #1	*Phase Sum LSB*							
Word #2	Undef.	Undef.	Undef.	*Number of Samples (starting from Bit 18!)*				
Word #3	Undef.	Undef.	*Error Compensation*					
Word #4	DC Probe 650k		DC Probe 450k		DC Ref. 650k		*DC Ref. 450k*	
Word #5	*Status II (10 Bits!)*		*Undef.*	*Undef.*	*Undef.*	*Undef.*	*Packet#*	*EOT*

Figure 8: Phase Meter Data Format [AD 24]

The “Packet#” is a 4-bit number which is incremented for each packet of data. To ensure the integrity and sequentiality (no blocks lost) of the data, PMACQ shall implement the following two checks:

1. SOB xor EOB = 1 (“Integrity”)
2. SOB(i) = (SOB(i – 1) + 0x10) & 0xFF (“Sequentiality”)

If both checks are successful, PMACQ shall extract the following data from the transmitted block:

1. Summed Phase (48 Bits) Φ
2. Number of Samples (19 Bits) N
3. Error Compensation (24 Bits) Φ_C

For monitoring purposes, the Phase Meter status data will be evaluated. In addition to the DC level of the four photodetectors (“status I”), the “status II” field of the data block provides the following information:

Bit	Description
Bit #9	Reset Detected
Bit #8	Trigger Detected
Bit #7	FC Overflow
Bit #6	PM Overflow
Bit #5	200k Probe Detected
Bit #4	PLL Locked
Bit #3	650k Probe Detected
Bit #2	450k Probe Detected
Bit #1	650k Ref Detected
Bit #0	450k Ref Detected

Figure 9: Status II Data Format [AD 24]

The main task PMACQ LCU is to apply the formula shown in Figure 10: Calculation of ΔL to the raw data delivered by the “Phase Meter” in order to generate the quantity ΔL, which is relevant for PRIMA operations. ΔL must then be written to the PMC-5565 reflective memory board, together with a timestamp coming from the TIM board and a validity flag, in order to be delivered to the (d)OPD controller.

$$\Delta L = \frac{c}{4\pi}\left[\frac{\Phi}{\nu N} - \frac{\Phi_C}{\Delta\nu}\right]$$

Figure 10: Calculation of ΔL

Within the above formula, c, ν and Δν are constants which represent the speed of light, the frequency of the stabilized laser, and the frequency shift between the two metrology channels. They are configured through the PMACQ database, and initially set to the following values:

Constant	Value
c	$3 * 10^8$ m/s
ν	c/1319 nm = $2.27445 * 10^{14}$ Hz
Δν	78 MHz

Figure 11: PMACQ Constants

The "Phase Meter" package shall run on the two dedicated "Phase Meter" LCUs, lprmac and lprma2. It consists of a LSF application [AD 18] that provides the database and command handling, and implements a ESO standard state machine.

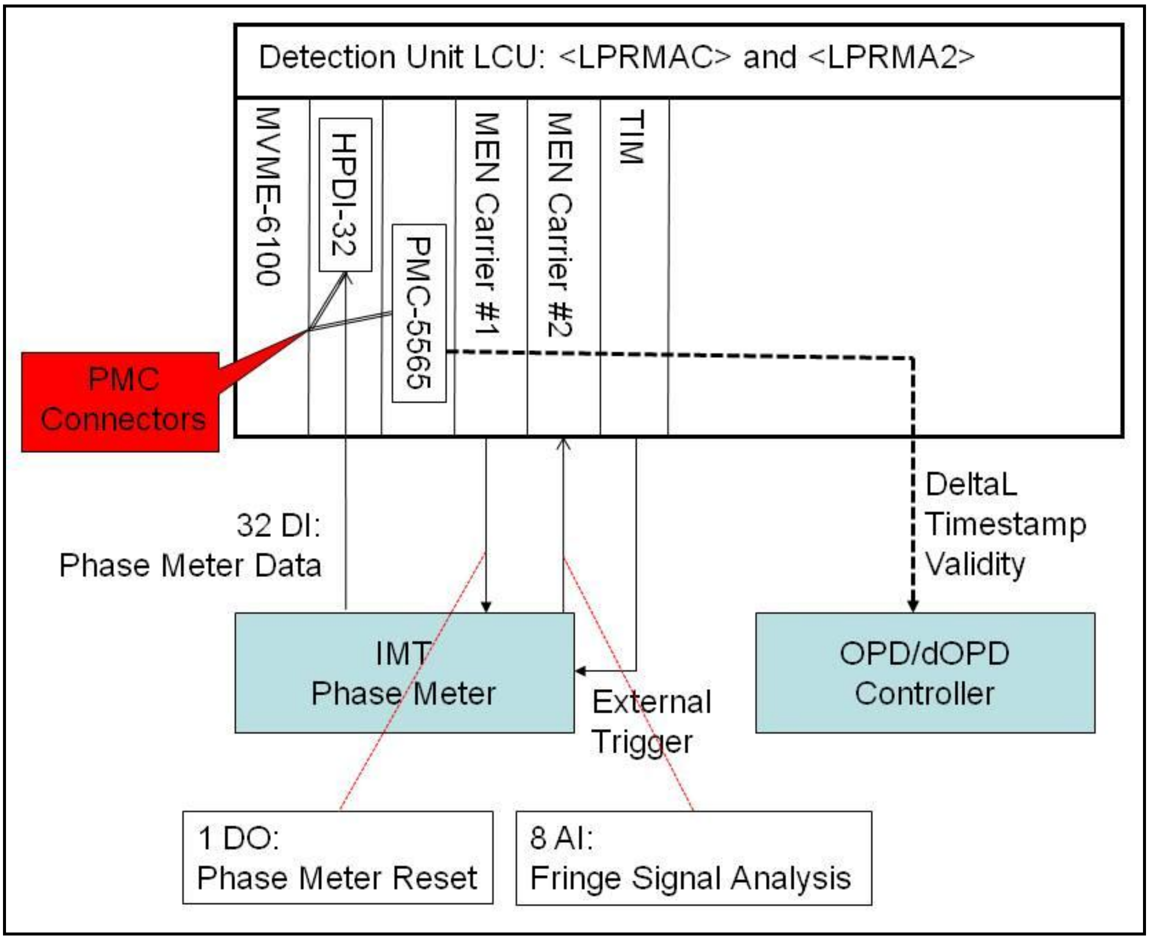

Figure 12: PMACQ LCU

The state chart of PMACQ is outlined in Figure 13: PMACQ State Chart. While in sub state IDLE, it is possible to change and query the recording rate of the data with the SETRATE and GETRATE commands, and to zero the metrology in one of three possible ways. The actual data acquisition is started with the STRTMET command, and stopped with STOPMET. During the data acquisition is running, the sub state is changed to ACTIVE. During ACTIVE, data is read from the Phase Meter, the value ΔL is calculated and is delivered to the (d)OPDC via the RMN network.

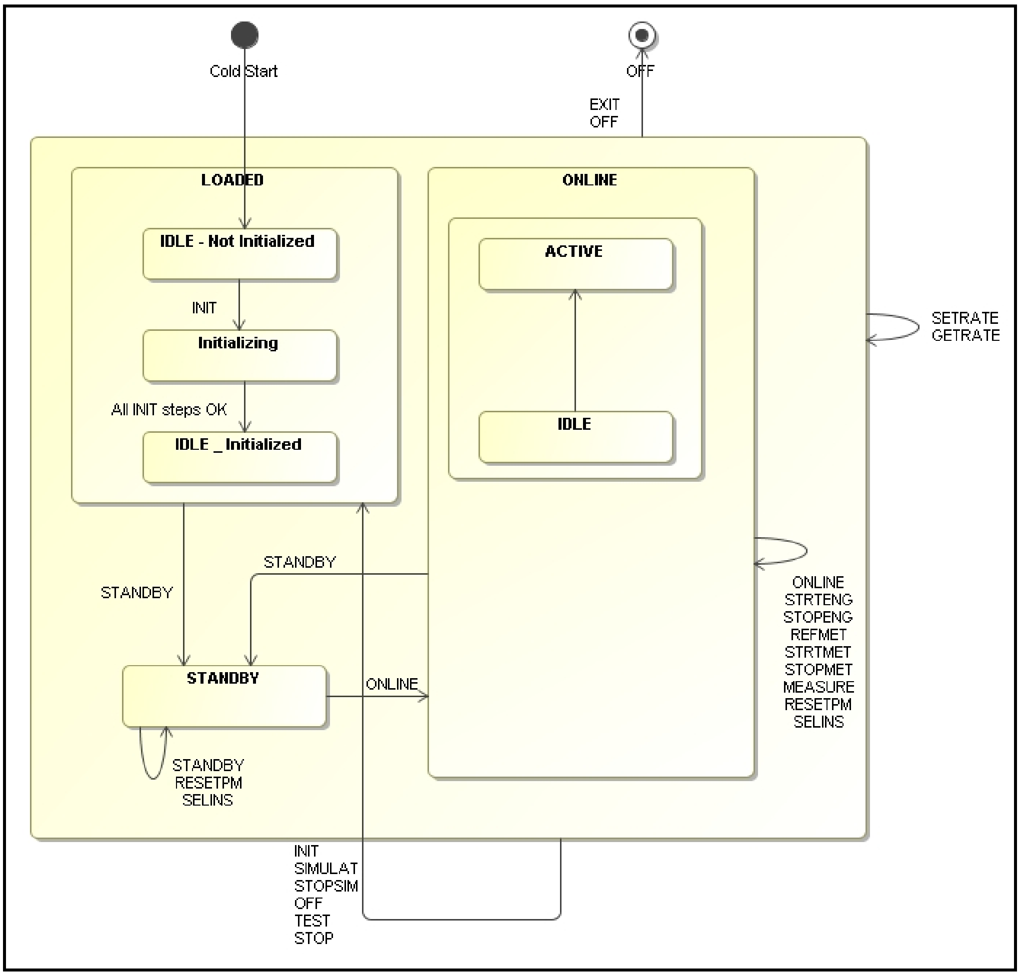

Figure 13: PMACQ State Chart

The acquisition algorithm shall run at 8 kHz. WindView measurements have been taken to verify CPU calculation times, see Figure 14: PMACQ LCU, 10 ms measurement.

The measurement outlines that the available CPU time is equally divided between the data acquisition/Delta L calculation (50 us), and CPU idle time (75 us). The CPU IDLE time (=not used) is more than 50 percent.

RTDScope can be started for each of the PMACQ LCU's by pressing the "RTDScope" button in the PMACQ GUI. Command INIT, ONLINE in the RTDScope GUI to display the data delivered by the RT algorithm. The following information can be displayed in realtime, see **Error! Reference source not found.**:

- Calculated "deltaL" value
- DC levels of the Photodiodes: Probe 650k, Probe450k, Ref. 650k, Ref. 450k

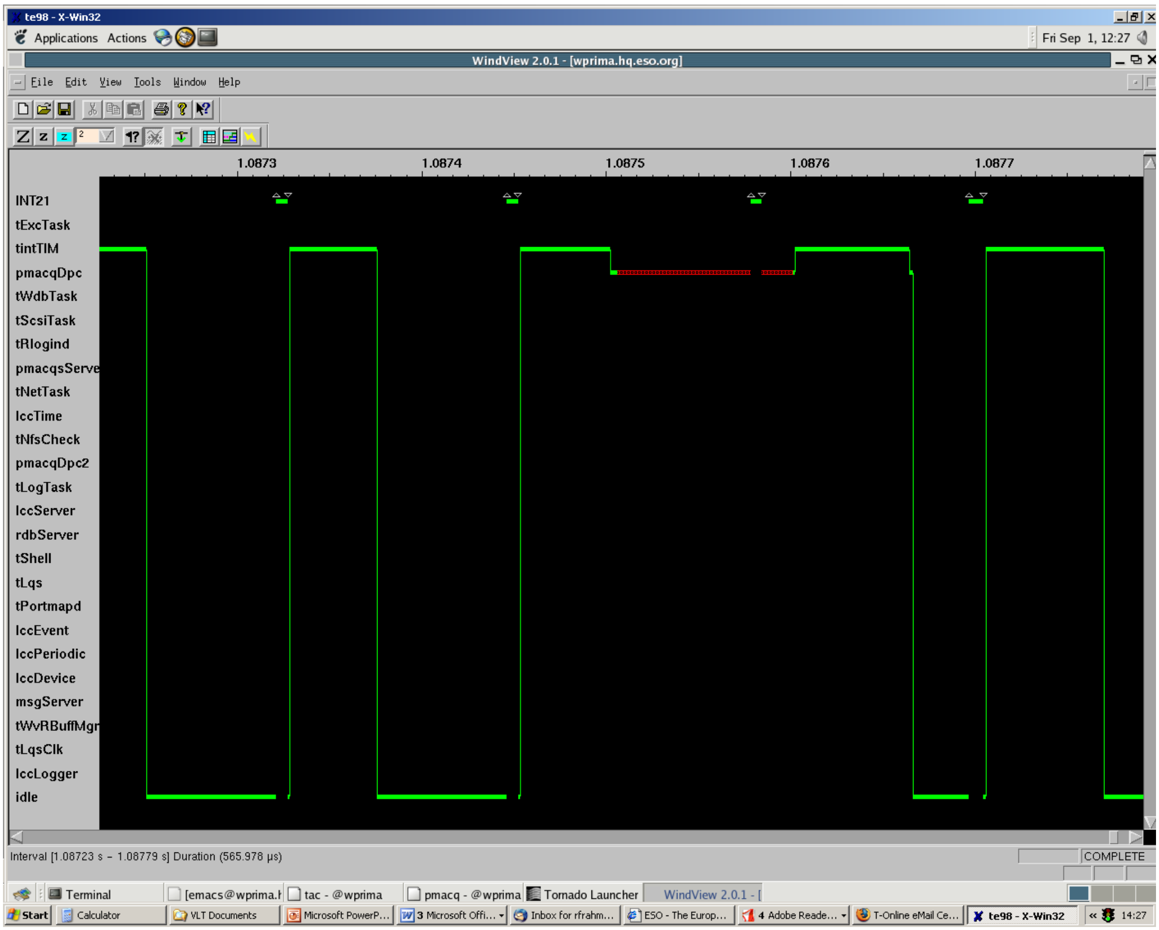

Figure 14: PMACQ LCU, 10 ms measurement

2.3.2 Command Description

Command	SETRATE
Parameters	<Rate> (REAL)
Reply	OK/ERROR
Description	The SETRATE command is used to change the data rate of the Phase Meter.

Command	GETRATE
Parameters	None
Reply	<Rate> (REAL)
Description	The GETRATE command is used to query the data rate of the Phase Meter.

Command	REFMET
Parameters	<Reference Mode> (STRING) <Timestamp> (STRING)
Reply	OK/ERROR
Description	The REFMET command is used to perform a software referencing of the PRIMA metrology in one of four different ways. ZERO just sets the currection factor to 0, while AVG sets the correction factor to the arithmetic mean of the last <n> measurements, and CURRENT sets it to the current measurement. Moreover, a timestamp can be supplied within <TS>. In that case, the correction factor is set to the measurement taken at that specific UTC timestamp which follows the mode parameter.

Command	STRTMET
Parameters	None
Reply	OK/ERROR
Description	The STRTMET command is used to start processing data coming from the Phase Meter, and to deliver the quantity Delta L to the RMN network.

Command	STOPMET
Parameters	None
Reply	OK/ERROR
Description	The STRTMET command is used to stop processing data coming from the Phase Meter, and to stop delivering the quantity Delta L to the RMN network.

Command	STRTENG
Parameters	<Number of Samples> (INTEGER)
Reply	OK/ERROR
Description	The STRTENG command is used to start recording data coming from the Phase Meter into engineering files.

Command	STOPENG
Parameters	None
Reply	OK/ERROR
Description	The STRTENG command is used to stop recording data coming from the Phase Meter into engineering files.

Command	RESETPM
Parameters	None
Reply	OK/ERROR
Description	The RESETPM command is used to perform a hardware reset of the phase meter.

Command	MEASURE
Parameters	None
Reply	<DeltaL> (STRING) <Timestamp> (STRING)
Description	The MEASURE command returns the latest Phase Meter measurement together with a timestamp.

Command	SELINS
Parameters	<Instrument Name> (STRING)
Reply	OK/ERROR
Description	SELINS selects the instrument to be used for the observation. The value must be either FSUA, AMBER or MIDI. NOTE: This information is ONLY used for the engineering files, and not required!

2.3.3 Task Description

Task Name	**pmacqMonitor**
Description	pmacqMonitor serves as a background monitor running at low frequency (1 Hz), which is used to update the status of the OLDB with the data delivered from the Phase Meter and the results/errors of evaluating this data.
Basic Course	1. `pmacqMonitor` shall evaluate the status of the “raw” Phase Meter quantities

	summed phase, number of samples, error compensation, "status II" information, and the DC level of the four photodiodes. Additionally, `pmacqMonitor` shall read the quantity ΔL calculated by `pmacqISR`. `pmacqMonitor` shall store these values into the OLDB. 2. `pmacqMonitor` shall read the umber of blocks that failed either the integrity or sequentality check. If one of these numbers has changed, `pmacqMonitor` issues either the error `pmacqERR_SEQUENTIALITY` or `pmacqERR_INTEGRITY` to the CCS log system. 3. `pmacqMonitor` shall calculate statatistic data, to be logged on the CCS logging system as FITS keywords. 4. `pmacqMonitor` shall read the status of the "fringe sensor analysis board" is running on the A-B LCU, and update the OLDB accordingly with the following information: - REF 450k OUT DC - REF 450k OUT RMS - REF 650k OUT DC - REF 650k OUT RMS - PROBE 450k OUT DC - PROBE 450k OUT RMS - PROBE 650k OUT DC - PROBE 650k OUT RMS A "visibility"attribute shall be derived for each pair of RMS/DC values, to be calculated with the formula: visibility = sqrt(2) * RMS / DC

Task Name	**pmacqsISR**
Description	`pmacqsISR` acts as the interrupt service routine called whenever new Phase Meter data has been transferred to the local memory of the CPU by the HPDI32 digital input board. The purpose of `pmacqsISR` is to evaluate the new Phase Meter data (check for lost blocks), calculate the quantity ΔL, and finally to deliver ΔL to the (d)OPDC via the reflective memory network.
Basic Course	1. `pmacqsISR` resets the HPDI32 DMA status register to clear the interrupt request 2. `pmacqsISR` reads the HPDI32 board status register. In case the status register indicates that the RX FIFO is not empty, the data transfer failed since all received data should have been transferred to the local memory of the CPU by DMA. In that case, `pmacqsISR` tries to recover by resetting the FIFO. 3. `pmacqsISR` converts the input buffer from little endian to big endian. 4. `pmacqsISR` checks the integrity and sequentiality of the data by evaluating SOB xor EOB = 1 ("Integrity") and SOB(i) = (SOB(i – 1) + 0x10) & 0xFF ("Sequentiality"). In case one of the equations is not fulfilled, the received data is ignored, and an error logged to the CCS log system. 5. `pmacqsISR` calculates the quantity ΔL, using the received data from the Phase Meter, and the constants c, ν and Δν from the OLDB. It shall then subtract from ΔL the value set via the last REFMET command. 6. `pmacqsISR` writes ΔL (together with a time stamp and validity flag) to the RMN, using a dedicated function of the CMM module RMACRFM. 7. `pmacqsISR` increases a global variable `blockCounter` that reflects the number of correctly processed Phase Meter data blocks. 8. In case of failures, `pmacqsISR` shall send a error message to the pmacq background monitor using `pmacqsShmSendError()`.

2.3.4 Database Description

The structure of the PMACQ database is outlined in Figure 15: PMACQ OLDB Structure.

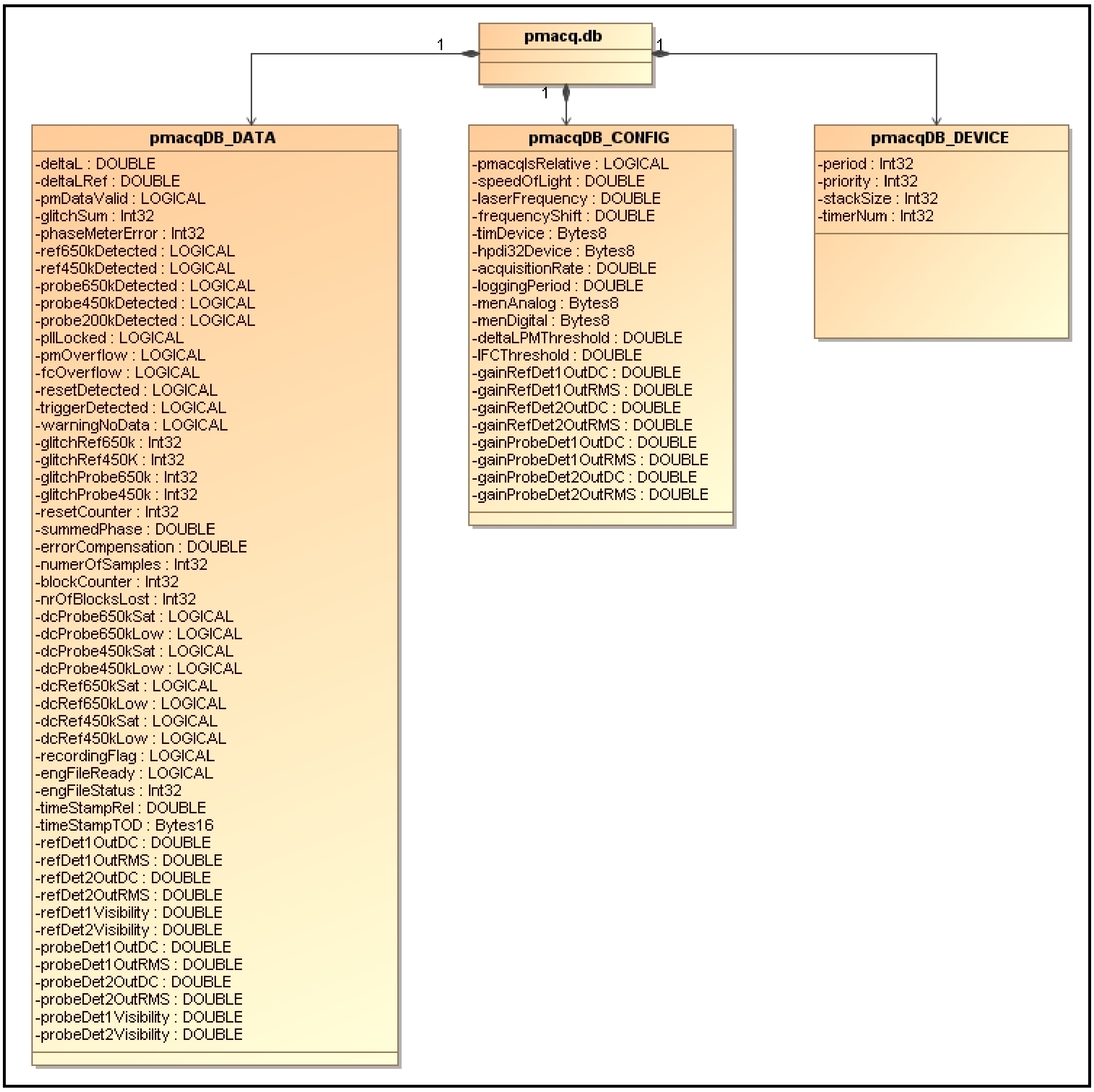

Figure 15: PMACQ OLDB Structure

All PMACQ configuration data is stored relative to the database point “<alias>pmacq:config”:

Attribute	Data Type	Description and Unit	Default Value
pmacqIsRelative	LOGICAL	Indicates if this instance of pmacq measures the DIFFERENTIAL (A-B) or the SINGLE CHANNEL (-B) signal.	
speedOfLight	vltDOUBLE	Constant which represents the speed of light [m/sec].	299.792.458
laserFrequency	vltDOUBLE	Constant which represents the frequency of the stabilized laser [Hz].	c/1319
frequencyShift	vltDOUBLE	Constant which represents the frequency shift between the two metrology channels [MHz].	78.000.000

timDevice	vltBYTES8	Name of the TIM device that will be used to send the external trigger to the Phase Meter.	"/tim0"
hpdi32Device	vltBYTES8	Name of the HPDI32 device used to read the data from the Phase Meter.	"/hpdi32"
acquisitionRate	vltINT32	Data rate at which the Phase Meter data is evaluated and recorded. Actually this is the configuration of the TIM timer which outputs the external trigger signal to the Phase Meter.	8 kHz
loggingPeriod	vltDOUBLE	Period [sec] for periodic FITS logs.	30
deltaLPMThreshold	vltDOUBLE	Threshold for additional warnings defined in VLT-SPE-ESO-15736-3899.	345.0
lFCThreshold	vltDOUBLE	Threshold for additional warnings defined in VLT-SPE-ESO-15736-3899.	10.99
gainRefDet1OutDC gainRefDet1OutRMS gainRefDet2OutDC gainRefDet2OutRMS gainProbeDet1OutDC gainProbeDet1OutRMS gainProbeDet2OutDC gainProbeDet2OutRMS	vltDOUBLE	Initial gain values for M36 analog IN board used to read the status of the "fringe sensor analysis board".	3.3 12.2 3.3 12.2 3.3 12.2 3.3 12.2

Table 7: PMACQ Control Database Point

All PMACQ runtime data is stored relative to the database point "<alias>pmacq:data":

Attribute	Data Type	Description and Unit
phaseMeterError	vltINT32	Error bits from the "Status II" field, as sent by the Phase Meter. Only bits 0 to 9 are used. Please note that all error bits are only monitored, however in case of an error no consequent action is carried out.
Ref450kDetected Ref650kDetected Probe450kDetected Probe650kDetected Probe200kDetected PLLLocked FCOverflow PMOverflow ResetDetected TriggerDetected	LOGICAL	Individual status attributes of the phase meter, extracted from "phaseMeterError".
warningNoData	LOGICAL	Flag to indicate that no new data is received from the phase meter. Set by the PMACQ background monitor task and checked every second.
glitchRef650k glitchRef450k glitchProbe650k glitchProbe450k resetCounter	vltINT32	Glitch counters for maintenance. Number of phasemeter HW reset commands (see RESETPM command), sent by the operator.
summedPhase	vltDOUBLE	Phase sum (48 Bits), as delivered by the Phase Meter. Originally, this value is delivered as a scaled integer, with LSB = $2\pi/1024$ rad, but converted to vltDOUBLE for the OLDB.
errorCompensation	vltDOUBLE	Error compensation value delivered by the Phase Meter.
numberOfSamples	vltINT32	Number of samples (19 Bits), as delivered by the Phase Meter.

errorCompensation	vltDOUBLE	Error compensation (24 Bits), as delivered by the Phase Meter. Originally, this value is delivered as a scaled integer, with LSB = 0.66 µm, but converted to vltDOUBLE for the OLDB.
deltaL	vltDOUBLE	Result of applying the formula from Figure 10: Calculation of ΔL to the raw data set delivered by the Phase Meter.
deltaLRef	vltDOUBLE	Reference value subtracted from the current deltaL measurement, set via the REFMET command.
pmDataValid	LOGICAL	Combined PMACQ status attributes for PMCS.
glitchSum	Int32	Combined PMACQ status attributes for PMCS.
blockCounter	vltINT32	Once the metrology has been started with the STRTMET command, PRIMET increases the database attribute `blockCounter` with every data set received from the Phase Meter.
nrOfBlocksLost	vltINT32	Once the metrology has been started with the STRTMET command, PRIMET verifies the sequentiality of the Phase Meter data with the equation `SOB(i) = ( SOB(i – 1) + 0x10 ) & 0xFF`. In case blocks are found to be missing, the database attribute `nrOfBlocksLost` is incremented.
timeStampRel timeStampTOD	DOUBLE vltBYTES16	Relative timestamp which reflects the time STRTMET was commanded.
recordingFlag engFileReady engFileStatus	LOGICAL LOGICAL vltINT32	Flag to indicate if the STRTENG command is active. Flag to indicate that the requested number of samples has been recorded. Percentage of completion of the current engineering file.
dcProbe650k	vltUINT8	DC level of the photodetector for the 650k probe signal.
dcProbe450k	vltUINT8	DC level of the photodetector for the 450k probe signal.
dcRef650k	vltUINT8	DC level of the photodetector for the 650k reference signal.
dcRef450k	vltUINT8	DC level of the photodetector for the 450k reference signal.
dcProbe650kSat dcProbe650kLow dcProbe450kSat dcProbe450kLow dcRef650kSat dcRef650kLow dcRef450kSat dcRef450kLow	LOGICAL	“Saturation” and “low signal” flags of the above photodetector signals. Kept for 1 second (to appear on the PMACQ GUI), then reset by the PMACQ background monitor.
refDet1OutDC refDet1OutRMS refDet2OutDC refDet2OutRMS refDet1Visibility refDet2Visibility probeDet1OutDC probeDet1OutRMS probeDet2OutDC probeDet2OutRMS probeDet1Visibility probeDet2Visibility	vltDOUBLE	Status of the “fringe sensor analysis board”, i.e. current state of the analog signals: - REF 450k OUT DC - REF 450k OUT RMS - REF 650k OUT DC - REF 650k OUT RMS - PROBE 450k OUT DC - PROBE 450k OUT RMS - PROBE 650k OUT DC - PROBE 650k OUT RMS A “visibility”attribute shall be derived for each pair of RMS/DC values, to be calculated with the formula: visibility = sqrt(2) * RMS / DC

Table 8: PMACQ Data Database Point

2.3.5 Test and Simulation Support

The following files provide TAT tests for `pmacq`:

- `pmacqTestStdCmds.tcl` for testing the standard commands (TCL/TK script file).
- `pmacqTestCommands.tcl` for testing all PMACQ specific commands (TCL/TK script file).

Simulation support is enabled via the SIMULAT command. In that case, the (simulated) phase meter results are generated at the rate of the system clock (100 Hz).

2.4 Position Sensor Detection

2.4.1 Overview

All functionality of the “Position Sensor Detection” package of PMCS is contained in the separate CMM module `pmpsd`. PMPSD is normally controlled by PMCS, although it can also be used as a stand-alone application. A maintenance panel is provided to send commands and display the status of PMPSD.

The following LCU architecture has been selected for the PMPSD application:

- 1 Motorola MVME-6100 PPC CPU board.
- 2 VMIVME-3123 analog input boards.
- 1 ESO Time Interface Module (TIM) board.
- 1 MEN carrier board, equipped with 2xM58 modules.

A schematic of the HW layout of the PMPSD LCU and all cabling is given in Figure 17: PMPSD LCU.

The principle of one PSD cell is given in Figure 16: PMPSD HW Setup [AD 11].

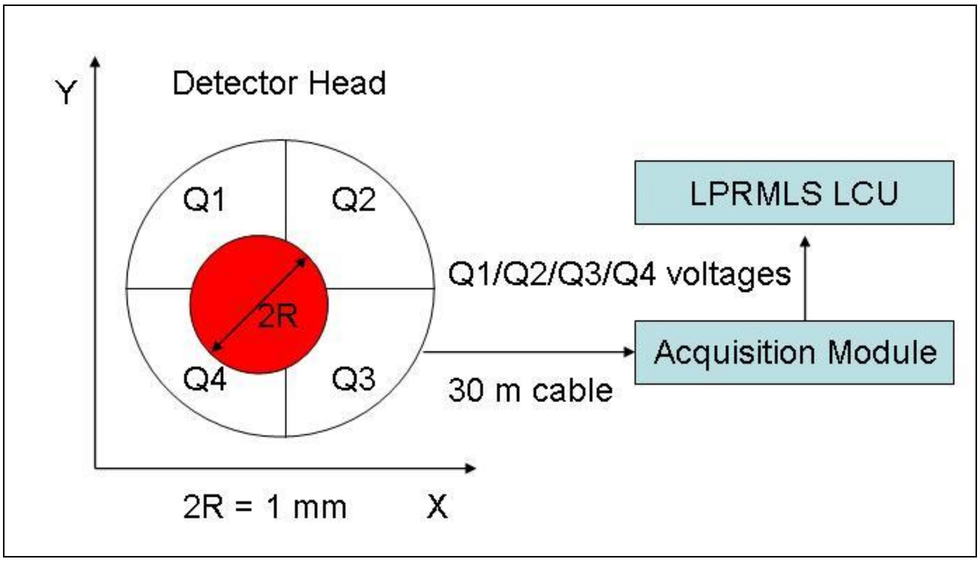

Figure 16: PMPSD HW Setup

Each PSD delivers four analog signals, one voltage per quadrant (Q1, Q2, Q3, Q4), in the range of [0-10V]. Using this setup, the lateral displacement of each beam can be computed using the formulas:

$\Delta X = [(Q2+Q3) - (Q1+Q4)] / sum$
$\Delta Y = [(Q1+Q2) - (Q3+Q4)] / sum$

with sum = Q1+Q2+Q3+Q4.

A total of 8 photo detectors (PSD’s) are required to monitor the lateral position of all PRIMA beams: While two metrology beams are always directed to FSU B, the other two metrology beams are configurable to serve as an input to one of the instruments FSU-A, AMBER or MIDI. Setting up the cabling to use one specific instrument is a manual task. For each of the two beams to these four devices, one PSD is used.

The LCU shall compute the lateral beam displacement of each beam. Four active detectors are simultaneously operated. They correspond to the metrology arms monitoring the input channels Ip1 and Ip3 (channel A) and Ip2 and Ip4 (Channel B). The lateral displacements ΔX and ΔY of the sensors coordinate systems shall be transformed into the necessary corrections within the [U,V,W] coordinate system of the light duct of the two selected star separators.

For a description of the coordinate transformation please refer to [AD 11].

In total, 32 analog input signals must be processed by PMPSD, which shall be read via two VMIVME-3123 analog input boards. The processing of the signals is done via a TAC control loop. The principle is outlined inFigure 18: PMPSD TAC Configuration. For simplicity, only the four active PSDs are shown.

Since this is a closed loop system, one “DigitalTF” controller block is foreseen for each beam. For each combination of PSD – STS VCM LCU, a different set of control parameters might be necessary. The SW must be able to swap the parameters after receiving the configuration.

The star separators involved in the observation must previously be made known to PMPSD via PMCS. The corrections to the VCM LCUs are then sent via a dedicated ethernet interface, using the second Ethernet interface of the Motorola MVME-6100 CPU.

The sensitive area of each PSD is 10 mm x 10 mm, which is compatible with a 1 mm beam diameter, and a ±4.5 mm lateral displacement. The exact beam diameter of the laser footprint on the PSD is measured once with a dedicated stand-alone device, written to the OLDB and regarded stable afterwards.

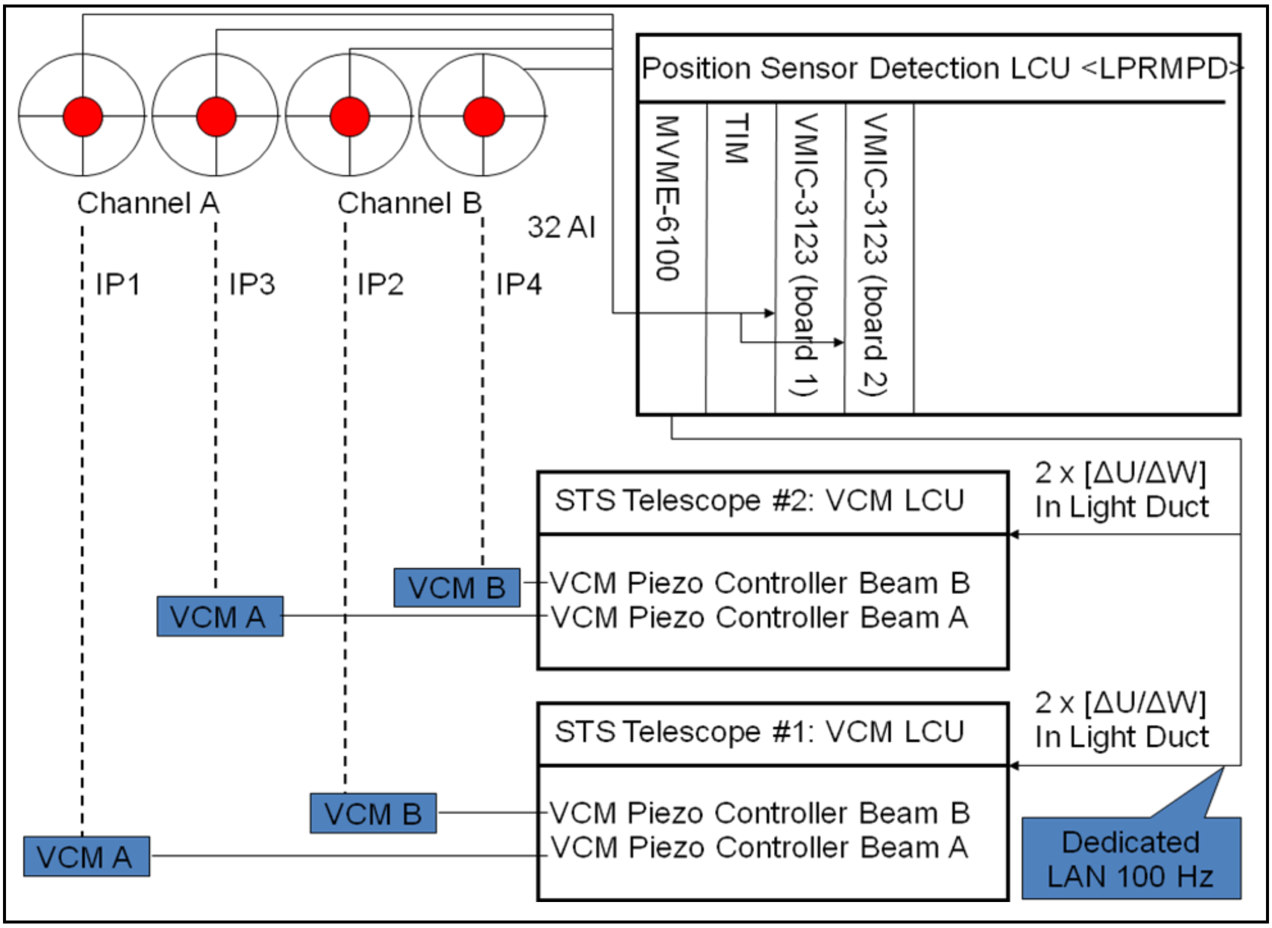

Figure 17: PMPSD LCU

Error vectors sent to the Star Separators during observation

During the observation, the information read from the PSD's is used to send error vectors to the Star Separators mounted on the two telescopes used for that observation. The frequency of the correction commands shall be 100 Hz.

PMPSD must first be configured to know the type and number of the two telescopes that are involved in the observation. These can be either two AT's, or two UT's. The CDT command SETMCFG shall be used to set up PMPSD with this information.

Due to the required frequency of 100 Hz, it is not possible to send the correction commands via a WS process. Instead, a direct UDP socket connection to the STS VCM LCUs shall be used via a dedicated ethernet network.

The following data shall be transmitted:

Field Name	Data Type	Description
beam1ErrU	vltDOUBLE	Error offset of beam 1 along the “U” axis [mm]
beam1ErrV	vltDOUBLE	Error offset of beam 1 along the “V” axis [mm]
beam1ErrW	vltDOUBLE	Error offset of beam 1 along the “W” axis [mm]
beam2ErrU	vltDOUBLE	Error offset of beam 2 along the “U” axis [mm]
beam2ErrV	vltDOUBLE	Error offset of beam 2 along the “V” axis [mm]
beam2ErrW	vltDOUBLE	Error offset of beam 2 along the “W” axis [mm]
Timestamp SEC	vltDOUBLE	Timestamp [sec]

Timestamp USEC	vltDOUBLE	Timestamp [usec]

Table 9: PMPSD Interface to the STS VCM LCUs

The X and Y coordinates delivered by each PSD device shall be multiplied with constant factors in order to convert them to the unit "mm". The "W" coordinate is always zero, since the PSD's are 2-dimensional. A conversion matrix shall be used to convert the PRIMET [X,Y,Z] system into the STS's [U,V,W] system.

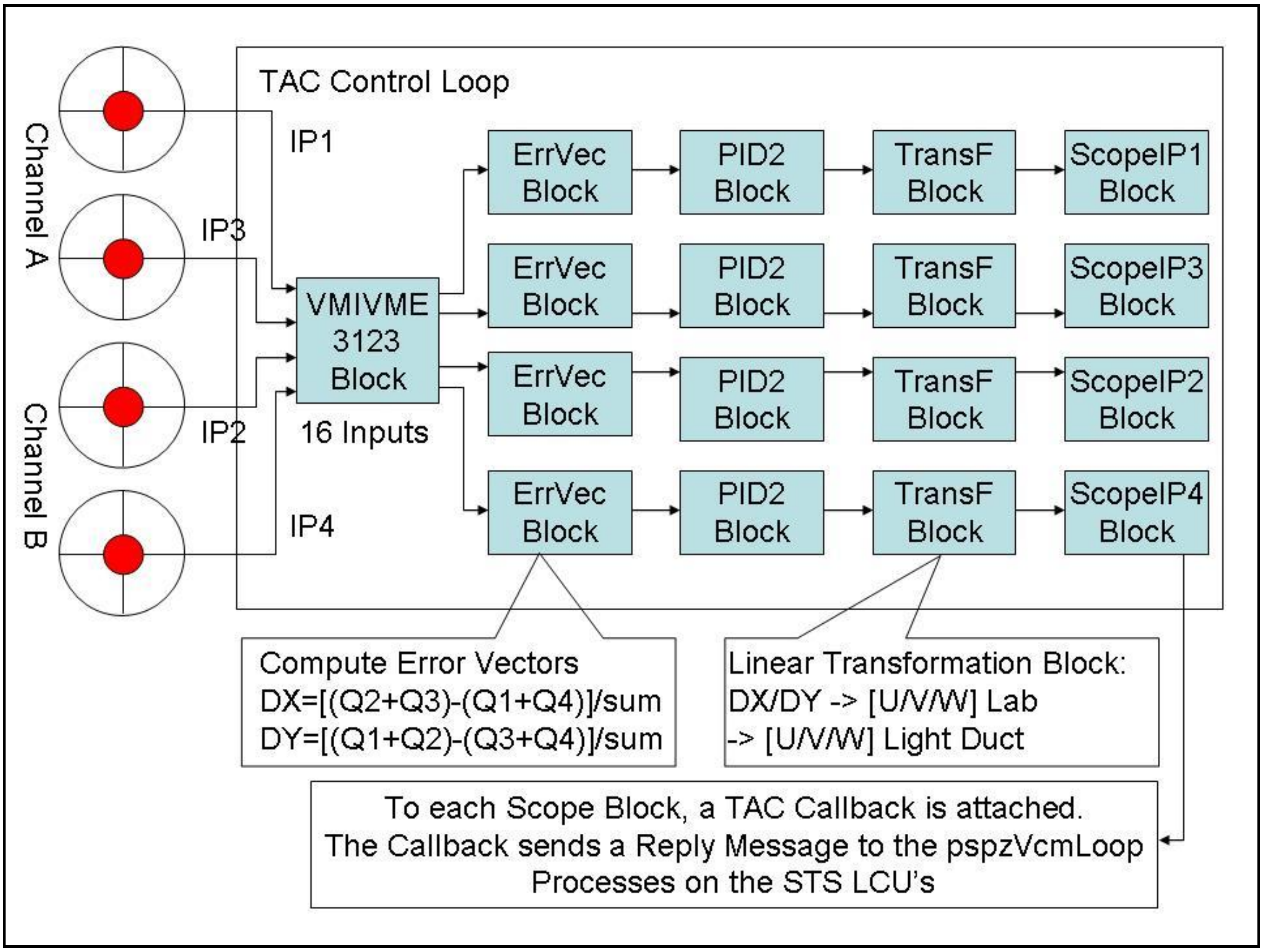

Figure 18: PMPSD TAC Configuration

The state chart of PMPSD is shown in Figure 19: PMPSD State Chart.

To every instrument AMBER, MIDI, FSU-A (and FSU B), a fixed pair of PSD's devices is assigned. This assignment is hardcoded within PMPSD. The SELINS command must be sent to PMPSD before starting the pupil optimization loop in order to determine the PSD ID's that shall be used.

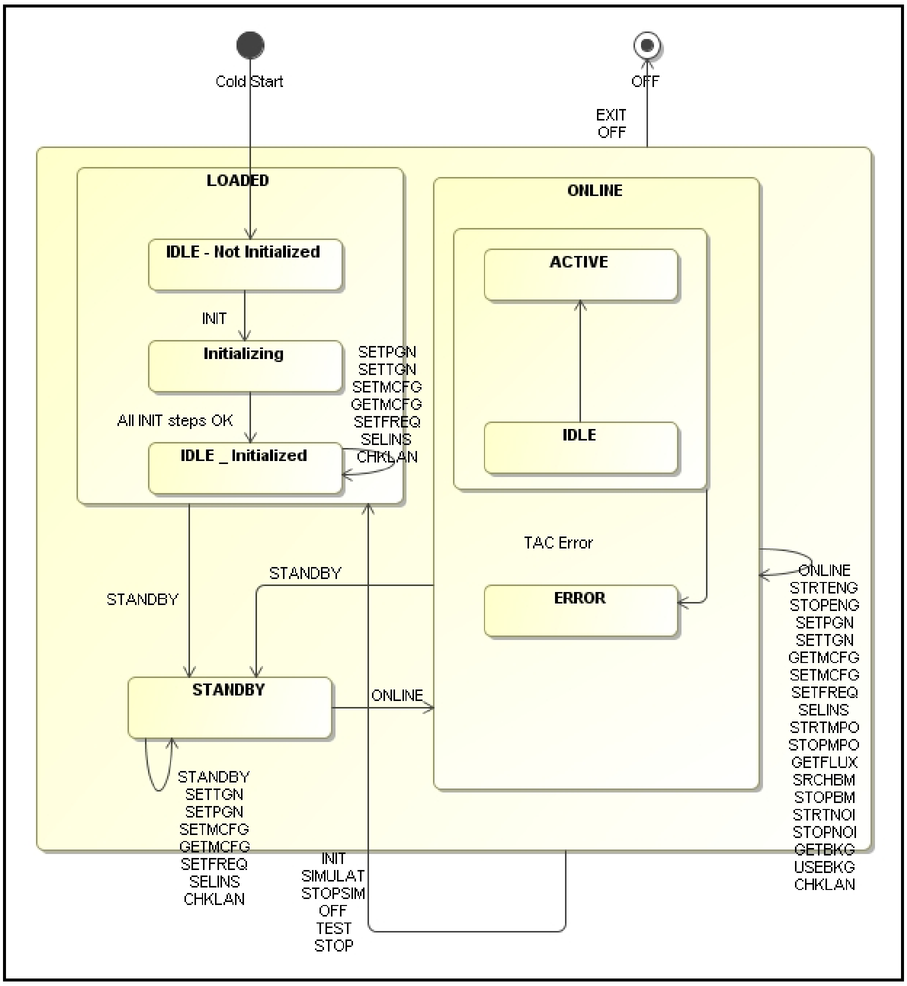

Figure 19: PMPSD State Chart

RTDScope can be started for the PMPSD LCU by pressing the "RTDSCOPE" button in the PMPSD GUI. Command INIT, ONLINE in the RTDScope GUI to display the data delivered by the RT algorithm. The following information can be displayed in realtime by connecting to the "Scope" block of the TAC RT algorithm:

- For each active quadcell: R = sqrt(x^2+y^2), R is the "running value".
- For each active quadcell: Q_SUM. Q_SUM is the "running value".

Thus, in total, 8 values can be displayed in realtime via RTDScope.

2.4.2 Command Description

Command	SETPGN
Parameters	<PSD ID> (INTEGER) <Gain> (REAL)
Reply	OK/ERROR
Description	SETPGN is used to set the PGA gain of quadcell <psdID> to one of the values 1, 10, or 100.

Command	SETTGN
Parameters	<PSD ID> (INTEGER) <Gain> (REAL)

Reply	OK/ERROR
Description	SETPGN is used to set the TRANSIMPEDANCE gain of quadcell <psdID> to one of the values 1, 2.22, 5.6, or 10.

Command	STRTMPO
Parameters	<PSD ID> (INTEGER)
Reply	OK/ERROR
Description	STRTMPO is used to start the pupil optimization loop between PMPSD and the star separators, which is active during the observation. Corrections are sent via socket communication to the process on the star separator LCUs in charge of controlling the variable curvature mirrors.

Command	STOPMPO
Parameters	<PSD ID> (INTEGER)
Reply	OK/ERROR
Description	STRTPPO is used to stop the pupil optimization loop between PMPSD and the star separators, which is active during the observation.

Command	SETMCFG
Parameters	<Telescope #1> (STRING) <Telescope #2> (STRING)
Reply	OK/ERROR
Description	SETCONF is used to define the IDs of the two telescopes that are used for one PRIMA observation. The parameters must be either ATx or UTx.

Command	GETMCFG
Parameters	None
Reply	<Telescope #1> (STRING) <Telescope #2> (STRING)
Description	GETMCFG is used to retrieve the IDs of the two telescopes that are used for one PRIMA observation, and the selected instrument.

Command	STRTENG
Parameters	<Number of Samples> (INTEGER)
Reply	OK/ERROR
Description	The STRTENG command is used to start recording data read from the quadcells into a engineering file.

Command	STOPENG
Parameters	None
Reply	OK/ERROR
Description	The STRTENG command is used to stop recording data read from the quadcells into a engineering file.

Command	SETFREQ
Parameters	<Frequency> (REAL)
Reply	OK/ERROR
Description	SETFREQ is used to set the correction rate of the messages sent to the STS of the telescopes.

Command	GETMFLX
Parameters	None

Reply	<X, Y, Q_SUM, IP1> (REAL) <X, Y, Q_SUM, IP2> (REAL) <X, Y, Q_SUM, IP3> (REAL) < X, Y, Q_SUM, IP4> (REAL)
Description	GETMFLX returns the flux (= sum of voltages) read from the 4 active quadcells Ip1, Ip3, Ip2, Ip4.

Command	SELINS
Parameters	<Instrument Name> (STRING)
Reply	OK/ERROR
Description	SELINS selects the instrument to be used for the observation. The value must be either FSUA, AMBER or MIDI.

Command	SRCHBM
Parameters	<Quadcell ID> (INTEGER)
Reply	OK/ERROR
Description	SRCHBM performs a spiral search, using the VCM of the STS as the actuator, to maximize the flux on the selected quadcell.

Command	STOPBM
Parameters	<Quadcell ID> (INTEGER)
Reply	OK/ERROR
Description	STOPBM stops any ongoing a spiral search.

Command	GETBKG
Parameters	<Quadcell ID> (INTEGER)
Reply	OK/ERROR
Description	GETBKG measures the background noise of one specific or all quadcells (i.e. the voltage output with the laser switched off) over a predefined period, calculates the average, and stores these measurements in the OLDB.

Command	USEBKG
Parameters	<Quadcell ID> (INTEGER)
Reply	OK/ERROR
Description	USEBKG subtracts the background noise measured with GETBKG from the actual measurements of one specific or all quadcells.

Command	STRTNOI
Parameters	<Quadcell ID> (INTEGER)
Reply	OK/ERROR
Description	Start injecting white noise as the user offsets to the STS VCM.

Command	STOPNOI
Parameters	<Quadcell ID> (INTEGER)
Reply	OK/ERROR
Description	Stop injecting white noise as the user offsets to the STS VCM.

Command	CHKLAN
Parameters	None
Reply	OK/ERROR
Description	CHKLAN verifies that the STS VCM LCUs connected via the private LAN are reachable via PING.

2.4.3 Task Description

The callbacks executed by the TAC probe blocks are also described in this chapter. Four procedures are required: `pmpsdCBBeam1`, `pmpsdCBBeam2`, `pmpsdCBBeam3` and `pmpsdCBBeam4`. Each procedure shall process one pair of beams, directed to either the FSU-A, FSU-B, MIDI or AMBER. The procedures are called by TAC whenever new input data is received by the probe blocks.

Task Name	**pmpsdMonitor**
Description	`pmpsdMonitor` serves as a background monitor running at low frequency (1 Hz), which is used to update the status of the OLDB with the signals of the measurements for X, Y, and the signal strength delivered by the four active PSD devices.
Basic Course	1. For each active PSD x (x=1..4), PMPSD shall update the OLDB with the following attributes: X Position, Y Position, Q1-Q4, Q(Sum), Beam Detected Flag, Beam Saturated Flag, Radial Motion RMS, Radial Motion MEAN, User and Correction (=calculated by TAC) Offsets. 2. `pmpsdMonitor` shall calculate statistical information, to be logged to the CCS logging system as FITS keywords in slow intervals (30 sec).

Task Name	**pmpsdServerTACCallback**
Description	The procedure `pmpsdServerTACCallback` shall be attached as a callback to each of the four probe blocks of the active PSDs. It is called whenever the corresponding probe block is evaluated, i.e. when new input data is available from the PSD devices.
Basic Course	1. In case USEBKG has been commanded for that PSD, subtract the background noise from the measured analog voltages. 2. In case SRCHBM has been commanded, retrieve the next user offset from the “spiral” input file, and convert the offets to the STS [U,V,W] coordinate system. Otherwise, use the most recent “Beam Center” positions as the user offsets. 3. Calculate Q(SUM), and from this result the attributes “Beam Detected”, “Beam Saturated”, X Position, Y Position. 4. In case of “Beam Detected”: Start demodulation fuction for configurable number of seconds (default: 0.1 sec) to determine the shape of the beam. Return FAILURE to the beam search command in case the beam is distorted. 5. Otherwise: Stop ongoing SRCHBM, store the current position of the beam as the new “Beam Center”. Log a message to the CCS log system tat the beam was found. 6. In case of PPO active: Determine if the guiding should be stopped automatically in case of: beam lost, too large radial motion RMS, or beam saturated. Log a message to the CCS log system and stop PPO in these cases. 7. In case of PPO (still) active: Feed the X/Y positions into a TAC DigitalTF controller block, and use the outputs of that block as new controller offsets. 8. Convert the controller offsets from PMPSD [X,Y,Z] coordinates into the STS’s [U,V,W] coordinate system. 9. In case of STRTENG, store the PSD attributes into the local memory of the LCU. 10. Hand over the values SQRT(R) and Q(SUM) to the TAC RTDScope block. 11. Calculate statistical values to be logged to the CCS logging system at a fixed logging period: radial motion MEAN/RMS, X position MEAN/RMS, Y position MEAN/RMS, Q(SUM) MEAN/RMS 12. In case of PPO active: Send the user offsets + controller offsets, packed into one message for both beams, to the STS VCM LCUs of the corresponding telescopes.

2.4.4 Database Description

The structure of the PMACQ database is outlined in Figure 20: PMPSD OLDB Structure.

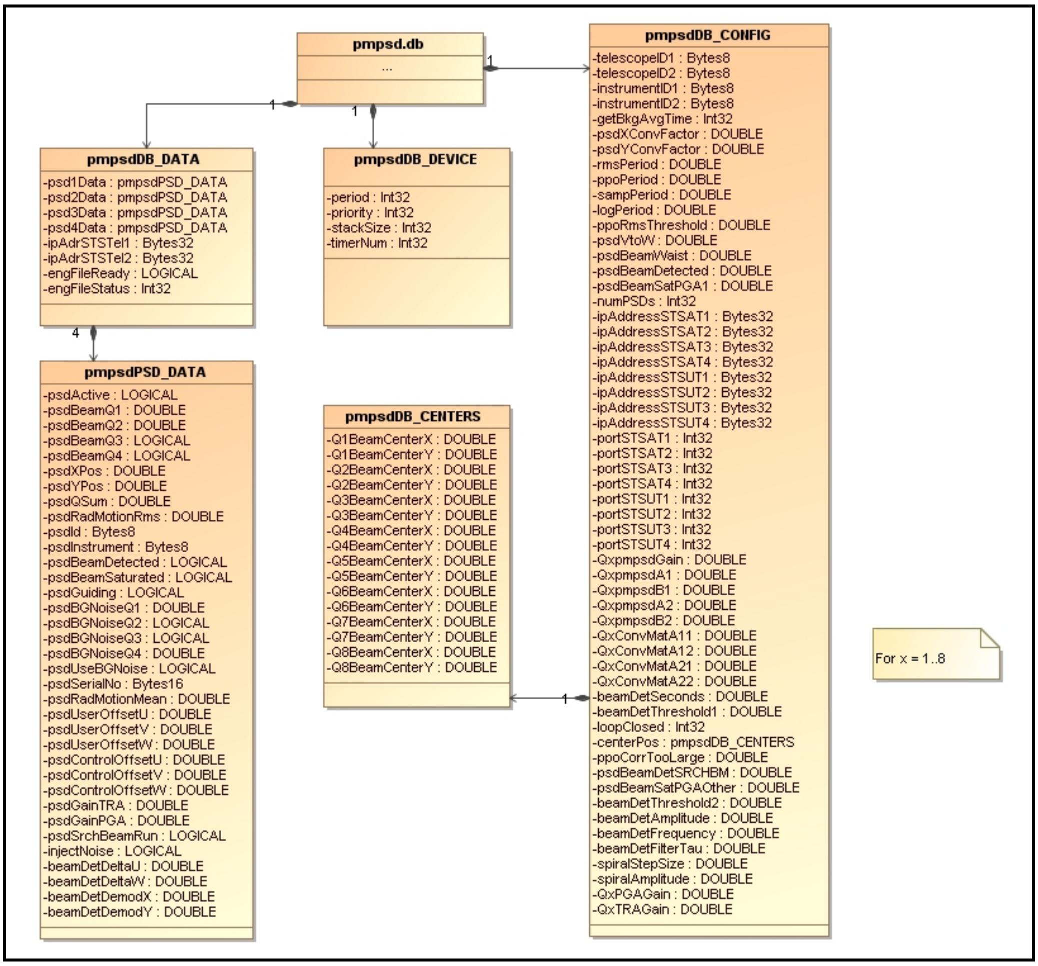

Figure 20: PMPSD OLDB Structure

PMPSD configuration data is stored relative to the database point “<alias>pmpsd:config”:

Attribute	Type	Description
Q1BeamCenterX Q1BeamCenterY Q2BeamCenterX Q2BeamCenterY Q3BeamCenterX Q3BeamCenterY Q4BeamCenterX Q4BeamCenterY Q5BeamCenterX Q5BeamCenterY Q6BeamCenterX Q6BeamCenterY Q7BeamCenterX Q7BeamCenterY Q8BeamCenterX Q8BeamCenterY	vltDOUBLE	Beam centers found during the most recent SRCHBM command. These values are permanently backup’ed into $VLTDATA/config by PMPSD.

rmsPeriod	vltDOUBLE	Time (in seconds) to compute the RMS of X, Y, and SUM.
ppoPeriod	vltDOUBLE	Frequency of the pupil optimization messages.
sampPeriod	vltDOUBLE	Frequency of the TAC control loop.
logPeriod	vltDOUBLE	Period [sec] for periodic PMPSD FITS logs.
ppoRmsThreshold	vltDOUBLE	Maximum RMS for the pupil motion. If larger, PPO is stopped automatically.
psdVtoW	vltDOUBLE	Conversion factor to convert [VOLTS] to [WATTS].
psdBeamWaist	vltDOUBLE	PSD Beam waist (=footprint of laser spot on one PSD), measured once by external detector during commissioning.
psdBeamDetected	vltDOUBLE	Threshold [V] to determine if the beam on one PSD is “Detected”.
psdBeamSaturated	vltDOUBLE	Threshold [V] to determine if the beam on one PSD is “Saturated”.
numPSDs	vltINT32	Number of PSD devices.
ipAddressSTSAT1 ipAddressSTSAT2 ipAddressSTSAT3 ipAddressSTSAT4 ipAddressSTSUT1 ipAddressSTSUT2 ipAddressSTSUT3 ipAddressSTSUT4	vltBYTES32	IP addresses of the VCM LCUs of the STS installed in AT<x> and UT<x>.
portSTSAT1 portSTSAT2 portSTSAT3 portSTSAT4 portSTSUT1 portSTSUT2 portSTSUT3 portSTSUT4	vltINT32	Port number used for the UDP socket connection to the STS VCM LCUs of the corresponding AT<x> or UT<x>.
Q<x>pmpsdGain Q<x>pmpsdA1 Q<x>pmpsdB1 Q<x>pmpsdA2 Q<x>pmpsdB2 Q<x>convMatA11 Q<x>convMatA12 Q<x>convMatA21 Q<x>convMatA22	vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE vltDOUBLE	Gain, A1, A2, B1, B2 settings for the TAC “DigitalTF” blocks used for the control loop between PMPSD and the STS VCM LCUs. Conversion matrix PMPSD[X,Y,Z] -> STS[U,V,W].
noiseAfterQuad divideFactor loopClosed	vltINT32 vltINT32 vltINT32	OLDB attributes for test purposes.
psdXConvFactor	vltDOUBLE	Constant factor that is multiplied with the X position delivered by each PSD device, in order to obtain the measurement in the unit “mm”.
psdYConvFactor	vltDOUBLE	Constant factor that is multiplied with the Y position delivered by each PSD device, in order to obtain the measurement in the unit “mm”.
telescopeID1	vltBYTES8	ID of the first telescope involved in the observation, as set by the SETCONF command. The format is either “AT<x>” or “UT<x>”, where <x> is a number between 1 and 4.
telescopeID2	vltBYTES8	ID of the second telescope involved in the observation, as

		set by the SETCONF command. The format is either “AT<x>” or “UT<x>”, where <x> is a number between 1 and 4.
instrumentID1	vltBYTES8	Name of the instrument used for PRIMA: AMBER, MIDI, or FSUA.
instrumentID2	vltBYTES8	Name of the Fringe Sensor, always “FSUB”.
Q<x>PGAGain Q<x>TRAGain	vltDOUBLE	PGA and TRA gain default values per quadcell, to be set upon INIT.
beamDetAmplitude beamDetFrequency beamDetFilterTau beamDetSeconds beamDetThreshold1 beamDetThreshold2	vltDOUBLE	Parameters for the demodulation algorithm to determine the beam shape after the beam has been found via the SPIRAL command.
spiralStepSize spiralStepAmplitude	vltDOUBLE	Parameters to generate the SPIRAL within the SRCHBM command.

Table 10: PMPSD Control Database Point

PMPSD runtime data is stored relative to the database point “<alias>pmpsd:data”:

Attribute	Type	Description
psdActive	LOGICAL	Flag to indicate if this PSD device is “active”, or has been disabled by the user.
psdBeamQ1 psdBeamQ2 psdBeamQ3 psdBeamQ4	vltDOUBLE	Analog voltages read from the 4 quadrants of the PSD device.
psdXPos psdYpos	vltDOUBLE	X, Y position of the laser beam on the PSD device, derived from the Q1-Q4 voltages.
psdQSum	vltDOUBLE	Sum of the Q1-Q4 voltages.
psdRadMotionRms	vltDOUBLE	Standard deviation of radial motion of the beam.
psdRadMotionMean	vltDOUBLE	Mean value of radial motion.
psdID	vltBYTES8	Identification of the PSD device.
psdInstrument	vltBYTES8	Instrument assigned to this PSD device.
psdBeamSaturated psdBeamDetected	LOGICAL	Flags to indicate if the beam on this PSD device has been “detected”, or if the PSD measurements are “saturated”.
psdGuiding	LOGICAL	Flag to indicate if Pupil Optimization (i.e. corrections to the STS VCM LCU) is “active” for this PSD device.
psdBGNoiseQ1 psdBGNoiseQ2 psdBGNoiseQ3 psdBGNoiseQ4	vltDOUBLE	Background noise of this PSD device, as measured using the GETBKG command.
psdUseBGNoise	LOGICAL	Flag to indicate if the background noise shall be subtracted from the current PSD measurements (see USEBKG command).
psdSerialNo	vltBYTES16	Serial number of this PSD device.
psdUserOffsetU psdUserOffsetV psdUserOffsetW	vltDOUBLE	User guiding offsets (e.g. beam search offsets).
psdControlOffsetU psdControlOffsetV psdControlOffsetW	vltDOUBLE	Controller guiding offsets after applied conversion matrix, i.e. in STS [U,V,W] system.

psdGainTRA psdGainPGA	vltDOUBLE	"Transimpedance" and "PGA" gain settings for the PSD device, see SETTGN/SETPGN commands.
pmpsdSrchBeamRun	LOGICAL	Flag to indicate if SRCHBM is currently running.
injectNoise	LOGICAL	Indicates that white noise is sent to the STS VCMs, following a STRTNOI command (engineering GUI).
beamDetDeltaU beamDetDeltaW beamDetDemodX beamDetDemodY	vltDOUBLE	Outputs of the demodulation algorithm to determine the shape of the beam.

Table 11: PMPSD Data Database Point

2.4.5 Test and Simulation Support

The following files provide TAT tests for `pmpsd`:

- `pmpsdTestStdCmds.tcl` for testing the standard commands (TCL/TK script file).
- `pmpsdTestCommands.tcl` for testing all PMPSD specific commands (TCL/TK script file).

Simulation support is enabled via the SIMULAT command. In that case, the (simulated) quadcell meter results are generated at the rate of the system clock (100 Hz).

2.5 PMCS Metrology WS Software

2.5.1 Overview

The PRIMA Metrology WS SW shall be contained in the separate CMM module pmcs. The functionality of the PRIMA Metrology WS SW is limited to setting up the system, and providing a single interface from PSS to the four PRIMET LCU's:

1. Configure the telescopes that shall be used for the observation: UTx and UTy or ATx and ATy.
2. Configure the instrument used for the observation: FSU-A, AMBER or MIDI.
3. Provide a state machine which enforces a certain order of commands to be sent until PRIMET is in it's final state ONLINE_RECORDING.
4. Provide a interface to the PRIMET interlock system (Siemens S7 safety PLC), using the OPC/UA protocol.

The PRIMA metrology control SW needs to know the instrument and telescope configuration to determine the parameters for the control loop between PMPSD and the VCM LCUs of the two Star Separator LCUs involved in the observation.

PMCS WS shall be implemented using WSF [AD 18], and implement the interfaces outlined in [AD 09]. The interface between each LCU and PMCS WS is implemented by a set of commands and a local database. The CCS scan system is used to automatically update the database attributes on the workstation when they change in the local database of one LCU. The pmcsControl process shall be started on the PRIMA WS to act as the command handler. PMCS WS interacts with the following processes running on the PRIMET LCUs:

- `pmacqServer`: Command handler task for the two PRIMET "Phase Meter" LCUs.
- `pmlssServer`: Command handler task for the PRIMET "Laser Stabilization" LCU.
- `pmpsdServer`: Command handler task for the PRIMET "Position Sensor Detection" LCU.

2.5.2 Interface to the PRIMET interlock system

The interface to the interlock system (safety PLC) is implemented using the OPC/UA protocol via a commercial library from Siemens [AD 28].

Unfortunately, the license issue for that library is complicated: After installing the library, it is required to send a fax to Siemens with a key generated during the installation, based on the MAC address of wprima. For this key, Siemens will send back a license file. Only with that file, the libary is usable.

Installation instructions:

(1) Please unzip all files from Softnet.zip to a folder on a PC, then ftp all "rpm" files in the "sw" folder to a temporary directory on wprima (for example, using "winscp"). As user "root", please "cd" to that temporary directory and run the following commands:

rpm -ihv cpitps-redist-1.1-b018.i386.rpm
rpm -ihv LIS-dummy-1.0-b001.i386.rpm
rpm -ihv pcmx-4.0-b012_plaintcp.i386.rpm
rpm -ihv s7h1-4.01-b016.i386.rpm

(2) Reboot wprima

(3) Check if the process "tnsxd" is running:
ps –ax | grep tnsxd
If it is not running, please start it manually:
/opt/lib/cmx/CMXinit start

(4) Obtain the ID for the license (as user "root") by running the script:
/usr/bin/get_address

(5) Create the file */usr/share/siemens/*.license.dat
Note: If this file already exists, please open it and remove all lines.
Paste the license string from the Siemens FAX into this file, then save.

(6) The file should now contain exactly one line of text, with the string pasted above.

(7) Reboot wprima, check if the process "tnsxd" is running:
ps -ax | grep tnsxd
ONLY if this process is NOT running, please start it manually:
/opt/lib/cmx/CMXinit start

(8) Run the following command:
/opt/bin/tnsxcom -u /usr/share/siemens/s7/example/tns_inp.dat

Within the VLTSW, the CMM module “ic0fbs7” provides generic functions to read from and write into data blocks of the PLC, for all basic data types: INTEGER, DOUBLE, LOGICAL. The following information shall periodically be read from the S7 PLC:

Address	Type	Description
DB4,W0,1	vltUINT16	remW1: laserEnStorRoom = (remW1 & 0x0100) laserEnAnteCh = (remW1 & 0x0200) laserEnUT1= (remW1 & 0x0400)

		laserEnUT2= (remW1 & 0x0800) laserEnUT3= (remW1 & 0x1000) laserEnUT4= (remW1 & 0x2000) laserEnG2= (remW1 & 0x4000) laserEnILCom = (remW1 & 0x0004) laserEnILComAck = ((remW1 & 0x0008) noInterlock = ((remW1 & 0x0040) localAckStoR = ((remW1 & 0x0080)
DB4,W2,1	vltUINT16	remW2: localAckAnteCh = ((remW2 & 0x0100) localAckUT1= ((remW2 & 0x0200) localAckUT2= ((remW2 & 0x0400) localAckUT3= ((remW2 & 0x0800) localAckUT4= ((remW2 & 0x1000) localAcknOK= ((remW2 & 0x2000) localAckAnteCD = ((remW2 & 0x4000) localAckG2 = ((remW2 & 0x8000) laserEnAnteCD= ((remW2 & 0x0001)
DB4,X4.0,1	vltLOGICAL	System failure
DB4,X4.1,1	vltLOGICAL	Laser enabled ON
DB4,X4.2,1	vltLOGICAL	Laser disabled OFF
DB4,X4.3,1	vltLOGICAL	Laser interlocked

The following data shall be written to the S7 PLC upon request by the user:

Address	Type	Description
DB5,X0.0,1	vltLOGICAL	Remote reset of PRIMET interlock system

2.5.3 Sequence Diagrams for common PMCS Command Scenarios

PMCS SELINS Command

CAUTION: It is not possible to actually *change* the PRIMA instrument (AMBER, MIDI, or FSU-A) via PMCS! Proximity switches read on the “Laser Stabilization” LCU allow to read, but not to modify the configured instrument. The SELINS command therefore uses the scenario outlined in Figure 21: PMCS SELINS Command:

- PMCS shall send GETINS to the “Laser Stabilization” LCU, which returns the instrument which is currently connected.
- PMCS compares the returned instrument with the parameter of the SELINS command, and returns ERROR if the two values differ.
- Otherwise, PMCS forwards the SELINS command to the two “Phase Meter” and the “Pupil Detection” LCUs, which do not know at all the configured instrument.
- PMCS returns OK to the originator of the SELINS command

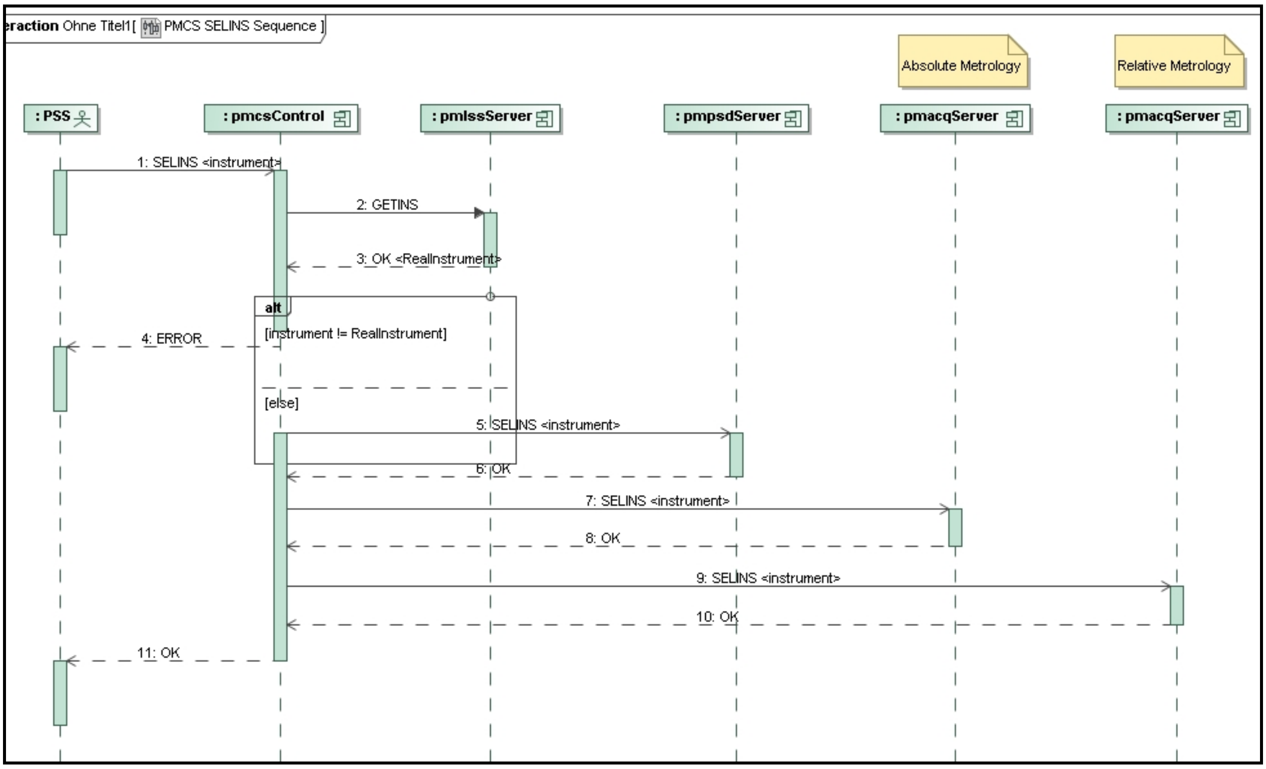

Figure 21: PMCS SELINS Command

PMCS SETCONF Command

Most PMCS commands consist of simple forwarding (and sometimes renaming) of one single command to a specific LCU, and returning the LCU's reply to the sender of the message. Figure 22: PMCS SETCONF Command displays the sequence diagram for these kind of actions. In addition to SETCONF, this approach applies to GETCONF, GETMFLX, STRTMPO, STOPMPO, STRTMFS, STOPMFS, SETMSRC, GETMSRC, SET, and POWER.

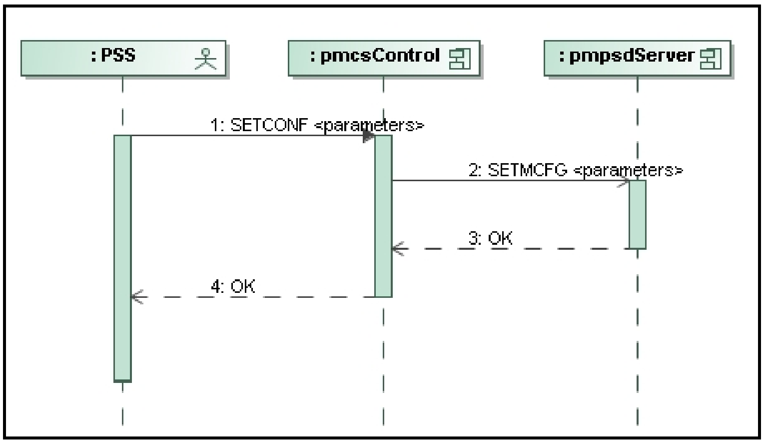

Figure 22: PMCS SETCONF Command

PMCS STRTMET Command

Since PMCS always controls all four PRIMET LCU's, commands to the "Phase Meter" SW PMACQ are distributed to the two identical LCUs lprmac and lprma2. If one of the LCUs is intended not to be used, it must be ignored via the PRIMA MSW table. The sequence diagram for the STRTMET command is outlined in Figure 23: PMCS STRTMET Command. Identical sequence diagrams can be applied to STOPMET and REFMET.

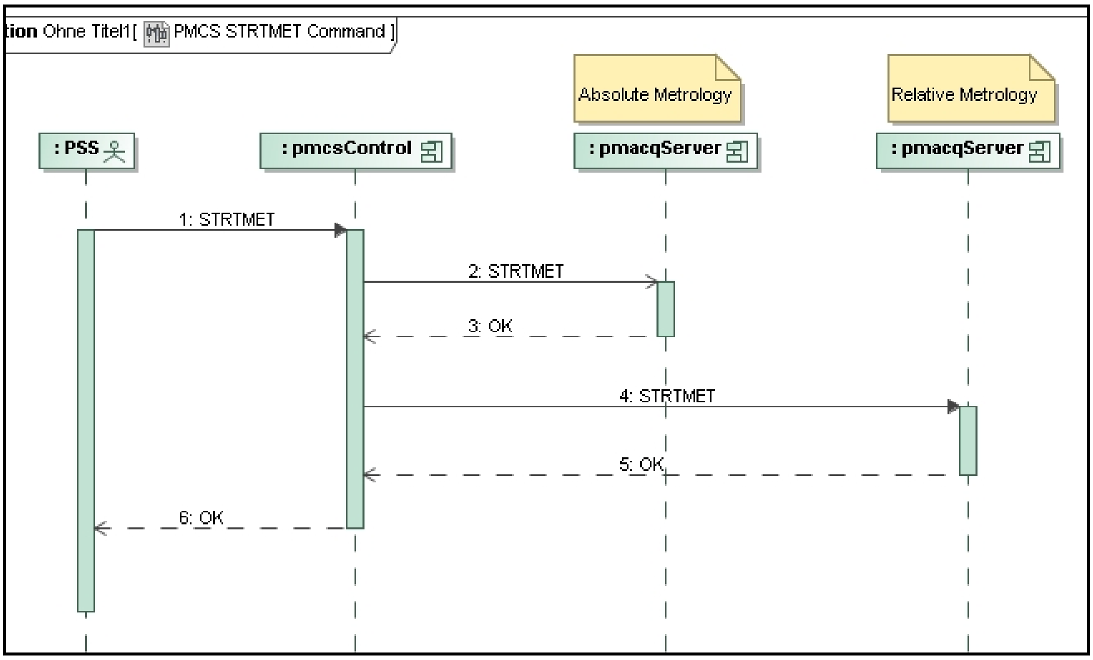

Figure 23: PMCS STRTMET Command

2.5.4 PMCS State Diagram

Since PMCS is implemented using the CCS standard module WSF [AD 18], it inherits the state machine to a large extent from the default WSF state machine. Most commands are only accepted in certain states. Please cross-check with Figure 24: PMCS State Diagram to verify the validity of a certain command within a certain state.

The PMCS state is independent of the LCU states, which are handled by the PRIMA mode-switching module MSW. PMCS shall automatically go to ONLINE state after startup.

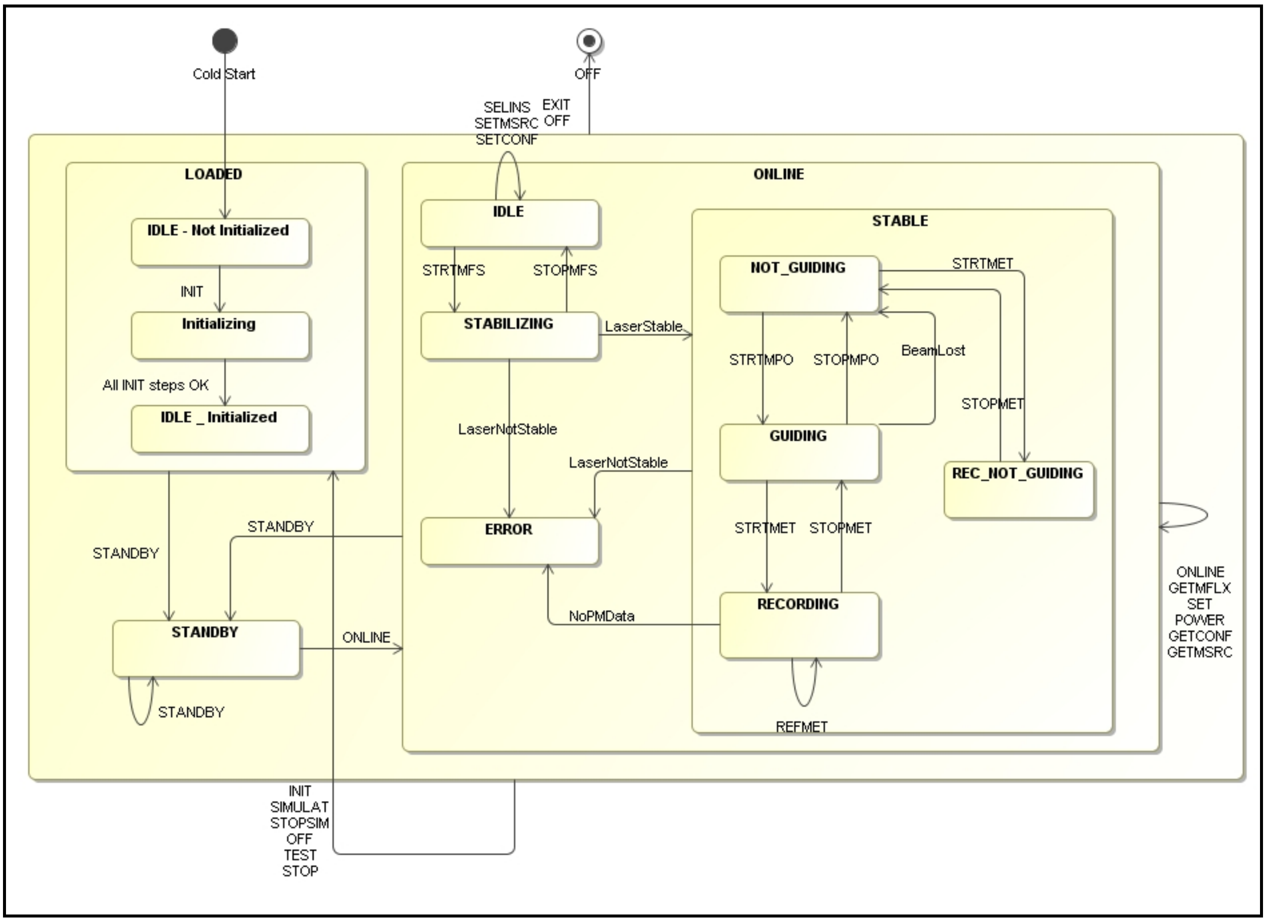

Figure 24: PMCS State Diagram

2.5.5 PMCS OLDB Structure

The structure of the PMCS OLDB is outlined in Figure 25: PMCS OLDB Structure.

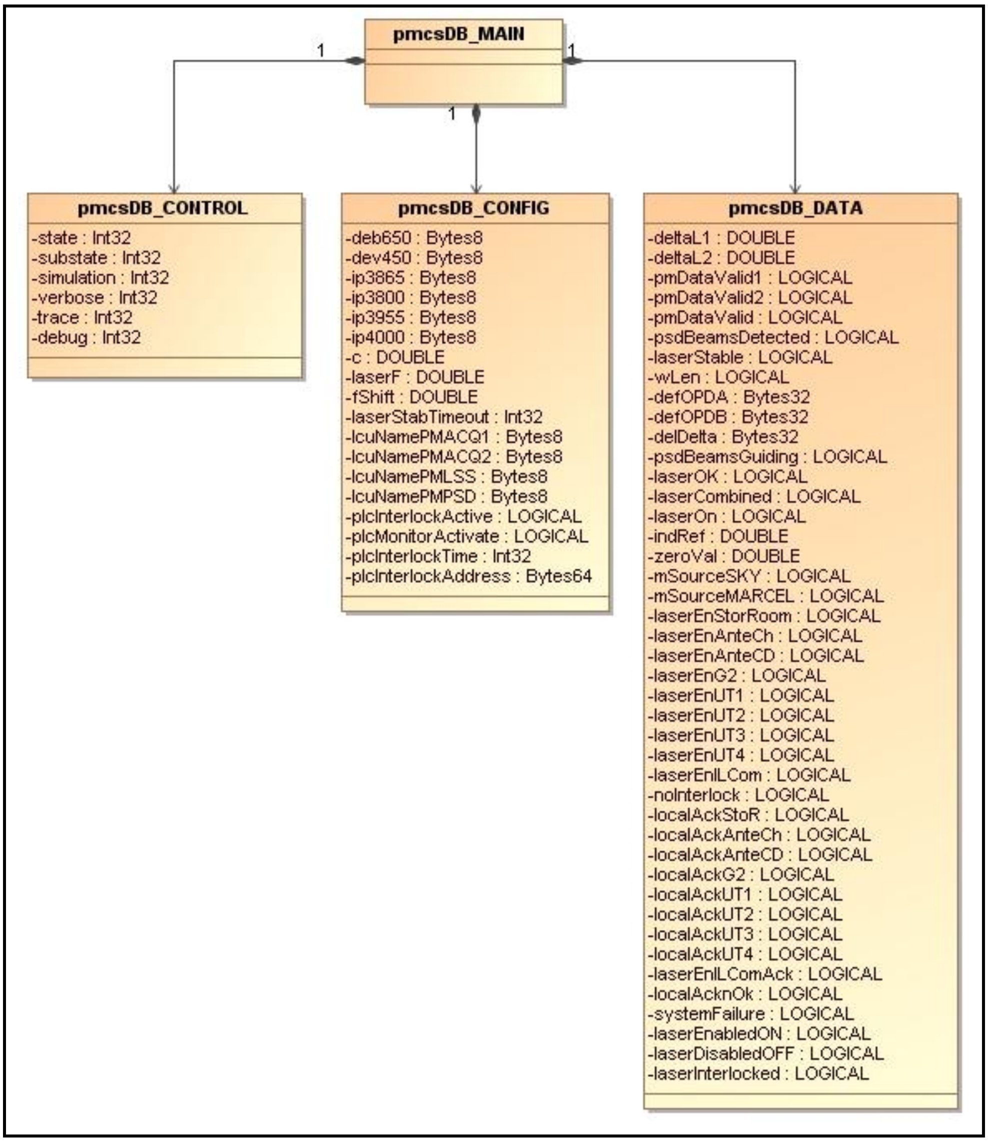

Figure 25: PMCS OLDB Structure

PMCS runtime data is stored relative to the database point “<alias>pmpsd:data”:

Attribute	Data Type	Description and Unit
deltaL1 deltaL2	vltDOUBLE	Result of the DIFFERENTIAL Phase Meter LCU. Result of the SINGLE CHANNEL Phase Meter LCU.
pmDataValid1 pmDataValid2	LOGICAL	Result of the DIFFERENTIAL Phase Meter LCU. Result of the SINGLE CHANNEL Phase Meter LCU.
pmDataValid	LOGICAL	Combined Phase Meter status.
psdBeamsDetected	LOGICAL	Combined PSD status.
laserStable laserOn laserOK laserCombined	LOGICAL	Attributes of the Laser Stabilization LCU. laserCombined = laserOK && laserStable
 indRef zeroVal wLen	 vltDOUBLE vltDOUBLE vltDOUBLE	Attributes added for PAOS: see VLTSW20070382: Index of refraction. Last measured zero position. Calibrated laser wavelength [nm].

defOPDA defOPDB defDelta	vltBYTES32 vltBYTES32 vltBYTES32	PRIMET definition of positive OPD for channel A. PRIMET definition of positive OPD for channel B. Definition of positive DOPD (should always be 'CHA-FSUb').
mSourceSKY mSourceMARCEL	vltLOGICAL	Status of the SETMSTR command: SKY = Flip mirror devices set to IN MARCEL = Flip mirror devices set to OUT
laserEnStorRoom laserEnUT1 laserEnUT2 laserEnUT3 laserEnUT4 laserEnG2 laserEnILCom laserEnILComAck noInterlock localAckStoR localAckAnteCh localAckUT1 localAckUT2 localAckUT3 localAckUT4 localAcknOK localAckAnteCD localAckG2 laserEnAnteCD systemFailure laserEnabledON laserDisabledOFF laserInterlocked	vltLOGICAL	Periodically updated status information from the S7 safety PLC (PRIMET interlock system), read via OPC/UA.
psdBeamsGuiding	vltLOGICAL	TRUE if the pupil guiding loop is active for all 4 quadcells.

Table 12: PMCS Data Attributes

PMCS configuration data is stored relative to the database point “<alias>pmpsd:config”:

Attribute	Data Type	Description and Unit	Default Value
dev650 dev450	vltBYTES8 vltBYTES8	on LCU <lprmls>: Device on 650k channel (FSUA, AMBER, MIDI) on LCU <lprmls>: Device on 450k channel (FSUB fixed)	 FSUB
ip3865 ip3800 ip3955 ip4000	vltBYTES8	PRIMET IP3 input channel connected to 38.65 Mhz PRIMET IP1 input channel connected to 38.00 Mhz PRIMET IP4 input channel connected to 39.55 Mhz PRIMET IP2 input channel connected to 40.00 Mhz	
c	vltDOUBLE	Constant that represents the speed of light [m/sec].	299.792.458
laserF	vltDOUBLE	Constant that represents the frequency of the stabilized laser [Hz].	c/1319
fShift	vltDOUBLE	Constant that represents the frequency shift	78.000.000

		between the two metrology channels [MHz].	
laserStabTimeout	vltINT32	Timeout value (in sec.) for reaching the "laser stable" condition.	300
lcuNamePMACQ1 lcuNamePMACQ2 lcuNamePMLSS lcuNamePMPSD	vltBYTES8	CCS environment names of the PRIMET LCUs.	lprmac lprma2 lprmls lprmpd
plcInterlockActive plcMonitorActivate plcInterlockTime plcInterlockAddress	vltLOGICAL vltLOGICAL vltINT32 vltBYTES32	Activate monitoring of the S7 safety PLC Flag to indicate if monitoring is active Time [sec] for periodic status updates Address of S7 PLC for Siemens softnet library.	1 0 2 h1_0,VFD1,SIMATIC1

Table 13: PMCS Config Attributes

2.5.6 Command Description

Command	REFMET
Parameters	<PMACQ ID> (STRING) <Reference Mode> (STRING) <Timestamp> (STRING)
Reply	OK/ERROR
Description	The REFMET command is used to perform a software referencing of the PRIMA metrology in one of four different ways. ZERO just sets the currection factor to 0, while AVG sets the correction factor to the arithmetic mean of the last <n> measurements, and CURRENT sets it to the current measurement. Moreover, a timestamp can be supplied within <TS>. In that case, the correction factor is set to the measurement taken at that specific UTC timestamp which follows the mode parameter. pmacqID shall be pmacqABS or pmacqREL.

Command	STRTMET
Parameters	None
Reply	OK/ERROR
Description	The STRTMET command is used to start processing data coming from the (two) Phase Meters, and to deliver the quantity Delta L to the RMN network.

Command	STOPMET
Parameters	None
Reply	OK/ERROR
Description	The STOPMET command is used to stop processing data coming from the (two) Phase Meters, and to deliver the quantity Delta L to the RMN network.

Command	STRTMFS
Parameters	None
Reply	OK/ERROR
Description	The STRTMFS command is used to close the light source stabilization control loop.

Command	STOPMFS
Parameters	None
Reply	OK/ERROR
Description	The STRTMFS command is used to open the light source stabilization control loop.

Command	SET
Parameters	<Accuracy> (REAL)
Reply	OK/ERROR
Description	SET is used by the instrument to set the required laser precision for the light source stabilization loop.

Command	POWER
Parameters	<On/Off Flag> (STRING)
Reply	OK/ERROR
Description	The POWER command is used to switch the laser diode ON or OFF.

Command	STRTMPO
Parameters	None
Reply	OK/ERROR
Description	STRTMPO is used to start the pupil optimization loop between PMPSD and the star separators, which is active during the observation. Corrections are sent to the process ID of the process on the star separator LCUs that is in charge of controlling the variable curvature mirrors.

Command	STOPMPO
Parameters	None
Reply	OK/ERROR
Description	STOPMPO is used to stop the pupil optimization loop between PMPSD and the star separators, which is active during the observation.

Command	GETMFLX
Parameters	None
Reply	<X, Y, Q_SUM, IP1> (REAL) <X, Y, Q_SUM, IP2> (REAL) <X, Y, Q_SUM, IP3> (REAL) < X, Y, Q_SUM, IP4> (REAL)
Description	GETMFLX returns the total flux of all 4 active quadcells, as well as the X(rms) and Y(rmx) correction positions.

Command	SELINS
Parameters	<Instrument Name> (STRING)
Reply	OK/ERROR
Description	SELINS is used to verify if the expected instrument is really connected to the PRIMA MET switchyard. This is a manual operation, which however can be verified via proximity switches.

Command	GETCONF
Parameters	None
Reply	<Telescope #1> (STRING) <Telescope #2> (STRING)
Description	GETCONF is used to return the IDs of the two telescopes that are used for the PRIMA observation. The parameters must be either two ATx or UTx.

Command	SETCONF
Parameters	<Telescope #1> (STRING) <Telescope #2> (STRING)
Reply	OK/ERROR

Description	GETCONF is used to set the IDs of the two telescopes that are used for the PRIMA observation. The parameters must be either two ATx or UTx.

Command	SETMSRC
Parameters	<Source> (STRING)
Reply	OK/ERROR
Description	SETMSRC is used to select the source for the current operational mode: MARCEL or SKY. Depending on this command, the flip mirror devices of pmlss are moved IN or OUT of the beam.

Command	GETMSRC
Parameters	None
Reply	<Source> (STRING)
Description	GETMSRC is used to retrieve the source for the current operational mode: MARCEL or SKY. Depending on this command, the flip mirror devices of pmlss are moved IN or OUT of the beam.

Command	RESIL
Parameters	None
Reply	OK/ERROR
Description	Remote RESET of the PRIMA interlock system.

2.5.7 AUTREP Database

The PRIMA WS performs data logging for a number of FITS keywords. The PRIMA logs are available via the AUTREP database, http://autrep.pl.eso.org.

Two dictionaries are used on the PRIMA WS:
- dicTCS 1.56
- dicISS 1.48.1.16 (this branch adds a number of PRIMA specific keywords)

2.5.8 Test and Simulation Support

The following files provide TAT tests for `pmcs`:

- `pmcsTestStdCommands.tcl` for testing the standard commands (TCL/TK script file).
- `pmcsTestCommands.tcl` for testing all PMCS specific commands (TCL/TK script file).

Simulation support is enabled via the SIMULAT command. In this case, evhDummy servers must be started on the PRIMA WS. The following evhDummy configuration files are available and automatically started within the PMCS TAT tests:

- pmacqServerDummy.dat: PMACQ command server dummy.
- pmpsdServerDummy.dat: PMPSD command server dummy.
- pmlssServerDummy.dat: PMLSS command server dummy.

___oOo___