Gemini

Primary Mirror Control System -

Preliminary Package Software Design Description

SPE-C-G0064

Issue: 1.1

John Maclean

05 January 1996

Royal Greenwich Observatory
Madingley Road
Cambridge
CB3 OEZ

Contents

1. INTRODUCTION....4
1.1 PURPOSE....4
1.2 SCOPE....4
1.3 DEFINITIONS, ACRONYMS AND ABBREVIATIONS....4
1.3.1 Co-ordinate system description....6
1.4 OVERVIEW....6
1.5 SOFTWARE DESIGN METHODOLOGY....7
1.6 DOCUMENT HISTORY....8
2. REFERENCES....9
3. DECOMPOSITION AND DEPENDENCY DESCRIPTION....10
3.1 INTRODUCTION....10
3.2 SYSTEM PERSPECTIVE....10
3.3 CONTEXT DIAGRAM....11
3.3.1 External Entities....12
3.3.2 External Data Flows....13
3.4 MODULE DECOMPOSITION AND DEPENDENCY DESCRIPTION....14
3.4.1 Decomposition....14
3.4.2 Level 0 data flow....15
3.4.3 PCS Executive....17
3.4.4 Thermal management....18
3.4.5 Cell Sensing....19
3.4.6 Air Pressure Support....19
3.4.7 Passive Support....20
3.4.8 Active Optics....21
3.4.9 Primary Mirror Engineering Screens....21
3.4.10 I/O Subsystem....22
4. DETAIL DESCRIPTION....23
4.1.1 PCS Executive....23
4.1.2 Thermal management....23
4.1.3 Cell Sensing....26
4.1.4 Air Pressure Support....26
4.1.5 Passive Support....29
4.1.6 Active Optics....36

4.1.7 Primary Mirror Engineering Screens 43
4.1.8 I/O Subsystem 43

5. INTERFACE DESCRIPTION....... 49

5.1 INTRODUCTION 49
5.2 PCS EXECUTIVE 49
5.2.1 External interfaces 49
5.3 THERMAL MANAGEMENT 51
5.3.1 External interfaces 51
5.4 CELL SENSING 53
5.4.1 I/O sub-system interface 53
5.5 AIR PRESSURE SUPPORT 53
5.5.1 I/O sub-system interface 53
5.6 PASSIVE SUPPORT SYSTEM 54
5.6.1 I/O sub-system interface 54
5.7 ACTIVE OPTICS 54
5.7.1 I/O sub-system interface 54
5.8 PRIMARY MIRROR ENGINEERING SCREENS....... 55
5.8.1 External interfaces 55
5.9 I/O SUBSYSTEM 58
5.9.1 External interfaces 58
5.10 INTERFACE CONTROL DOCUMENTS 67

6. APPENDICES 68

6.1 APPENDIX A - SELECTION OF VME/CAN INTERFACE....... 68
6.1.1 Introduction 68
6.1.2 Option 1 68
6.1.3 Option 2. 69
6.1.4 Option 3 70
6.1.5 Option 4 70
6.1.6 Summary 71
6.2 APPENDIX B - SELECTION OF NODE CONTROL UNIT....... 72
6.2.1 Selection of Node CPU 72
6.2.2 Requirements 72
6.2.3 Off-the-shelf Solutions 73
6.2.4 Custom solution 73
6.2.5 Summary of Selection Process....... 75
6.2.6 Development 75

6.3 PROTOTYPE EPICS I/O SUB-SYSTEM..75

1. Introduction

1.1 PURPOSE

This document is a deliverable of the preliminary design phase of the Gemini primary mirror control system work package. Its purpose is to formally describe the proposed design of the hardware and software that implement the primary mirror control system functions. This document incorporates the *Control System Design Description*. Note that this is an incomplete, preliminary, version of the Software Design Description as called for at PDR by the work scope agreement for this package.

The audience of this document is:

- The PDR reviewing panel.
- Individuals working on subsequent phases of the project.
- Individuals working on connected Gemini work packages

1.2 SCOPE

The software product to be developed from this document will be called the *Primary Mirror Control System* or PCS. The software will consist of a number of separate processes residing on a single VME crate running the VxWorks operating system.

The scope of the PCS has been described in detail in the “PCS Package Requirements Specification”[1].

1.3 DEFINITIONS, ACRONYMS AND ABBREVIATIONS

The following is a list of abbreviations used in this and related PCS documents.

ADC	Analogue to Digital Converter
AO	Active Optics
APSS	Air Pressure Support System
CAN	Controller Area Network
CDR	Critical Design Review
DAC	Digital to Analogue Converter
DHS	Data Handling System
ECS	Enclosure Control System
EPICS	Experimental Physics Instrument Control System
GIS	Gemini Interlock System
GUI	Graphical User Interface

HRWFS	High Resolution Wave Front Sensor
I/O	Input/Output
ICD	Interface Control Document
ICS	Instrument Control System
IOC	Input Output Controller
LAN	Local Area Network
LUT	Look Up Table
M1	Primary mirror
M2	Secondary mirror
NCB	Node Control Box
NCU	Node Control Unit
OCS	Observatory control system
OPI	Operator Interface
PCS	Primary mirror Control System
PDR	Preliminary Design Review
PRS	Package Requirement Specification
PSS	Passive Support System
PTP	Package Test Procedure
PWFS	Peripheral Wave Front Sensor
RGO	Royal Greenwich Observatory
RO	Royal Observatories
Rx	Rotation of M1 about x axis (see 1.3.1 Co-ordinate system description)
Ry	Rotation of M1 about y axis (see 1.3.1 Co-ordinate system description)
Rz	Rotation of M1 about z axis (see 1.3.1 Co-ordinate system description)
SC	Standard Controller (previously known as, SIC Standard Instrument Controller)
SDD	Software Design Description
SDR	System Design Review
TAI	International Atomic Time
TBD	To Be Determined
TCS	Telescope Control System
TMS	Thermal Management System
Tx	Translation of M1 in x (see 1.3.1 Co-ordinate system description)
Ty	Translation of M1 in y (see 1.3.1 Co-ordinate system description)
Tz	Translation of M1 in z (see 1.3.1 Co-ordinate system description)
WP	Work Package

1.3.1 Co-ordinate system description

For a full description of the co-ordinate system see [2] (Interface Control Document ICD-G-0008). This document describes the co-ordinate system thus

Azimuth angle A:

With the telescope pointing north azimuth angle, A = zero degrees. A is defined positive with clockwise rotation. The telescope will point east when A = 90° and west when A = -90° or A = 270°.

Altitude or elevation angle, E:

The elevation angle E is measured from the horizon (E=0°) and is positive toward the zenith (E=90°). It is the complement of the zenith angle.

"Park" position of the telescope:

When not in use, the telescope will be "parked" facing East (A = 90°). The telescope will be in the zenith pointing position (E=90°).

The x, y, and z, axis for the primary mirror *are shown below*

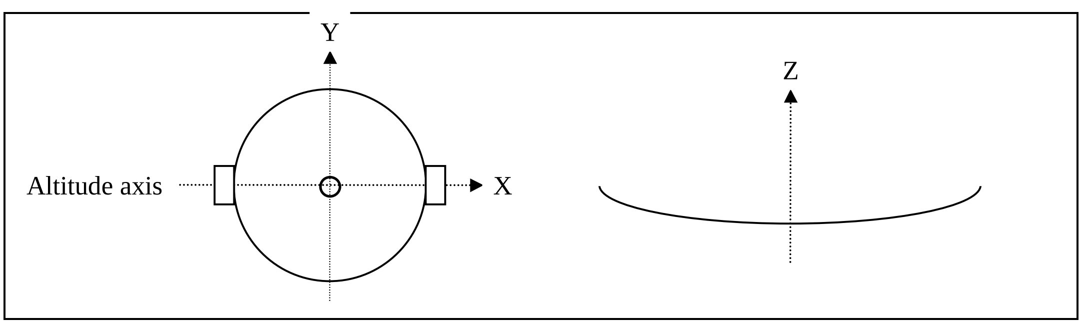

Figure 1-1 Primary mirror co-ordinate system

1.4 OVERVIEW

The format of this document is based upon IEEE Std 1016-1987 "IEEE Recommended Practice for Software Design Descriptions" [3].

The design is broken up into a number of subsystems or *entities* and each has a set of attributes. The values given to these attributes form the design of the control system. The following attributes have been identified as relevant to a control system design:

- Name
- Purpose - Or Requirements.

- Subordinates - Can the subsystem be broken down into still smaller subsystems.
- Internal Dependencies - Relationships with other subsystems.
- Internal Interfaces - How the subsystem communicates with other subsystems.
- External Dependencies - Or Resources. External systems that are required by the subsystem.
- External Interfaces.
- Function - Or Design Detail. What the system does and how it does it.

Each subsequent section of this document presents a design view of the PCS control system. Each design view defines of a subset of the subsystem attributes.

Section 3 presents the Decomposition Description design view. This outlines how the PCS is organised internally and how it fits in with the rest of the Gemini Control System. The Name, Subordinates and Internal/External Dependency attributes are defined in the Decomposition Description.

Section 4 presents the Detail Description design view. For each subsystem, the requirements, as specified in the PCS PRS [1], are given. The control scheme for each entity is described where appropriate. *Detailed designs are not presented at this stage.*

Section 5 presents the Interface Description design view. This presents the hardware and software Internal and External Interfaces.

The appendices describe the results of trade studies and prototype work.

1.5 Software Design Methodology

The software design methodology follows that of Yourdon and De Marco with the real-time extensions of Ward and Mellor, [4]. The design presented here uses data flow diagrams which consist of the following components:

Circular Bubbles: These represent processes in the system which handle data.

Arrowed lines: These represent data flows within the system. Data flows in the direction of the arrow.

Rectangular boxes: These represent “terminators”, or things in the outside world which provide or receive data.

Parallel bars: These represent data stores, or places in the system where data may be stored and retrieved at a later time.

Dotted lines: These represent control flows in the system

Data items are described using “Backus Naur Form” or BNF. This is a method of defining the relationship between data objects using a series of short-hand symbols to represent combination, choices and repetition. The symbols are:

=	“consists of”

+	"and" (sequence)
[|]	"or" (selection)
{ }	"repeating" (iteration)
()	"optional components"
@	"reference to (key of) another data object"
;	"end of sentence"
<...>	"Instance of"

Table 1-1 BNF Notation

1.6 DOCUMENT HISTORY

ISSUE	DATE	COMMENTS
0.0	01 November 1995	1st Draft, incomplete
0.1	21 November 1995	Draft, incomplete
0.2	29 November 1995	Draft, incomplete
0.3	1 December	Late draft, for comments

Table 1-2 Document history

2. References

[1] Primary Control System Design review Report, REV-C-G0035, March 10, 1995

[2] Interface Control Document ICD-G-0008

[3] IEEE Std 830-1984 “IEEE Guide to Software Requirements Specifications”.

[4] “Structured Development for Real-Time Systems”, Ward, P.T. & Mellor, S.J., Yourdon Press, 1985.

[5] ICD 9 -EPICS Time Bus Driver A.N.Johnson

[6] *PCS IOC/NCU CAN Protocol, . J Maclean*

[7] *TCS Interface Design Description 4 - The TCS/PCS interface*

[8] *M1 Support node box and CAN cable assignment*, 15 September 1995, Eugene Huang.

[9] *Description of Active Control algorithms*, 28 March 1995, Larry Stepp.

[10] *DeviceNet Specification release 1.2,* Allen Bradley.

[11] *devCAN- CAN Bus Device Support,* Issue 1, A.N.Johnson, 23 November 1995.

[12] *drvIpac - Industry Pack Driver*, Issue 1, A.N.Johnson, 19 September 1995.

[13] *drvTip810 - CAN Bus Driver*, Issue 1, A.N.Johnson, 23 November 1995.

[14]Peregrine McGehee “Gemini Software Design Description”, chapter 16 - “Details of the M1 Control System”.

[15]Gemini M1 support system Design Requirements document, Work Scope No 47990-GEM01015.

[16]ICD 7b - TCS Subsystem Interfaces. Chris Mayer, Steve Wampler and Steven Beard

[17]ICD 12 - Interlock System. Peregrine M. McGehee

[18]ICD 13 - Standard Controller. A. N. Johnson

3. Decomposition and Dependency Description

3.1 Introduction

This section describes, how the PCS fits into the overall Gemini control system, the division of the PCS software into separate design entities and the way the design has been structured.

3.2 System Perspective

The software control of the Gemini telescopes is split into a hierarchical arrangement of distinct systems. There are 4 principal systems, these are the OCS, TCS, ICS and DHS. Beneath the TCS are the PCS, MCS, SMCS, ECS and CRCS. The PCS software system provides the interface between the TCS and the primary mirror support and thermal management hardware. The relationship between the PCS and other components of the Gemini Control System is shown in Figure 3-1.

During an observing run, the PCS will receive commands from the TCS. However, for calibration and diagnostic purposes, an engineering screen system will also be provided so that commands can be sent directly to the PCS without going through the TCS. Thus it will be possible to use the PCS as a standalone system. It is not possible for the PCS to issue a command to a higher level system such as the TCS.

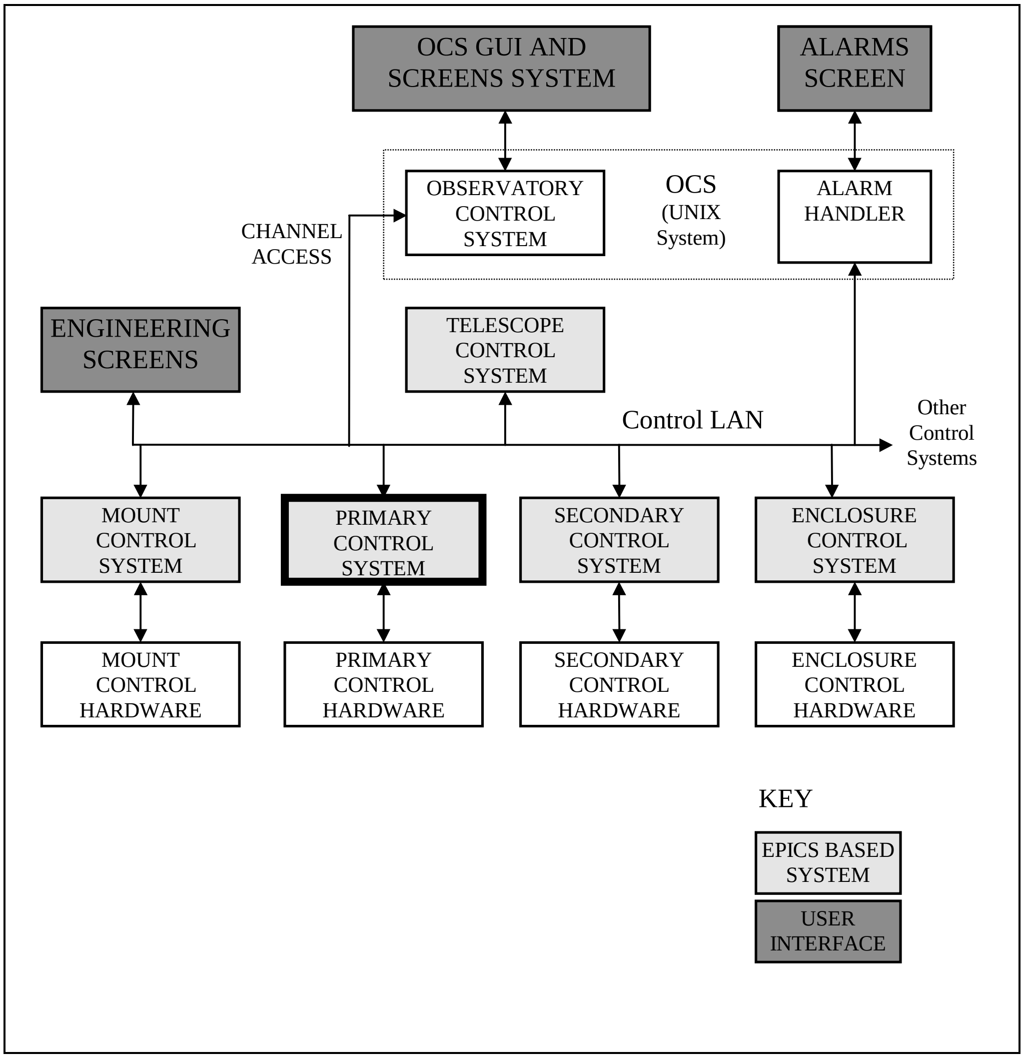

Figure 3-1 Relative position of PCS in Gemini Control system

3.3 Context Diagram

The context diagram shows as rectangles the external “terminator” entities which provide date to or receive data from the PCS.

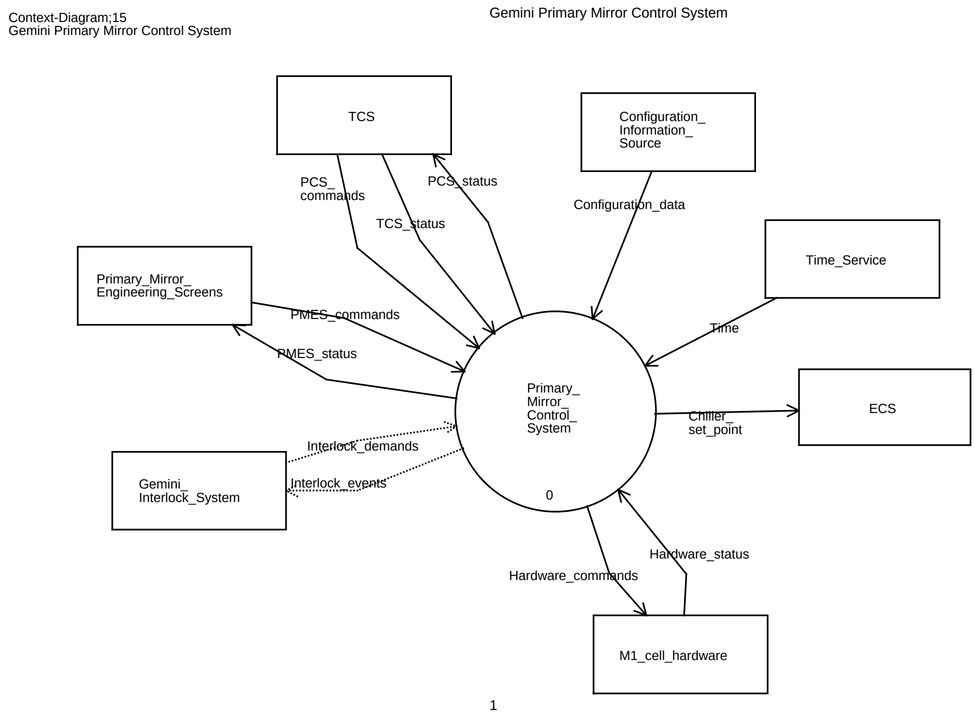

Figure 3-2

PCS Context Diagram

3.3.1 External Entities

TCS: Telescope Control System. This is the Gemini Principle System responsible for controlling the Telescope and all of its sub-systems, of which the PCS is one.

Configuration Information Source: This is considered logically as a single source that holds all of the files required for configuration of the PCS.

Time Service: This provides time via the time bus for the PCS. It is described in “ICD-9” [5].

Gemini Interlock System: The system responsible for the safety of all Gemini personnel and equipment, which connects all the Gemini software and hardware systems.

Primary Mirror Engineering Screens: A user interface for engineering staff to the PCS. May be used as a diagnostic aid during normal operation or as a stand-alone tool during commissioning.

ECS: Enclosure control system. Controls Thermal Management chiller unit.

M1 Cell Hardware: This describes all the M1 cell hardware controlled or monitored by the PCS. This includes the following sub-systems;

- **Passive Support System:** This is comprised of the hydraulic axial and lateral support systems and position sensors. It is responsible for M1 definition and partial support.

- **Air Pressure Support System:** This is the pressurised volume of air or 'air bag' under the mirror that complements the axial hydraulic support system.
- **Thermal management System:** This is comprised of the radiator panels, their heaters and temperature sensors located behind the mirror.
- **Cell Sensing System:** A number of temperature sensors, strain gauges and accelerometers used to provide diagnostic information on the conditions within the M1 cell structure.
- **Active Optics System:** The pneumatic axial support units.

3.3.2 External Data Flows

Interlock events: Signals to GIS that something in PCS has generated an interlock

Interlock demands: Signal from GIS that something in PCS should be interlocked

PMES commands: Engineering level commands for calibration or diagnosis

PMES status: Engineering level status for calibration or diagnosis

PCS Commands: The commands issued to the PCS from the TCS. This flow is described in TCS IDD4 [7].

TCS Status: Certain items of telescope status held by the TCS are required by the PCS. In practice the TCS does not have to send the information. The PCS can monitor the information using the facilities of EPICS channel access.

PCS Status: Status information provided by the PCS to the TCS describing the current state of the PCS including any alarm conditions. In practice the PCS does not have to send the information. The TCS can monitor the information using the facilities of EPICS channel access.

Configuration Data: This represents all the miscellaneous data required to configure the PCS at run time and if necessary update it during operation. This information includes.

- VxWorks boot files
- EPICS database configuration files
- Look up tables for control algorithms

Time: TAI provided via Gemini time bus

Chiller set point: Demanded coolant temperature for TMS.

Hardware Commands and **Hardware Status:** Commands and sensor data sent and received from the M1 cell hardware, this includes the following;

- **Target active forces:** Desired forces for pneumatic actuators.
- **AO sensor data:** Readings from 'Pneumatic' load cells and other diagnostic data from Node Control Units.

- **Definition sensor data:** Position of M1 with respect to M1 cell., status of passive (hydraulic) support system and readings from ‘Hydraulic’ load cells.
- **Target M1 definition:** Desired position of M1 in Tx, Ty, Tz, Rx, Ry and Rz with respect to mirror cell.
- **Air pressure data:** Readings from APSS sensors
- **Target air pressure:** Desired air pressure of APSS
- **Target Temperature:** Desired temperature of radiator panels
- **M1 thermal status:** Readings from Thermal management system sensors
- **Cell environment status:** Readings from cell sensing system sensors

3.4 Module Decomposition and Dependency Description

3.4.1 Decomposition

It is necessary to decompose the PCS software into a number of separate entities. The structure of this decomposition is shown in Figure 3-3. At the lowest level of the diagram is the *I/O subsystem*. This provides an interface for the higher level software to the hardware in the M1 cell .

On top of the I/O layer reside the main functional sub-systems of the PCS, the *thermal management, cell sensing, APSS, Passive support,* and *active optics* sub-systems. The objective here is to ensure that there is a low level of coupling between these modules. Each of these modules is responsible for controlling its area of responsibility via the I/O sub-system as directed by the *PCS Executive*.

The PCS executive layer is responsible for supervising the functioning of the PCS. It handles the command interface with the TCS and supervises configuration of the PCS.

The Primary Mirror engineering screens system provides access to the PCS for engineering functions such as fault diagnosis and calibration procedures. It can potentially provide access to all entities and levels of PCS software via the EPICS database.

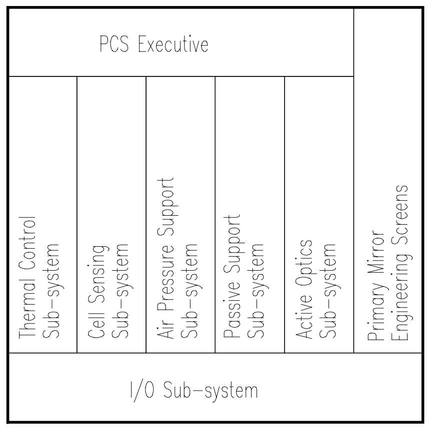

Figure 3-3 PCS Software Structure

3.4.2 Level 0 data flow

The level 0 data flow diagram illustrates the inter-module dependencies and coupling.

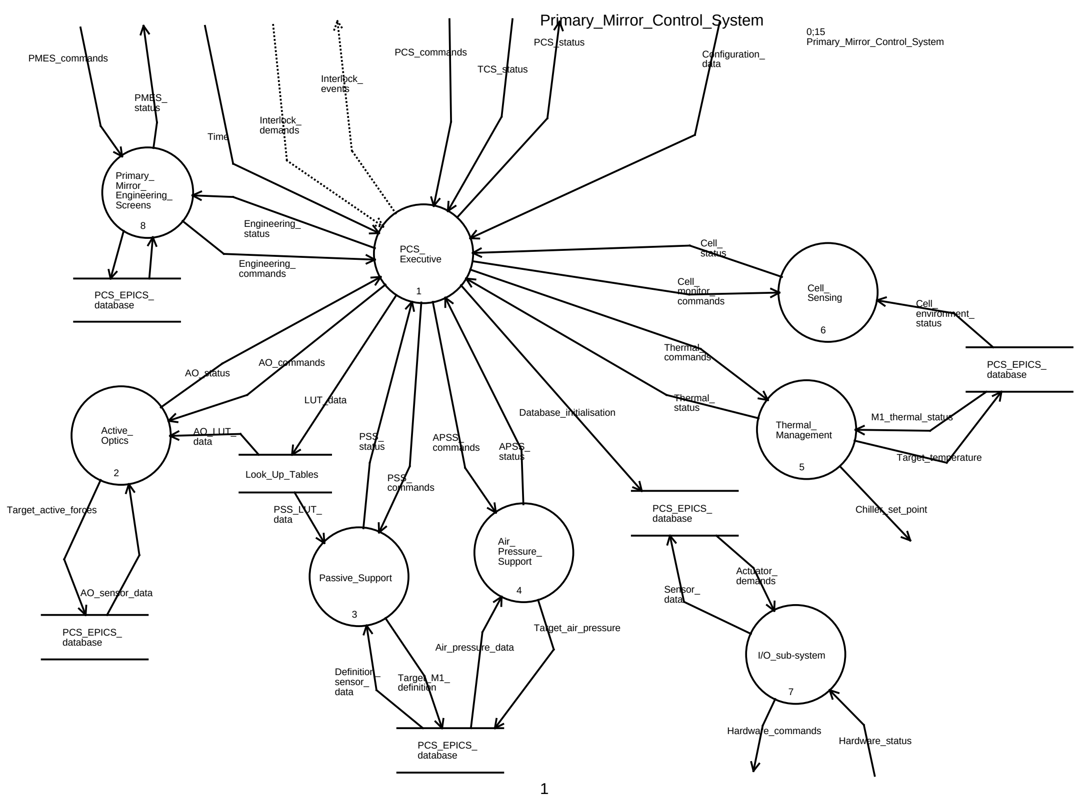

Figure 3-4 Level 0 data flow

3.4.2.1 External Data Flows

The data flows going off of the diagram are the same as those going to and from the terminators on the context diagram.

3.4.2.2 PCS data stores

PCS EPICS Database: This is a collection of EPICS records that are used to represent the state of all I/O signals of the PCS. Data is available in engineering or raw units. The alarm state of a record indicates the validity of the data and its status regarding limit levels. At this level it is not intended to represent the linking of records to process this data. Data processing is conceptually handled in the process bubbles although in reality some processing will be implemented via linking of EPICS records.

Look Up Tables: These are a set of Look Up Tables that are used by PCS sub-systems to implement control algorithms. The contents of the LUTs are updated from the configuration source at the instigation of the PCS executive. This would happen at boot up or when commanded to do so by the TCS.

3.4.2.3 PCS Internal Data Flows

AO Status: Status of active optics system, e.g. in position, not in position, healthy, not healthy.

AO Commands: Command information for the active optics system comprising; Zernike parameters, Telescope Zenith angle and ambient temperature.

AO LUT data: Look up tables for AO system control algorithms.

Target Active Forces: Demanded force output of the 120 pneumatic actuators.

AO Sensor data: Status from AO sensors including; The 120 ‘Pneumatic’ load cell readings plus diagnostic information from axial NCUs.

LUT data: Look up tables contents for PCS sub-systems.

PSS status: Status of Passive Support System. , e.g. in position, not in position, healthy, not healthy.

PSS commands: Command information for the Passive Support System comprising; 3/6 zone operation, demanded axial definition, demanded lateral definition, Telescope Zenith angle and ambient temperature.

PSS LUT data: Look up tables for PSS control algorithms.

Definition Sensor data: Status from passive support sensors comprising; the 120 axial ‘hydraulic’ load cell readings, the 60 lateral load cell readings, axial and lateral displacement sensor readings.

Target M1 Definition: Commands to PSS actuators including; Master cylinder positions, control valve positions.

APSS commands: Commands to the Air Pressure support system including move and emergency stop with parameters Telescope Zenith angle, sum of hydraulic load cells and ambient temperature.

APSS status: Status of Air Pressure Support System. , e.g. in position, not in position, healthy, not healthy.

Air pressure data: Readings from air pressure sensors of APSS plus diagnostic information from their NCUs.

Target air pressure: Demanded output pressure from APSS pneumatic control valves.

Database initialisation: Start up values of Process variables.

Sensor data: This flow represents all input data received from the I/O system.

Actuator demands: This flow represents all output data from the I/O system to hardware actuators.

Thermal commands: Commands to the thermal management system including move and emergency stop with parameters; chiller water temperature, ambient temperature desired mirror temperature.

Thermal status: Status of thermal management System, e.g. in position, not in position, healthy, not healthy. With parameters; average M1 bulk temperature, ambient air temperature

M1 thermal status: Status from the thermal management system temperature sensors including; M1 bulk temperature, ambient air temperature, radiator panel temperature.

Target temperature: Set points for the 24 Thermal management system radiator panel heaters.

Cell environment status: This flow contains environmental data from the cell including average cell temperature, strain gauge readings and accelerometer readings.

Cell status: Data from the accelerometers, cell thermal sensors and strain gauges.

Cell monitor commands: Commands to the cell sensing system.

Engineering commands: Engineering system level commands.

Engineering status: System level engineering status.

3.4.2.4 PCS Processes

The PCS entities shown in Figure 3-3 are also shown as processes in the level 0 DFD. They are described in greater detail in the following sections.

3.4.3 PCS Executive

3.4.3.1 Purpose

The PCS Executive is responsible for supervising the functioning of the PCS.

3.4.3.2 Function

It processes commands from either the TCS or the engineering system, checking that they are valid and are not disallowed by the current system state.

Any sequencing of operations required by high level commands will be performed here.

Any selection of data sources will be performed here. For example if the TCS becomes unavailable the executive layer will decide where to get elevation information from.

3.4.3.3 Subordinates

TBD

3.4.4 Thermal management

3.4.4.1 Purpose

In order to cool and heat the primary mirror a radiation plate system will be used. This system is active during the day as well as during the evening and consists of a cooled or heated water-glycol mixture circulation system. Heat will be added or removed from the coolant by a modular chiller which will be mounted in the plant room, remote from the telescope.

The thermal management system will monitor M1 temperatures and control the temperature of individual radiation plates via heaters which heat the coolant flowing into individual panels.

3.4.4.2 Function

The radiating panel system will consist of 24 radiating panels under the mirror cooled or heated by a circulating liquid coolant.

A network of thermal sensors are placed in at 42 locations on and around the primary mirror and cell structure and are monitored via the CAN I/O system. These sensors are to be monitored at a nominal* 1 Hz rate and are used to indicate the current temperature distribution on the upper and lower surfaces of the primary mirror, in the air around the primary mirror and of the radiator panels.

*This type of slow moving quantity and non-critical readings may read at a slower rate or on an event basis if it is necessary to reduce the load on the IOC CPU. Reading on an event basis would entail the NCU only sending a message on a change of status. For example the NCU would only send a temperature reading to the VME when the temperature changed by 0.1°C from the previously reported value.

3.4.4.3 Subordinates

Calc radiation plate temperature

Assemble TMS status

Calculate chiller temperature

Process TMS sensor readings

3.4.5 Cell Sensing

3.4.5.1 Purpose

The M1 cell sensing system consists of sensors and associated electronics for monitoring conditions in the M1 cell assembly. The data gathered by these sensors shall be available for engineering purposes.

3.4.5.2 Function

The sensors to be monitored are 3 accelerometers, 16 temperature transducers and 32 strain gauges.

3.4.5.3 Subordinates

TBD

3.4.6 Air Pressure Support

3.4.6.1 Purpose

The passive support system is augmented by the use of an Air Pressure Support System (APSS). The APSS bears 80% of the weight of the primary mirror cell at zenith and leaves the loading on the passive support system constant as the telescope tilts in zenith angle. The passive support system, the load cells, and the primary mirror must all survive failure of the APSS.

3.4.6.2 Function

The air pressure support system employs the 'air bag' principle but is a fundamental departure from conventional air bag systems in that there is no physical bag. In this instance the 'bag' is formed by the rear face of the mirror and the upper face of the cell, with the gap between the two at the inner and outer diameters of the mirror being closed using inflatable sealing devices. These seals are also used to actively compensate the gravitational forces on the mirror peripheries.

The air pressures in the inner and outer seals will be controlled by the PCS via proportional pneumatic valves.

3.4.6.3 Subordinates

Calculate from load

Calculate from Zenith angle

APSS control

Generate APSS status

3.4.7 Passive Support

3.4.7.1 Purpose

This system is also known as the *Support and Definition System* and may be referred to elsewhere by this title.

The primary mirror is mounted in a cell and faces upwards towards the zenith. This cell provides for axial and lateral support of the primary mirror in a changing gravity vector as the telescope tracks an object. The TCS will issue commands to both translate and tip the optical axis of the primary mirror. Preliminary coarse alignment of the primary mirror shall be done at installation.

This passive support system shall be capable of aligning the primary mirror along the six axes: x, y, z, tip/tilt, and rotation about the optical axis to maintain the optical alignment to the required accuracy. X and Y axis alignment and rotation about the optical axis shall be provided by the passive lateral supports. Rotation about the optical axis will be achieved by pushing on one set of lateral supports whilst pulling on the other. Z displacement, tip and tilt alignment shall be provided by the passive axial supports.

3.4.7.2 Function

The M1 Support and Definition System consists of the following five parts:

- Axial support
- Axial defining system for Tz, Rx, and Ry.
- Lateral support
- Lateral defining system for Ty, Rz.
- Lateral defining system for Tx.

The axial system is divided into either 3 or 6 zones depending on the amount of wind buffeting which must be resisted. This division into zones is controlled by valves which subdivide the hydraulic actuators The overall height of a zone is controlled by a hydraulic actuator, referred to as a Master Cylinder Unit, which can add or subtract small amounts of fluid from the zone.

Lateral control is provided by similar master cylinder units. Y-definition is divided into 2 zones. X-definition is provided by one zone

3.4.7.3 Subordinates

Axial support

Lateral support

3.4.8 Active Optics

3.4.8.1 Purpose

The figure of the primary mirror will be adjusted by the active optics support system which shall be capable of maintaining the figure of the primary mirror to the required accuracy by correcting for deformations in the mirror figure caused by changing gravity vector as well as dynamic effects due to thermal expansion stresses and wind loading.

3.4.8.2 Function

The active system is implemented as actuators consisting of a pneumatic actuator with a load cell mounted on top. An actuator is astatic and adds or subtracts a force from the passive hydraulic support under which it is mounted. The air pressure in the active actuator is controlled by a pressure regulator that maintains constant pressure in response to a demanded pressure command. The force applied to the back of the mirror by the pneumatic actuator is commanded by a demand signal originating from the PCS and is maintained by a control loop around the load cell and pressure regulator.

In operation there will be an open loop model, developed during commissioning, which continually generates a vector of forces to apply to the mirror and a closed loop model which uses WFS information to generate corrections to the forces.

3.4.8.3 Subordinates

Control AO

Generate AO Status

3.4.9 Primary Mirror Engineering Screens

3.4.9.1 Purpose

The primary mirror support system shall be capable of being tested and calibrated when configured as a stand-alone system separate from the other observatory systems.

3.4.9.2 Function

The Primary Mirror Engineering Screens (PMES) allow engineering access to all aspects of the PCS at either high or low level. It will be implemented as a hierarchical screen system. Each screen will provide a graphical user interface.

3.4.9.3 Subordinates

TBD

3.4.10 I/O Subsystem

3.4.10.1 Purpose

The purpose of the I/O sub-system is to allow the higher levels of PCS software to communicate with the large amount of I/O hardware in the mirror cell. It allows commands to be passed from the PCS to actuators within the M1 cell and data to be passed from sensors within the M1 cell to the PCS.

The I/O subsystem also informs the higher level software of the integrity of the data received or transmitted.

3.4.10.2 Function

The I/O sub-system is a distributed system. The I/O associated with a particular actuator or geographical area is brought together at a Node Control Unit (NCU, also known as a Node control Box, NCB). There are 157 NCUs in the primary control system. These are connected to the PCS VME rack via 4 CAN buses as shown in Figure 3-5. Each CAN bus has approximately 40 nodes connected to it.

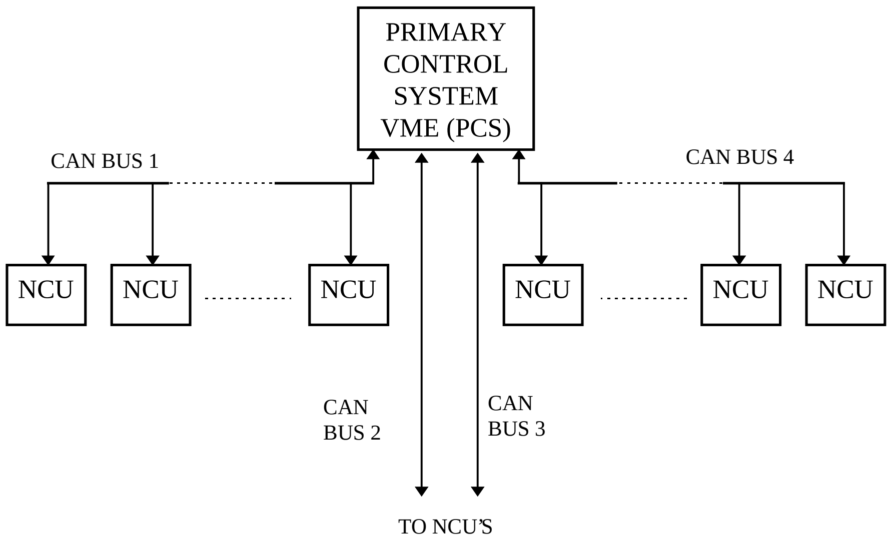

Figure 3-5 I/O sub-system block diagram

3.4.10.3 Subordinates

EPICS/CAN software interface

NCU software

VME/CAN interface hardware

NCU control board

4. Detail Description

4.1.1 PCS Executive

It is intended that the PCS command interface will be similar to that developed for the MCS and will borrow heavily from it. The command interface will be implemented using epics CAD, CAR and SIR records.

4.1.2 Thermal management

4.1.2.1 Introduction

Note: A user requirements document for the thermal management system will be drawn up by the IGPO that will clarify the way in which the TMS will be used.

This section sets out the proposed method for thermal control of M1. Note that at present, there is no provision in this work package to implement control of surface heating.

The prediction of the temperature at the beginning of the next night will be done by a separate client program running on a Unix host. The TCS work package will provide a simple prototype for this that will do the following

a) fetch from the weather server the lowest temperature from the previous night

b) send the following command

M1:bulktemp <temperature> <target time>

<temperature> is the lowest temperature from the previous night as provided by the weather server

<target time> is the time at which this temperature is required

N.B. If the weather server is not available the TCS will prompt the user for a value.

The advantages of doing this operation in a client program are that the prediction is available at any time without requiring an IOC to be started. With the initial simple prediction this is hardly relevant but if more sophisticated predictions become available that require complex models based on historical and current data then it will be much more convenient to run these on a UNIX host. Note also that the initial prototype provided by the TCS will only send the above commands once each time the client is executed.

Whilst the TCS is running it will monitor the bulk mirror temperature provided by the PCS as well as the dew point provided by the weather server. If these are less than n (TBD) degrees different then the TCS will send the command

M1:surface ON <dew_point + n>

where <dew_point + n> is the desired set point of the surface.

When the temperature difference becomes greater than n degrees or the TCS is shut down it will send the command

M1:surface OFF

Alternatively, the same functionality could be provided by the PCS itself if it connected directly to the weather server. The main advantage of this would be protection against dewing at all times not just when the TCS is running

N.B. This proposal presupposes the existence of a weather server to provide the dew point measurement and that the surface heating work package is implemented in the same IOC as the PCS. The command M1:surface is not provided by the PCS

Whilst the TCS is running at night, it will periodically send the following command to the PCS

M1:bulktemp <external temp> <current time>

On receipt of an M1:bulktemp command the PCS will execute an algorithm that given;

- target M1 temperature
- the time at which the target temperature is required
- the current mirror temperature
- the current air temperature (from the PCS’s own sensors)

will compute:

- the required chiller set point
- the required radiation panel heater values

This algorithm will execute within the PCS but will be provided by the Gemini Optics Group. The chiller set point will be written into the PCS's own database. The ECS will be required to monitor the chiller set point computed by the PCS and set the chiller to this temperature. If the ECS cannot obtain this value from the PCS (because the PCS is down) it will revert to a default value for chiller temperature.

Figure 4-1 shows the data flow diagram for the thermal management scheme to be implemented in the PCS.

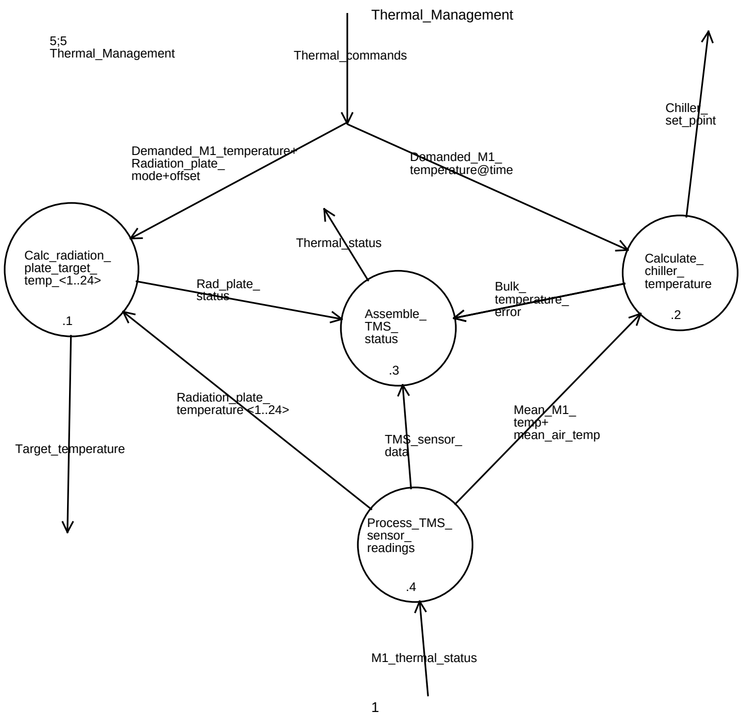

Figure 4-1 Thermal Management data flow

4.1.2.2 Thermal Management Processes

Calc radiation plate target temperature: The output values of the 24 individual radiation plate heaters are calculated based on the mode of operation, the individual offset values, the demanded M1 temperature and the actual radiation plate temperature.

The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Calculate chiller temperature: The required chiller temperature is calculated form the demanded M1 temperature, the time it must be attained, the actual mean M1 temperature and the actual air temperature.

The algorithms to be used in this process will be supplied by the Gemini optics group.

Process TMS sensor readings: Data from the thermal sensors associated with the TMS are processed to produce mean air temperature, mean M1 temperature and radiation plate temperatures.

Assemble TMS status: Combines status from the calculation processes and thermal sensors.

4.1.2.3 Thermal management data flows

Data flows into and out of the diagram are those described on the level 0 diagram.

Thermal commands: this consists of two sub flows

Demanded M1 temperature@ time: The required M1 temperature and the time at which it must be attained.

Radiation plate mode and offset: The mode in which the radiation plates will be operated (on or off) and the required individual heater offset per panel. This information will be entered via the engineering interface.

Radiation plate status: Demanded plate temperatures + operating mode.

Bulk temperature error: The difference between demanded and actual M1 temperature.

Radiation plate temperature <1..24>: The actual radiation plate temperatures.

Mean M1 temp + mean air temp: Mean M1 temperature calculated from thermal sensors on M1 and mean air temperature calculated from the temperature sensors mounted on the aperture plate above the mirror.

TMS sensor data: Combination of the *Radiation plate temperature <1..24>* and *Mean M1 temp + mean air temp* flows.

4.1.3 Cell Sensing

4.1.3.1 Thermal Sensors

TBD

4.1.3.2 Strain Gauges

TBD

4.1.3.3 Accelerometers

The accelerometer node box will continually monitor the 3 accelerometers. It will return to the PCS, at a 1 Hz rate, the rms and peak values for the preceding second. If the value exceeds the specified limit, (0.5g), an operator alarm will be raised.

4.1.4 Air Pressure Support

4.1.4.1 Introduction

The APSS controls the air pressure in the air pressure support, in the inner seal and outer seal. The air pressure of the air pressure support is controlled by up to 6 proportional valves whose set point may be determined either from the elevation angle or the sum of the axial load cells. The air pressure of the seals will be calculated as a function of Zenith angle.

The APSS 'safe mode' is defined as all valves being set to open to allow venting of the system to atmospheric pressure. Over pressure relief valves for both the APSS and seals will probably be provided as part of the support system. There may be a requirement for the PCS to monitor the status of these valves.

Figure 4-2 shows the data flow of the Air Pressure Support System.

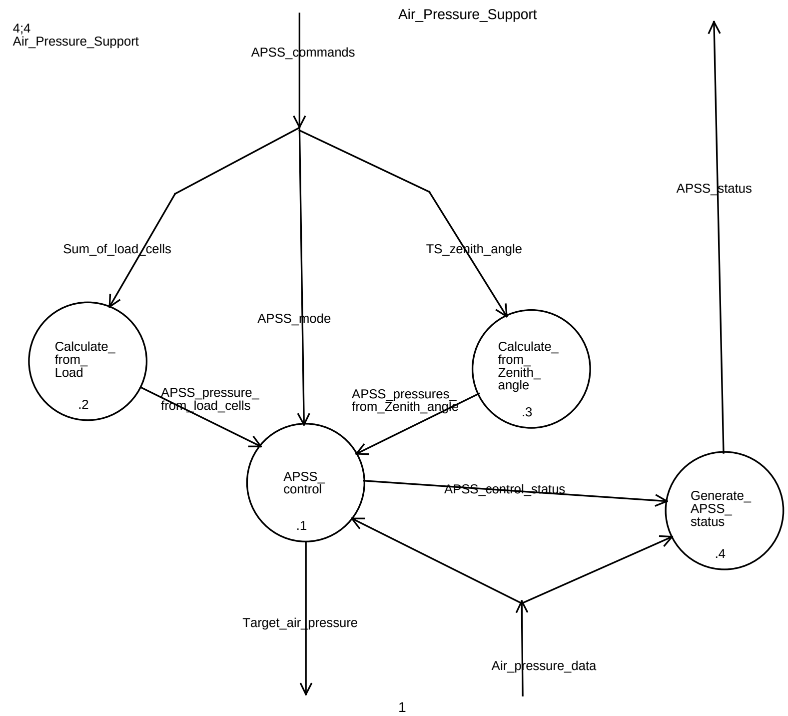

Figure 4-2 APSS data flow diagram

4.1.4.2 APSS processes

Calculate from load: The target air pressure for the air pressure support is calculated from the sum of the 'hydraulic' axial ***or*** lateral load cells.

Calculate from Zenith angle: The target air pressure for the air pressure support and both air seals is calculated from the *current* Zenith angle supplied by the TCS. To avoid the possibility of over pressuring the systems the APSS uses the actual Zenith angle.

APSS control: The initial intent is to servo the air pressure from the sum of upper load cell readings, the aim being to keep the load cell readings constant (equivalent to 20% of the total mirror weight at Zenith pointing). In operational modes a cross check will be performed on the air pressure reading by checking it against the value provided by the equation given below. An alarm will be raised if there is a discrepancy larger than TBD. In engineering mode this cross check will be disabled.

The air pressure P required for the APSS is given by the following equation.

$$P = 0.432(\cos\theta - 0.2)\ \mathrm{N / cm^2}$$

where θ is Zenith angle, from 0° to 75°

The air pressure of the seals will be calculated as a function of Zenith angle. The algorithm to be used will be supplied by the Gemini optics group.

If the Zenith angle signal from the TCS is lost the APSS will calculate the approximate Zenith angle by using the sum of the axial load cells, the load on the lateral load cells and the air pressure of the support. This will be made available to the other PCS systems to allow safe operation when the TCS is down.

Generate APSS status: The status of the APSS system comprises of processed sensor data, giving average air pressures and the status of the control system. The control system status includes the cross check on the air pressure support system pressure and any alarm conditions.

4.1.4.3 APSS data flows

The flows on and off of the diagram are those shown on the level 0 data flow diagram.

APSS commands: this consists of three sub flows

> **Sum of upper load cells:** The total load on the axial supports as measured by the 'hydraulic' axial load cells and/or the lateral load cells.
>
> **TS Zenith angle:** The *current* telescope Zenith angle
>
> **APSS mode:** This can be 'operational' or 'engineering'. In engineering the cross checking of demanded air pressure is disabled.

APSS pressure from load cells: The required air pressure support pressure as calculated from the sum of axial load cells.

APSS pressures from Zenith angle: The required air pressure for the seals and air pressure support calculated from current Zenith angle.

Target air pressure: The required air pressure for the seals and support system.

Air pressure data: Actual air pressures seen by sensors in the seals and air pressure support.

APSS control status: State of the control system. Including demanded pressures, air pressure support cross check alarm.

APSS status: State of the APSS; including processed sensor data, giving average air pressures and the status of the control system.

4.1.5 Passive Support

4.1.5.1 Introduction

The passive support system encompasses both axial and lateral hydraulic systems. The axial system can be used in either 3 or 6 zone mode. The Passive Support process of the level 0 diagram is divided in to axial and lateral sub-processes as shown in Figure 4-3. Note that there will be coupling between axial and lateral position sensors and it may not be possible to treat the position as separate axial and lateral processes as has been conceptually done here. *Note: It will be possible to generate separate X, Y and Z displacements and rotations from a matrix calculation. L. Stepp, 13/12/95.*

Note that the PCS control system in general operates in a fundamentally different way from other control systems on the telescope. This is because the PCS holds its own look up tables, whereas, for example. It is the TCS not the MCS that holds the LUTs for the telescope pointing model. This allows the PCS to be treated as a stand alone system that does not require the TCS to be available for it to operate.

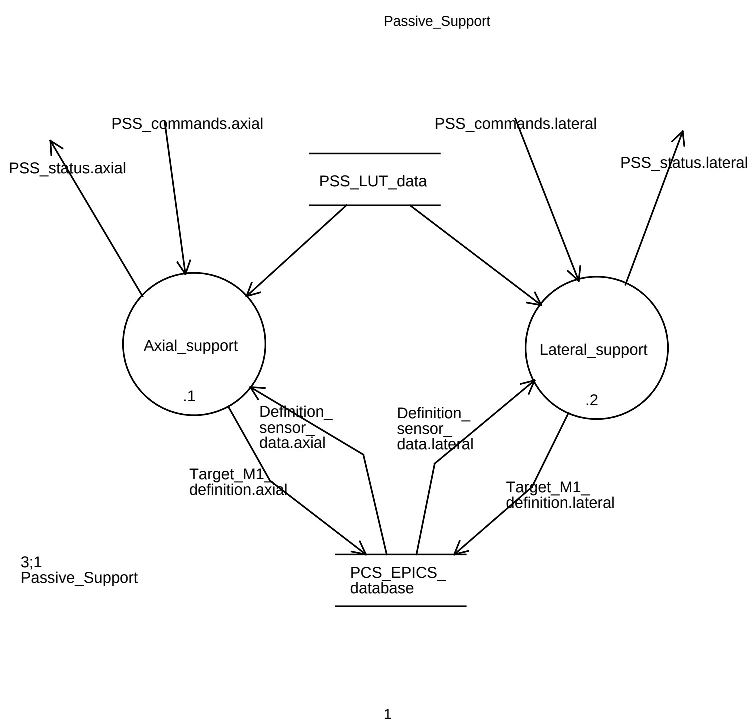

Figure 4-3 Passive support data flow diagram

The flows coming in and out of the passive support process are sub-divided into their axial an lateral components. Theses data flows are described in greater detail in the relevant sections below.

4.1.5.2 Axial support

The axial support system is responsible for M1 definition in Tz, Rx and Ry. Position information for these terms is provided by 4 position sensors mounted normal to the rear of the mirror at hard points defined by the bipod mountings of the M1 cell. Approximate positions are shown in Figure 4-4. If a failure is detected in one position

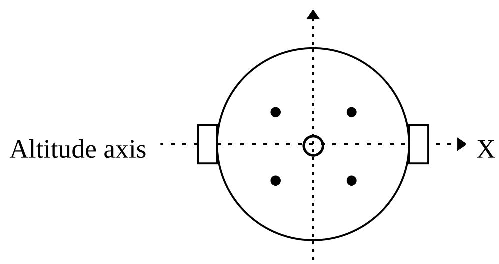

Figure 4-4 Axial position sensor locations

sensor, the system will disable further movement and inform the operator.

Actuation is provided by 6 hydraulic master cylinders, one per zone. These are driven by stepper motors.

Due to the changing gravity vector as the mirror moves, the volume of oil in the hydraulic zones will need to be changed via movement of the master cylinders. An update rate of once every 10 seconds for master cylinder position will be used.

The axial system in the 3 zone mode functions as a kinematic mount and can correct for rigid body piston and tip/tilt and is servoed off the position sensors.

When higher spatial frequency modes are present due to wind buffeting the axial support can be configured as a six zone system that is capable of correcting for astigmatism as well as focus and tip/tilt. In this mode it is servoed off of differential pressure sensors between each zone and by the wave front sensor data.

The control valves between zones will have 3 positions OPEN, CLOSED and INTERMEDIATE. The INTERMEDIATE position will provide controlled leaks between the zones to make them stiff at high frequency but compliant at low frequencies. The fluid volume in each passive zone is controlled based on feedback from position sensors and pressure sensors. The proper readings of these sensors are contained in look up tables as a function of telescope position and cell temperature. The frequency at which the number of effective zones changed could be set by the flow constrictors. This would result in a mirror that resisted high frequency effects, such as wind buffeting, extremely well and yet appeared as a kinematic mount to low frequency effects, such as gravity and temperature.

The TCS will command the PCS to switch from 6 zone mode into 3 zone mode whenever the telescope is to move more than 1[TBD] degree. [1]. The TCS will supply both target and actual elevation information to PCS. PCS will determine when to switch from 6 to 3 zone mode. The PCS will not return to 6 zone mode until commanded to do so by the TCS.

The PCS will be responsible for providing a demand signal to the hydraulic actuators but will not be responsible for closing a local loop around the actuators with local sensors. This will be done internally by the node box.

It will not be possible for the PCS to damage the primary mirror in either 3 or 6 zone mode [1]

The data flow of the Axial support process is shown in Figure 4-5. This shows details of both 3 and 6 zone control. The 6 zone control mode is described in greater detail in section 4.1.5.2.3.

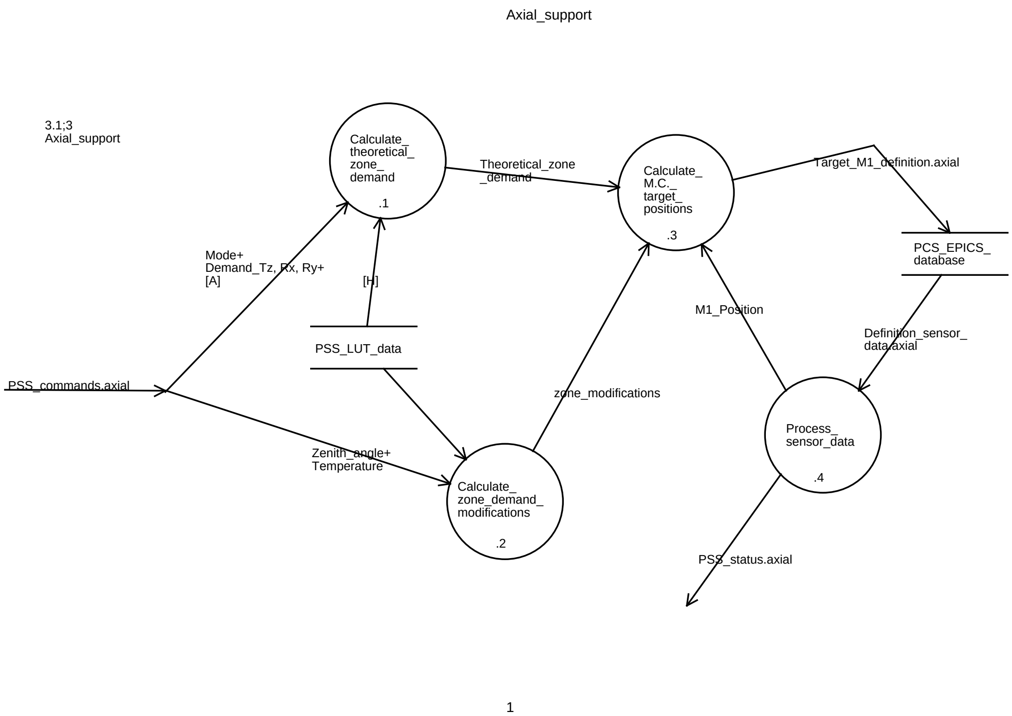

Figure 4-5 Passive axial support data flow

4.1.5.2.1 Axial processes

Calculate theoretical zone demand: This has two modes of operation. If *Mode* is 3-zone then the required position of the 3 master cylinders is calculated from the demanded values of *Tz, Rx and Ry*. If *Mode* is 6-zone then the required master cylinder positions are calculated and the required inter-zone pressures are calculated from the matrices [A] and [H] as described in section 4.1.5.2.3. The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Calculate zone demand modifications: It is expected that hydraulic pressures and sensor readings will change due to cell flexure and other effects as a function of (at least) Zenith angle and cell temperature. This process uses the Zenith angle and temperature (plus other terms if required) to calculate the required modifications to the theoretical master cylinder positions. A look up table (or tables) is used to perform these calculations. Initially the LUT would be full of zeros, until replaced by data obtained during calibration. The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Calculate M.C. target positions: This is the main control process. In 3-zone mode it takes the theoretical master cylinder position demands and the modification values to create a demand for the master cylinders. The actual M1 position in terms of *Tz, Rx and Ry* is fed back in to the loop to create a *proportional* servo loop. The final demanded value for the master cylinders is fed to them via the EPICS database and the I/O system.

In 3 zone mode since two zones are connected together it is possible to use either of the individual zone master cylinders to control the larger zone. There are various strategies that could be used for selecting which master cylinder to use. The preferred method is use the master cylinder with the largest travel available in the desired direction. If faster response is required an alternative would be to drive both master cylinders in parallel.

In 6-zone mode the inter-zone pressures are also fed-back to this process and are used as the feedback input to the proportional loop.

The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Process sensor data: This takes the 4 position sensor readings and converts them from their spherical co-ordinate system to give actual *Tz, Rx and Ry*. In 6-zone mode the inter-zone pressure sensors are also read.. The 'hydraulic' load cell data is used to calculate the total load per sector and the total load on the support system.

The individual zone pressure sensor data is processed to detect a unexpected pressure change that would indicate a leak. The master cylinder readings will be summed to give a volume/displacement check that will indicate a slow leak. This calculation will have to take into account temperature effects.

All this data is used to form the *PSS_status.axial* data flow.

The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Data stores: These are described in the level 0 diagram.

4.1.5.2.2 Axial data flows

PSS commands.axial: this consists of two sub flows

> **Mode + demand Tz,Rx,Ry + [A]:** Mode is either 3-zone or 6 zone; demand Tz, Rx, Ry are the demanded values received from the TCS, note that these are *offsets* from the current position; [A] is a matrix with 3 Zernike correction terms, it is fully described in section 4.1.5.2.3.
>
> **Zenith angle + Temperature:** The current telescope zenith angle. Temperature is the current temperature.

Theoretical zone demand: The predicted master cylinder positions to meet the demanded position given a perfect system.

Zone modifications: Predicted modifications to master cylinder positions required by current external conditions acting on system.

[H]: 6-zone influence matrix, described in section 4.1.5.2.3.

Target M1 definition.axial: target positions for axial master cylinders.

M1 position: Actual position of M1 plus inter-zone pressures.

Definition sensor data: Position sensor readings, pressure sensor readings, load cell readings.

PSS Status.axial: Actual position of M1 + hydraulic pressure sensor alarms + 'hydraulic' load cell information.

4.1.5.2.3 Axial 6 zone control

This section is taken from Larry Stepps note of 28 March 1995 [9].

When the axial supports used in the six zone mode, three bending modes are coupled to the cell. These consist of two orthogonal astigmatic bending modes, and one trefoil mode. Because these modes are controlled by the hydraulic part of the support, it does not make sense to try to correct them with pneumatic active optics system. Two methods will be used for control of these bending modes.

When the PWFS is in use, the measured figure errors in the modes described above can be corrected by the hydraulic system. Therefore, the associated elements of the Zernike vector [Z] (see section 4.1.6.1 for description), Z_4, Z_5 and Z_9, are set to zero. These three values are instead placed in a 3 x 1 vector [A]. The PCS executive will be responsible for performing this function. The TCS will simply pass [Z] to the PCS whatever the PSS mode.

The 3 bending modes are controlled by changing the zonal fluid pressures using the master cylinder units. A 6 x 3 hydraulic pressure influence matrix [H] will be developed, such that:

[P] = [H] x [A]

where:

[P] = a 6 x 1 vector listing the pressure differentials between adjacent zones

[H] = a 6 x 3 matrix in which each column is a hydraulic pressure influence function

[A] = a 3 x 1 vector of coefficients of three aberrations under hydraulic control

If the PWFS is not in use, no information about the three bending modes is available In this case, the differential pressures will be maintained at a zero level, using a slow servo bandwidth of 0.01 Hz to filter out wind effects.

4.1.5.3 Lateral control

Lateral control consists of two parts, Y-definition which controls Ty and Rz; and X-definition which controls Tx. Both are similar to the 3-zone axial system. Position is servoed off-of position sensor values.

Tx and Rz are given by one pair of position sensors and Ty by another pair.

If a failure is detected in one position sensor, the system will disable further movement and inform the operator.

Note that 4 load cells will be available in the X-definition system to provide diagnostic information.

The data flow for the lateral support is shown in Figure 4-6

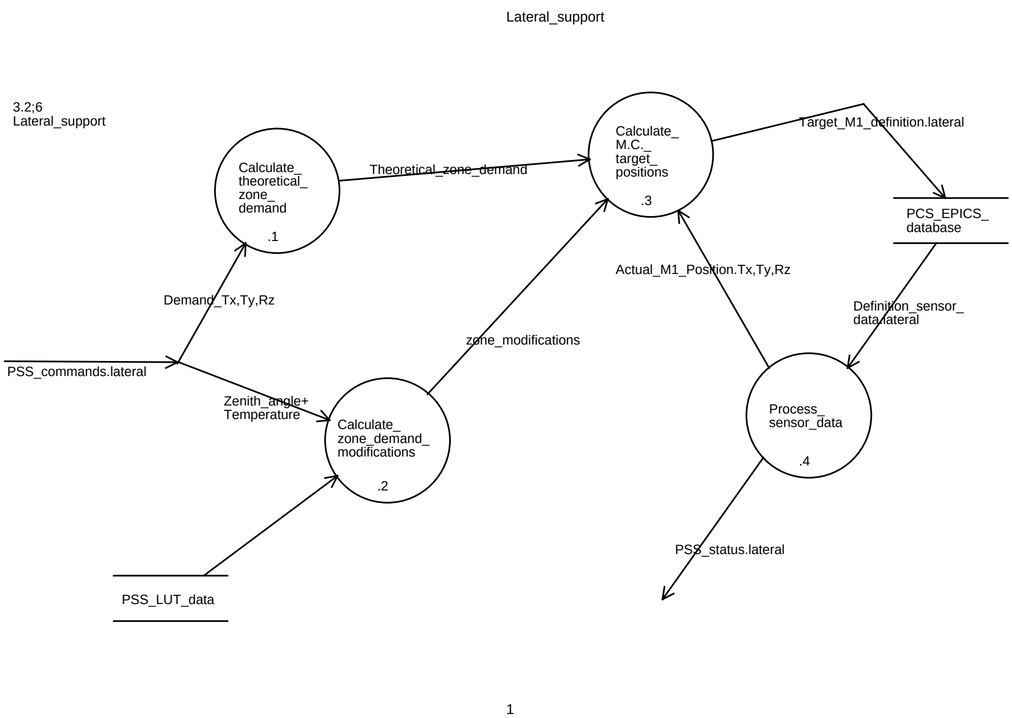

Figure 4-6 Lateral support data flow diagram

4.1.5.3.1 Lateral processes

Calculate theoretical zone demand: The required position of the 3 master cylinders is calculated from the demanded values of *Tx, Ty and Rz*. The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Calculate zone demand modifications: It is expected that hydraulic pressures and sensor readings will change due to cell flexure and other effects as a function of (at least) Zenith angle and cell temperature. This process uses the Zenith angle and temperature (plus other terms if required) to calculate the required modifications to the theoretical master cylinder positions. A look up table (or tables) is used to perform these calculations. Initially the LUT would be full of zeros, until replaced by data obtained during calibration. The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Calculate M.C. target positions: This is the main control process. It takes the theoretical master cylinder position demands and the modification values to create a demand for the master cylinders. The actual M1 position in terms of

Tx, Ty and Rz. is fed back in to the two servo loops (one for Ty, Rz and one for Tx) to create a *proportional* loop. The final demanded value for the master cylinders is fed to them via the EPICS database and the I/O system.

The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Process sensor data: This takes the 4 position sensor readings and converts them from their spherical co-ordinate system to give actual *Tx, Ty and Rz.* The load cell data is used to calculate the total load per sector and the total load on the support system.

The individual zone pressure sensor data is processed to detect a unexpected pressure change that would indicate a leak. The master cylinder readings will be summed to give a volume/displacement check that will indicate a slow leak.

All this data is used to form the *PSS_status.lateral* data flow.

The actual algorithms to be used in this process will be supplied by the Gemini optics group.

Data stores: These are described in the level 0 diagram.

4.1.5.3.2 Lateral data flows

PSS commands.lateral: this consists of two sub flows

> **Demand Tx, Ty and Rz:** Demand *Tx, Ty and Rz* are the demanded values received from the TCS, note that these are *offsets* from the current position.
>
> **Zenith angle + Temperature:** The current telescope zenith angle.. Temperature is the current temperature.

Theoretical zone demand: The predicted master cylinder positions to meet the demanded position given a perfect system.

Zone modifications: Predicted modifications to master cylinder positions required by current external conditions acting on system.

Target M1 definition.lateral: Target positions for lateral master cylinders.

M1 position: Actual position of M1.

Definition sensor data: Position sensor readings, pressure sensor readings, load cell readings.

PSS Status.lateral: Actual position of M1 + hydraulic pressure sensor alarms + load cell information.

4.1.5.4 Implementation

Figure 4-7 shows the implementation scheme for control of a hydraulic zone. Elevation, temperature and demanded changes in M1 definition are sent from the TCS to the PCS at a rate of 0.1Hz. The PCS converts these demands to changes in master cylinder position. The new master cylinder position is sent via CAN bus to the relevant NCU. The NCU closes a local proportional loop around the stepper motor and encoder to position the master cylinder piston.

Once in position the motor and encoder are powered down in order to reduce power dissipation. Node control units read pressure and position sensors and filter the signal to improve the signal to noise ratio before passing the filtered values to the PCS. The PCS uses either the position sensor or pressure sensor data to close a slow proportional loop around the master cylinder.

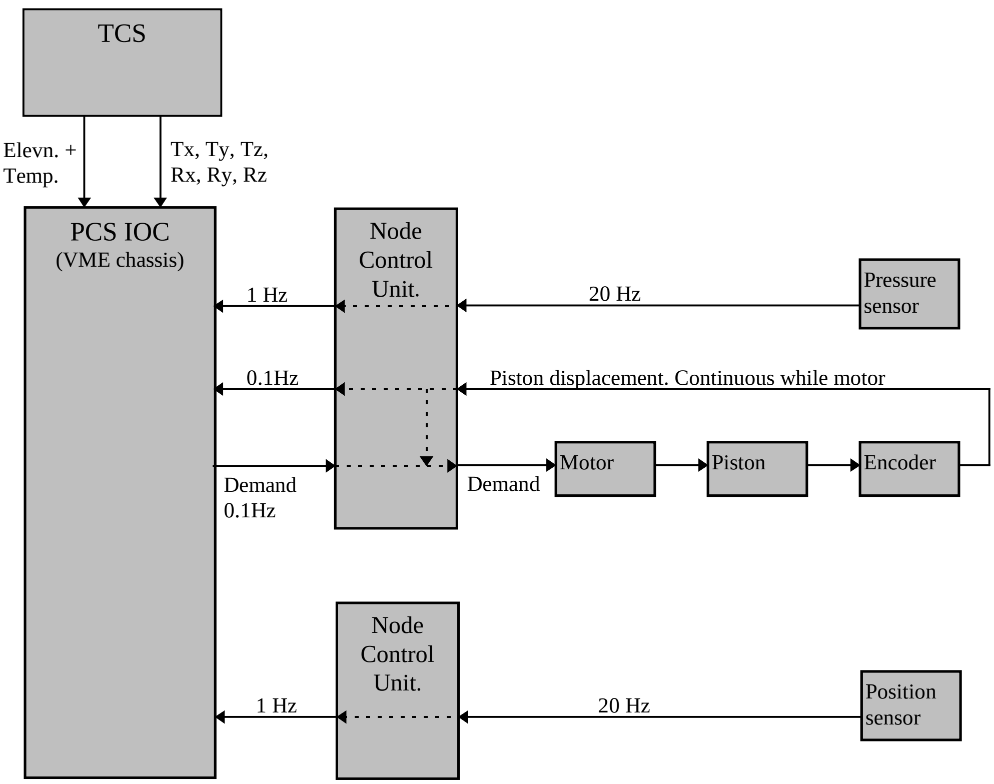

Figure 4-7 Passive Support Control implementation block diagram

4.1.6 Active Optics

The active optics system is divided into two processes as shown in Figure 4-8. data flows into and out of the diagram are those shown on the level 0 diagram. The **Control AO** is responsible for generating the actuator target forces. The **Generate AO Status** process forms status information from data provided by the I/O sub-system and the state of the Control AO process.

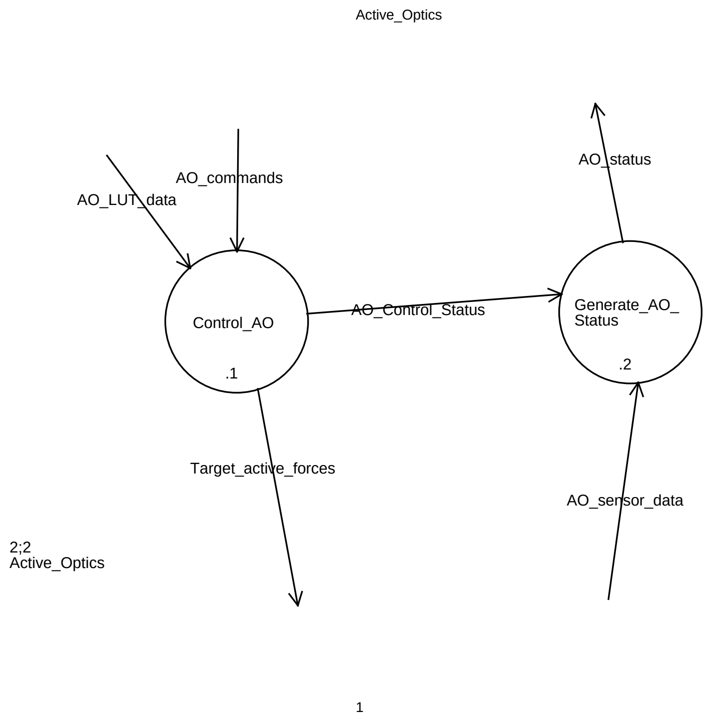

Figure 4-8 Active Optics data flow

4.1.6.1 Active Optics Control

The Control AO process is decomposed as shown in Figure 4-9. The Control algorithm is that described by Larry Stepp in his note of 28 March 1995 [9]. This section is basically a précis of this note. The algorithm is shown as a flow chart in Figure 4-10.

During operation the PWFS will measure wavefront errors every few seconds. These measurements will be averaged over periods of the order of one minute. Corrections to the M1 figure derived from this data will be passed to the PCS by the TCS as Zernike terms, de-rotated to the primary frame of reference.

When PWFS data is not available and in between Zernike updates a “Look Up Table” will be used to predict the correct actuator forces. Wavefront sensor feedback will be used to improve the correction accomplished by the LUT.

The Look Up Table

The LUT can be thought of as a list of 120 equations, one for each actuator. Each will have the form:

$$F_i^{LUT} = C_{i,1} + C_{i,2}\sin(\theta) + C_{i,3}\cos(\theta) + C_{i,4}T$$

Where:

F_i^{LUT} = the force at the *i*th actuator required by the LUT

$C_{i,1}, C_{i,2}, C_{i,3}, C_{i,4}$ = coefficients derived by calibration measurements

θ = zenith angle

T = Temperature in °C

The sine and cosine terms correct for errors that vary with zenith angle.

The constant term, $C_{i,1}$, corrects for initial mirror figure errors and other “constant” errors. There should be some change to this term in time, which would be detected by the HRWFS. Therefore this term is broken down into 2 terms, one of which is based on a long-term average and one the other based on measurements from the HRWFS at the beginning of the night.

This LUT will be implemented through matrix multiplication:

$[F^{LUT}] = [C] . [B]$

$[F^{LUT}]$ is a 120 x 1 vector, [C] is a 120 x 8 matrix and [B] is an 8 x 1 vector as shown below:

$$\begin{bmatrix} F_1^{LUT} \\ F_2^{LUT} \\ . \\ . \\ F_{120}^{LUT} \end{bmatrix} = \begin{bmatrix} C_{1,1} & C_{1,2} & C_{1,3} & C_{1,4} & C_{1,5} & C_{1,6} & C_{1,7} & C_{1,8} \\ C_{2,1} & C_{2,2} & C_{2,3} & C_{2,4} & C_{2,5} & C_{2,6} & C_{2,7} & C_{2,8} \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ C_{120,1} & C_{120,2} & C_{120,3} & C_{120,4} & C_{120,5} & C_{120,6} & C_{120,7} & C_{120,8} \end{bmatrix} \times \begin{bmatrix} 1 \\ \sin(\theta) \\ \cos(\theta) \\ T \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The terms in the [C] matrix are;

Column 1, $C_{i,1}$ = Long term average constant term

Column 2, $C_{i,2}$ = Coefficients for $\sin(\theta)$

Column 3, $C_{i,3}$ = Coefficients for $\cos(\theta)$

Column 4, $C_{i,4}$ = Coefficients for temperature

Column 5, $C_{i,5}$ = adjustments to constant terms from HRWFS

Columns 6 - 8 are reserved in case additional terms are required

Corrections involving feedback from the Peripheral Wavefront Sensor

The forces calculated from the LUT will be augmented using feedback from the PWFS. The forces applied to M1 will be a sum of the two:

$[F] = [F^{LUT}] + [F^{PWFS}]$

Each of these is a 120 x 1 vector. The forces calculated using the PWFS information will be updated periodically (about once per minute), and the current value will be equal to the old value plus a correction:

$[F^{PWFS}] = [F^{PWFS}]_{old} + [F^{PWFS}]_{delta}$

The correction $[F^{PWFS}]_{delta}$ will be calculated by a matrix operation based on modal pre-calculation. In order to represent the first 10 bending modes with a Zernike polynomial, 19 Zernike terms are required. The primary mirror will not be used to correct tilt, focus or coma, therefore the 5 associated Zernike terms will be deleted from the information used by the PCS. As a result 14 Zernike terms derived from the PWFS will be used in this matrix calculation. The equation is:

$[F^{PWFS}]_{delta} = [M] \times [Z]$

Where;

[M] is a 120 x 14 matrix. Each column is a list of 120 active forces required to produce a unit magnitude of a single Zernike term.

[Z] is a 14 x 1 vector. It contains magnitudes of the individual Zernike terms in the wavefront measured by the PWFS. These values will have been calculated so as to include corrections for the field position of the PWFS.

Reduction of forces that are out of range

To ensure that the “required” forces do not exceed that available from the actuators force reduction unit vectors must be used to bring the forces within bounds.

First, the amount each force is out of range must be calculated. Since active forces can be positive or negative each “required” force must be checked against high and low limits. A 120 x 1 over range vector [O] is created, containing the amount (plus or minus) each force exceeds the associated limit. It is likely that most of the elements of the [O] vector will be zero.

A 120 x 120 force reduction matrix [R] is created by combining the 120 force reduction unit vectors with each column in the matrix composed of one force reduction unit vector. A 120 x 1 vector of force reductions $[F]_{reductions}$ is then calculated as follows:

$[F]_{reductions} = -K \times [R] \times [O]$

Where;

K is a coefficient slightly larger than one used to ensure there is a little margin left after the force reductions

The final force vector is then calculated by adding the force reductions to the original force vector.

$[F]_{final} = [F]_{original} + [F]_{reductions}$

At this point the forces are checked again, and if some are still over the limits (because of cross coupling) the operation is repeated. Once all the forces are in bounds the forces are checked to ensure they are applying zero net force and moment to the mirror and then sent to the actuators.

The data flow for this operation is shown in Figure 4-9. It shows that matrices [C], [M] and [R] are obtained from the Look Up Table store. They are down loaded in to this store at boot up from an appropriate Unix host.

The results of the force reduction and net force and moment checks are passed to the Generate AO status module so that it is possible to keep check on the state of this process.

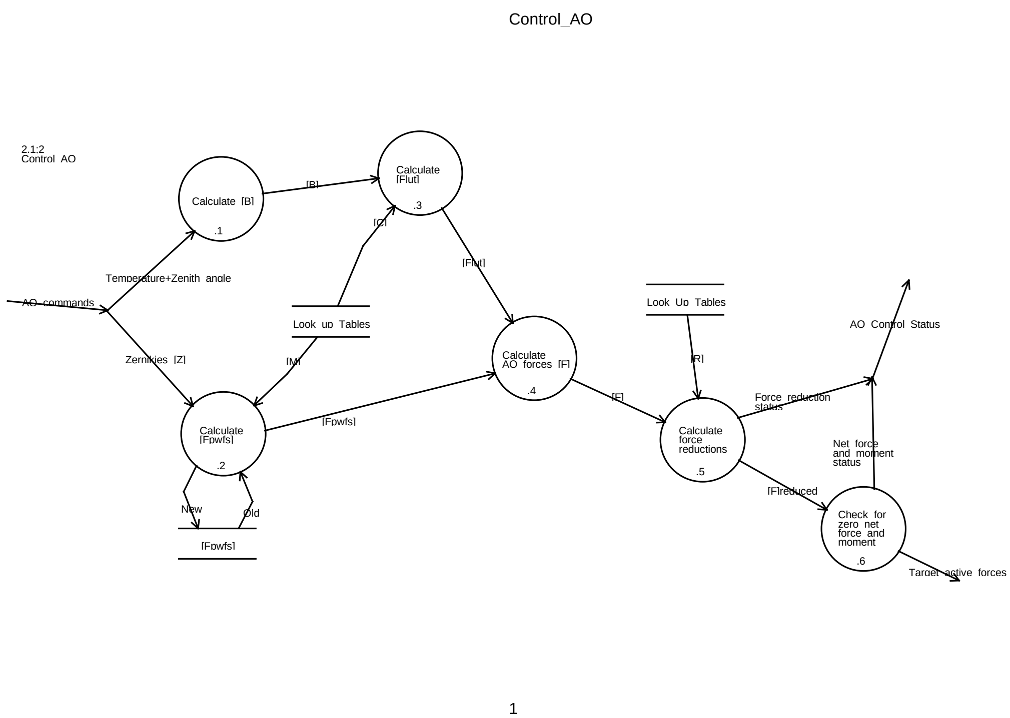

Figure 4-9 Active Optics Control data flow

Figure 4-10 shows the algorithm in flow chart form. This shows the relative frequency of the LUT and PWFS loops. Also note that if the zero net force and moment check fails the forces will not be sent to the actuators and an error will be flagged.

Note: This assumes that the matrices transformation will naturally achieve a balanced moment and zero net force.

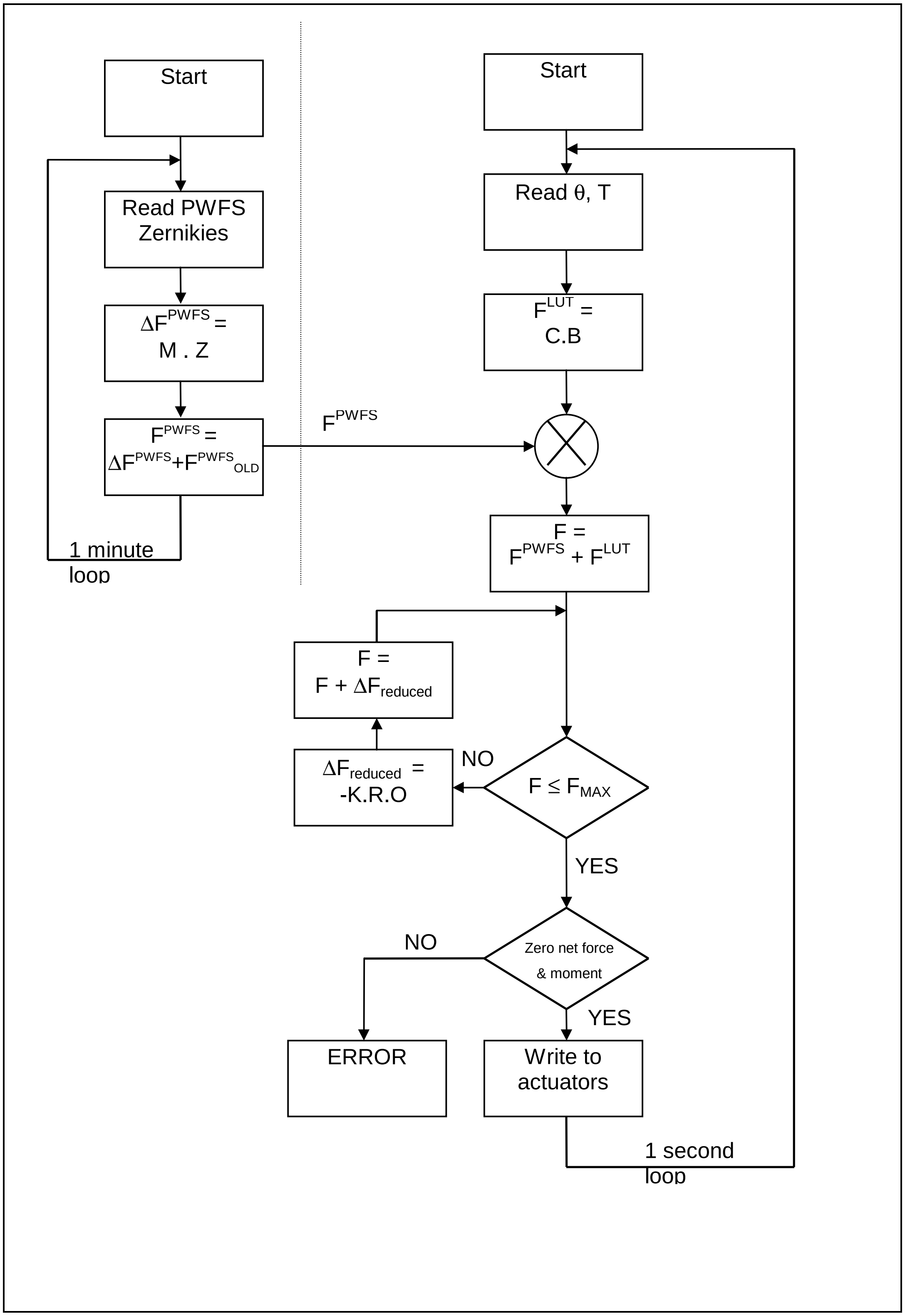

Figure 4-10 Active Optics control; flow chart

4.1.6.2 Active Optics status

Figure 4-11 shows the data flow of the Generate AO status process.

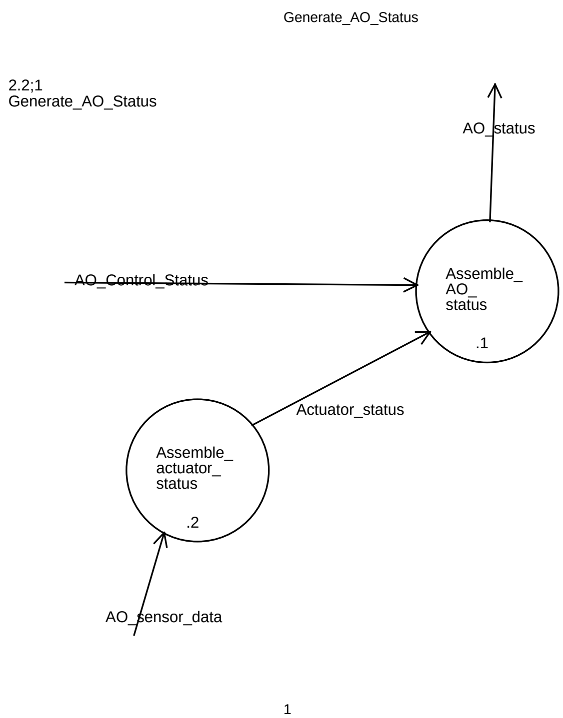

Figure 4-11 AO Status data flow

Assemble actuator status takes incoming data from the node control boxes via the EPICS database. It organises and processes this in to a form useful to higher processes.

The **Actuator status** flow consists of the 120 “active” load cell readings, error flags if any of these values are invalid and the actual net force and moment exerted by the AO actuators. A reading may be considered invalid if, for example, communication has been lost with that node.

The **AO Control status** flow contains information on the state of the AO control process.

The **Assemble AO status** merges this information in to a form that presents the complete status of the AO process.

4.1.6.3 Implementation of AO control

Figure 4-12 shows how the AO system is implemented. Command information in the form of Zernikies plus elevation information are fed to the PCS VME from the TCS. The Zernikies are fed at the rate of approximately once per minute. The Elevation is read at 1Hz

Using the set of algorithms described in section 4.1.6.1, the PCS calculates the required demand for each of the 120 axial pneumatic actuators. This algorithm is executed at 1Hz.

The Demand is fed to each Node Control Unit from the PCS at 1Hz. The NCU uses this demand as an input to a PI loop it is closing round the Pneumatic Valve with feedback from the ‘pneumatic’ load cell. If the PI loop is closed digitally it will run at more than 60 Hz, i.e. more than 20 times the nominal bandwidth of the actuator.

The load cells are read by the NCU at 100Hz. These values are averaged and fed back to the PCS at 5Hz. This ensures the PCS only ever uses load cell data with a maximum temporal spread of 200 ms.

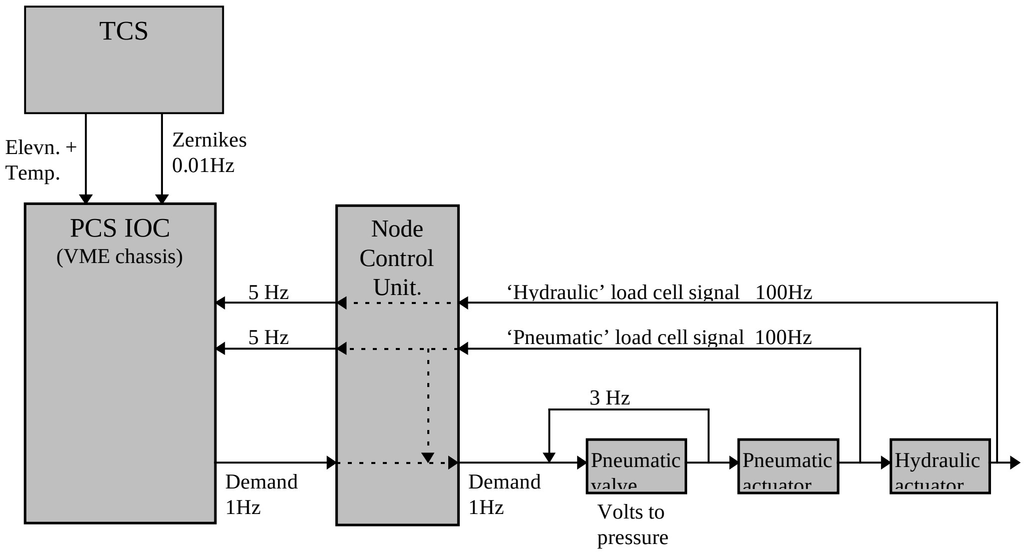

Figure 4-12 Active Optics Control implementation block diagram

4.1.7 Primary Mirror Engineering Screens

The engineering screens will be implemented with a combination of the EPICS EDD2/DM2 tools and PvWave.

4.1.8 I/O Subsystem

4.1.8.1 Introduction

The various components involved in getting signals to/from EPICS record in the PCS VME to/from the actuator and sensor hardware are shown in Figure 4-13.

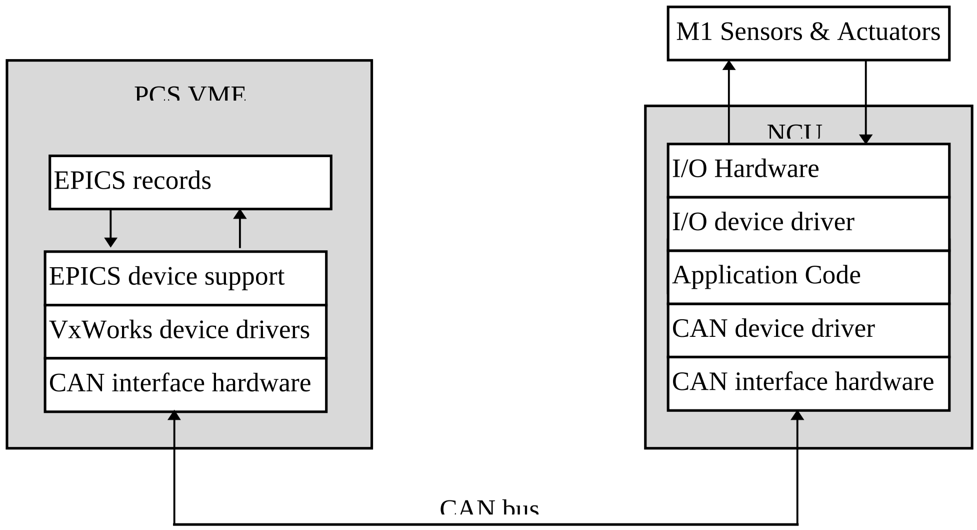

Figure 4-13 I/O sub-system, structure

The two major components are the VME and the NCUs which are inter-connected by CAN bus.

4.1.8.2 VME

The VME components are the CAN interface hardware and EPICS/CAN interface software. Selection of the CAN interface is described in section 6.1 this took account of hardware and software costs. Part of the criteria for determining hardware cost was the number of CAN bus interfaces required in the VME.

4.1.8.2.1 Calculating the number of CAN busses required

To determine this it was necessary to take in to account the following factors;

- Physical bus loading: The characteristics of the bus transceiver circuitry limit the number of nodes that may be connected to one bus. The most commonly used transceiver ICs (Phillips 82C250) can support up to 64 nodes on one bus.
- Bus length. The bit rate attainable on CAN systems is directly related to total bus length and the total length of cable drops from the main bus.
- Bus traffic loading: Theoretically it is possible for CAN to work with close to 100% bus loading. However in order to have low message queuing times CAN bus systems are generally run at less than 50-60% loading

Considering the physical bus loading first. The total number of nodes in the system is approximately 160. Using four busses gives a total of about 40 nodes per bus. This is less than the 64 nodes per bus requirement and allows some room for extra nodes to be connected at a later date. It also means that the balancing of nodes per bus does not have to be exact. For example one bus may have 45 nodes connected and another bus only have 35 nodes.

Next we can consider the bus length and bit rate. Although the cell is only about 8 meters across, the actual length of a bus will be much more because of the need to ‘snake’ around the interior partitions of the cell. For a four bus system a worst case assumption would be a bus length of, say, 30m. For each node there will be a drop from the main bus. A worst case assumption would be a drop of 0.5m per drop, giving a total drop length per 40 node bus of 20m.

The specification for CAN gives maximum cable lengths for given bit rates. The DeviceNet specification [10] (DeviceNet is a standard that uses CAN as its communications medium) includes a specification for allowable drop lengths. These are summarised in Table 4-1.

Bit rate	Total bus length	Cumulative drop length
1 Mbit per second	40m	-
500 kbit per second	100m	39m
250 kbit per second	200m	78m
125 kbit per second	500m	156m

Table 4-1 CAN bus cable lengths

It can be seen that a bit rate of 500kbits/second would be possible given our design parameters. However because a slower bit rate generally allows more reliable communications and will reduce EMI emissions a bit rate of 250 kbits/second has been selected. If this does prove too slow, the 500kbits/second rate could be used.

Given these parameters it is possible to estimate the traffic loads on the bus. Table 4-2 shows the predicted message rates for the main system components.

Source	Qty	I/P or O/P Hz	Readings/Sec
Axial			
Pneumatic L.C.	120	5	600
Hydraulic L.C.	120	5	600
Temperature	54	1	54
Joucu. demand	120	1	120
Joucu. pressure	120	1	120
Node temp	360	1	360
APSS			0
Joucu. demand	6	1	6
Joucu. pressure	6	1	6
Pressure sens.	6	1	6
Node temp	18	1	18
Preload			0
Joucu. demand	6	1	6
Joucu. pressure	6	1	6
Pressure sens.	6	1	6
Node temp	18	1	18
Air Seal			0
Joucu. demand	2	1	2
Joucu. pressure	2	1	2
Node temp	6	1	6
Master Cylinders			0

Position	8	1	8
Demand	8	1	8
Switches	8	1	8
Node temp	8	1	8
Cell sensing			0
Strain Gauges	32	1	32
Lateral			0
L.C.	60	5	300
Position sensors	8	1	8
Total			2308

Table 4-2 Predicted I/O message rates

Assuming a 4 bus system we can predict the expected traffic from a heavily loaded bus as shown in Table 4-3 which assumes a bus with 46 nodes attached.

Source	Qty	I/P or O/P Hz	Readings/Sec
Axial (40 nodes)			
Pneumatic L.C.	40	5	200
Hydraulic L.C.	40	5	200
Temperature	20	1	20
Joucu. demand	40	1	40
Joucu. pressure	40	1	40
Node temp	120	1	120
APSS			0
Joucu. demand	2	1	2
Joucu. pressure	2	1	2
Pressure sens.	2	1	2
Node temp	6	1	6
Preload			0
Joucu. demand	2	1	2
Joucu. pressure	2	1	2
Pressure sens.	2	1	2
Node temp	6	1	6
Cell sensing			0
Strain Gauges etc	16	1	16
Total			660

Table 4-3 Predicted I/O traffic for one CAN bus

Therefore for a heavily loaded bus we could expect 660 messages per second.

Bus loading can be calculated by dividing the number of bits to be transmitted by the bit rate of the bus (1)

$$BusLoading\% = \frac{transmittedBits}{BusRate} \times 100 \qquad (1)$$

In the worst case each CAN message would consist of 62 bits. The number of bits to be transmitted on the bus per second is

$$transmittedBits = NoMessages \times bits / message \qquad (2)$$

$$transmittedBits = 660 \times 62 = 40920$$

Assuming a bus rate of 250kbits/second, from (1)

$$BusLoading\% = \frac{transmittedBits}{BusRate} \times 100 = \frac{40920}{250 \times 10^3} \times 100 = 16.4\%$$

This message rate assumes messages are sent from the NCUs periodically without being requested by the VME. To take a worst case scenario where the VME must poll each NCU for each message, the number of messages double. The number of bits to be transmitted does not quite double because the ‘polling’ messages are shorter than the data messages.

A polling message consists of 44 bits so the total number of bits to be transmitted is now

$$transmittedBits = 660 \times 62 + 660 \times 44 = 40920 + 29040 = 69960$$

Again assuming a bus rate of 250kbits/second, from (1) bus loading is now

$$BusLoading\% = \frac{transmittedBits}{BusRate} \times 100 = \frac{69960}{250 \times 10^3} \times 100 = 28\%$$

So given the worst case data transmission requirements and a modest bus rate of 250kbits/second bus loading is 28%.

In practice it is expected that most messages will be transmitted periodically by the NCUs with only NCU status being polled for by the VME. NCU status gives information on the health of the NCU and should not be confused with messages containing sensor data.

By having messages generated periodically by the NCUs, without an explicit polling message being sent by the VME, not only reduce the load on the bus but also on the VME CPU which does not have to generate the polling messages.

The one polling message from the VME to the NCUs, to request NCU health status, will act like a ‘heartbeat’ message from the VME. This enables the NCUs to detect the presence of the VME. If the VME disappears it will then be possible for the NCU to enter a safe mode as required.

The actual bus load is expected to be in the region of 20%. This has been born out by small scale simulations and analysis of network traffic.

4.1.8.2.2 VME software

Andrew Johnson has written the EPICS/CAN interface software shown as the EPICS device support and VxWorks device driver layers in Figure 4-13. Documentation describing this software is *devCAN- CAN Bus Device Support* [11], *drvIpac - Industry Pack Driver* [12] and *drvTip810 - CAN Bus Driver* [13].

4.1.8.3 NCU

The PCS package is responsible for the NCU CPU hardware. The selection of this hardware is described in section 6.2 which also describes the functionality of the CPU board.

EMC testing of this board will be included as EMC tests of an entire NCU.

Responsibility for the NCU software is divided between the PCS and mirror support packages. The mirror support package is responsible for application specific code such as that required for PID control of the pneumatic actuators. The PCS is responsible for the CAN interface software and general structure of the code.

The cost of the NCU software will probably be split 50/50 between the two packages.

5. Interface Description

5.1 Introduction

This section describes the internal and external interfaces of the PCS.

The majority of external interfaces are connected to the I/O subsystem and fed into the PCS EPICS database. From here they are available to the other entities in the PCS. To make this document more useful, the external interfaces required by each entity are briefly described in a sub-section relating to that entity. The external signals are fully detailed in section 5.9 *I/O Subsystem.* Each signal is give a reference number so a cross reference can be made between the two.

To some extent this duplicates the information found in the individual interface descriptions. It has been done to assist the reader to view the interface from the point of view of either an individual design entity or the I/O sub-system.

5.2 PCS Executive

5.2.1 External interfaces

5.2.1.1 PCS/TCS

This interface is described in *TCS Interface Design Description 4 - The TCS/PCS interface* [7]

5.2.1.2 PCS/GIS

An *Interlock Event* is an event occurring within the PCS that causes the PCS to signal to the GIS that this event has occurred.

An *Interlock Demand* is a signal from the GIS to the PCS. On receipt of an interlock demand the PCS will take the appropriate action to ensure that no damage can be caused to the system or personnel.

The physical interface to the GIS is TBD. The GIS is described in ICD 12 - Interlock System [17]

5.2.1.2.1 Interlock Demands

Interlock Demand	PCS response	Mechanism
TMS Chiller pump not running or coolant pressure low	Remove power from radiation plate heaters	Hardware
Telescope tipped below 15 degrees.	Release APSS air pressure	Hardware
Surface heating disabled	Switch off surface heating	Hardware
Earthquake protection triggered	Do not attempt to drive system	Hardware
Global PCS stop	Stop all movement and cancel all pending commands	Software

Table 5-1 PCS Interlock demands

5.2.1.2.2 Interlock Events
None

5.2.1.3 PCS/Time Service
This is the standard Gemini time bus interface described in *ICD 9 -EPICS Time Bus Driver A.N.Johnson* [5]

5.2.1.4 PCS/Configuration data

5.2.1.4.1 EPICS Database Configuration
At present the preferred method is to use the technique proposed for the TCS. This uses an ASCII file to hold the default values for important records. This is used at boot-up to initialise the EPICS database.

Another possibility would be to use the EPICS Back Up and Restore Tool, BURT.

5.2.1.5 PCS LUT Configuration
Each LUT will be in a separate file.

Data will be in ASCII format, space delimited. Matrices formatted one row to a line, rows terminated with a carriage return.

5.2.1.5.1 Files required

Function
[C] AO open loop LUT
[M] modal pre-calculation matrix for AO Zernike terms
[R] Force reduction matrix for AO forces
[H] Hydraulic pressure influence matrix for 6 zone PSS mode
Mirror definition LUTs

Table 5-2 LUT Configuration files

In the first instance the LUTs will be generated by calculation and will be provided by the Gemini optics group. The tables will subsequently be modified as the result of calibration data.

5.2.1.6 PCS Alarms
Alarm information is made available to any system that requires it via EPICS channel access.

Major alarms will be generated for the following cases

I. **APSS over limit:** The sum of the 120 upper load cell forces is less than 10% of M1 weight.

II. **APSS failure:** The sum of the 120 upper load cell forces is greater than 30% of M1 weight.

III. **Lateral M1 defining near limit:** The displacement at any lateral position sensor is larger than TBD mm.

IV. **Axial M1 defining system near limit:** The displacement at any axial position sensor is larger than TBD mm.

V. **Axial actuator near limit:** Upper or lower load cell of an axial actuator (or actuators) exceeds operating range.

VI. **Lateral actuator near limit:** The load cell of an lateral actuator (or actuators) exceeds operating range.

VII. **Accelerometer alarm:** The acceleration on the mirror exceeds 0.5g.

VIII. **Loss of hydraulic pressure:** A sudden loss of hydraulic pressure in any hydraulic sector.

IX. **Hydraulic leak:** Detection of a slow hydraulic leak

X. **Master cylinders near end of range:** Master cylinder within TBD % of end of range.

XI. **Major I/O fault:** The loss of communications with one or more entire CAN buses.

XII. **Position sensor data inconsistent:** Indicating failure of one or more sensors. (Mirror movement must also be disabled in this case).

5.3 Thermal management

5.3.1 External interfaces

The PCS/Thermal management system will be described by ICD 1.2.4/1.2.10.

5.3.1.1 Electronic Interfaces

The electronics interface consists of

- Outputs from PCS: Control signals are sent out from the PCS, via its Node Control Units (NCUs), to the radiation plate heaters
- Inputs to the PCS from thermal sensors.

5.3.1.1.1 Radiation plate control signal output from PCS

The PCS provides a control signal to each of the 24 radiation plate heaters via a nearby NCU.

5.3.1.1.2 Connector

2 way, polarised, type TBD.

5.3.1.1.3 Signal

Note: The control signal shall be opto-isolated at the input to the Radiation Plate control unit.

Description:	2 wire, PWM, nominal 24V.	
Signal level:	Off:	< 1V
	On:	24V nominal, > 20V (into a 10KΩ Load), < 28V
PWM Characteristics:	Minimum frequency = 2.4Hz	
	Resolution = 8 bit (1 part in 256)	

Figure 5-1 shows examples of PWM timing.

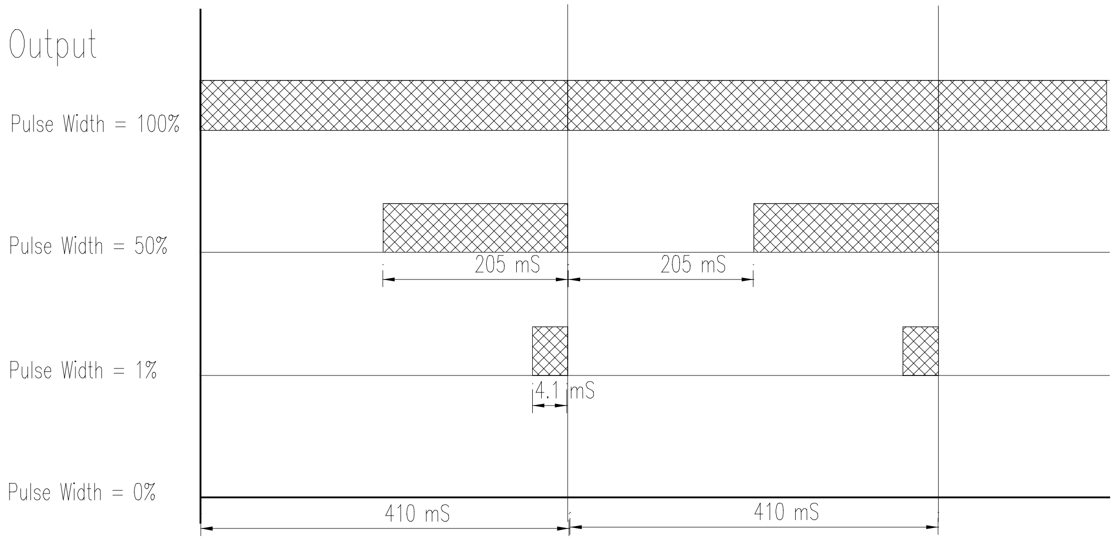

Figure 5-1 PWM Timing Examples

5.3.1.1.4 Thermal sensor inputs to PCS

Signal Function	Number	Described in Table	Signal Reference
Radiation plate temperature sensors	24	Table 5-13	A4
M1 temperature sensors	12	Table 5-13	A4
Air temperature sensors	6	Table 5-13	A4

Table 5-3 Thermal management system I/O sub-system interfaces

5.3.1.1.4.1 Sensors

The sensors shall be Platinum resistance devices (Pt100).

The sensors shall be connected in a differential bridge or 4-wire configuration (See M1 assembly CDR, M1 support system drawings).

Working Temperature Range: -35°C to +35°C

Accuracy: 0.1°C

Differential accuracy 0.05°C

5.4 Cell Sensing

5.4.1 I/O sub-system interface

Signal Function	Number	Described in Table	Signal Reference
Cell Temperature	16	Table 5-13	A3
Strain Gauge	32	Table 5-16	C1
Accelerometer	3	Table 5-16	C1

Table 5-4 Cell Sensing I/O sub-system interfaces

5.5 Air Pressure Support

5.5.1 I/O sub-system interface

Signal Function	Number	Described in Table	Signal Reference	Notes
APSS air pressure input	6 (TBD)	Table 5-13	A9	Rate = 1 Hz
Inner seal pressure	1	Table 5-13	A9	Rate = 1 Hz
Outer seal pressure	3 (TBD)	Table 5-13	A9	Rate = 1 Hz
APSS air pressure output	6 (TBD)	Table 5-13	A5	Rate = 1 Hz
Inner seal pressure output	3 (TBD)	Table 5-13	A5	Rate = 1 Hz
Outer seal pressure output	1	Table 5-13	A5	Rate = 1 Hz

Table 5-5 APSS I/O sub-system interfaces

5.6 Passive Support System

5.6.1 I/O sub-system interface

Signal Function	Number	Described in Table	Signal Reference	Notes
Upper (hydraulic) axial load cell	120	Table 5-13	A2	
Position sensors	8	Table 5-13	A9	Rate = 1 Hz, Operating range = ± 1000µm
Hydraulic zone pressure	12	Table 5-13	A9	Rate = 1 Hz
Axial support differential pressure sensors	6	Table 5-13	A9	Rate = 1 Hz
Inter zone valve position	6	Table 5-13	A11 + A12	Rate = 1 Hz, Positions = Open | intermediate | closed
Lateral load cell	60	Table 5-15	L1	Rate = 1 Hz
Lateral hydraulic pressure sensor	3	Table 5-13	A9	Rate = 1 Hz
X-definer hydraulic pressure sensor	2	Table 5-13	A9	Rate = 1 Hz
Axial master cylinder demand	6	Table 5-14	M1 + M2	
Inter-zone valve position	6	Table 5-13	A14 + A15	
lateral master cylinder demand	2	Table 5-14	M1 + M2	
X-definer master cylinder demand	1	Table 5-14	M1	

Table 5-6 PSS I/O sub-system interfaces

5.7 Active Optics

5.7.1 I/O sub-system interface

Signal Function	Number	Described in Table	Signal Reference	Notes
Axial ‘pneumatic’ load cell readings	120	Table 5-13	A1	
Demanded actuator load	120	Table 5-13	A13	
Axial pneumatic preload readings	6	Table 5-13	A9	Rate = 1 Hz
Axial pneumatic preload demand	6	Table 5-13	A13	Rate = 0.1 Hz

Table 5-7 AO I/O sub-system interfaces

5.8 Primary Mirror Engineering Screens

5.8.1 External interfaces

5.8.1.1 Screens

The PMES will run on a remote UNIX workstation. This may be located virtually anywhere, e.g. the Mount engineering console, the control room or at sea level. Entering of commands through this screen system will **not** disable the acceptance of commands from the TCS

It will consist of a hierarchical screen system. Each screen will be capable of calling lower level screens that show greater detail until the lowest level screens are reached. The hierarchy of these screens is shown in Figure 5-2

5.8.1.1.1 PMES Top Level Screen

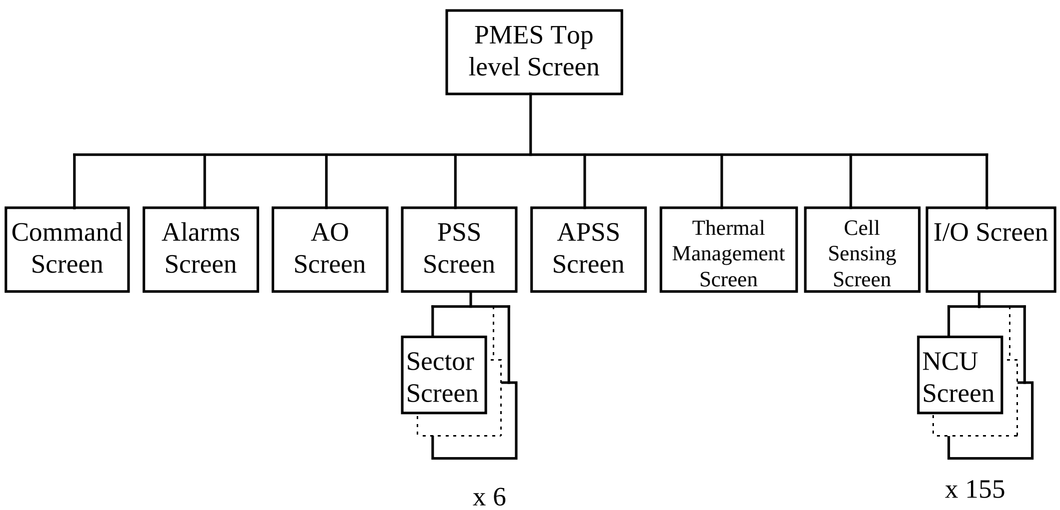

Figure 5-2 Primary Mirror Engineering Screens Hierarchy

Gives a top level summary of PCS status.

- Allows call-up of sub-displays
- Shows summary of active alarms

5.8.1.1.2 Command Screen

Allows entry of TCS commands from engineers screens.

5.8.1.1.3 Alarms Screen

Shows

- list of current major and minor alarms in EPICS database

- current interlock events
- current interlock demands

5.8.1.1.4 AO Screen

Shows status of AO system, including

- Point Map of actuator forces
- Contour Map of actuator forces
- Allows access to related NCU screens
- Allows setting of zone pre-load pressures
- Allows toggling of zero net force and moment check (on or off)

5.8.1.1.5 PSS Screen

Shows status of PSS system, including

- Graphical and numerical display of M1 De-centre and tip tilt WRT M1 cell
- Strip chart history of M1 De-centre and tip tilt WRT M1 cell
- Zone pressures
- Allows switch between 3/6 zone control
- Allows access to related NCU screens
- Access to automated sequencing routines for control of computer valves e.g. for filling hydraulic system

5.8.1.1.6 APSS Screen

Shows status of APSS system, including

- Strip chart histories of Actual and Target ‘air bag’ pressures plus error
- Numeric display of current and target ‘air bag’ pressures plus error
- Allows ability to set target ‘air bag’ pressure
- Allows ability to set target seal pressures
- Allows access to related NCU screens
- Strip chart histories of Actual and Target inner and outer seal pressures plus error
- Numeric display of current and target inner and outer seal pressures

- Allows ability to switch Zenith altitude check on or off

5.8.1.1.7 Thermal management Screen

Shows status of thermal control system, including

- Average M1 bulk, ambient air and radiator panel temperatures
- Individual sensor temperatures
- Radiation panel control mode
- Strip chart history of temperature error
- Allows the ability to set individual radiation panel temperature offsets
- Allows access to related NCU screens

5.8.1.1.8 Cell Sensing Screen

Shows status of Cell Sensing system, including

- Average cell temperature
- Strip chart history of individual temperature sensor values
- Strip chart history of strain gauge values
- Accelerometer values as strip-charts

5.8.1.1.9 I/O Screen

Shows status of I/O sub-system, including

- Status of CAN buses
- Access to all NCU screens

5.8.1.1.10 Sector Screens

Shows status of individual sectors including

- Zone pressure
- Inter zone pressure
- Total force on zone
- Computer valve status
- Control of computer valves

5.8.1.1.11 NCU Screens
Shows status of individual NCUs including

- General node status/health
- Values of all I/O signals
- Allows any output to set

5.8.1.2 I/O sub-system interface
All I/O signals are accessible to the PMES via the EPICS database. Some I/O signals are intended specifically for diagnostic information. These signals might be used only by the PMES, they are detailed below.

Signal Function	Number	Described in Table	Signal Reference
Axial Joucomatic temperature	136	Table 5-13	A8
Internal node air temperature	136 8 8 5	Table 5-13 Table 5-14 Table 5-15 Table 5-16	A6 M9 L2 C2
Internal cooling plate temperature	136 8 8 5	Table 5-13 Table 5-14 Table 5-15 Table 5-16	A7 M10 L3 C3

Table 5-8 PMES specific I/O sub-system interfaces

5.9 I/O Subsystem

5.9.1 External interfaces

5.9.1.1 Node Box Types
There are 4 basic types of node box, these being

- Axial
- Master Cylinder
- Lateral
- Cell Sensing

The details of each node box type are given below.

5.9.1.1.1 Axial Type Node Box

The principle task of this type is to control a Joucomatic valve and read back 2 load cell values. There are a total of 136 boxes of this type of node box accounted for as follows

Function	Number
Axial supports, 1 box per support	120
Pneumatic pre-load, 1 per sector	6
APSS, 1 per Joucomatic valve	6
Outer seal	3
Inner seal	1

Table 5-9 Axial type node box distribution

The axial node box signals are described in Table 5-13

Figure 5-3 shows in a simplified schematic the signal flows in an axial support NCU. Note that the ADC signals are sent back to the PCS via the CAN bus. The servo loop around the Joucomatic pneumatic valve and the pneumatic load cell is closed by the microcontroller. This loop is not within the scope of this package.

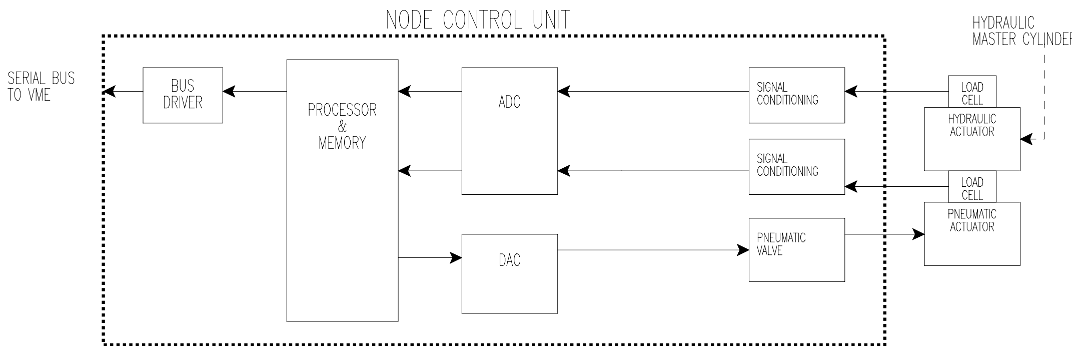

Figure 5-3 Schematic view of axial NCU

5.9.1.1.2 Master Cylinder Node Box

This type of node box is capable of driving 2 stepper motor channels. Each channel has;

- Stepper motor drive
- Absolute encoder input
- Limit switch input
- Solenoid drive

Axial master cylinders are composed of two actuators, one coarse and one fine. One axial master cylinder unit can therefore be controlled from one node box. Lateral Master Cylinders have only one actuator, therefore two lateral Master Cylinders can be driven from one node box. There are 8 node boxes of this type allocated as follows

Function	Number of boxes
Axial zones	6
Lateral zones	2

Table 5-10 Master cylinder node box allocations

The Master Cylinder node box signals are described in Table 5-14

5.9.1.1.3 Lateral Type Node Box

These units read in 8 lateral load cells each. There are 8 of theses node boxes, their signals are described in Table 5-15

5.9.1.1.4 Cell Sensing Type Node Box

These units are used to read strain gauges and accelerometers. Each unit can read up to 8 channels. Four boxes are allocated for strain gauge measurement and one for accelerometer measurement. The cell sensing node box signals are described in Table 5-16

5.9.1.2 Node Box Allocation

The physical location of the node boxes in the mirror cell is given by the drawing *M1 Cell/Control node box interface. 90-GP-0002-1009.* The boxes in this drawing are labelled as follows

- ANB1 - ANB120 For all the axial support node boxes
- SNB1 - SNB41 For 'Spare' node boxes, i.e. for all other boxes.

Some boxes have secondary functions. The mapping of these numbers to box types is given in Table 5-11 for primary functions and in Table 5-12 for secondary functions. This mapping is based on Eugene Huangs note of 15 September 1995 [8]

Drawing label	Type	Function	Number
ANB1 - ANB120	Axial	Axial actuator control	120
SNB 7, 12, 18, 19, 25, 30	Axial	Pneumatic pre-load control. Zones 1 to 6 respectively	6
SNB 4, 9, 15, 22, 28, 33	Axial	APSS. Zones 1 to 6 respectively	6
SNB 41	Axial	Inner seal	1
SNB 8, 17, 29	Axial	Outer seal	3
SNB 2, 6, 5, 32, 31, 35	Master cylinder	Axial hydraulic zones. Zones 1 to 6 respectively	6
SNB 36	Master Cylinder	Lateral support zones	1

SNB 1	Master Cylinder	X-definer	1
SNB 37, 38, 39, 40	Cell sensing	Strain gauge measurement at bipod feet	4
SNB 20	Cell sensing	Accelerometers	1
SNB 3, 10, 11, 16	Lateral	Lateral load cell reading +X side	4
SNB 21, 26, 27, 34	Lateral	Lateral load cell reading -X side	4
Total			157

Table 5-11 Node box primary functions

Drawing label	Node Box Type	Function	Number
ANB 34, 39, 46, 51	Axial	Axial position sensors	4
ANB 85, 102, 103, 120	Axial	Lateral position sensors	4
TBD	Axial	axial support hydraulic pressure sensors	7
TBD	Axial	axial support differential hydraulic pressure sensors	6
TBD	Axial	Lateral support hydraulic pressure sensors	3
TBD	Axial	X-definer hydraulic pressure sensors	2
TBD	Axial	Radiation plate temperature sensors	24
TBD	Axial	M1 temperature sensors	12
TBD	Axial	Ambient air temperature sensors	6
TBD	Axial	M1 cell temperature sensors	16
TBD	Axial	Control valves for hydraulic system	12
TBD	Lateral	Monitor 4 X-definer load cells	4
Total			100

Table 5-12 Node box secondary functions

5.9.1.3 Node box signal descriptions

Signal Reference	Function	Description	Type	Units	Operating range	Max	Min	Rate	Required Resolution	Actual Resolution Bits	Actual Resolution Units	Reference
A1	Pneumatic Load	Pneumatic Load Cell reading	Analogue Input	N	±400N	+500N	-500N	5Hz	0.2N	14 bit	0.061N	[1]
A2	Support Load	Hydraulic Load Cell reading	Analogue Input	N	0N to 2000N	+2000N	0N	5Hz	±0.2N	14 bit	0.122N	[1]
A3	External Temperature	Cell sensing system	Analogue Input	°C	-15°C to +25°C	-40°C[2]	+60°C[2]	1Hz	±0.1°C	14 bit	0.006°C	[1]
A4	External Temperature	Thermal Management system	Analogue Input	°C	-35°C to +35°C	-40°C[2]	+60°C[2]	1Hz	±0.05°C	14 bit	0.006°C	[1] [3]
A5	Joucomatic pressure	Output pressure of valve	Analogue Input					1Hz		14 bit		[2]
A6	Internal Temperature	Air temperature inside node	Analogue Input	°C	-15°C to 105°C	105°C[2]	-25°C[2]	1Hz	0.5°C	10 bit	0.127°C	[2]
A7	Cooling plate temp	Cooling plate temp inside node	Analogue Input	°C	-35°C to +35°C	105°C[2]	-25°C[2]	1Hz	0.5°C	10 bit	0.127°C	[2]
A8	Joucomatic Temperature	Joucomatic Temp inside node	Analogue Input	°C	-35°C to +95°C	105°C[2]	-25°C[2]	1Hz	0.5°C	10 bit	0.127°C	[2]
A9	Spare channel 1		Analogue Input	V	-	+6.2V	-6.2V	-	-	14 bit	-	
A10	Spare channel 2		Analogue Input	V	-	+6.2V	-6.2V	-	-	14 bit	-	
A11	Binary input channel 1	General purpose binary input	Binary Input	High/ Low	-	High	Low	1Hz	Binary	1bit		
A12	Binary input channel 2	General purpose binary input	Binary Input	High/ Low	-	High	Low	1Hz	Binary	1bit		
A13	Pneumatic demand	Demand signal to Joucomatic	Analogue output	mN	±400N	+400N	-400N	1Hz	0.2N	14 bit	0.061N	[1]
A14	Binary output channel 1	Output. PWM or binary modes. PWM for rad panels	Binary output	% High/ Low	0% - 100% Hi/Lo	100% High	0% Low	0.1Hz	Not given	8bit	0.4%	[1] [3]
A15	Binary output channel 2	Binary Output. For solenoid valves etc.	Binary output	High/ Low	-	High	Low	1Hz	Binary	1bit		

Table 5-13 Axial type node box signal description

Signal Reference	***Function***	***Description***	***Type***	***Units***	***Operating range***	***Max***	***Min***	***Rate***	***Required Resolution***	***Actual Resolution Bits***	***Units***	***Reference***
M1	Motor 1 demand	Output to motor channel 1. 2A per phase 4 wire	Stepper Motor drive					0.1Hz		24 bit		[1] [2]
M2	Motor 2 demand	Output to motor channel 2. 2A per phase 4 wire	Stepper Motor drive					0.1Hz		24 bit		[1] [2]
M3	Encoder 1	Input reading from encoder 1	4 wire SSI					0.1Hz		24 bit		[1] [2]
M4	Encoder 2	Input reading from encoder 2	4 wire SSI					0.1Hz		24 bit		[1] [2]
M5	Limit switches (x2) chan. 1	Input form limit switch close to limit of travel	Opto-isolated binary	on|off	on|off	-	-	Interrupt	on|off	1	on|off	
M6	Limit switches (x2) chan. 2	Input form limit switch close to limit of travel	Opto-isolated binary	on|off	on|off	-	-	Interrupt	on|off	1	on|off	
M7	Solenoid valve drive chan. 1	Output. Controls coarse cylinder shut off valve.	Opto-isolated binary	on|off	on|off	-	-	As required	on|off	1	on|off	
M8	Solenoid valve drive chan. 2	Output. Spare to control fill valve.	Opto-isolated binary	on|off	on|off	-	-	As required	on|off	1	on|off	
M9	Internal Temperature	Air temperature inside node	Analogue Input	°C	-15°C to 105°C	105°C	-25°C[2]	1Hz	0.5°C	10 bit	0.127°C	[2]
M10	Cooling plate temp	Cooling plate temp inside node	Analogue Input	°C	-35°C to +35°C[2]	105°C	-25°C[2]	1Hz	0.5°C	10 bit	0.127°C	[2]

Table 5-14 Master Cylinder node box signal description

Signal Reference	Function	Description	Type	Units	Operating range	Max	Min	Rate	Required Resolution	Actual Resolution		Reference
										Bits	Units	
L1	Lateral support load	Load Cell reading	Analogue input	N	+5000N - 100N or +100N - 5000N	+5100 N or -5100N	-100N or +100N	5Hz	±8N	14 bit	0.32N	[1] [2]
L2	Internal Temperature	Air temperature inside node	Analogue Input	°C	-15°C to 105°C	105°C[2]	-25°C[2]	1Hz[1]	0.5°C	10 bit	0.127°C	[2]
L3	Cooling plate temp	Cooling plate temp inside node	Analogue Input	°C	-35°C to +35°C[2]	105°C[2]	-25°C[2]	1Hz[1]	0.5°C	10 bit	0.127°C	[2]

Table 5-15 Lateral node box signal descriptions

Signal Reference	Function	Description	Type	Units	Operating range	Max	Min	Rate	Required Resolution	Actual Resolution		Reference
										Bits	Units	
C1	Bipod support strain	Strain gauge reading	Analogue input	N/cm^2	±70,000N/cm^2	+70,000 N/cm^2	-70,000 N/cm^2	1Hz	200N/cm^2	14 bit	8.5N/cm^2	[1] [2]
C2	Internal Temperature	Air temperature inside node	Analogue Input	°C	-15°C to 105°C	105°C[2]	-25°C[2]	1Hz[1]	0.5°C	10 bit	0.127°C	[2]
C3	Cooling plate temp	Cooling plate temp inside node	Analogue Input	°C	-35°C to +35°C[2]	105°C[2]	-25°C[2]	1Hz[1]	0.5°C	10 bit	0.127°C	[2]

Table 5-16 Cell Sensing node box signal descriptions

5.9.1.4 CAN Message Protocol

The CAN protocol to be used between the PCS IOC and the NCUs is described in the document *PCS IOC/NCU CAN Protocol, J.Maclean.* [6]

5.9.1.5 CAN bus timing parameters

In order to verify that the register programming of the C167 CAN controller registers meets the timing requirements of the CAN specification a Mathcad work sheet was produced. For details of the C167 CAN registers consult the Siemens C167 data-sheet. The worksheet verifies that the selected values are valid at the proposed bit rate of 250 kbits/s.

5.10 Interface Control Documents

The following table shows the ICDs pertaining to the PCS. Much of the information in this document defines the contents of the ICDs. Where applicable the relevant sections are shown.

ICD	Title	Description	Relevant SDD section
1.1.1/1.2.10	Telescope structure, drives and brakes to M1 control	PCS thermal enclosure to TS structure	-
1.1.11/1.2.10	Telescope control to M1 control	PCS to TCS interface	5.2.1.1
1.1.13/1.2.10	Interlock system to PCS		5.2.1.2
1.2.10	M1 control	Overview of PCS interfaces	-
1.2.10/3.1	M1 control to Observatory Control	OCS engineering screens?	-
1.2.10/3.2	M1 control to Data handling system	PCS LUTs generated by the DHS	5.2.1.5
1.2.10/3.6	M1 control to system coolant	PCS thermal enclosure coolant connection	-
1.2.10/3.8	M1 control to System cables	Cabling to/from the PCS VME chassis	-
1.2.2/1.2.10	M1 support system to M1 control	Including APSS, PSS, Cell Sensing and AO.	5.4, 5.5, 5.6, 5.7
1.2.4/1.2.10	Radiation plate system to M1 control	The TMS	5.3
1.2.5/1.2.10	Surface heating to M1 control	At present surface hearing in not included in the PCS package	-
1.2.9/1.2.10	AtmDC to M1 control	There is no interface between these systems	-

Table 5-17 PCS ICDs

6. Appendices

6.1 Appendix A - Selection of VME/CAN interface

6.1.1 Introduction

At the time CAN was selected as the serial bus for use in the PCS no EPICS/CAN interface existed. A number of options presented themselves. This section explains the reasons for selection of the eventual interface.

Both hardware and software components had to be considered as part of the total solution. the constraints upon the selection were

- Cost
- Efficient EPICS interface
- VME hardware
- Maintainability

Initial indications were that between 4 to 8 CAN bus interfaces would be required per telescope. It is now intended that 4 busses be used per telescope.

Each of the options considered are examined to explain their relative strengths and weaknesses.

The options were considered were;

Option	Hardware	Software
1	MVME162 IP carrier CPU board + TEWS TIP810 CAN interface industry packs.	DESY EPICS interface via CDI
2	ESD VME-CAN2, two channel VME interface	ESD VxWorks drivers + custom EPICS interface
3	MVME162 IP carrier CPU board + TEWS TIP810 CAN interface industry packs.	HiDEOS based custom EPICS interface
4	Greenspring ‘dumb’ industry pack carriers + TEWS TIP810 CAN interface industry packs	Custom EPICS interface and VxWorks drivers

Table 6-1 VME/CAN interface hardware options

6.1.2 Option 1

Hardware: MVME162 IP carrier CPU board + TEWS TIP810 CAN interface industry packs

Software: DESY EPICS interface via CDI

Industry packs are a modular I/O system for VME. They allow customised I/O solutions to be built for low cost. Industry packs are small ‘daughter boards’ that plug into ‘motherboards’, the motherboards are also known as carrier boards. The carrier board is a VME card either 3U or 6U in height. A 3U carrier board can take two Industry Pack modules, a 6Uboard can take four Industry pack modules. The carrier may have its own CPU or may be 'dumb' in which case it is mapped into the main CPU address space.

The industry pack modules perform the I/O. There are many types available, for example analogue input, analogue output, multiple RS232channels, SCSI interface. One of the available types is a CAN interface.

The MVME162 is a Motorola product. It is has 4 IP slots and its own 68040 CPU. The TEWS TIP810 IP is a single channel CAN interface. It is relatively simple having no CPU and a Phillips 80C200 CAN controller IC.

The cost of providing 4 CAN bus interfaces per telescope would be

Item	Cost
MVME162 (x2)	£2444
TEWS TIP810 IP (x8)	£1600
VxWorks licence (for MVME162) (x2)	£610
Total	£4654

Table 6-2 Option 1 costs

The DESY software was the first to be completed. It is run on both the MVME162 IP carrier and MVME167 IOC and required VxWorks on the MVME162. The EPICS interface was carried out via a piece of DESY software called the Common Interface Driver (CDI). This is intended to simplify the task of both producing hardware diver code and using the drivers. The CDI is a substantial piece of code that is not included in the standard EPICS release. Some CDI documentation is available but it does not appear to be either complete or detailed.

The estimated man-power to implement the required software was 4-6 weeks.

6.1.3 Option 2.

Hardware: ESD VME-CAN2, two channel VME interface

Software: ESD VxWorks drivers + custom EPICS interface

This option was considered because it was one of the few VME/CAN interface boards available at the time with VxWorks support.

The cost for 4 bus interfaces per telescope would be

Item	Cost
ESD VME-CAN2 two channel VME interface (x4)	£6400
VxWorks drivers	£1500
Total	£7900

Table 6-3 Option 2 costs

The estimated man-power to implement the required software was 6-8weeks.

The ESD board was expensive since only two channels are provided per board. The VxWorks code was also expensive.

6.1.4 Option 3

Hardware: MVME162 IP carrier CPU board + TEWS TIP810 CAN interface industry packs

Software: HiDEOS based custom EPICS interface

The hardware for this option is the same as for option 1. The software would be based on HiDEOS, a distributed operating system written to enable the use of MVME162 and Industry Packs with EPICS. A number of interfaces have been written with HiDEOS although a CAN interface is not one of them. At the time the decision was made, HiDEOS was not part of the standard EPICS release, although subsequently it has become so.

The hardware cost for 4 bus interfaces per telescope would be

Item	Cost
MVME162 (x2)	£2444
TEWS TIP810 IP (x8)	£1600
Total	£4044

Table 6-4 Option 3 costs

HiDEOS was written using C++ and requires that drivers written to interface with it use C++. The estimated man-power to implement the required software was 4-6 weeks.

6.1.5 Option 4

Hardware: Greenspring 'dumb' industry pack carriers + TEWS TIP810 CAN interface industry packs

Software: Custom EPICS interface and VxWorks drivers

Two types of 'dumb' IP carriers are marketed by Greenspring, a 2 IP slot and a 4 IP slot version. These are 'dumb' in the sense that they have no CPU but simply provide a VME bus interface so that the I/O they carry may be mapped into the address space.

In this application the lack of a CPU is not felt to be too important. Once the CAN controller ICs have been initialised most of the work a CPU has to do to interface to the CAN bus is simply reading and writing from buffers. An additional CPU would probably be of little assistance in this case, since data would have to be written and read by the IOC CPU to the bus interface card in any case.

VxWorks drivers must be provided to interface to EPICS. The estimated man-power to implement the required software was 6-8weeks.

This solution (using a 2 slot IP carrier) was adopted for the UKIRT mirror support upgrade project. This presented the possibility of sharing software development costs.

The hardware cost for 4 bus interfaces per telescope would be

Item	Cost
Greenspring 4 slot IP carrier (x2)	£660
TEWS TIP810 IP (x8)	£1600
Total	£2260

Table 6-5 Option 4 costs

6.1.6 Summary

The implementation costs for each option are shown below

Option	Hardware cost	Software cost
1	£4654	4-6 weeks
2	£7900	6-8 weeks
3	£4044	4-6 weeks
4	£2260	6-8 weeks

Table 6-6 VME/CAN interface costs, summary of options

In terms of providing a CAN interface all options use a Phillips CAN controller so are very similar solutions at that level.

Hardware cost is important not only for this work package. It should be born in mind that other packages (i.e. the MCS) and projects will be using this solution.

Option 1 was rejected principally because of concerns about the ease of use and maintainability of the CDI. For this application the CDI is a complication that will increase system complexity where it is not required.

Option 2 was rejected because of the high hardware cost.

Option 3 was considered very carefully and is second in the list of preferred options. It was rejected because HiDEOS was not then part of the standard EPICS release and because it requires the use of C++ which is not a standard Gemini language (and would require *another* learning curve to be ascended). Also, there was concern over the efficiency of a HiDEOS implementation.

Option 4 was the option selected. It has the lowest hardware cost and although it required custom software this development effort could be shared with the UKIRT group. It also meant that the software (and hardware) would be running on UKIRT for at least 18 months before the PCS was integrated with the M1 cell. The only concern is over the load this implementation puts on the IOC CPU. Initial tests suggest this will be at an acceptable level. More detailed tests are planned.

The way the software has been implemented means that if necessary the code can be run on a MVME162 using the same IPs. This means that if CPU power does become an issue the PCS could be divided into two IOCs in the same chassis. For example, an MVME162, with CAN IPs, could run an EPICS database that takes care of all the I/O processing, while a MVME167 could run an EPICS system that interfaces to the TCS and does all the ‘number crunching’. The two would simply have to pass commands and status to each other.

6.2 Appendix B - Selection of Node Control Unit

6.2.1 Selection of Node CPU

This section summarises the process undertaken to select the Node Control Boards (NCBs). The process started by defining the requirements of the NCB. The market was then searched to find an off-the-shelf solution. Unfortunately no such solution could be found so the possibility of designing a custom board was investigated. This appeared feasible, the biggest issue being selection of a micro- controller. The most promising micro-controller solutions were investigated and a device selected.

6.2.2 Requirements

In summary the requirements of the NCB were.

- To Communicate with off-board memory mapped I/O
- To communicate via CAN bus. Average bus loading 20% at 250kbits/second. Up to 20 messages per second to or from the node.
- To communicate with off-board peripherals via I^2C bus.
- Ability to read values from an external ADC at up to 250Hz
- Ability to average ADC readings.
- Ability to run PID loop around load cell and proportional (Joucomatic) valve at 20Hz[*].
- Programmable in ‘C’
- Good development environment
- On board, galvananically isolated CAN bus driver.
- On board RS232 buffer and interface.

- Minimum 32Kbytes RAM
- Minimum 64Kbytes Program memory. FLASH memory with on-board programmability preferred.
- Temperature range; -25°C to +85°C
- Cost less than £140 per node

* This requirement was only a 'possible' until very late in the decision process.

6.2.3 Off-the-shelf Solutions

A number of solutions were looked at. All were very similar, the most promising were;

- Phytec; mini-module 592
- Stzp; CAN multi-modul592

These boards are very similar. They are both 'credit card' sized, both use the Phillips 80C592 micro-controller, have up to 64K RAM and EPROM and cost in the region of £130.

Both had similar drawbacks. The most serious of these being the lack of galvanic isolation of the CAN bus drivers. Another major deficiency was the lack of FLASH memory.

Another concern was that both are manufactured in Germany and priced in Deuchmarks. With the lack of stability in the £/DM exchange rate there is the real possibility that the budget price will be exceeded by the time we have to purchase the production quantities.

For these reasons and after some unsuccessful experimentation with some Phytec mini-module 592 boards it was decided to investigate the possibility of producing a custom design.

6.2.4 Custom solution

Given the large number of boards required (350+, large in a telescope context) this is a situation that could justify a custom solution. Initial costings suggested we could achieve our requirements within time and budget. The largest issue to be resolved was the question of micro-controller and CAN controller selection.

CAN controller ICs are available from many vendors. They can generally be classified as first or second generation devices. Both generations are compatible with the CAN standard. The second generation devices tend to be much more efficient to use in terms of CPU time.

The first issue to resolve was whether to use a micro-controller with an on-board CAN controller or a separate CAN controller. Using a separate CAN controller offered a larger range of micro-controllers and CAN controllers. It would then be possible to use a micro-controller available from several suppliers (such as an 80C32). Since the CAN controller could not be second sourced this was not a real advantage.

An external CAN controller is much less efficient in terms of CPU time. An on-board CAN controller uses dual port memory to exchange data with the CPU, an external device requires the CPU to perform external bus accesses to the

CAN controller. Other advantages of an onboard CAN controller are; less external connections, greater reliability, lower assembly costs and lower power dissipation.

Therefore the decision was taken to use a micro-controller with an integrated CAN controller.

Three such micro-controllers are generally available;

- Motorola 68HC05X: An 8-bit device with a first generation CAN controller.
- Phillips 80C592: An 8-bit device. Essentially an 80C51 with a Phillips 80C200 CAN controller (a first generation CAN controller.
- Siemens C167: A 16-bit device with a second generation CAN controller.

It soon became apparent that the Motorola device was suffering supply problems and that technical support from Motorola was poor. The Motorola device was therefore rejected.

The factors considered in choosing between the Philips and Siemens device are shown in the matrix below. A '+' means that the device is superior to the device with a '-'.

Factor	Phillips 80C592	Siemens C167
Device cost	£15.58	£34.38
Completed board cost	£110	£120
Development kit availability	+	+
Development kit cost	£5800	£9200
CPU power	-	+
CAN Controller	-	+
Ease of development	-	+
Dual source supply	-	+
Integrated peripherals	-	+

Table 6-7 Comparison of 80C592 and C167

The completed board cost for both devices should be within budget. The development kit cost for the C167 exceeds the budget amount for this line item which was based on that for a 80C592. However because of savings in other areas this cost can be absorbed within the overall budget.

The C167 has approximately 12 times the processing power of the 80C592. Rough calculations suggest that up to 40% of the 80C592 CPU time would be absorbed before the requirement for the PID loop are included. It is likely that much of the remaining CPU power would be taken up when running this. The 80C592 would be uncomfortably close to using all its CPU. This would leave little room for expansion and could cause unpredictable behaviour if the CPU were hit with a cluster of interrupts in a short time.

The C167 has a more advanced CAN controller than the 80C592. This will make programming simpler, reduce the load on the CPU and reduce the risk of messages being overwritten between being received and used.

Software development is likely to be easier with the C167. This device was designed to support programming with high level languages and has a much simpler memory map. The 80C592 is not suited to programming with ‘C’ and has a more complex memory map.

The 80C592 is a very popular device and is not likely to be withdrawn but it is only manufactured by Phillips. The C167 is second sourced by SGS-Thomson and Siemens claim it will be in production for at least ten years due to contractual obligations with Bosch.

Both devices are well featured with integrated peripherals with the C167 perhaps having the advantage in terms of useful features.

6.2.5 Summary of Selection Process

A search for an off-the shelf solution did not yield a suitable product. Given the quantity involved a custom solution is viable. Selection of a micro-controller and CAN controller was seen as the critical issue in a custom design.

An integrated on-board CAN controller was chosen as the preferred solution. Of the devices available the Siemens C167 was chosen as the best long term solution. The initial cost is higher but can be accomplished within the overall budget. Simpler software development and room for expansion suggest this option will have the lower life-cycle cost.

6.2.6 Development

A C167 evaluation kit was procured. This consisted of an evaluation board and a ‘lite’ version of the ‘C’ compiler. This showed that the device, including the CAN controller, was fairly straight forward to use.

A paper design was commissioned from a Cambridge company with CAN experience to get a more accurate costing. This showed the cost per board will be in the region of £120 pounds. An order for 3 prototype boards based on this design has now been placed with delivery expected in early December. An In-Circuit-Emulator has also been purchased with delivery expected in late November.

The next stage will be to evaluate the prototype boards. Once this is complete it is intended to purchase 12 pre-production boards that will incorporate any changes required as a result of the prototype evaluation. These boards should be identical to the production version and will be used in combination with prototype motherboards for tests on the test rig.

6.3 Prototype EPICS I/O sub-system

A very simple prototype EPICS system has been constructed to examine ways of dealing with the large number of I/O records that will be required.

Some experimentation has been done to find a way of handling the large number of EPICS records that will be required in the PCS. One of the big problems is naming so many records and then assigning CAN ‘addresses’ to them. The best solution found is to use Capfast to produce a ‘node box’ symbol that contains all the I/O records associated with a node box. A text substitution facility of Capfast is used to generate unique record names for the records in each instance of the node box symbol. A ‘C’ program was written to post process the EPICS short report file, inserting the correct CAN ‘address’ for each record. This has been found to work well.

Some experimentation has also been done with the EPICS display managers. Again handling the large number of signals in the PCS is a potential problem. Very simple PCS demo displays have been produced with both EDD2/DM2 and MEDM. Of the two EDD2/DM2 is felt to be the best option because of its greater functionality. MEDM does produce the “nicer” displays however.

The prototype now consists of an example engineering display which is connected to a small EPICS database. The database is constructed around a small number of node control units. The I/O records are connected to a real CAN bus. Since we do not yet have any real NCUs, these are simulated using a CAN development tool running on a PC.

It is intended to give a small demonstration of this system at the PDR.