LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY
- LIGO -
CALIFORNIA INSTITUTE OF TECHNOLOGY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Technical Note	**LIGO-T950037-00 -**	**Cxx**	06/28/95

Pre-Stabilized Laser Control and Data System Software Design Document

R. Bork

Distribution of this draft:

California Institute of Technology
LIGO Project - MS 102-33
Pasadena CA 91125
Phone (818) 395-2966
Fax (818) 304-9834
E-mail: info@ligo.caltech.edu
WWW: http://www.ligo.caltech.edu

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

1 INTRODUCTION

1.1. Purpose

The purpose of this paper is to document the design for the control software developed for the Pre-Stabilized Laser (PSL) subsytem of the LIGO interferometers.

1.2. Scope

This design document covers all control and monitoring aspects for the PSL, including:

- Realtime Processing
- Operator Interfaces
- Alarm Management
- Slow Data Archival (<=10Hz)

This document does not cover what the CDS overall specification calls out as 'Remote Diagnostics' or 'Data Acquisition'.

1.3. Definitions, Acronyms, and Abbreviations

1.3.1. Definitions

1.3.1.1 Physical Input/Output Channels

Physical I/O channels are defined as those software data channels which directly receive data from / send data to VME I/O module channels. The point of I/O reference is the control system software.

1.3.1.2 Virtual Input/Output Channels

Virtual I/O channels are ones which strictly communicate data between software modules.

1.3.1.3 CDS Diagnostic Data Channel

A CDS diagnostic data channel is one which is not required for operation of the system, other than to provide diagnostic information to allow verification of proper system operation.

1.3.2. Acronyms

1. ADC. . . . Analog to Digital Converter
2. AOM . . . Acousto-Optic Modulator
3. CDS . . . . Control and Data System
4. CIM . . . . Computer Integrated Manufacturing
5. DAC. . . . Digital to Analog Convertor
6. EPICS . . Experimental Physica and Industrial Control System
7. FDR . . . . Final Design Review
8. FSS. . . . . Frequency Stabilization Servo

9. I/O . . . . . Input/Output
10. LANL. . . Los Alamos National Laboratory
11. LCU . . . . Logical Control Unit
12. LLA . . . . Laser Loop Amplifier
13. LS. . . . . . Laser Steering
14. MEDM. . Motif EPICS Display Manager
15. PC . . . . . Pockels Cell
16. PDR . . . . Preliminary Design Review
17. PM . . . . . Phase Modulation
18. PSA . . . . Power Stabilization Amplifier
19. PSL . . . . Pre-Stabilized Laser
20. PSS. . . . . Power Stabilization Servo
21. RAM . . . Random Access Memory
22. RF . . . . . Radio Frequency
23. SNL . . . . State Notation Language
24. TBD . . . . To Be Determined
25. VME . . . Versa Modular Eurocard

1.4. References

1. Prestabilized Laser Control Requirements LIGO-T950001-1-C
2. PSL CDS Software Requirment Specification LIGO-T950022.
3. EPICS User's Manual, LANL

1.5. Overview

This document is intended, to the extent possible, to follow the same organizational layout and section numbering as the PSL CDS Software Requirement Specification LIGO-T950022. This is done to assist in cross referencing and matching designs with the design requirements.

1.6. Document Relationship

This design document is one of several requirements specification, design, and test documents developed for the PSL CDS. An outline of the these pertinent documents and the relationship of this document is outlined in Figure 1: PSL CDS Document Tree.

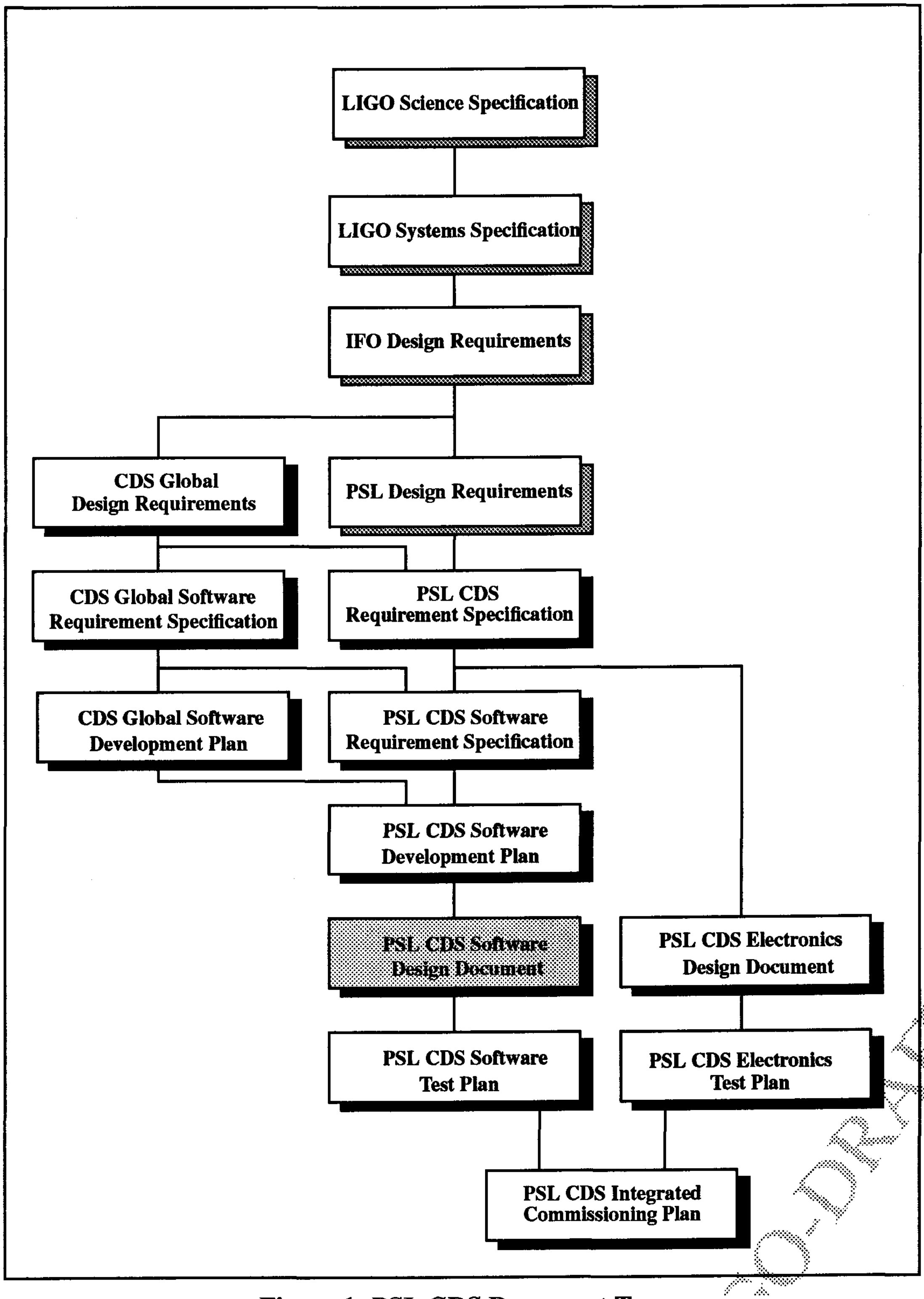


Figure 1: PSL CDS Document Tree

2 GENERAL DESCRIPTION

2.1. Product Perspective

A basic block diagram of the software architecture is shown in the following figure. The software specified in this document is primarily that real-time control software which resides in the PSL VME processor. The PSL is a subsystem of an interferometer, which is the scientific instrument of the LIGO project. Also described are the operator interfaces specific to the PSL and interaction with the LIGO alarm manager and data archiver.

2.2. Product Functions

The primary function of this software is to provide real-time control and monitoring capabilities for the PSL within the overall framework of the LIGO Control and Data System (CDS). In Version 0 of the software (required to be complete by the time of the PSL Final Design Review), the software will be expected to provide, at minimum, Remote Manual operation capabilities. This is defined as having all the necessary capabilities to provide for remote operation and monitoring of the PSL via operator displays running on a UNIX workstation. Prior to start of LIGO commissioning at the various sites, this PSL software is intended to be extended to provide more automatic functionality as prescribed in this document. This will include such items as automatic startup and shutdown of the PSL subsystem.

2.3. Evolution

This document is intended to evolve with the design phases of the project. This evolution will be marked by the revision number of the document. To maintain a history, the revision number shall be:

1, for conceptual design

2, for final design

3 and above for as-built designs

Though later revisions will be updates and changes to this base document, a version will be locked and maintained in LIGO document control for each revision number, thereby allowing comparison to previous releases.

2.4. General Constraints

In keeping with the general guidelines for LIGO CDS, this software is developed within the context of the Experimental Physics and Industrial Control System (EPICS), a tool set developed by Los Alamos National Laboratory (LANL) and other research facilities.

EPICS is a set of tools and runtime functions to ease and enhance the development of real-time controls. Since this is the first LIGO CDS software design, a brief description of EPICS is given in the following sections for document completeness.

2.4.1. EPICS Core

The core consists of a database comprised of pre-defined 'records', which provide a storage of data as well as real-time functions. Approximately 20 pre-defined record types exist, such as analog input/output, binary input/output, calculation, waveforms, etc. Each record type has its own set of 'fields' to store data, including the channel (signal) name, present value, hardware I/O connection, when it should process, alarm limits, display limits, etc. The fields of these records are defined off-line by the programmer/user, then downloaded and run on VME based microprocessors. VxWorks is the operating system on which it is built, therefore, any VME processor which VxWorks supports, EPICS will run on. For this project, it will be a Motorola 68040.

On the VME processor, the records will process as defined by the programmer when the fields were defined. The record can be processed at a standard scan rate (10Hz to .1Hz), on a user defined event or hardware interrupt, or 'passive'. A passive record will process only when data is sent to the record, such as when an operator enters a new setpoint, when caused to process by another record, as defined by the programmer, or when other user provided software writes to the record.

Actual processing of a record depends on the record type. For example, an analog input record will get data from the defined ADC channel, convert it to engineering units, check alarm and range limits, and cause any linked records to process.

The second part of the 'core' is Channel Access (CA), which provides automatic networking communications. Every record which is defined becomes automatically accessible by any other computer on the network through the use of the record (channel) name. CA uses a discovery protocol, ie one need not know which processor has the information (there will be many processors on the LIGO CDS net), only the name of the channel and the data field to be read. When a user process (or an extension tool, such as the EPICS display manager) wants data, a request is made to CA with the channel name and when the data is desired (immediately, whenever the value changes, 10Hz, etc.). CA will then broadcast a message to all processors requesting that data. The processor which has that data will respond, and a direct socket will be setup between the requester (CA client) and the CA server on the data provider to actually transfer data.

The last part of the core is the 'sequencer'. Stringing together EPICS records to provide complex or high speed (EPICS records take 80 to 300 usec each to process) control algorithms can be difficult (or impossible), therefore EPICS provides some standard scripts to allow a programmer to write sequences in 'C' and connect with the EPICS database. Using the EPICS State Notation Language (SNL) scripts, custom C codes will be developed to handle most of the designs of the PSL software. This code is pre-compiled by EPICS into a '.c' file, then cross-compiled with VxWorks to run the the VME processors.

2.4.2. EPICS Extensions

EPICS extensions are those programs which run on UNIX, and either provide EPICS editing or runtime monitoring capabilities.

Editors for creating a database are:

1. DCT: An ascii file creator.
2. GDCT: A graphical ascii database tool.

3. CAPFAST: A commercial schematic capture package.
4. Access: Commercial relational database (Later to be replaced by Oracle).

The software interfaces which allow the use of Access were developed at LIGO, and this is the tool of choice for developing EPICS databases for LIGO projects. The primary reason is that the first three tools do not provide all of the data sorting, querying and report generation capabilities of a true database, nor do they allow for entering of data not directly needed by EPICS to run, such as cable runs, equipment locations, and other data associated with data channels.

Two extensions for EPICS exist to create and run operator interfaces:
1. edd/dm: Original EPICS display manager, written in X intrinsics
2. medm: Motif version developed at Argonne National Lab (ANL).

Medm will be used for LIGO developments.

Other extensions provide EPICS diagnostic capabilities and interfaces to other software packages. These include:
1. Probe: Allows dynamic connection to single channel for diagnostics
2. DP: Provides a text page layout of channels which can be dynamically defined at runtime
3. ALH: Provides monitoring, display and logging of alarms
4. AR/ARR: Provide archiving and retrieval of data to/from mass storage.
5. BURT: Back-up and restore tool.
6. CaWave: Interface routines to PvWave software.
7. Km: Allows connection of a knob box to Medm.

3 DESIGN SPECIFICS

3.1. Introduction

3.1.1. Design Layout

3.1.1.1 CIM Model

The PSL CDS software design, as with the PSL CDS software requirements, follows a CIM model, as shown in the following figure. The shaded area is the portion of software to be described in this document.

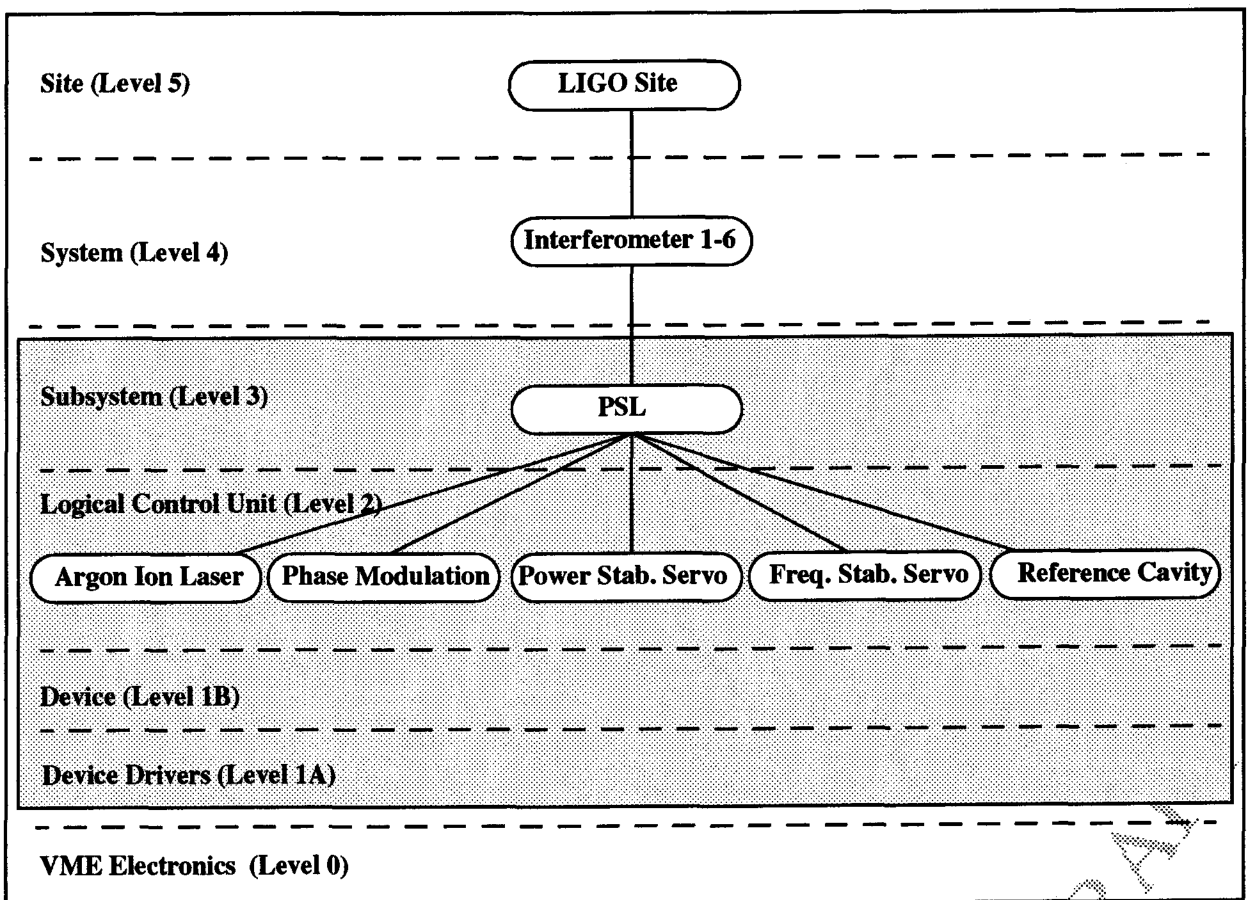


Figure 2: PSL CIM Model

3.1.1.2 Software Architecture

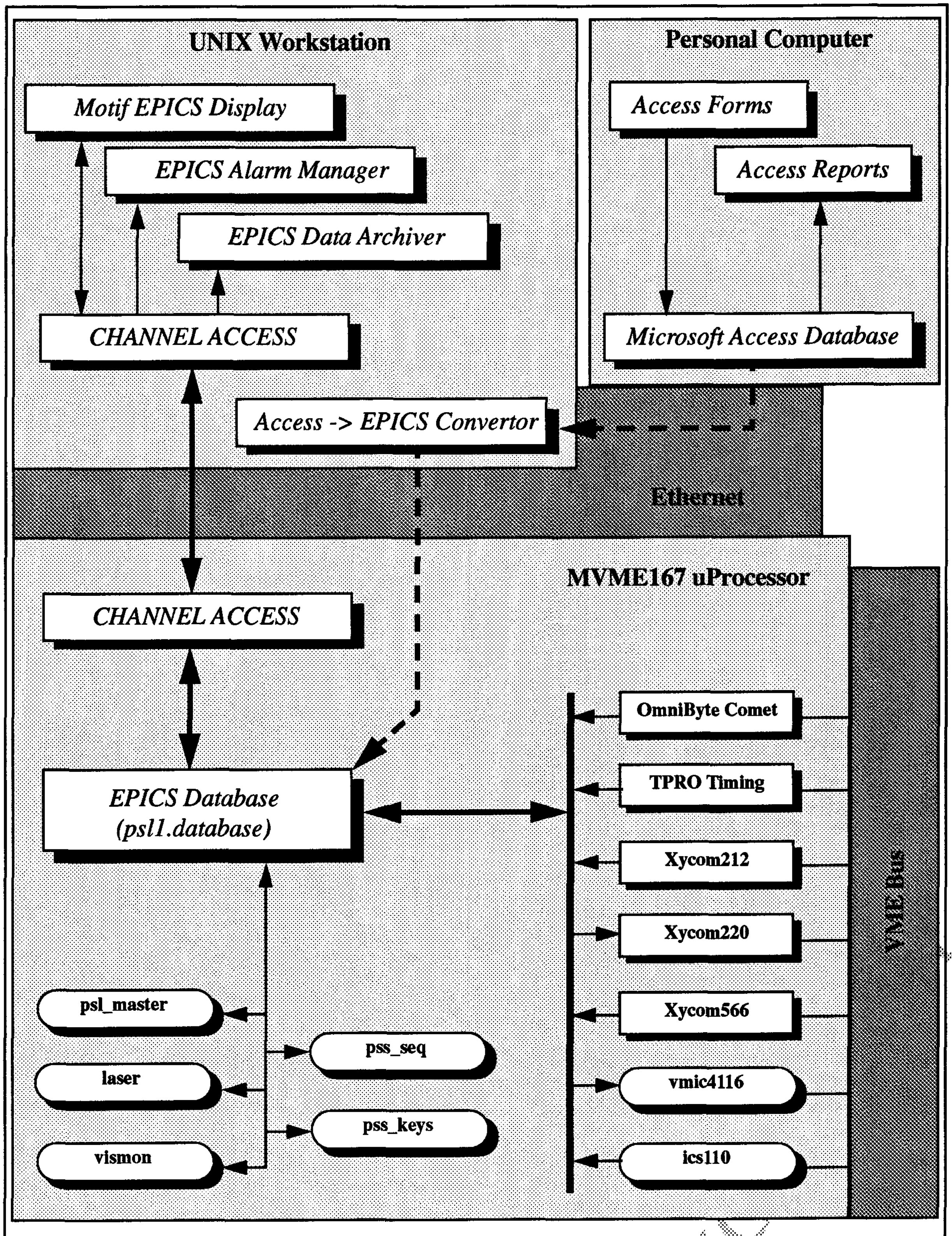


Figure 3: PSL Software Architecture

3.1.2. Design Diagrams

In sections which follow, code design diagrams employ a combination of EPICS records and state diagrams for embedded C code. These diagrams follow a standard as shown in the following figure.

EPICS records are shown as rectangles, which contain the EPICS record name, type and short description. Embedded EPICS State Notation Language (SNL) and C code are enclosed in a shaded region. Code actions are shown within ovals. If a transition rule is to be met before the next action occurs, a horizontal line appears in the path, and the rule is listed.

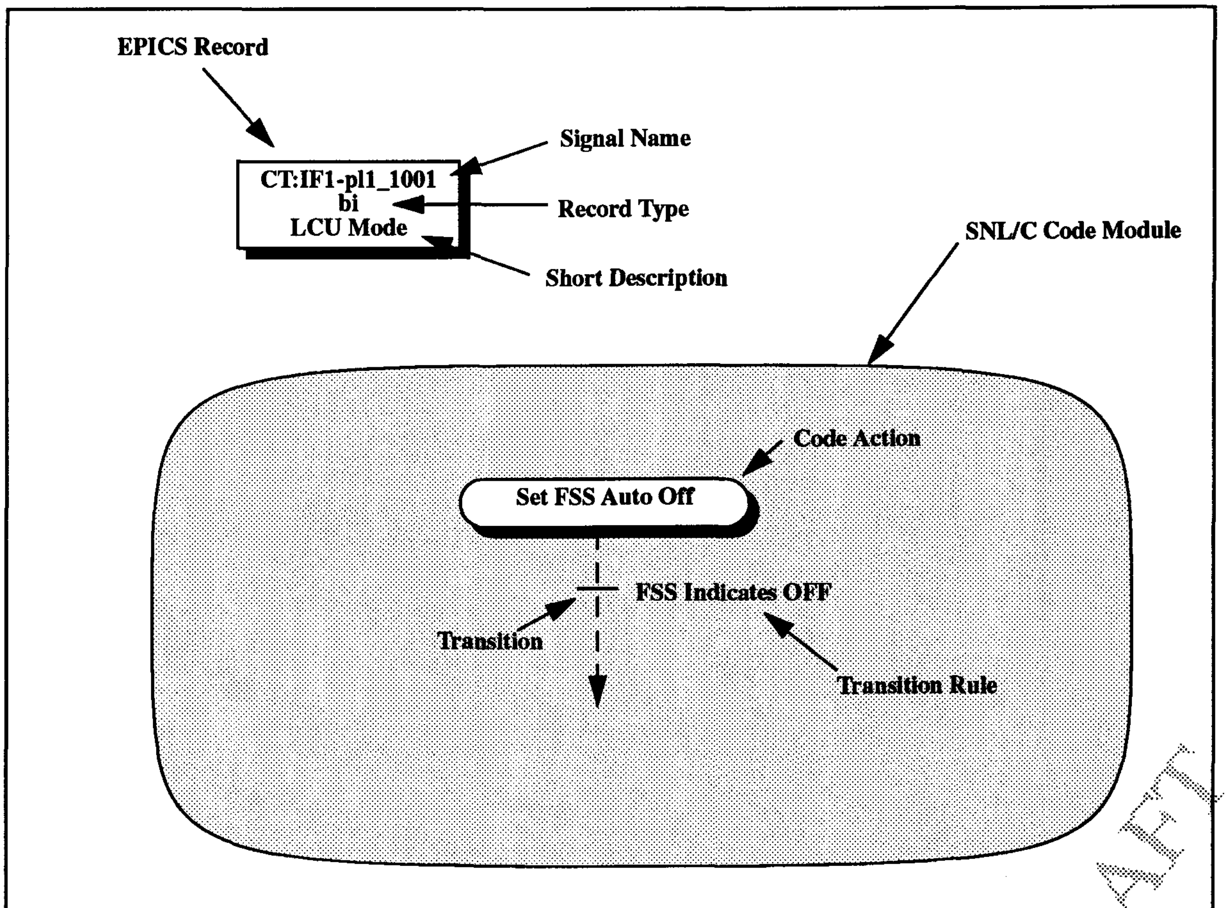


Figure 4: Design Diagram Legend

3.1.3. Directory Tree

In keeping with the LIGO CDS standard, PSL software exists in three directory areas, the specific top directory depending on the state of code development. These areas are:

1. Development

2. Prototype
3. Release

Within each of these areas, there exists a tree structure, as shown in Figure 4: PSL CDS Software Directory Structure. The prototype area (for the 40 Meter prototype) top directory is /home/CDS/a/M40 on kater. The release area for LIGO is ***TBD***.

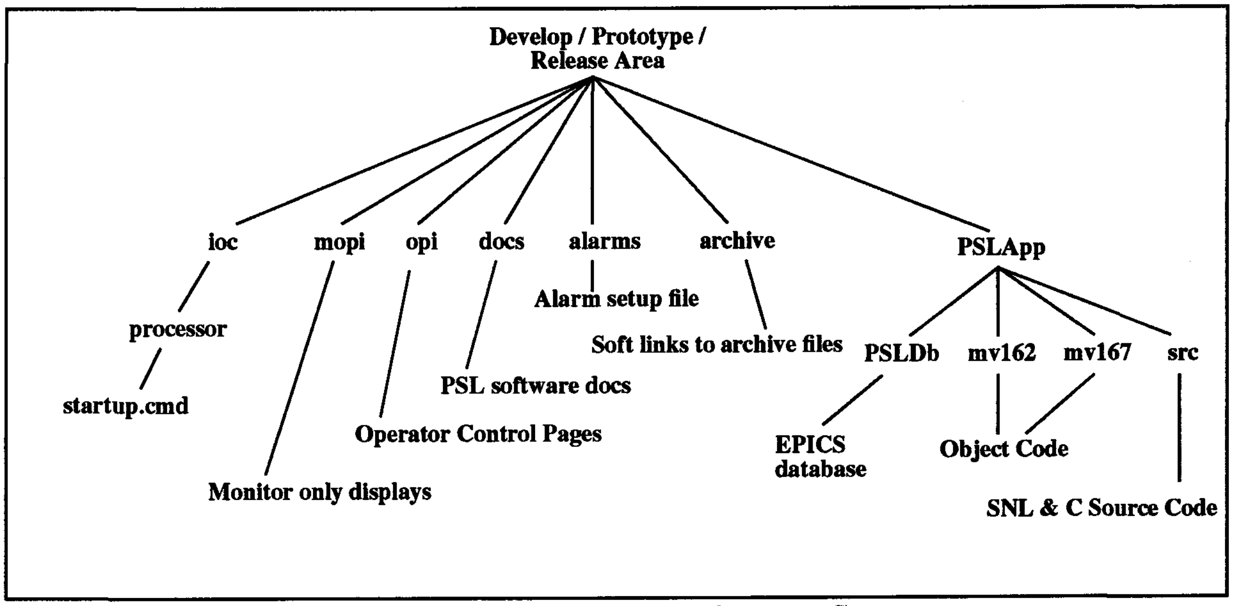


Figure 5: PSL CDS Software Directory Structure

3.1.3.1 ioc Subdirectory

The ioc subdirectory contains another directory with the name of the processor which is to run the PSL real-time code (scipe1 for the 40 Meter prototype). This directory contains a startup.cmd file, which the 68040 processor will run automatically when it boots. This file contains:

1. Paths to VxWorks
2. Paths to EPICS
3. Commands to load EPICS
4. Commands to load and run the EPICS database files
5. Commands to load and run the PSL SNL and C codes

This file, and its execution by the VME microprocessor, allows the PSL CDS VME system to automatically come on line whenever the VME crate is powered up or rebooted. This process takes approximately 30 seconds to complete.

3.1.3.2 PSLApp Subdirectory

The PSLApp directory contains all of the code developed for the PSL in four subdirectories:

1. PSLDb: Contains the ascii and compiled versions of the EPICS database.
2. src: Contains all source code (SNL and C) developed for the PSL sequencers, drivers, etc.
3. mv162/mv167: Contain the object code compiled from src for the two supported target pro-

cessors (MVME162 and MVME167). For the 40 Meter prototype, the target is an MVME167, but may later be an MVME162. Both are Motorola 68040 based modules, the latter being a less expensive version, with less RAM, slower clock, and no hardware floating point.

3.2. General Design Layouts

This section describes software designs and standards common to the PSL CDS software

3.2.1. Physical Input/Output (I/O)

All realtime control data interfaces across the CDS network via the EPICS Channel Access protocols. Certain blocks of digitized/waveform data destined for disk storage for passing to later analysis software is transferred by custom C code using NFS.

For communication to VME modules, built-in EPICS drivers are used, except for the ICS-110A module. For the latter, custom code is used due to performance requirements (see section xxxx). All physical I/O channels to/from VME are listed in Appendix B.

3.2.2. General Processing

3.2.2.1 Data Conversions

Most data conversions from raw data to engineering units take place within EPICS database records. For data coming from the ICS-110A 16bit ADC module, this conversion takes place within the ICS-110A driver (see section xxxx).

3.2.2.2 Automatic sequencing/Closed loop control

All automatic sequencing and closed loop control code is developed using the EPICS State Notation Language (SNL), with embedded custom C code. In most cases, SNL will be used more for its built-in ability to allow communication with EPICS records, and therefore Channel Access for network communication, rather than for its sequencing features.

3.2.2.3 Alarms

PSL alarms are displayed and archived via the EPICS alarm handler (ALH). The PSL alarms show up in the alarm tree below the site and respective interferometer. The layout is shown in Figure 6: PSL Alarm Tree, which uses the 40 meter prototype as an example.

Below the PSL part of the alarm tree are listed the individual channels which can alarm and their alarm states. Beside each signal name is an icon labelled with a 'G'. Selecting this icon brings up a help window (as shown), with guidance information, such as what the channel is, what drawings does it appear on, what should the operator do to take corrective action, etc.

This program is started by the operator via an icon on the PSL Subsystem display. Once started, the ALH program operates continuously, anunciating and logging alarms (when they occur/clear and when they were acknowledged by an operator). The file from which ALH gets its information necessary to monitor alarms is contained under the alarms directory as 'psl.alhConfig'.

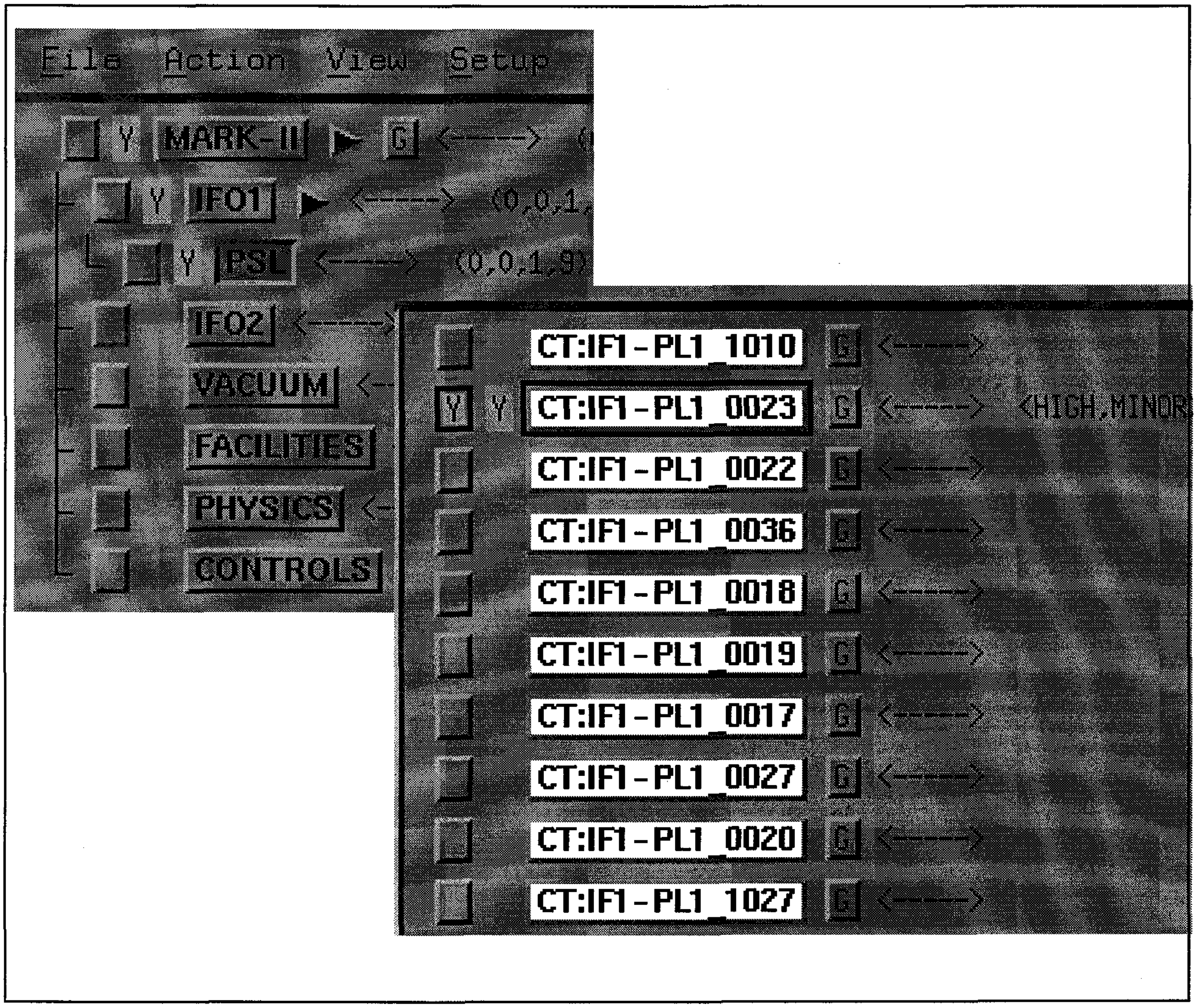


Figure 6: PSL Alarm Tree

3.2.2.4 Data Archival

Only a limited amount of PSL data gets archived on a continuous basis. These data channels are listed in Appendix C. The data archiver is started via operator selection of an icon on the PSL Subsystem display, and runs continuously thereafter.

3.2.3. Operator Displays

The various designs for operator displays are shown in Appendix A. These are screen dumps from the prototype development work.

For development of operator displays, certain standards are incorporated, as listed in the following paragraphs.

3.2.3.1 Colors

Defined color standards on operator displays:

Red: Alarm/abnormal state indication

Yellow: Warning/abnormal state indication

Green: Normal state indication

Black background in text/digital fields w/white lettering: Operator settable control values

White background in text/digital fields w/black alphanumerics: Monitor/readback signals

Blue background w/white text: Labels and CDS software messages

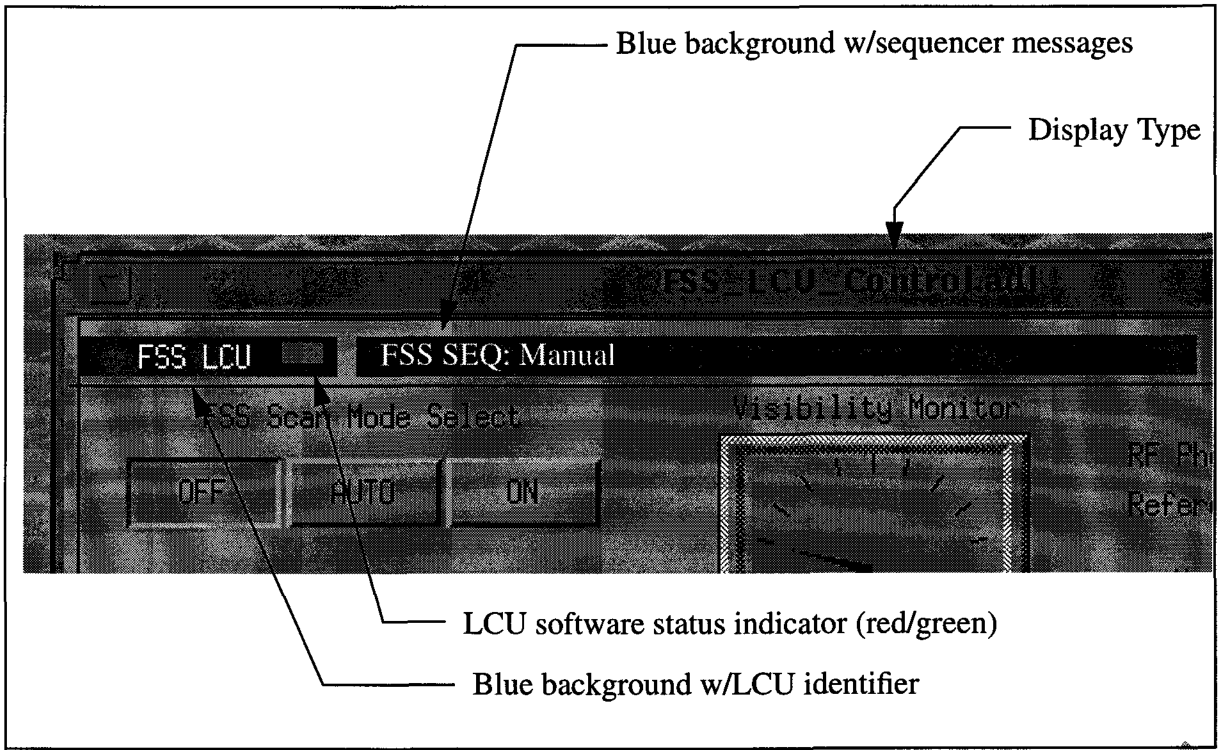


Figure 7: Common Display Features

3.2.3.2 Xterm Banner

The window banner contains the name of the display file. The name ends with the display type: Control or Monitor. The display types look almost identical, but those which contain 'Monitor' in the name do not allow interactive control action. This is a temporary security issue solution, by allowing open access to 'Monitor' files, but only authorized group access to 'Control' files.

3.2.3.3 Display Area Banner

Within the Xterm window, displays contain another blue background banner, with the name of the Logical Control Unit (LCU), the status of any automatic sequencing/closed loop control algorithms associated with the LCU, and a message area for display of text messages from the automatic control software.

3.3. PSL Subsystem Software

3.3.1. Introduction

PSL subsystem software refers to that software defined within level 3 of the CIM model, as shown in Figure 3: PSL CIM Model. This level performs those functions which coordinate the activities of the PSL as a whole, or otherwise cross LCU boundaries.

3.3.2. Normal Operation (Real-time)

In normal operation, the PSL Subsystem software must provide an automatic means to startup/shutdown the PSL subsystem. As shown later, the software associated with each LCU contains the sequences to startup/shutdown their own equipment. Therefore, the PSL subsystem software just orchestrates the LCU sequences from above.

The basic PSL subsystem sequencer is shown in Figure 8: PSL Subsystem Sequencer. The only PSL subsystem level record associated with the sequence is the PSL Mode Select. The other records shown, plus those monitored to transition through the sequences, belong to the various PSL LCU.

The PSL Mode Select has Manual/Auto Off/Auto On positions. In Manual, the sequencer is disabled, and setpoints/operations are manually selected by the operator. In Auto Off/Auto On, the laser, power stabilization servo and frequency stabilization servo are sequenced Off/OnWhen in Auto On, the sequencer:

1. Checks the state of the various LCU to determine a starting point for the sequence. Normally, this is from the Laser Off state, but the sequencer is designed such that it can start/stop the sequence from any point. The following steps assume startup from Laser Off.
2. Commands the laser LCU ON by setting the Laser LCU Mode to AUTO and Laser On/Off Select to ON. The transition to the next step is indication that the laser is On and warmed up to maximum power. This is handled by the Laser sequencer (described later).
3. Sets the PSS Mode to Auto. This causes the PSS sequencer to power lock the laser. Transition to the next step is indication of power lock and pss sequencer done.
4. Sets the FSS Scan Mode to AUTO and LLA Gain Select to AUTO. This causes the FSS sequencer to automatically scan and frequency lock the laser.
5. When the FSS indicates lock, the PSL Subsystem sequencer will reset the PSL Mode Select to Manual.

When set to Auto Off, the sequencer basically reverses the procedure, turning off the FSS, then PSS, and finally the laser. Once it gets an indication that the laser is off, it again sets the PSL Mode Select to Manual.

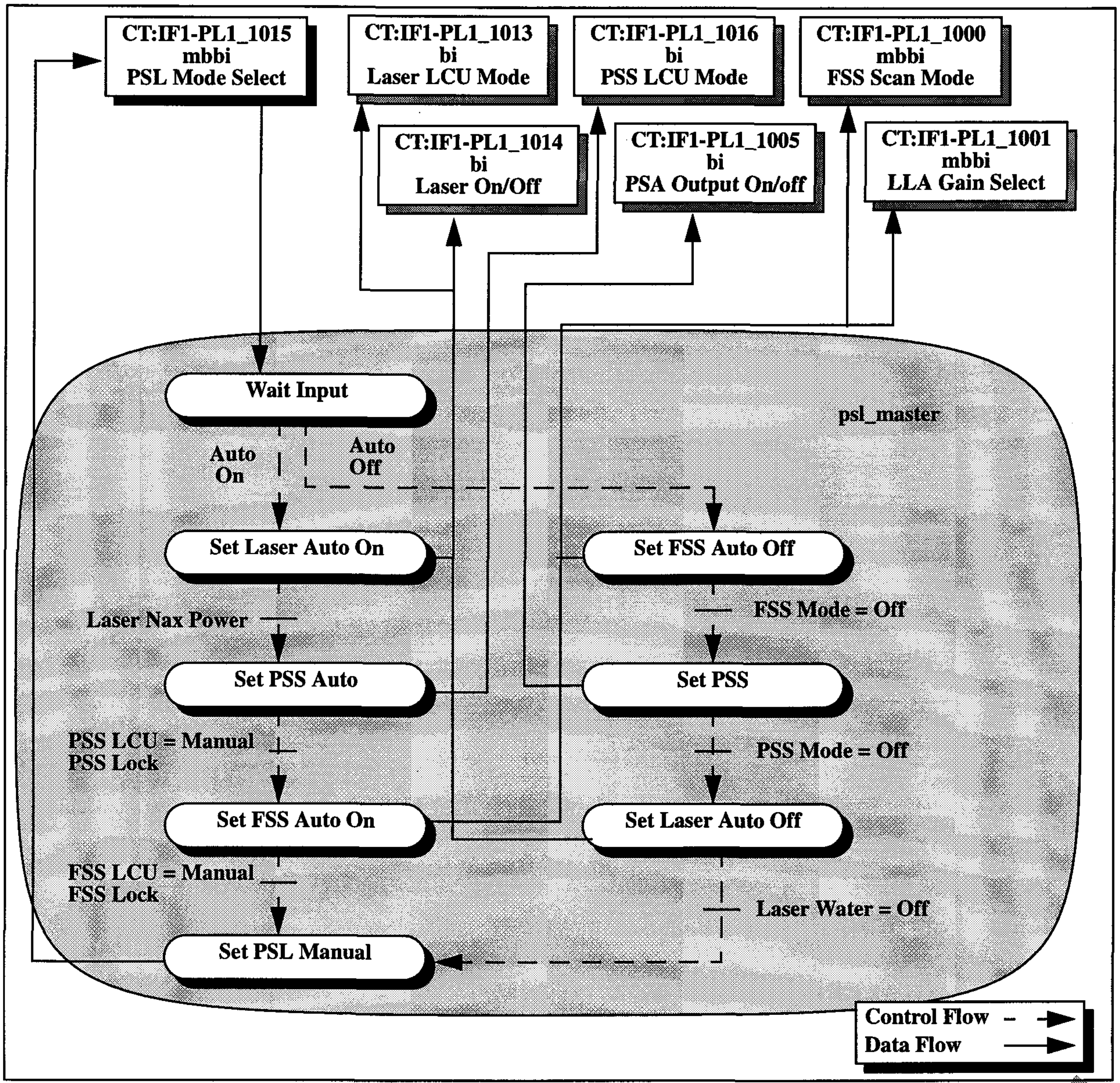


Figure 8: PSL Subsystem Sequencer

3.3.3. Diagnostics (Real-time)

3.3.3.1 VME Crate Diagnostics

This level of software is responsible for monitoring and reporting the condition of the electronic hardware associated with the PSL VME crate. Unfortunately, EPICS does not have operator friendly diagnostics for this purpose, although it does have some scripts which may be run by logging into the VME processor. For LIGO designs, dedicated software and operator displays will be developed to show the system status at a glance without having to know the EPICS scripts and VxWorks operating system calls.

For the PSL, this VME diagnostic software design is shown in Figure 9: VME Diagnostic Software. Diagnostics are limited to ensuring that the proper modules are installed and responding on the VME backplane. Without additional equipment, the VME modules do not produce diagnostics to ensure that internally they are operating properly. Within the VME processor, software runs which checks the VME module by checking the module ID (if provided by manufacturer) and ensures no bus error occurs when reading/writing module (indicating that module is installed and responding). The status information is then set in a binary EPICS record for each module, and passed on to the operator display. The operator display has a picture of the VME crate, with each module highlighted in red or green, indicating the status. This display will evolve in time to show the various VME crates associated with an entire interferometer.

For LIGO designed modules within the VME crate, diagnostics are run by the LCU software, described later in this document. The results will be displayed in this operator screen, same as the other VME modules.

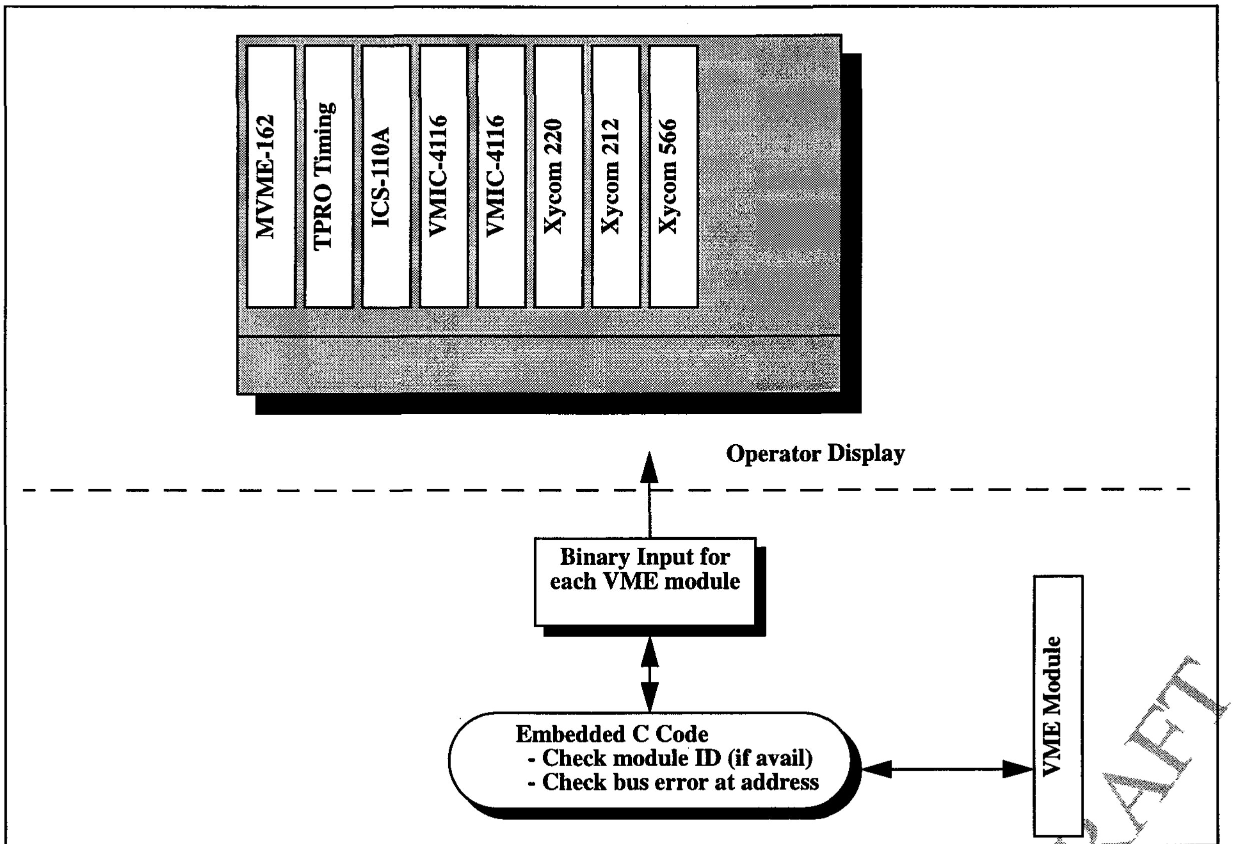


Figure 9: VME Diagnostic Software

3.3.3.2 Software Heartbeat

To test the availability and response of all PSL sequencing software, a simple heartbeat mechanism is incorporated. An outline sketch is shown in Figure 10: PSL Subsystem Heartbeat Diagnostic. The heartbeat itself is composed of two records. One generates the heartbeat, a simple roll over counter which increments every 10 seconds. When seen by the sequencers/closed loop algorithms, they set their heartbeat image to the same value. When the next 10 second scan comes around, a calc record ensures that all sequencers have responded properly, issuing an alarm if a sequence has failed to respond. The heartbeat is then incremented, and the cycle continues indefinitely. The heartbeat response feature is built into every PSL sequencer, though it will not be shown explicitly in following sections where the PSL sequencers are described in detail.

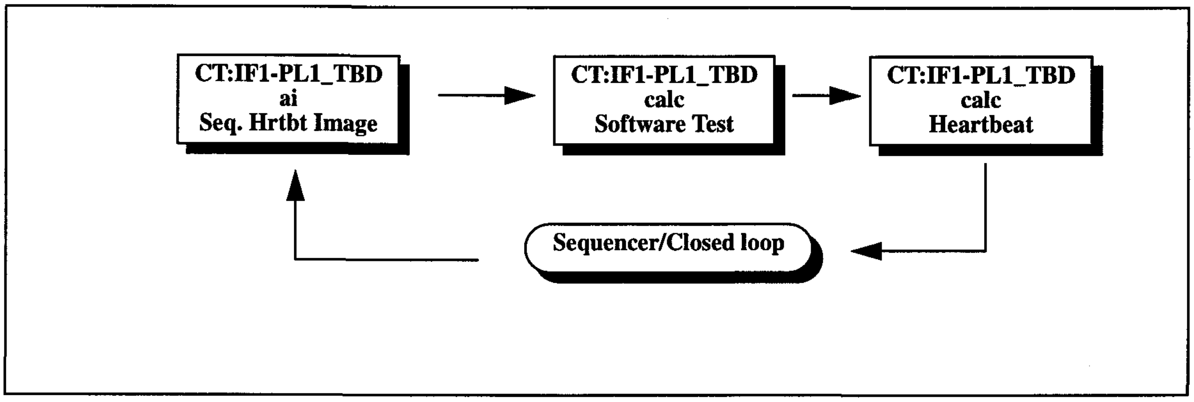


Figure 10: PSL Subsystem Heartbeat Diagnostic

3.3.4. Abnormal Condition Detection/Response (Real-time)

At the subsystem level, abnormal conditions are:

1. VME crate error, as determined by the VME diagnostic software
2. Heartbeat monitor error
3. PSL automatic startup/shutdown sequence error

For the first two, alarms are issued and registered in a branch of the PSL alarm tree. For the latter, an alarm is issued and text is also provided on one of the PSL display pages to notify the operator of the problem.

3.3.5. Operator Displays

Two operator display are designed for the PSL subsystem level. These are:

1. PSL Subsystem Overview
2. PSL Hardware/Software Status Page

These display pages are shown in Appendix A.

3.4. PSL LCU and Device Software (Level 2 & 1B)

3.4.1. Introduction

3.4.2. Argon Ion Laser LCU Software

3.4.2.1 Introduction

The Laser LCU software makes use of standard EPICS records and drivers to interface to the laser via the laser's 37 pin D connector, which, in previous LIGO prototype operations, connected to the hand held control unit provided by the manufacturer.

3.4.2.2 Normal Operation (Real-time)

In normal operation, the Laser LCU monitors laser parameters, and provides automatic sequencing to the ON and OFF states. For the latter, two pushbuttons exist on the PSL subsystem and Laser LCU displays to actuate a transition between states. These are the LCU Mode (Auto/Manual) and Laser Power On/Off. SNL software exists on the VME processor which monitors these inputs (and others), and determines the operations to be performed.

When the LCU Mode is set to 'Manual', all controls within the Laser LCU display are active to the operator, and the operator can manually adjust laser parameters. The one parameter which must still go through the automatic sequencer is the laser power on/off request. The EPICS records and a section of the Laser sequencer which handles this is shown in Figure 11: Laser LCU Sequence Response in Manual Mode. As shown, if the Laser LCU Mode is 'Manual', and the laser is requested on, the laser sequencer will first verify that the appropriate interlocks are complete and then set the actual EPICS binary output record which is attached to the laser on/off line. If the interlocks are not complete, a message will be issued to the operator display and the On/Off request will be reset to OFF. The reason for this design is as a safety backup. For instance, if the display pushbutton tied directly to the Power On/Off, the operator could set it On, but the interlocks not be made up, and the laser will not actually turn on. But now the operator leaves to go check interlocks, or forgets about the On request, interlocks make up at a future time, and the laser will immediately come on. This could then occur when personnel are not expecting it. When normal LIGO safety procedures are followed, this should not result in injury or damage, but this software provides a second level of verification.

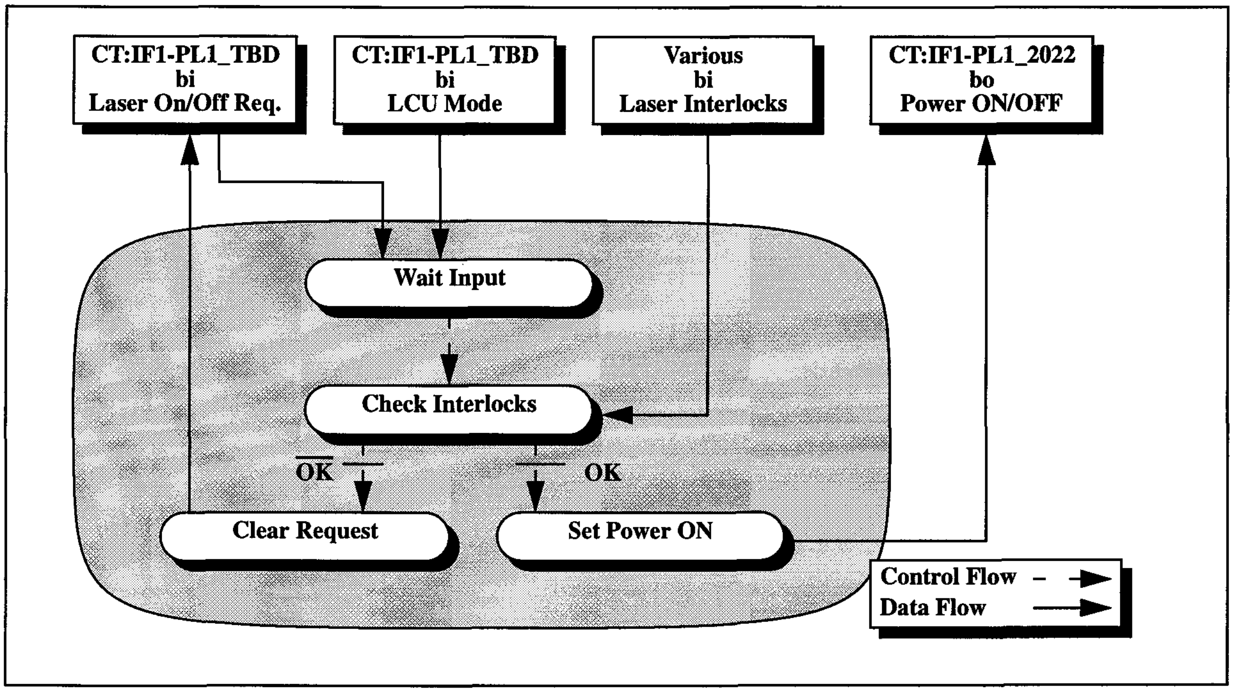


Figure 11: Laser LCU Sequence Response in Manual Mode

Upon operator (or other software) selection of the Auto LCU Mode, the sequencer will perform the steps as outlined in Figure 12: Laser LCU Automatic Sequencer to automatically startup or shutdown the laser. The sequencer will additionally respond if laser interlocks drop out while the laser is on (or transitioning to the on state) or the operator returns the Manual LCU mode. In the latter case, if the sequencer is operating an auto on or auto off sequence, it will suspend operation where it is, leaving the laser in whatever transition state that was reached in the sequence.

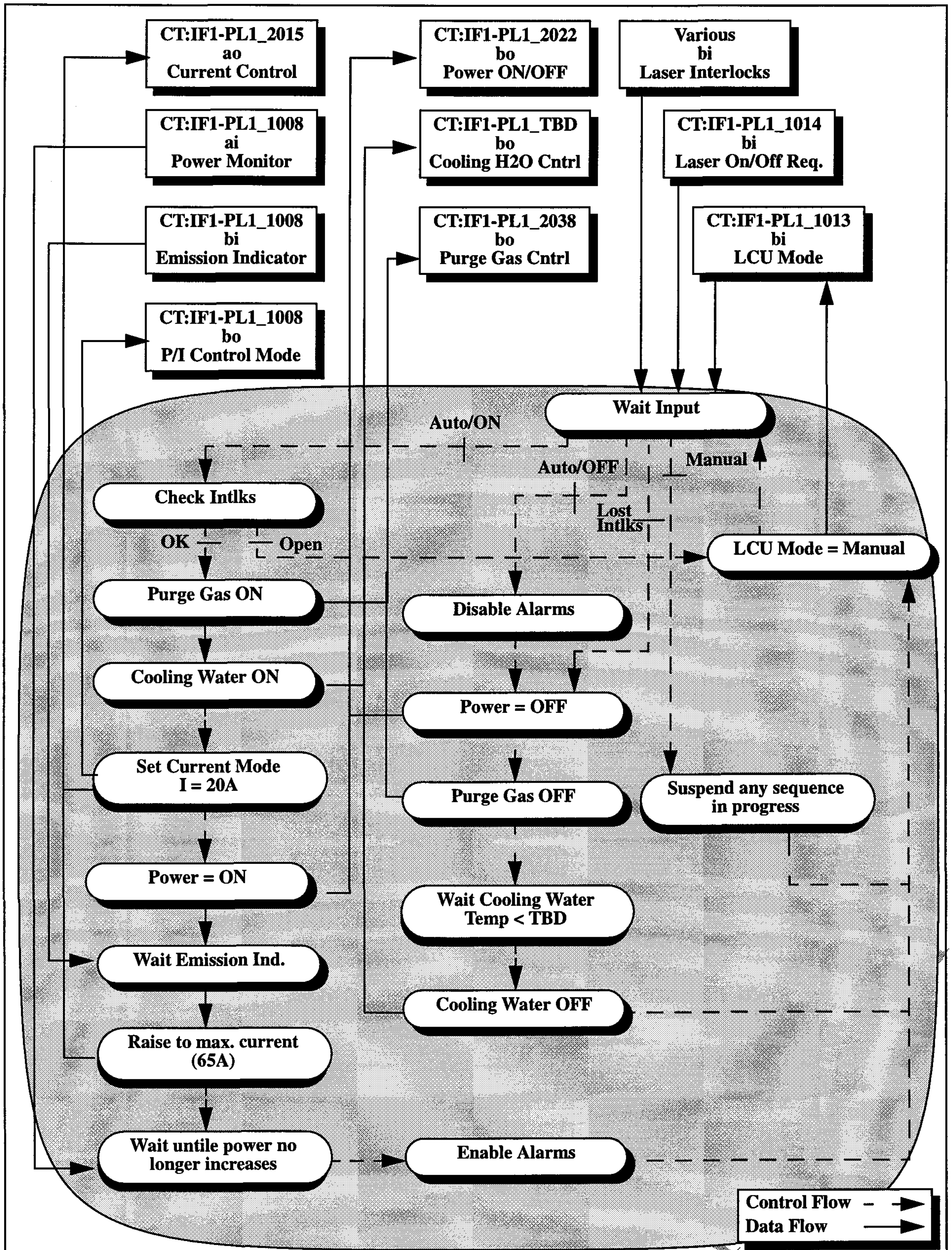


Figure 12: Laser LCU Automatic Sequencer

3.4.2.3 Diagnostics (Real-time)

No specific diagnostic routines exist for the Laser LCU.

3.4.2.4 Abnormal Condition Detection/Response (Real-time)

Warnings and alarms generated by the laser LCU are grouped into two categories:

1. Those which are set to a fixed value and are monitored at all times.
2. Those which are set dependent to the operating state of the laser.

This is done to avoid generating meaningless alarms, such as no laser emission when the operator has requested the laser OFF.

3.4.2.4.1 Continuous set alarms and warnings

The following table shows those channels for which alarms/warnings are always set and monitored.

Table 1: Laser Continuous Alarm/Warning Monitor Channels

Channel	*Setpoint*	*Level*
Head Cover Intlk	TTL High	Alarm
Tube Fill Status	TTL High	Warning
High Water Temp	TTL High	Alarm
Regulator Fault	TTL High	Alarm
Purge Gas Tank Pressure	300 lbs.	Warning
Purge Gas Tank Pressure	50 lbs.	Alarm

3.4.2.4.2 Alarms/Warnings in ON state

Once the laser is stable in the ON state, the automatic sequencer (if Laser LCU set to AUTO mode) will set the following alarm/warning levels as shown in Table 2. If any of these channels, or the continously monitored channels, alarms, the automatic sequencer will shut down the laser.

Table 2: Laser Alarm Setpoints in ON State

Channel	*Setpoint*	*Level*
Laser Key Switch	OFF	Alarm
Laser Water Flow	Low	Alarm
Laser On/Off Status	OFF	Alarm

3.4.2.4.3 Alarms/Warnings in OFF state

In the OFF state, the Water Flow alarm is set to activate if water is still flowing 15 minutes after the return water temperature drops below TBD degrees.

3.4.2.5 Operator Displays

A single operator display represents the Laser LCU. It is shown in Appendix B.

3.4.2.6 Data Archival

Both laser power and current are continuously archived.

3.4.2.7 Alarm Handler

A separate branch is built for laser alarms beneath the PSL subsystem. It contains those channels listed in

3.4.3. Power Stabilization Servo (PSS) Software

3.4.3.1 Introduction

3.4.3.2 Normal Operation (Real-time)

The PSS software consists of a combination of standard EPICS records and two SNL-based processes. The SNL processes are:

1. pss_keys: The PSA mode select and PSA power select are mbbi records with associated operator push buttons for selection. Since these must drive descrete bits in the Xycom 220 binary output VME module as part of a single 'word', and EPICS does not have this conversion capability, an SNL program was developed for the sole purpose of providing this conversion process. Depending on the PSA Mode selected, this routine will set the 'ringdown enable' and 'output enable' bits accordingly. For the power select, this routine will set the appropriate values to the PSA gain select channels 0 through 4.
2. pss_lock_sequence: This routine will automatically 'lock' the pss to the laser output power requested by the operator (or other CDS software). This sequence, and associated EPICS records, is outlined in the following figure.

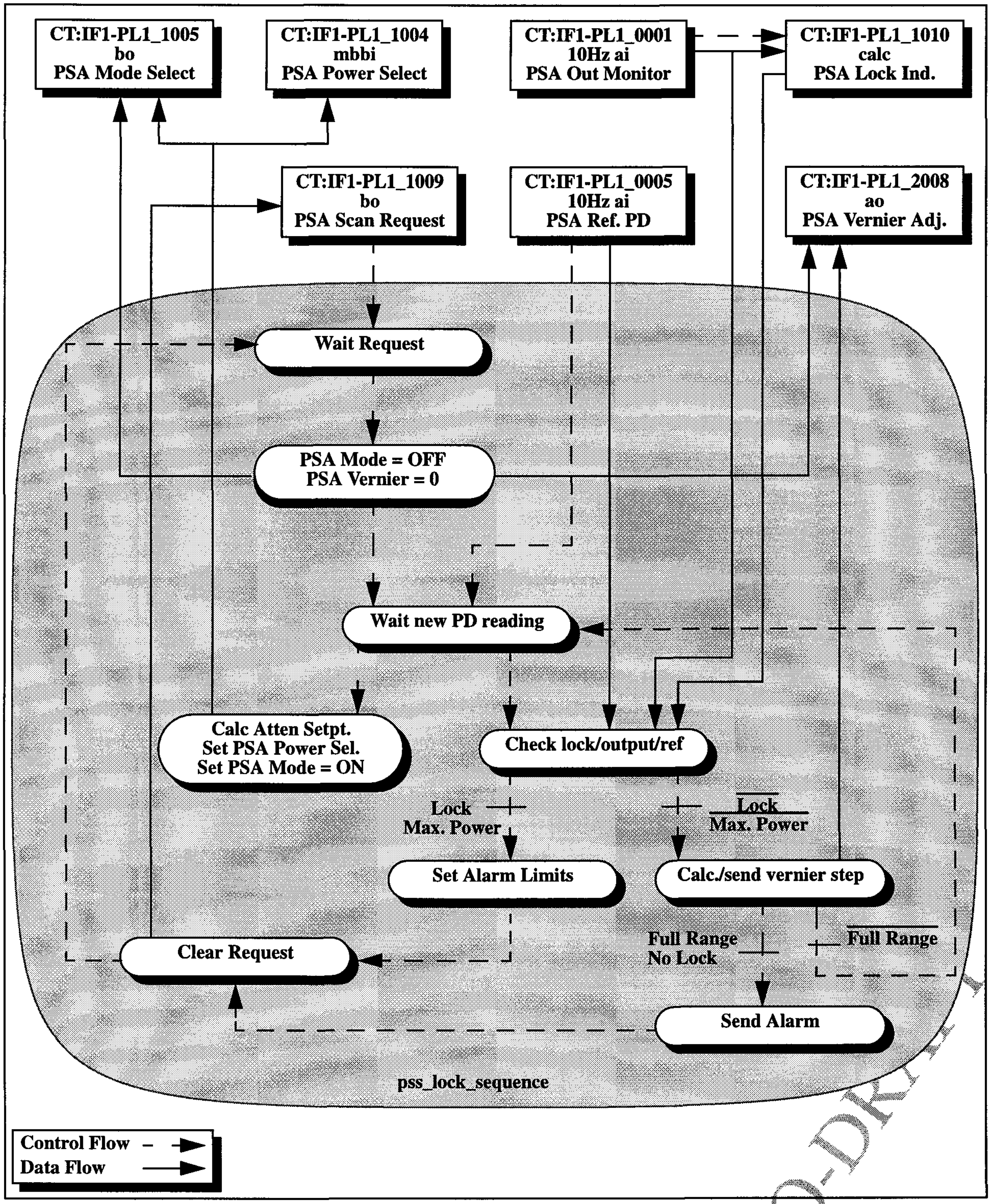


Figure 13: PSS Lock Sequence and Associated EPICS Records

3.4.3.3 Abnormal Condition Detection/Response (Real-time)

TBD

3.4.3.4 Diagnostics (Real-time)

3.4.3.4.1 Ringdown Measurement

For initial ringdown measurements, since the first PSL is slated for the Caltech 40M, the equipment will be self-contained within the PSL controls. For LIGO, portions of this measurement equipment and software may be incorporated into the CDS Remote Diagnostics. This is due to the requirement to provide ringdown measurements not only for the PSL reference cavity, but the mode cleaner and arms as well.

A sketch of the hardware involved is shown in Figure 14: Cavity Ringdown Electronics. These components are:

1. Xycom220 binary output module: sets PSA module to either normal output or ringdown modulation output.
2. VMIC4116 DAC: Provides variable amplitude modulation at cavity ringdown frequency (Note: This will be tested to verify output risetimes are fast enough to meet specs. If not, two channels of VMIC4116 will be used, one set to the high modulation amplitude, one to the low, with fast switches used to modulate between the two levels.)
3. Omnibyte Comet Digitizer: This module contains four 5MHz ADCs, which can be skewed to provide up to 20MHz sampling. Since the requirements call for 10MHz, each cavity photodiode will be connected to two ADC channels. In this configuration, each photodiode will be sampled at 10MHz into 128K sample buffers, allowing a maximum sample time of 12.8msec.
4. 10MHz clock: Clocks data into the Comet module. Source of 10MHz TBD.

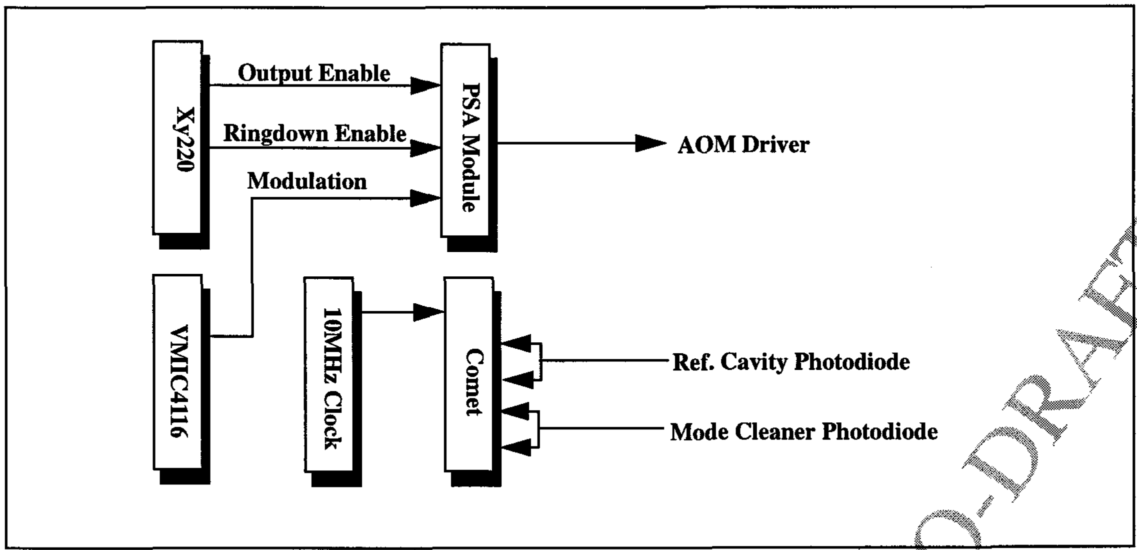


Figure 14: Cavity Ringdown Electronics

A diagram of the software functions and sequence is shown in .

CT:IF1-PL1_0006
ai
Ref. Cavity Diode

3.4.3.4.2 Intensity Noise Spectrum Measurement

For this measurement, the PSL software must coordinate with the CDS Remote Diagnostics system. This procedure is ***TBD***.

3.4.3.5 Operator Displays

Three operator displays are designed for the PSS:
1. Normal operations
2. Ringdown measurement
3. Intensity Noise Spectrum measurement

These displays are shown in Appendix B.

3.4.3.6 Data Archival

TBD

3.4.3.7 Alarm Handler

TBD

3.4.4. Phase Modulation LCU Software

This section is ***TBD***, awaiting the design of the control electronics.

3.4.4.1 Introduction

3.4.4.2 Normal Operation (Real-time)

3.4.4.3 Abnormal Condition Detection/Response (Real-time)

3.4.4.4 Diagnostics (Real-time)

3.4.4.5 Operator Displays

3.4.4.6 Data Archival

3.4.4.7 Alarm Handler

3.4.5. Frequency Stabilization LCU Software

3.4.5.1 Introduction

The FSS LCU software is comprised of:

1. A group of independent EPICS records for monitoring and manual control (see index for list).
2. A group of connected EPICS records for monitoring visibility and initiating the Slow PZT Scan.
3. A sequencer program for operating FSS scan modes.

An overview of the EPICS records which control and monitor the visibility and FSS modes and their relation to the sequencer is shown in the following figure.

3.4.5.2 Normal Operation (Real-time)

'Normal' operation is defined as the FSS is locked (Visibility > Threshold) and/or the FSS is not set to automatically operate in a scan (Slow PZT ramp) mode. Figure 14: FSS PZT Scan Sequence shows the basic records associated with the normal mode.

In this mode:

1. The Ref PD Monitor and RF PD Monitor record values are set by the ICS110 device driver (see Device Driver section) and caused to process at a 10Hz rate.
2. These records forward link to the visibility monitor, which calculates visibility.
3. The visibility record, in turn, forward links to the lock indicator, which determines if the FSS is/is not in a lock condition.

Two records also monitor the FSS operating modes when selected by the operator. These two records feed into a calculation record, which assigns an overall FSS operating mode as a function of the two previous records. This information is fed to the Slow PZT scan sequence, along with lock and threshold information.

3.4.5.3 Abnormal Condition Detection/Response (Real-time)

Abnormal operation is defined as:
1. FSS is not locked.
2. A continuous Slow PZT scan mode has been selected for testing.

To handle these modes, an SNL-based sequencer is used to provide closed loop control. This program (vismon1) is outlined in Figure 14: FSS PZT Scan Sequence.

After initial startup, the program will wait for a new event from the EPICS records (described previously). It will move out of this state when:
1. The operator (or other software) has selected a new FSS operating mode. The sequencer will note the new operating mode and set wideband/bypass, as appropriate, to the LLA.
2. The operator (or other software) has selected a new threshold for the visibility/lock indication. The sequencer will note this information for later use if it goes into a scan mode.
3. The FSS has dropped out of lock or a continuous scan mode has been selected. The sequencer sets the LLA scan enable to ON, and returns to the wait state to allow the EPICS records to transfer the scan enable data to the LLA.
4. Scan is enabled and a FSS scan mode is selected. The sequencer now ramps the Slow PZT from 0 to 850VDC (via the LLA) until:
   - In an Auto scan mode, FSS lock is attained.
   - In a Continuous scan mode, the operator deselects continuous scan.

This latter sequence is shown in the bottom right portion of the figure. Two items of note are:
1. Fast Read ICS110: Because the ICS110 has a 64KByte buffer which is wants to fill (requiring 10msec at a 100KHz sample rate) prior to stopping acquisition, the PSL software must give descrete start and stop commands to the ICS110 to meet the 1msec/point ramp requirement. (See also the ICS-110 Device Driver section of this document) Therefore, this code is embedded into this sequencer to synchronize with the ramp DAC commands at the higher speeds. This requires a synchronization as well with the normal ICS-110 device driver, to ensure that they do not issue conflicting commands to the module.
2. The 'sleep' cycle at the end of a ramp cycle. After a full ramp has been generated, if lock is not attained or in a continuous scan mode, the process will remove itself from the processor for approximately 200msec. Since this sequencer has a high priority and basically locks all other tasks out for ~850msec (if a full ramp is generated), time is needed by other tasks to catch up and avoid running 'blind', due to no data updates in the rest of the system.

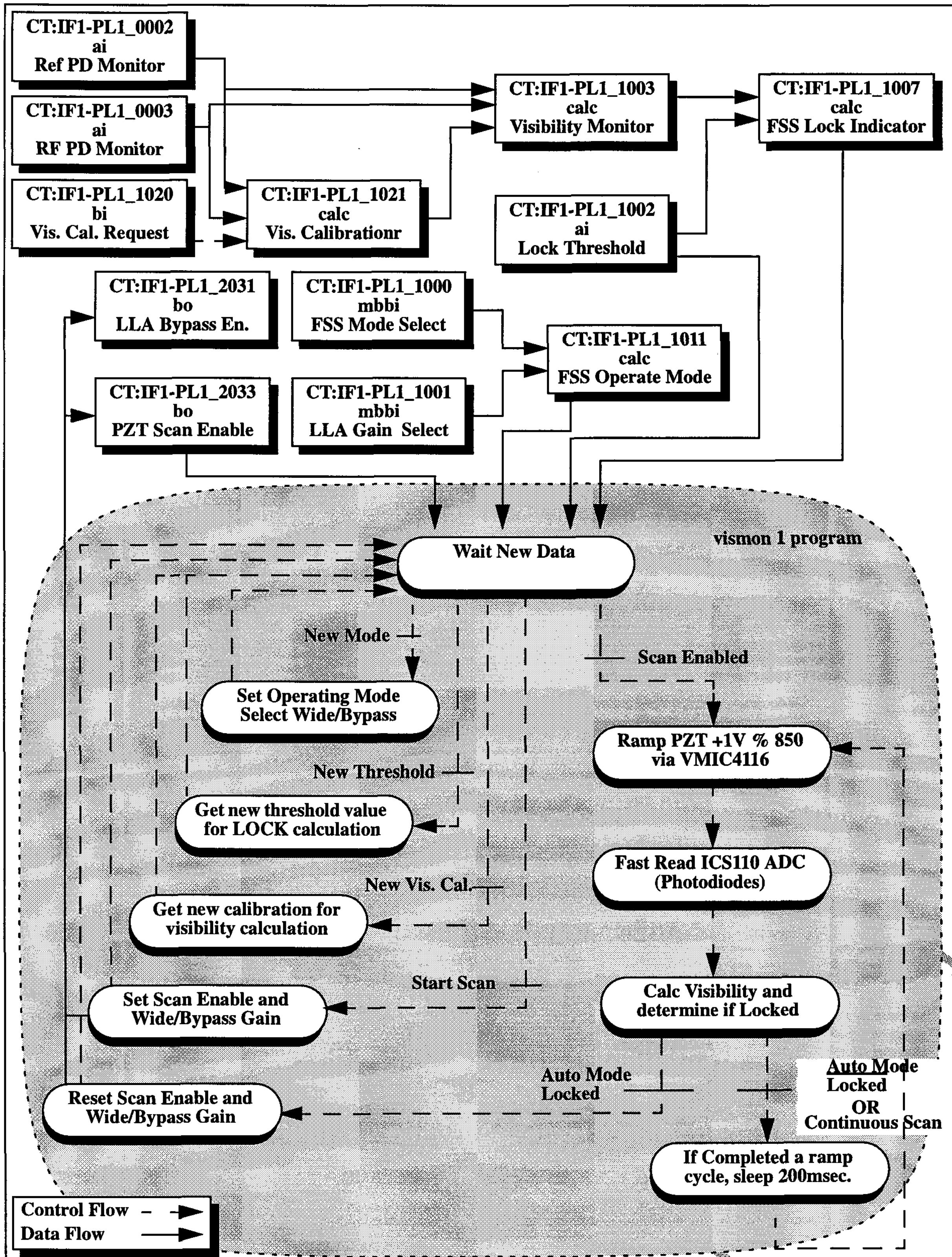


Figure 15: FSS PZT Scan Sequence

3.4.5.4 Diagnostics (Real-time)

3.4.5.5 Operator Displays

Two displays are designed for the FSS, one for normal operations and one for the frequency noise spectrum measurement. These displays are shown in Appendix A.

3.4.5.6 Data Archival

TBD

3.4.5.7 Alarm Handler

TBD

3.4.6. Reference Cavity LCU Software

3.4.6.1 Normal Operation (Real-time)

3.4.6.2 Abnormal Condition Detection/Response (Real-time)

3.4.6.3 Diagnostics (Real-time)

3.4.6.4 Operator Displays

3.4.6.5 Data Archival

3.4.6.6 Alarm Handler

3.5. PSL Hardware Interface Device Drivers (Level 1A)

3.5.1. Introduction

The PSL Design incorporates a number of commercial VME modules in a single VME crate. Hardware interfaces exist at the VME backplane to the various modules, and to an ethernet connection at the VME processor.

3.5.2. Crate Layout

The layout of this crate, and the module base addresses for software communications are shown in Figure 15: PSL VME Crate Layout and Module Base Addresses.

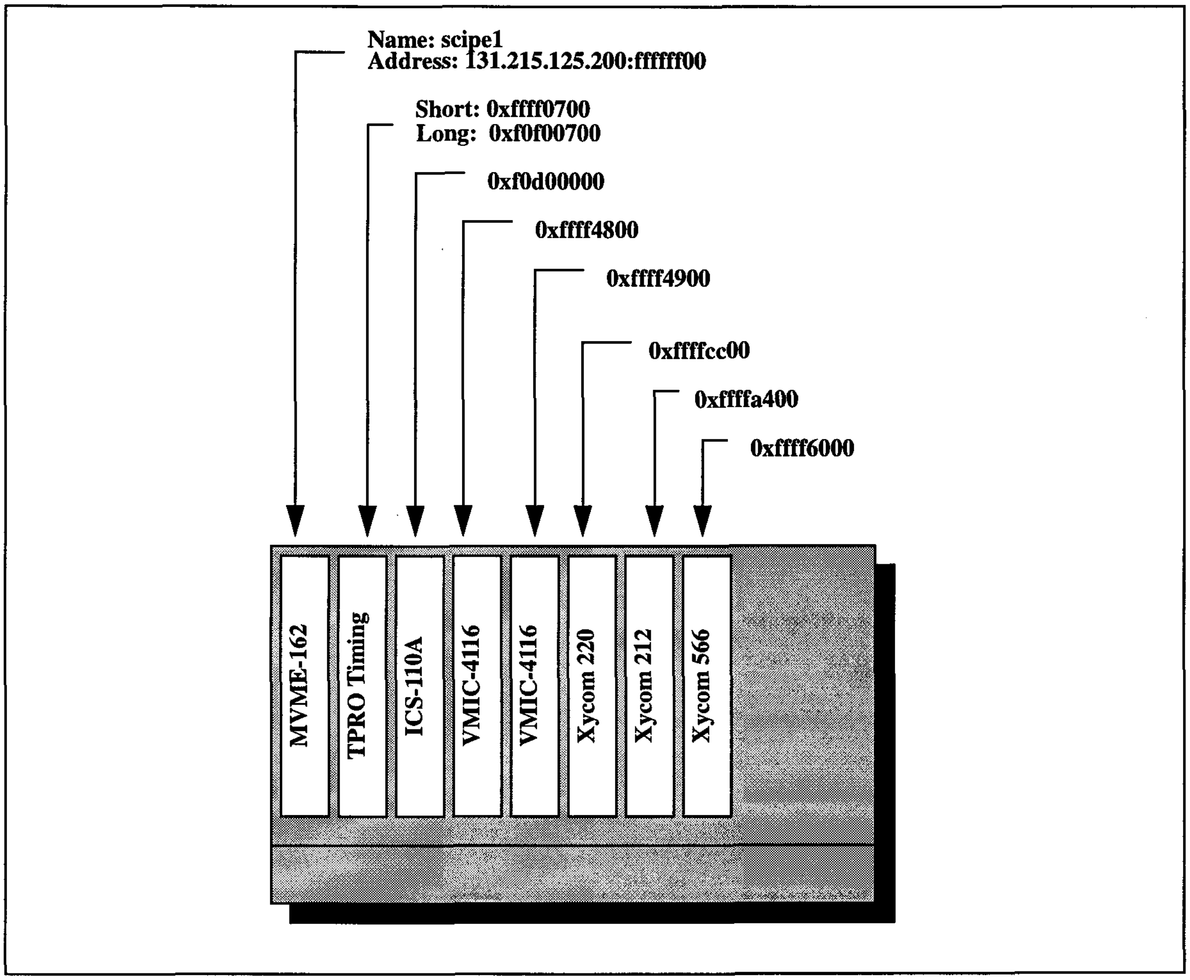


Figure 16: PSL VME Crate Layout and Module Base Addresses

3.5.3. Target Processor

The target VME processor is a Motorola MVME162. During prototype tests on the 40 meter, this will be a MVME167, as it was readily available.

3.5.4. Network Interface Software

The network interface will be through the Channel Access protocol of EPICS. Standard sockets, NFS, and remote login will also be available for off-line functions and some direct data storage functions.

3.5.5. VME I/O Module Drivers

3.5.5.1 EPICS Drivers

Drivers for the Xycom modules presently exist in EPICS, and will be used for this project. A driver for the VMIC4116 and TPRO timing module are being incorporated into EPICS by LANL. The TPRO module will replace the LANL master timing module within the core of EPICS (for LIGO use only). An EPICS driver is also to be developed by LANL for the ICS-110A module, but will not be used for the PSL (though it will be used in other applications). The reason for this is the different operating modes and performance requirements of the PSL software, which are not within the capabilities of EPICS, and the particular design of the module, requiring other than typical addressing and readout schemes. Further discussion of this and the design to be incorporated is detailed in the following section.

3.5.5.2 ICS-110A Driver

The ICS-110 16 bit ADC module driver design is shown in the following figure. This is not designed as a standard EPICS driver, but rather as an EPICS SNL-based, multi-threaded program. The primary reasons for this are:

1. ICS module is read through a FIFO, rather than each channel at a register in memory.
2. The ICS module wants to fill its 64K byte FIFO prior to issuing an interrupt that it has data. This means that it takes at least 10msec before it is ready with data, and it is not desirable to hold up record processing for this long.
3. The ICS110 is operated directly by the FSS sequencer (see section) in a ‘Fast’ Mode, which must be synchronized with this driver to avoid collisions of commands.

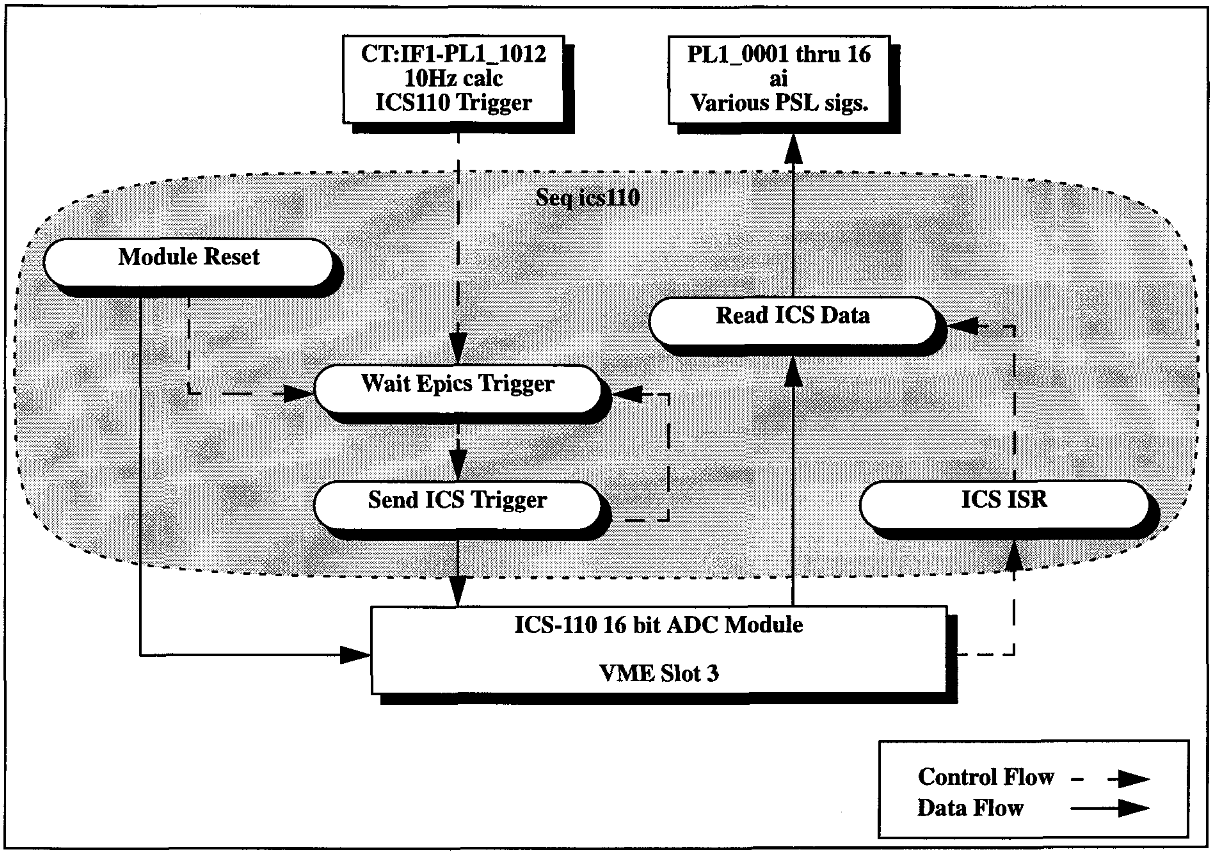


Figure 17: ICS-110A VME Module Device Driver

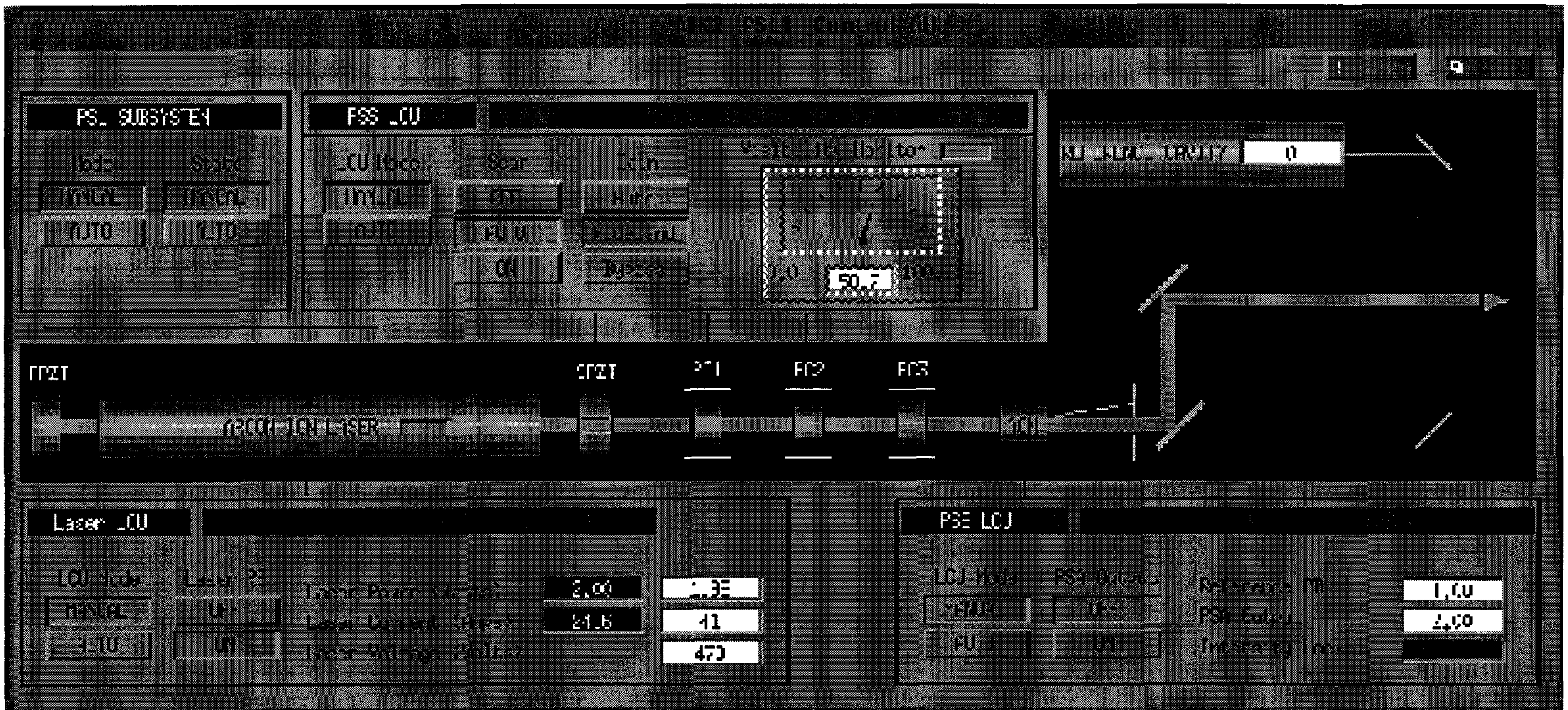
The key portions of the sequencer (program ics110) are shown within the dotted line rectangle. These parts are:

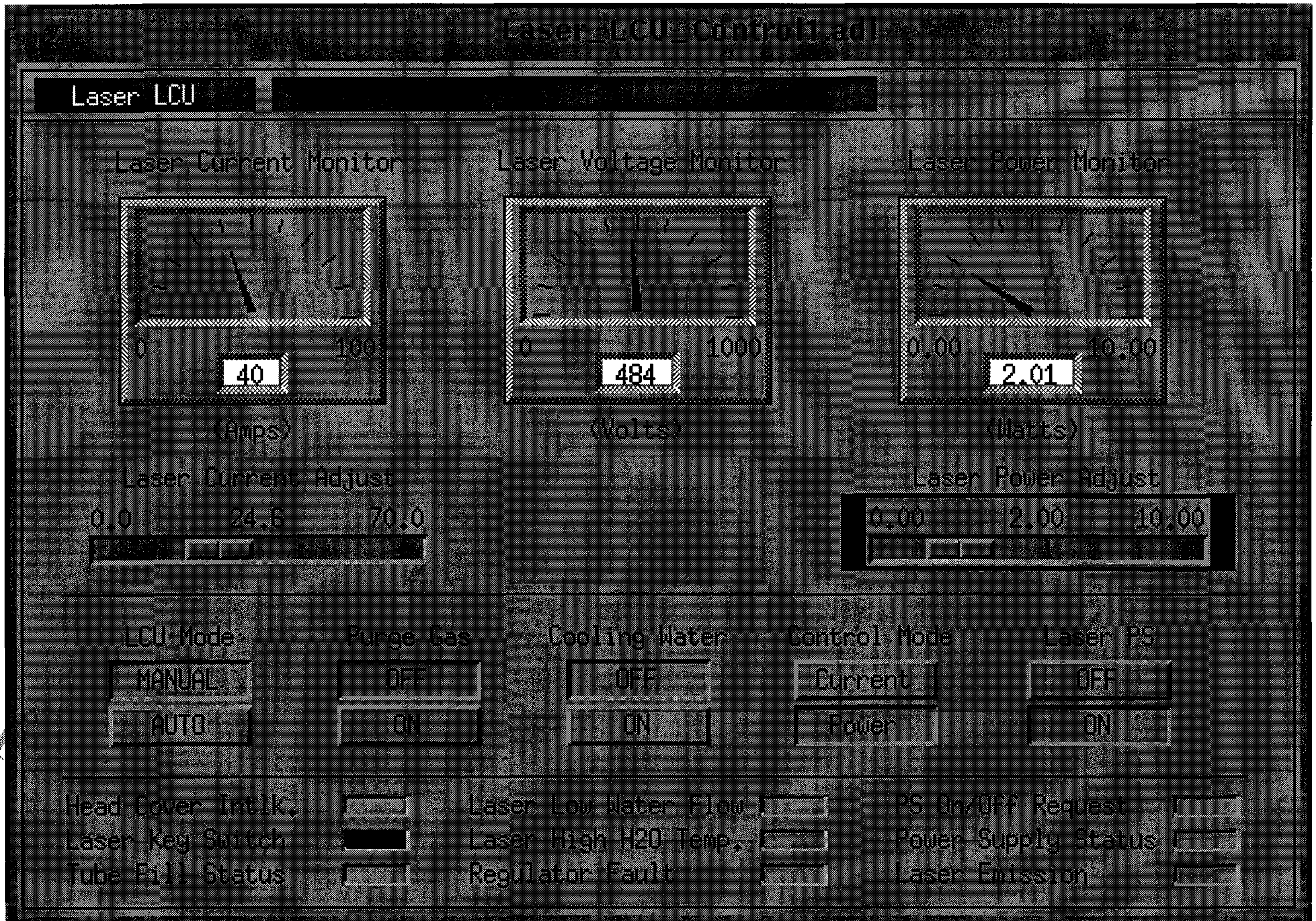
1. Module Reset: Executed on program initialization. It:
   - Hard resets the ICS110 module.
   - Sets the clock rate to sample at 100KHz
   - Sets the number of channels to 32
   - Sets the ICS110 interrupt vector to call the ICS Interrupt Service Routine (ISR)
   - Enables the VME interrupts
   - Sets up a semaphore for the ISR to call the ICS data readout routine.
2. Wait EPICS trigger: The sequencer is triggered at a 10Hz rate by the ICS110 trigger record in the EPICS database. This record is nothing more than a counter set to scan at 10Hz. When this trigger event is seen by this routine, and the Slow PZT is not scanning, the Send ICS Trigger function is called.
3. Send ICS Trigger: Sends a start acquisition/enable interrupt command to the ICS110 VME module, then returns to step 2 above.
4. ICS ISR: This ISR is called when the ICS110 FIFO is half full. This routine disables the ICS110 interrupt and gives a semaphore to the Read ICS Data routine.
5. Read ICS Data: Upon receiving the semaphore from the ICS ISR, this module reads in the first 16 data values of the ICS110 FIFO, corresponding to the first 16 ADC channels. This data

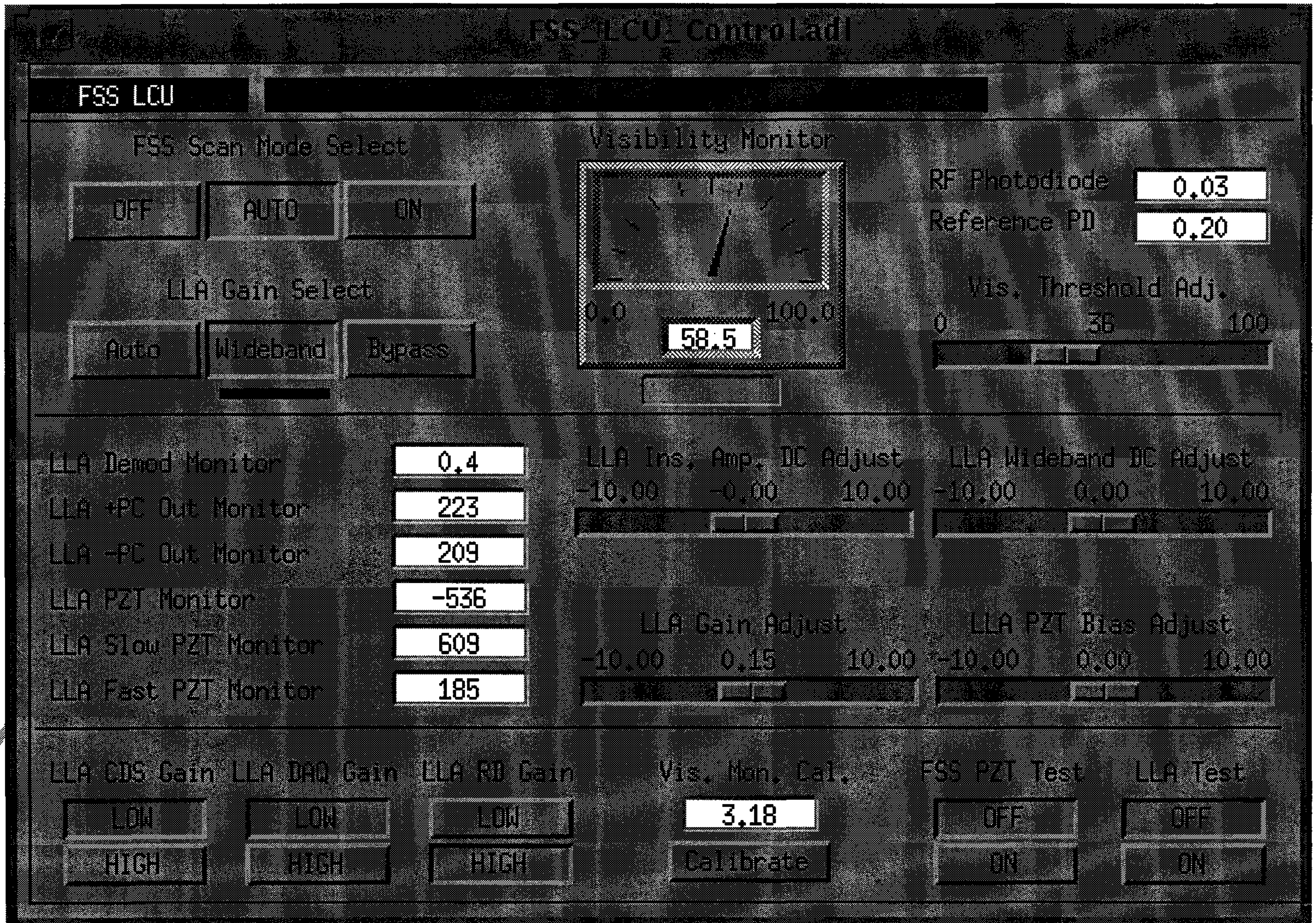
is then put into the EPICS database records, causing them to process and update.

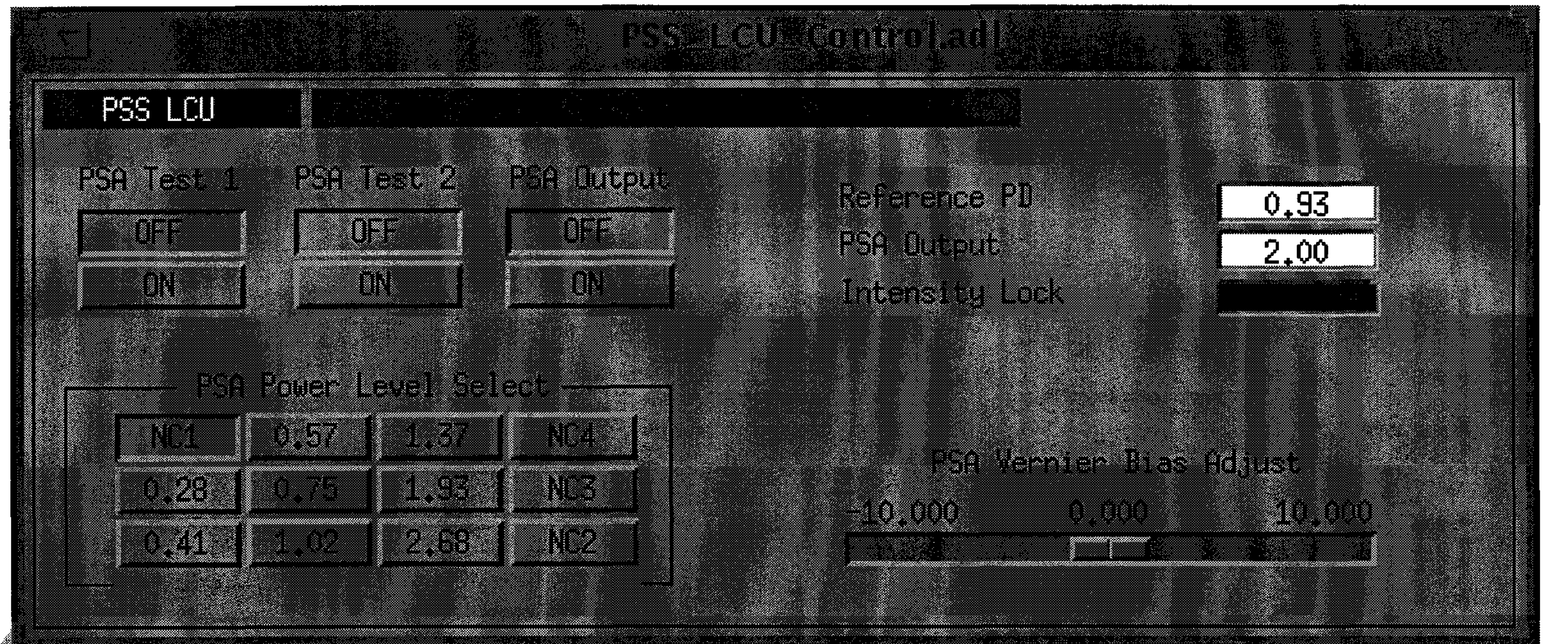
Appendix A: Operator Displays

This Appendix provides raster screen dumps of the various operator displays designed for the PSL.


FSS LCU
Scan
ON
50.7
Laser LCU
LCU Mode
2.00
24.6
41
473
PSL LCU
Reference PD
PSA Output
1.00
2.00


Laser_LCU_Control1.adl
Laser LCU
Laser Current Monitor
0
100
40
(Amps)
Laser Voltage Monitor
0
1000
484
(Volts)
Laser Power Monitor
0.00
10.00
2.01
(Watts)
Laser Current Adjust
0.0
24.6
70.0
Laser Power Adjust
0.00
2.00
10.00
LCU Mode
MANUAL
AUTO
Purge Gas
OFF
ON
Cooling Water
OFF
ON
Control Mode
Current
Power
Laser PS
OFF
ON
Head Cover Intlk.
Laser Key Switch
Tube Fill Status
Laser Low Water Flow
Laser High H2O Temp.
Regulator Fault
PS On/Off Request
Power Supply Status
Laser Emission


FSS_LCU_Controlad!
FSS LCU
FSS Scan Mode Select
OFF
AUTO
ON
LLA Gain Select
Auto
Wideband
Bypass
Visibility Monitor
0.0
100.0
58.5
RF Photodiode
0.03
Reference PD
0.20
Vis. Threshold Adj.
0
36
100
LLA Demod Monitor
0.4
LLA +PC Out Monitor
223
LLA -PC Out Monitor
209
LLA PZT Monitor
-536
LLA Slow PZT Monitor
609
LLA Fast PZT Monitor
185
LLA Ins. Amp. DC Adjust
-10.00
-0.00
10.00
LLA Wideband DC Adjust
-10.00
0.00
10.00
LLA Gain Adjust
-10.00
0.15
10.00
LLA PZT Bias Adjust
-10.00
0.00
10.00
LLA CDS Gain
LLA DAQ Gain
LLA RD Gain
LOW
HIGH
Vis. Mon. Cal.
3.18
Calibrate
FSS PZT Test
LLA Test
OFF
ON


PSS_LCU_Control.adl
PSS LCU
PSA Test 1
PSA Test 2
PSA Output
OFF
ON
OFF
ON
OFF
ON
Reference PD
0.93
PSA Output
2.00
Intensity Lock
PSA Power Level Select
NC1
0.57
1.37
NC4
0.28
0.75
1.93
NC3
0.41
1.02
2.68
NC2
PSA Vernier Bias Adjust
-10.000
0.000
10.000