European Organisation
for Astronomical
Research in the
Southern Hemisphere

Organisation Européenne
pour des Recherches
Astronomiques
dans l'Hémisphère Austral

Europäische Organisation
für astronomische
Forschung in der
südlichen Hemisphäre

VERY LARGE TELESCOPE

PULPO I USER MANUAL

Doc. No.: VLT-MAN-ESO-xxxxx-xxxx

Issue: 1.3

Date: 09.11.2010

Prepared: Nicolas Haddad

Name Date Signature

Approved:

Name Date Signature

Released:

Name Date Signature

CHANGE RECORD

ISSUE	DATE	SECTION/PARA. AFFECTED	REASON/INITIATION DOCUMENTS/REMARKS
1.1	Aug 1998	All	Creation
1.2	Dic 2008	All	Updating according to last firmware
1.3	Sept 2010	All	Updated according to firmware 3.56

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Purpose

The purpose of this manual is to provide operational and interfacing information for PULPO I multifunctional module. This information should allow the users to interface PULPO I with an ESO cryostat, shutter driver and FIERA controller, understand the communication protocol and control it through the serial link.

1.2 Applicable Documents

1.3 Reference Documents

1.4 List of Abbreviations and Acronyms

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

CAS	Central Alarm System
DCS	Detector Control Software
ESO	European Southern Observatory
GUI	Graphical User Interface
HW	Hardware
ICS	Instrument Control Software
INS	Instrumentation Software Package
I/O	input/output
ISF	Instrument Summary File
IWS	Instrument Workstation
LCC	LCU Common Software
LCU	Local Control Unit
MS	Maintenance Software
N/A	Not Applicable
NC	Normal close (in reference to relay outputs)
PID	Proportional Integral and Derivative controller
RAM	Random Access Memory
RTAP	Real-Time Application Platform
SLED	Stabilized LED Light Source
SMT	Surface Mount Technology
SW	Software
TBC	To Be Clarified
TBD	To Be Defined
TCS	Telescope Control Software
TRS	Time Reference System
VLT	Very Large Telescope
WS	Workstation

2 BRIEF DESCRIPTION

PULPO I (Spanish for octopus), is an ESO standard multifunctional module designed to interface with ESO cryostat, shutters and ESO CCD control systems. It is able to:

- Read up to six temperature sensors (4 leads PT100).
- Control 3 heaters (up to 7 watts over 75 ohms each), using a digital PID algorithm.
- Read out full range vacuum sensor Balzer model PKR 250 or full range Edwards model WRG-D-NW25.
- Provide versatile control alternatives for various type of instrument shutter (IRIS, SESO, EMMI, SUSI II, VIMOS.)
- Provide current to a light source (LED) through a 12 bits DAC controlled current source (0 to 20mA) for CCD testing. The LED mode can also be used for driving the SLED device.
- Issue, in case of sensors values out of limits, alarm signals, either via internal buzzer or NC relay connected to external devices like telephone dialler, alarm concentrators, or Paranal CAS system. .
- Communicate with host computer through an opto-isolated RS232 serial port.
- Display information via front panel liquid crystal display (LCD).
- Accept operational mode modifications through front panel key pads.
- Log sensor values on internal RAM for later recovery through RS232 link.
- Retain operational parameters in RAM (battery backup) to resume operation after a power cycling.
- Perform exposure timing .
- Storage of controlling parameters and set up values in non volatile RAM.

3 HARDWARE

PULPO boards are enclosed in an aluminium box (H 70mm, W 105mm and D 250mm) made up of 2 matching off-the-shelf aluminium profiles and two custom made covers (front and rear covers). All sensors, heaters and power supply connections are located on the rear panel and a keypad and liquid crystal display (LCD) on the front panel to allow sensor value display and PULPO programming.

Inside the box, there are six printed circuit boards. Two of them, Power Supply and Main board, have Eurocard size (160x100mm), other two, Front Panel and Back Panel, are approximately 60mm x 100mm in size are used to interface the LCD, keypad and rear connectors to the PULPO CPU, and the last two boards are the CPU and shutter driver which are installed in sockets on the Main board. Bellow we can find a more detailed description of each board.

Currently, there are 2 hardware versions of the PULPO Power Supply and Mother boards deployed in Paranal, La Silla and Garching. The versions are 2.0 and 2.2 and they are software compatible, in the sense that all versions of PULPO firmware run without modification in any hardware version.

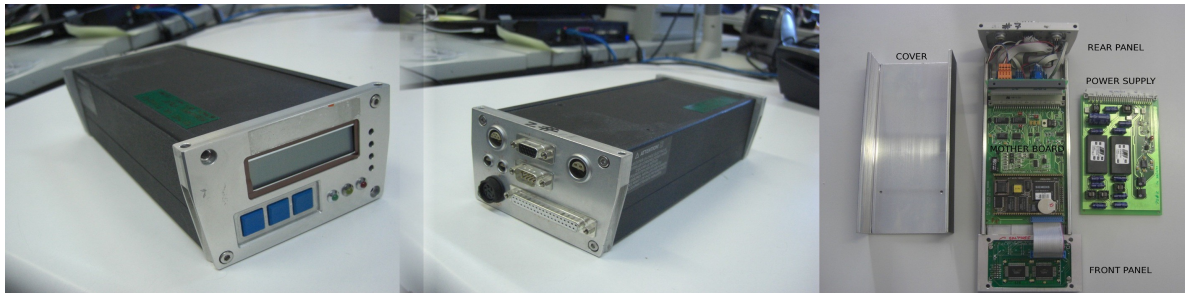


Figure 1: Mosaic of PULPO pictures

3.1 Front Panel

The front panel board is used to give physical support to the LCD and keyboard pads. It contains 3 LEDs to indicate PULPO peripheral activity like open shutter status (green LED), close shutter status (yellow LED) and heaters activity (red LED). There are also 4 potentiometers (only two are installed, the others are for future applications), the lower one to set the maximum current to heaters (IMAX on PULPO menu 1) that will activate the over current protection, the other potentiometer is used to adjust the LCD contrast . This board is interconnected to PULPO main board through a flat ribbon cable.

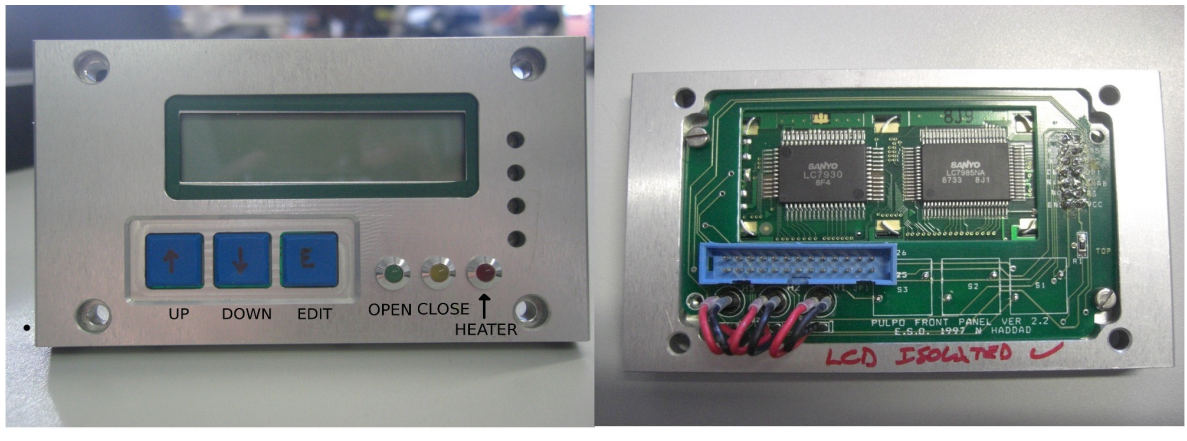


Figure 2: Front panel (front and rear view)

3.2 Rear Panel

This board provides electrical connection between the Power Supply board and the Main Board, it also connects the above mention boards to the eight connectors located in the back plate.

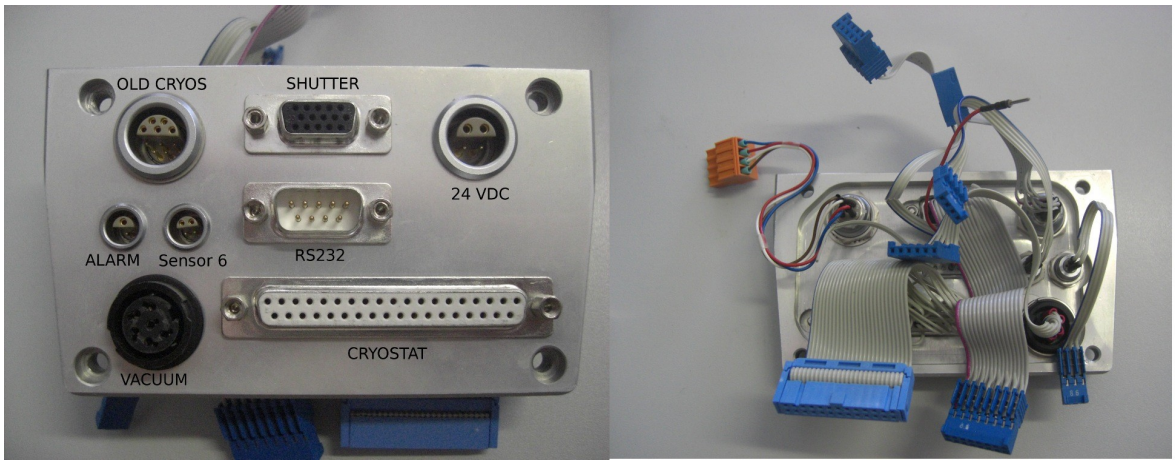


Figure 3: Rear panel

3.3 Power Supply Board

To perform all its functions, PULPO needs a set of various voltages; 24VDC@1A for heaters (3 heaters @ 320mA each), +/-15VDC@150mA for signal conditioning, 5VDC@100mA for digital circuitry, 5VDC and 12VDC with galvanic isolation from the others voltages to provide opto-isolation to the RS232 and power to external sensors. This opto-isolation is needed to avoid ground loops between the cryostat and PULPO.

The input voltage to the Power Supply board is a 24V DC, 2Amp with sense leads. All the others voltages needed by PULPO are generated out of this 24VDC via DC/DC modules.

All voltages have auto resetting SMT fuse protection.

The power supply has one jumper (J2, three pins) that defines the low level of the RX RS232 signal to the Phytec MiniModul micro-controller. If pins 1 and 2 are shorted, that low level will be 0 V, and if pins 2 and 3 are shorted, then the low level will be -15V (default setting).

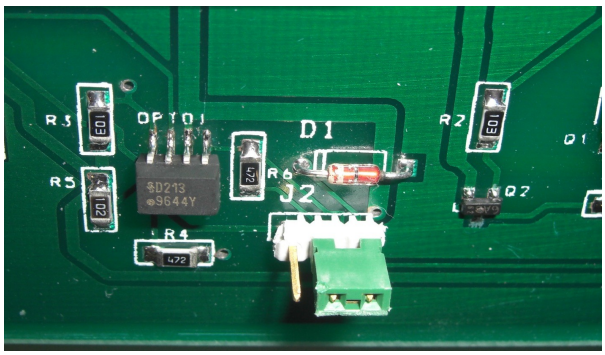


Figure 4: Power supply jumper in default position

3.4 Mother Board

PULPO heart is located on the Main board. This board contains a credit card size micro-controller board (Phytec MiniModul 535), consisting in a Siemens 80C535 micro-processor running at 12MHz populated with 64KB of code EPROM, 64 KB of RAM, one serial port, one 8 channels multiplexed 10 bits ADC, 24 I/O ports. The 80C535 has three fully programmable 16 bits counters and some of the I/O ports can be used as interrupts inputs. The RAM contained in the Phytec micro-controller is battery backed up, allowing PULPO to keep the setting values and operation mode in case of power failure.

The micro-controller board is piggy-back mounted on the Eurocard size Main Board which contains all the necessary conditioning electronics for sensors signals (temperature, vacuum), shutter, LED and heaters driving. The Main board contains a serial ADC (16bits of resolution) and multiplexers to select the temperature sensor. Other important component of the Mother Board is a heater watch dog circuitry which disable the power to the heaters if the watch dog is not reset every 100ms, this prevents the overheating of the detector in case the micro controller stop working while the power to the heaters is active. There is also a sensor to measure the current delivered to the 3 heaters and a comparator will disable the heaters if the current is over the limit defined by the potentiometer located in the front panel.

Apart from the micro-controller, there is a 44 pin base to install another piggy-back board containing the electronic to drive/interface shutters. Three of this 44 pins are used to read the shutter board identification (code).
The shutter board identification code is a way to allow the design of different shutter interface modules that can have a specific software driver in the PULPO firmware. Up to now only one opto-isolated module (code 001)has been built and is in use for all the different shutters co-existing on ESO visible instruments.

The shutter interface module has 5 opto-isolated outputs and 4 opto-isolated inputs plus galvanic isolated power supply (12VDC) that can be interfaced with the SESO shutter driver or any external shutter driver accepting external control. The opto-isolated inputs are connected to interrupt sensitive inputs on the micro-controller, allowing accurate timing of external events, mainly open and close delays.

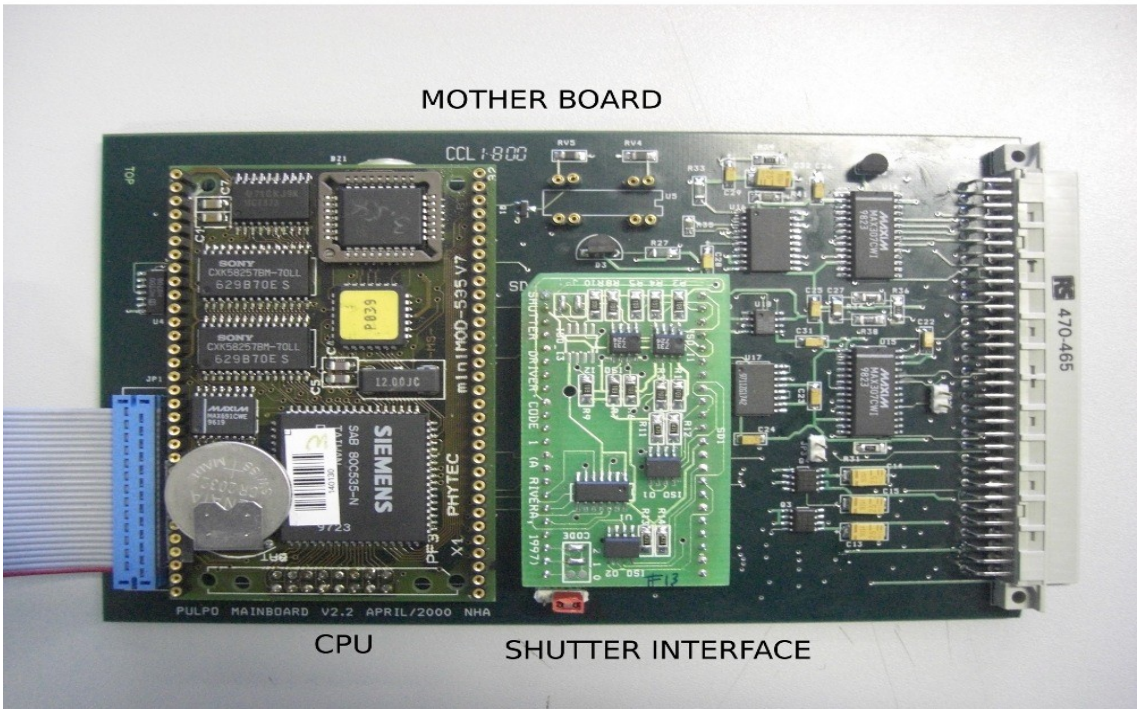


Figure 5: PULPO Mother Board (V2.2)

3.4.1 Main Board Jumpers (version 2.0)

The Main Board is suited with 4 jumpers (shown in figure 2-2). Jumper 3 and 4 should be left open if PULPO is connected to a cryostat generation 3 (VLT). For cryostats generation 1 and 2 (La Silla except SUSI II) both jumpers should be closed.

Jumpers 1 and 2 are used to select the output circuitry for heater 3. Only one of them should be closed. When J1 is closed, heater 3 is driven through a power MOSFET, and when J2 is closed, the driving is done through a DAC controlled current source. It is recommended to use J1 closed as the MOSFET output is more efficient and reduce power dissipation inside PULPO.

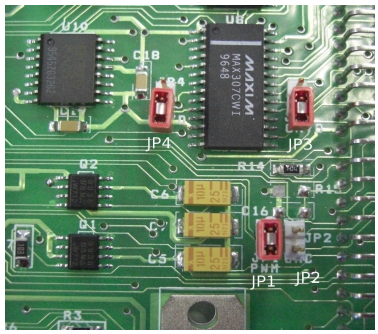


Figure 6: Jumpers on hardware version 2.0

3.4.1 Main Boards Jumpers (hardware version 2.2)

The Main Board version 2.2 has two jumpers, designated JP2 and JP3 which are the analogous of Jumper 3 and Jumper 4 of version 2.0. They are normally left OPEN. There is another 3 position jumper designated J2 that is used to select the output circuitry for the BIAS LED source. If the jumper is set between pin 1and 2, the output to the -BIAS_LED pin will be a voltage set by the MAX532 D/A converted amplified by 4.0. If the jumper is set between pins 2 and 3, then the -BIAS_LED pin is connected to an OPAM configured as current source (0 to 25mA).

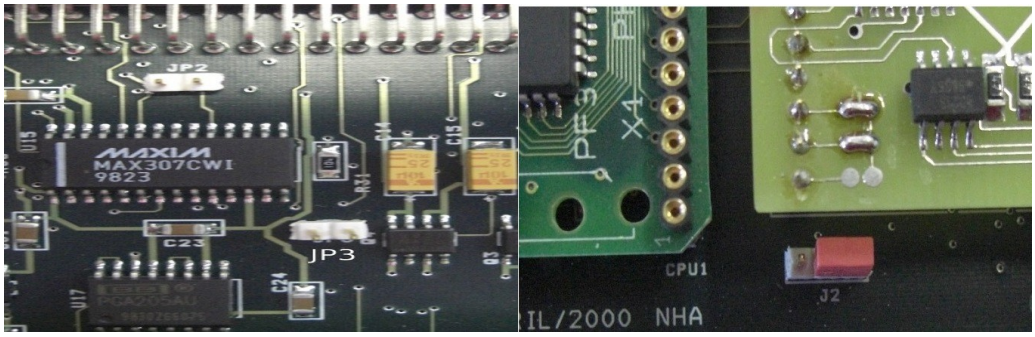


Figure 7: Jumpers hardware version 2.2

4 USER INTERFACE

PULPO has two ways to communicate with users, one is the Front Panel, and the other is the serial port (RS232).

4.1 Front Panel

The front panel consist of a two lines 16 characters liquid crystal display (LCD) with three keypads (arrow UP, arrow DOWN and EDIT). The keypads allows users to scroll through the sensors readout or to change the **Display** mode to **Menu** mode for parameter edition. There are also three LEDs to signal special conditions, as Shutter Open (green), Shutter Close (yellow), heater operation (RED). These LEDs can be disabled via Menu mode or through serial command (*LD*). By default, when PULPO is power cycled, the LED are disabled

Besides the above mentioned elements, the front panel has 4 holes on the right side to allow access to 4 potentiometers. The lower one (R5 on FRONT PANEL V2.0 schematics) is used to adjust the LCD contrast, the immediate upper one (R4) serves to adjust the over current trip point for the heaters. To do that, select Menu Mode 1 and press the upper arrow key till the ***Imax*** label appears , now start to turn the potentiometer to set the over current trip point to the desired value.

Remember that on ESO cryostat, heaters 1 and 2 are connected in parallel and they are run in PWM mode, so you must set the trip current to at least the maximum current of heater 1 plus the maximum current on heater 2 (~ 650 mA). If the trip point is set too low, the heater will not work properly

The upper 2 potentiometers (R3 and R2) were put for future implementations of shutter interface boards and are not installed.

4.1.1 Display Mode

Normally, the LCD is displaying the sensors values like temperature, vacuum level, heaters power, heaters current, etc. The arrow up/down keys allow the user to display the different sensors readouts. In this mode, the Edit key has no effect except when displaying the temperatures associated with heaters 1,2 or 3, in which case the display format is changed, showing side by side, the sensor temperature and the reference temperature associated.

The temperatures are shown in Kelvin scale but the user can change this to Celsius using the last option of Menu #3 which toggle between both scales.

The LCD content in Display mode is:

Line	Variable	Unit
1	PT100 #1	K or C
2	PT100#2	K or C
3	PT100#3	K or C
4	PT100#4	K or C
5	PT100#5	K or C
6	PT100#6	K or C
7	Reference 100ohm	K or C
8	Heater Current (sum)	mA
9	Heater 1 Power	Watts
10	Heater 2 Power	Watts
11	Heater 3 Power	Watts
12	Vacuum Gauge	mBar
13	Maximum readable temperature	K

Please note that the Heater Current, displayed in line 10 is the sum of the three heaters. With the current hardware implementation, it is not possible to measure independently the current for each heater, unless some of the heaters are disabled via MENU#1 or through the serial interface.

The reference temperature, displayed in line 7, is in reality an internal 100 ohms 0.01% resistor which is used to calibrate the internal current source used to read the external PT100.

The current source (J505) is a JFET current source that should deliver 1 mA +/- 20%. Due to this big dispersion and to avoid also drifts induced by temperature inside PULPO, a high precision resistor is used to measure every 80ms the current delivered by the current source and correct the PT100s readout. Starting with firmware version 3.56, the Display Mode include (lines 13) the maximum temperature that PULPO can read. The temperature set point must be lower than this value.

This value can also be read through the serial port command, SE,8. Check the annex XX for further information on this command.

4.1.2 Menu Mode

If the user wants to program some of the PULPO functions, he/she must go into MENU mode, depressing momentarily both arrow keys at the same time. The upper line of the LCD will read MENU 1 and the second one will display the status of HEATER1. Depressing the upper or lower key will change the variable/status displayed on the second line, keeping the first line inalterable. To edit floating point or integer values, like temperature setting, the user has to press one of the arrow keys

PLUS the Edit key. Depending on the arrow key selected, the value will be incremented or decremented. String values like ON, OFF, ENABLE, DISABLE, etc are modified pressing the Edit key alone.

There are three Menus modes (MENU1, MENU2 and MENU3). Menu1 contains all the variables and settings associated with the 3 heaters, like heater enabling/disabling, temperature set-point, maximum heaters current, etc. Menu2 contains information related with LED enable/disable and PULPO alarms. Menu3 is used for manual shutter operation, bias LED current level setting, etc.

Once the Menu Mode is selected, there are two ways to return to Display Mode: either wait 1min with no activity on the key pads or press one arrow key, wait 2 seconds and then press also the other arrow key.

To obtain a more detailed information on the MENU modes, look into Annex B.

4.2 Serial port

The serial channel (RS232 @ 9600 bauds) allows user to fully control/monitor PULPO. Through it, CCD temperature, shutter status, dewar pressure, system status are accessible to the CCD controller or LCU. Communication. The serial port is commanded by the controller/LCU, with PULPO acting as an slave.

The serial command protocol consists of a 2 ASCII character command, followed by parameters separated by commas. A complete description of all commands implemented on software version 3.56 can be found on Annex X.

Depending if PULPO is connected to a dumb terminal or to an intelligent host, the serial port has two different ways of operation. The first mode is called Terminal Mode (TM), and when selected, PULPO will echo every character received through the serial port adding a Line Feed when a Carriage Return (CR) is detected in the input string. This mode is normally used when PULPO is controlled from an hyper-terminal
The second mode is called Controller Mode, and in this case, PULPO doesn't echo back any character and no line feed (LF) is added at the end of the string. This mode optimises the throughput of the serial link. This is the default mode when PULPO is power cycled and is the one used for the communication between the SLCU and PULPO.

5 PULPO SOFTWARE

PULPO software is written in C51, a C compiler optimized for microprocessors of the 8031 family. The software is a combination of some Finite State Machines combined with Interruption Driven Routines.

5.1 Finite State Machines

5.1.1 Automata_KUI

This automata check the status of the front panel keypad, and refresh/update the display.

5.1.2 Automata_ATuner

In case the Autotuner routine is enabled, it takes control of the heater and generates an oscillation in around the temperature set point. Then analyse the period and amplitude of the oscillation and set the optimum PID parameters.

5.1.3 Automata_APID1

Once every second check the actual temperature of the control sensor of PID number 1 (associated with heater 1) and update the proportional error, integral error and derivative error. Then apply the new power to the heater for the next second.

5.1.4 Automata_APID2

Once every second check the actual temperature of the control sensor of PID number 2 (associated with heater 2) and update the proportional error, integral error and derivative error. Then apply the new power to the heater for the next second.

5.1.5 Automata_APID3

Once every second check the actual temperature of the control sensor of PID number 3 (associated with heater 3) and update the proportional error, integral error and derivative error. Then apply the new power to the heater for the next second.

5.1.6 Automata_Alarms

Check the current temperature and vacuum values for all the available sensors and compare them against the alarm trip points. If any value is out of range, it set the appropriate alarm bits and activate the buzzer and output relay to signal the alarm condition.

5.2 IRQ Driver

The IRQDRIVER module contains all the IRQ handling routines to respond to the following interrupts.

- Timer0: used to generate IRQs each 10msec to sincronize the sensor readout, and update the PWM control to the heaters. It also check the state of the keypad and set the values in a byte for further analysis.
- Timer1: used to measure the delay in the between the open and close command and the arrival of the corresponding status.
- Timer2: used to count down the 1000Hz pulses coming from the RTC and do the exposure timing.
- Ext3: detects the arrival of the OPEN Status
- Ext4: detects the arrival of the CLOSE status
- RX/TX: detect the arrival of any character through the RS 232 port and store it in a temporal buffer till a [CR] is detected.

The IRQDRIVER module also contain some routines to initialize the serial port, initialize PULPO, perform a software reset of PULPO, etc.

6 PULPO STATUS BITS

To keep record of the current status of different operations and also flag errors that might have been detected, PULPO software uses 9 bytes, called StatusBits[n], with n ranging from 0 to 8. Each bit on those status bytes has a special meaning which is explained in detail on Annex A. The serial command SB,n (n from 1 to 9) must be used to access the content of each one of these status bytes.

A careful reading of these annex is recommended to better understand the functioning of Alarms, Shutter, etc.

6.1 Example of SB command when in Terminal Mode

When PULPO is in Terminal Mode, the SB command has a formatted output that helps to understand the meaning of each bit. This is not available in Command Mode, in which case the requester has to perform the logical AND operation to determine the state of each bit in the word. The folowing list is the real output to the SB commands

```
sb,1
LEDe|Dion|Bzon|LNal|VAal|----|----|TAla
  1    0    0    0    0    0    0    0
sb,2
SErr|SRav|SCav|SOav|ShNc|ExAc|ExPa|ShOp
  0    1    0    0    1    0    0    0
sb,3
S5nc|S4nc|S3nc|S2nc|S1nc|H3nc|H2nc|H1nc
  1    1    1    1    1    1    1    1
sb,4
TuAc|DePa|P3on|P2on|P1on|VAnc|S7nc|S6nc
  0    0    1    1    1    1    0    1
sb,5
HOcu|BLon|SRem|SnDe|LOac|Tcse|Rcse|RTCe
  0    0    0    0    1    0    0    0
sb,6
WRst|----|AGEn|LnEn|----|----|----|TEna
  0    0    0    1    0    0    0    0
sb,7
VAEn|S1En|S2En|S3En|S4En|S5En|S6En|----
  0    0    0    1    0    0    0    0
sb,8
AvOn|A1On|A2On|A3On|A4On|A5On|A6On|----
  0    0    0    0    0    0    0    0
sb,9
----|----|----|SNFa|SRem|ShCs|ShOs|SCtr
  0    0    0    0    0    0    0    1
sb,10
HAvO|HA1O|HA2O|HA3O|HA4O|HA5O|HA6O|----
  0    0    0    0    0    0    0    0
```

7 ALARM SYSTEM

PULPO has one relay output and an internal buzzer which are used for signalling an alarm condition. The relay output is normally connected directly or through an alarm concentrator and signalling module like the SELCO to Paranal CAS which distribute the alarm through a paging system. Also the buzzer is activated and a flashing message on the front panel is displayed. To acknowledge an alarm, it's enough to press any of the three keypads in the front panel. This is only an acknowledge to stop the buzzer and close again the relay, but if the alarm condition is still active, an internal status bit will reflect the condition. The serial command **RA** (Reset Alarm) can also be used to remotely acknowledge the alarm condition.

7.1 Proper handling of a PULPO alarm

- Press any front panel key to acknowledge alarm. The alarm acknowledge will stop the buzzer and close the relay. If this action is not performed, the buzzer and relay will stay in the activated mode even if the condition that generated the alarm has disappeared.
- Read the LCD to discover which condition produced the alarm

- Take actions to fix the problem. While the alarm condition is valid, the front panel LCD will continue flashing.

7.2 Types of Alarms

PULPO firmware V3.56 can generate alarms due to

- Temperature on sensors 1 to 6 over the defined threshold
- Vacuum over threshold limit

There are enabling bits for all the alarms, and also a global alarm enable. The user can enable each individual alarm or disable all of them at the same time.

7.3 Setting the alarm threshold

The alarm threshold for every sensor can be set via the front panel Menu 2 or via the serial port.
The command **TT,n,t** sets the temperature (in Kelvin) threshold for sensor **n** at temperature **t**
To set the trip point for the vacuum alarm, the command is **VL,v** were **v** is the vacuum trip point in mBar. To read the current trip point setting, the command can be issued without the last parameter

Ex:
TT,1,148 sets the threshold level for sensor 1 at 148 K

VL,1e-4 sets vacuum alarm trip point for vacuum level worse or equal to 1 x 10-4 mBar .

7.4 Enabling Alarms

The global and individual alarms can be enabled either through the front panel or via the RS-232 link. In case of using the serial port, the command to be issue is the **A**larm **E**nable (**AE,n,m**), were n stand for the sensor and m must be one for enabling and 0 for disabling. Check for more details in the Annex A and Annex C

7.5 Conditions for generating alarms

For having an alarm generated by PULPO, the following conditions must be met:

7.5.1 Temperature Alarms

- The enable bit for the individual alarm must be on. Each sensor (from temperature sensor 1 to sensor 6) has a bit called **AlarmSn_Enab** (n is an integer from 1 to 6), if this bit is set, the alarm for that sensor is enabled. The bit is set through the serial command **AE,n,1**. To disable or unset the bit, the command **AE,n,0** must be issued.
- The temperature for the sensor has to be over the threshold value. The threshold value for each temperature sensor is set either via front panel or the serial command **TT**, as mentioned above. If the software detects that the temperature for a given sensor is over the threshold value for that sensor, then the **AlarmSn_On** bit will be set, meaning that a condition alarm was detected for that sensor.
- The **TAlarm** bit, on status byte 0 must be set. This is automatically done if for any sensor both the **AlarmSn_Enab** and **AlarmSn_On** are set.
- The global alarm enabling bit must be set. This is done via front panel or using the serial command **EA,0,1**

7.5.2 Vacuum Alarms

- The enable bit for the vacuum alarm, **Valarm_Enab**, must be set
- The vacuum level must be higher than the trip point set by command **VL** (check annex C) or through the front panel Menu. This will set the **AlarmVa** bit on byte StatusBits[7]
- **VAlarm** bit set on byte StatusBits[0]. This bit is set by the routine CheckForAlarms if the previous two conditions are met
- Global alarm enabling bit must be set.

8 HARDWARE INTERFACING

8.1 Temperature Sensing

Up to 6 Pt100 temperature sensors can be interconnected to PULPO. The 6th Pt100 sensor is reserved for implementing the LN2 exhaustion alarm, the other five sensors can be freely distributed inside the CCD cryostat or outside to measure ambient temperature. The inputs for this five sensors are available on the DB37 connector on the PULPO back plate. The naming convention used for them is PT1 till PT20, in which each group of 4 correspond to one Pt100 sensor. So PT1, PT2, PT3 and PT4 are used for connecting the first Pt100, PT1 and PT4 are the current leads and PT2 and PT3 are the sensing leads.

PULPO uses an 8 channels analogue multiplexed stage to select the sensor to be readout. One of the 8 channels is connected to an internal 100 0.1% ohms reference resistor, this resistors allows to calibrate the 1mA current source that excites the PT100s. Other channel is dedicated to the vacuum sensor, so there are 6 input left to temperature sensing.

When one of the temperature sensor is selected, 1mA of current circulates through it, generating a tiny voltage which is differentially amplified (G=72.62) inside PULPO. The amplifying stage is digitally controlled as the vacuum gauge output voltage need a lower gain (G=9.078) to be accommodated inside the ADC input range (0-10V). Each 10msec, the micro-processor inside PULPO monitors 1 sensor plus the reference resistor. A digital filtering algorithm is used to update the sensor temperature, removing any noise that might still be present in the signal.

8.2 Vacuum Sensing

PULPO can be interfaced to Balzer Compact Full Range Gauge, model PKR 250 and also to Edwards Wide Range Gauge, model WRG-D-NW25. These gauges can measure vacuum in the range off 1000 to 5 x10-9 mbar. Over the whole measuring range, the measuring signal is output as logarithm of the pressure. The operating voltage supplied to the gauge should be between 15VDC to 30VDC with less than 1Vpp of ripple.

PULPO feeds 24VDC to the gauges, and read back the gauge's signal output using channel 8 on the analogue multiplexer. The voltage is converted to a digital word using the same 16 bits ADC used for the temperature readout. As the vacuum to voltage conversion is different for Balzer and Edwards, PULPO has to be configured to use the proper conversion formulae. This is done with the command **VI,n**. If n is equal 1, then PULPO use the Balzer formulae, and if n=2, the Edwards formulae is used.

The vacuum gauge power supply can be shut down using the serial command **VA,0**. To turn it on again, send command **VA,1**.

The output of the vacuum gauge is accessible through the PULPO front panel LCD or through the RS-232 command **SE,8**.

8.3 Temperature Control

PULPO has 3 independent heaters, each one able to deliver up to 7W into a 75 ohms load. The first 2 heaters are implemented using power MOSFET transistor that switch ON or OFF (PWM) the 24VDC into the load. On hardware version 2.0, the third heater can be either linear (DAC controlled current source) or ON/OFF via MOSFET. The selection for this heater operation mode is done via jumper on the main board. On hardware version 2.2, the 3 heaters are implemented through MOSFET working in PWM.

The temperature control is accomplished using a digital **P**roportional **I**ntegrative and **D**erivative (PID) algorithm. Once per second, the actual temperature is compared against the reference temperature (set point). The error between these two quantities is used to generate a correction signal which is applied to the heaters circuitry. The Proportional part of this signal depends only on the difference between the actual and the reference temperature, the Integrative part takes into account the history or evolution of this error, and finally the Derivative part evaluates the speed of change in the error. With suitable parameters, the PID algorithm guaranties no error on the controlled variable.

The parameter values for the PID depends on characteristics of the system under control, and they can be determined in various ways. Specifically PULPO can use an Auto-tuning routine which consist in producing critical oscillations on the detector temperature and then determines the period and amplitude of these oscillations to compute the PID parameters. Once the optimum parameters are determined, PULPO starts automatically to control the temperature.

To avoid damages to the detector due to PULPO malfunction, hardware and software protections were implemented on the system.

There are two conditions that prevent the heaters operation:

1. When the CPU hangs up and doesn't trigger the heaters watch dog circuitry. In this case, the output current is hardware disabled and PULPO has to be reinitialized to re-establish proper operation.

2. Heater over-current detection. The current delivered to heaters is monitored by PULPO using a precision high side current sense amplifier (MAX471), this chip, with a 10Kohm load, generates a voltage proportional to current (5 V/A). This signal is used to trigger an over-current protection circuit that will trip whenever the output current is greater than the trip point set through the front panel potentiometer (Imax=999mA). The same signal is also read using a 10 bit DAC and displayed on the front panel LCD.

9 PULPO START UP SEQUENCE

9.1 Initialisation Sequence

When the 24VCD power is applied to PULPO, the micro-processor starts an initialisation routine which checks the hardware operation and test the peripherals attached to PULPO (shutter drivers, vacuum gauge etc..). Messages are displayed into the front panel LCD and setting of internal flags are performed during the initialisation.

Important: make sure to have all the peripherals connected to PULPO before turning the power ON, otherwise, some internal flags and status will show wrong values.

9.2 Shutter Test

This is the first test done after the power is applied to PULPO, it is also performed after a **RS** (Reset Shutter) command is received through the serial link. During this test, the PULPO micro-controller perform the following actions:

Read the shutter piggy board code, the code is made up of 3 bits, with code 111 reserved for no board.

Code 000 is used for an IRIS shutter driver (never implemented).

Code 001 is used for an opto-isolated I/O board meant to be connected to external shutter drivers like SESO. Others codes are still available for future shutters boards. Once the shutter board is read, PULPO checks the compatibility of this board with the shutter type to be controlled, this shutter type is stored in the variable *ShutterIdentifier* inside PULPO. The *ShutterIdentifier* value can be set through the serial link, using the command **SI,n**. The values for n are:

VALUE	NAME	DESCRIPTION	RS232
0	NO_SHUTTER_BOARD	PULPO does not control any shutter	SI,0
1	BOARD1_NO_STATUS	opto-isolated module with no shutter status available	SI,1
2	IRIS_NO_STATUS	IRIS shutter with no status available	SI,2
3	IRIS_OPEN_NOCLOSE	IRIS shutter with only OPEN status available	SI,3
4	IRIS_OPEN_CLOSE	IRIS shutter with OPEN and CLOSE status available	SI,4
5	IRIS_CLOSE_NOPEN	IRIS shutter with only CLOSE status available	SI,5
6	TWOBLADES_OPEN_NOCLOSE	opto-isolated module inside PULPO and external two blades shutter driver with only OPEN status available (ex: SUSI II)	SI,6
7	SESO	SESO shutter driver (ex: VLT TEST Camera)	SI,7
8	EMMI	EMMI shutter driver	SI,8
9	WFI	Wide Field Imager	SI,9
10	EFOSC	EFOSC	SI,10
11	FORS	FORS1 and FORS2	SI,11

ShutterIdentifier 1,6,7,8,9, 10, 11 needs PULPO fitted with the opto-isolated (code 001) shutter board, *ShutterIdentifier* 2,3,4,5 needs shutter board code 000.

9.3 Heater Test

The Heater Test, also available through **RH** command, starts by turning off the heater MOSFET and measures the offset current (should be very small) and then activates each heater MOSFET to ON (~300mA) and measuring back the current to determine if the heaters are connected or not. The current measurement is also used to compute the heater resistance.

9.4 Pt100 Test

Once the Heater Test is finished, PULPO test the Pt100s, checking which sensors are connected and sets the status bit inside PULPO RAM. This test can be executed through the RS command

9.5 Vacuum Gauge Test

For this test, also available through **RV** command, PULPO turns on the power to the vacuum gauge and after waiting a couple of seconds, read back the vacuum level. If the vacuum level is better than 0.5x10-9, PULPO assumes the vacuum gauge is not connected.

After the initialisation routine is finished, PULPO goes into display mode. In this mode, the LCD refresh each ¼ of seconds the temperature readings, current to heaters, and vacuum gauge level.

For a proper reading of the vacuum gauge, the command **VI,n** must be issue through the serial link or select the appropriate gauge on Menu 3

10 PREPARING PULPO FOR OPERATION

To set up PULPO for operation, the user must have access to a terminal connected to PULPO via RS232 (9600 bauds) or make use of the command window of PULPO Maintenance Panel.

10.1 Set Date and Year

Use command **ST,hh:mm:ss** and **SD,dd/mm/yy** to load the on board real time clock. This is necessary to have the logged data properly tagged.

10.2 Set Up PIDs

10.2.1 Disable Heaters

Start disabling all heaters with:

HE,1,0
HE,2,0

HE,3,0

or use Menu Mode 1 in the front panel display.

10.2.2 Disable Alarms

To avoid annoying buzzer and front panel warnings while the dewar variables (temperature, vacuum, LN2 exhaustion detector) are not stabilised , it is recommended to make a global alarm disabling with command **AE,0,0**. Take the opportunity to set the trip points for the temperature and vacuum alarms. In steady state operation, PULPO is capable of maintaining the reference temperature with less than 0.1K, so the trip point can be set 0.5 or 1K higher.

```
TT,1,150.5      // set point for sensor 1 is 150K and alarm trip point is at 150.5K
TT,2, 330       // trip point for unused sensors set at very high value
TT,3,330
....
VT,1e-4         //vacuum alarm trip point at 1e10-4 mB
```

10.2.3 Heaters Interconnection

Tell PULPO how the heaters are interconnected, using command **DM,n**. The ESO cryostats Generation 3 (VLT) have heater 1 and 2 working in parallel, and heater 3 is not used. So issue **DM,1** or set PULPO front panel in Menu Mode 1 scroll till CABLING is displayed end press the EDIT key until **1=2 3** label is displayed.

10.2.4 Associate Pt100 sensor to Heaters

To control the detector temperature through the PID controller programmed in software, we need to associate one Pt100 sensor to each heater. In this way, the software will read this sensor and compare its value with the reference value programmed either via RS232 link or through the PULPO front panel.
This operation can only be done through the serial link. Use the command Control Sensor (**CS,n,m**), to tell PULPO that control sensor m is associated to heater n.

If heater 1 and 2 are working in parallel, they must have the same temperature sensor associated, so give commands **CS,1,1** and **CS,2,1** if both are using sensor 1 as control sensor (Generation 3 cryostat).

10.2.5 Setting Heaters mode operation

Set the heaters to work in PWM mode with 1 sec maximum duty cycle. For that issue commands:

```
HM,1,1
HM,2,1
HM,3,1
```

10.2.6 Set PIDs Set Points

Load the set point reference for the three heaters via Menu Mode 1 or commands **SP,n,f**. If there are heaters working in parallel (same reference Pt100 sensor) make sure to set the same set points for them.

SP,1,150 // set point for PID controlling heater 1 is 150K
SP,2,150 // set point for PID controlling heater 2 is 150K
SP,3,160 // set point for PID controlling heater 3 is 160K

10.2.7 Check trip point for Heater Over Current

Each heater delivers ~300mA to the cryostat 75ohms resistors, so is important to set this limit at least a 10% higher than the combined current of all the heaters connected. This is done setting the front panel display in Menu Mode 1 and scroll it till IMax value is displayed. Then start turning the appropriate potentiometer (Fig 3.1) to increase or lower this value. If only 2 heaters are in use, you can set Imax to ~700mA, but if the 3 heaters are in use, set IMax to 999mA.

10.2.8 Set maximum temperature gradient

Use command TS,f or the Menu Mode 1 to set the maximum variation in temperature that PULPO will allow in any of the controlled variables. By default this value is set to 5.0 K/min. This function is useful when cooling down the detector, as it will activate the heaters to avoid a decreasing in temperature bigger than the value set by TS. If TS is equal 1, then detector will lower its temperature at a maximum rate of 1 K/min.

Example: **TS,5.0**

10.2.9 Set values for PID parameters

As a first approximation set all the PID proportional constant to 50 and the integral constant to 200. Maintain the derivative constant always at 0.

KP,1,50 //Proportional gain of PID 1 set to 50
KI,1,200 //Integral gain of PID1 set to 200
KD,1,0 //Derivative gain of PID1 set to 0
KP,2,50
KI,2,200
KD,2,0
KP,3,50
KI,3,200
KD,3,0

This setting can also be done through Menu Mode 1 on the front panel display.

Another alternative, when using heater 1 or heater 1 and 2 in parallel, is to activate the AutoTuner routine. For that, the cryostat must be connected to PULPO and filled with LN2. Once the temperature to be controlled is near or lower the set point, activate the AutoTuner with **AT,1** or through the control panel Menu Mode 1. The process take some minutes to determine the PID parameters and then automatically enable the heaters to keep the temperature under control.

10.2.10 Enable heaters

Use the Menu Mode 1 or command HE,n,m to enable the wired heaters.

HE,1,1 //Enable Heater 1
HE,2,1 //Enable Heater 2

HE,3,1 //Enable Heater 3

10.2.11 Enable Alarms

Once the temperatures under control are in the steady state regime, you can enable the Temperature alarms.

AE,0,1 // Alarm Global Enable
AE,1,1 // Enable temperature alarm for Pt100 sensor 1
...
AE,7,1 // Enable vacuum level alarm
etc

ANNEX A. PULPO STATUS BITS

PULPO has 6 status bytes located in bit addressable RAM positions, used for maintaining information on PULPO operation. They are accessible through the serial link, using the command ***SB,n*** where n can be any number from 1 to 6.

STATUS BYTE 1	BIT	FUNCTION
TAlarm1	0 (LSB)	Set if any of the temperature sensors is over the alarm trip point
Sp01	1	Spare Not Used
Sp02	2	Not Used
VAlarm	3	Set if vacuum level worse than vacuum limit
LN2Alarm	4	Set if LN2 gas temperature over alarm limit
Buzzer_On	5	Set if Buzzer activated
Dialer_On	6	Set output alarm relay in OPEN state
LED_Enab	7 (MSB)	Set if front panel status LEDs enabled

STATUS BYTE 2	BIT	FUNCTION
ShOpen	0 (LSB)	Set if shutter open
ExpPaused	1	Set if exposure paused by CCD controller (PE,1)
ExpActive	2	Set if exposure active
Shutter_Not_Connected	3	Set if no shutter detected at start-up shutter test
Status_Open_Available	4	Set if OPEN status available at start-up shutter test
Status_Close_Available	5	Set if CLOSE status available at start-up shutter test
Status_Remote_Available	6	Set if REMOTE status available at start-up shutter test
Shutter_Error	7 (MSB)	Set if Shutter error detected at start-up or normal exp

STATUS BYTE 3	BIT	FUNCTION
Heater1_Not_Connected	0 (LSB)	Set if Heater1 not connected at start-up test
Heater2_Not_Connected	1	Set if Heater2 not connected at start-up test
Heater3_Not_Connected	2	Set if Heater2 not connected at start-up test
Sensor1_Disconnected	3	Set if sensor 1 not connected at start-up test
Sensor2_Disconnected	4	Set if sensor 2 not connected at start-up test
Sensor3_Disconnected	5	Set if sensor 3 not connected at start-up test
Sensor4_Disconnected	6	Set if sensor 4 not connected at start-up test
Sensor5_Disconnected	7 (MSB)	Set if sensor 5 not connected at start-up test

STATUS BYTE 4	BIT	FUNCTION
Sensor6_Disconnected	0 (LSB)	Set if sensor 6 not connected at start-up test

Sensor7_Disconnected	1	Set if sensor 7 not connected at start-up test
Vac_Sensor_Disconnected	2	Set if vacuum gauge not connected at start-up test
PID1_On	3	Set if PID for heater 1 is working
PID2_On	4	Set if PID for heater 2 is working
PID3_On	5	Set if PID for heater 3 is working
DefaultPIDpar	6	Set if PIDs have default Kp, Ki, Kd constant. This happens when PULPO detect a corrupt internal database at start-up time.
Tuner_Active	7 (MSB)	Set if PID AutoTuner for heater 1 and 2 is active.

STATUS BYTE 5	**BIT**	**FUNCTION**
RTCError	0 (LSB)	Set if PULPO detect errors in the Real Time Clock (RTC)
RAMCheckSumError	1	Set if PULPO internal RAM has check sum error
RTCRAMCheckSumError	2	Set if RTC internal RAM has check sum error
LogRunning	3	Set if PULPO logger routine is running
Shutter_NoDefect	4	Set if SESO shutter driver has DEFECT status not asserted
Shutter_Remote	5	Set if external shutter driver is in REMOTE
BiasLedOn	6	Set if BIAS LED is ON
HeaterOverCurrent	7 (MSB)	Set if heaters over current detected

STATUS BYTE 6	**BIT**	**FUNCTION**
TAlarm1_Enab	0 (LSB)	Set if temperature alarm enabled
Spare51	1	Spare Not Used
Spare52	2	Spare Not Used
Spare53	3	Set if vacuum alarm enabled
LN2Alarm_Enab	4	Set if LN2 exhaustion alarm enabled
Alarm_Global_Enab	5	Set if Global Alarm enabled
Spare56	6	Spare reserved for future used
WatchDogReset	7 (MSB)	reserved for future used

STATUS BYTE 7	**BIT**	**FUNCTION**
Spare60	0 (LSB)	Spare Not Used
AlarmS6_Enab	1	Enable Sensor6 Alarm
AlarmS5_Enab	2	Enable Sensor5 Alarm
AlarmS4_Enab	3	Enable Sensor4 Alarm
AlarmS3_Enab	4	Enable Sensor3 Alarm
AlarmS2_Enab	5	Enable Sensor2 Alarm
AlarmS1_Enab	6	Enable Sensor1 Alarm
VAlarm_Enab	7 (MSB)	Enable Vacuum Alarm

STATUS BYTE 8	BIT	FUNCTION
Spare70	0 (LSB)	Spare Not Used
AlarmS6_On	1	Set if Sensor6 Alarm On
AlarmS5_On	2	Set if Sensor5 Alarm On
AlarmS4_On	3	Set if Sensor4 Alarm On
AlarmS3_On	4	Set if Sensor3 Alarm On
AlarmS2_On	5	Set if Sensor2 Alarm On
AlarmS1_On	6	Set if Sensor1 Alarm On
AlarmVa	7 (MSB)	Set if Vacuum Alarm On

STATUS BYTE 9	BIT	FUNCTION
ShutterOpenCnrt	0 (LSB)	Set if temperature alarm enabled
ShOpStatus	1	Shutter Open Status (1 if open, refreshed every 10ms)
ShClStatus	2	Shutter Close Status (1 if closed, refreshed every 10ms)
ShRemStatus	3	1 if Shutter driver in Remote mode
ShNoFaultStat	4	Shutter Driver Not in Fault state
Spare85	5	Not used
Spare86	6	Not used
Spare87	7 (MSB)	Not used

STATUS BYTE STATUS BYTE 10	BIT	FUNCTION
Spare100	0 (LSB)	Spare Not Used
History Alarm S6	1	1 if S6 has triggered an alarm since last RA command
History Alarm S5	2	1 if S5 has triggered an alarm since last RA command
History Alarm S4	3	1 if S4 has triggered an alarm since last RA command
History Alarm S3	4	1 if S3 has triggered an alarm since last RA command
History Alarm S2	5	1 if S2 has triggered an alarm since last RA command
History Alarm S1	6	1 if S1 has triggered an alarm since last RA command
History Alarm Vac	7 (MSB)	1 if Vac has triggered an alarm since last RA command

ANNEX B. MENU MODES

Below there is a detailed explanation of each item accessible through the front panel Menu Modes. The column with RS232 heading shows the serial command available, if any, to perform the same action.

Menu Mode 1

VARIABLE	VALID RANGE	FUNCTION	RS232 COMMAND
Heater1	ON / OFF	Turn ON/OFF PID for Heater 1	HE,1,1 / HE,1,0
Heater2	ON / OFF	Turn ON/OFF PID for Heater 2	HE,2,1 / HE,2,0
Heater3	ON / OFF	Turn ON/OFF PID for Heater 3	HE,3,1 / HE,3,0
Ref T1	100.0 to 333.0 K	Set reference temperature for Heater1	SP,1,f
Ref T2	100.0 to 333.0 K	Set reference temperature for Heater2	SP,2,f
Ref T3	100.0 to 333.0 K	Set reference temperature for Heater3	SP,3,f
PIDTuner	ON / OFF	Turn ON/OFF Auto Tuner routine	AT,1 / AT,0
Tslope	0.5 to 10.0 K/min	Set the maximum rate of change on the controlled temperatures (heaters 1,2 and 3)	TS,f
Imax	0.0 to 999 mA	Display (set using potentiometer in front panel) the threshold heater current to trip the heater over-current protection. This correspond to the **sum** of the three heaters.	
KProp 1	0 to 1000	Set value for PID 1 proportional constant	KP,1,f
KInt 1	0 to 1000	Set value for PID 1 integral constant	KI,1,f
KDeri 1	0 to 0	Set value for PID 1 derivative constant	KD,1,f
KProp 2	0 to 1000		KP,1,f
KInt 2	0 to 1000		KI,2,f
KDeri 2	0 to 0		KD,2,f
KProp 3	0 to 1000		KP,3,f
KInt 3	0 to 1000		KI,3,f
KDeri 3	0 to 0		KD,3,f
CABLING	1 2 3 1=2 3 1 2=3 1=3 2 1=2=3	Three heaters working independently Heaters 1 and 2 working in parallel, heater 3 independent Heater 2 and 3 working in parallel, heater 1 independent Heater 1 and 3 working in parallel, heater 2 independent Heaters 1, 2 and 3 working in parallel	DM,0 DM,1 DM,2 DM,3 DM,4

Menu Mode 2

VARIABLE	VALID RANGE	FUNCTION	RS232 COMMAND
LEDs	ENAB / DISAB	Enable or disable front panel LEDs	
ALARMS	ENAB / DISAB	Global enable/disable for alarms	AE,0,0 / AE,0,1
TALARM	ENAB / DISAB	Enable or disable temperature alarm	AE,n ,0/1 n=1..6
VALARM	ENAB / DISAB	Enable or disable vacuum alarm	AE,7,0/1
LN2ALARM	ENAB / DISAB	Enable or disable LN2 exhaustion alarm	AE,8,0/1
T1 LIMIT	100.0 to 350.0K	Set trip point for sensor 1 temperature alarm	TT,1,f
T2 LIMIT	100.0 to 350.0K	Set trip point for sensor 2 temperature alarm	TT,2,f
T3 LIMIT	100.0 to 350.0K	Set trip point for sensor 3 temperature alarm	TT,3,f
T4 LIMIT	100.0 to 350.0K	Set trip point for sensor 4 temperature alarm	TT,4,f
T5 LIMIT	100.0 to 350.0K	Set trip point for sensor 5 temperature alarm	TT,5,f
T6 LIMIT	100.0 to 350.0K	Set trip point for sensor 6 temperature alarm	TT,6,f
VA LIMIT	1.0e-9 to 1.0 mbar	Set trip point for vacuum alarm	VL,f

Menu Mode 3

VARIABLE	VALID RANGE	FUNCTION	RS232 COMMAND
SHUTTER	OPEN / CLOSE	Open or close the shutter	OS / SC
BUZZER	ON / OFF	Turn on/off the internal buzzer	
BIAS LED	ON / OFF	Turn on/off the bias LED	
BIAS POWER	0 to 100%	Set current level to bias LED (0 to 25mA)	BP,f (f: 0..100)
KELVIN	ON / OFF	Set temperature display mode in Kelvin or Celsius	
GAUGE	EDWARDS BALZER	Defines the formula to convert from voltage to mBar	Vl,n (n=1 => Balzer n=2 =>Edwards)

Note: on the RS232 column, f stand for a floating point value.

ANNEX C. SERIAL COMMANDS (Firmware Version 3.56).

Below you will find the complete list of serial commands recognised by PULPO. The commands consist of two alphabetic characters follows by arguments. The end of the command string is always a carriage return (ASCII 0x13). The answer from PULPO is always a OK plus parameters if any.

The example shows in `Courier` font the commands sent by the controller and in ***`Courrier Bold Italic`*** the answer from PULPO.

AE Alarm Enable

```
Enable alarm warning for the various sensors attached to PULPO

syntax: AE,n[,m]        n=0,8 m=0,1
where       n=0 => all sensors
            n=1 => Temperature Sensor 1 Alarm
            n=2 => Temperature Sensor 2 Alarm
            n=3 => Temperature Sensor 3 Alarm
            n=4 => Temperature Sensor 4 Alarm
            n=5 => Temperature Sensor 5 Alarm
            n=6 => Temperature Sensor 6 Alarm
            n=7 => Vacuum gauge Alarm
            n=8 => LN2 alarm

            m=0 => disable alarm
            m=1 => enable alarm
if the parameter m is not included, PULPO will return the enable status for
the given sensor.

example:    AE,0,0      //disable all alarms
            OK
            AE,1        // read alarm enable status
            OK,0        // Alarm for sensor 1 disabled

related commands: TT (Temperature Alarm Trip point)
                  VL  (Vacuum trip Level point)
                  RA  (Reset Alarms)
```

AV AVerage

```
Computes the average value and standard deviation for temperature sensors.
The values are in ADUs and not converted to temperature.
syntax: AV,n[cr]

example:    AV,1
            OK,42350,10.5
```

AT Auto Tuner

```
Turns the PID Auto Tuner routine ON or OFF. This routine allows PULPO to
determine the PID parameters for heater 1 or 1 and 2 when they are
```

connected in parallel. To use it, wait till the cryostat has reached 2 or 3 degrees below the desired set point and activate the AutoTuner. It will take some minutes to compute the PID values.

syntax: AT[,n] n=0,1

example:
```
AT,1        //start AutoTuner
OK
AT,0        //stop AutoTuner
OK
AT          //check AutoTuner status
OK,0        //AutoTuner off
```

related command: KP, KI, KD, HE, SP

BP Bias Power

Set the current level (%) to the bias LED. Full power (100%) is approximately 25mA.

syntax: BP[,f] with f=0->100.0%

example:
```
BP,50       //set LED current to 12.5 mA
OK
BP          //read Power to Bias LED
OK,50.0
```

related commands: SM,2 (Shutter mode 2=> LED operation instead of shutter)

CD Close Delay

Request PULPO to send, via RS232, the last shutter Opening Delay. The answer is in microseconds.

syntax: CD[cr]

example:
```
CD          //request shutter close delay
OK,45000    //Close Delay was 45msec
```

related command: OD, '>'

CM Controller Mode

Set serial link in controller mode. In this mode, PULPO doesn't echo back any character. This is the normal mode for connecting PULPO to the CCD controller.
syntax: CM

CS Control Sensor

This command associates the sensor 'm' to heater 'n'. If the parameter 'm' is missing, PULPO send back the sensor 'm' associated with heater 'n' according to the internal database. For ESO dewars, the normal setting is sensor 1 with heater 1
syntax: CS,n[,m] n:1->3 m:1->7

```
example:    CS,2,1      //sensor 1 associated with heater 2
            OK
            CS,2        //read sensor associated with heater 1
            OK,1
related commands: SP, HE
```

DM DeBug mode

Send information through serial link for debugging purposes. **This command must be issue only when in Terminal Mode (TM), not even in the PULPO Maintenance Panel as the fcdpServer will not recognize the data output from PULPO**

```
syntax: DB[,AutomataNumber]

List of Automatas:
KUI=0       Automata Keyboard User Interface
ATUNER=1    Automata Auto Tuner
APID1=2     Automata PID1
APID2=3     Automata PID2
APID3=4     Automata PID3
AALARMT=5   Automata Temperature Alarm
AALARMV=6   Automata Vacuum Alarm
AALARMH=7   Automata Heater Alarm
END_AUTOMATAS=8

Debug=0.. END_AUTOMATAS => debug only that automata
Debug=END_AUTOMATAS => debug all automatas
Debug=0xFF => no debug information
```

DL Disable Leds

```
Instructs PULPO to disable the front panel LEDs. At startup time the LEDs
are automatically enabled, so one of the first commands sent by the
controller is this one, to avoid light pollution near the detector.

There is no command to remotely enable the LEDs, it has to be done locally
through MENU2

syntax: DL[cr]
```

DM Dewar Model

```
Read back or set the dewar model connected to PULPO. The dewar model is
related with the heaters and sensor wiring inside the vessel. There are 5
(0..4) possible connections of heaters:
      0 => three independent heaters
      1 => 1 and 2 in parallel and 3 independent
      2 => 2 and 3 in parallel and 1 independent
      3 => 1 and 3 in parallel and 2 independent
      4 => 1, 2 and 3 in parallel
Note that the standard heater configuration is heater 1 and 2 working in
parallel (DM=1), and heater 3 does not exist.
```

```
syntax: DM[,m]  m=0..4

example:    DM,1  //set dewar model 1
            OK
```

EC ECho mode

```
Set serial link in Echo Mode. In this mode, all characters received through
the serial port are echoed back. This mode was implemented for testing the
serial link reliability. This mode is abandoned once a carriage return is
detected in the incoming string.
This mode should be used with PULPO connected to the SLCU.

syntax: EC
```

FV Filter Value

```
Read or set the value for the A coefficient in the digital recursive filter
used to calculate the sensors temperature. The low pass filter has the form
of Yi= A*Yi-1 + (1-A)*Xi, were Xi is the sensor readout, Yi-1 is the
previous computed value for the temperature.

syntax: FV[,f]  f= 0.0 -> 1.0
```

HE HEater n ON/OFF

```
Enable or disable the PID routine for heater 'n'. When 'm' is 0, the PID is
turn off, 'm' set to 1 turns the PID ON in normal mode and values for 'm'
equal to 2 and 3 are used by the AutoTuner routine to find out the optimal
PID parameters; these are not meant to be used by the CCD controller or in
terminal mode.

syntax: HE,n[,m]   n:1->3  m:0->3
related commands: CS, KP, KI, KD
```

HM Heater Mode

```
This command defines the sampling time for the PID controller. The normal
value is 1 (1 sec sampling time).
      'm'=0 or 1  => PWM with 1sec of duty cycle (normal mode)
      'm'=2       => PWM with 10sec of duty cycle (rarely used)

syntax: HM,n[,m] n:1->3 and m:0->2

example:    HM,1,1      //set heater1 mode equal 1
            OK
            HM,1
            OK,1
```

HR Heater Resistance

```
This command was used in PULPO previous hardware version (V1.0) to get from
the user the heaters resistance value. It is not longer used, as in the new
hardware version 2.0, PULPO is able to compute this value during the
initialization routine.
```

```
syntax: HR,n[,m] n:1->3 and m:resistance

example:    HR,1          //read resistance for heater 1
            OK,75.6       //resistance equal 75.6 ohms
```

IN INit

```
Initialize PULPO to a known state. This include a complete test to the
hardware attached to PULPO, with messages to the LCD and setting of status
bytes. This command takes more than 30 sec to complete, meanwhile, the user
should avoid any further command.

syntax: IN
```

KP Proportional Constant (kp) for PID controller

```
Read or set the proportional constant for the PID controller 'n'
(associated with heater 'n')

syntax: KP,n[,f]  n:1->3  f: floating point value >= 0.0

example:    KP,1,50.5   //set prop constant for heater1 equal to 50.5
            OK
            KP,1
            OK,50.5
```

KI Integral Constant (ki) for PID controller

```
Read or set integral constant for the PID controller.

syntax: KI,n[,f] n:1->3  f:f. point value >=0.0
```

KD Derivative Constant (kd) for PID controller

```
Read or set derivative constant for PID controller.

syntax: KD,n[,f] n:1->3  f:f. point value >=0.0
```

LO LOg data

```
This command startup the logging routine. Each 't' seconds, PULPO stores in
its internal RAM the values for the sensors specified in the command
string. The internal buffer is defined as circular, so the oldest data is
overwritten.

Before using this command, make sure the time and date are correctly set in
PULPO (the internal RTC drift a bit).

syntax: LO,t,Xn[,Xm[,Xp]  X:T(temperature),P(power),V(vacuum)
      n,m,p:1->7 for Temp, 1->3 for Power and 1 for Vacuum
      Temperature index can go from 1 to 7
      Power index can go from 1 to 3
      Vacuum index must be 1 (only one vacuum gauge can be attached)
```

t:time in sec

example: LO,5,T1,P2,V1 //store each 5sec temperature on sensor 1,power delivered through heater 2 and vacuum level.
OK

LB Log Begin

Set the starting record from which the LD command will start dumping out the logged data. If the circular buffer is not full, then the starting point will be the record 0, and if the circular buffer is full, then the starting point for the dump will be set to the older record available.

syntax: LB

LG Log Check

Output the number of records already written and the space that left on the circular buffer before overwrite occurs.

syntax: LC

example: LC
OK,1300,400 //1300 records already written and 400 to go

LD Log Dump

Send logged information through serial link. When in Terminal Mode, all information is sent with one LD command. In Controller Mode, one record of data is sent for each LD request. **Don't use this command through the PULPO Maintenance Panel.**

syntax: LD

example: LD
01/05 14:58:00 290.0 293.0
01/05 14:58:30 290.1 293.0
etc..

LS Log Stop

Request PULPO to stop the logger. If the previous command was a Log Dump, then the first LS will stop the dumping but PULPO will continue logging data, the second LS will stop the logging.

syntax: LS

LR Log Restart

request PULPO to restart the logger, without overwritten the previous information and using the same input string issue with the last LO command.

syntax: LR

OS Open Shutter

Instruct PULPO to open the shutter. No timing is performed. The shutter will remain open until a Shutter Close (SC) or Initialize (IN) command is received.
This command does not refresh the Open Delay and Close Delay values

syntax: OS

OD Open Delay

Request PULPO to send via RS232 the last Opening Delay. The delay is measured in microseconds.
Note that after the ">" command, PULPO will automatically return the Open Delay. The Close Delay must be requested with the CD command.

syntax: OD

PW set or read PoWer

Read or set the power delivered to heater 'n'. The parameter 'duty' defines the duty cycle with 100% corresponding to ~ 7W for a 75 ohms heater. **If this command is used to set a power output then is important to disable the PID associated with the heater (HE,n,0), as the PID will reset the power to zero after the 1 second sampling rate.**

syntax: PW,n[,duty] n:1->3 duty:0->100(%)

PE Pause Exposure

This command is used to pause ('n'=1) an active exposure or continue a paused exposure ('n'=0). This command works only for scientific exposures.

syntax: PE,n n:0..1

RA Reset Alarm

Set back AlarmHistory to 0x00 and reset the alarm condition. The AlarmHistory is a byte status that stores information on which alarms have been generated since the last RA command.

syntax: RA

RS Reset Shutter

Run a test on the shutter and update the status. It will reset all the error flags or conditions if the shutter is working properly.
The test consist in reading the code of the shutter piggy back card for consistency with the shutter identifier, and then exercise the shutter to check if the OPEN and CLOSE status are available. This information is updated on the StatusBits[1] (can be read with SB,2).

The test takes approximately 30 sec to complete.

syntax: RS

RO Reset OverCurrent on heaters

Reset the hardware heater watchdog (74HCT123) to enable current to heaters. The OverCurrent condition can be detected by the activation of the MSB in the StatusBits[4] (read through SB,5).

syntax: RO

RH Reset Heaters

Perform a test on the heaters connected to PULPO. It sends the maximum current to each heater and determines the resistance. According to this value it set the 3 LSB on the StatusBits[2] (SB,3) to signal if the heater is connected or not.

syntax: RH

RT Reset Temperature sensors

Test all the sensor inputs to determine which ones are connected. It set the status bytes accordingly.

syntax: RT

RV Reset Vacuum sensor

Test the vacuum gauge.
Activates the 24VDC to the vacuum gauge and then read the vacuum. If the value is better than 5 x 10-9 mBar then it is assumed that the vacuum gauge is NOT connected. The StatusBits[3], bit 2 is modified accordingly.

syntax: RV

SA Show triggered Alarms

Show triggered alarms. This command was first implemented on PULPO

syntax: SA

```
Example:    SA
            OK,S2,S9  // alarm active on sensor 1 and 9 (vacuum)
```

SB Status Byte

Send the status byte through the serial port.

syntax: SB,n n:1->6

SI Shutter Identifier

Set or read the shutter identifier connected to PULPO

syntax: SI[,n]

SC Shutter Close

Close shutter. No timing performed.

syntax: SC

SD Set Date for real time clock

Set or read the date in the real time clock

syntax: SD[,dd/mm/yy]

example: SD,01/05/98 //set date to May 5, 1998
OK

SE SEnd value sensor n

Transmit through the serial link the current value for sensor 'n'.
If the sensor is not connected, an error message is generated.

syntax: SE,n
n: 1->7 temperature sensors (K)
n: 8 J505 current source (mA)
n: 9 vacuum sensor (mBar)

The J505 is the current source used inside PULPO to energize the Pt100 sensors. In principle it should be 1.0 mA but there is a dispersion of aprox. +/- 20%

SL Shutter Logic

Defines the logic level to open the shutter. This value should normally be 0 (low TTL level) but depending on the external shutter driver it might be necessary to use a high level (1) instead.
Low TTL level means that a low level will be sent to the external shutter driver to OPEN the shutter.

syntax: SL[,n] n=0,1

SM Set Mode for shutter

Define or read the type of exposure to perform when the following 'start exposure' ('>')are received.

syntax: SM[,n] with n:0->4
n=0 : dark exposure
n=1 : science exposure
n=2 : operate bias LED instead of shutter
n=3 : operate bias LED AND shutter at same time

SP Set Point temperature for Heater n (1 -> 3)

Set or read out the reference temperature (set point in Control jargon) for heater 'n'.

syntax: SP,n[,t] t: 100.0K -> 333.0K

ST Set Time for Real time Clock

Set or read the time on the real time clock installed in PULPO Main Board.

syntax: ST[,hh:mm:ss]

TM Terminal Mode

Set serial link in Terminal Mode. In this mode, all the characters received through the RS-232 are echoed back and a Line Feed is added at the end. This mode is very comfortable when working with PULPO connected to a dumb terminal.

syntaxis: TM

TS Temperature Slope

Set the maximum temperature variation the PID controller will allow on their controlled sensors.

syntax: TS[,f]   f: 0.0 -> 5.0 K/min

TT Temperature Trip point

Set or read back the temperature value for sensor n that will trip the temperature alarm

syntax: TT,n[,f] n:1->6

VI Vacuum Identifier

Defines which type of vacuum gauge is connected to PULPO

syntax: VI[,n] n=1,2

n=1 : Balzer/Pfeiffer type
n=2 : Edwards type

VA VAcuum power

Turn the vacuum gauge off or on, or read the present status.

syntax: VA[,n] n=0,1

VL Vacuum Limit

Defines or read the vacuum limit to trigger the Vacuum Alarm

syntax: VL,f

VS Version Software

Return the PULPO software version

syntax: VS

example: VS
OK,2.42

XD eXposure Delay

Makes a 1 second exposure and send back the open and close shutter delays in one string, 'OK,od,cd', where od is the open delay and cd is the close delay measured in microseconds.

syntax: XD

example: XD
OK,29300,35000

XT eXposure Time

This command loads the internal timer with the count for the next exposure. If the exposure is active, a + or - sign in front of the floating point value will instruct PULPO to increment or decrement the remaining time by the value of 'f'. One restriction of this command is that you can add or subtract only integer values but the initial exposure command accepts floating point values. The maximum exposure time is 65534 sec (~18.2 hrs)

syntax: XT[,[+/-]f f:floting point number (sec)

example: XT,10.5 //load internal timer to perform a 10.5 sec exposure
OK
XT,-3 // decrement the remaining time by 3 sec
OK
XT,+20 //increment the remaining time by 20 sec
OK
XT // request remaining time
OK,26.3 // still 26.3 seconds to go

ANNEX D. PULPO REPLAY AND ERRORS

Any command reply has the following forms:

OK
OK,<parameters>
ER,ErrorNumber

The ErrorNumbers are defined in the file common.h and the current list is

```
#define CMD_OK 0
#define UNDEFINED_CMD 1
#define BAD_PARAMETER 2
#define OUT_OF_RANGE 3
#define SENSOR_NOT_CONNECTED 4
#define NO_EXPOSURE_ACTIVE 5
#define EXP_ALREADY_RUNNING 6
#define EXP_ALREADY_PAUSED 7
#define EXP_UNDEFINED 8
#define RTC_TIMEOUT 9
#define VAC_GAUGE_DEFECTIVE 10
#define LOGGING_OFF 11
#define NO_SENSOR_ASSOCIATED 12
#define NO_SHUTTER_STATUS_AVAILABLE 13
#define RTC_ERROR 14
#define NO_LOG_ACTIVITY 15
#define SHUTTER_OPEN_TIMEOUT 16
#define SHUTTER_CLOSE_TIMEOUT 17
#define VAC_PWR_OFF 18
#define SHUTTER_ALREADY_CLOSED 19
#define NO_SHUTTER_BOARD_CONNECTED 20
#define INSTRUMENT_SHUTTER_UNDEFINED 21
#define SHUTTER_INSTRUMENT_INCONSISTENCY 22
#define NO_INTEGER_VALUE 23
#define OUT_OF_TIME 24
#define PAUSED_NOT_ALLOWED_DURING_DARK 25
#define NOT_TERMINAL_MODE 26
#define NON_EXISTING_SENSOR 27
#define NOT_YET_IMPLEMENTED 28
```

ANNEX E. PULPO REAR PANEL CONNECTORS

Connector LEMO size 2:4
24VDC Power Supply plus sense

PIN	SIGNAL NAME	DESCRIPTION
1	POWER1	24VDC @ 2Amp
2	POWER2	sense+
3	POWER3	GND
4	POWER4	sense-

Connector DB37 female
Heaters, and Pt100 sensors

PIN	SIGNAL NAME	DESCRIPTION
1	HEATER1-	Heater1-
2	POWER HEAT	24VDC, input to heater1,2 or 3
3	PT4	Pt100 #1 current-
4	PT6	Pt100 #2 sense+
5	PT8	Pt100 #2 current-
6	PT13	Pt100 #4 current+
7	PT14	Pt100 #4 sense+
8	PT15	Pt100 #4 sense-
9	PT16	Pt100 #4 current-
10	PT17	Pt100 #5 current+
11	PT18	Pt100 #5 sense+
12	PT19	Pt100 #5 sense-
13	NC	Not Connected
14	NC	Not Connected
15	NC	Not Connected
16	NC	Not Connected
17	NC	Not Connected
18	HEATER2-	Heater2-
19	PT2	Pt100 #1 sense+
20	PT1	Pt100 #1 current+
21	PT5	Pt100 #2 current+
22	PT9	Pt100 #3 current+
23	PT10	Pt100 #3 sense+
24	PT12	Pt100 #3 current-
25	NC	Not Connected
26	NC	Not Connected
27	PT20	Pt100 #5 current-
28	NC	Not Connected
29	HEATER3-	Heater3-
30	POWER HEAT	24VDC, input to heater1,2 or 3

31	PT3	Pt100 #1 sense-
32	PT7	Pt100 #2 sense-
33	PT11	Pt100 #3 sense-
34	NC	Not Connected
35	NC	Not Connected
36	NC	Not Connected
37	NC	Not Connected

Connector LEMO size 2:10
Heaters and Sensors for old Cryostats (La Silla)

PIN	SIGNAL NAME	DESCRIPTION
1	PT1	Pt100 #1 current+
2	NC	Not Connected
3	NC	Not Connected
4	PT3	Pt100 #1 sense-
5	PT2	Pt100 #1 sense+
6	PT8	Pt100 #2 current-
7	PT7	Pt100 #2 sense-
8	PT6	Pt100 #2 sense+
9	POWER HEAT	24VDC
10	HEATER1-	Heater1-

Note: When using PULPO to control temperature on old Cryostats, make sure jumpers JP3 and JP4 inside PULPO Main Board are closed. These old cryostats have the 2 PT100 connected in series!

Connector LEMO size 0:4
LN2 exhaustion alarm

PIN	SIGNAL NAME	DESCRIPTION
1	LN2ALARM1	Pt100 #6 current+
2	LN2ALARM2	Pt100 #6 sense+
3	LN2ALARM3	Pt100 #6 sense-
4	LN2ALARM4	Pt100 #6 current-

Connector LEMO size 0:2
Relay Output

PIN	SIGNAL NAME	DESCRIPTION
1	RELAY1	contact 1 on Normal Close relay
2	RELAY2	contact 2 on Normal Close relay

Connector DB9 male
RS232 link

PIN	SIGNAL NAME	DESCRIPTION
1	NC	Not Connected

2	TX	TX from host RS232
3	RX	RX from host RS232
4	5VDC	PS for opto-isolation modem
5	GND	GND from host
6	NC	Not Connected
7	NC	Not Connected
8	NC	Not Connected
9	NC	Not Connected

Connector DIN 6
Vacuum gauge PKR 250

PIN	SIGNAL NAME	DESCRIPTION
1	VACUUM1	Identification (Not Used)
2	VACUUM2	Signal Output from PKR 250
3	VACUUM3	Signal common from PKR 250
4	VACUUM4	Supply (~24VCD from PULPO)
5	VACUUM5	Supply common
6	VACUUM6	Screen

Connector DB15 female, high density
PULPO shutter module CODE 0 (for IRIS shutters)

PIN	SIGNAL NAME	DESCRIPTION
1	SHUTTER1	opto input, OPEN status
2	SHUTTER2	opto GND
3	SHUTTER3	opto input, CLOSE status
4	SHUTTER4	
5	SHUTTER5	opto input, REMOTE status
6	SHUTTER6	opto GND
7	SHUTTER7	opto input, AUXILIARY status
8	SHUTTER8	opto GND
9	SHUTTER9	opto output collector OPEN
10	SHUTTER10	opto output emiter OPEN
11	SHUTTER11	
12	SHUTTER12	BIAS+ (5VDC common to PULPO)
13	SHUTTER13	BIAS- from DAC controlled current source
14	SHUTTER14	current to COIL+
15	SHUTTER15	current to COIL-

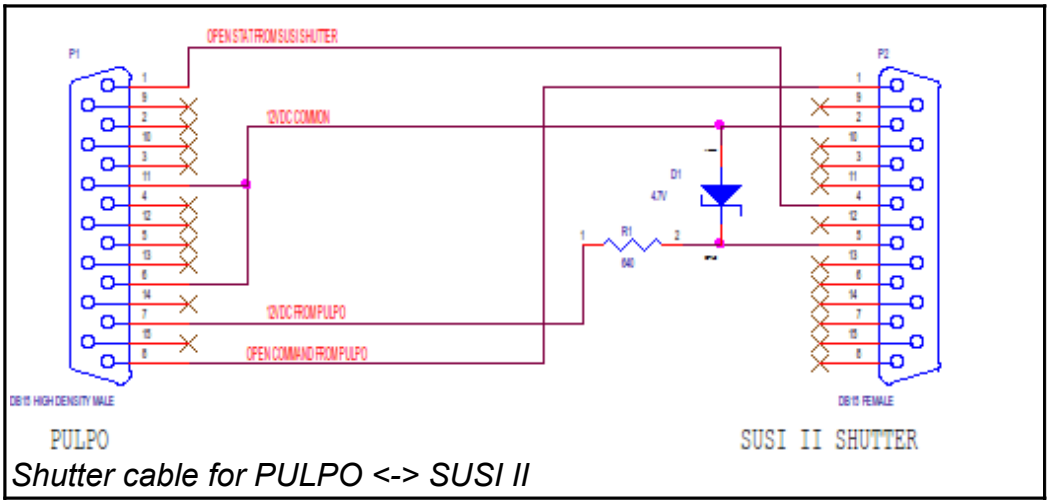
Connector DB15 female, high density
PULPO shutter module CODE 1 (opto-isolated I/O for SUSI II, SESO, FORS, etc.)

PIN	SIGNAL NAME	DESCRIPTION
1	SHUTTER1	opto input, OPEN status
2	SHUTTER2	opto input, CLOSE status
3	SHUTTER3	opto input, REMOTE status
4	SHUTTER4	opto input, AUXILIARY status (FAIL status when driving SESO)
5	SHUTTER5	opto GND
6	SHUTTER6	12VDC GND (shorted to opto GND, pin 5)

7	SHUTTER7	12VDC isolated from PULPO
8	SHUTTER8	opto collector1, OPEN command
9	SHUTTER9	opto collector2, SPARE1
10	SHUTTER10	opto collector3, SPARE2
11	SHUTTER11	opto common emiter for collectors 1,2,3
12	SHUTTER12	BIAS+ (5VDC common to PULPO)
13	SHUTTER13	BIAS- from DAC controlled current source
14	SHUTTER14	opto collector 4, SPARE3
15	SHUTTER15	opto emiter for collector 4

ANNEX F. CABLES

CABLE BETWEEN PULPO AND SESO SHUTTER CONTROLLER	
PULPO SHUTTER CONNECTOR DB15 FEMALE, HIGH DENSITY	SESO CONNECTOR DB15 MALE
1	3
2	2
3	4
4	5
5	9,10,11,12
6	NC
7	NC
8	6
9	NC
10	NC
11	13
12	NC
13	NC
14	NC
15	NC

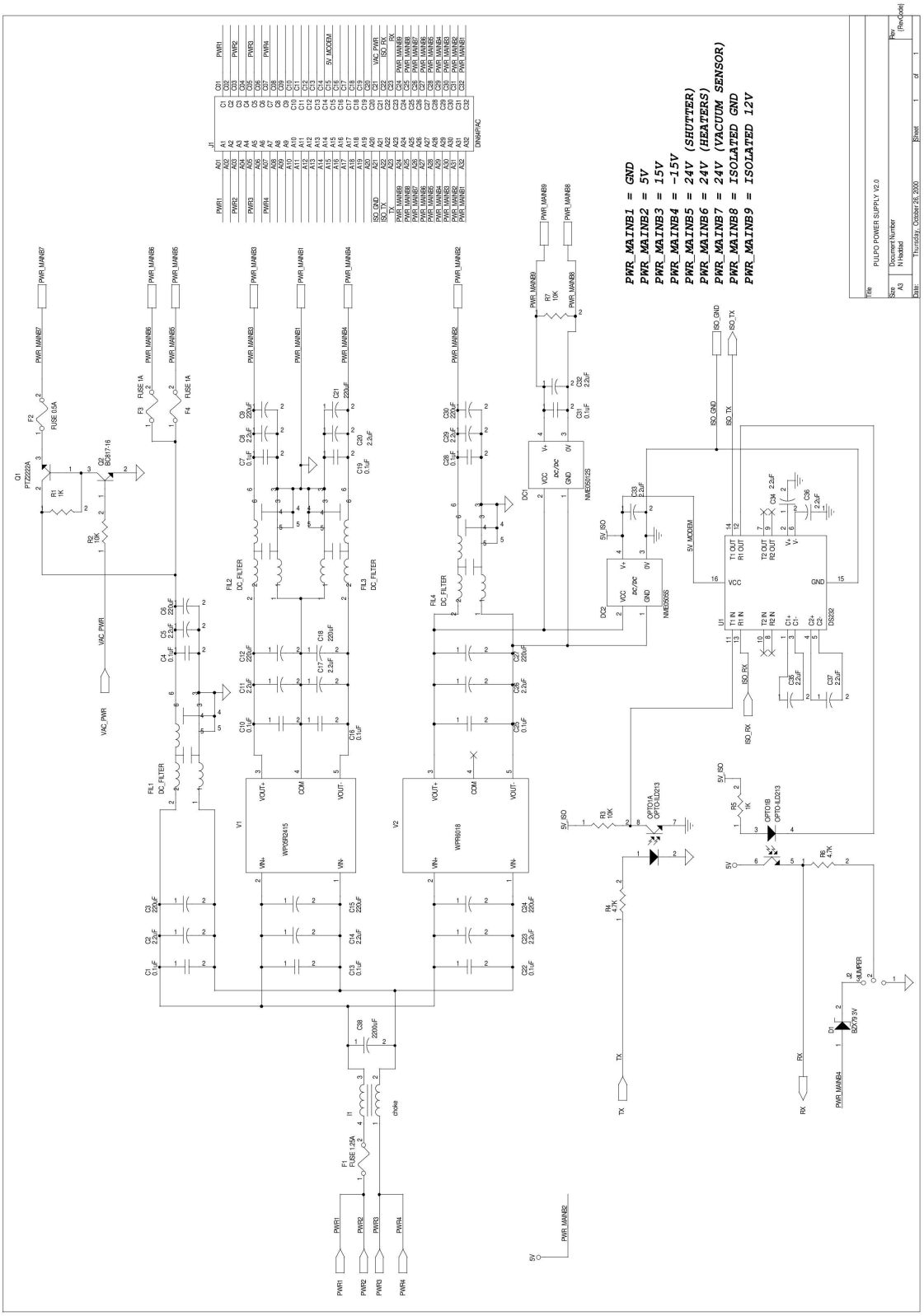


Shutter cable for PULPO <-> SUSI II

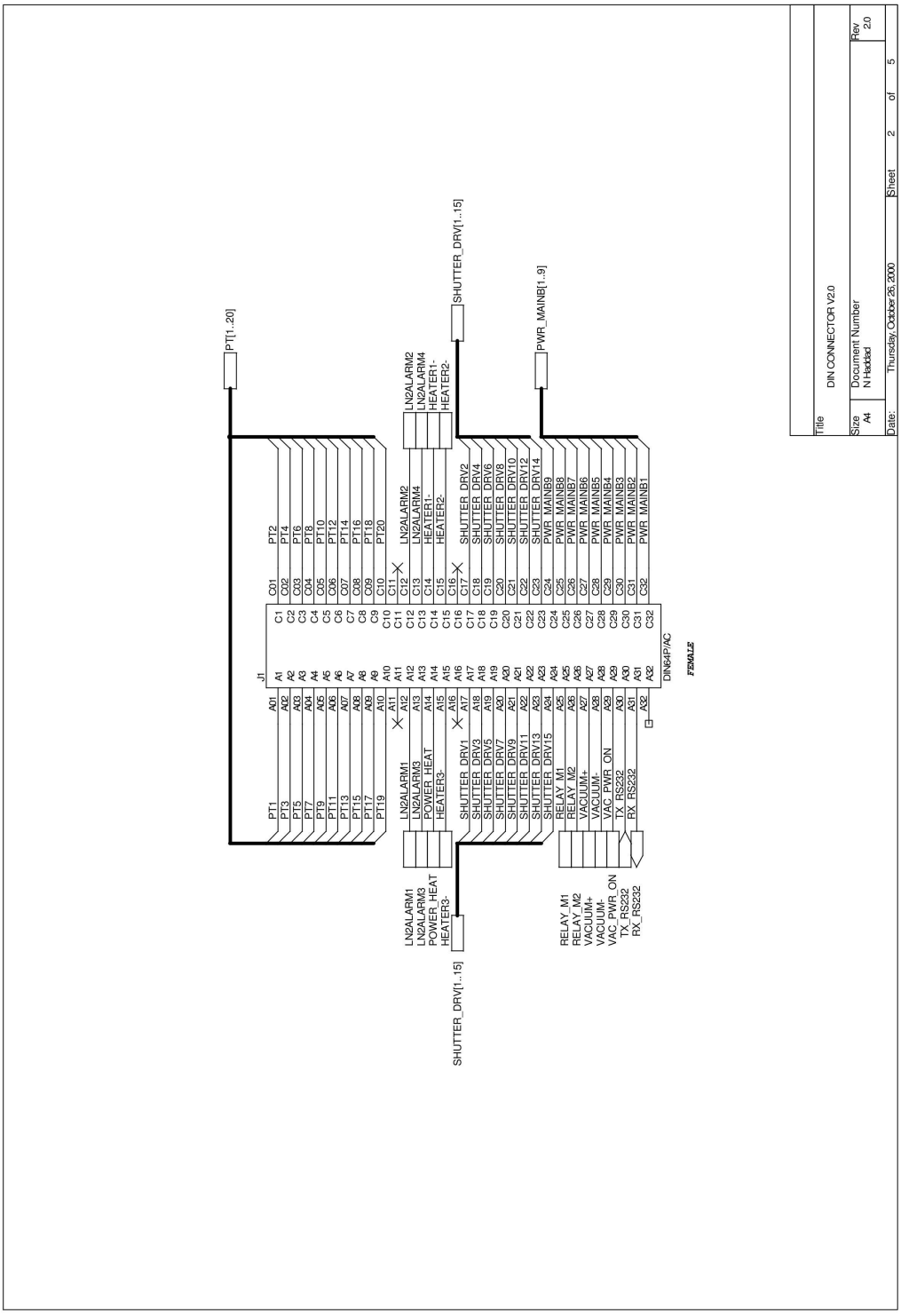
CABLE BETWEEN PULPO AND ESO SHUTTER CONTROLLER	
PULPO SHUTTER CONNECTOR DB15 FEMALE, HIGH DENSITY	ESO SHUTTER CONTROLLER DB15 MALE
1	3
2	2
3	4
4	5
5	9,10,11,12
6	NC
7	NC
8	6
9	NC
10	NC
11	13
12	NC
13	NC
14	NC
15	NC

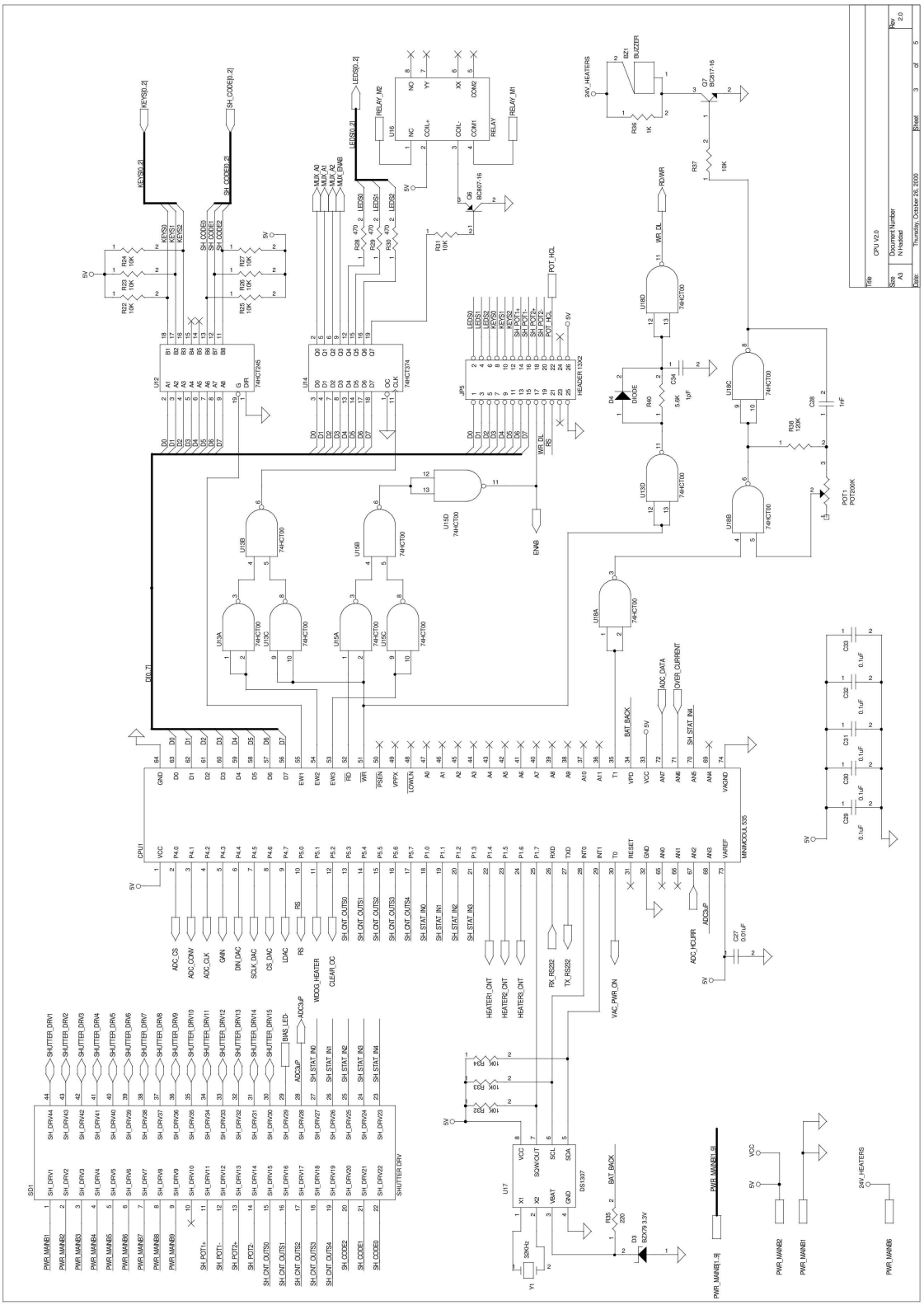
ANNEX G. Schematics 2.0 and 2.2 of PULPO

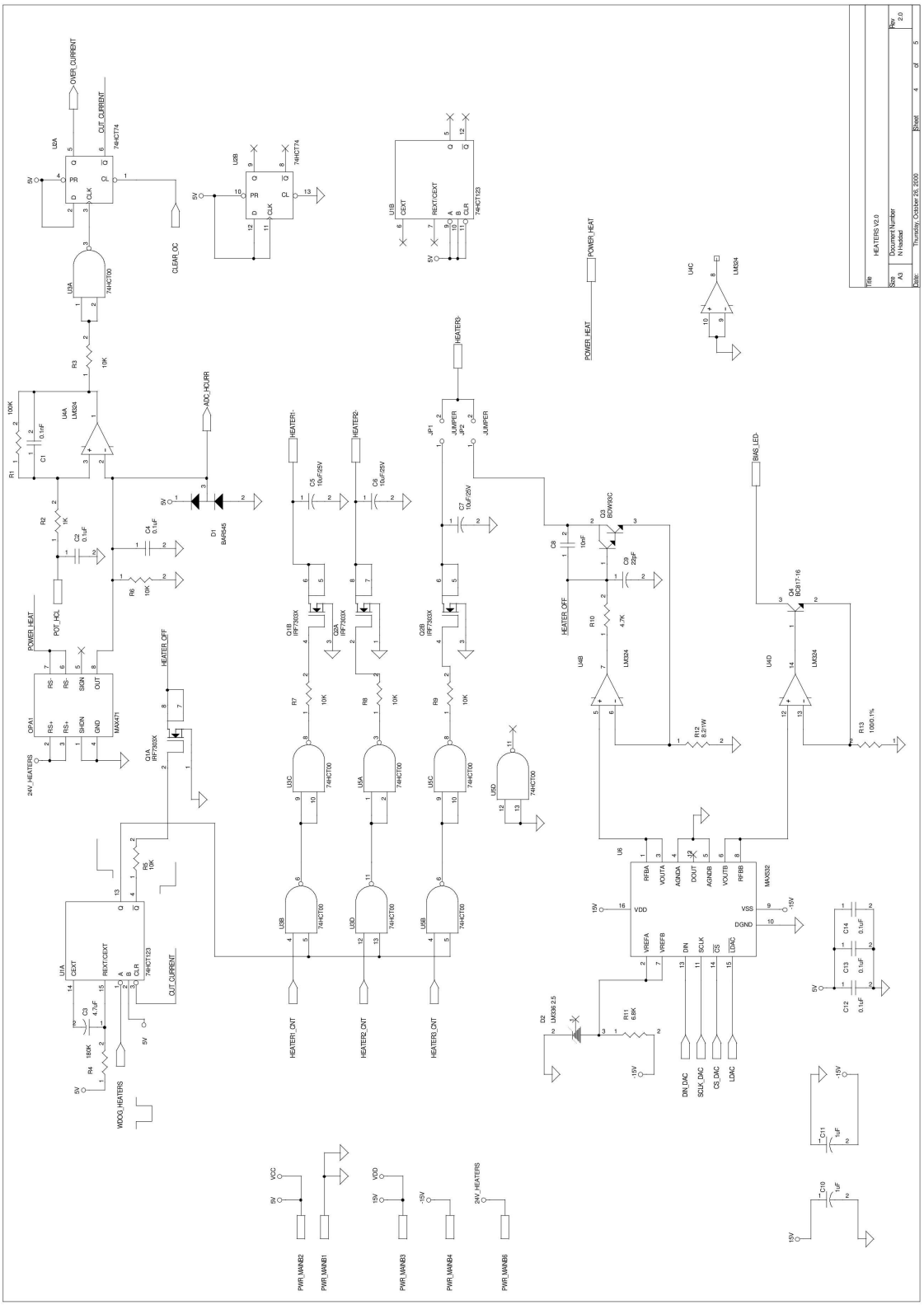
The following pages contains the schematics for hardware version 2.0 and 2.2 of PULPO I.

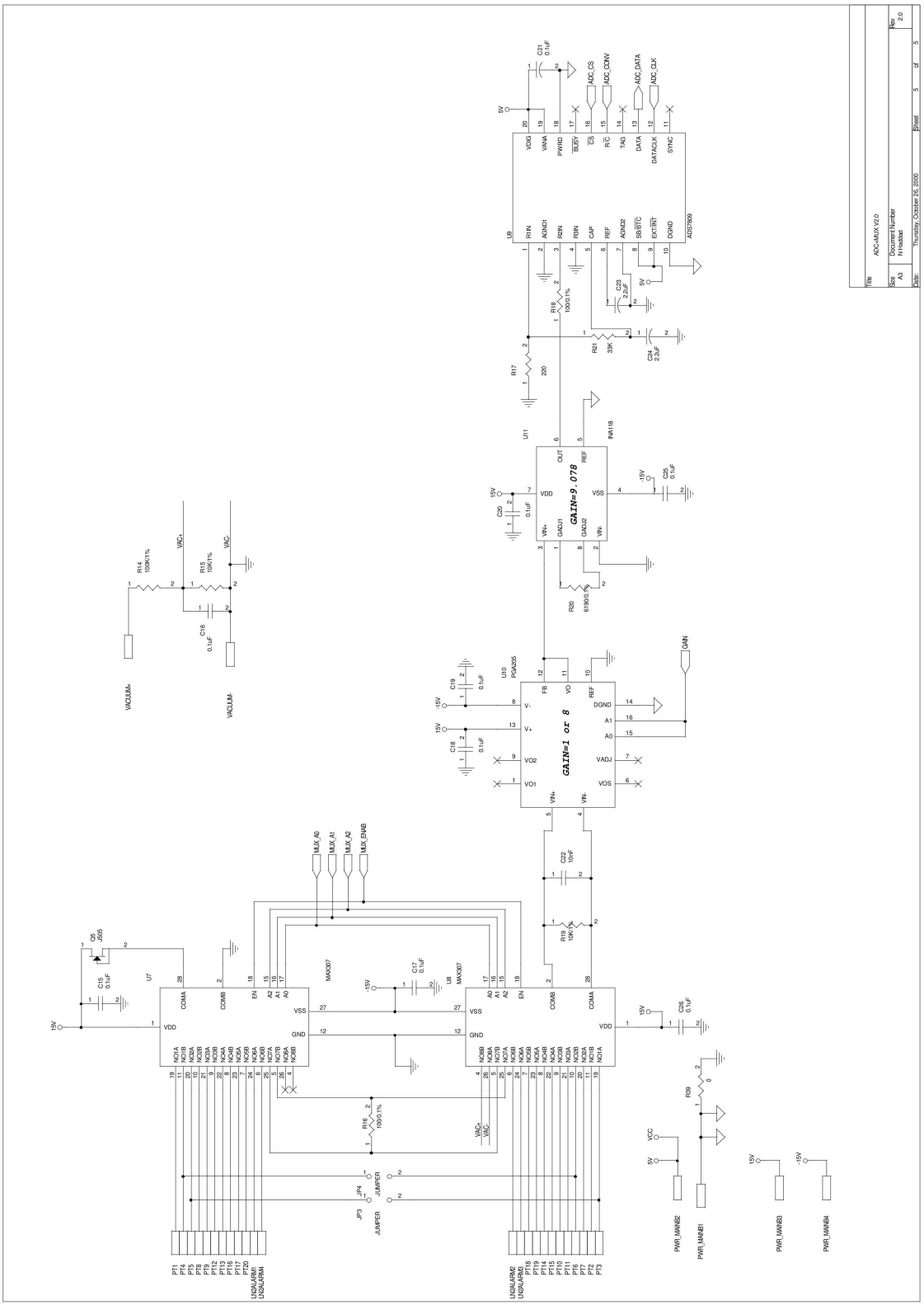

PWR_MAINB1 = GND
PWR_MAINB2 = 5V
PWR_MAINB3 = 15V
PWR_MAINB4 = -15V
PWR_MAINB5 = 24V (SHUTTER)
PWR_MAINB6 = 24V (HEATERS)
PWR_MAINB7 = 24V (VACUUM SENSOR)
PWR_MAINB8 = ISOLATED GND
PWR_MAINB9 = ISOLATED 12V
PULPO POWER SUPPLY V2.0
Thursday, October 28, 2010

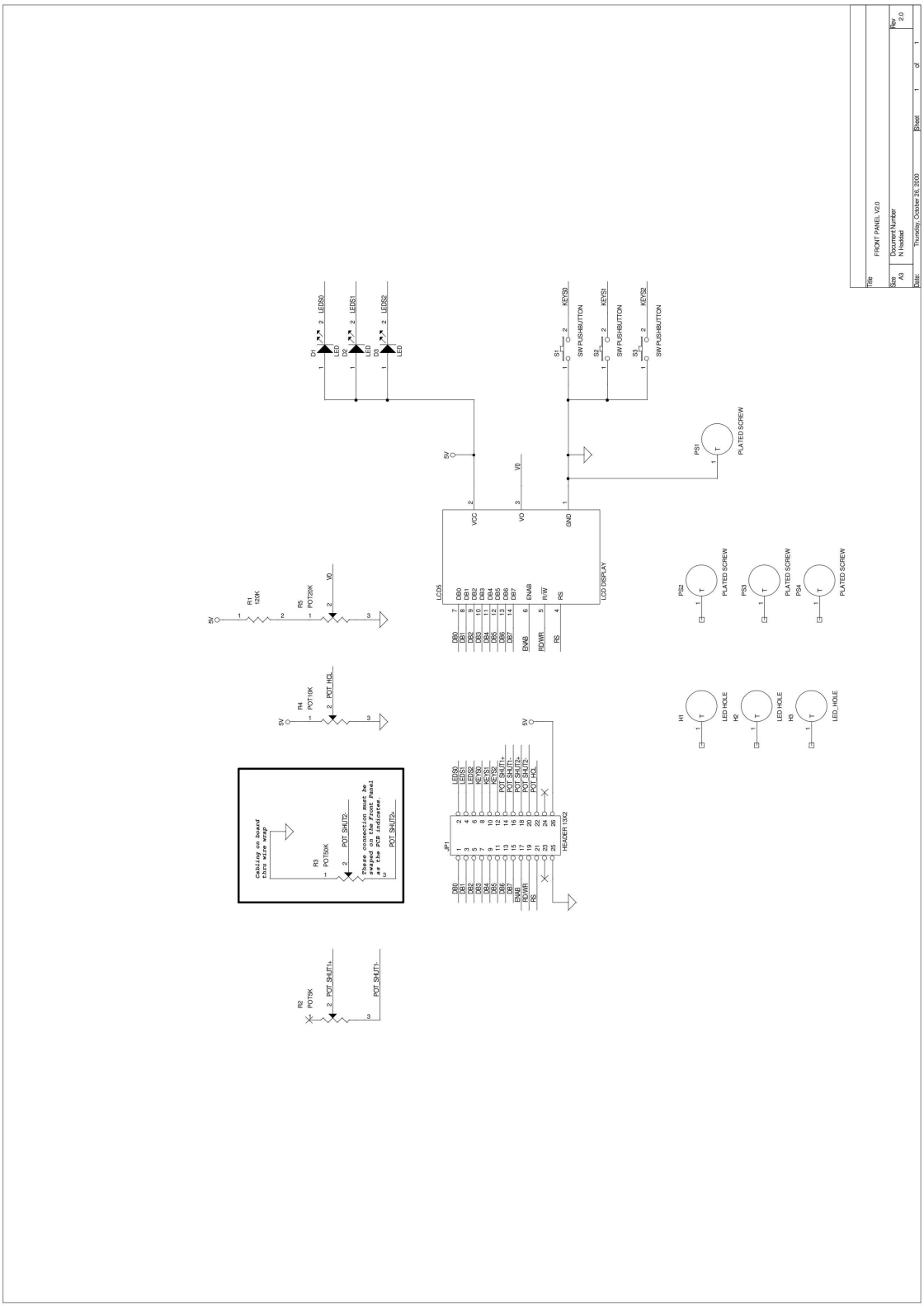

HEATERS
ADC_HCURR
POWER_HEAT
CLEAR_OC
PWR_MAN[1..9]
DIN_DAC
SCLK_DAC
CS_DAC
LDAC
WDOG_HEATERS
OVER_CURRENT
HEATER1-
HEATER2-
HEATER3-
BIAS_LED
HEATER1_CNT
HEATER2_CNT
HEATER3_CNT
POT_HCL
HEATER MODULE
MUX-ADC
PWR_MAN[1..9]
PT[1..20]
LN2ALARM[1..4]
GAIN
MUX_A[0..2]
MUX_ENAB
VACUUM+
VACUUM-
ADC_CS
ADC_CONV
ADC_CLK
ADC_DATA
MUX-ADC MODULE
CPU
RELAY_M[1..2]
RX_RS232
TX_RS232
VAC_PWR_ON
SHUTTER_DRV[1..15]
CPU MODULE
CONNECTOR
(Value)
Title
PULPO MAN BOARD
Size
A3
Document Number
[Doc]
Rev
[RevDate]
Date:
Thursday, October 28, 2010
Sheet
1
of
5

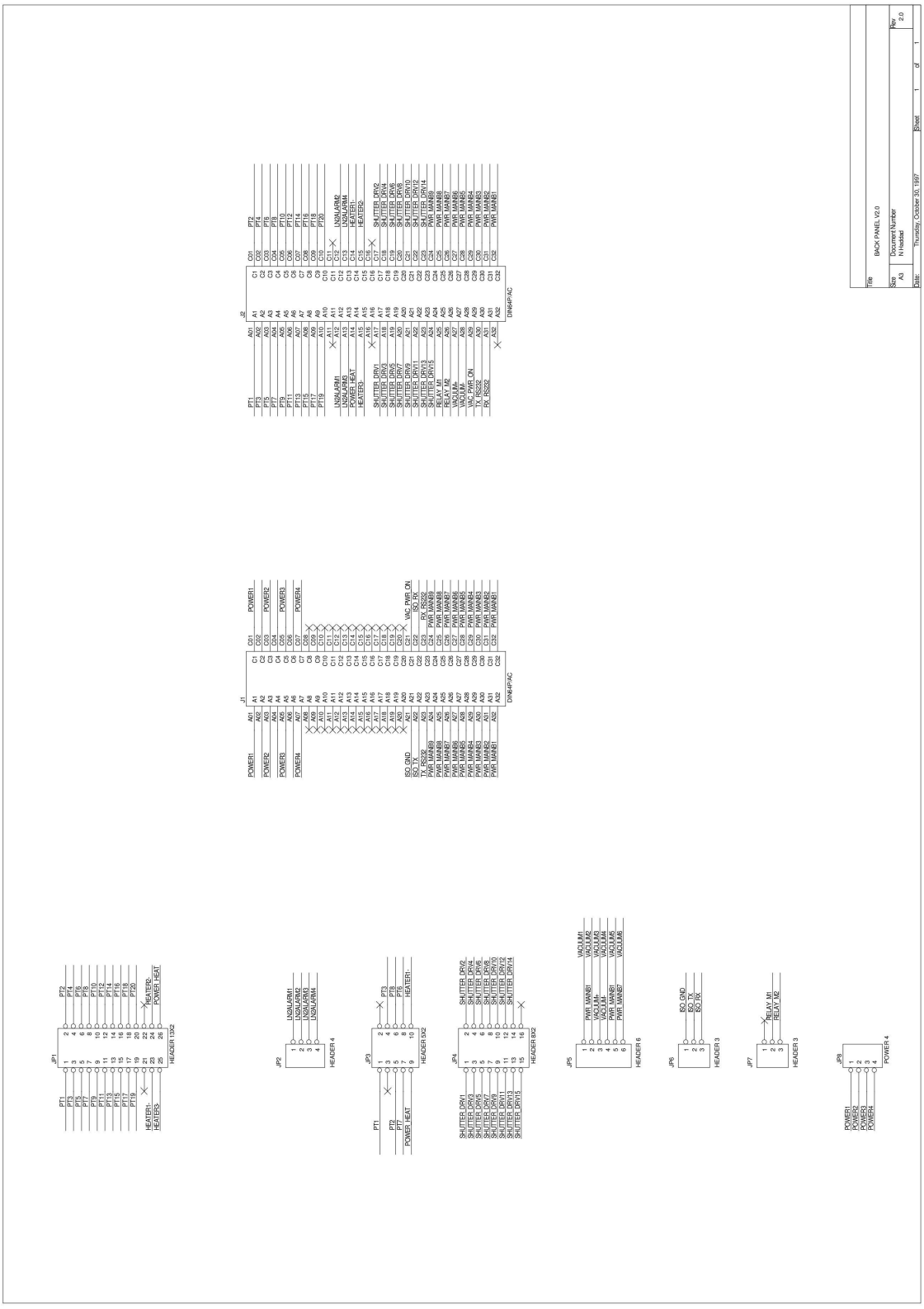

PT[1..20]
SHUTTER_DRV[1..15]
PWR_MAINB[1..9]
J1
DIN64P/AC
FEMALE
PT1
PT3
PT5
PT7
PT9
PT11
PT13
PT15
PT17
PT19
PT2
PT4
PT6
PT8
PT10
PT12
PT14
PT16
PT18
PT20
LN2ALARM1
LN2ALARM3
POWER_HEAT
HEATER3-
LN2ALARM2
LN2ALARM4
HEATER1-
HEATER2-
SHUTTER_DRV1
SHUTTER_DRV3
SHUTTER_DRV5
SHUTTER_DRV7
SHUTTER_DRV9
SHUTTER_DRV11
SHUTTER_DRV13
SHUTTER_DRV15
SHUTTER_DRV2
SHUTTER_DRV4
SHUTTER_DRV6
SHUTTER_DRV8
SHUTTER_DRV10
SHUTTER_DRV12
SHUTTER_DRV14
PWR_MAINB9
PWR_MAINB8
PWR_MAINB7
PWR_MAINB6
PWR_MAINB5
PWR_MAINB4
PWR_MAINB3
PWR_MAINB2
PWR_MAINB1
RELAY_M1
RELAY_M2
VACUUM+
VACUUM-
VAC_PWR_ON
TX_RS232
RX_RS232
Title
DIN CONNECTOR V2.0
Size
A4
Document Number
N Hackstad
Rev
2.0
Date:
Thursday, October 26, 2000
Sheet
2
of
5

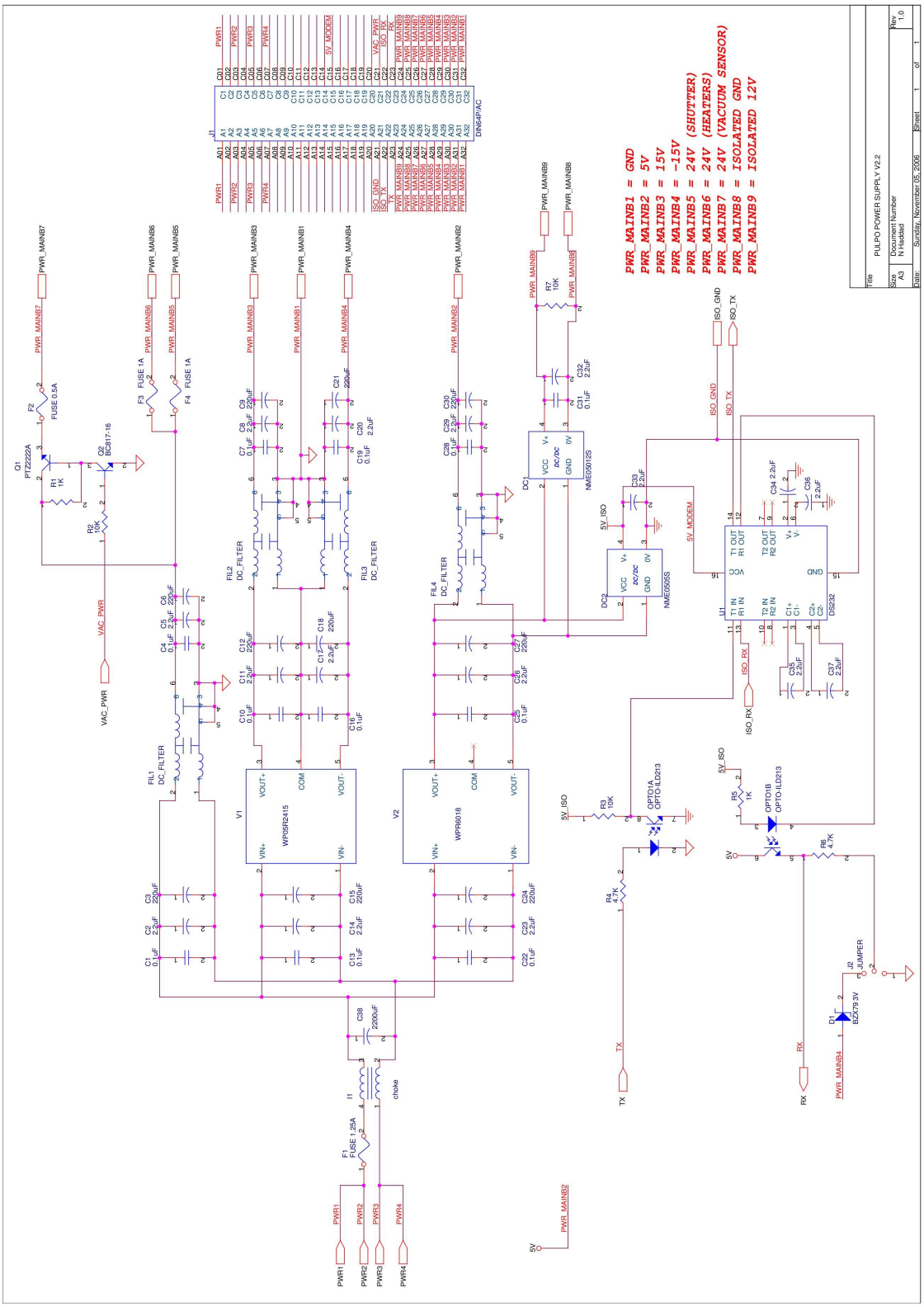


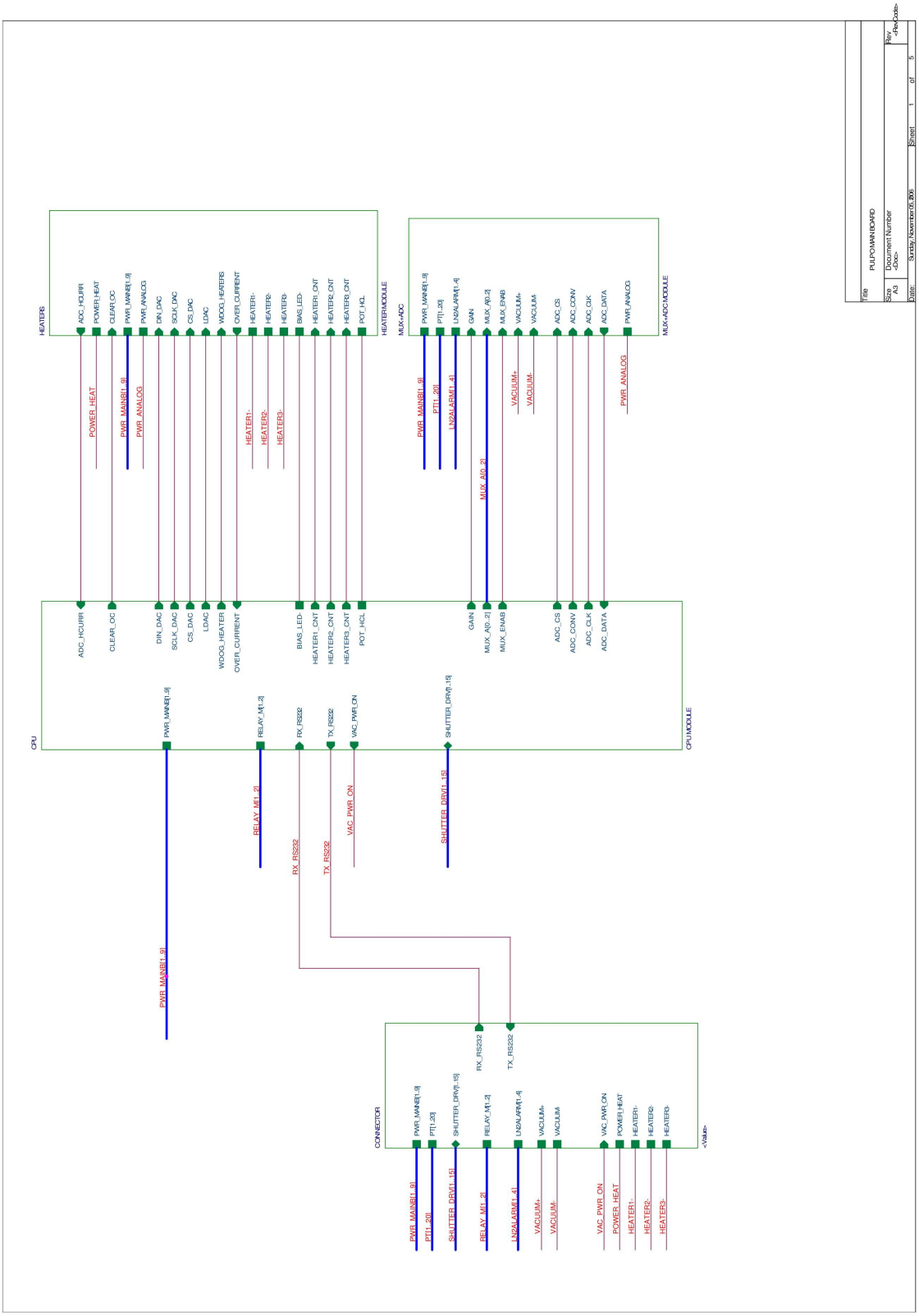

HEATERS V2.0
Thursday, October 26, 2000
OVER_CURRENT
OUT_CURRENT
CLEAR_OC
ADC_HOURR
POWER_HEAT
HEATER_OFF
HEATER1
HEATER2
HEATER3
BIAS_LED
HEATER1_CNT
HEATER2_CNT
HEATER3_CNT
WDOG_HEATERS
24V_HEATERS
DIN_DAC
SCLK_DAC
CS_DAC
LDAC
PWR_MAN82
PWR_MAN81
PWR_MAN83
PWR_MAN84
PWR_MAN86

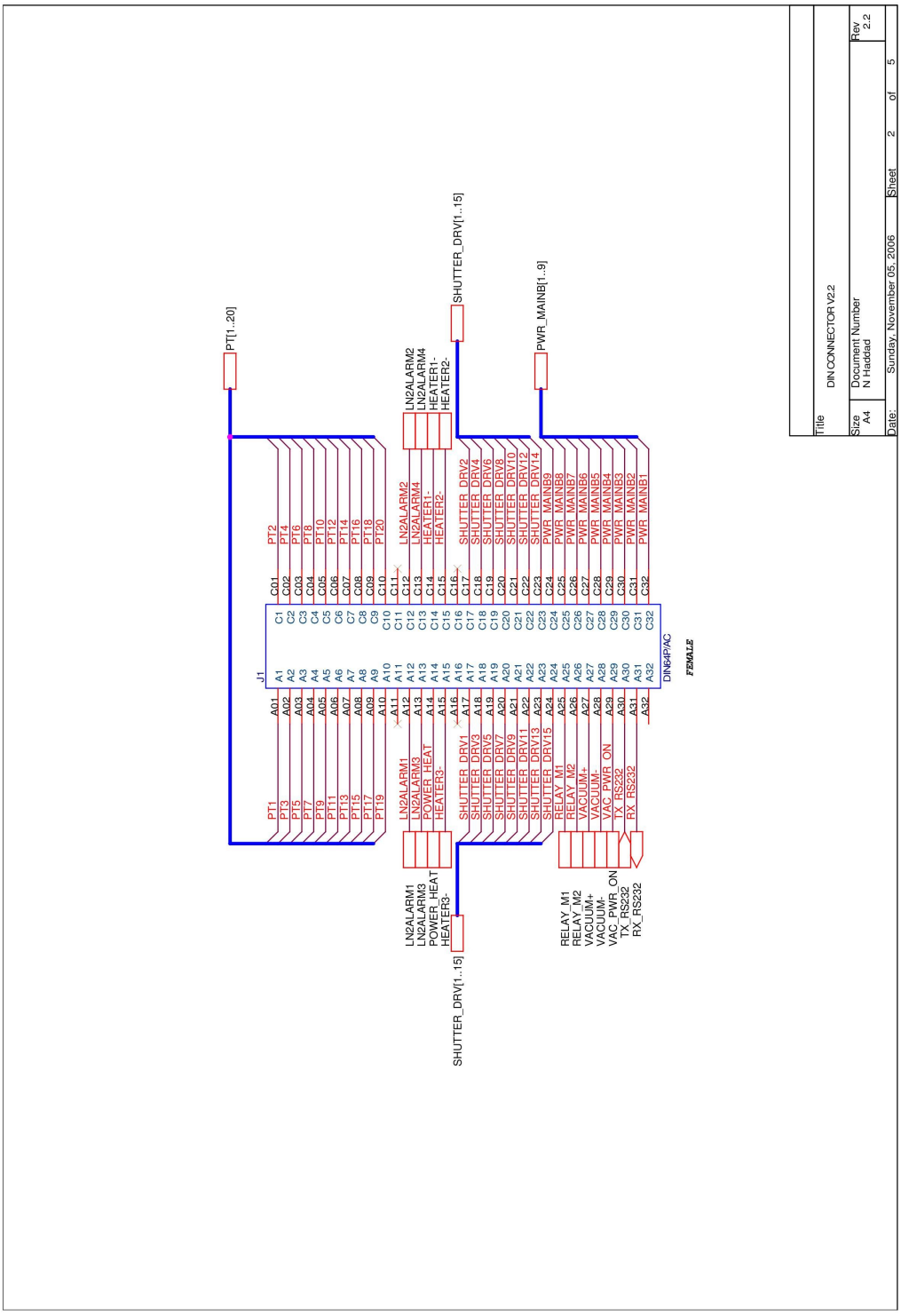

ADC-MUX V2.0
GAIN=9.078
GAIN=1 or 8
INA118
PGA505
MAX307
ADS7809
JUMPER
PWR_MAN82
PWR_MAN81
PWR_MAN83
PWR_MAN84

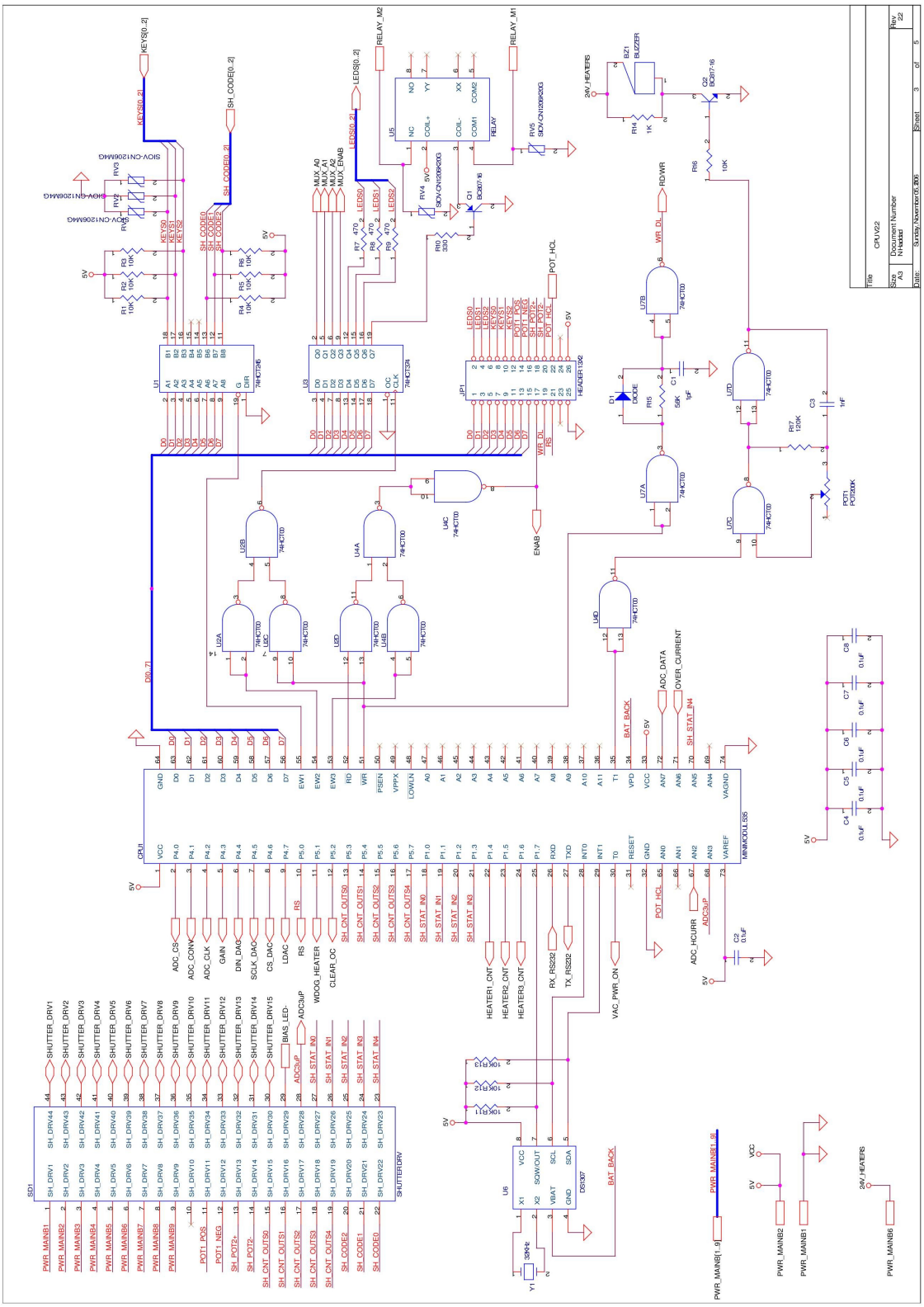

FRONT PANEL V2.0
Thursday, October 26, 2000
LED0
LED1
LED2
KEY0
KEY1
KEY2
SW PUSHBUTTON
LCD5
LCD DISPLAY
HEADER 13X2
PLATED SCREW
LED HOLE
LED_HOLE
POT_SHUT1-
POT_SHUT2-
POT_SHUT1+
POT_SHUT2+
POT_HOL
POT_FOC
Cabling on board this wire
These connection must be wired on the front Panel as the pcb indicates

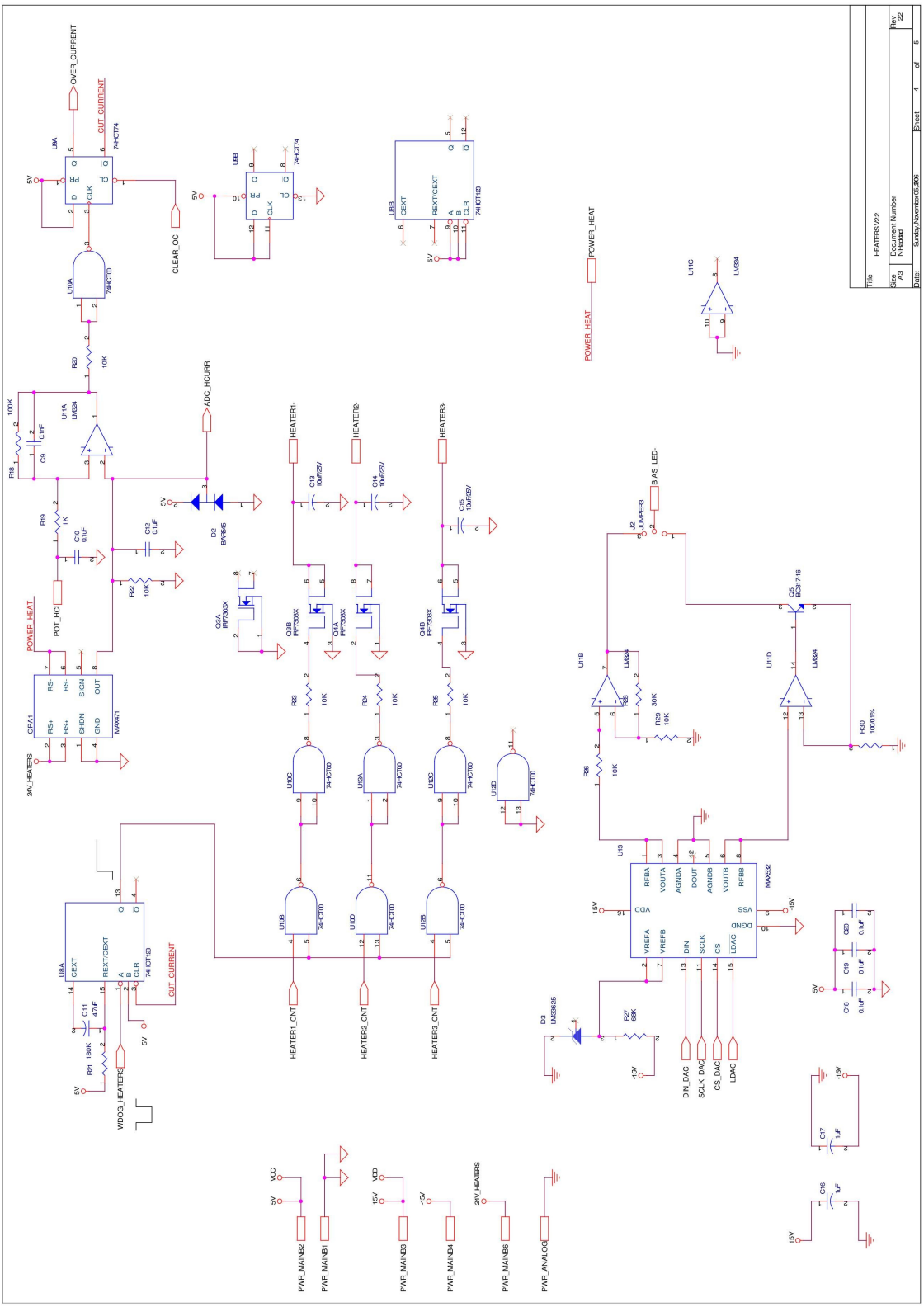


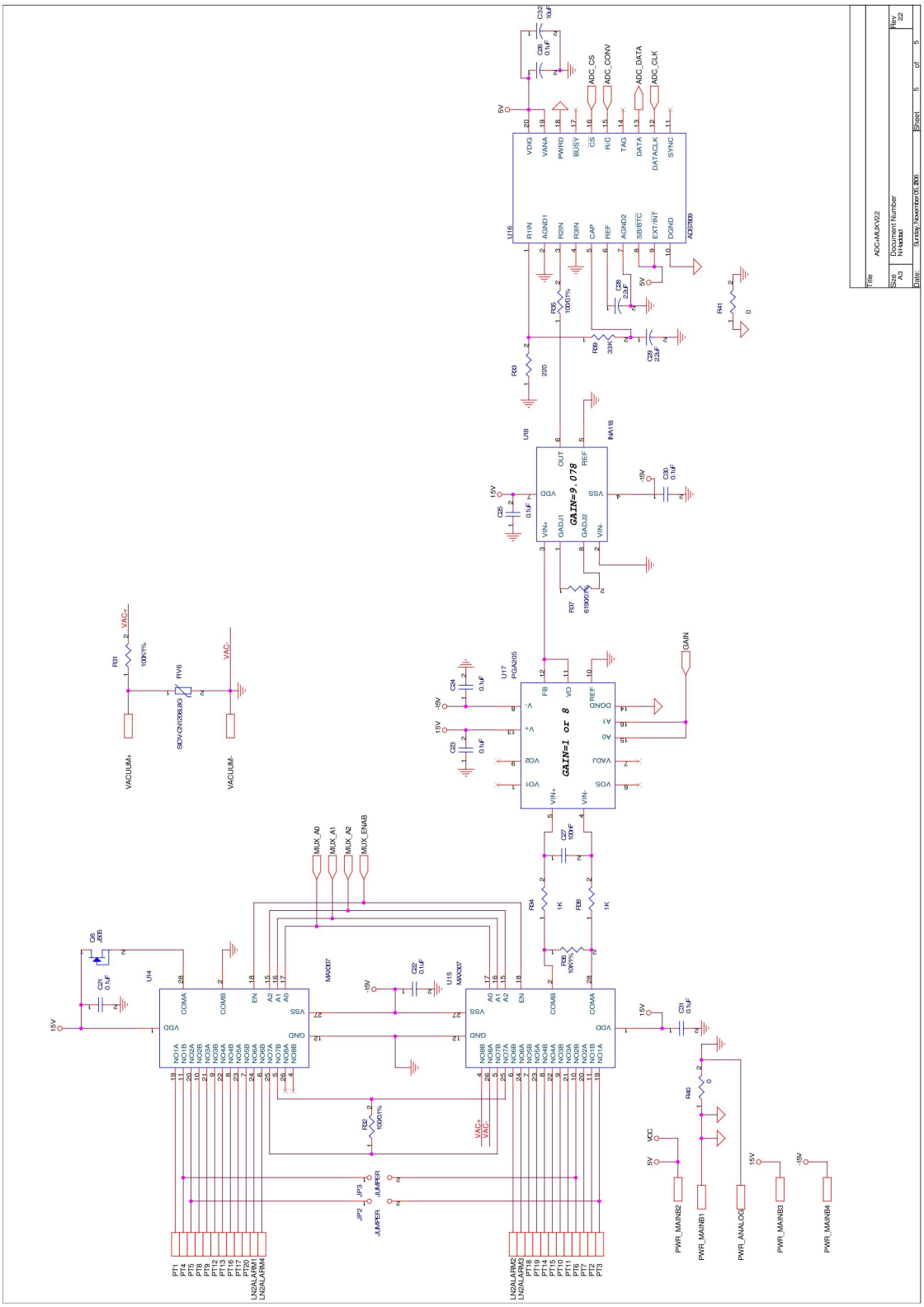

PWR_MAINB1 = GND
PWR_MAINB2 = 5V
PWR_MAINB3 = 15V
PWR_MAINB4 = -15V
PWR_MAINB5 = 24V (SHUTTER)
PWR_MAINB6 = 24V (HEATERS)
PWR_MAINB7 = 24V (VACUUM SENSOR)
PWR_MAINB8 = ISOLATED GND
PWR_MAINB9 = ISOLATED 12V
PULPO POWER SUPPLY V2.2
VAC_PWR
ISO_GND
ISO_TX
ISO_RX
5V_ISO
5V_MODEM
TX
RX
choke
JUMPER

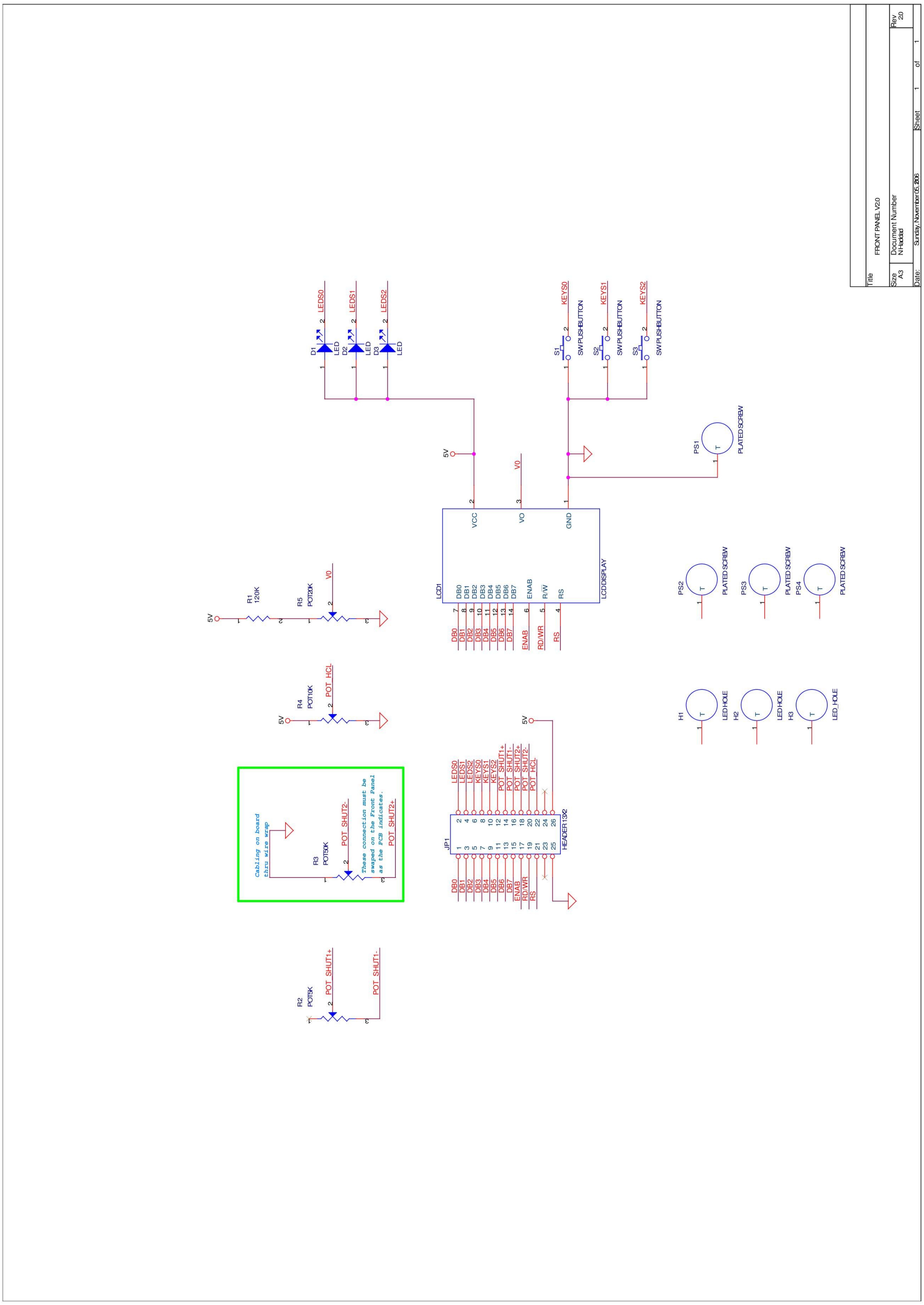

HEATERS
ADC_HOURR
POWER_HEAT
CLEAR_OC
PWR_MAINB[1..9]
PWR_ANALOG
DIN_DAC
SCLK_DAC
CS_DAC
LDAC
WDOG_HEATERS
OVER_CURRENT
HEATER1-
HEATER2-
HEATER3-
BIAS_LED
HEATER1_CNT
HEATER2_CNT
HEATER3_CNT
POT_HCL
HEATERMODULE
MUX-ADC
PWR_MAINB[1..9]
PTT[1..2X]
LN2ALARM[1..4]
GAIN
MUX_A[0..2]
MUX_ENAB
VACUUM+
VACUUM-
ADC_CS
ADC_CONV
ADC_CLK
ADC_DATA
PWR_ANALOG
MUX-ADCMODULE
CPU
PWR_MAINB[1..9]
RELAY_M1..2]
RX_RS232
TX_RS232
VAC_PWR_ON
SHUTTER_DRV[0..1]
CPU MODULE
CONNECTOR
SHUTTER_DRV[0..1]
RELAY_M1..2]
LN2ALARM[1..4]
VACUUM+
VACUUM-
POWER_HEAT
HEATER1-
HEATER2-
HEATER3-
Title
PULPOMAINBOARD
Size
A1
Document Number
ADC2
Rev
<RevCode>
Date:
Sunday, November 05, 2006
Sheet
1
of
5


PT[1..20]
SHUTTER_DRV[1..15]
PWR_MAINB[1..9]
LN2ALARM2
LN2ALARM4
HEATER1-
HEATER2-
J1
DIN64P/AC
FEMALE
LN2ALARM1
LN2ALARM3
POWER_HEAT
HEATER3-
RELAY_M1
RELAY_M2
VACUUM+
VACUUM-
VAC_PWR_ON
TX_RS232
RX_RS232
Title
DINCONNECTOR V2.2
Size A4
Document Number
N Haddad
Rev 2.2
Date: Sunday, November 05, 2006
Sheet 2 of 5

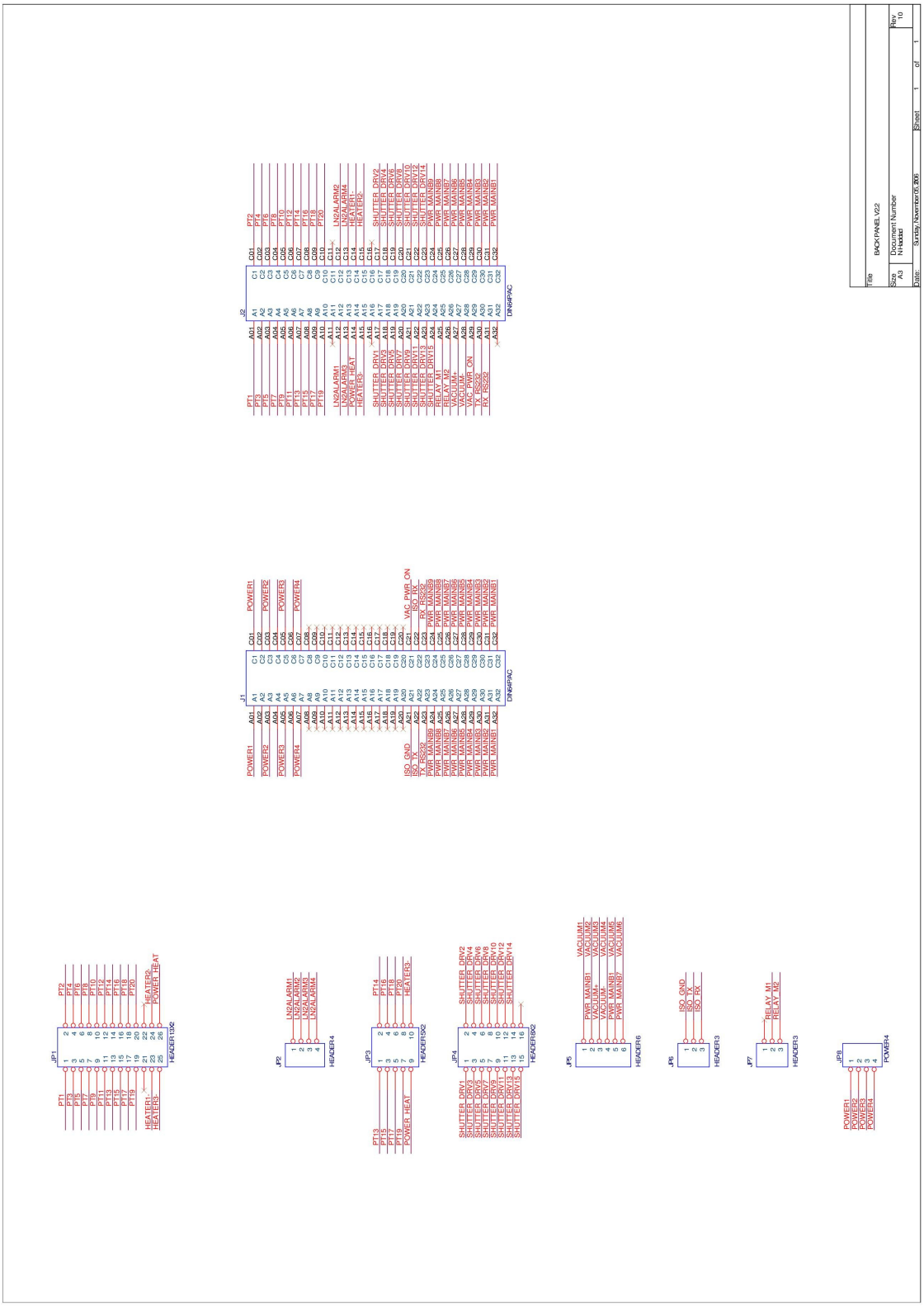


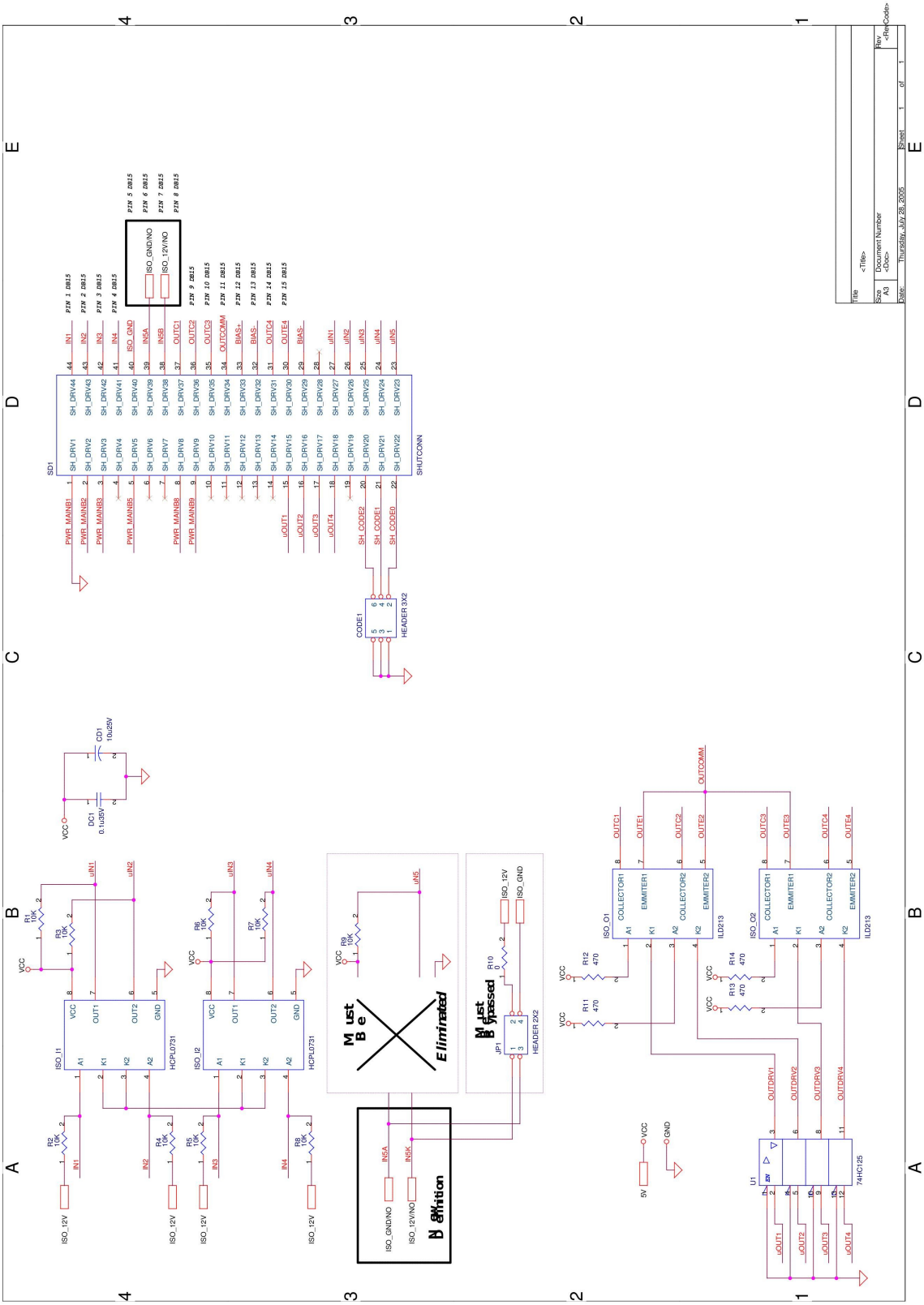

OVER_CURRENT
CUT_CURRENT
CLEAR_OC
ADC_HCLRR
POWER_HEAT
POT_HC
BIAS_LED
HEATER1-
HEATER2-
HEATER3-
HEATER1_CNT
HEATER2_CNT
HEATER3_CNT
WDOG_HEATERS
DIN_DAC
SCLK_DAC
CS_DAC
LDAC
PWR_MAIN82
PWR_MAIN81
PWR_MAIN83
PWR_MAIN84
PWR_MAIN86
PWR_ANALOG


ADC-MUX/V22
ADC_CS
ADC_CONV
ADC_DATA
ADC_CLK
GAIN=9.078
GAIN=1 or 8
PGA205
INA118
MAX307
MUX_A0
MUX_A1
MUX_A2
MUX_ENAB
GAIN
VAC+
VAC-
VACUUM+
VACUUM-
PWR_MAINB2
PWR_MAINB1
PWR_ANALOG
PWR_MAINB3
PWR_MAINB4



FRONT PANEL V2.0
LCDDISPLAY
PLATED SCREW
SW PUSHBUTTON




SD1
SHUTCONN
CODE1
HEADER 3X2
ISO_GND/NO
ISO_12V/NO
ISO_I1
ISO_I2
HCPL0731
ISO_O1
ISO_O2
ILD213
U1
74HC125
JP1
HEADER 2X2
Must Be Eliminated
Must Be Bypassed
VCC
GND
5V
ISO_12V
ISO_GND