# USING VOODOO TO TEST SDSU CONTROLLERS

Rosa Clavero Jiménez

*Issac Newton Group of Telescopes*

*August, 2010*

# INDEX

# 1.- CONTROLLERS FOR SCIENCE CAMERAS

## 1.1.- ASSEMBLING THE SYSTEM

- After a controller has been repaired and before it is used in the telescope, it has to be check in the laboratory.

- The first step is reproduced the system in the laboratory. You need to assembling it:

  - The **Controller**
  - A **UltraDas machine**
  - A **Double Power Supply**, one for the controller and the other for the shutter.

- You can see these components in the following photos:


- To join these components it is necessary the following wires:

  - **Doble mains cable**
  - **PowerSupply cable**
  - **Doble Optical Fiber:** It is necessary a doble fiber, one to send information to the Ultradas machine and the other to receive from it in the controller.



-As we don't know if the controller has been well repared, it is not advisable use a CCD camera because it may be damaged. So, the first check must be done with the following Test Boxes connected to the frontside of the controller with their corresponding wires:


– **SDSU ID Code Test Box:**

- Each camera has its own ID code with two character in hexadecimal representation. You can check it in ***appendix 1*** of this document or in the following web page:

http://www.ing.iac.es/~eng/detectors/SDSU/detector_id.html

- Each hex character can be writen with four bits in binary representation. In the Test Box you can move the position of the switches to reproduce the ID code of the camera you are going to simulate. The most and less significant bit (MSB and LSB) of each group are indicated.


– **Temp Test Box**

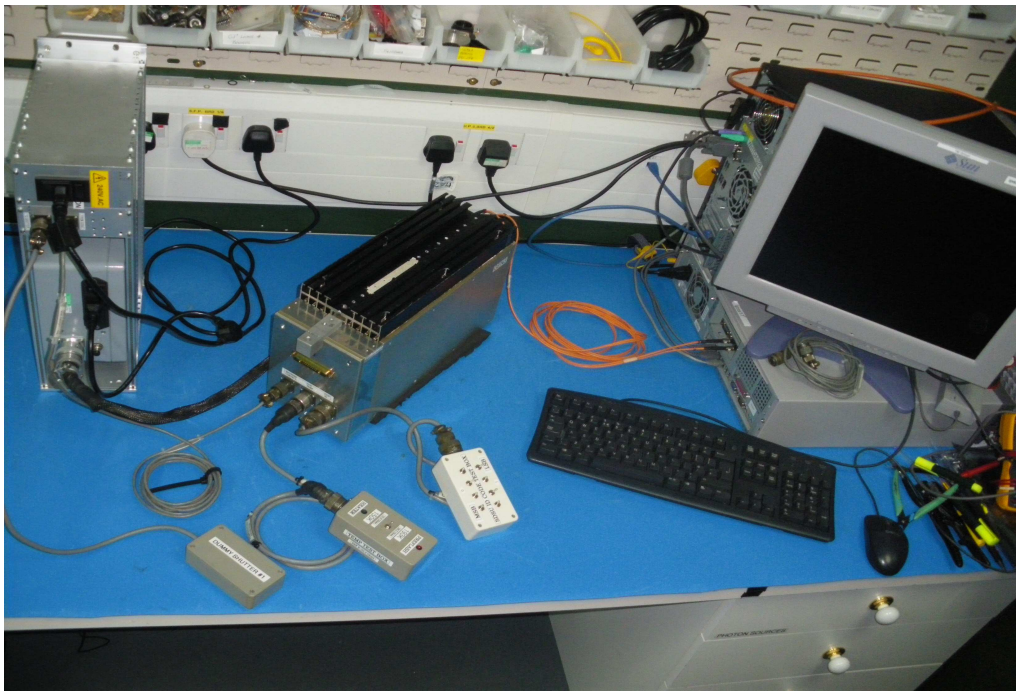
- It has a switch which reproduces two diferent temperatures. There is a red led which turn on when the preflash is working. As well a green led switches on when the heater is turned on.
So, you can check the good operation of temperature, preflash and heater.


– **Dummy Shutter Box**

- This test box replaces the shutter and it is plugged in to the shutter card which is next to the power supply. Another cable joins the shutter card with the controller. There is a red led in the box which switch on when the shutter is open.

- Also, you can change the shutter status with a button and check it with two leds which are on the shutter card.

- You can see the whole system in the photo below:



- Turn on the power supplies. Check the two leds in the UltraDas backside. They must be turned off. In other case, exchange the position of the fibers.


- Now, the system is ready to test the controller with Voodoo.

## 1.2.- STARTING THE SYSTEM

### 1.2.1.- STARTING VOODOO

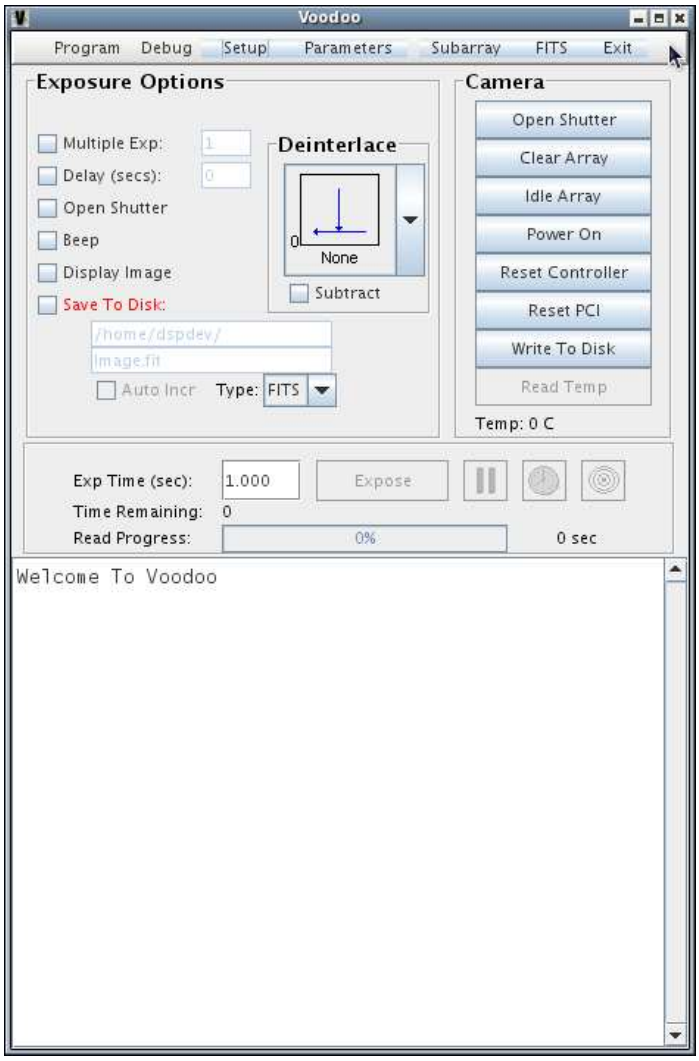
- After assembling the system, you must turn on the power supply of the controller and the shutter. The DAS machine is normally turned on in the laboratory, so you only have to start with the account:

Login: **DSPDEV**
Password: *******

- Voodoo can be started up writing in a terminal:

> **voodoo &**

- The ***main Voodoo*** **window** is shown below.

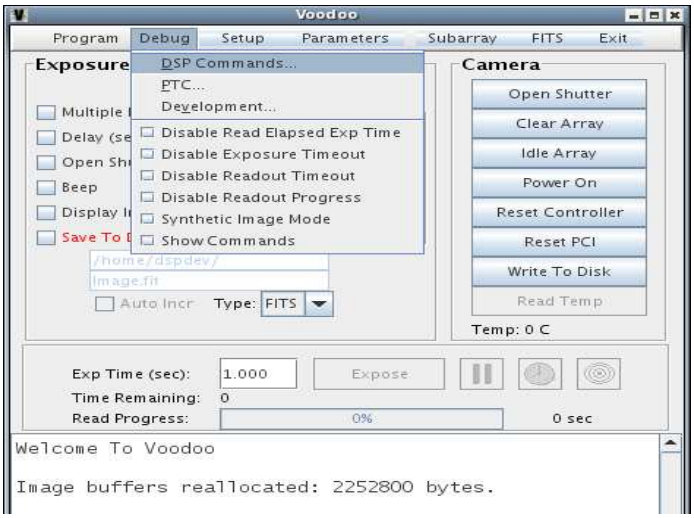


## 1.2.2.- CAMERA ID CODE:

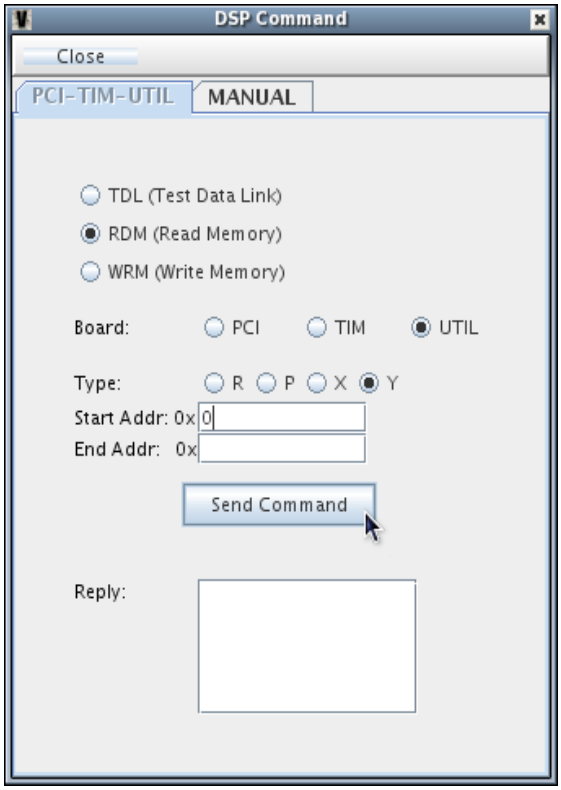
- **NOTE:** It is very important to make sure which camera we are working with and load correctly the configuration file. Otherwise, the controller boards and the camera can be damaged!!

- Once the ***main Voodoo* window** appears, select in the drop down menu:

***Debug→DSP Command***




- The ***DSP Command* window** will then appear:




- Mark the following options in the ***PCI-TIM-UTIL* tab**:

✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔Start Addr: **0**

- After that, select the ***Send Command* button**.

- We have an answer with a code. We are interested in the last two numbers which are the camera code in hexadecimal representation. You can check it in ***appendix 1*** of this document or in the following web page:

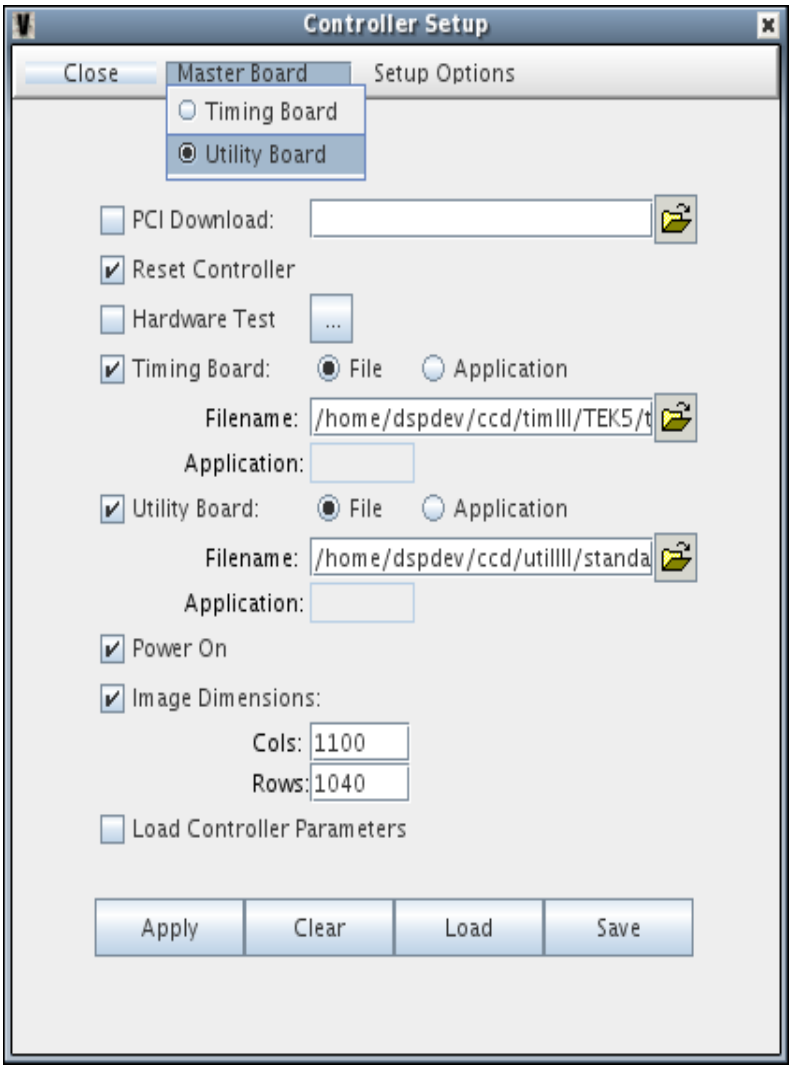
http://www.ing.iac.es/~eng/detectors/SDSU/detector_id.html

### 1.2.3.- LOAD THE CONFIGURATION FILE

- Select in the drop down menu of the ***main Voodoo*** **window**:

***Setup***

- The ***Controller Setup*** **window** starts up.



- Mark the following options:

You have to resize the window manually. Select in the drop down menu:

***Master Board→Utility Board***

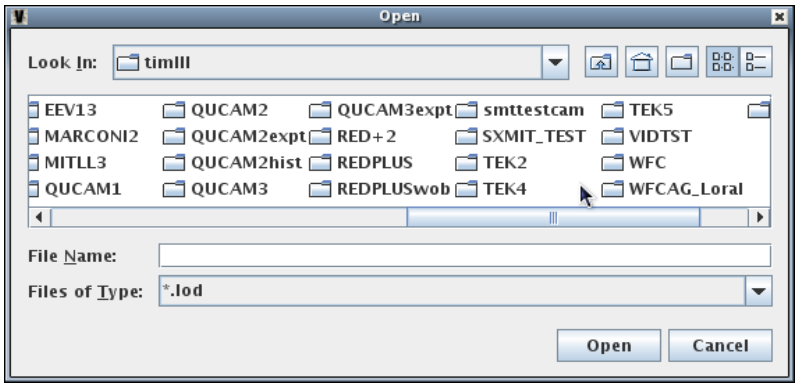
✔ **Reset Controller**

✔ **Timing board→File**

Press the icon in the right of the box to search the file. A window starts up. You must go to the following directory:

***/home/dspdev/ccd/timIII/***

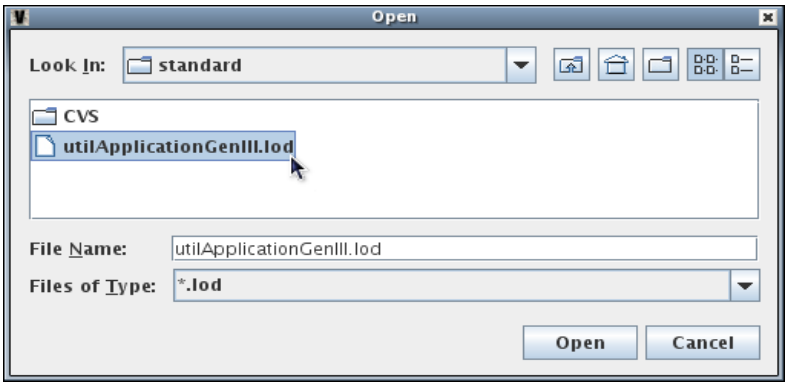
Here, there is a list of folders for each camera. Enter in the corresponding folder to the camera you are using and select the most recent file.




✔ **Utility board→File**

Press the icon in the right of the box to search the file. In this case, the file is the same for all cameras. So, search in the emerging window the following file:

***/home/dspdev/ccd/utilIII/standard/utilApplicationGenIII.lod***




✔ **Power on**

✔ **Image dimensions**

It refers to the digitised area of the camera. You can find this information in **appendix 1** or in the following web page:

http://www.ing.iac.es/Engineering/engweb6a.htm

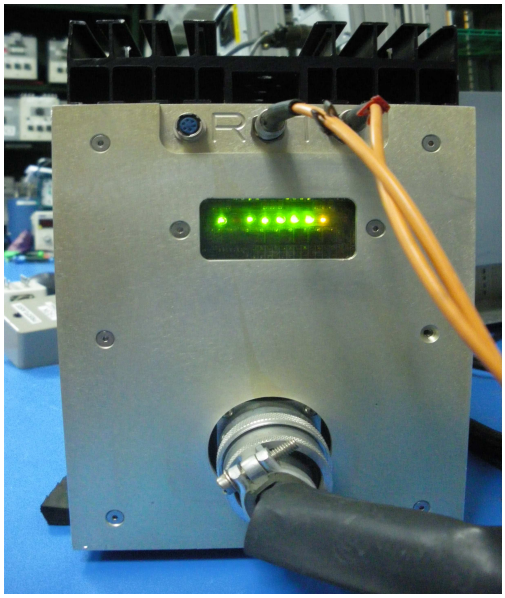
Fill the box for the columns corresponding to the number of pixel in the x direction and the rows that represent the y direction.

- You can save the configuration pressing the ***Apply*** **button**. This action will then download the code and power on the controller. The main Voodoo window will show the progress of the download and power on. Notice that there is a small error message:

"Checking Controller Configuration.....failed"

- Don't worry about this, it doesn't matter.

- Finally, to be sure that the controller is powered on, verify the leds on the power control board are switched on, with the exception of the second one.

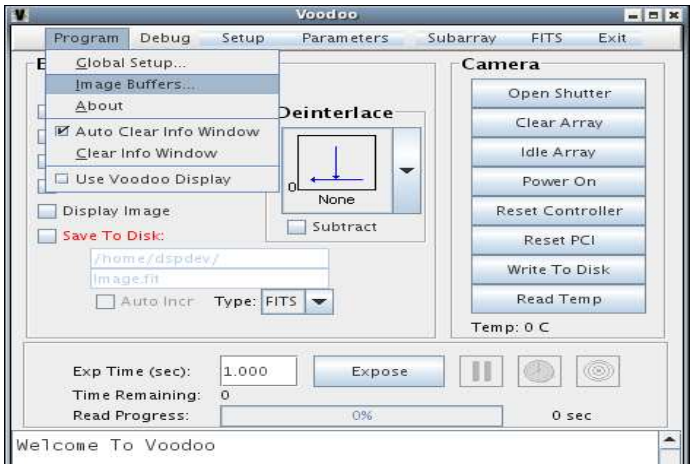


## 1.2.4.- IMAGE DIMENSION

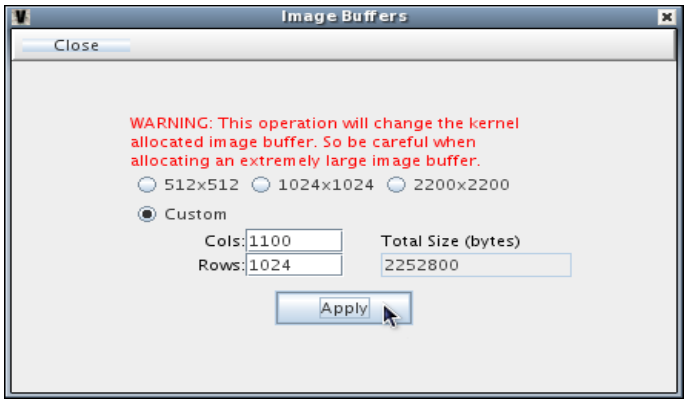
- Although you have defined the image dimensions in the configuration file for the controller, it is necessary to do again for the buffer in the Das machine.

- Select in the drop down menu of the ***main Voodoo*** **window**:

***Program→Image Buffers***




- The emerging window doesn't start up with the correct size and not all the buttons are visible, so you have to resize the window manually.




- Mark

***✔ Custom***

- And then fill the box for columns and rows which represent pixels in the x and y direction respectively. If you are not sure of the image dimensions for the camera you are using, then refer to ***appendix 1*** of this document.

- Press the ***Apply*** **button** and close the window.

- Sometimes Voodoo crashes when you do that. Don't worry but you have to start again.

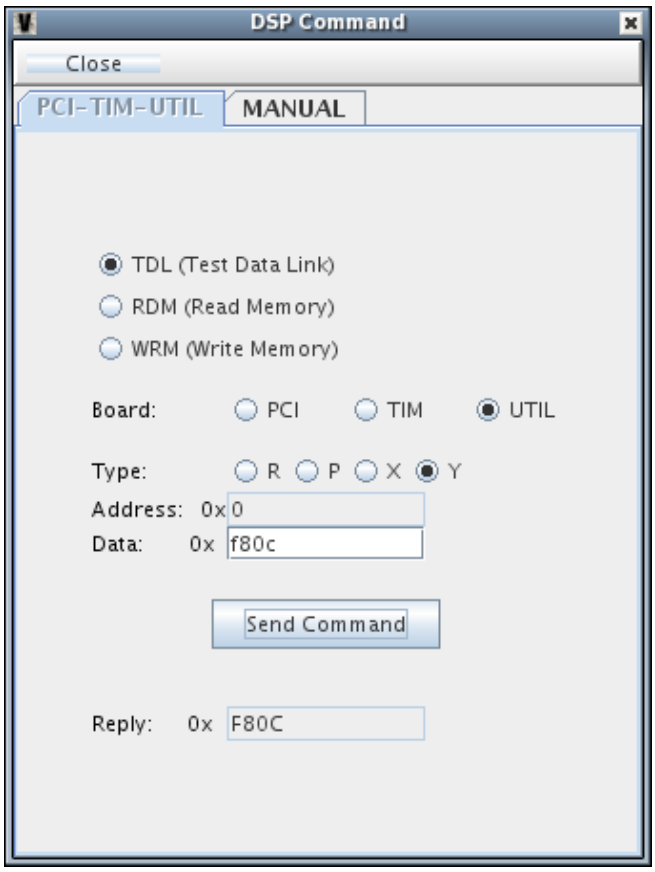
# 1.3.- UTILITY BOARD

## 1.3.1..- CHECK THE CONECTION WITH THE CONTROLLER

- In the ***main Voodoo window,*** select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window,** selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:




✔ TDL (test DATA links)
✔ Board: UTIL

-You can choose any memory type, in this case it is not important.

-Insert in the ***Data*** **box** some code in hexadecimal representation and press ***Send Command*** **button**. You should obtain the same message in the ***Reply*** **box.**

-The message in the ***main Voodoo*** **window** must be:

"Sending TEST-DATA_LINKcommand...done"

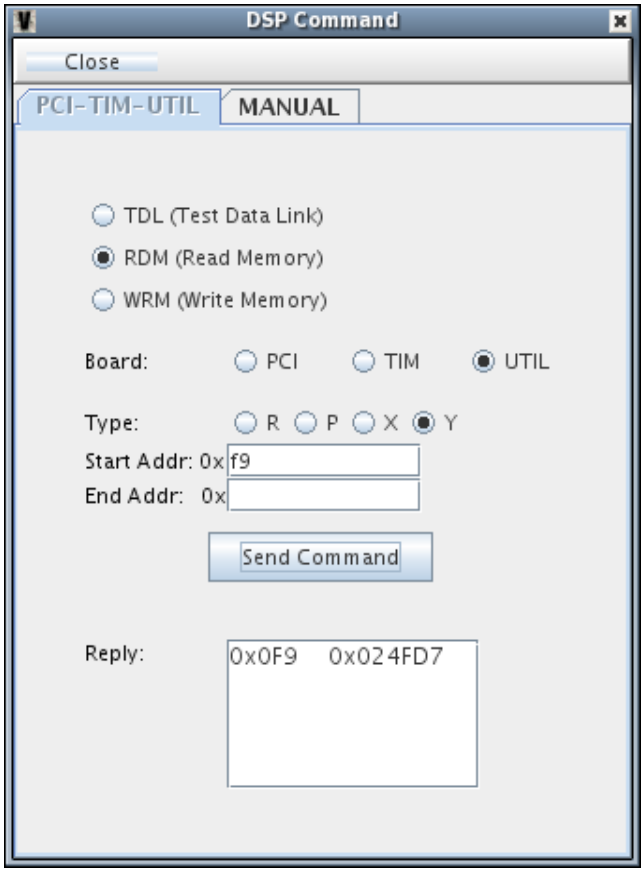
### 1.3.2.- CURRENT TEMPERATURE

- To check the utility board is correctly reading the temperature we are going to use the ***Temp Test Box*** where there is a switch with two positions corresponding to different temperatures. For example switch to 150 ºK position and read the temperature in the following way:


- In the ***main Voodoo*** **window**, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:




✔ **RDM (Read Memory)**
✔Board: **UTIL**
✔Type: **Y**
✔Start Addr: **F9**

- After that, select the ***Send Command*** **button**. We have an answer with a number in the hexadecimal representation. This is the current temperature in millikelvin. If you convert into decimal representation, you can check it is the temperature selected in the test box. Change the switch and repeat the previous steps.

## 1.3.3.- DEMANDED TEMPERATURE

- The demanded temperature is the optimal temperature at which the camera works. It is a temperature control, so if the current temperature is smaller than that value, a heater resistor is switched on.

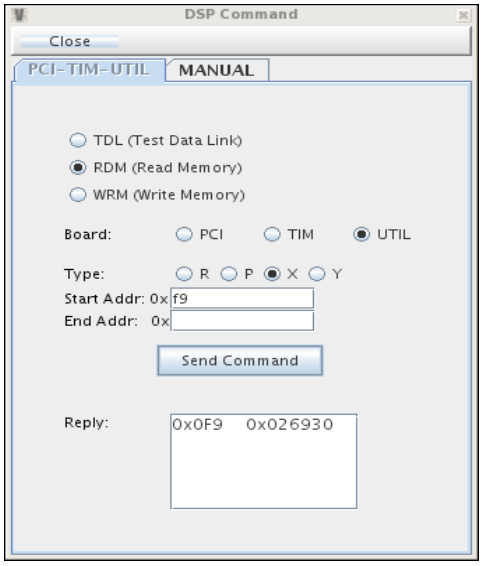
- This value is 158 ºK in the configuration file by default for all cameras (26930 in hexadecimal representation).

### 1.3.3.1.- TO READ THE DEMANDED TEMPERATURE:

- In the main Voodoo window, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:

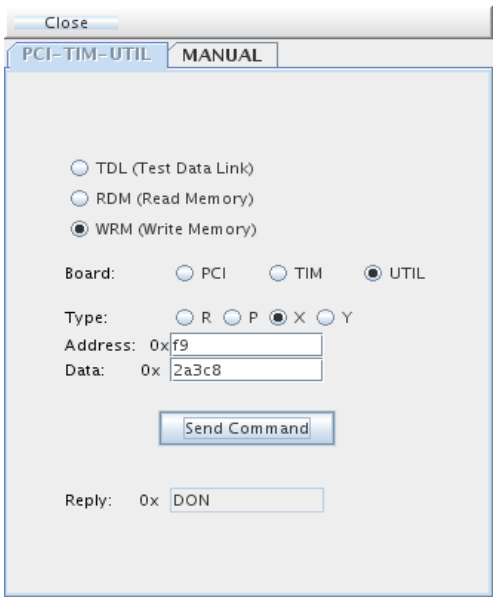



- ✔ **RDM (Read Memory)**
- ✔ Board: **UTIL**
- ✔ Type: **X**
- ✔ Start Addr: **F9**

- After that, select the ***Send Command*** **button**. We have an answer with a number in the hexadecimal representation. This is the demanded temperature in millikelvin.

- But each camera has its own demanded temperature and it is necessary you define manually this value. You can look it up in ***appendix 1*** of this document.

### 1.3.3.2.- TO WRITE THE DEMANDED TEMPERATURE:




- You can change this number in the same window selecting:

- ✔ **WRM (Write Memory)**
- ✔ Board: **UTIL**
- ✔ Types: **X**
- ✔ Start Addr: **F9**

- Insert the demanded temperature in the ***Data*** **box.** Remember that you must write it in milikelvin in the hexadecimal representation.

- When you press the ***Send Command*** **buttom,** the demanded temperature is changed in the configuration file until the controller is reset.

### 1.3.4.- HEATER

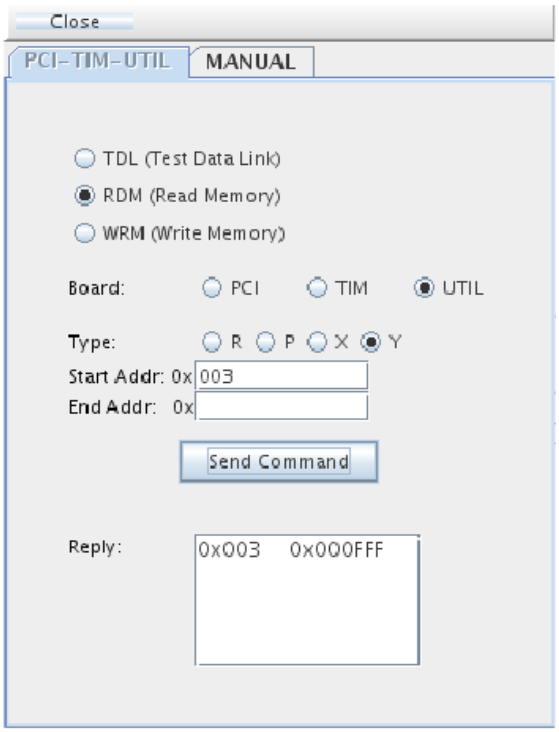
- If the current temperature is smaller than the demanded temperature, the heater resistor is switched on. You can check its status in a particular situation using the ***Temp Test Box***.

- To read the status of the heater, follow these steps:

- In the main Voodoo window, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the **PCI-TIM-UTIL tab** and mark the following options:




✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Addr: **003**

- Press the ***Send Command button***.

-If the heater is switched off, the answer will be 0, and if the heater is switched on, we will have an answer with the code FFF corresponding to 12 V. You can check this with the ***Temp Test*** **Box** where the green led will switched on when the heater is working.

- To sum:

**- 000→Heater Off→Green Led Off**
**- FFF→Heater On→Green Led On**

- If the demanded temperature is between the two temperature positions in the test box (see section 3.3), when the switch is in 150 K position, the heater will be turned on, and if it is in 193 K position, it will be turned off. This is a important thing to check the correct working of the heater.



- Moreover, it is advisable to do the following test. First, change the demanded temperature to a value bigger than the two position of the Temp Test Box, for example, 200 ºK (see section 3.3). Then, check the status of the heater. In this case, the heater should be switched on in both positions.

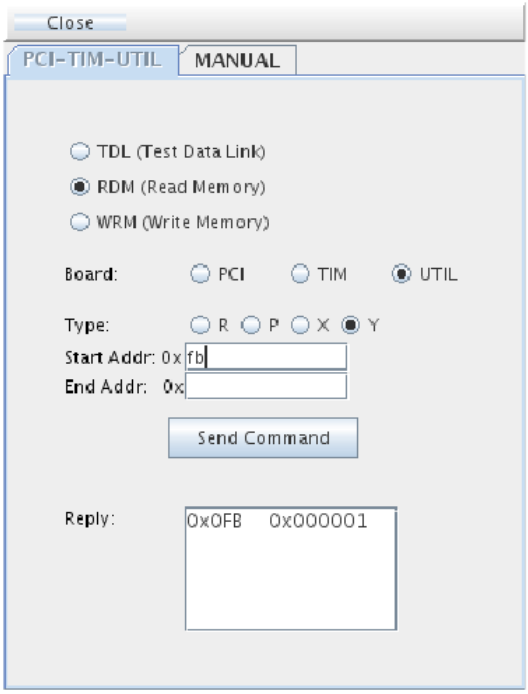
## 1.3.5.- SHUTTER

### 1.3.5.1.-TO READ THE STATUS OF THE SHUTTER

- In the ***main Voodoo*** **window**, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:




✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Addr: **FB**

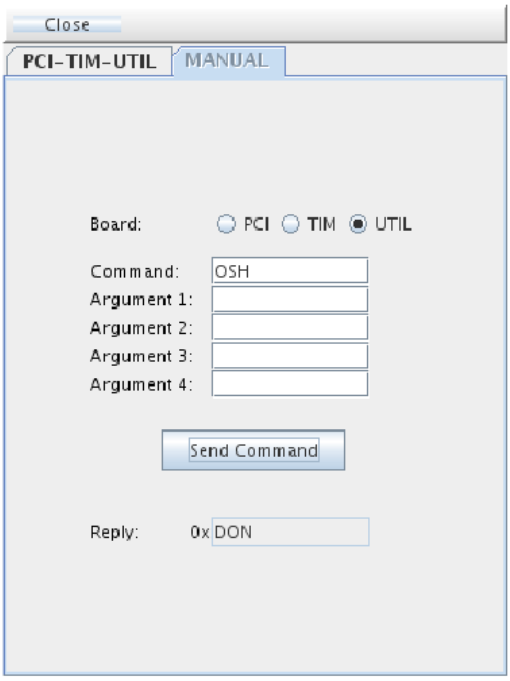
- After that, select the ***Send Command*** **button** and the answer will be:

- **0** : Shutter is **open**
- **1** : Shutter is **close**
- **2** : **Fault**

### 1.3.5.2.-TO CHANGE THE STATUS OF THE SHUTTER

- If you want to check the shutter status changes, you have to send a command to open/close the shutter:

- In the ***DSP Command*** **window**, selec the ***MANUAL*** **tab** and mark the following options:




✔ Board: **UTIL**

- Write in the ***Command*** **Box**:

- **OSH** : Open shutter
- **CSH** : Close shutter

- The answer must be “DON” in the ***Reply*** **box**. If the shutter is open, the red led in the ***Dummy Shutter*** **Box** should be switched on.

- After that, read the status again and check it has changed.

### 1.3.6.- PREFLASH

- To define the preflash command, the programme uses two spaces in the memory.
The **demanded preflash** time is stored in **X:FA**. This value is the time during the preflash led is switched on. It is two seconds in the configuration file by default.
The **current preflash** time is stored in **Y:FA**. When you send a preflash command, this parameter becomes zero. It starts to count until it is the same as the demanded preflash time. When the two values are equal, the preflash command finishes.

#### 1.3.6.1.- TO READ THE DEMANDED PREFLASH TIME

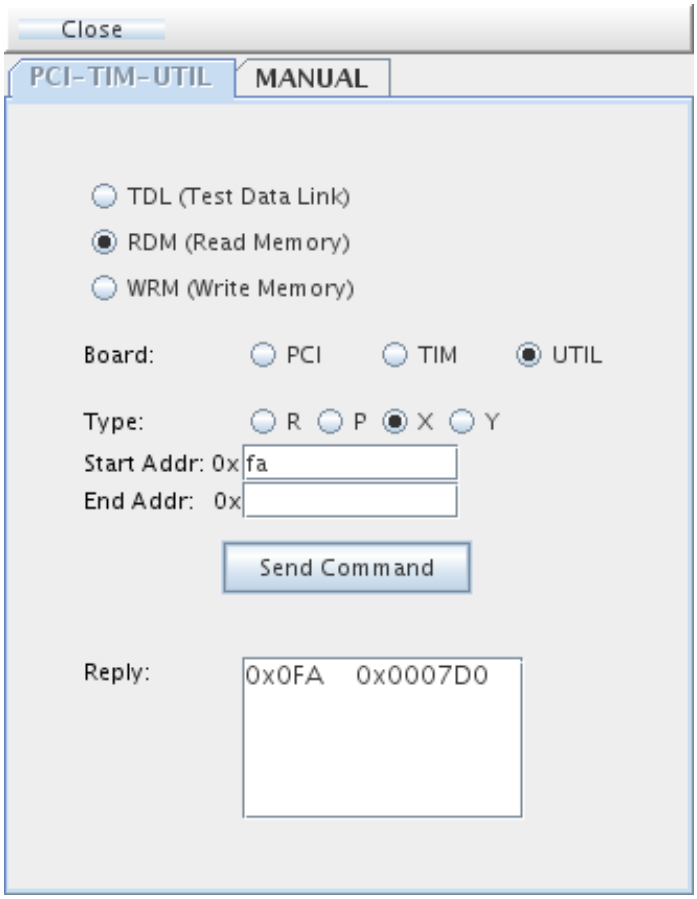
- In the ***main Voodoo*** **window**, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **X**
✔ Start Addr: **FA**

- After that, select the ***Send Command*** **button** and the answer should be **7D0** which corresponds to two milliseconds:



1.3.6.2.- PREFLASH COMMAND

- You can do a preflash with a manual command.

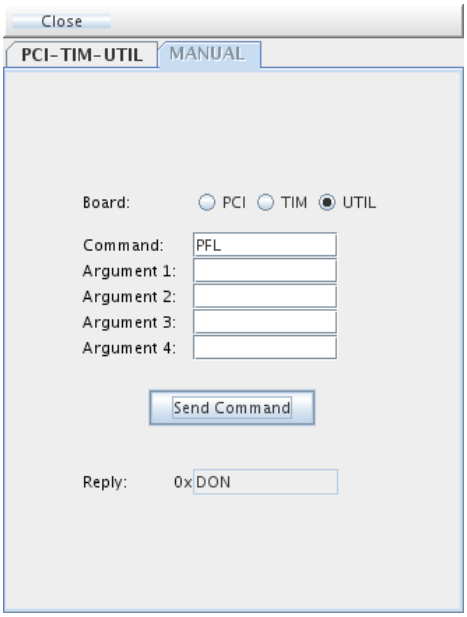
- In the ***DSP Command*** **window**, select the ***MANUAL*** **tab** and select:

✔ Board: **UTIL**

- Write in the ***Command*** **Box**:

**- PFL**

- You can check the preflash is working with the ***Temp Test*** **box**. The red led will switch on during the preflash time.

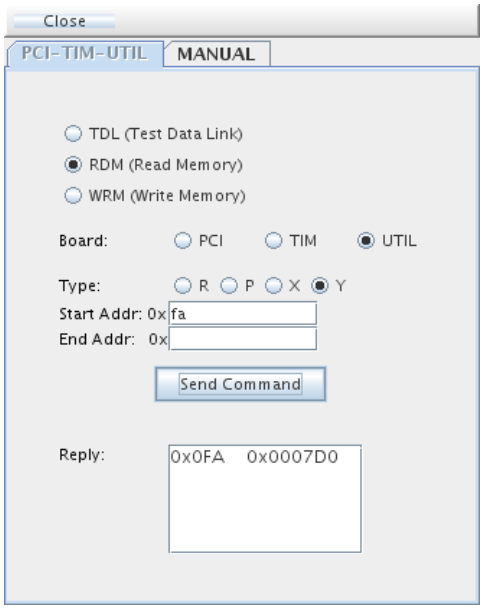




1.3.6.3.- TO READ THE CURRENT PREFLASH TIME

- If you were fast enough, it would be possible read the Y:FA space memory while the preflash command is running. You would see how the time is counting from zero to the demanded preflash time. But it is enough if you read the current preflash time after the preflash command finishes. In this case, you should check that the two times are the same.
- In the ***DSP Command*** **window**, select the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Addr: **FA**



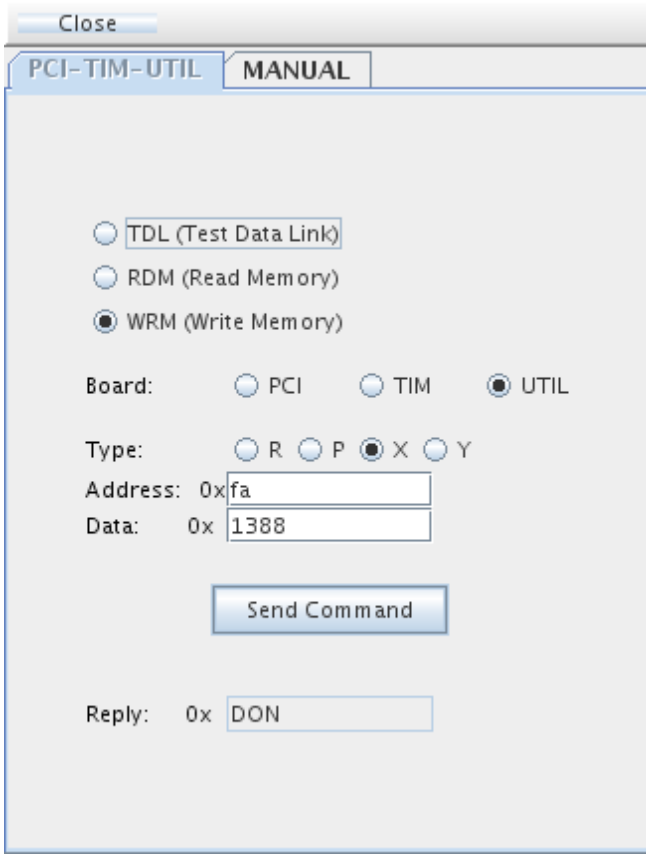
### 1.3.6.4.- TO CHANGE THE DEMANDED PREFLASH TIME

- Another test it is advisable to do is change the demanded preflash time and check the red led in the Temp Test Box is switched on during the new defined time. For this:

- In the ***DSP Command*** **window**, select the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **WRM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **X**
✔ Start Addr: **FA**

- Introduce in the ***Data*** **Box** the new demanded preflash time which has to be written in milisecond in the hexadecimal representation. For example, five second is 1388. Press the **Send Command** ***button*** and check the ***Temp Test*** **Box**.



1.3.6.5.- FORCE THE PREFLASH STATUS

- The preflash status is saved in **Y:1**, so it is possible to force the preflash led to switch on by writing in this memory space. In this case, the red led will turn on until you change again this memory space or reset the controller.

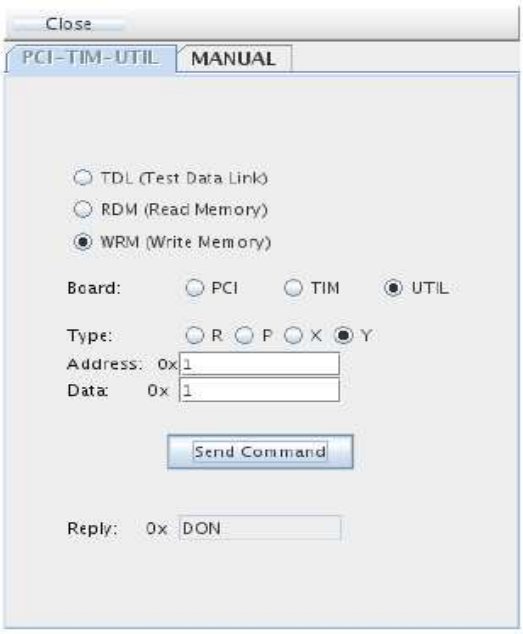
- In the ***DSP Command*** **window**, selec the PCI-TIM-UTIL tab and mark the following options:

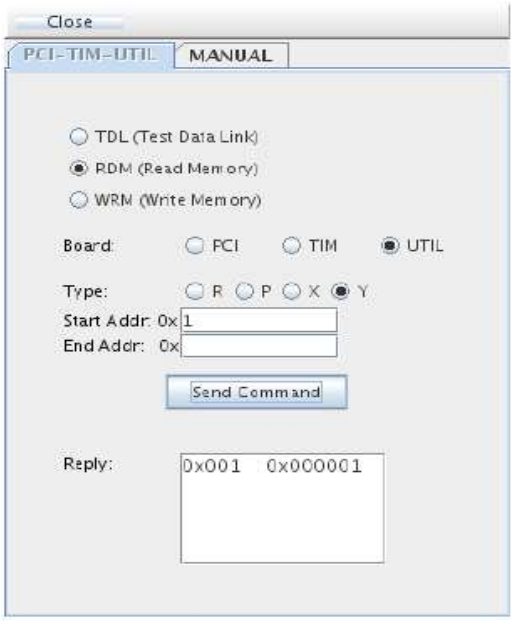
✔ **WRM (Write Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Addr: **1**

- Write in the ***Data*** **box**, **1** to switch **on** the preflash and **0** to switch it **off**. To sum:

**- 0→Preflash Off→Red Led Off**

**- 1→Preflash On→Red Led On**




1.3.6.6.- READ THE PREFLASH STATUS

- You can check the status in that space of the memory and compare with the led in the Temp Test Box.
- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **RDM (Write Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Addr: **1**

– If the answer is **1**, the preflash is switched **on** and if it is **0**, the preflash is switched **off.**

# 2.- CONTROLLERS FOR AUTOGUIDER CAMERAS

## 2.1.- ASSEMBLING THE SYSTEM

- After a controller has been repaired and before it is used in the telescope, it has to be check in the laboratory.

- The first step is reproduced the system in the laboratory. You need to assembling it:

  - The **Controller**
  - A **UltraDas machine**
  - A **Double Power Supply**, one for the controller and the other for the peltier.

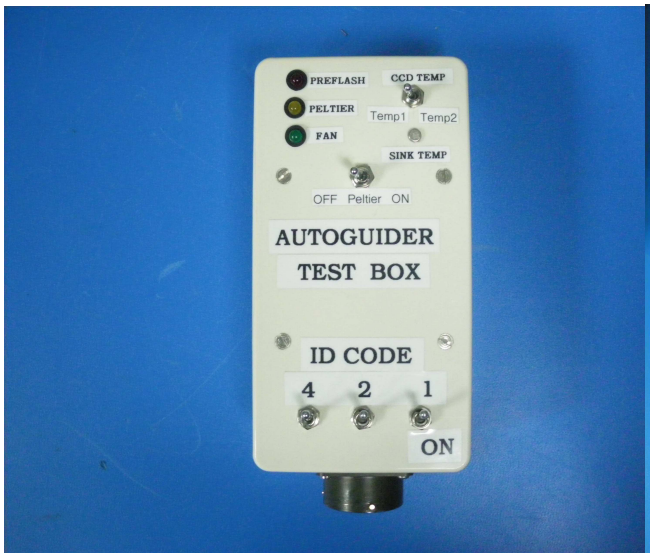
- You can see these components in the following photos:


- To join these components it is necessary the following wires:

  - **Doble mains cable**
  - **PowerSupply cable**
  - **Doble Optical Fiber:** It is necessary a doble fiber, one to send information to the Ultradas machine and the other to receive from it in the controller.



-As we don't know if the controller has been well repared, it is not advisable use a CCD camera because it may be damaged. So, the first check must be done with a ***Autoguider Test Box*** connected to the frontside of the controller with its corresponding wire:




- The Test Box simulates the Autoguider Camera:

  - **ID Code:** Each camera has its own ID code with two character in hexadecial representation. The first one is 5 in all autoguider cameras. The second one goes from 0 to 7. You must move the switches in the Test Box to reproduce this character in binary representation.

  - **Sink Temperature:** To simulate the sink temperature there is a temperature sensor .

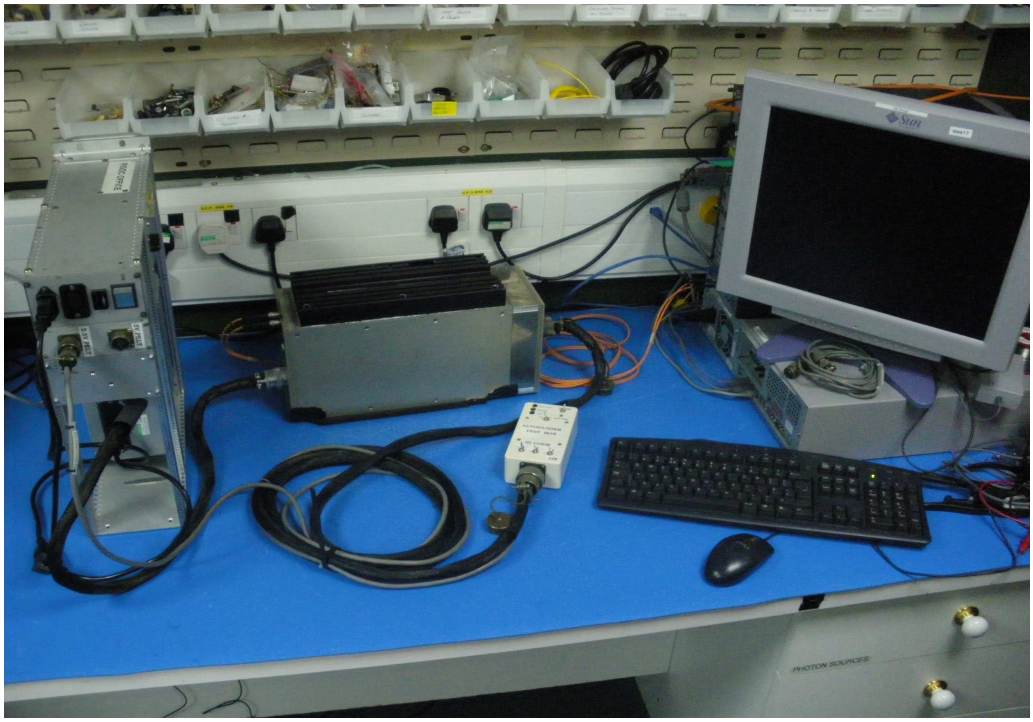
  - **CCD Temperature:** There is a switch which reproduces two different temperatures.

  - **Preflash:** The red led is switched on when the preflash is turned on.

  - **Peltier:** There is a switch that must keep pushed to the peltier is working. In this case the yellow led is turned on.

  - **Fan:** The green led is turned on when the fan is working. In our case is always.

- You can see the whole system in the photo below:



- Turn on the power supplies. Check the two leds in the UltraDas backside. They must be turned off. In other case, exchange the position of the fibers.


- Now, the system is ready to test the controller with Voodoo.

## 2.2.- STARTING THE SYSTEM

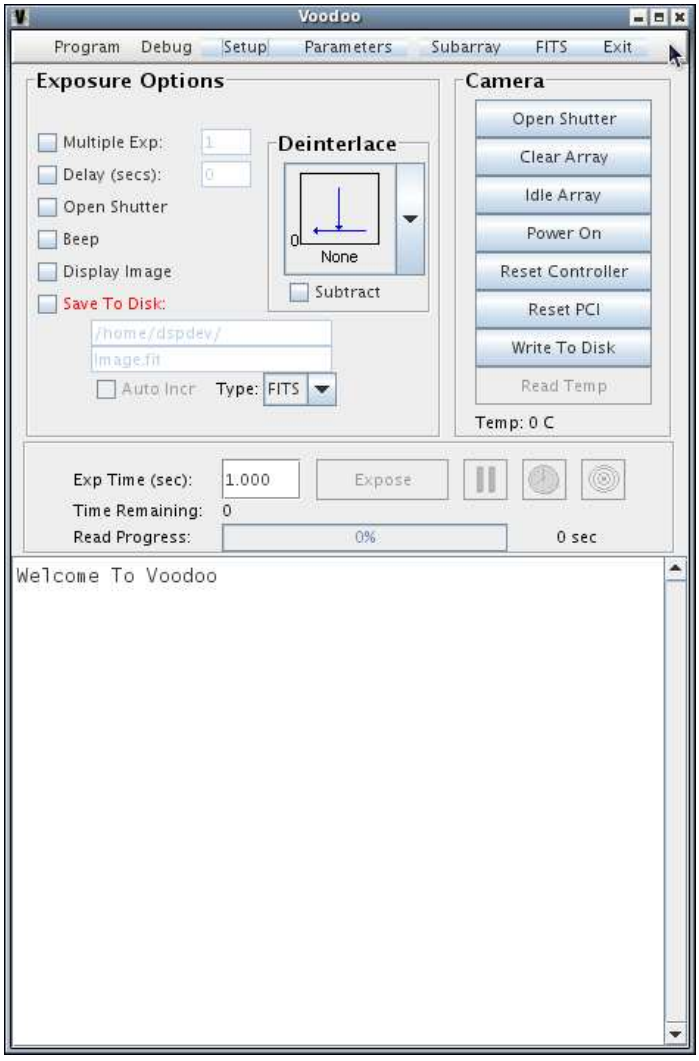
### 2.2.1.- STARTING VOODOO

- After assembling the system, you must turn on the power supply of the controller and the peltier. The UltraDAS machine is normally turned on in the laboratory, so you only have to start with the account:

Login: **DSPDEV**
Password: *******

- Voodoo can be started up writing in a terminal: > **voodoo &**

- The ***main Voodoo*** **window** is shown below.

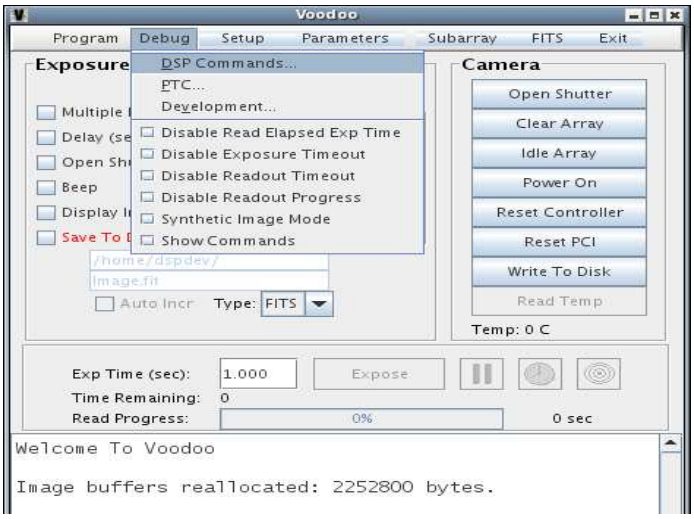


## 2.2.2.- CAMERA ID CODE:

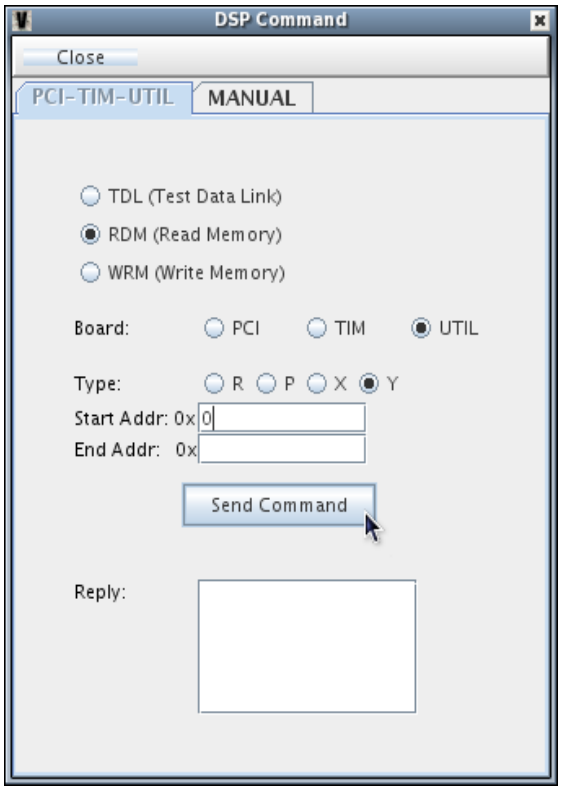
- **NOTE:** It is very important to make sure which camera we are working with and load correctly the configuration file. Otherwise, the controller boards and the camera can be damaged!!

- Once the ***main Voodoo*** **window** appears, select in the drop down menu:

***Debug→DSP Command***




- The ***DSP Command*** **window** will then appear:




- Mark the following options in the ***PCI-TIM-UTIL*** **tab**:

✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔Start Addr: **0**

- After that, select the ***Send Command*** **button**.

- We have an answer with a code. We are interested in the last two numbers which are the camera code in hexadecimal representation. You can check it in ***appendix 1*** of this document or in the following web page:

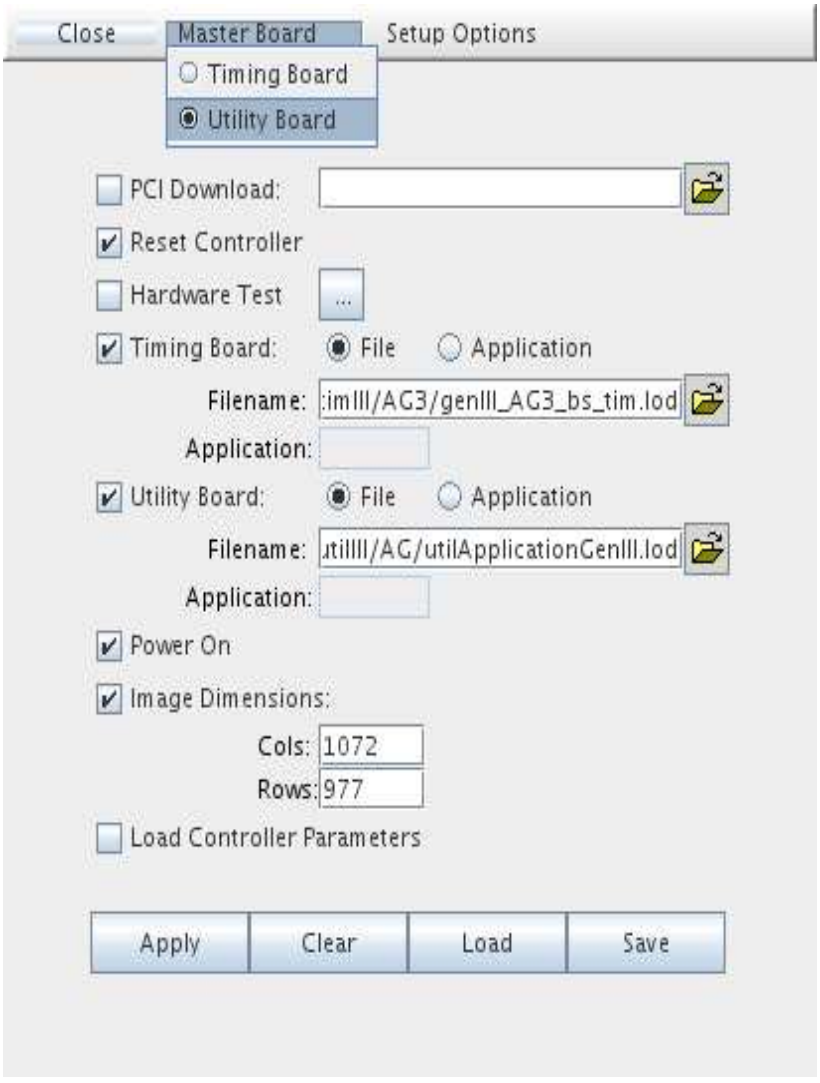
http://www.ing.iac.es/~eng/detectors/SDSU/detector_id.html

### 2.2.3.- LOAD THE CONFIGURATION FILE

- Select in the drop down menu of the ***main Voodoo*** **window**:

***Setup***

- The ***Controller Setup*** **window** starts up.



- Mark the following options:

You have to resize the window manually. Select in the drop down menu:

***Master Board→Utility Board***

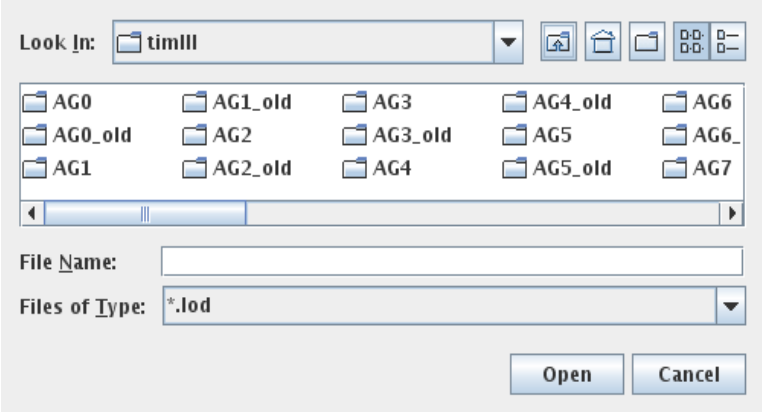
✔ **Reset Controller**

✔ **Timing board→File**

Press the icon in the right of the box to search the file. A window starts up. You must go to the following directory:

***/home/dspdev/ccd/timIII/***

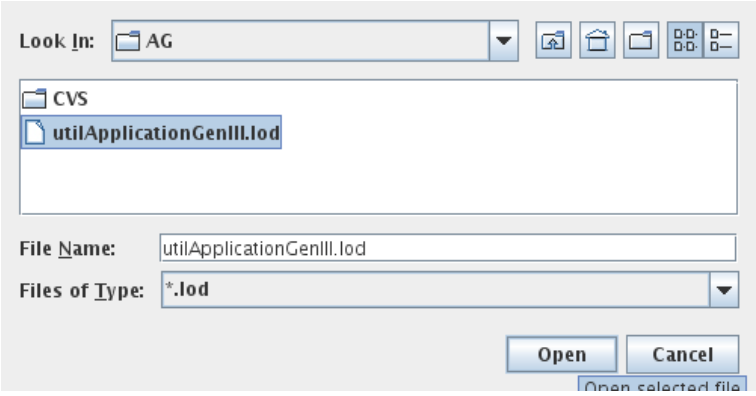
Here, there is a list of folders for each camera. Enter in the corresponding folder to the camera you are using and select the most recent file.




✔ **Utility board→File**

Press the icon in the right of the box to search the file. In this case, the file is the same for all cameras. So, search in the emerging window the following file:

***/home/dspdev/ccd/utilIII/AG/utilApplicationGenIII.lod***




✔ **Power on**

✔ **Image dimensions**

It refers to the digitised area of the camera. You can find this information in **appendix 1** or in the following web page:

http://www.ing.iac.es/Engineering/engweb6a.htm

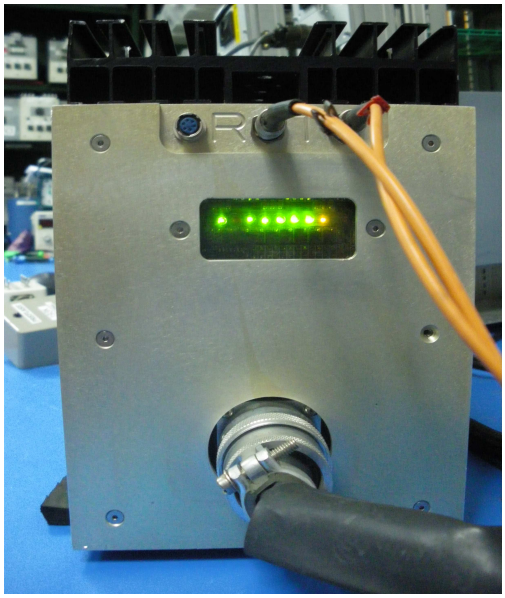
Fill the box for the columns corresponding to the number of pixel in the x direction and the rows that represent the y direction.

- You can save the configuration pressing the ***Apply*** **button**. This action will then download the code and power on the controller. The main Voodoo window will show the progress of the download and power on. Notice that there is a small error message:

"Checking Controller Configuration.....failed"

- Don't worry about this, it doesn't matter.

- Finally, to be sure that the controller is powered on, verify the leds on the power control board are switched on, with the exception of the second one.

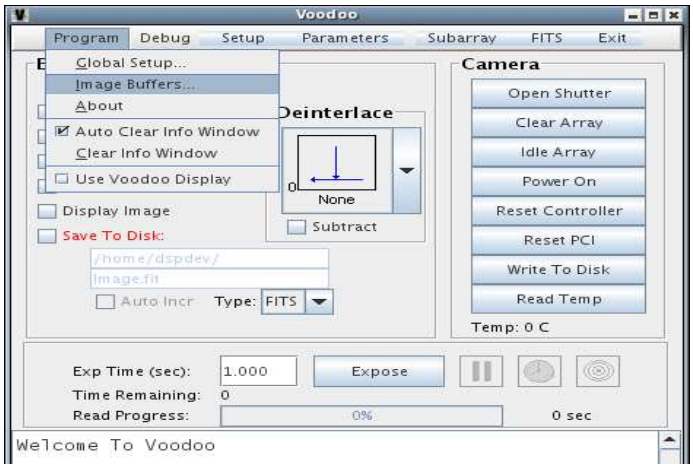


## 2.2.4.- IMAGE DIMENSION

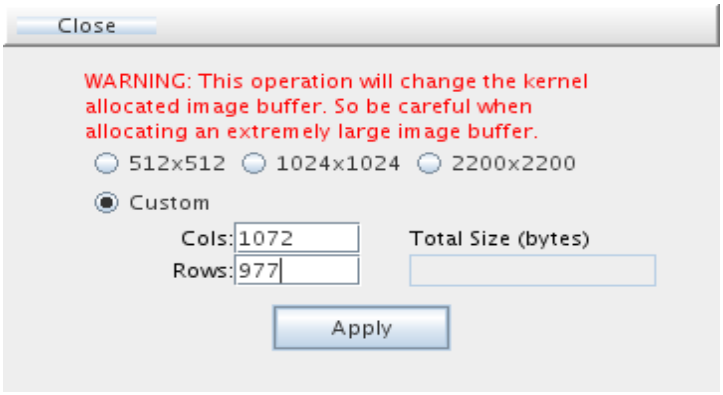
- Although you have defined the image dimensions in the configuration file for the controller, it is necessary to do again for the buffer in the Das machine.

- Select in the drop down menu of the ***main Voodoo*** **window**:

***Program→Image Buffers***




- The emerging window doesn't start up with the correct size and not all the buttons are visible, so you have to resize the window manually.




- Mark

✔ ***Custom***

- And then fill the box for columns and rows which represent pixels in the x and y direction respectively. If you are not sure of the image dimensions for the camera you are using, then refer to ***appendix 1*** of this document.

- Press the ***Apply*** **button** and close the window.

- Sometimes Voodoo crashes when you do that. Don't worry but you have to start again.

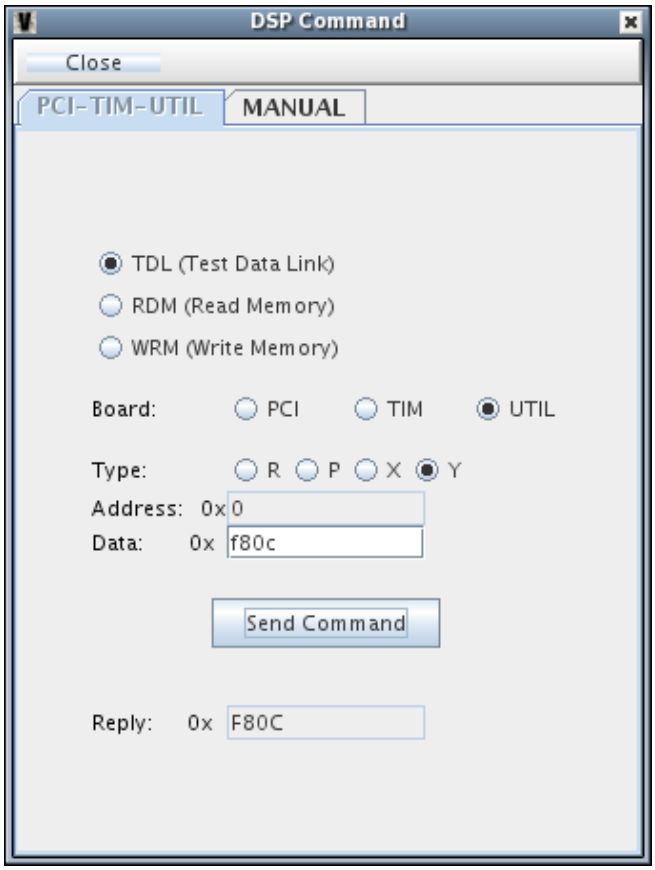
## 2.3.- UTILITY BOARD

### 2.3.1..- CHECK THE CONNECTION WITH THE CONTROLLER

- In the ***main Voodoo window,*** select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window,** selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:




✔ TDL (test DATA links)
✔ Board: UTIL

- You can choose any memory type, in this case it is not important.

-Insert in the ***Data*** **box** some code in hexadecimal representation and press ***Send Command*** **button**. You should obtain the same message in the ***Reply*** **box.**

- The message in the ***main Voodoo*** **window** must be:

"Sending TEST-DATA_LINKcommand...done"

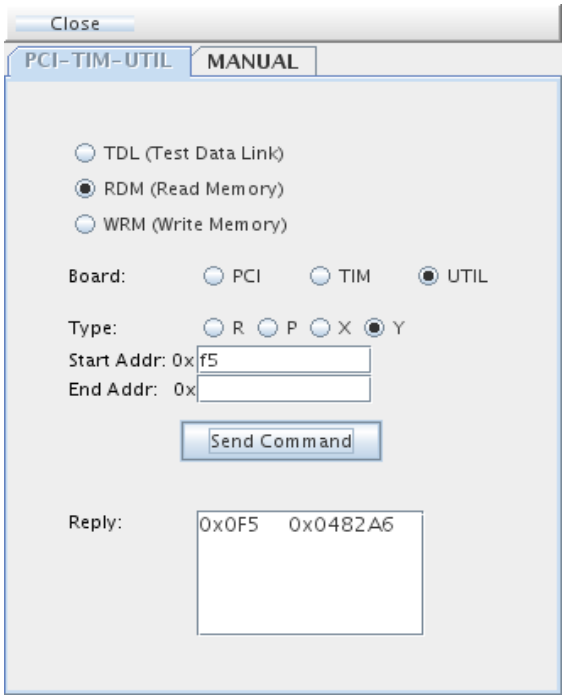
## 2.3.2.- SINK TEMPERATURE

- The Autoguider cameras have two temperature, the sink temperature and the CCD temperature. To check the utility board is correctly reading the temperature we are going to use the ***Autoguider Test Box***. There is a temperature sensor in it which simulate the sensor inside the camera. Read the temperature in the following way:

- In the ***main Voodoo*** **window**, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:




✔ **RDM (Read Memory)**
✔Board: **UTIL**
✔Type: **Y**
✔Start Addr: **F5**

- After that, select the ***Send Command*** **button**. We have an answer with a number in the hexadecimal representation. This is the sink temperature in millikelvin. It must be similar to the room temperature.

- Now, put your finger over the sensor and wait some seconds. Read the temperature again and check that it is bigger than the first time.

### 2.3.3.- CCD TEMPERATURE

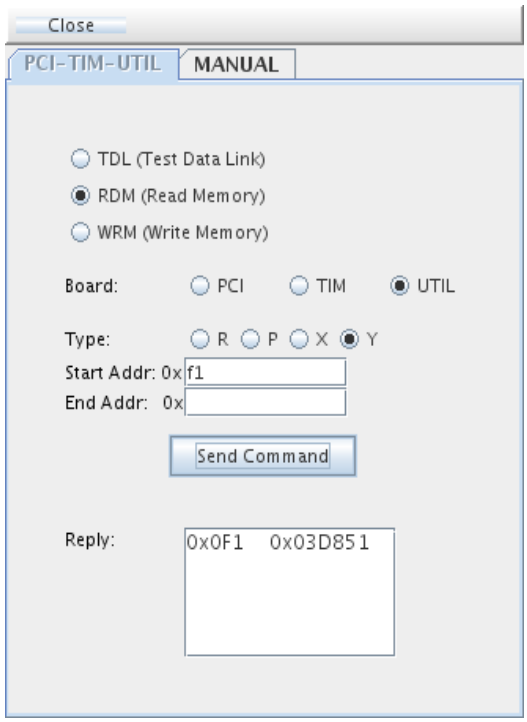
- The CCD is cooled with the Peltier effect. The Autoguider cameras don't have a temperature control because the minimum temperature reach with this method it is enough for its use. The Test Box has a switch with two positions corresponding to different temperatures. Read them in the following way:

- In the main Voodoo window, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Addr.: **F1**




- After that, select the ***Send Command*** **button**. We have an answer with a number in the hexadecimal representation. This is the CCD temperature in millikelvin.

- Change the switch and check that you read a different temperature.

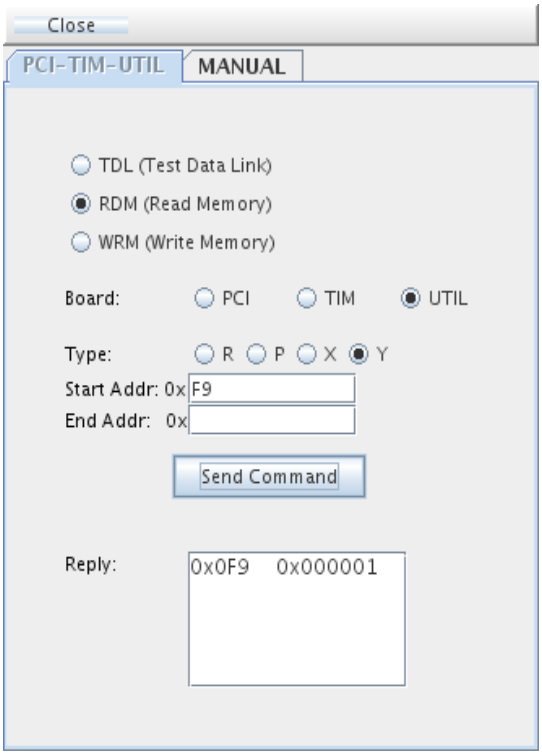
### 2.3.4.- PELTIER

- You can test the peltier status with the ***Autoguider Test Box.*** If you keep pushing the Peltier switch , the green led will be switched on. Check the status with Voodoo:

- In the main Voodoo window, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the PCI-TIM-UTIL tab and mark the following options:




✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Addr: **F9**

- Press the ***Send Command button***.

- And the answer will be:

**- 0→Peltier Off→Green Led Off**

**- 1→Peltier On→Green Led On**

## 2.3.5.- PREFLASH

- To define the preflash command, the programme uses two spaces in the memory.
The **demanded preflash** time is stored in **X:FA**. This value is the time during the preflash led is switched on. It is two seconds in the configuration file by default.
The **current preflash** time is stored in **Y:F2**. When you send a preflash command, this parameter becomes zero. It starts to count until it is the same as the demanded preflash time. When the two values are equal, the preflash command finishes.

### 2.3.5.1.- TO READ THE DEMANDED PREFLASH TIME

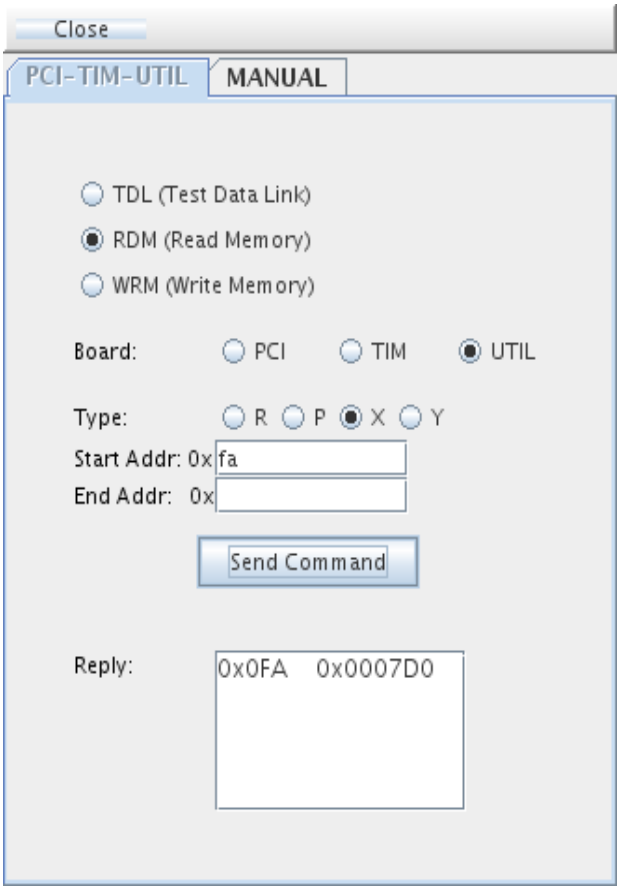
- In the ***main Voodoo*** **window**, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **X**
✔ Start Addr: **FA**

- After that, select the ***Send Command*** **button** and the answer should be **7D0** which corresponds to two milliseconds:



2.3.5.2.- PREFLASH COMMAN

- You can do a preflash with a manual command.

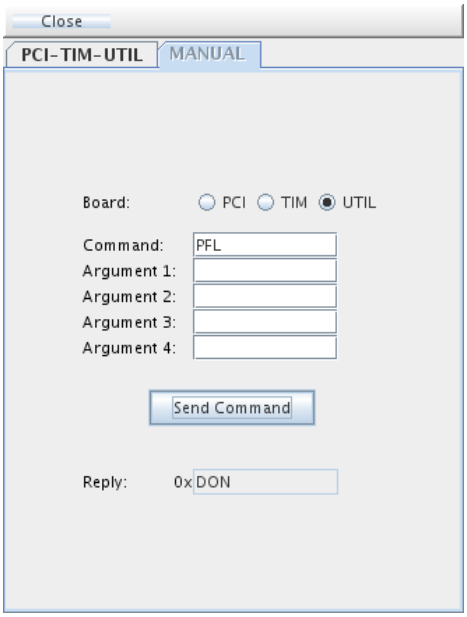
- In the ***DSP Command*** **window**, select the ***MANUAL*** **tab** and select:

✔ Board: **UTIL**

- Write in the ***Command*** **Box**:

**- PFL**

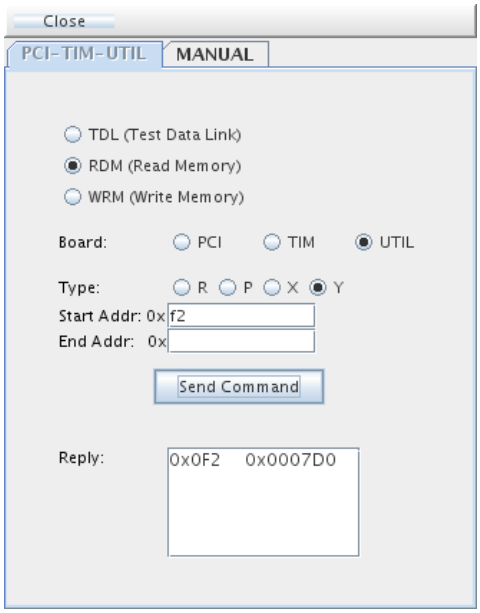
- You can check the preflash is working with the ***Autoguider Test*** **box**. The red led will switch on during the preflash time.




2.3.5.3.- TO READ THE CURRENT PREFLASH TIME

- If you were fast enough, it would be possible read the Y:F2 space memory while the preflash command is running. You would see how the time is counting from zero to the demanded preflash time. But it is enough if you read the current preflash time after the preflash command finishes. In this case, you should check that the two times are the same.
- In the ***DSP Command*** **window**, select the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **RDM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Addr: **F2**



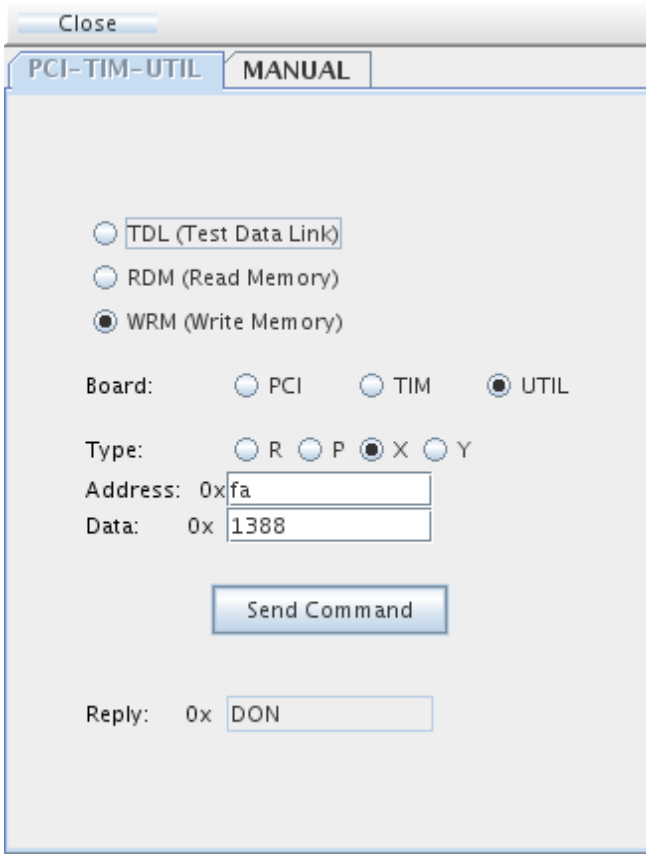
2.3.5.4.- TO CHANGE THE DEMANDED PREFLASH TIME

- Another test it is advisable to do is change the demanded preflash time and check the red led in the Test Box is switched on during the new defined time. For this:

- In the ***DSP Command*** **window**, select the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **WRM (Read Memory)**
✔ Board: **UTIL**
✔ Type: **X**
✔ Start Addr: **FA**

- Introduce in the ***Data*** **Box** the new demanded preflash time which has to be written in milisecond in the hexadecimal representation. For example, five second is 1388. Press the **Send Command** ***button.*** Send a Preflash command and check the led in the ***Test*** **Box**.



### 2.3.6.- SHUTTER

- The Autoguider Cameras don't use shutter, so if you read the status, the answer must be fault.

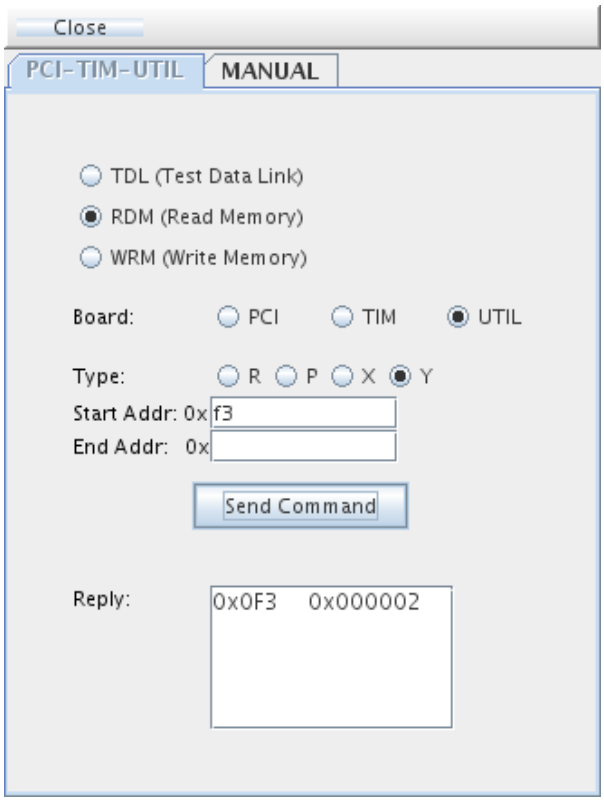
- In the ***main Voodoo*** **window**, select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window**, selec the ***PCI-TIM-UTIL*** **tab** and mark the following options:

✔ **RDM (Read Memory)**

✔ Board: **UTIL**
✔ Type: **Y**
✔ Start Adres: **F3**

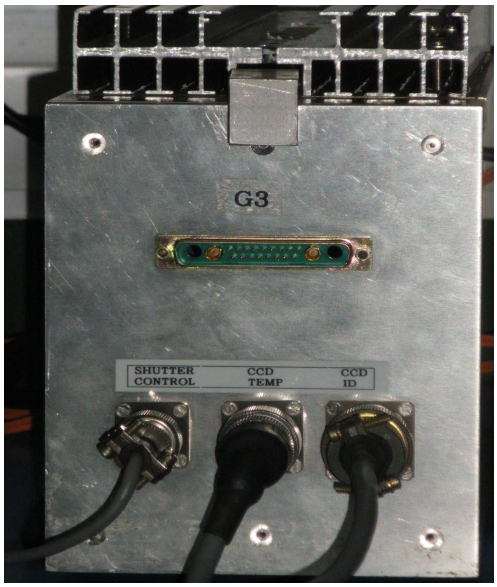



- After that, select the ***Send Command*** **button** and the answer will be**:**

**- 2** : **Fault**

# 3.- TESTING THE CONNECTOR

- Before a camera is connecting to the controller, the output voltages of the signal connector must be tested. Otherwise, the camera may be damaged. You can see this conector in the photo below:




- Each pin of the conector has a number. The upper pin on the left is the number one and you count from left to right.

- Each kind of camera has diferent range of voltage, but the pins in the connector have the same function in all of them. You can find the table with the voltages for each camera in:

http://www.ing.iac.es/~eng/detectors/SDSU/hardware/SDSU_wiring.htm

- Some of these tables may not have been updated, so take them as orientative numbers. Don't try to observe the exact number, although they must be similar. Exact voltage values can be found in the code.

- After choose a camera, the first step is starting the system with the test boxes. *See section 2* of this document.

- There are two types of voltages, the *Bias* which are continuos voltages and the *Clocks* which are changing voltages. So you need an oscilloscope as well.

- Turn on the oscilloscope and configure it:

- You must measure in **DC mode**.
- Fix the voltage division (the vertical axis)in **5 V/div**.
- Fix the time division (the horizontal axis)in **5 µs/div**.
- Set the trigger level in **-1 V**, approximately.

- If you don't see the voltages with these settings, change them until the signal appears in the oscilloscope screen.

## 3.1.- CLOCK VOLTAGES

- It is convenient if you know the basic theory of how a CCD works, to know what you are doing and to understand the meaning of the clocks. It is going to be briefly explained in this document, but if you need further information about it you can look up the book *"Electronic Imaging in Astronomy", Ian S. McLean.*

- When you take an exposure the first command sent is to clear the charge in the CCD. This is done with the Dump Gate***, DG voltage***. After an exposure, the readout process is done by an output amplifier usually placed in a corner of the CCD. This process involves a number of clocks to transfer the charge from the pixels to the output amplifier. These clocks are usually in group of three (three fase CCD). The process is split in two big tasks: move lines to the output register (***vertical clocks***) and move pixels of the output register to the output amplifier (***horizontal clocks***). The transported charge is collected in the summing well and is read out by measuring the voltage inside the well. The cycle to read the charge begins with reseting the summing well by firing the ***reset voltage***. Once the summing well is reset, the charge is then dropped into it with the ***SW voltage***.

- It is possible to observe these voltages with an oscilloscope. But it is necessary the CCD is in ***idling mode*** to measure them. You can do this with Voodoo.

- In the ***main Voodoo window,*** select in the drop down menu:

***Debug→DSP Command***

- In the ***DSP Command*** **window,** selec the ***MANUAL*** **tab** and select:

✔ Board: **TIM**

- Write in the ***Command*** **Box**:

- **IDL** : Start idling mode.
- **STP** : Stop idling mode.

- The answer must be "DON" in the ***Reply*** **box**.

- Measure the voltages in the followings pins:

- ***PIN.4.- Reset Voltage***

  We are going to use it as reference. So, take another channel in the oscilloscope and check that both channel measure the same signal. Use the second channel to measure the others pins.

- ***PIN.1.- SW voltage***

- ***PIN.5-6-7.- Horizontal clocks***

- ***PIN.8-9-17.- Vertical clocks***

  As a vertical clock move a line down to the output register, while the horizontal clock dump the charge of all pixels of the output register before another vertical clock is sent, you must change the configuration of time division in the oscilloscope. Fix the time division in **5 ms/div**.

- ***PIN.3.- DG voltage***

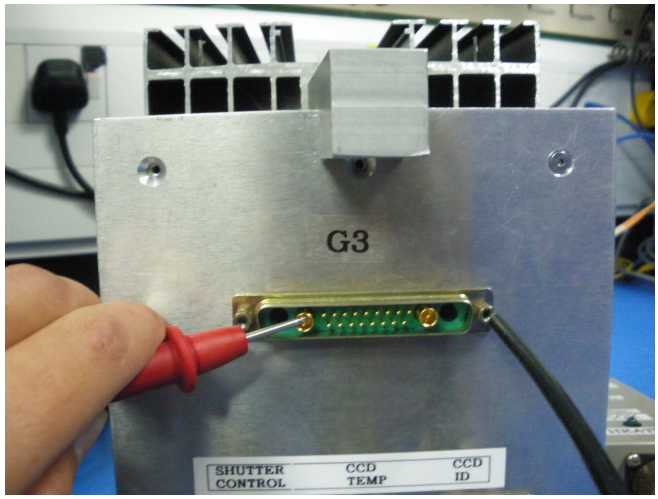
  Fix the time division in 5 µs/div It is necesary take an exposure to measure this voltage. You can do that with Voodoo.
  In the ***main Voodoo*** **Window** set the exposure time, for example 1 s, and select the ***Exposure*** **button**.
  You can only measure this voltage just before sending the command. This is not a continous clock but a single pulse as part of the clear.

## 3.2.- BIAS VOLTAGES

- After checking the Clock voltages, you can use the oscilloscope and the table in the web to measure the rest of pins which are the Bias or continuos voltage.

## 3.3.- LOAD RESINTANCE

- The last test before connecting a camara is measure the resistance in the video pins of the connector. Turn of the controller and check they are 10 kΩ.



# 1.- TAKING AN EXPOSURE

## 4.1.- INTRODUCTION

- The last step to test the controller is connect a camera and check that the electronics of the controller doesn't modify the feature of the camera. You can find the information for each detector in:

http://www.ing.iac.es/Engineering/engweb6a.htm

- Start with the account:

Login: **WHT observer**
Password: *******

- Configure the camera you are going to use. Write in a ***terminal*** **window**:

>**obssys -c**
>**c**
>**1**

- And modify the file. You can use ***Dewars*** to configure the camera as well.

-. After that, start the ***Observing system***:

>**startobssys**

## 4.2.- STARTING IRAF

- Besides, you need to use ***Iraf***. Write in a terminal window to start it:

>**xgterm &**

- And the ***xgterm*** **window** starts up. Write here:

**>ds9 &**

- The ***SAO Image*** **window** starts up.

- To start Iraf, write in the xgterm window:

**>ecl**

- Now, you are in Iraf. Check that you are in the directory in which you want to work.

## 4.3.- BIAS AND NOISE

- The **Biasframe** is taken to measure the zero point of the pixels. It is a zero exposure with the shutter close. The read noise is caused by the electronic noise produced in the output transistor of the CCD. This signal is the biaslevel and should be subtracted from the image.

- Choose one of the two readout speed writing in the ***terminal*** **window**:

>**rspeed fast/slow**

- And make a bias:

>**bias**

- You can check the image with **Iraf** writing in the ***IRAF*** **window**:

>**display *ImageName*[1] 1 fill+;imexam**

- You can check the bias and noise level pushing the m key with the cursor over the image. Compare with the datas in the webpage.

- Repeat the process for the other readout speed.

- If you want further information about IRAF, you can look up *A beginner's Guide to Using IRAF.*

## 1.1.- GAIN

- The observed quantity in a experiment is the stream of photons, but the detected quantity is a small voltage which is amplified and digitized. The conversion gain of a CCD camera is the number of electrons represented by each digital interval (ADU) of the analogue to digital converter.

- We calculate the gain using the **photon transfer method** with a script which is stored in:

***/homedspdev/scripts/gain.cl***

- If you want to know more information about the method used, you can look up in:

http://www.ing.iac.es/~eng/detectors/QC/qc.htm

- You can see also:

http://www.ing.iac.es/~eng/detectors/QC/quickcheck.htm

- Copy this file in the folder in which you are working.

- We need **two flats** to use this script. The sensitivity of the chip may vary from pixel to pixel. To filter out this difference a flat field is used. This is a image which is uniformly illuminated.
Save the two image in your folder.

- A **format file** is also required in the same directory as the images. Its name must be ***cform***. The format file must contain the following information:

- bias-strip-x-location-start bias-strip-x-location-end bias-strip-y-location-start bias-strip-y-location-end

- image-area-x-location-start bias-strip-x-location-end bias-strip-y-location-start bias-strip-y-location-end

- statistics-box-size-x statistics-box-size-y <RETURN>

- Run the script ***gain.cl*** writing in the **IRAF window**:

>**gain**

- And you get the results for the gain, bias and noise.

## 4.5.- DARK CURRENT

- The **Darkframe** is used to find out the dark current present in the CCD. This current is caused by the thermal vibration of the atoms which every electronic device is affected by.This kind of frame is an exposure in complete darkness with the same integrationtime as the science image.

- So, take an exposure of 1 hour. Write in a **terminal window**:

>**Dark 3600**

- You need a ***Bias frame*** as well:

>**Bias**

- The Dark Current is the different between the count level in the Dark Frame and the Biasframe.

- To express the dark current in electrons, multiple by the gain.

## 4.6.- LINEARITY

- CCD cameras are quite linear detectors over an immense dynamic range. That is, the output voltage signal is exactly proportional to the amount of light falling on the CCD to very high accuracy.

- We can make the linearity curves representing in a graph the time exposure in the x axes and the count level minus the bias level in the y axis. For this, take different exposures increasing the time exposure until the image is almost saturate.

- The graph would be a straight line for a perfect detector. So, you can also plot the deviation with respect to a straight line against the count level minus the bias level.

- To take an exposure, write in the ***terminal*** **window**:

>**run** ***exposuretime***

# APPENDIX.1:

# DETECTOR INFORMATION

- You can find in the table the necessary information for the detectors which are in the laboratory. If you want further information or you are using a detector that is not in the table, look up the followings webpages:

http://www.ing.iac.es/~eng/detectors/SDSU/detector_id.html

http://www.ing.iac.es/Engineering/engweb6a.htm

| SCIENCE CCD CAMERA | ID CODE | | DEMANDED TEMPERAT. (ºK) | DIGITISED AREA | |
|---|---|---|---|---|---|
| | BINARY | HEX. | | X PIXELS COLUMNS | Y PIXELS ROWS |
| **TEK2** | 0000- 0010 | 02 | 168 | 1100 | 1040 |
| **TEK5** | 0000- 0101 | 05 | 168 | 1100 | 1040 |
| | | | | | |
| **SITe1** | 0001- 0001 | 11 | 168 | 2088 | 2120 |
| | | | | | |
| **EEV10a** | 0010- 1010 | 2A | 153 | 2148 | 4200 |
| **EEV12** | 0010- 1100 | 2C | 158 | 2148 | 4200 |
| **EEV13** | 0010- 1101 | 2D | 153 | 2148 | 4200 |
| | | | | | |
| **WFC** | 0011- 0001 | 31 | 153 | 2154 | 4200 |
| **WFCWHT** | 0011- 0010 | 32 | 153 | 2154 | 4200 |
| | | | | | |
| **QUCAM2** | 0100- 0011 | 43 | 170 | 1080 | 1050 |
| **QUCAM3** | 0100- 0100 | 44 | 163 | 1080 | 1050 |
| | | | | | |
| **MITLL3** | 0110- 0011 | 63 | 160 | 2148 | 4200 |
| | | | | | |
| **MARCONI2** | 0111- 0010 | 72 | 168 | 2148 | 4700 |
| **RED+** | 0111- 0011 | 73 | 158 | 2148 | 4200 |
| **AUXCAM** | 0111- 0100 | 74 | 158 | 2148 | 4200 |
| **RED+2** | 0111- 0101 | 75 | 158 | 2148 | 4200 |
| | | | | | |
| **AG0** | 0101-0000 | 50 | 236 | 1072 | 977 |
| **AG1** | 0101-0001 | 51 | 243 | 560 | 536 |
| **AG2** | 0101-0010 | 52 | 248 | 1072 | 977 |
| **AG3** | 0101-0011 | 53 | 248 | 1072 | 977 |
| **AG4** | 0101-0100 | 54 | 248 | 1072 | 977 |
| **AG5** | 0101-0101 | 55 | 243 | 560 | 536 |
| **AG6** | 0101-0110 | 56 | 243 | 560 | 536 |
| **AG7** | 0101-0111 | 57 | Ambient | 560 | 536 |

# APPENDIX.2:
# MEMORY SPACES AND COMMANDS

## SCIENCE CAMERAS

| | | |
|---|---|---|
| **ID Code** | Y:0 | Read |
| **Current temperature** | Y:F9 | Read |
| **Demanded temperature** | X:F9 | Read/Write |
| **Heater status** | Y:003 | 000-On/FFF-Off |
| **Shutter status** | Y:FB | 0-Open/1-Close/2-Fault |
| **Demanded preflash** | X:FA | Read/Write |
| **Current preflash** | Y:FA | Read |
| **Preflash status** | Y:1 | 0-Off/1-On<br>Read/Write |

## AUTOGUIDER CAMERAS

| | | |
|---|---|---|
| **ID Code** | Y:0 | Read |
| **Sink temperature** | Y:F5 | Read |
| **CCD temperature** | Y:F1 | Read |
| **Peltier status** | Y:F9 | 0-Off/1-On |
| **Demanded preflash** | X:FA | Read/Write |
| **Current preflash** | Y:F2 | Read |
| **Shutter status** | Y:F3 | 2-Fault |

## COMMANDS

| | |
|---|---|
| **OSH** | Open shutter |
| **CSH** | Close shutter |
| **PFL** | Preflash |
| **IDL** | Start Idling mode |
| **STP** | Stop Idling mode |