A temperature controller board for the ARC controller

Simon Tulloch*
European Southern Observatory, Karl Schwarzschild Straβe 2, Garching, Munich 85748, Germany

ABSTRACT

A high-performance temperature controller board has been produced for the ARC Generation-3 CCD controller. It contains two 9W temperature servo loops and four temperature input channels and is fully programmable via the ARC API and OWL data acquisition program. PI-loop control is implemented in an on-board micro. Both diode and RTD sensors can be used. Control and telemetry data is sent via the ARC backplane although a USB-2 interface is also available. Further functionality includes hardware timers and high current drivers for external shutters and calibration LEDs, an LCD display, a parallel i/o port, a pressure sensor interface and an uncommitted analogue telemetry input.
Keywords: ARC, temperature controller, RTD

1. INTRODUCTION

Any scientific camera system will require a temperature controller to maintain a stable detector temperature. Many systems also require a pressure gauge controller to supply power and read telemetry from the gauge head. This board combines all of these functions. It occupies one slot of the ARC controller backplane and communicates with the Timing (main processor) board via its Serial Communications Interface. Power is also drawn from the backplane although for high heater powers an external source must be used since the backplane can only supply limited current. There are four temperature input channels that can each be configured for both Resistance Temperature Detectors (RTDs) such as the commonly used PT100 or for diode sensors. A pluggable EEPROM on the board comes preprogrammed with calibration curves for the Pt100, DT670, S900 and 1N4148 sensors. There is room in the EEPROM for additional non-standard sensor curves. Four-wire sensor connection topology is supported which reduces offset errors when using RTD sensors over long cable runs. There are two 9W servo loops that can be independently configured and run in parallel. Servo output current, voltage and power is continuously monitored using a sense resistor. This provides useful telemetry and also protection against cable faults and short circuits. The heart of the board is a 32MHz 16-bit microcontroller programmed with 32kB of compiled ´C´ code. This performs command interpretation and telemetry transmission and implements the two PI control loops. Loop parameters can be freely programmed by the user and tuned to the specific application. The main features of the board are shown schematically in Figure 2.

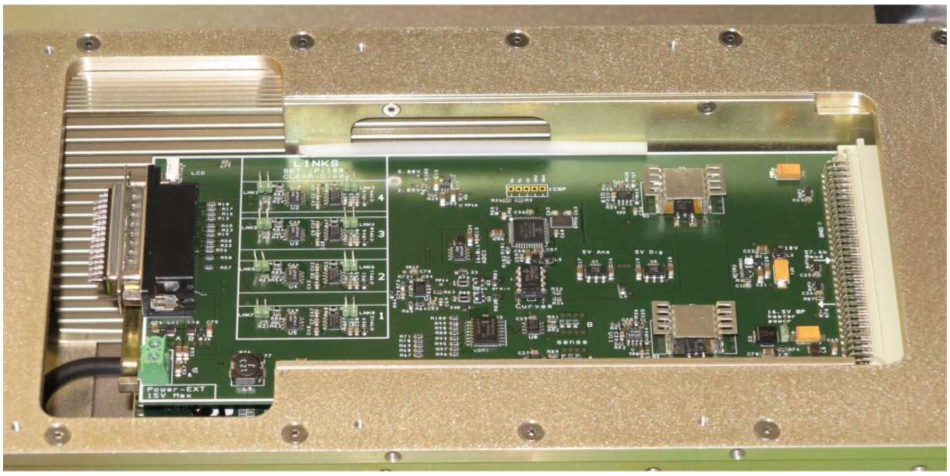


Figure 1. The temperature controller board mounted within the ARC controller.

*stulloch@eso.org; phone Tel. +49-89-32006170 Fax. +49-89-3200683

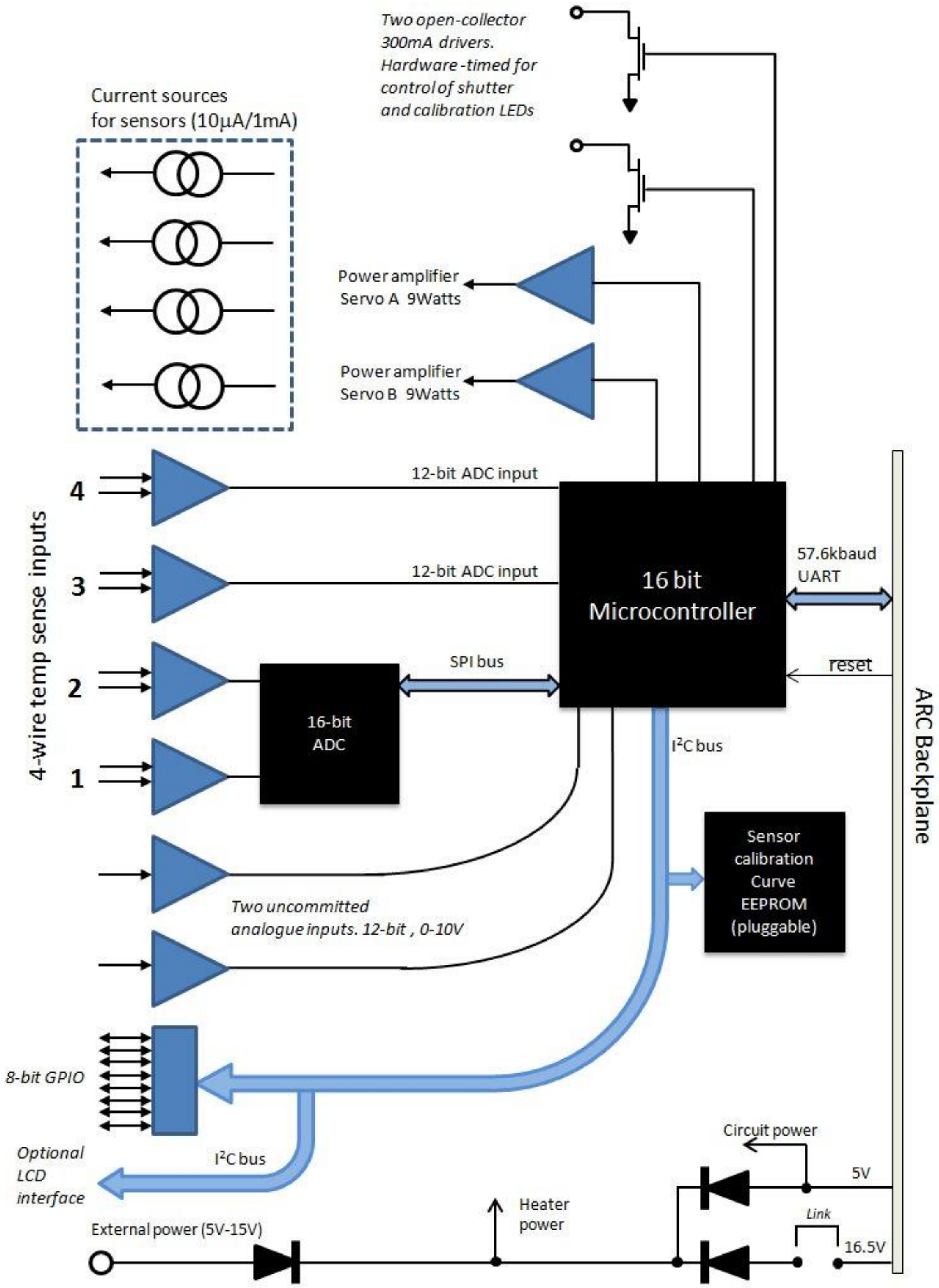


Figure 2. System schematic.

2. HARDWARE DESIGN

2.1 Controller board

This consists of a 4-layer PCB measuring 228 x 100mm. The layout is shown in Figure 3. Outer layers are ground planes to reduce the possibility of EMC issues with adjacent ARC boards. The processor is a PIC24FV microcontroller in a TQFP44 package. It communicates with the ARC backplane via a buffered two-wire serial interface. One further pin allows the ARC to perform a hardware reset of the PIC. Once up and running the board will function entirely independently of the rest of the ARC. Even if the analogue circuitry of the ARC is powered off, the temperature servos will continue to run. Power for the servo heaters is by default drawn from the 5V backplane bus. If more than 0.45W per channel is required it becomes necessary to set a link to draw power from the 16V backplane bus, however, this cannot be guaranteed since it is interrupted if the user performs a power-off of the ARC analogue circuitry. For higher power applications of up to 9W per channel, an external source of up to 15V can be attached to a screw-connector at the board edge. This input is LC filtered to avoid EMC issues.

The two servo loops are implemented using a Proportional-Integral algorithm. The output demands are calculated to 16-bit precision and presented at the PIC output pins as a 32kHz PWM signal. These are then converted to linear heater voltages via second-order 1.2Hz LP filters. To avoid noise coupling to the imaging sensor of the camera, low-ripple on the heater current is essential hence the use of such heavy filtering.

The PIC micro contains a 12-bit ADC with input mux that is used for the on-board servo-amp sensors, temperature input channels 3 and 4 and the two un-committed analogue input channels. The two servo loops, however, require higher resolution so an external LTC1865 16-bit converter is used to digitise temperature channels 1 and 2. This communicates with the PIC via an SPI interface. Other peripherals such as the INA219 power monitors, the pluggable EEPROM, the external LCD and the 8-bit GPIO port, interface via a lower speed I^2C bus.

For flexible operation it was important that the design could accept a wide variety of temperature sensor types. RTDs differ from diode sensors both in the energisation current they require and also in their output sensitivities. RTD sensors typically require a 1mA current, diodes 10uA. RTD sensitivities are <0.5mV/K, whereas diodes give 2-3mV/K. The input circuitry to the board must cope with these widely differing requirements. It was decided to use manual link settings to achieve this. Each input channel therefore has two links that must be set or cleared, one to change the input gain, the other to change the output energisation current. As can be seen from Figure 6 the silk-screen clearly shows the link functions so link setting is a straightforward process.

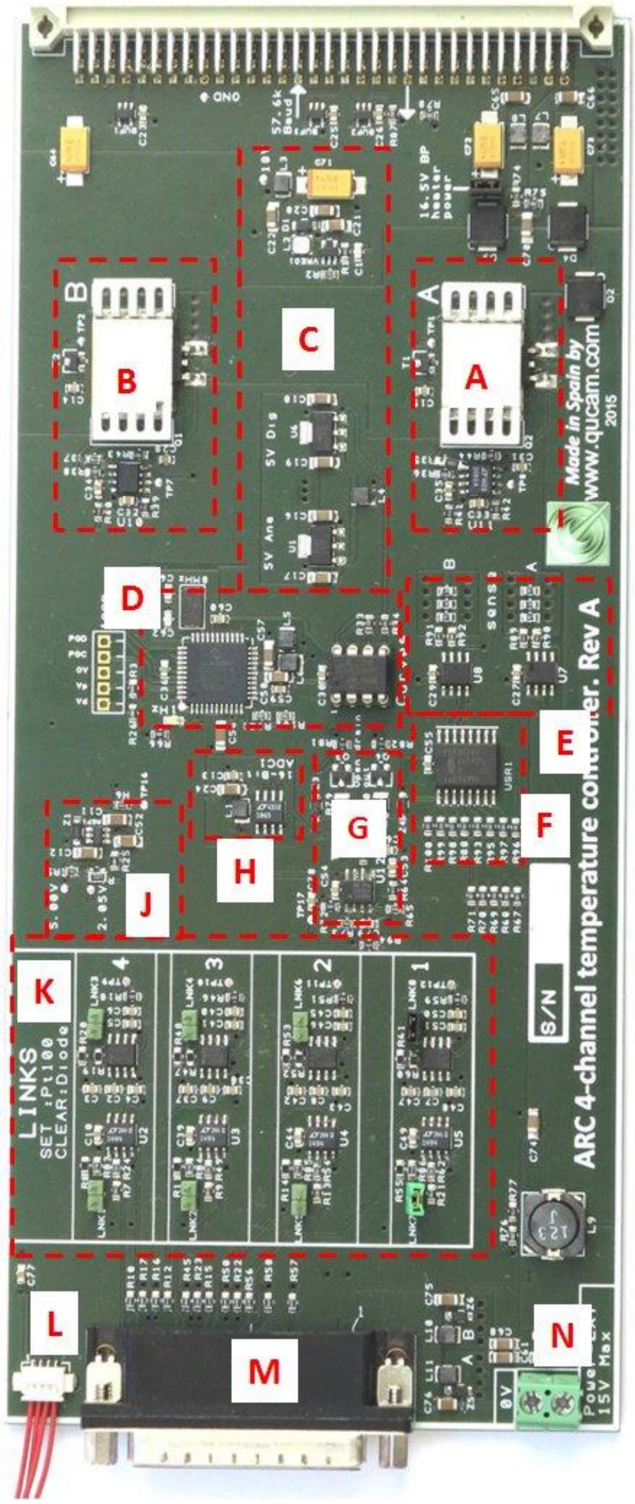
The servo amplifiers use surface mounted heatsinks each of which is instrumented with a temperature sensor. Servos will automatically power-off if they overheat. Further protection is given by continuous monitoring of the servo heater current to detect short-circuit conditions. Additionally, the user can program temperature alarm thresholds and hard maximums for each servo temperature. If the servo temperature exceeds the alarm then a pin on the 44-way edge connector is set high. This could be used, for example, to power an external indicator LED. If the maximum is exceeded then the servo is powered off automatically to protect the camera and a second indicator pin set high.

Two additional open-drain digital outputs are provided on the board. Each can sink up to 300mA. Flyback-diodes are included on-board so inductive loads can be driven safely. These outputs are intended to drive shutters and external calibration light sources. Their outputs can be configured as monostables (for exposure timing) using on-board hardware timers or simply set high or low by user command.

Stability of both the input preamps and the sensor-energisation current sources is ensured by the use of 0.1% 25ppm precision resistors and a 10ppm voltage reference. Initial component tolerances are further calibrated-out for each board with the calibration gain and offset constants stored in the internal program EEPROM of each PIC.

2.2 Diode temperature sensor

The board comes with two temperature sensors included. These consist of small PCB tiles mounting a 1N4148 small-signal diode. They are manufactured on a large "snap-off" panel. Each tile has a 3mm mounting hole and two lead pads.



A: Power servo A

B: Power Servo B

C: Power regulation

D: Microcontroller and external EEPROM containing sensor calibration curves.

E: Current, voltage and power sensing

F: 8-bit GPIO

G: 2-channel analogue input with 12-bit ADC conversion.

H: 16-bit ADC

J: 10ppm voltage reference

K: 4-channel temperature sensor preamps and current sources

L: External LCD interface

M: 44-pin sensor/heater/digital-i/o connector

N: External power input connector for high-power operation.

Figure 3. The controller board indicating key functions. All components are on the upper surface of this 4-layer PCB.

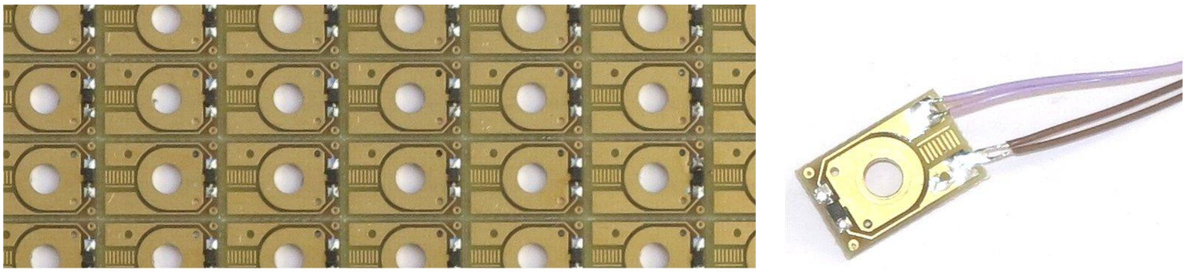


Figure 4. The 1N4148 diodes temperature sensors are manufactured on a panel. A single sensor with leads attached is shown on the right.

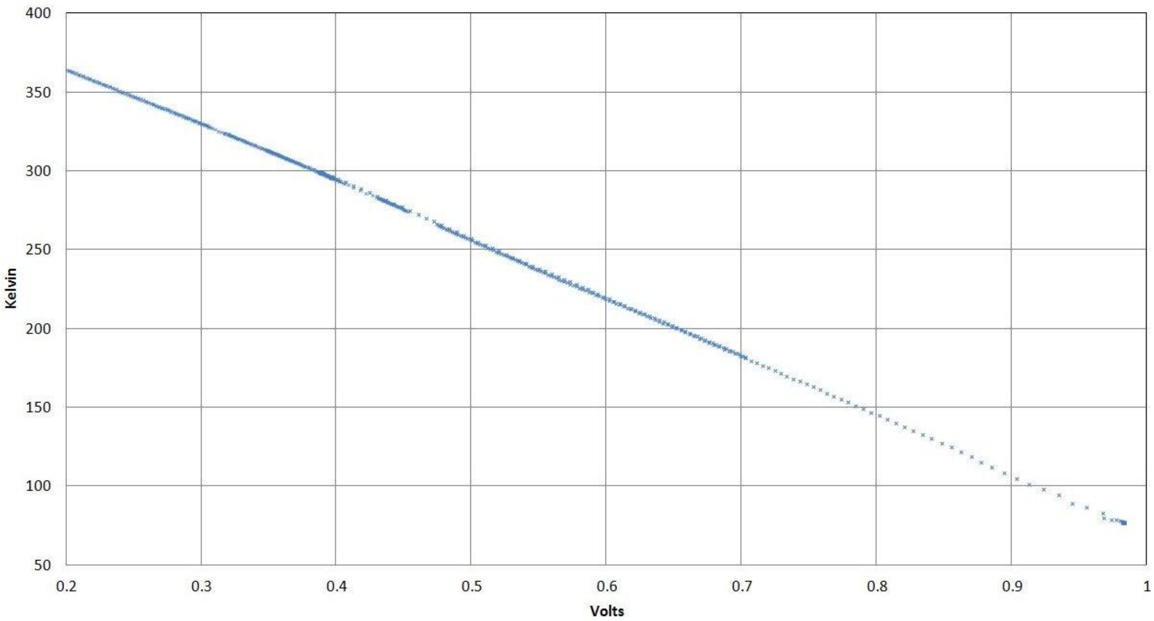


Figure 5. The temperature response of the 1N4148 diode. The graph shows band-gap voltage versus temperature for a 10µA energisation current.

Although not intended as a temperature sensor, the 1N4148 was found to perform very well, giving a fairly linear response right down to LN2 temperatures and a small spread in response between devices from the same batch.

3. PROGRAMMING THE CONTROLLER

There is a vocabulary of 51 commands making controller operation highly flexible. Communication with the ARC timing board uses the standard ARC syntax consisting of 24-bit words. The first word of each command or response is a header encoding the source, destination and length of the message. The next word encodes a 3-character ASCII command mnemonic and is followed by up to 6 binary parameters. Conforming to this syntax means that the temperature board can make use of the ARC SDK library and can also be used through the OWL data acquisition program. The board can also be used in stand-alone mode, with a USB interface circuit taking the place of the ARC backplane. The USB option offers a more user-friendly pure-ASCII interface. The board automatically detects which interface is present and switches the command interpretation routines accordingly.

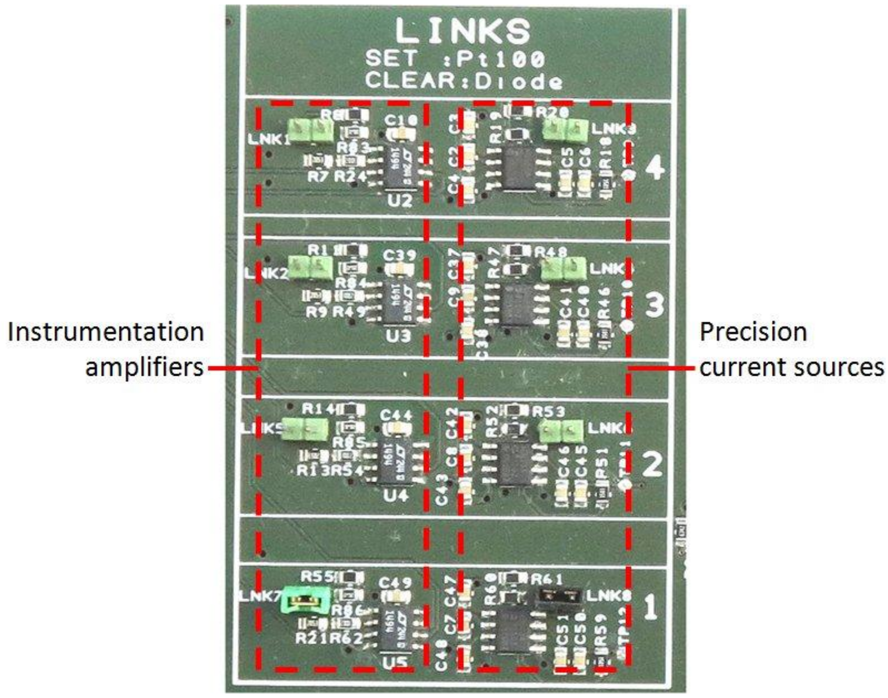


Figure 6. Close of up of the sensor preamps and current sources showing the link configurations. Here, channel 1 is configured for Pt100 (RTD) sensors, the other three channels for diode sensors.

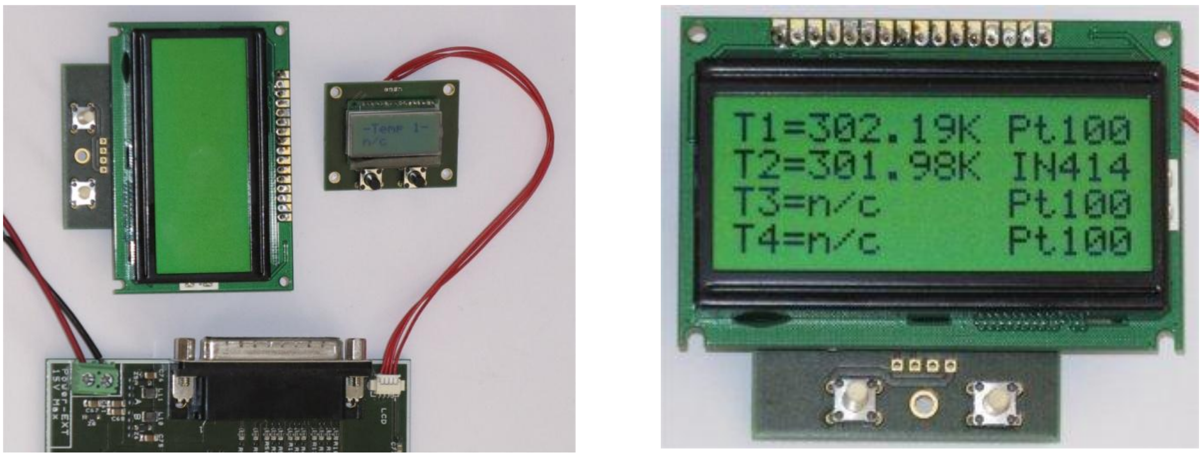


Figure 7. Two sizes of LCD can be attached to the board. Push buttons control the backlight and select up to 16 info pages.

3.1 Programming with the ARC SDK

By way of example, a typical line of user ´C´ code to communicate with the temperature board is shown below:

result= CameraAPI.Command(3, SET, TAR,2,180000);

Here, the first parameter ´3´ is the address of the temperature controller board on the ARC bus (this is invariant and cannot be changed by the user). Note that this is also the address of the ARC-50 utility board meaning that both boards cannot be present in the same ARC controller. The second parameter is the command mnemonic , in this case ´SET´ meaning that a parameter is about to be written, followed by ´TAR´ meaning the target temperature of a servo channel. The next parameters ´2´ and ´180000´specify that it is servo channel 2 that must be set to a target temperature of 180K. Since the ARC interface deals entirely in 24-bit binary numbers it is necessary to express all temperatures in units of mK and all voltages in mV. Two other examples follow.

result= CameraAPI.Command(3, KEL, 2); #Read back temperature of sensor 2
result= CameraAPI.Command(3, HPO, 1); #Read back heater power in servo loop 1

The user also has the option to program the controller board using the "Debug" sub-menu that forms part of the ARC-supplied OWL data acquisition program. This is shown below in Figure 8. Note that when using this GUI that the temperature board shares an address with the utility board so the ´UTL´ radio button must be set.

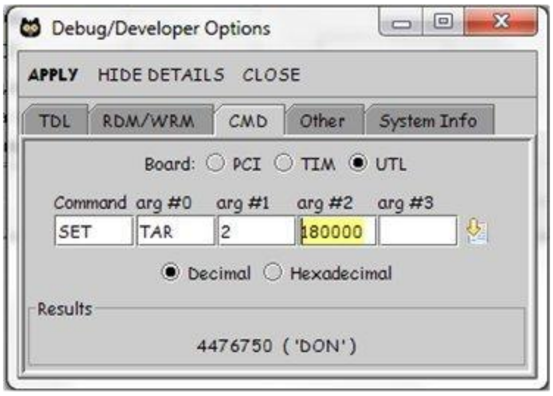


Figure 8.The OWL GUI can be used to communicate with the temperature board. In this example servo loop 2 has been set to target temperature 180K.

One of the two un-committed analogue input channels of the board can be used to connect to a Pfeiffer PKR251 or an MKS 972 pressure gauge head. The gauge voltages are converted into units of mBar. The dynamic range of these heads is astronomical and exceeds the 24-bit integer limit of the ARC interface. It was therefore necessary to encode the pressure telemetry as a 16-bit integer. Figure 9. shows the format together with a couple of examples.

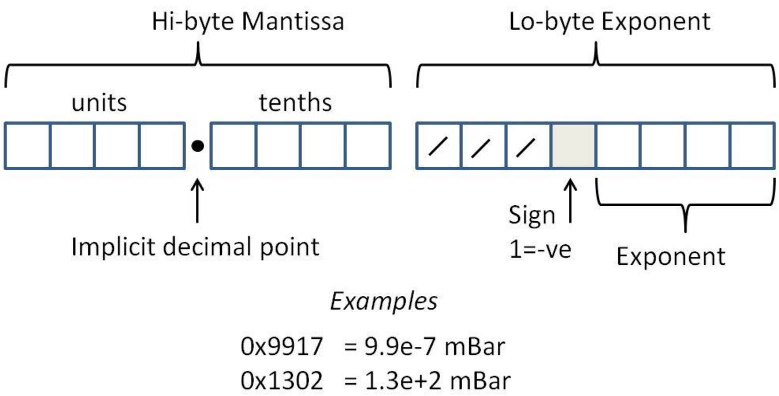


Figure 9. Pressure telemetry is encoded as a fixed-point float using the scheme shown here.

3.2 USB-2 Interface

This was initially intended for use during the product development phase when an ARC controller was not always available. The interface contains an FTDI chip so the controller appears as a Virtual-Coms port on the PC. Any terminal program such as "Realterm" can then be used for communication at 56.8kBaud. Alternatively commands can be sent from Python programs using the PySerial package. The command interface is entirely textual which greatly simplifies the passing of float parameters meaning that temperatures and voltages can be expressed directly in units of Kelvin and Volts. The interface was later developed to include a full Python-based GUI (see Figure 10). The board can be fully powered from the USB port although the servo heaters are then limited to only 450mW per channel.

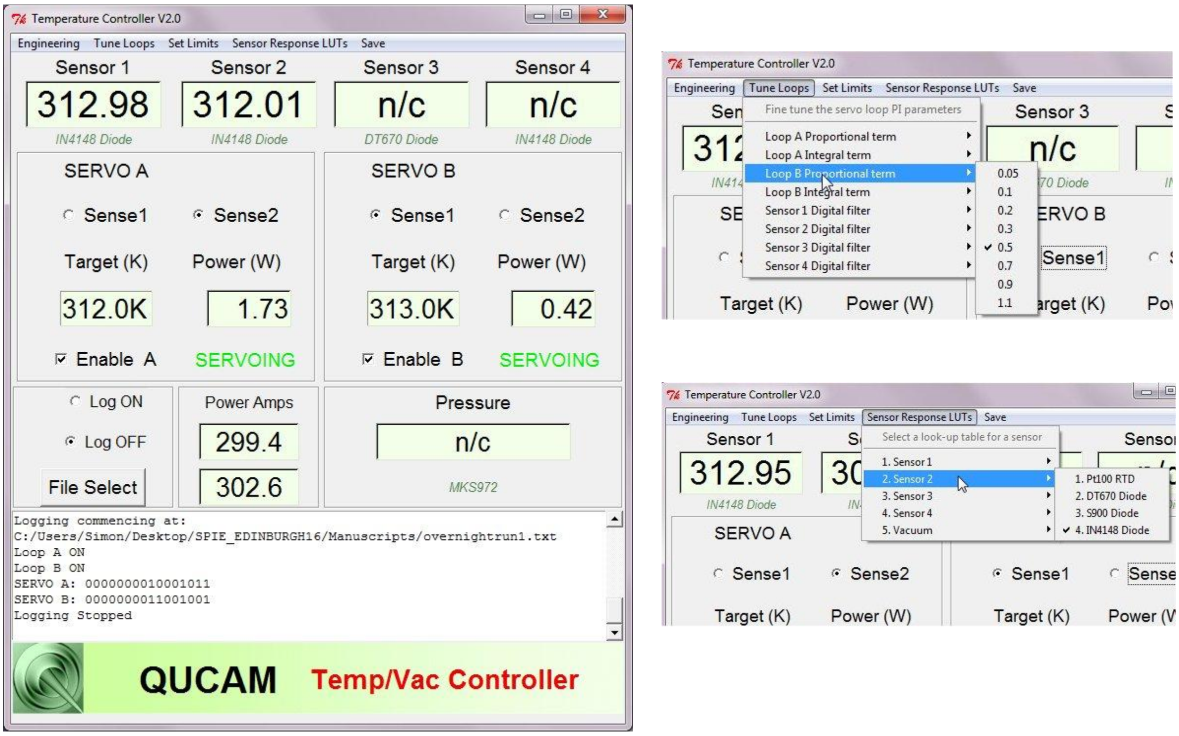


Figure 10. The Python GUI that can be used with the temperature controller when used independently of the ARC controller. In this configuration a custom USB-2 interface board is plugged into the board´s backplane connector.

3.3 Programming examples with the USB interface

The code example below shows a typical sequence of ASCII commands to setup a servo loop over the USB interface. If the same commands are to be sent over the ARC controller interface it is necessary to use units of mK for the temperature parameter and multiply P and I loop parameters by 1000. The in-line comments should be self explanatory.

SET TAR 1 180.2 # set the target temperature of servo loop 1 to 180.2Kelvin

SET SEN 1 2 # use temperature input channel 2 for servo loop 1

SET MAP 2 1 # Map sensor 2 to the calibration curve 1 (by default this is an 1N4148)

SET PRO 1 1.2 # Set proportional term for loop 1 to 1.2

SET INT 1 0.08 # Set integral term for loop 1 to 0.08

ENA 1 # Enable servo

3.4 Additional features

Full use was made of the available 32kbyte program space. This permitted the addition of some extra features. Single pole low-pass digital filters were implemented on each of the temperature input channels with bandwidths programmable to 0.03Hz, 0.1 and 0.3Hz. When disabled the bandwidth defaulted to the preamplifier hardware bandwidth of 30Hz. The filters could be useful to suppress sensor noise in noisy environments, some experimentation will be necessary in each specific application. When setting up a new system there is a useful command to read back the front-end sensor noise voltages. This returns the RMS voltage noise in units of μV. One potential problem with PI loops is known as "integral term wind-up". This can cause the integral term to saturate if a camera system is slow to cool down to operating temperature. Once close to the target temperature the integral term can then take a long term to settle down. The controller code avoids this by only switching on the integral term when the temperature is close to target. The width of the error window within which the integral term becomes active can be programmed by the user. Configuring the controller can take a sequence of many commands. To make things easier the "SAV" command stores the current configuration in an EEPROM block in the PIC processor. This configuration is reloaded when the board is powered up. The board always powers on with the servos disabled but apart from this it fully remembers its past configuration at the time of the SAV command.

4. CALIBRATION AND TESTING

Each of the analogue input channels was calibrated for initial errors in gain and offset using a special calibration board.

4.1 Gain and offset trimming

The calibration board is shown in Figure 11. It contains a precision reference, a resistor chain and a high stability op-amp buffer. By selecting links, calibration voltages can be applied across the full dynamic range of the temperature controller. The input voltage at each link setting is measured using a $5^1/_2$ digit 0.015% Rigol DVM. The corresponding digital number produced by the temperature controller ADC at each link setting is then plotted against voltage. A linear least-squares fit through the data provides a calibration slope and offset for each analogue channel that is then burned into the on-chip EEPROM of the microcontroller. The precision current sources used to energise the temperature sensors were also measured using the Rigol and the current values stored in the EEPROM. This first order correction was sufficient to keep input referenced errors below +/- 5μV for temperature channels 1 and 2 (16-bit ADC) and below +/-10μV for channels 2 and 3 (12-bit ADC). In this test the amplifier links were configured for use with a Pt100 sensor so these systematic errors corresponded to approximately +/-15mK and +/-30mK respectively.

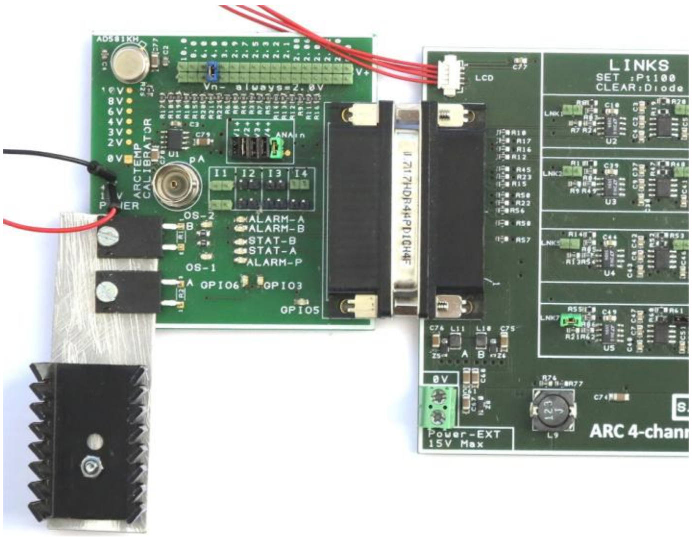


Figure 11. The external calibrator used to remove component tolerance errors in each temperature board. Calibration coefficients are then burnt into the on-chip microcontroller EEPROM.

4.2 Servo stability

The servo loops were extensively tested using a heat-sink block with a temperature sensor and heater resistor attached since no cryogenic camera was available during the early stages of development. One of the first tests was the overnight stability of the servo loop. One such run is shown in Figure 12. This proved the long term stability and also allowed the temperature noise to be measured. With the diode sensor in use the noise was found to be 13mK RMS. This is close to the size of a digital interval implying that the system was limited by the ADC resolution rather than the preamplifier noise.

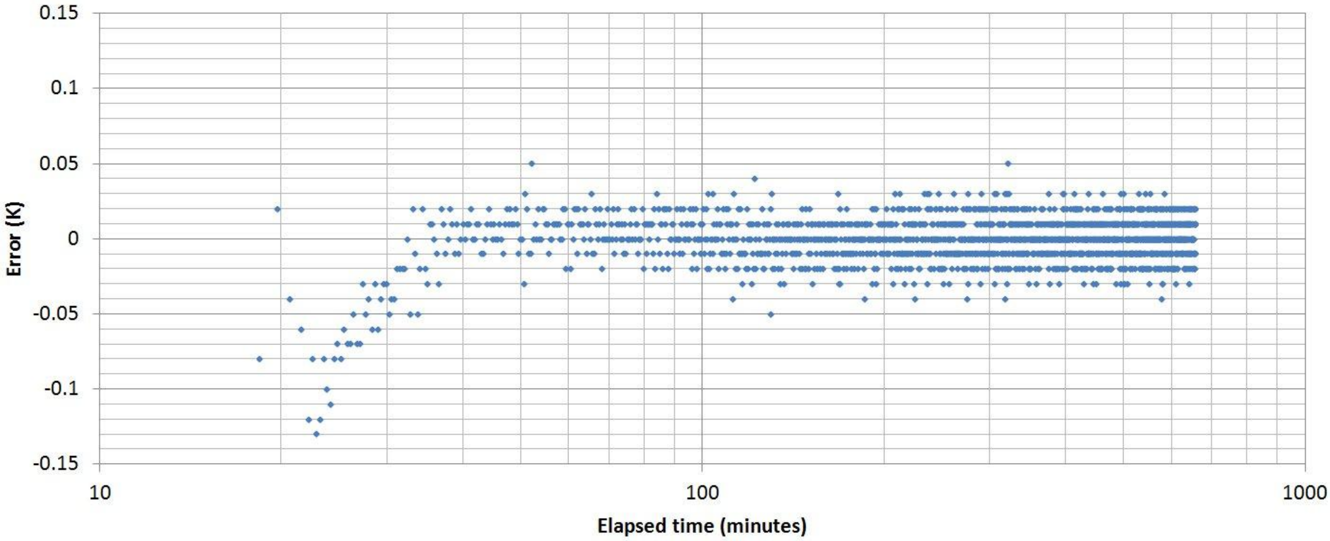


Figure 12. An overnight stability test. Once at temperature, the servo noise was 13mK RMS. A diode sensor was used.

4.3 Transient response

Another important test involved switching the demand temperature between two alternating values to investigate the transient response for various settings of the Proportional and Integral loop parameters. One such test is shown in Figure 13. It is clear here that the Integral term must be kept below 0.1 to avoid overshoot and that with the Proportional term below 1.0 the response becomes very sluggish. The user can freely tune the parameters for each loop depending on their specific application.

4.4 Slope control

Scientific imagers can be damaged by rapid temperature changes so the controller contains a slope control algorithm. This gradually ramps the servo target between the current temperature and the new demanded temperature. Its behaviour is shown in Figure 14.

5. CONCLUSION

The board worked as intended. Most of the development was done using a heat-sink block but a final test was performed with a cryogenic CCD camera to ensure that there was no noise degradation of the images. The board offers an economical replacement option for commercial temperature controllers most of which are highly over-specified for imaging applications. The same is true for the 1N4148 diode sensors which have been proven as a low-cost replacement for commercial sensors costing several hundred Euros. Combining the functions of a temperature, shutter, vacuum controller with the ARC chassis offers additional benefits in reducing system and cabling complexity in scientific imaging systems. Further information can be found at http://www.qucam.com/.

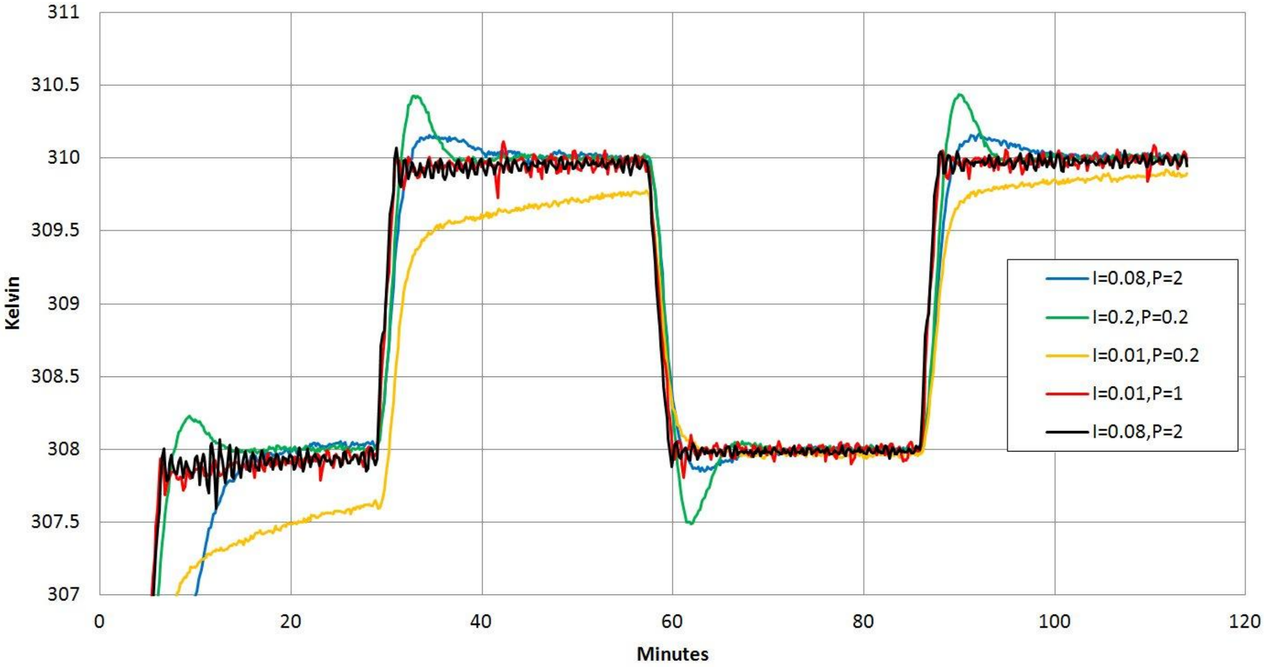


Figure 23 . Transient response of the controller with a wide range of PI loop parameters settings.

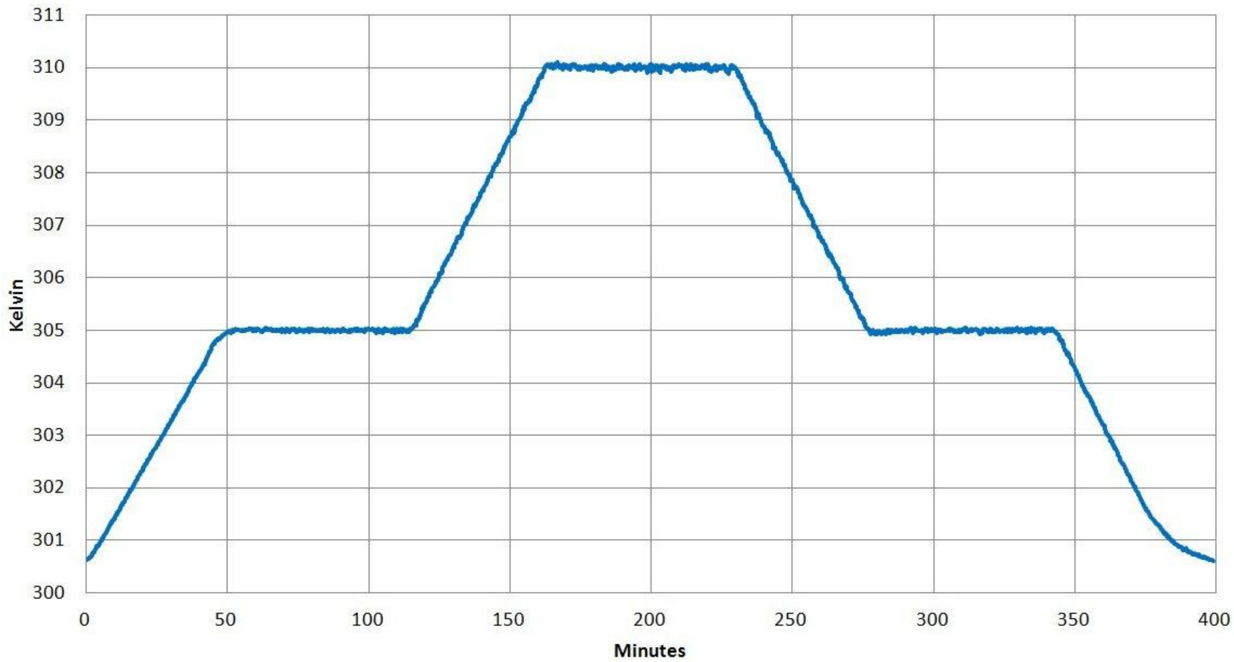


Figure 14. Test of the slope-control algorithm. A maximum slope of 0.1K per minute had been programmed.