Mercury Firmware Release Notes

Version 2.2.6.20 • 18 March 2016

OXFORD INSTRUMENTS

The Business of Science®

Mercury Firmware Version 2.2.6.20

No application code changes. Added magnet firmware 2.4.

General bug fixes and features added

Fixed bug introduced with 2.2.6.19 magnet firmware 2.3 which caused internal magnet board SPI bus errors.

This release includes the following firmware images:

magnet: v2.4 *** NEW ***

heater: v2.3

temp: v2.2

aux: v3.1

level: v2.2

tempPressure: v2.2

pressure   v1.10

oidbDrivers   v3.0.7

Mercury Firmware Version 2.2.6.19

No application code changes. Added magnet firmware 2.3.

General bug fixes and features added

This magnet firmware version introduces a faster (x5) ADC read loop. This faster read enables reliable operation of the "Catch" routine so this can be enabled as a magnet control feature.

There is also a fix for a calculation rounding error which gave rise to errors in very slow magnet sweeps.

Added iPS functions

1. Catch routine can now be enabled.

This release includes the following firmware images:

magnet: v2.3 *** NEW ***

heater: v2.3

temp: v2.2

aux: v3.1

level: v2.2

tempPressure: v2.2

pressure   v1.10

oidbDrivers   v3.0.7

March 2016   Version 2.1.6

Mercury Firmware Version 2.2.6.18

General bug fixes and features added

Corrected a regression introduced in version 2.2.6.14. For example the output of a heater would be killed after a second or so, because cryosys was not writing to the "reset_trips" file, causing the card's watchdog to bite.

This release includes the following firmware images:

magnet: v2.2

heater: v2.3

temp: v2.2

aux: v3.1

level: v2.2

tempPressure: v2.2

pressure   v1.10

oidbDrivers   v3.0.7

Mercury Firmware Version 2.2.6.17

This release includes updates for the application firmware (Cryosys) and drivers (OIDB) running on the Mercury target platform. It also includes an updated set of device firmware for the Mercury expansion cards. The update can be applied to Mercury iTC units as well as Mercury iPS-M and Mercury iPS-S units. The upate can automatically update all the devices in the system (by choosing the *Update -> Application* option) or it can be used to update a targeted device by selecting that device from the drop-down list control. Unless there are special reasons for updating one device only, it is recommended that the *Application* option is selected to update the whole system. The update process will re-calibrate each device after update and will keep an update log on the system. The complete update will take about 5 minutes depending on how many devices are installed.

Added iPS functions

1. "Current epsilon" configurable attribute. Ability to configure how close the Magnet current must be to a setpoint or a limit to be considered being equal. New remote command to support this was added "DEV:GRP?:PSU:CEPS" New column "Epsilon" on Slave configuration screen added.

2. Modified "Rate limit table" lookup. When looking up the target current in the ramp rate limits table the value will include the limit itself (both upper and lower). In other words target current will be compared to the ranges defined in the table using "less than or equal" and "greater than or equal" operators (as opposed to using "less than" and "greater than").

3. Added "Monitor" output to PSU group. New functionality to assert a selectable output on auxiliary card whenever the magnet current is not zero. For selecting an output to use, there is a new column on Slave configuration screen but this is currently hidden.

4. Restored error flags for Magnet's board voltage and current ADCs. Magnet boards (PSUs) will now report in the error log the fact that voltage or current ADC was reset after the communication with it was lost, (after the ADC failed to produce a valid reading in a predetermined number of attempts). This functionality requires magnet firmware 2.1 or later and oidbDriver/oidbMsg version 3.0.8 to work. Also undefined two error flags which have no meaning in Magnet firmware.

5. Cryosys now includes the Magnet board's related definitions (Modbus registers) from the proper header file, i.e the header used by the Magnet's board firmware project (mpsRegs.h).

6.   Added support for legacy commands R17 (TripCurrent) and R23 (LeadResistance).   Fixed support for legacy command R19 (TripField).

7.   Remote command for Magnet PSU state will now recognise "Ramp to Field" state ("RTOF").

8.   Added buffering to "Catch" function to improve reliability of the magnet catch trigger which, previously, could be falsely tripped. Also added "Catch" enable/disable Boolean to the magnet configuration table.

March 2016   Version 2.1.6

General bug fixes and features added

1. GPIB interface. A complete re-work of the GPIB interface within the cryosys application and and update of the GPIB library to version 3.2.21 in the underlying file system.

2. Temperature sensor "voltage" signal (thermocouple mode) is now allowed to take negative values.

3. Added support for P-only control loops.

4. Fix for the screen locking function when operated remotely in "Legacy" mode.

5. Added a template "Rotator" file for driving rotating sample stages.   This adds old-style SMC1 functionality to the Mercury function suite.

6. iPS: Fixed quench status memory problem

7. Added ability to change LCD brightness from remote GUI.

8. On embedded target Modbus error messages will now show verbose reason for failure in plain English instead of a numerical code.

9. Cryojet: Fix for the start up sequence which could cause "fastCool" to be set to "ON".

10. Cryojet: Heat and Flow set point on the "Control Loop Configuration" screen will update their value and button state correctly at the end of sweep.

11. Heliox: Small change to the HelioxX template to prevent the control loop from getting stuck at the end of the re-condensation phase (where the template would sometimes fail to switch off the sorb heater). Also consolidated some other small fixes to this and other templates.

March 2016   Version 2.1.6

12. On all widgets showing values in "d" format the value will be now shown with at least six significant digits. This includes the requested number of digits after the decimal point (but the digits after the decimal point will be "dropped" as the value gets bigger). Also the new code will correctly render negative numbers with same number of significant digits.

13. SPI-based devices will from now on correctly resume communication after performing a "Application" firmware upgrade.

14. SPI-based devices will from now on correctly resume communication after performing a "Single device" firmware upgrade, but user will be prompted to power-cycle the unit. There should be a delay of at least 20 seconds after power down before powering back up.

15. Before a firmware upgrade the Arm version saves a snapshot of the device information table. Includes an Id, Serial number, Revision and Firmware version for all detected devices. The snapshot can be used to restore the serial number and revision if a firmware update causes the device serial number to be lost. The information is appended to a file so it serves also as a history record. This version will also save the device information during the upgrade to this version (after the unit was rebooted and is already running this version of cryosys but before it actually started upgrading firmware on all eligible devices). In other words we preserve the device information also during the upgrade to this version.

This Cryosys 2.2.6.17 release includes the following firmware images:

Device	Firmware version included
Magnet board	2.2
Heater card	2.3
Temperature sensor card	2.95
Auxiliary (Gas [control]) card	3.1
[Cryogen] Level meter card	2.2
Temp/Pressure card (obsolete)	2.2
Pressure sensor card	1.10
oidbDrivers	3.0.7

Mercury Firmware Version 2.1.6 – New Features

Improved System Response

The overall performance of the firmware has been improved to provide faster processing. The result is a better user interface response both on the screen and the remote interfaces. Acquisition speed may be increased to higher sample rates on customised systems.

The Graphic interface has been normalised and improved for better touch response and consistency. It has also been rebranded for better appearance and improved user experience. If users wish to retain the older layout, they can do so by setting the black background on the display tab of the settings screen.

Switch heater control (iPS)

Figure 1 Switch heater times configuration

On and off times are now programmable for the switch heater. If the user tries to start a ramp before the switch heater is ready, the unit will display a warning to indicate this condition. Times are set in milliseconds, and counted from the moment the switch changes. These should be configured to slightly larger values than the longest time it can take the switch to turn on/off in order to avoid failures in extreme cases.

Switch heater failure reports (iPS)

The GUI of the new iPS software includes functionality to report whether the intended switch heater current has been reached. When the switch heater has been activated by the system but the physical element has not reached its target (current or temperature), the screen will report this by indicating

thet the heater is activating as opposed to on. This will also be indicated by a darker colour in the switch heater button as seen in Figure 2
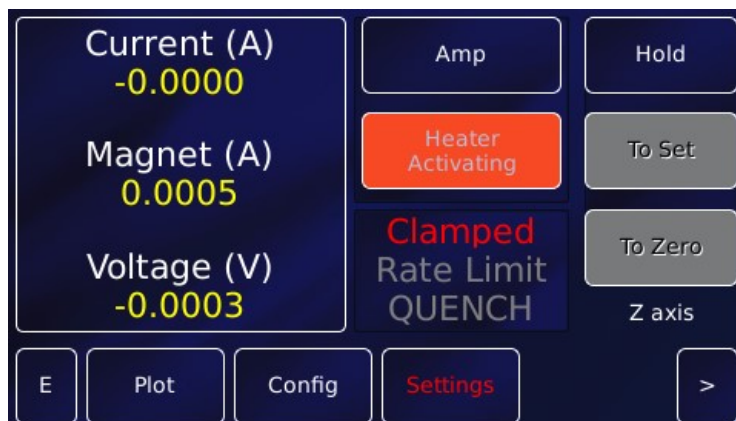

Figure 2. Switch Heater activating

Better Voltage Tracking (iPS)

Voltage tracking from slave units is now faster and more accurate, reducing the risk of individual PSU transients causing false quenches. The new voltage tracking allows high inductance/high power magnets to be managed more efficiently and improves current stability in Series and Matrix modes.

In addition to the Voltage tracking, a Clamping synchronisation mechanism has been added to the Matrix mode to ensure stability.

Pressure controlled Gasflow

Version 2.1 allows for flow-only PID loops, enabling the control of a needle valve directly from a pressure sensor without the existence of a heater. In this configuration, the flow is expected to increase the pressure when it increases and therefore the PID works in the same fashion as a heater PID would.

PID Tables and sweep functions work in the same manner for flow-only loops as they do for heater loops. Do note that the P band will be expected in Bar, meaning that a pressure band of 5mB will have a P value of .005

Parameter configuration storing

The system will now remember the last user configured value for parameters such as temperature set points, PSU ramp rates, and magnet persistent currents when turned off. Interlocks, alongside safety conditions on startup will prevent these parameters from creating unwanted behaviour during startup.

March 2016   Version 2.1.6

Temperature Interlock Recovery

Temperature interlocks have been improved to recover quickly after an out of range event. The system will restore heater and flow outputs more promptly even when in manual mode. Additional filtering has been added to avoid false alarms triggering the interlocks erroneously.

Improved Temperature Stability

False trip resets caused by stalling in the internal communication bus, decreased the stability of temperature control in iTC version 1.4.40 firmware. This problem has been addressed along with an improved response time of the bus, enabling the system more precise control and therefore higher stability.

Improved Cryojet Control Functionality

In addition to the improved basic features, there have also been some improvements to system specific features. In the case of the Cryojet control, new safety features have been added to ensure that the shield flow is always restored or set even after an interlock or system failure has occurred. The new algorithm will always set a shield flow when required and report on conditions that do not allow this to happen. Additionally, new flow tables and a higher shield temperature set point reduce the possibility of condensation in more turbulent or humid environments.

Mercury Firmware Version 1.4.43 – New Features

Voltage Damping

When driving an inductive load, the power supply (IPS) output control loop has to use the time constant of the IPS plus load circuit to predict the response time of the magnet in order to prevent excessive overshoots. In the previous version of the firmware there was an error in the damping causing the control loop to drop out of ramp mode too early. This resulted in the fine trim routine executing the last part of the ramp causing very long settling times to be observed. The new version takes into account the inductance value configured by the user to profile the end of ramps, thus obtaining a much more accurate end-of-ramp decay curve. The result is that settling times are in the order of a few of seconds when before they could be as long as 2 minutes depending on the ramp rate and the actual inductance. The correctly applied damping voltage will also help smooth out transient voltages, thus providing smoother ramps and a more stable output.

When the IPS is approaching the end of a ramp (approximately 1s before the target current reaches the end of the ramp), the ramp rate will be decayed exponentially to match the equivalent LCR circuit response. Once this decay reaches within 300µA of the target, the system will settle and fine trim to the final value.

In order for voltage damping to work correctly, the system must know the value of the magnet's inductance. This must be set in the magnet configuration screen (see Mercury IPS manual Section 4.3.9). The default value (999H) will result in an over-damped approach similar to the one on previous versions of the firmware.

Catch routines

When the system is in persistent mode, if a switch failure occurs (switch breaks open) causing the IPS leads to be connected directly to the magnet with a significant difference in the IPS output current and the magnet current. If this difference is not handled, the magnet will quench. The new version of the firmware has an intelligent detection algorithm that will adjust the IPS output current very quickly to try to match the magnet current to "catch" the magnet and prevent a quench.

The catch routine will first match the magnet current and then turn the switch heater on to hold the magnet at the stable current. If this cannot be done before the transient time of the system has elapsed, it will report a quench. Although the IPS will try to catch the magnet as close to its present current as possible, the resulting current may be any value between that of the magnet and that of the IPS output before the switch broke open.

In addition to catching the current, the routine will also balance the current of all IPS slaves. During the catch, when the output of each unit was changing quickly, the units may have caught at different currents so that each slave in the group is not making an equal contribution to the total current. So, although the initial catch takes only a couple of seconds, the system may require up to a few minutes to balance the currents and be fully ready after a catch. During this period, the unit will report to be

ramping (To Set mode) although the total current is not changing. At the end of the balance sequence the system will go back to the HOLD mode.

To check if the catch routine has been activated check the "Alarms" log. To do this, from the home screen tap the *Settings* button, then the *Alarms* button, turn on the alarm log history by tapping the *History* button (orange is ON, black is OFF). If the catch routine has run a CATCH alarm entry will appear in the log with a system date and time that it ran.

Notice

Setting a ramp before the catch is settled may result in incorrect behaviour. In addition, the magnet may have warmed up during the catch routine. Please allow a few minutes for the system to settle before ramping the magnet once again.

Intelligent switch heater control

The new version of the IPS contains a proprietary intelligent switch control. Currently, this functionality is only available to OI Engineers. There are a few new parameters on the magnet configuration table used to configure this feature (*Htr Res., Switch Res., Lead Res.*). If you do not have an OI-supplied intelligent switch control, leave these fields as zeros. If your system has been factory configured with intelligent switch control then these values will be non-zero.

Figure 3. Intelligent Switch Parameters

Mercury Firmware Version 1.4.40 – New Features

Precise Needle Valve Control

Improved temperature stability can be obtained if the flow of cooling fluid is also controlled by the iTC. An optional auxiliary I/O board contains drive circuitry for a stepper motor which can drive the opening and closing of a needle-valve to control the flow of a suitable fluid. This function is referred to as gas-control in the GUI.

Controlling the gas-flow regulates the cooling power, which provides control of the sample temperature. When the gas control is set to AUTOMATIC mode, the position of the needle valve is automatically controlled to provide variable cooling power. In conjunction with the heater, this enables accurate temperature control at cryogenic temperatures.

There are two possible configurations. The flow dynamics of these two arrangements are different.

- Automatic needle valve: the needle-valve is fitted to the main cryostat and controls the flow between the cryogen reservoir and a variable-temperature insert.

- Automatic gas flow siphon: the needle valve is fitted in an automated gas-flow siphon, linking the cryostat to a separate cryogen storage-vessel.

The position of the needle valve was previously controlled by a control function that operates relative to the present valve position rather than to an absolute position. The output of the control function was related to the valve position by a Valve Gearing parameter that defined how many steps of the stepper motor exist between the valve fully-open and the valve fully-closed positions.

The new release of Mercury allows real-time tracking of the needle valve position. Rather than update its state at the end of an open/close action, it does so as the valve moves along. This results in faster response time, more accuracy, and removes synchronization issues since the reading is absolute (whereas with the previous relative reading the application could lose track of the motor).

View of Current Alarms vs Historic Alarms

The Alarm Logs page displays a table of alarms that have been generated on the system. The table contains an entry when an alarm is first raised, and another entry when the alarm is restored. This table has the following columns:

- **Date/Time** The date and time when the alarm was raised.

- **Device UID** The identifier of the device that raised the alarm.

- **Alarm** A brief description of the alarm (e.g. Short-circuit).

This is a useful tool if a fault occurs. For example the Alarm Logs record if a particular temperature sensor is open-circuit or short-circuit.

To access the Alarm Logs:

- Tap the *Settings* button on the Home page.

- Tap the *Alarm* button on the *General Settings* tab.

- Scroll through the list to display more alarms.

The new release can display a list of currently active alarms, eliminating the need to sift through the Alarm Logs to find if a particular alarm is active. Alarms appear on both the current and historic Alarm Logs as they are generated, and they disappear from the current Alarm Logs when they are restored (creating a restore entry in the historic Alarm Logs). In order to switch between the current and historic Alarm Logs, click the History button in the Alarms screen (orange means you are looking at the historic Alarm Logs, whereas black refers to the current Alarm Logs).

In addition to current alarms, the new version also registers safety interlocks which were previously not reported as alarms.

Figure 4. Current (left) and historic (right) Alarm Logs.

Two Point Calibration Correction

The iTC uses tables to store calibration data for different sensor types. Calibration tables are saved as files in the Mercury file system. A calibration file contains a number of discrete data pairs, each mapping a sensor reading to a temperature value. Linear interpolation between adjacent pairs of points is used to provide temperature values from intermediate sensor readings. The iTC calculates setpoint limits and sensor limits from the chosen calibration file.

On many of our systems we use sensors that conform to generic calibration curves based on the ITS-90 international temperature scale. Due to sensor manufacturing variability there can be small deviations from the generic curve. If two known temperatures are available it is possible to make small adjustments to the scale and offset of the standard curve to match a particular sensor. The process can be used with either PTC, NTC, Diode or Thermocouple sensors. The adjustment is applied to the measured parameter, so for a resistance sensor the adjustment is applied to the measured resistance.

For best results the scaling adjustment should be applied at the high-parameter end and the offset at the low-parameter end. So, for the example for an NTC resistance sensor, apply the scaling at the known low-temperature point, ideally near end of the range. If a scale adjustment of more than a few percent is required to correct the temperature reading, investigate the measurement set-up to check there are not some other errors present.

Once the scaling adjustment has been made, change to the low-resistance end of the range and make the offset adjustment (for an NTC resistance sensor, this is a high-temperature point). Again this should be a small adjustment. As these adjustments are not independent, re-check the high-resistance point after the offset has been adjusted. A few iterations of these adjustments may be required to achieve the optimum accuracy.

In previous firmware versions, users were required to manually set the scale and offset for the curve, the new firmware can calculate these values by entering the true temperature at two points on the curve. Once true temperature points for the low and high ends of the range have been entered, tap the *Calculate* button (see Figure 3) for the system to do the calculations. Once this is done, save the correction parameters which will be applied to readings from that sensor.

Detailed procedure for setting up a two point calibration

You need a method of maintaining the temperature sensor at two known temperatures (T1 and T2) to perform this calibration. This procedure scales and offsets the calibration of this instance of this generic sensor across its temperature range. The calculated correction parameters will be applied to every reading of this sensor but will not be applied to any other sensor in the system even sensors of the same type using the same generic calibration curve.

For this procedure, T1 produces a lower sensor-resistance than T2:

- For a PTC sensor, T1<T2.

- For an NTC sensor, T1>T2.

Procedure:

1. Tap a configured panel on the Home page. The Temperature Sensor Details page is displayed.

2. Tap the *Calibration* field and from the drop-down menu select the generic file corresponding to your sensor (e.g. RP51.dat for a PT100).

3. Tap the *Cal Adj* button; the Generic Sensor Adjustment screen of Figure 3 will be shown.

4. Place the sensor in a location with temperature T1.

5. Enter the true temperature for T1 in the *Actual T* field and tap the *Calculate* button.

6. Place the sensor in a location with temperature T2.

7. Enter the true temperature for T2 in the *Actual T* field and tap the *Calculate* button.

8. Tap the *Calculate* button for the system to do the calculations.

9. Tap the *Save* button to apply the new calibration adjustment.

Notice

To enter a numerical value use the up and down arrows or tap the input field and use the virtual pop-up keypad.
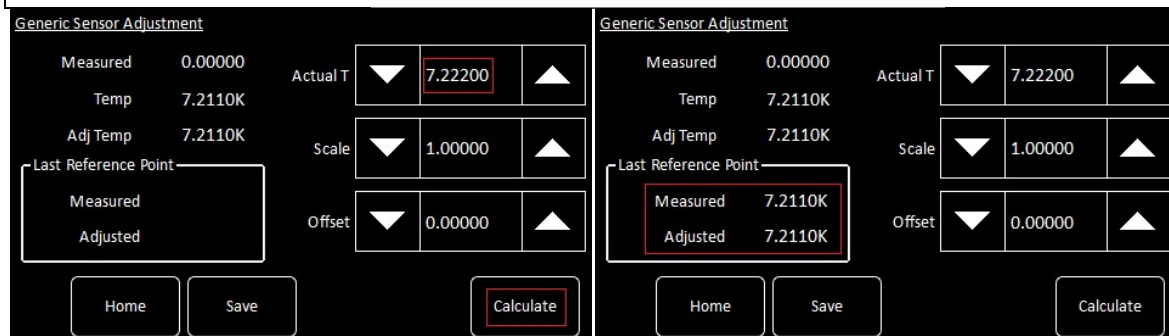

Figure 5. Calibration Adjustment, showing the actual temperature being entered (left) and the result of tapping "Calculate" (right).

Notice

The previous scheme, i.e. adjusting the Scale and Offset values manually, is still valid and can be used as an alternative method. For details please refer to the Mercury iTC manual.

Better Screen Response

The iTC contains a 4.3 inch full-colour liquid crystal display (LCD) with a touch-screen which is the main input device to the iTC. On previous versions of the firmware there were function call and touch position issues which sometimes resulted in either a touch being ignored or a control flagging a response (e.g. the virtual button changes colour) but the function call to execute the action did not occur.

In this release of firmware these issues have been addressed. The screen response has been improved by improving the touch screen position signaling and improved command processing has eliminated phantom taps.

Automated Firmware Updates

The user can either update the firmware for a particular card, or the application for the whole system. The firmware can be downloaded from the *mymercurysupport.com* website (see the cover page link) and transferred to the iTC by downloading it and extracting it to the root folder of a USB stick. The USB stick is then placed at the back of the unit for the update to be available in the iTC.

Notice

The system automatically disconnects any output signals while the firmware is being updated. It also re-connects the outputs once the update is completed.

The new firmware update procedure is completely automated and more robust than the previous one. Individual card firmware is now updated automatically by the application update if the existing version is older than the one in the update. Cards may still be programmed individually in case a user wants/needs a specific combination of firmware, but the automated procedure makes it easier to do an overall update (recommended).

The fully automated update includes retries in case a card update fails, and automatic recalibration and reconfiguration of the card after the update, eliminating issues due to temporary data loss during the update. The new firmware version will be displayed after each update, and can be seen in the Cards Settings tab.

1. Tap the *Settings* button on the Home page.

2. Select the *F/W Update* tab.

3. Tap the *Firmware Device* field and select *APPLICATION* from the drop-down menu.

4. Tap the *Firmware File* field and select a firmware file from the drop-down list.

5. Tap the *Apply* button. The iTC installs the new firmware and starts up in TRIAL mode.

Attention

Do not power-off the iTC in TRIAL mode!

6. Run some standard operations to test that the iTC is working as expected.

7. If the new firmware is acceptable:
   a) Tap the *Settings* button on the Home page.
   b) Select the *F/W Update* tab.
   c) Tap the *Save Changes* button to accept the new firmware version.

8. If the new firmware is not acceptable for some reason:
   a) Tap the *Settings* button on the Home page.
   b) Select the *F/W Update* tab.
   c) Tap the *Revert Changes* button to revert to the old firmware version.

For more details on the firmware update procedure please refer to the document *Mercury-ITC-and-IPS-firmware-updates.pdf* available on http://mymercurysupport.com/downloads/documents.

Remote Lock/Unlock Functions

The iTC contains several interfaces that can be connected to a remote PC. The remote PC can send data to the iTC and read data from the iTC. Only one interface can be active at a time. This can be selected from the drop-down menu. Select *Local* from the drop-down menu to allow the user to control the iTC from the GUI.

The new release allows for remote unlocked operation. When the remote lock is on, no feature in the unit is available for manipulation from the front screen. When the remote lock is off, only the settings features are disabled; allowing users to control their system from the front screen. The remote lock feature can be activated locally or remotely. The remote activation of the lock feature can be hard (SET:SYS:LOCK:ON) or soft (SET:SYS:LOCK:SOFT). In the first case, the user will not be able to deactivate it from the front screen (i.e. the navigation buttons will also be disabled). In order to turn off this feature remotely use the command SET:SYS:LOCK:OFF.

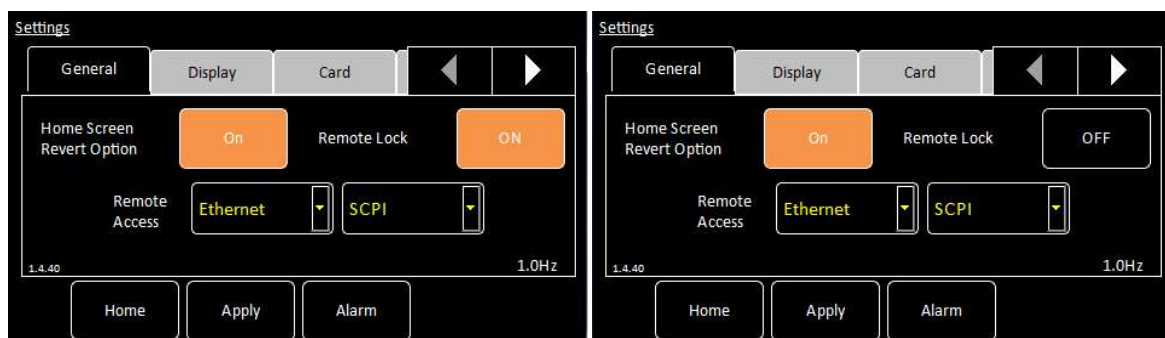| **Notice** |
| --- |
| When remote lock is off, since user controlled parameters can be changed on both the software and the unit, users are responsible for synchronizing the control elements in their own software. |

Figure 6. Remote lock (left) and unlock function (right).

Simplified Interface to Add or Remove Plot Traces.

The way the user creates or edits the plot has been reviewed in the current release and new signal selection interface shown in Figure 5 has been introduced.

The left hand side of the new interface contains a drop-down menu with available devices and a list of all signals available for plotting. By selecting a device in the drop down menu you can narrow down the list of signals to only those that belong to currently selected device, see the picture on the right of Figure 5. The right and left arrow buttons allow you to add a signal to plot (*Selected Signals* list) or remove it, up and down arrow buttons also let you arrange the order the traces are plotted in.

To plot the signal:

1. Tap the Plot button on Home screen; this will open the signal selection interface shown in Figure 5.

March 2016   Version 2.1.6

2. Tap once on the signal name, your selection will be highlighted in red.
3. Tap the right-arrow button to move the signal to the list of *Selected Signals*.
4. Repeat these steps to select all signals that you wish to plot.
5. To rearrange the signals order on the plot tap one signal from the *Selected Signals* list and move it up or down the list using up and down arrow buttons.
6. Tap the *save* button: this will create a new plot screen.

To add, remove or edit the order of traces in the current plot:

1. Double tap on the plot area; this will open the signal selection interface.
2. Use the arrow buttons to reorder, add or remove the traces from the list of *Selected Signals*.

To return to the Home screen from the plot screen:

1. Double tap on the plot area; this will open the signal selection interface.
2. Tap the *Home* button.

To remove all traces and terminate plotting:

1. Double tap on the plot area; this will open the signal selection interface.
2. Tap on each signal from the *Selected Signals* list and move it to the left list using the left arrow button.
3. When the *Selected Signals* list is empty, tap the *save* button.
4. Tap the *Home* button to return to Home screen.

Figure 5. New Interface for adding / removing plot traces.

Pressure Card Support

Version 1.4.40 has full pressure sensor support.

The iTC can interface with a wide variety of pressure sensors. A high-accuracy measurement circuit amplifies, scales and filters the voltage produced by the sensor. This conditioned voltage is then passed to a 24-bit analogue to digital converter (ADC). Additional sensors can be monitored by adding additional daughter boards. The pressure sensor input is via 9-way D-connector on the rear panel of the iTC.

There are several alternative sensor details. These depend on the sensor type.

Figure 6. Pressure Sensor Details

The page contains the following fields and controls:

- Sensor Name

- Sensor Readings

  The values shown are:

  I(mA) - the sensor reading in mA (only for Current sensors)

  V(mV) – the sensor reading in mV (for Voltage and Ratiometric sensors)

  P(mB) - the pressure measured by the sensor in mbar.

- Sensor Type

  Select a sensor type from a drop-down menu. Available types are:

  Voltage: for sensors returning a voltage signal (typically 0-5V or 0-10V). The maximum range of the pressure card is 10V, but the calibration file will determine the range of the selected sensor.

  Current: for sensors returning a current signal (typically 0-20mA or 4-20mA). The calibration file will determine the range.

  Ratiometric: these sensors return ratiometric voltages depending on the excitation of the sensor (output is typically in the 0-100mV range).

- Calibration

  Select a calibration file from a drop-down menu. The iTC uses calibration tables to store calibration data for different sensor types in the same way that temperature sensor calibrations are stored. The user can also create a new calibration file for a specific sensor. Calibration tables can be saved as files.

- Interpolation

   A calibration file contains a number of discrete data pairs, each mapping a sensor reading to a pressure value. Linear interpolation between adjacent pairs of points is used to provide pressure values from intermediate sensor readings. The iTC calculates setpoint limits and sensor limits from the chosen calibration file

- Excitation

   Indicates the voltage used to power (excite) the sensor. It will either be 0-10V or 0-15V. If the sensor output is required to go above 10V, then the excitation should be 15V. Check the sensor's datasheet to find out which excitation is adequate.

- Cal Adj button

   Tap once to access the Generic Sensor Adjustment page. This allows the user to adjust the scale and offset of a generic calibration curve to suit a particular sensor.

- Calibrate button

   Tap once to calibrate this measurement circuit. This routine takes approximately one minute. An error message is generated if the calibration fails for any reason.

Temperature Filtering

The temperature sensor board of the iTC can accept the following sensor types:

- Negative temperature coefficient (NTC) resistance

- Positive temperature coefficient (PTC) resistance

- Thermocouple

- Silicon diode

The iTC contains an additional temperature sensor that is used to compensate for fluctuations in ambient temperature. The unit can detect short-circuit inputs, open-circuit inputs, and inputs that are shorted to ground. Each input can be read at a maximum rate of 10 Hz.

The new version has a display filter for the temperature sensors in order to eliminate continuous value flickering in the user interface to make the fine detail of the temperature value easier to read. Note however, that control algorithms still use the source signal for greater stability and precision. The filter is a rolling buffer of 5 points which rejects the highest and lowest and takes a mean of the remaining 3 points plus the mean of the previous iteration.

Flow percent PWM output

This new firmware version includes a flow percentage PWM output on the auxiliary card, equivalent to the one available on the auxiliary output of the old ITC503. This output is provided for backward compatibility. The output has a 10 Hz fixed rate and a resolution of 1%.

The signal is output on pin 4 of the auxiliary port. The pin is alternately pulled to 0 V and released with a time dependent waveform, the mean value of which represents the analogue output. To use the output, a pull-up resistor should be fitted to a suitable reference voltage (e.g. the +5V on the auxiliary port pin 7, see Figure 7). A passive filter with a cut-off frequency of about 1 Hz should be connected between pin 4 and the device being used to measure the voltage.
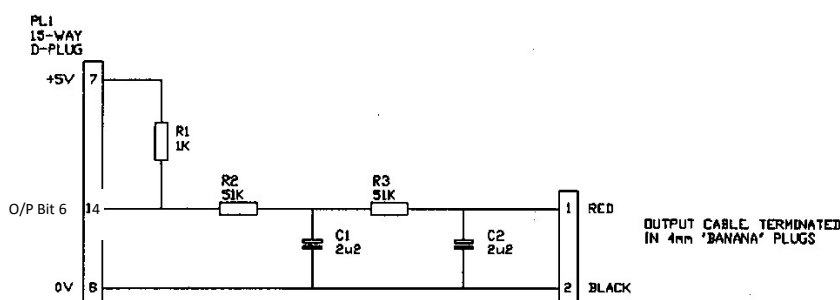

Figure 7. Example pull-up and filter circuit to add to and auxiliary port output
to filter the PWM signal to an analogue voltage

Set Rate Control for Temperature Ramps

In the old firmware the sweep function allows the setpoint to be programmed to follow a fixed temperature vs. time profile. The profile consists of a series of constant temperatures joined by linear ramps. During a ramp, the system updates the temperature setpoint every 100 ms. A program can contain up to 32 ramps and 32 steps. Each ramp and step lasts for a programmable time period.

Once a sweep has been programmed, it can be repeated as often as desired. It is also retained in non-volatile memory when the iTC is switched off. The Auto-PID facility may be used in conjunction with the sweep function to provide optimum control over a wide temperature range.

In the new firmware, additional functions allow for set point ramps. A ramp rate is set in K/m (Kelvin per minute) and when the ramp mode is enabled, any change in the set point will occur at this rate.

For this version of firmware RAMP mode can only be enable (or disabled) by remote SCPI command and is not available through the GUI. The mode is control loop specific so for iTC's with multiple PID control loops configured, setting one of them into ramp mode will not set the others into ramp mode.

To use ramp mode, follow the steps below:

- Send a remote command to set the desired ramp rate (K/min)

    e.g. SET:DEV:MB1.T1:TEMP:LOOP:RSET:5.25

    sets the ramp rate to 5.25K/min for the control loop controlled by the MB1
    temperature sensor input. The system returns

    STAT:SET:DEV:MB1.T1:TEMP:LOOP:RSET:VALID

- Send a remote command to read the temperature

    e.g. READ:DEV:MB1.T1:TEMP:SIG:TEMP

    The system returns

    STAT:DEV:MB1.T1:TEMP:SIG:TEMP:xxx.xxxxK

    Where xxx.xxxx is the temperature value.

Notice

Please note that the ramp will start from the current setpoint, not the current temperature. The
ramp starts as soon as the new set point is entered.

- Send a remote command to set the set point to the temperature just read as
  a starting point for the ramp

    e.g. SET:DEV:MB1.T1:TEMP:LOOP:TSET:xxx.xxxx

    sets the temperature set point to xxx.xxxxK for the control loop controlled by
    the MB1 temperature sensor input. The system returns

    STAT:SET:DEV:MB1.T1:TEMP:LOOP:TSET:VALID

- Send a remote command to put the loop in ramp mode

    e.g. SET:DEV:MB1.T1:TEMP:LOOP:RENA:ON

    sets Ramp mode ENAbled (RENA) ON for the control loop controlled by the
    MB1 temperature sensor input. The system returns

    STAT:SET:DEV:MB1.T1:TEMP:LOOP:RENA:VALID

- To start the ramp, send a remote command to set the set point to the desired temperature

  e.g. SET:DEV:MB1.T1:TEMP:LOOP:TSET:300.0

  sets the temperature set point to 300.0000K for the control loop controlled by the MB1 temperature sensor input. This will start the loop set point ramping at the ramp rate previously set

  STAT:SET:DEV:MB1.T1:TEMP:LOOP:TSET:VALID

- To disable ramp mode, send a remote command

  e.g. SET:DEV:MB1.T1:TEMP:LOOP:RENA:OFF

  sets Ramp mode ENAbled (RENA) OFF for the control loop. The system returns

  STAT:SET:DEV:MB1.T1:TEMP:LOOP:RENA:VALID

Notice

Please note that if ramp mode is disabled during a ramp the set point will jump to the final target setpoint.

Extended Remote Command Set

The iTC supports two separate sets of remote commands:

- SCPI commands are based on the SCPI command set, although the iTC may not be fully compliant with SCPI.

- Legacy commands were used in previous temperature controller models (e.g. ITC503).

The user configures the iTC to use one set or the other. It is recommended that new applications use the SCPI command set, as these commands provide greater functionality. The legacy command set is provided so that the iTC is compatible with any existing remote applications that have been written to drive older OINS equipment.

The new functionality is available remotely, along with some remote commands for older functionality which did not exist.   See the updated SCPI command list at www.mymercurysupport.com/documents .