SUBJECT: SPIRE Test Facility Control System software description

PREPARED BY: S. Ronayette

DOCUMENT No: SPIRE-RAL-DOC-002486

ISSUE: Issue 1.0 **Date:** 29th April 2005

APPROVED BY: **Date:**

Change Record

ISSUE	DATE	Changes
Issue 1.0	29th April 2005	First Issue

Prepared by S.Ronayette - April 2005.

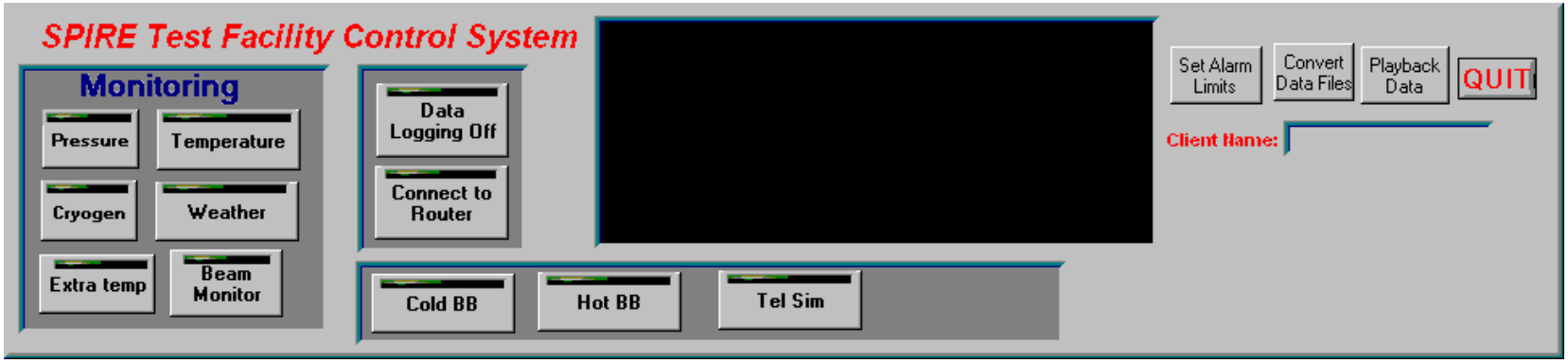
This note is not a comprehensive documentation about the TFCS code, but intend to give a good overview, and also help anyone who wishes to modify it, especially if the TFCS HK packet definition is modified or if the reading of the temperature sensors is modified.

1 TFCS-Menubar (in TFCS-Menubar.llb)

1.1 Overview

TFCS-Menubar.vi is the main VI at the top of the hierarchy. All the sub-Vis for monitoring the data from and sending command to the test facility equipment (pressure, cryostat and auxiliary temperatures, cryogen level, weather station, beam monitor, cold black body, hot black body, telescope simulator) can be called from its front panel. This VI also ensures the connection to the router, reception of the commands sent from SCOS and sends the data to the router. It logs all the data in a .tfc files, call the sub-VIs to playback those data or to convert it into a .txt file.

1.2 Front Panel

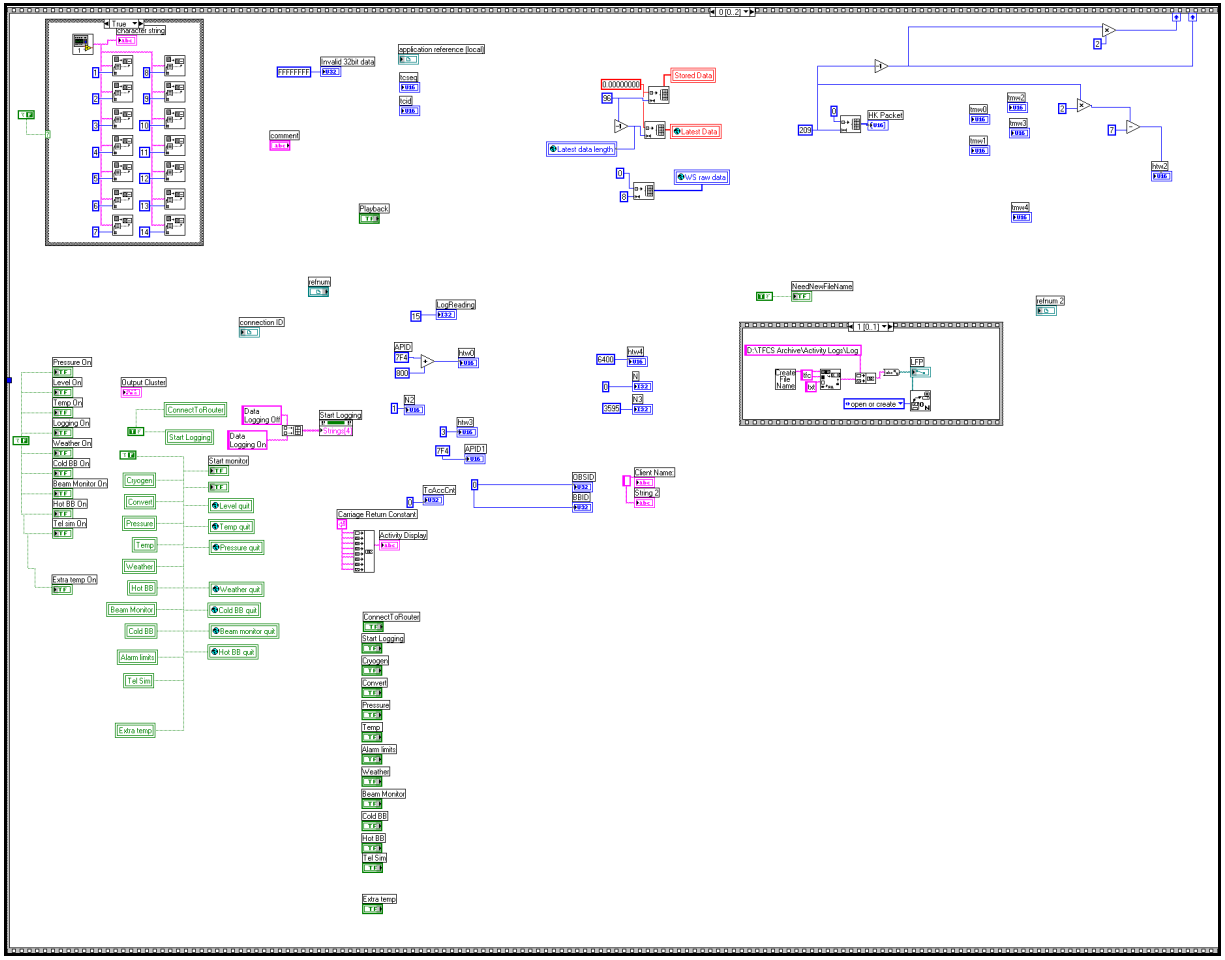


1.3 Block Diagram

1.3.1 Initialisation

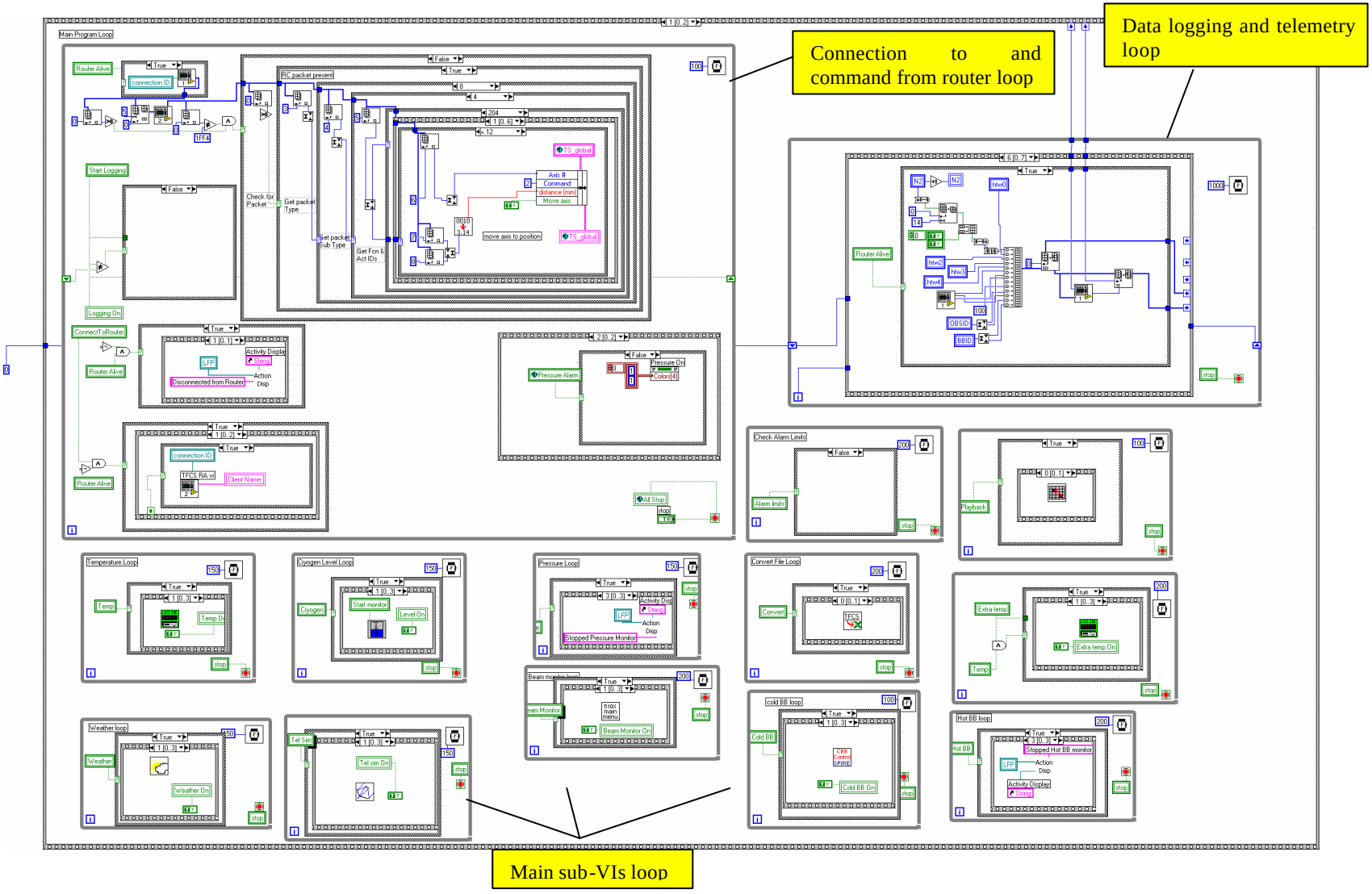
This first frame initializes the variables. Some of the main variables are described below:

- Latest Data: global variable. It contains an array of double that is filled with data in all the sub-VIs that have data to log
- Stored Data: array of double, containing one more element than Latest Data. The first element is the system time and the rest is the content of Latest Data. Stored Data is written into the data file .tfc every 15 seconds.
- HK packet: array of 16-bit integers. Contains the telemetry sent to the router (as defined in the TFCS data ICD)



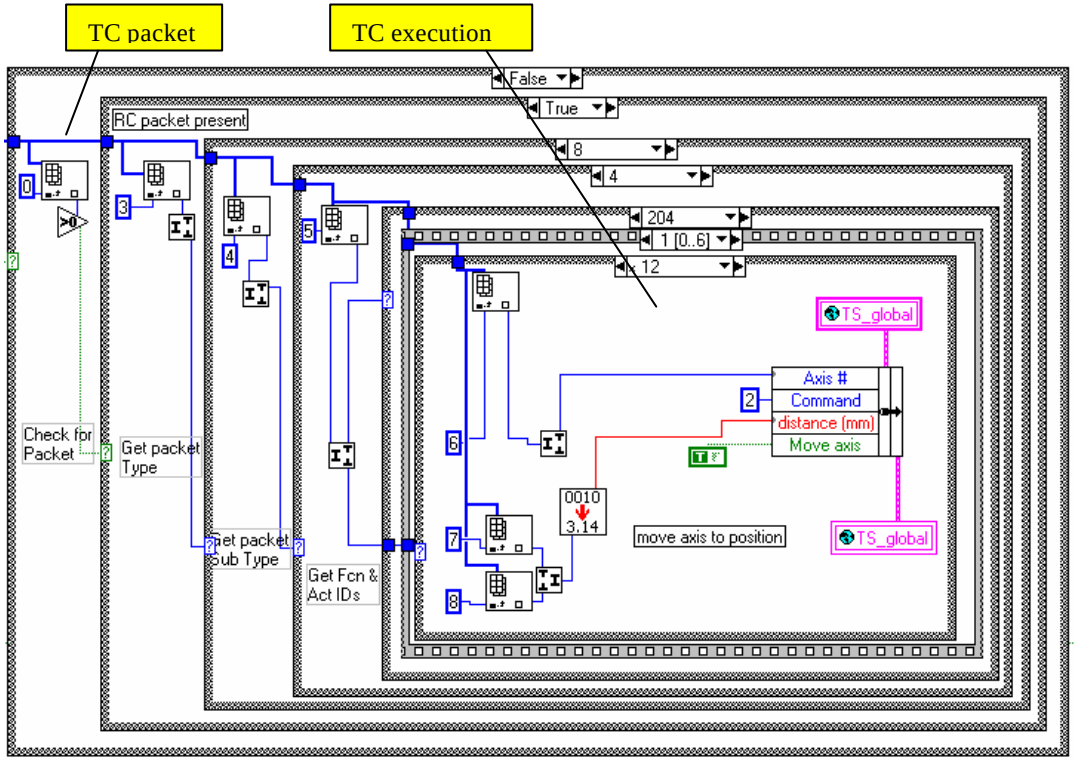
1.3.2 Main program frame

This frame contains several “while loop” that end all together when the QUIT button on the TFCS front panel is pressed.


Connection to and command from router loop
Data logging and telemetry loop
Main sub-VIs loop

1.3.2.1 Connection to and commands from router loop

This loop contains the VIs that establish the connection with the router. If the connection is established, a VI runs to check if a TC packet is received. If a packet is received, a succession of case structures checks for packet type, sub-type, function and ID. The example below shows the execution in the event of a command sent to move the telescope simulator:



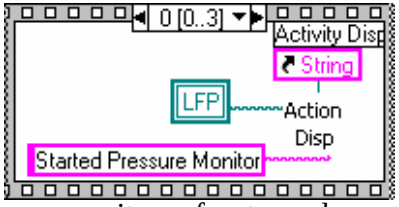
1.3.2.2 Main sub-VIs loops

There are 12 of them, one for each main sub-VI:

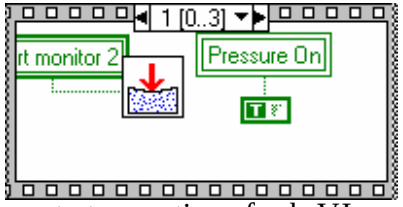
- Cryostat temperatures monitoring
- Cryogen level monitoring

- Pressure monitoring
- Weather station monitoring
- Telescope simulator
- Beam monitor
- Convert data to file
- Cold Black Body
- Hot Black Body
- Auxiliary temperatures monitoring
- Playback data
- Set alarm limits (unused?)

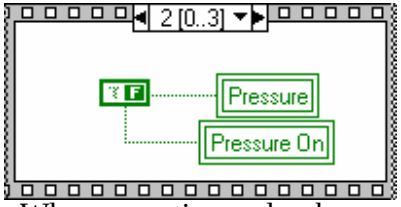
When a button on the front panel is pressed to call such a sub_VI, the following sequence executes (example for pressure):



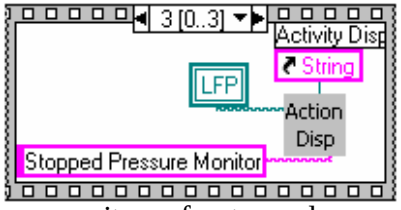
write on front panel
action taken



start execution of sub-VI



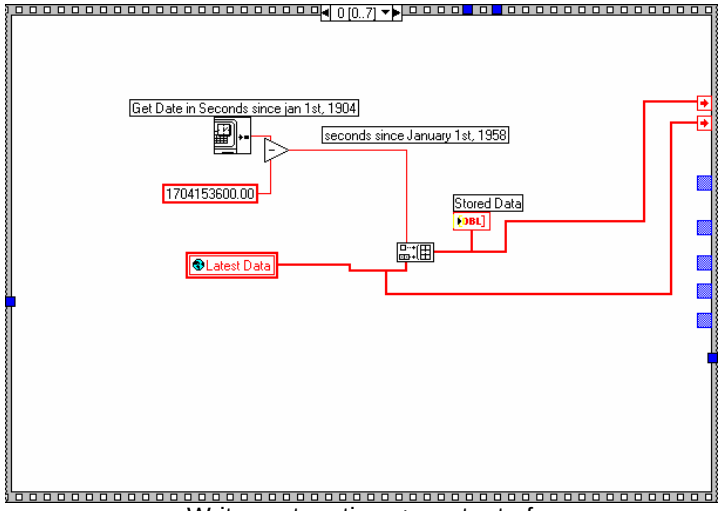
When execution ends, change
states of Booleans



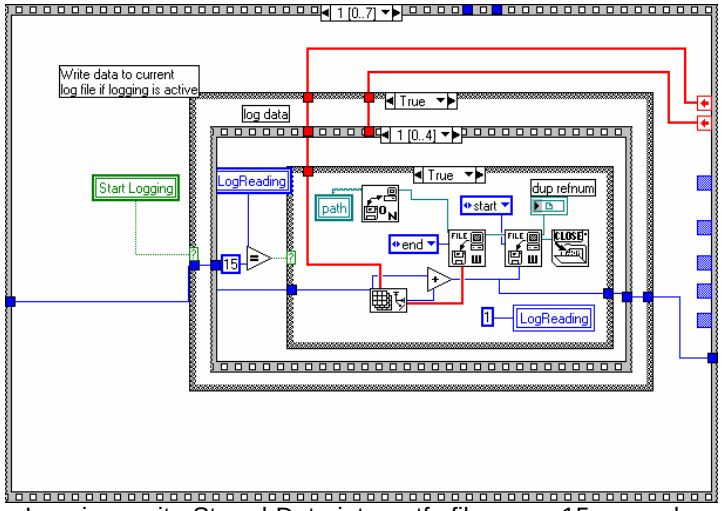
write on front panel
action taken

1.3.2.3 Data logging and telemetry loop

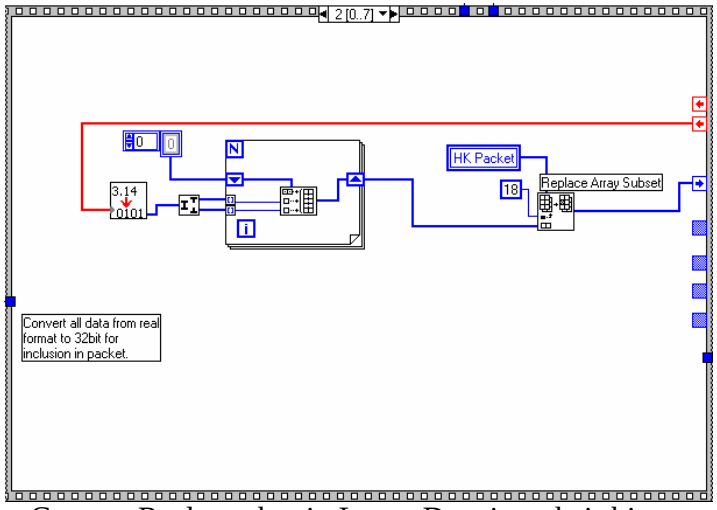
This loop executes every seconds. The different actions are detailed below:



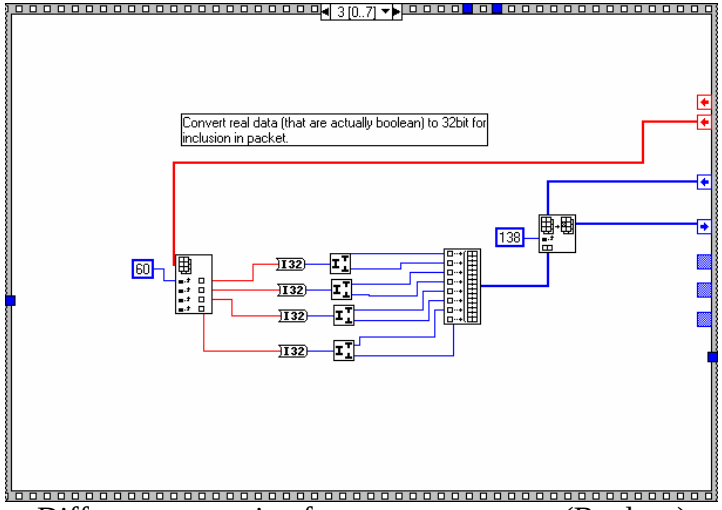
Write system time + content of
Latest Data into Stored Data



Logging: write Stored Data into a .tfc file every 15 seconds.
Create new file name if necessary

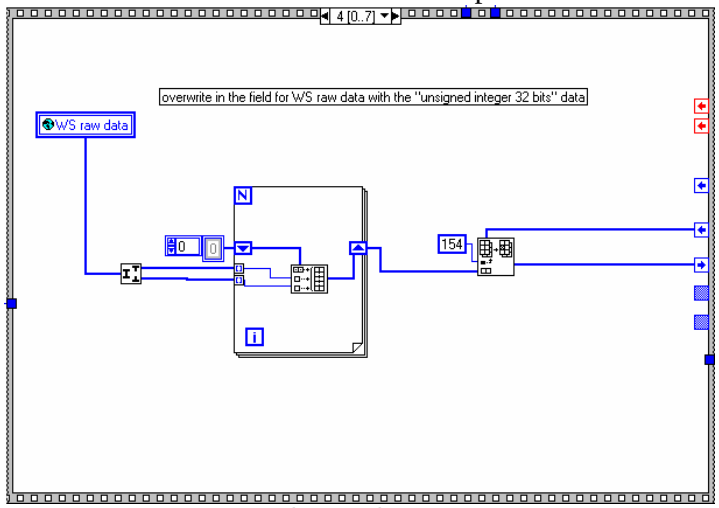


Convert Real number in Latest Data into their binary
representation on 32 bits (IEEE-754). Split it in 16-bits

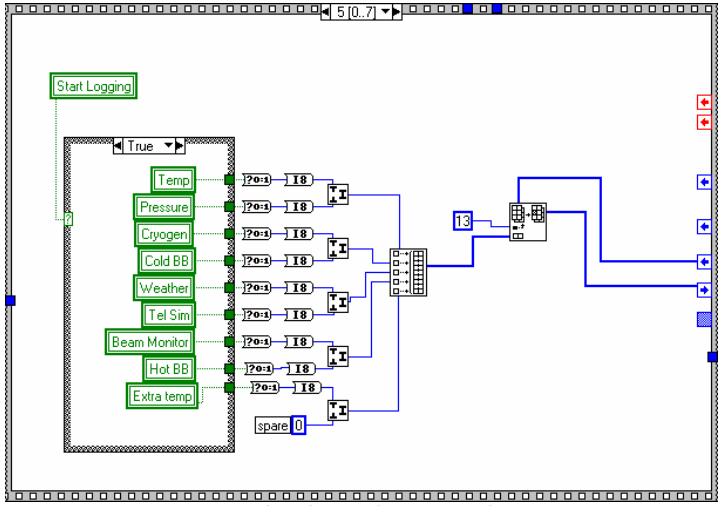


Different conversion for some parameters (Boolean)
and store them into HK packet

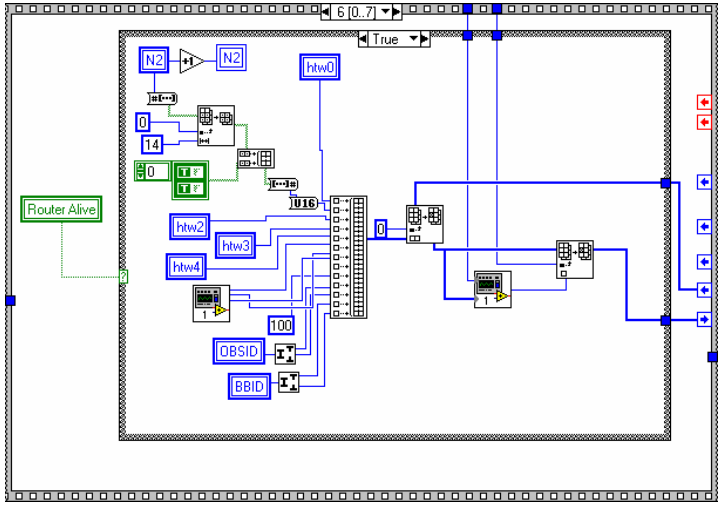
words and store into HK packet



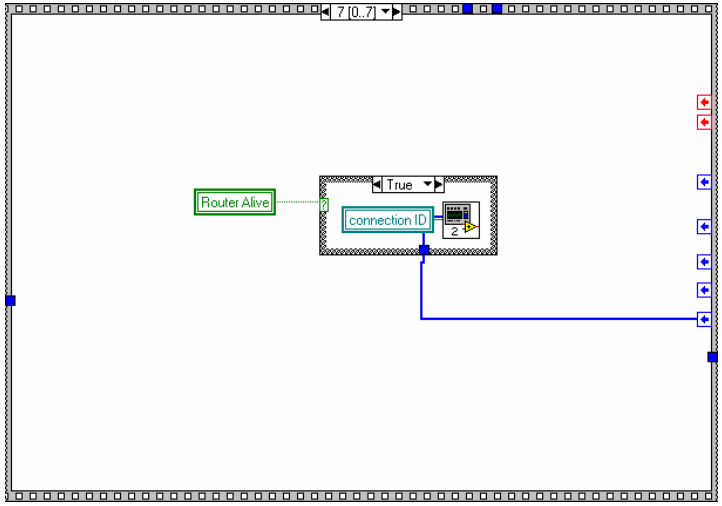
Store weather station parameters
(already 32 bits – don't need conversion)



store booleans in HK packet



Write header and checksum in HK packet



Send HK packet to router

1.3.2.4 Correspondence between HK packet, Latest data and data field

If the definition of the HK packet data field changes, it is useful to know the correspondence between the indexes of the different array in order to update the code and keep it in compliance with the ICD. It is summarized in the following table:

Octet location in Data field (defined in TFCS ICD)	HK packet array index in LabView	Latest Data array index
N	(N+16)/2	(N-20)/4
-	i	(i-18)/2

2 Temperature Monitoring (TempMonitor.llb)

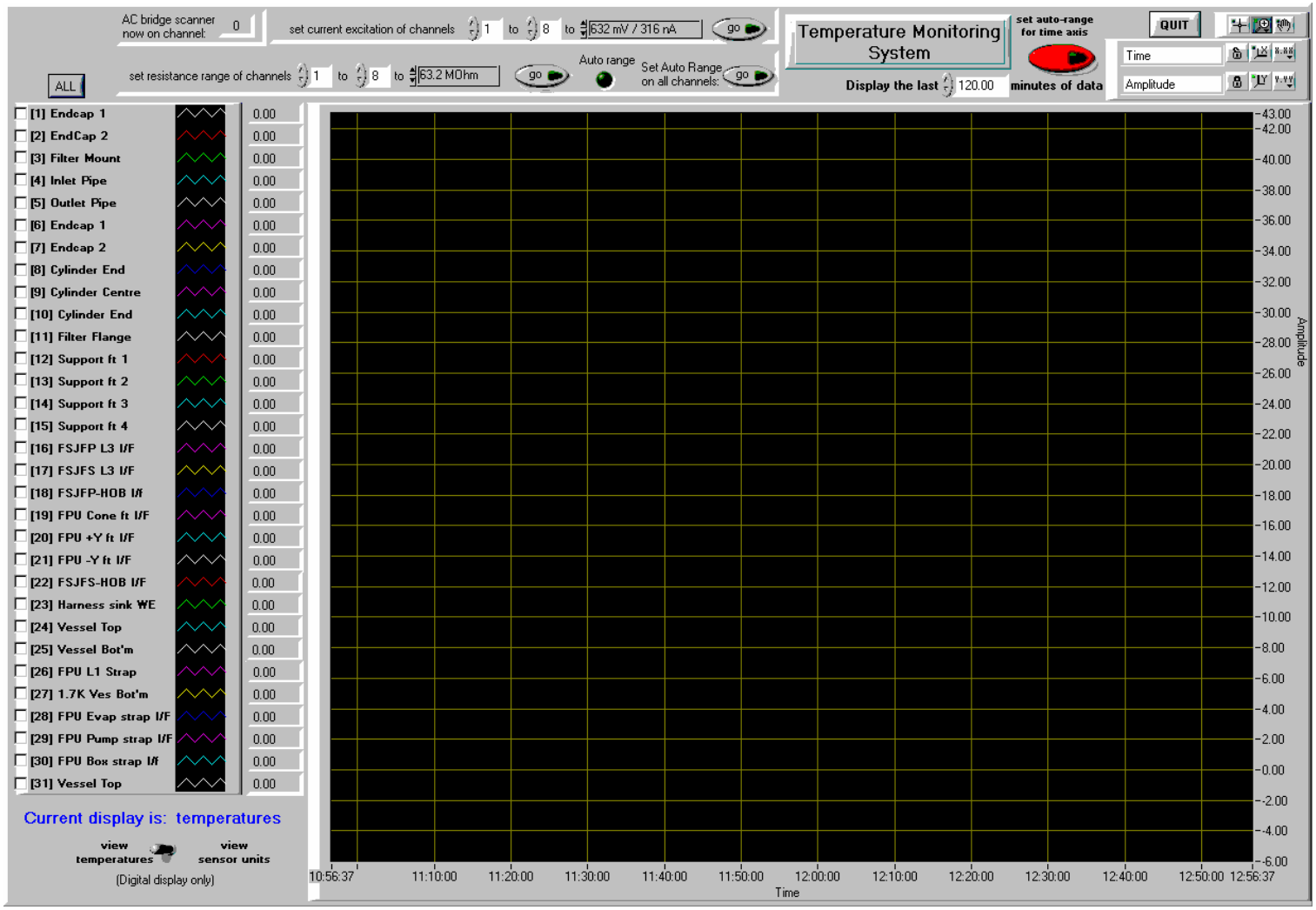
2.1 Overview

This sub-VI initialize the lakeshore units, read data from them every 15 seconds and log the cryostat temperatures into the Latest Data array (global variable). The description that follows is for the Basic Temperature Monitoring (cryostat). The VI for the extra temperatures monitoring (instrument) is very similar and is not described.

2.2 Front Pane

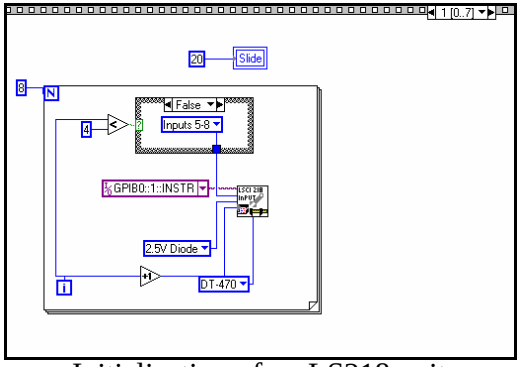
The front panel consists in an XY graph that displays the 31 temperatures. Tick boxes for each temperature allow the user to choose whether a temperature is visible on the graph or not (this does not affect the logging. All the temperatures are logged at all times). If the button “set auto-range for time axis” is ON, the range on the time axis is automatically set to the value specified in the field “display the last xx minutes of data” (can be set between 0 and 180 minutes)

From the front panel, command can be sent to set the current excitation on any of the 16 channels of the AC bridge. (refer to the CQM-thermometers.xls spreadsheet for the correspondence between temperature sensor and channel). The resistance range can be set as well, or set to “auto”.

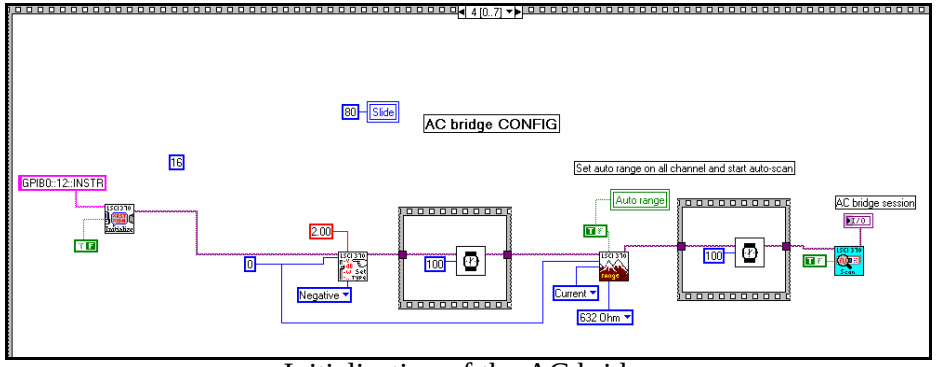

AC bridge scanner now on channel: 0
set current excitation of channels 1 to 8 to 632 mV / 316 nA go
set resistance range of channels 1 to 8 to 63.2 MOhm go
Auto range
Set Auto Range on all channels: go
ALL
Temperature Monitoring System
set auto-range for time axis
QUIT
Time
Amplitude
Display the last 120.00 minutes of data
[1] Endcap 1 0.00
[2] EndCap 2 0.00
[3] Filter Mount 0.00
[4] Inlet Pipe 0.00
[5] Outlet Pipe 0.00
[6] Endcap 1 0.00
[7] Endcap 2 0.00
[8] Cylinder End 0.00
[9] Cylinder Centre 0.00
[10] Cylinder End 0.00
[11] Filter Flange 0.00
[12] Support ft 1 0.00
[13] Support ft 2 0.00
[14] Support ft 3 0.00
[15] Support ft 4 0.00
[16] FSJFP L3 I/F 0.00
[17] FSJFS L3 I/F 0.00
[18] FSJFP-HOB I/f 0.00
[19] FPU Cone ft I/F 0.00
[20] FPU +Y ft I/F 0.00
[21] FPU -Y ft I/F 0.00
[22] FSJFS-HOB I/F 0.00
[23] Harness sink WE 0.00
[24] Vessel Top 0.00
[25] Vessel Bot'm 0.00
[26] FPU L1 Strap 0.00
[27] 1.7K Ves Bot'm 0.00
[28] FPU Evap strap I/F 0.00
[29] FPU Pump strap I/F 0.00
[30] FPU Box strap I/f 0.00
[31] Vessel Top 0.00
Current display is: temperatures
view temperatures
view sensor units
(Digital display only)
43.00
42.00
40.00
38.00
36.00
34.00
32.00
30.00
28.00
26.00
24.00
22.00
20.00
18.00
16.00
14.00
12.00
10.00
8.00
6.00
4.00
2.00
0.00
-2.00
-4.00
-6.00
Amplitude
10:56:37
11:10:00
11:20:00
11:30:00
11:40:00
11:50:00
12:00:00
12:10:00
12:20:00
12:30:00
12:40:00
12:50:00
12:56:37
Time

2.3 Block diagram

2.3.1 Initialization



Initialization of an LS218 unit



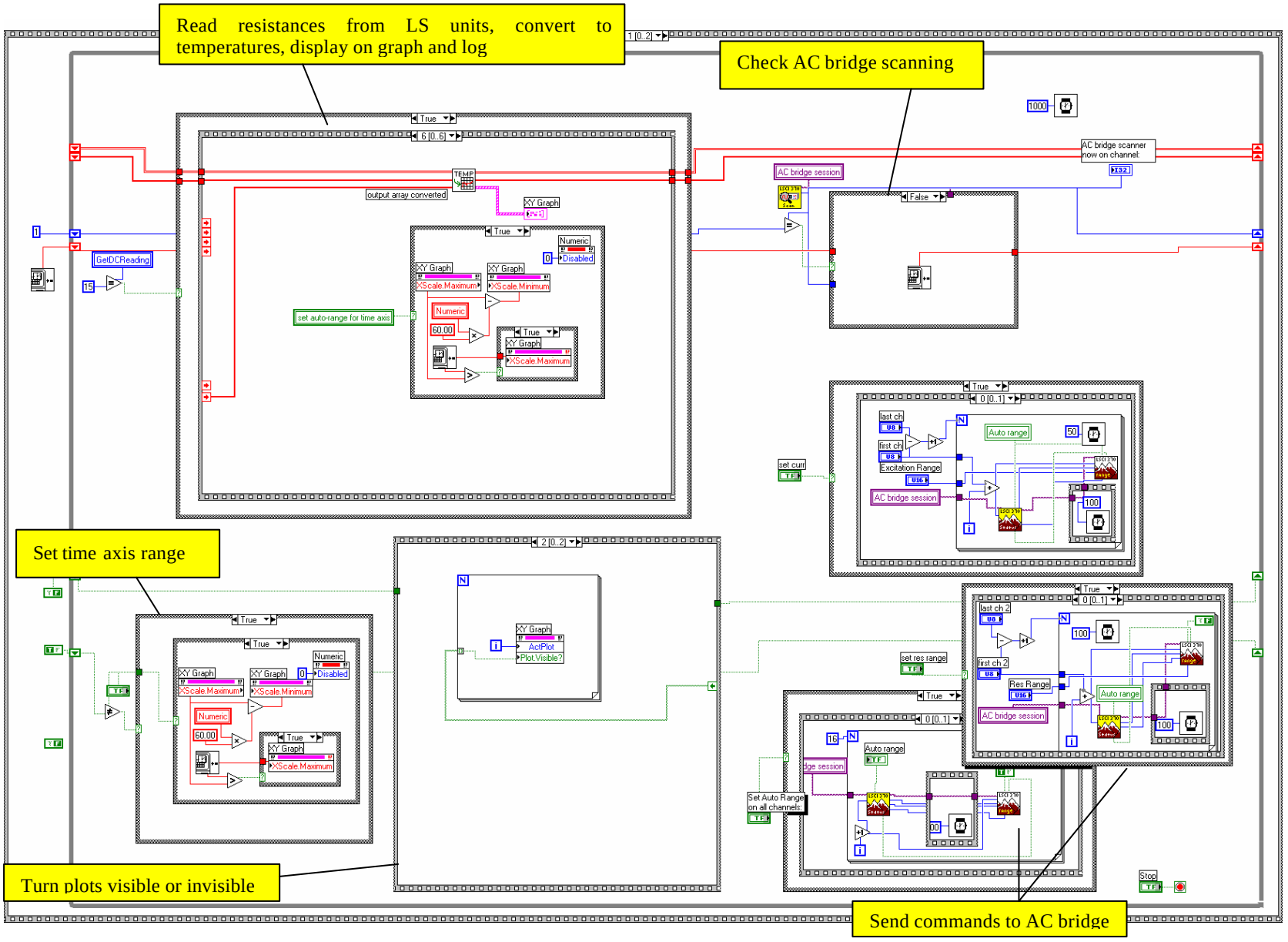
Initialization of the AC bridge

2.3.2 Main loop

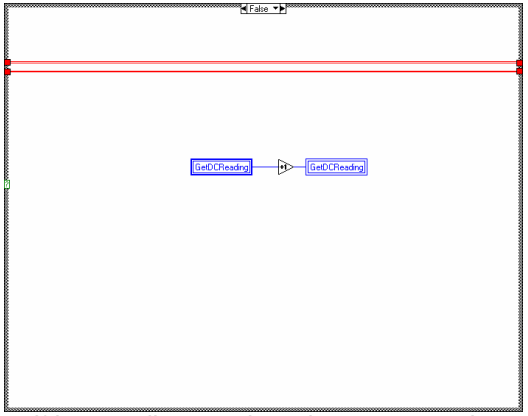
This loop executes every seconds. The main actions in this loop are the following:

- Read resistances from LS units, convert to temperatures, display on graph and log
- Set time axis range
- Turn plots visible or invisible
- Check AC bridge scanning
- Send commands to AC bridge

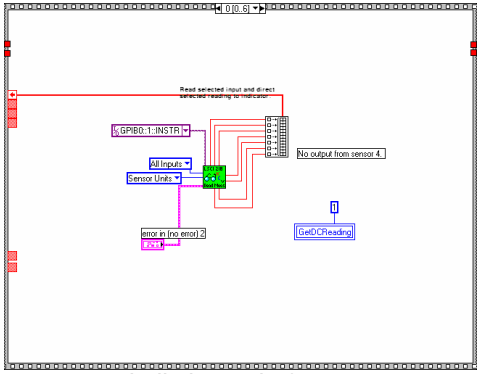
The VI doesn't send command to select a channel on the AC bridge, so the AC bridge must be on auto-scan mode (which is done in the initialization). However, the "check AC bridge scanning" feature sends a command to skip a channel if the scanner stays on the same channel for more than 30 seconds. This may happen indeed, when the auto-range is ON and the AC bridge can't find an appropriate range because the reading from the thermometer is invalid (usually on a dead thermometers - such a channel should be turned off by sending the following command to the LS370 unit: **INSET n, 0** where n is the channel number)


Read resistances from LS units, convert to temperatures, display on graph and log
Check AC bridge scanning
Set time axis range
Turn plots visible or invisible
Send commands to AC bridge
GetDCReading
output array converted
set auto-range for time axis
AC bridge session
AC bridge scanner now on channel:
Auto range
Excitation Range
Res Range
Set Auto Range on all channels
Stop

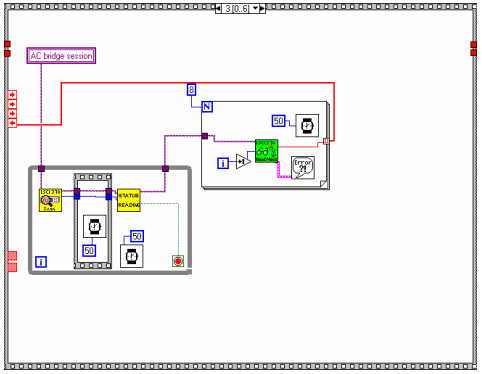
Details of the temperatures reading loop:



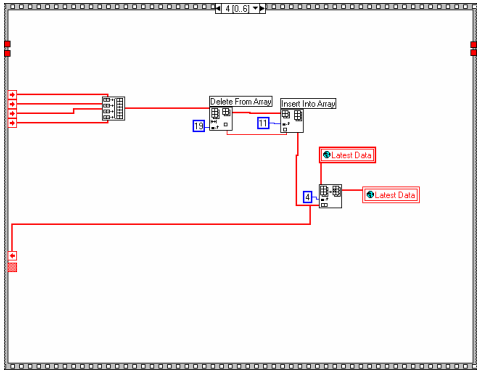
if last reading was less than 15 seconds ago, increment variable and do nothing else



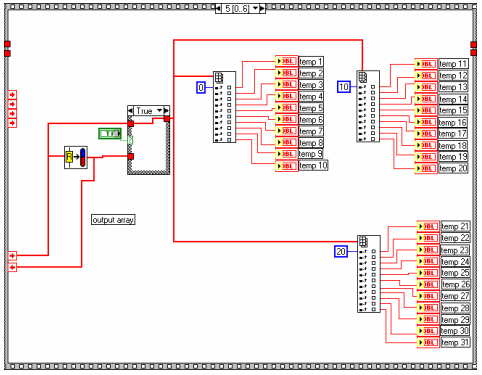
read all channel of a LS218 lakeshore unit



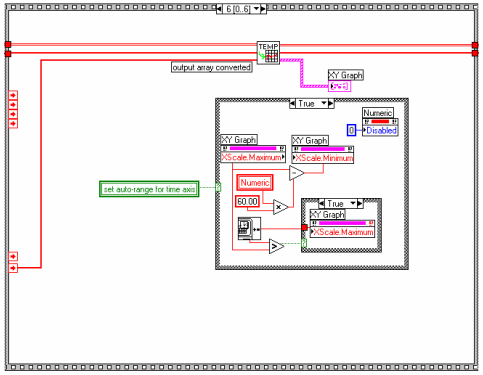
LS370 AC bridge: check that reading on the currently scanned channel is valid and read all channels at once



build array with all channel, in compliance with the spec in Excel spreadsheet
Store into Latest Data



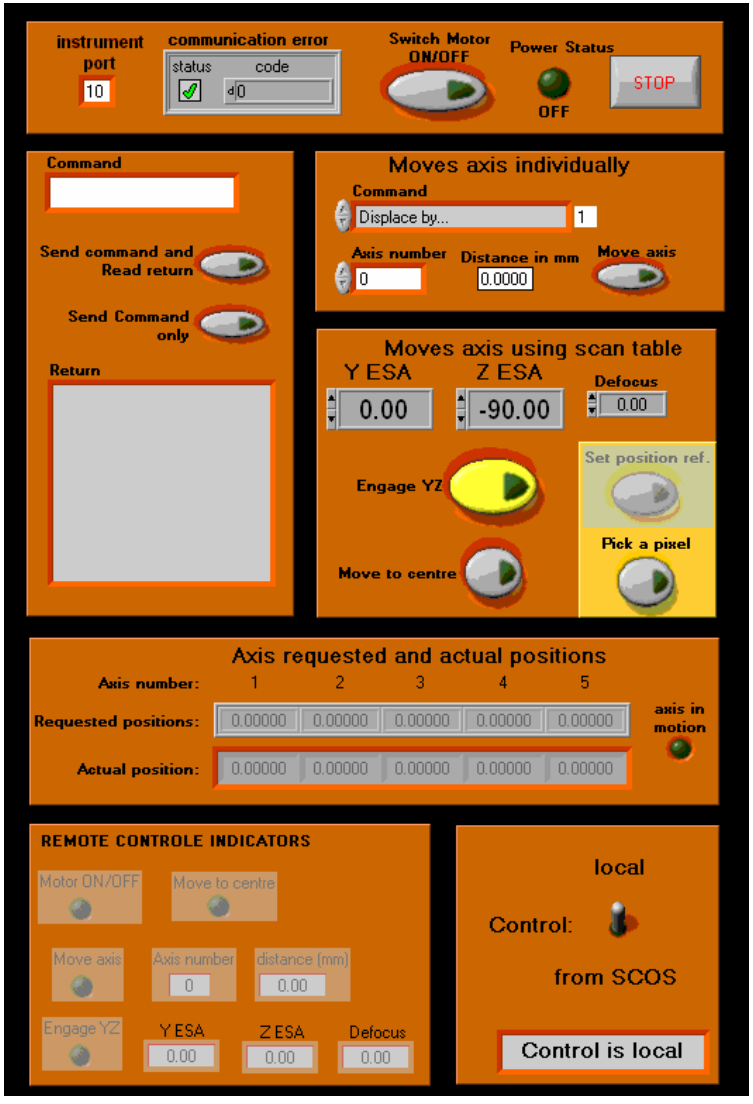
Convert resistances to temperatures and display either resistance or temperatures



Display temperatures on XY graph
set range of time axis

3 Telescope simulator (in telsim_control.llb)

3.1 Overview



The Telescope simulator VI ensures the connection with the Newport motion controller. It sends commands to the motion controller in order to move the actuators on the mirrors. It receives feedback from the motion controller (motor status, actuator position). It uses the dll "TelSim_Table.dll" to convert the required X, Y, Z position in the field of view to actuator position. To use a new scan table, stop the TFCS, place the .dll in the "data" folder, rename it "TelSim_Table.dll" and restart TFCS.

3.2 Front Panel

From the front panel, the user can:

- switch the motors ON and OFF.
- send any command and read the return.
- Move axis individually
- Send the beam to a position Y, Z in SPIRE's FoV Y, Z
- Read the commanded and actual position of the actuators
- Give control to SCOS

See "how to operate the Telescope Simulator during SPIRE testing" for more details.

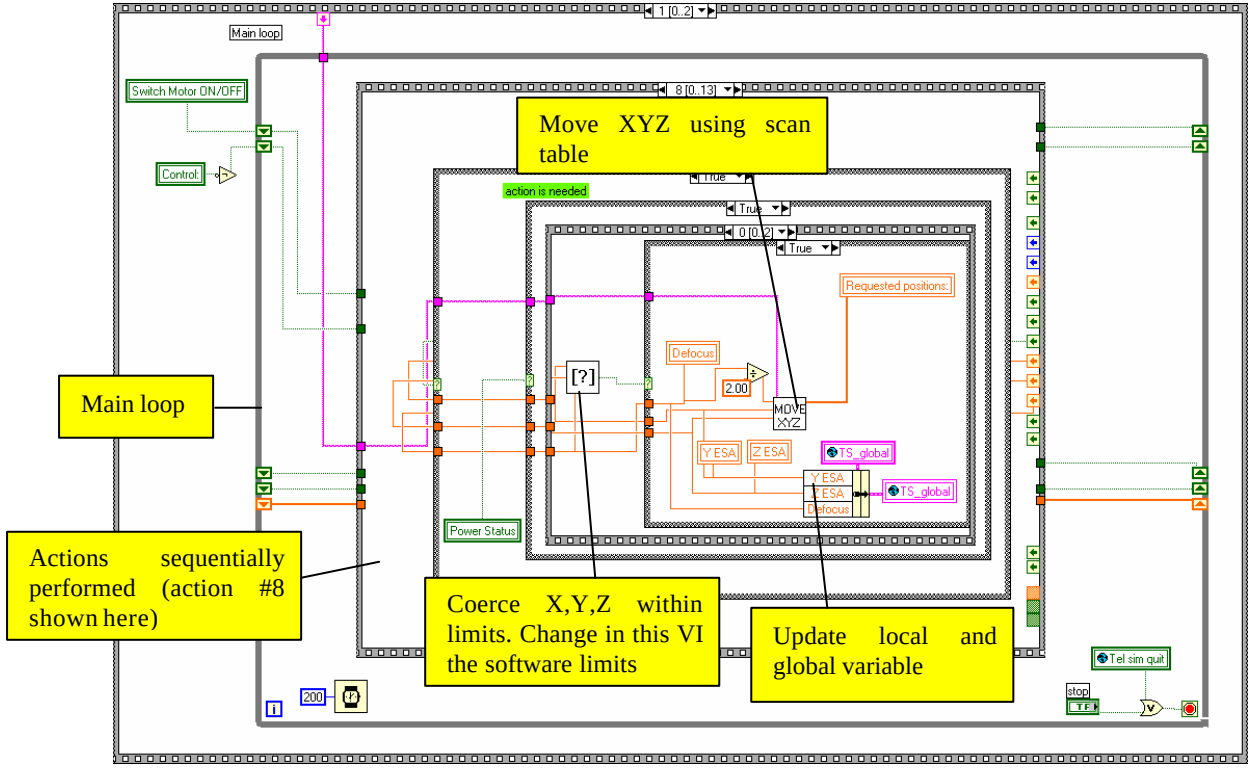
IMPORTANT NOTE: before switching the motion controller to stand-by, the actuators must be sent to 0 by pressing the button "move to centre".

Note: the "set position ref" button is disabled because I thought that is not a good idea to use it, taking into account the large

difference between the expected and the actual Y,Z coordinates used during PFM1 for pointing at the centre of the spectro FoV (c.f. word document “how to use the telescope simulator during spire testing”)

3.3 Block diagram

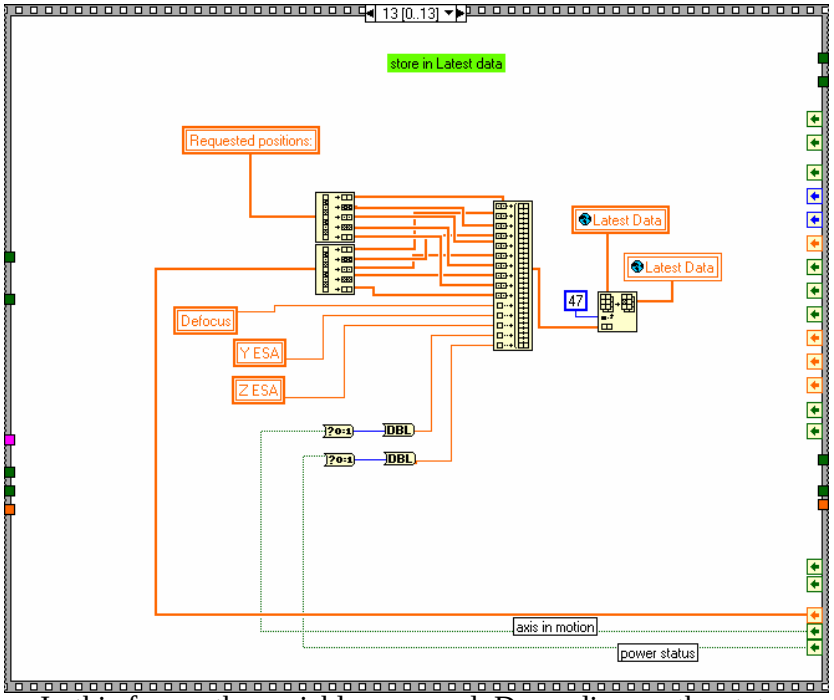
After a sequence of initialization, the program is in the main loop, in which various actions are performed sequentially. Most of these actions consist in checking an ON/OFF control from the front panel. If a control is "ON", the corresponding action is performed. There are 14 of these actions listed below:



(0) "GetReading" check
(1) Control type check (remote or local)
(2) Read control variables
(3) Turn motors ON or OFF
(4) Send command to MM4006
(5) Read return from MM4006
(6) Move one axis
(7) Move to centre
(8) Move using scan table
(9) Set position reference
(10) Pick a pixel
(11) Read motors status
(12) Read motors position
(13) Store data in global variable

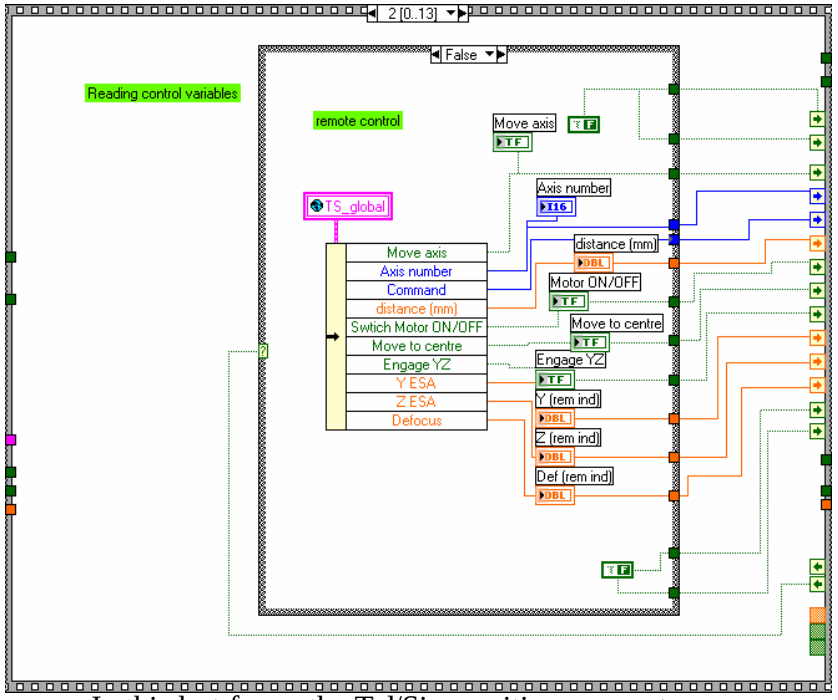
2 other frames are detailed below:

Action (2) Read control variable:



In this frame, the variables are read. Depending on the state of the control (remote or local), the variable passed to the rest of the program come either from the remote control global variable or from the local variable read on the front panel

Action (13): Store data in global variable



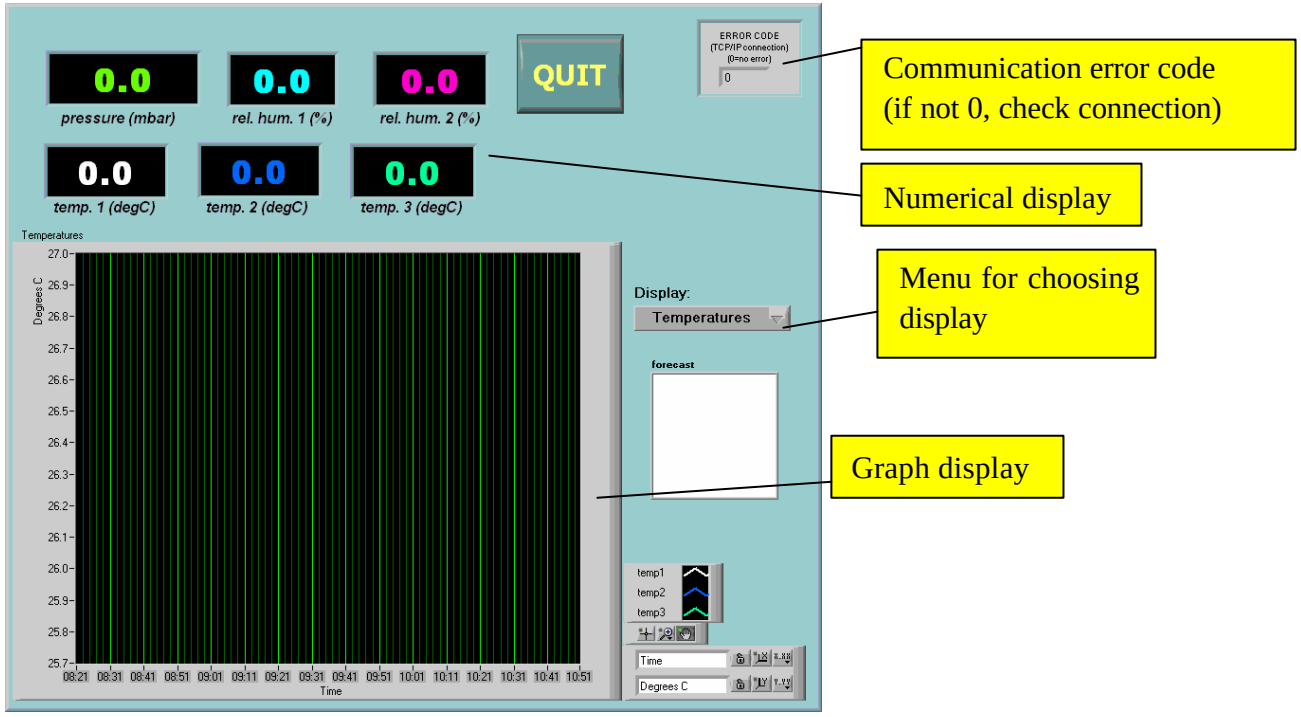
In this last frame the Tel/Sim position parameters are stored in the global variable. These are the data written to the tfcs log file and sent to the router.
They are: requested positions on 5 axis, actual position of 5 axis, defocus, Yesa and Zesa parameters, axis in motion flag and power status.

4 Weather station (in TFCS-Menubar.llb)

4.1 Overview

The weather station VI communicates with the weather station via TCP/IP. It requests at regular intervals the data measured by the weather station: atm. pressure, 3 temperatures, 2 relative humidity

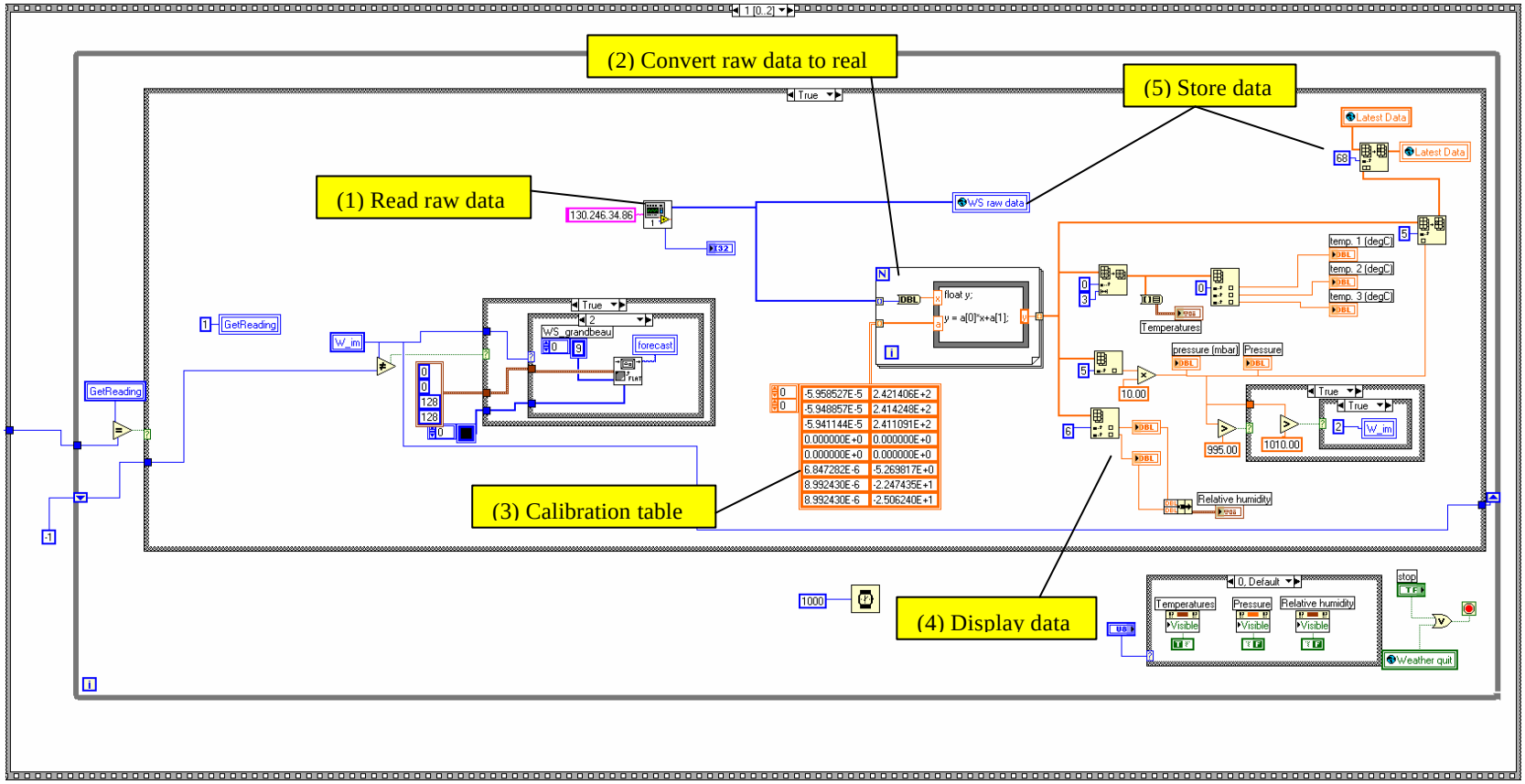
4.2 Front Panel



4.3 Block diagram

The code of the weather station VI is pretty straightforward. Every 30 seconds the following action are performed:

- SubVI “WS_read_raw.vi” is called with the appropriate IP (1). Returns the raw data (32 bits number)
- The parameter are converted to real value (2) using the conversion table (3)
- The converted data are sent to graphical and numerical display (4)
- Data are stored in global variables (5) (raw data are stored into HK packets, converted data are stored into TFCS log files)



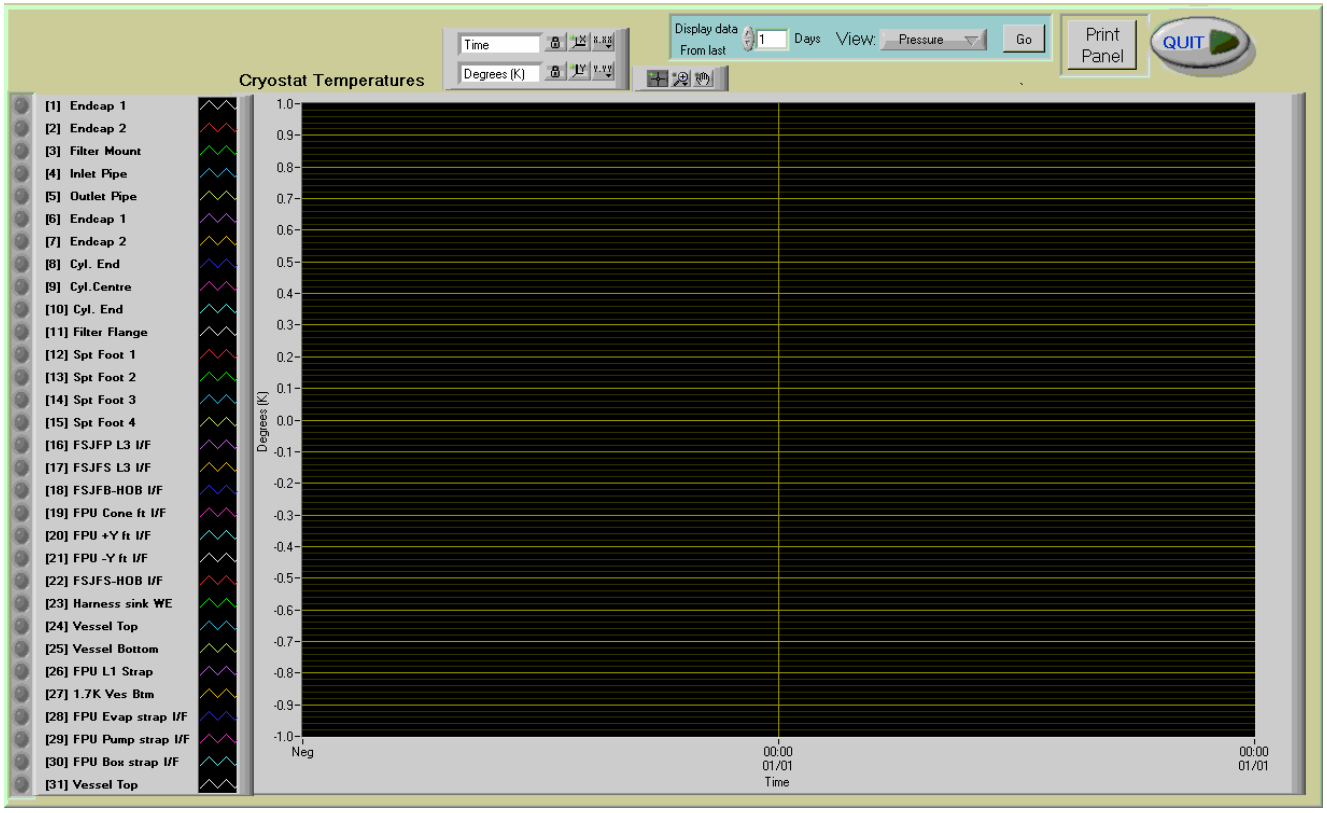
5 Playback display (in TFCS-Menubar.llb)

5.1 Overview

The playback display consists of a main program displaying the data returned by a subroutine called inside this main program. The subroutine (datepicker.vi) reads the .tfc log file of the TFCS to return the data for the time range desired. In order to read the data

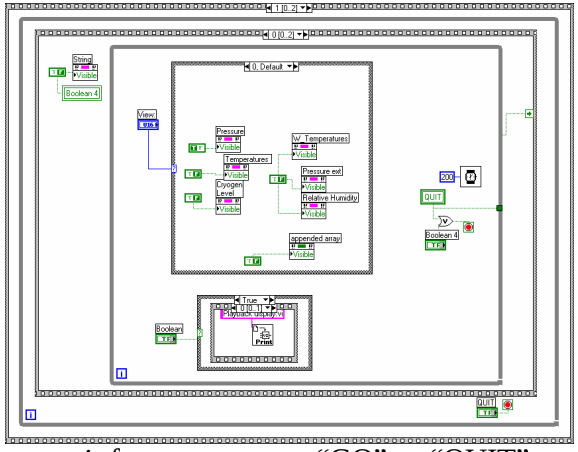
properly, the datepicker subroutine must comply with the structure of the TFCS log files (which is the same than the data field of the TFCS HK packets).

5.2 Front Pane

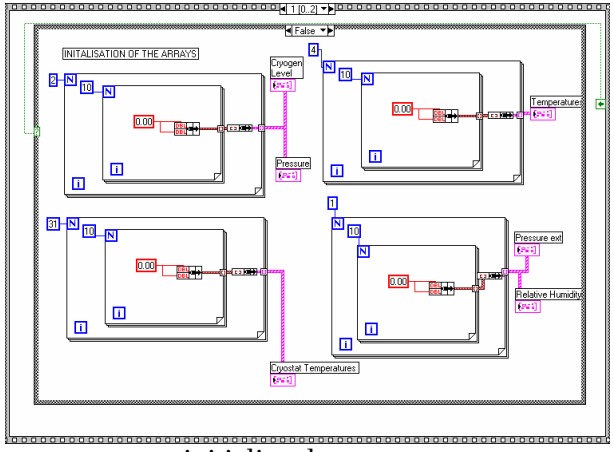


5.3 Block diagram

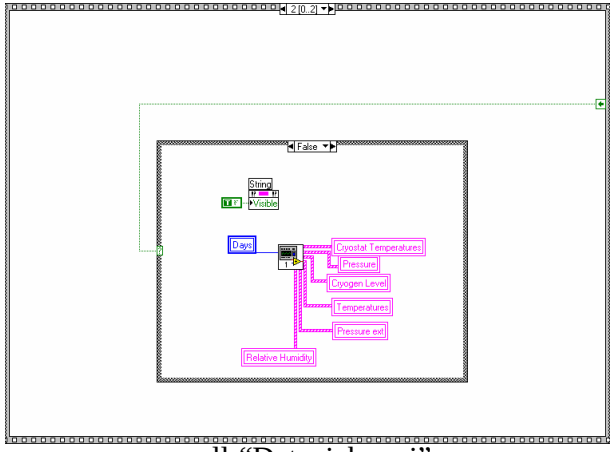
The sequence below shows the main steps executed in “playback Display.vi”



wait for user to press “GO” or “QUIT”
Display selected data

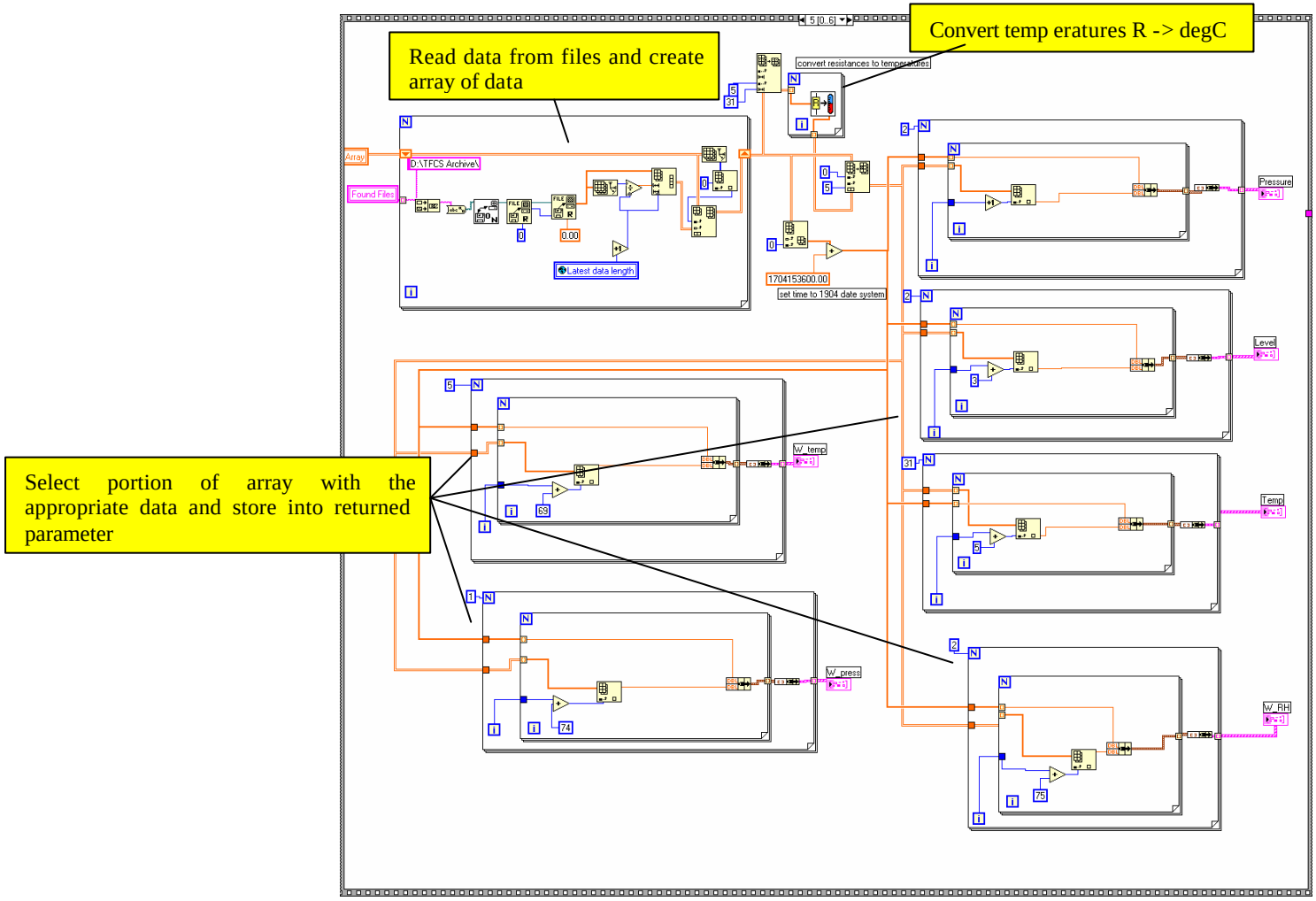


initialize data arrays



call “Datepicker.vi”

After retrieving the appropriate log files, the Datepicker.vi subroutine store the data into the arrays as shown below

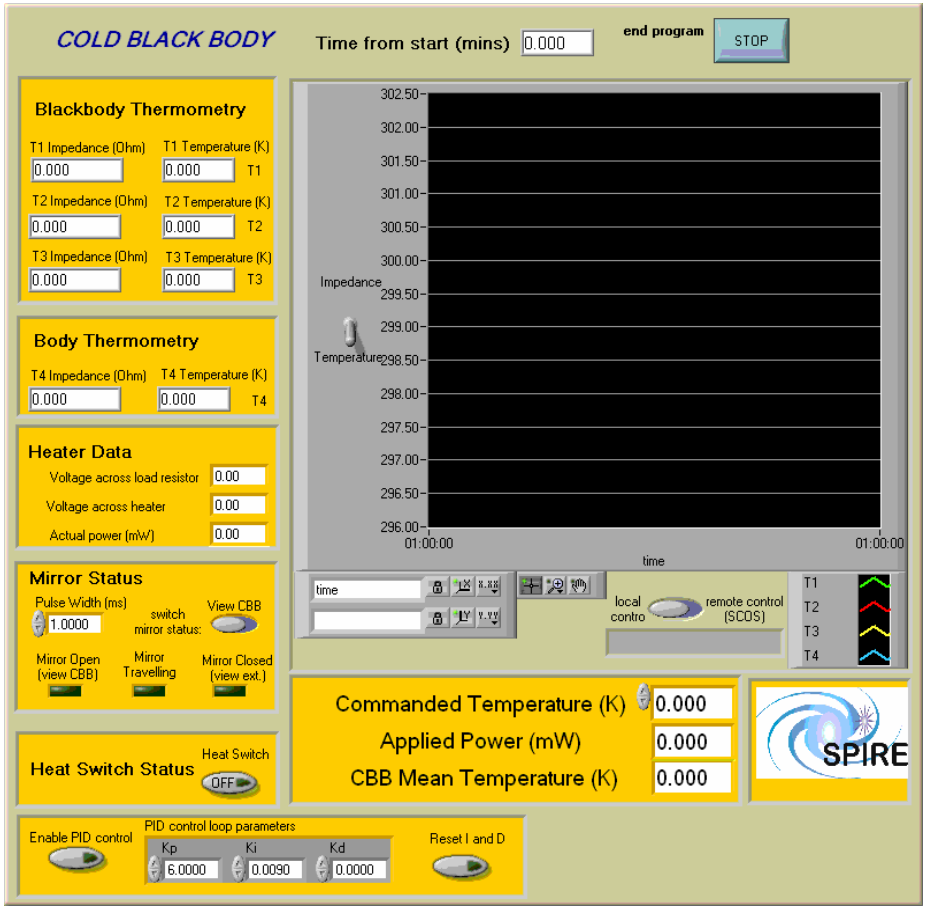

Read data from files and create array of data
Convert temp eratures R -> degC
Select portion of array with the appropriate data and store into returned parameter
convert resistances to temperatures
D:\TFCS Archive\
Found Files
Latest data length
1704153600.00
set time to 1904 date system
Pressure
Level
W_temp
Temp
W_press
W_RH

6 Cold Black Body (CBB_control.llb)

6.1 Overview

The CBB VI communicates with the CBB device in the lab that controls the motors for the flip mirrors and the heaters. It also reads the temperatures from the AC lakeshore unit. Like the telescope simulator, the CBB can be remotely controlled from SCOS.

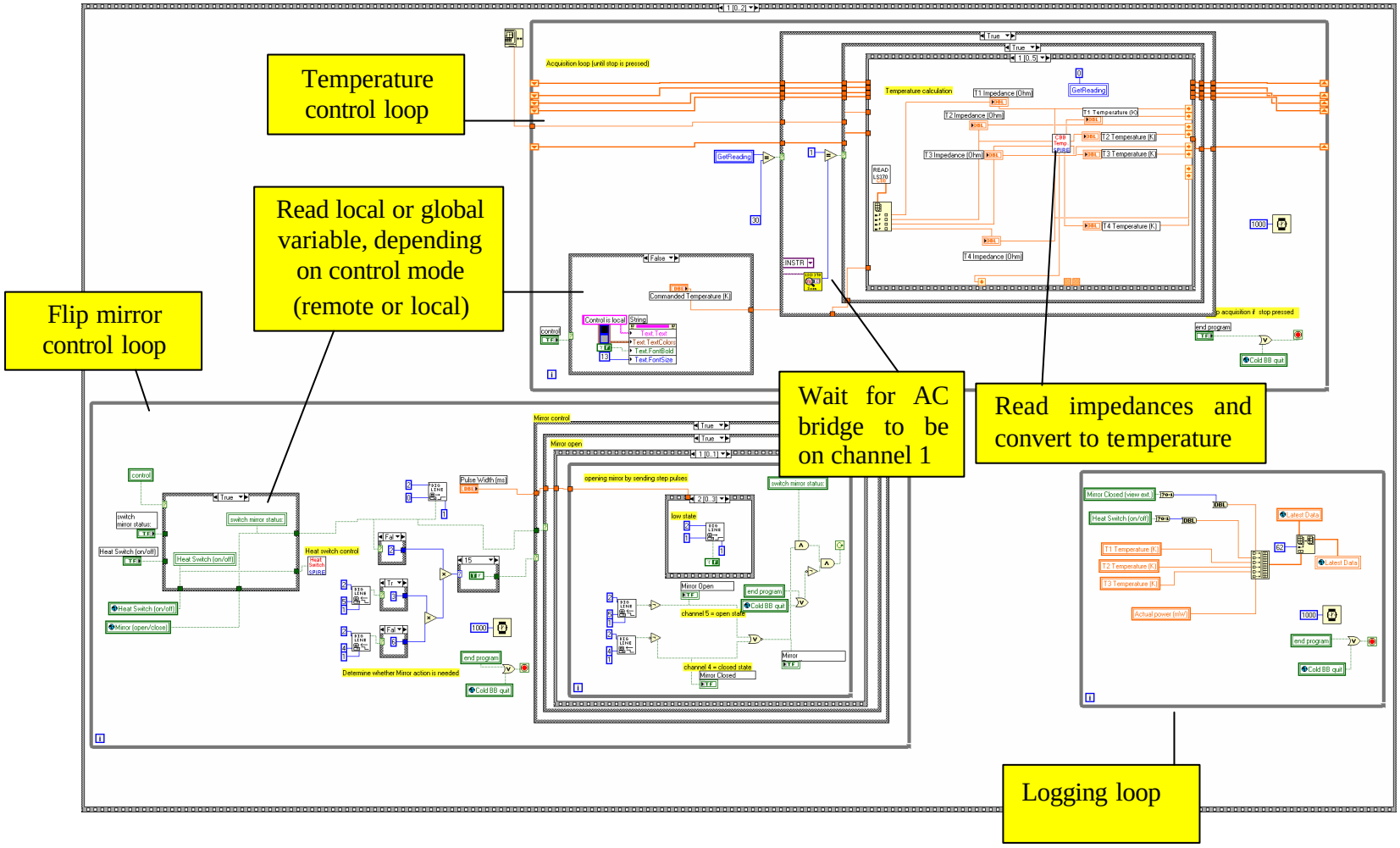
6.2 Front Panel



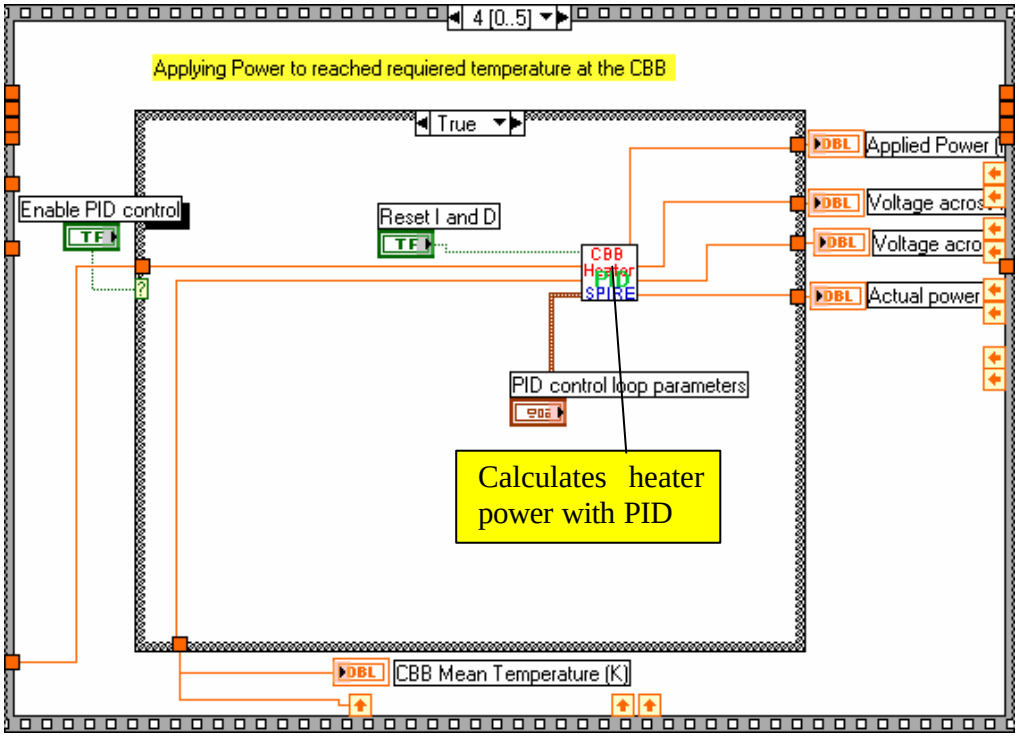
6.3 Block diagram

There are 3 main loops executing simultaneously. The first is for temperature control: it reads the temperatures, computes the CBB mean temperatures, sends commands for applying current to the heaters (with the PID control) and update the graph on the front panel. The second

loop controls the motors for the flip mirror. The third one logs the CBB parameters in the global variable (written to TFCS log files and sent to router).
Note that the program reads the impedances of the temperature sensors. The conversion to Kelvin is done in the software (CBB thermometers.vi)



The following frame is found in the temperature control loop. This is where the power to the heaters is applied. Depending on the state of the "enable PID" button, the calculated power comes either from the PID control loop or from an open loop conversion curve "Temp VS Applied Pwr". The PID calculates the applied power Pwr the following way:



$$e = f(T_{measured}) - f(T_{commanded})$$

$$P = K_p \times e$$

$$I = K_i e \times \Delta t + I_p$$

$$D = \frac{K_p \times (e - e_p)}{\Delta t}$$

$$Pwr = P + I + D$$

where $f()$ if the function "Applied Power VS temp", so e is the error measured in term of applied power, I_p is the previous integral term calculated, e_p is the previous error measured and Δt is the time spent since the last measurement. K_p, K_i and K_d are the proportional, integral and derivative gains, that the user can set on the front panel.

The "reset PID button" resets I_p and e_p to 0.

I found that using just the P and I terms works ok. Kp=6, Ki=0.009 and Kd=0 seem to be appropriate values for this system.

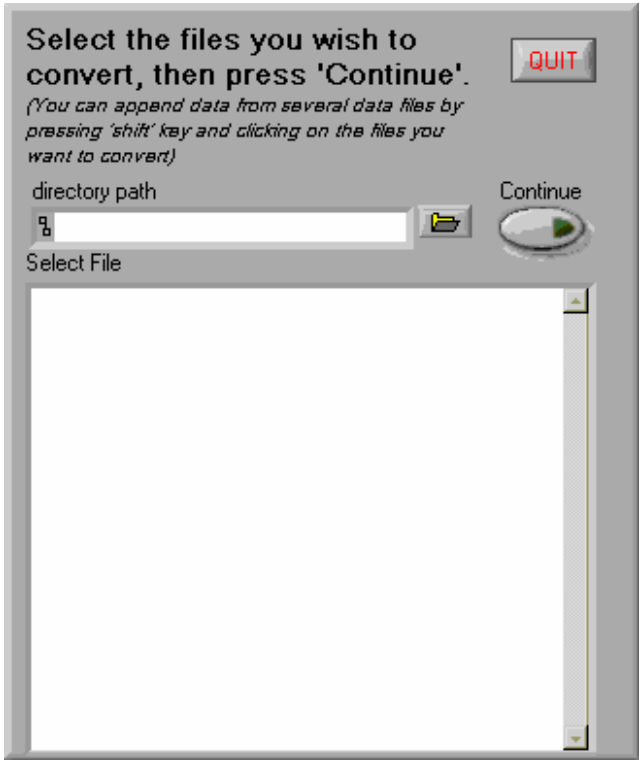
Note that, because the CBB's temperatures sensors are read by the AC bridge, the temperatures are updated only once every 16 channels have been read, which takes about 90 seconds. For the PID control to work properly, the temperature control loop starts, 15 seconds after a measurement, to wait for the AC bridge to have read the CBB thermometers (they are on channel 13 to 16, hence the program waits for the AC bridge to be back on channel 1). Once this is done, the temperature control loop executes (read temperatures, update graph, compute mean temperatures, apply power to heater...)

7 Binary File to Excel (in TFCS-Menubar.llb)

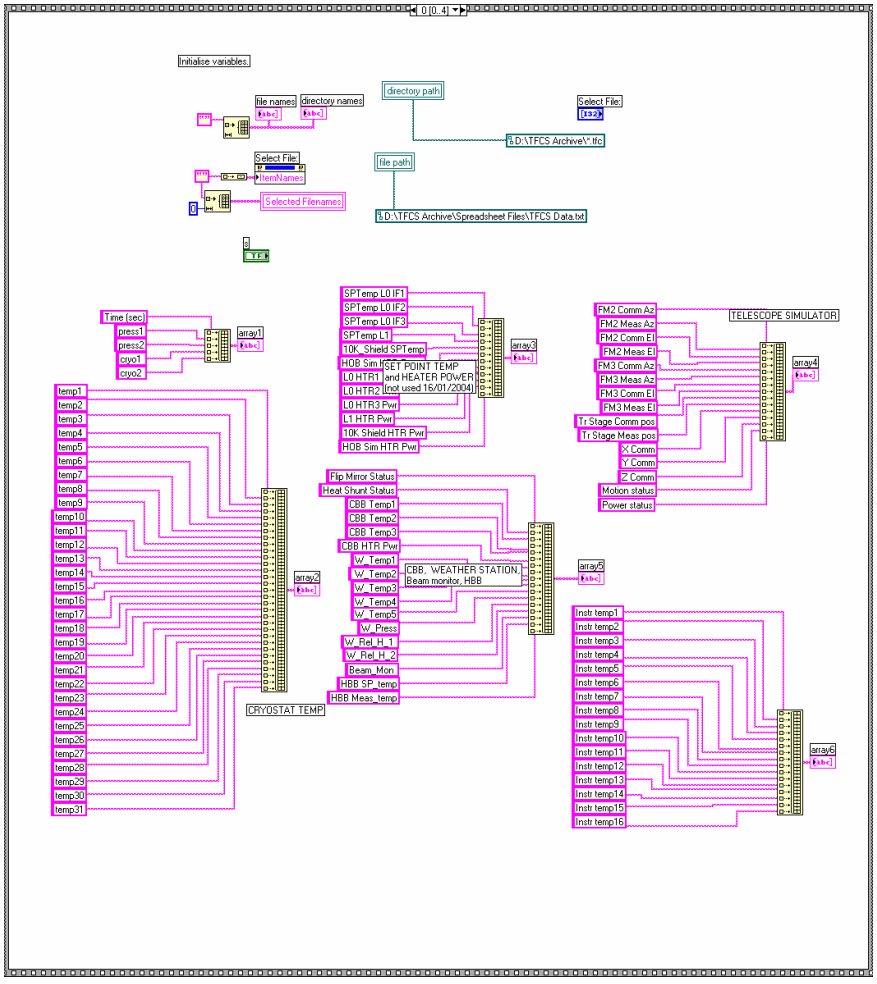
7.1 Overview

This VI reads the tfcs log files and convert them into a spreadsheet that is convenient to open with excel.

7.2 Front Panel



7.3 Block diagram



This first frame initializes some string arrays that will be written at the beginning of the file (first row). For consistency, it must be updated if the TFCS packet content is modified. The different parameters must appear in the same order than in the TFCS packets and log files.

The following frame is the main frame of the program. The selected log files are read, the data are formateed into a 2D array, the temperatures are converted from impedance to Kelvin, and the data are written into the text file (conveniently read with Excel) The user is alos asked if he wants to convert the time into days since 01/01/1900 (system time used by excel) or leave it in seconds since 01/01/1958 (system time used for SPIRE)