WINCAN v. 2.2

A Linux-based Slow Control interface for the AMS TRD Gas System

F. Bucci [a], ***S. Gentile*** [a], ***A. Lebedev*** [b], ***F.R. Spada*** [a,1]

[a] *University "La Sapienza" and INFN, Rome, Italy*

[b] *LNS, MIT, Cambridge, USA*

April 20, 2006

[1] francesca.spada@Roma1.infn.it

Table of Contents

List of Figures

1 Introduction

This note describes the WINCAN program to interact with the gas system of the TRD. The version of WINCAN described here has been developed to operate the engineering version of the gas system on Earth. This means that the communication with the apparatus does not suffer any delay due to the limited bandwidth limit of the link Earth – ISS or its non availability. This allows the immediate feedback of the apparatus to any issued command. This of course is not true when the system is operating in outer space. This version of WINCAN then uses the EPP CAN box as an interface, and works locally, i.e. it stores the data on the same computer on which the program itself resides. This will not be true in the final scheme, where the JMDC will control the system and the data will be stored in common files on remote disks.

WINCAN basically consists in a set of graphical user interfaces to issue low-level commands to the gas system components (listed, together with the slow control electronics boards, in section 2), such as open valves, operate pumps and read sensors.

WINCAN also contains higher level procedures that allow the execution of more complicate operations, such as the procedure to create the gas mixture that must be injected in the TRD. The operations that can be performed with the WINCAN user interfaces are described in section 3.

The package also contains a Root-based monitor program called SENSMON that allows the user to display the status of the system during its operation (section 4).

WINCAN is a Linux-based program that uses the XForms libraries [1] and A. Lebedev libraries [2] for data transmission in AMS data format. It communicates with the Universal Slow Control Module (USCM) via CAN bus using the LeCroy communication protocol. Via the USCM, WINCAN issues commands to the dedicated boards that control the subsystems of the gas apparatus.

2 Gas system layout

The scheme of the Gas System is shown in figure 1. In the box S the gases (xenon and carbon dioxide) are stored and the Xe:CO_2 [4:1] mixture is prepared. The box C contains the pumps that circulate the gas mixture into the TRD and the analyzers that monitor the gas composition. A manifold segment with a TRD element is shown as an example: 41 such modules exist.

Here is a list of the components that the software has to control in the Gas System,

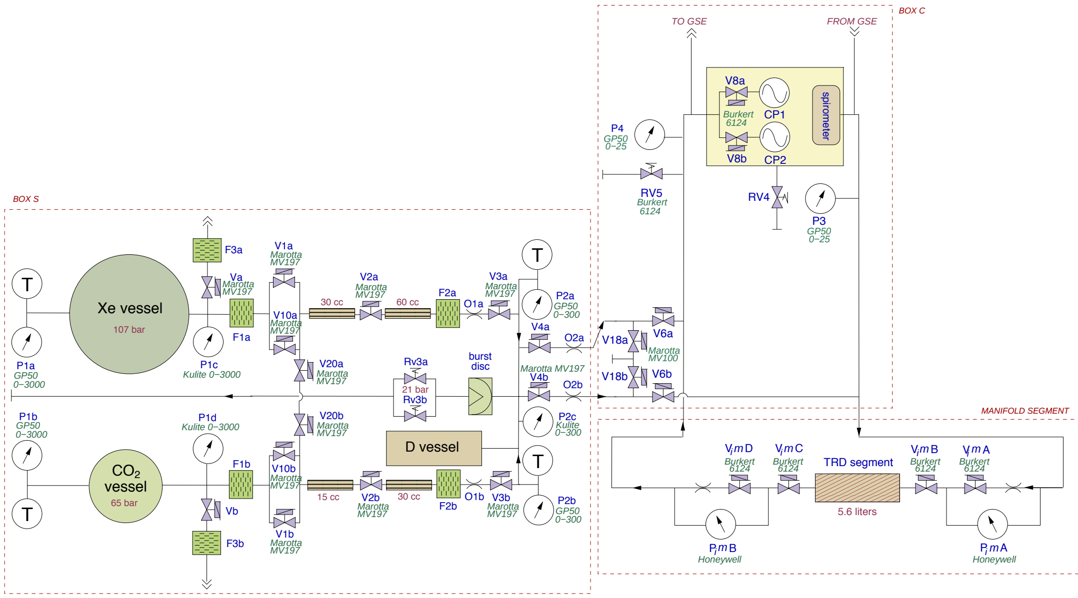


FIGURE 1 – Layout of the TRD gas system.

with the corresponding actions. For a more detailed description, see reference [3].

box S:

- n. 12 Marotta MV197 solenoid valves (V1a, V10a, V1b, V10b, V2a, V3a, V2b, V3b, V20a, V20b, V4a, V4b)
  - enable and disable the valves
  - test the enable status of the valves
  - open the valves for a programmable period ranging from 10 to 65,000 ms
  - test the mechanical status of the valves
- n. 4 GP50 pressure sensors (P1A, P2A, P1B, P2B)
  - read the pressure measurement in volts
  - convert into pressure unit (psia, atm, ...)
- n. 3 Kulite pressure sensors (P1C, P1D, P2C)
  - read the pressure measurement in volts

  - convert into pressure units

- n. 2 heaters
  - turn on the heaters
  - turn off the heaters

box C:

- n. 4 Marotta MV100 solenoid valves (V6a, V6b, V18a, V18b)
  - enable and disable the valves
  - test the enable status of the valves
  - open the valves for a programmable period ranging from 10 to 65,000 ms
  - test the mechanical status of the valves
- n. 2 Burkert flipper valves (V8a, V8b)
  - open the valves
  - close the valves
- n. 2 KNM circulation pumps (CP1, CP2)
  - turn on/off the pumps
  - change the pump speed
- n. 2 GP50 pressure sensors (P3, P4)
  - read the pressure measurement in volts
  - convert into pressure units
- n. 1 Multi Channel Analyzer (MCA) interfaced through RS232 connection
  - select the measurement channel
  - read the raw measurement from the monitor tubes
  - compare with the calibration and give the result
- n. 1 CO_2 analyser interfaced through RS232 connection
  - read the raw measurement

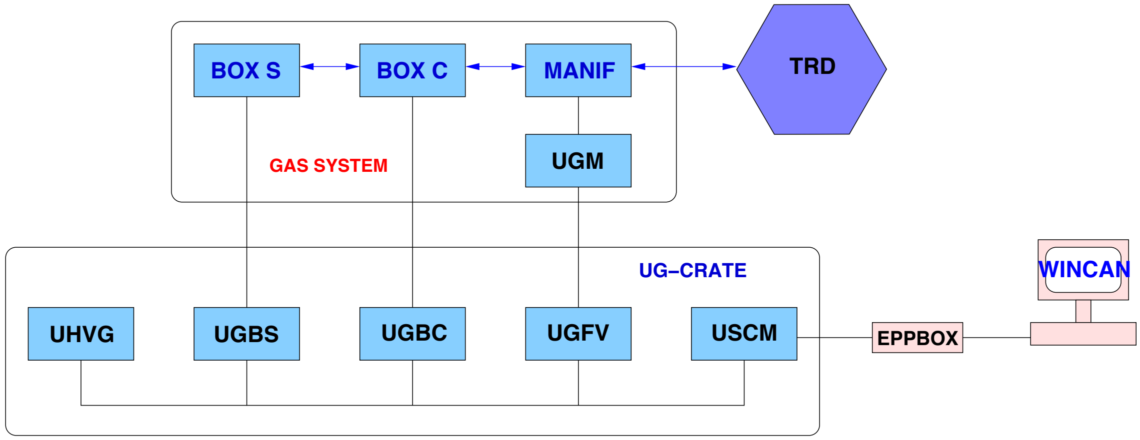


Figure 2 – Layout of the gas system slow control electronics. WINCAN resides on the control PC, which is in communication with the USCM through the EPP CAN box interface. Dedicated boards, UGBS, UGBC, UGFV send commands to the gas subsystems. The UHVG board provides the high voltage needed by the monitor tubes.

  - compare with the calibration and give the result

Manifold:

- n. 164 Burkert flipper valves ($V_i m A$, $V_i m B$, $V_i m C$, $V_i m D$, $m = 1, ..., 8$, $i = 1, ...,$ 5 or 6)
  - open the valves
  - close the valves
- n. 82 Honeywell pressure sensors ($P_i m A$, $P_i m B$, $m = 1, ..., 8$, $i = 1, ..., 5$ or 6)
  - read the pressure measurement in volts
  - convert into pressure units.

Figure 2 shows the layout of the slow control electronics and the correspondence with the gas subsystems. WINCAN communicates with the USCM through the EPP CAN interface. The USCM then communicates with the UGBS, dedicated board for box S, the UGBC, dedicated board for box C, the UGFV, dedicated boards for the manifolds and the UHVG that provides the high voltage needed by the monitor tubes in the box C. The following section describes the WINCAN interfaces that allow the user to interact with the system described above.

3 WINCAN user interfaces

WINCAN contains separate interfaces for the box S, box C, manifolds, multi channel analyzer and heaters. They have some common features: refer e.g. to figures 3, 5, 9 for the fields described in table 1. Every time a command is issued, the program answers with a status message: *command executed OK* (no communication error) , *command executed OK, but...* (there was communication with USCM but the command could not be executed for some other reason, e.g. wrong communication with the other boards), *command execution failed* (no communication with the USCM).

USCM	right-click or left-click the blanks below to set the USCM address
JMDC	dummy, set to 1
BOOT Flash	click to boot USCM from EPROM
Map	click to display a scheme of the gas system section
Quit	click to close the interface

TABLE 1 – Description of the common features in the WINCAN interfaces.

3.1 Box S user interfaces

The box S interface (see figure 3) communicates with the UGBS board to read sensors and operate Marotta valves. The interface contains the features described in table 2. Note that before being opened, the Marotta valves need to be enabled.

The heaters, positioned on the xenon and carbon dioxide tanks, are controlled by two separate interafaces, shown in figure 4. Their features are described in table 3.

3.2 Box C user interfaces

The box C interface (see figure 5) communicates with the UGBC board to read sensors and operate valves and pumps. The interface contains the features described in table 4. Note that differently from the Marotta valves, the Bürkert (or flipper) valves do not need to be enabled before being opened, and the opening time cannot be programmed: they have to be closed with a separate command.

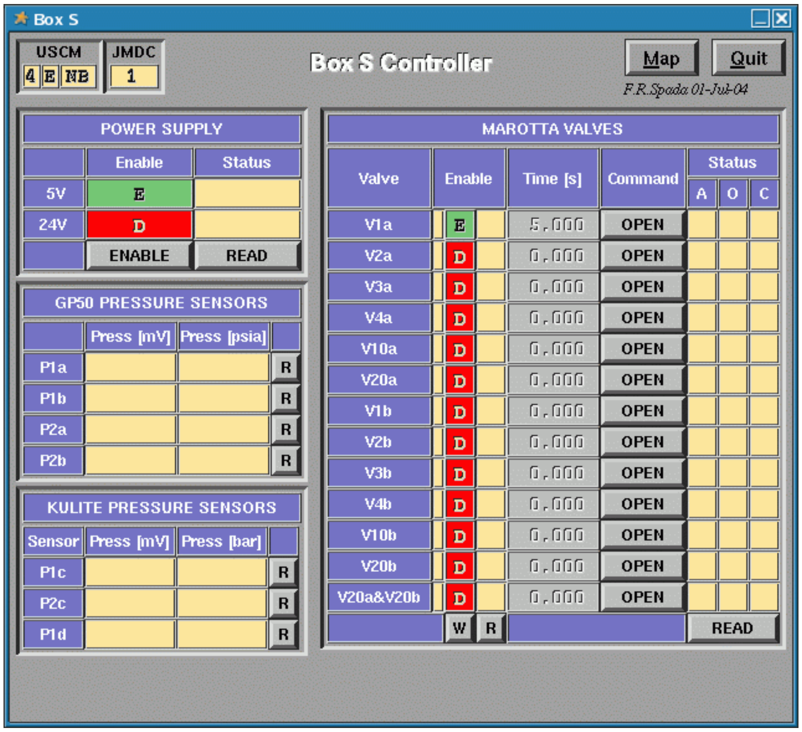
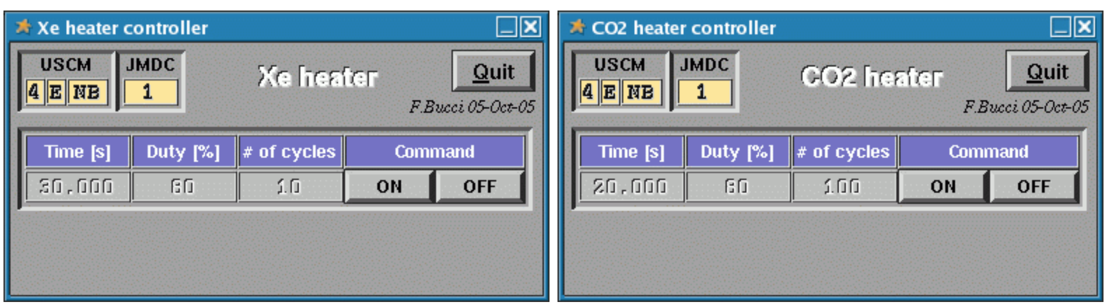


FIGURE 3 – The WINCAN user interface for the box S.
(routine: `boxs.c`)



(a) (b)

FIGURE 4 – The WINCAN user interfaces for the xenon (a) and carbon dioxide (b) heaters interfaces. (routines: `xe_heater.c` and `co2_heater.c`)

Box S	
POWER SUPPLY	
Enable column	switch between *E* and *D* to select the tensions to be enabled.
	Execute the selection with *ENABLE* button
Status column	read back the enable status with the *READ* button
GP50 PRESSURE SENSORS	
Press columns	read the measurement of each sensor in *mV* and *psia*
	with the *R* buttons
KULITE PRESSURE SENSORS	
Press columns	read the measurement of each sensor in *mV* and *psia*
	with the *R* buttons
MAROTTA VALVES	
Enable column	switch between *E* and *D* to select the valves to be enabled.
	Execute the selection with *W* button.
	Read back the selection with the *R* button
Time column	set the opening time (in *s*) for the valve
Command column	open the valve with the *OPEN* button
Status column	check the status of the valve with the *READ* button

Table 2 – Commands of the WINCAN interface for box S.

Heaters	
Time column	set the heating time (in *s*) in one cycle
Cycle	program the duty cycle (% of *Time* actually spent heating)
# of cycles column	set the number of heating cycles to be executed

Table 3 – Commands of the WINCAN interface for heaters.

The Multi Channel Analyzer is controlled through the interface shown in figure 6 and explained in table 5.

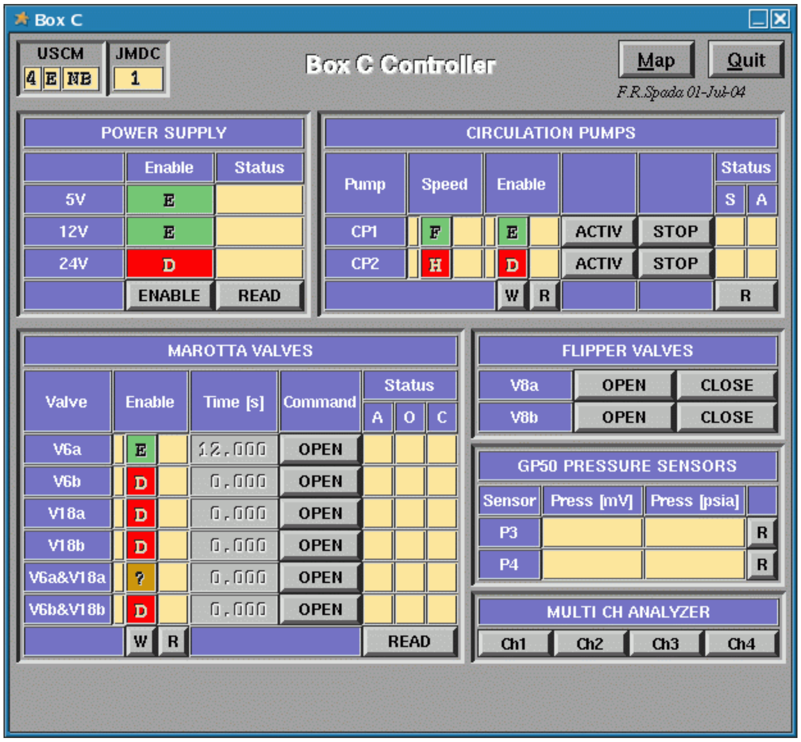


FIGURE 5 – The WINCAN user interface for the box C.
(routine: `boxc.c`)

3.3 Manifold user interface

This interface, shown in figure 7, communicates with the UGFV-AC and UGFV-BD boards that control the 8 manifold modules. All the pressure sensors of a manifold module can be read simultaneously selecting one multiplexer (1 to 8), while one couple of flipper valves, $\mathrm{V}_i m\mathrm{A}$ – $\mathrm{V}_i m\mathrm{C}$, or $\mathrm{V}_i m\mathrm{B}$ – $\mathrm{V}_i m\mathrm{D}$, can be operated at a time. The features are listed in table 6.

Box C	
POWER SUPPLY	
Enable column	switch between *E* and *D* to select the tensions to be enabled.
	Execute the selection with *ENABLE* button
Status column	read back the enable status with the *READ* button
CIRCULATION PUMPS	
Speed column	switch between *H* and *F* to select the pumps speed (half or full).
Enable column	switch between *E* and *D* to select the pump(s) to be enabled.
	Execute the selection with *W* button.
	Read back the settings with the *R* button
Commands:	Start the pump with the *ACTIV* button
	Stop the pump with the *STOP* button
Status column	read back the pump speed and activity with the *R* button
MAROTTA VALVES	
Enable column	switch between *E* and *D* to select the valves to be enabled.
	Execute the selection with *W* button.
	Read back the selection with the *R* button
Time column	set the opening time (in *s*) for the valve
Command column	open the valve with the *OPEN* button
Status column	check the status of the valve with the *READ* button
FLIPPER VALVES	
Commands:	Open the valve with the *OPEN* button
	Close the valve with the *CLOSE* button
GP50 PRESSURE SENSORS	
Press columns	read the measurement of each sensor in *mV* and *psia*
	with the *R* buttons
MULTI CH ANALYZER	
Ch# buttons	select the MCA channel to be read out

Table 4 – Commands of the WINCAN interface for box C.

Multichannel Analyzer	
Run time column	set the data acquisition time (in *s*) in one cycle
Interval column	set the time (in *s*) between one acquisition and the following one
Acquisitions column	set the number of data acquisitions

TABLE 5 – Commands of the WINCAN interface for the MCA.

FIGURE 6 – The WINCAN user interface for the Multi Channel Analyzer.
(routine: `mca.c`)

3.4 High voltage user interface

This interface communicates with the UHVG board that controls the high voltage supply to the monitor tubes. It is shown in figure 8 and described in table 7.

3.5 Dallas sensors user interface

The USCM can receive data from Dallas temperature sensors from 8 different buses. Moreover, it has a built-in sensor on bus number 9. The interface for Dallas sensors control is shown in figure 9 and its features are described in table 8.

3.6 Mixing loop user interface

With this interface (shown in figure 10) the user can control the gas mixing procedure. It can transmit instructions to empty the mixing vessel and inject Argon and CO_2 shots

Manifolds	
POWER SUPPLY	
Enable column	switch between *E* and *D* to select the tensions to be enabled.
	Execute the selection with *ENABLE* button
Status column	read back the enable status with the *READ* button
UGFV BOARD	
Switch:	*AC* or *BD* select the valve couple $V_i m$A – $V_i m$C or $V_i m$B – $V_i m$D
MODULE	
Switch:	1 to 8 select the module (the "*m*")
VALVE	
Switch:	1 to 5 (6) select the valves (the "*i*") in the module
MULTIPLEXERS	
Switch:	1 to 8 select the module
Command:	*SELECT* button confirms the selection
HONEYWELL PRESSURE SENSORS	
Command:	read all the pressure sensors of the module with the *READ* button

Table 6 – Commands of the WINCAN interface for the manifolds.

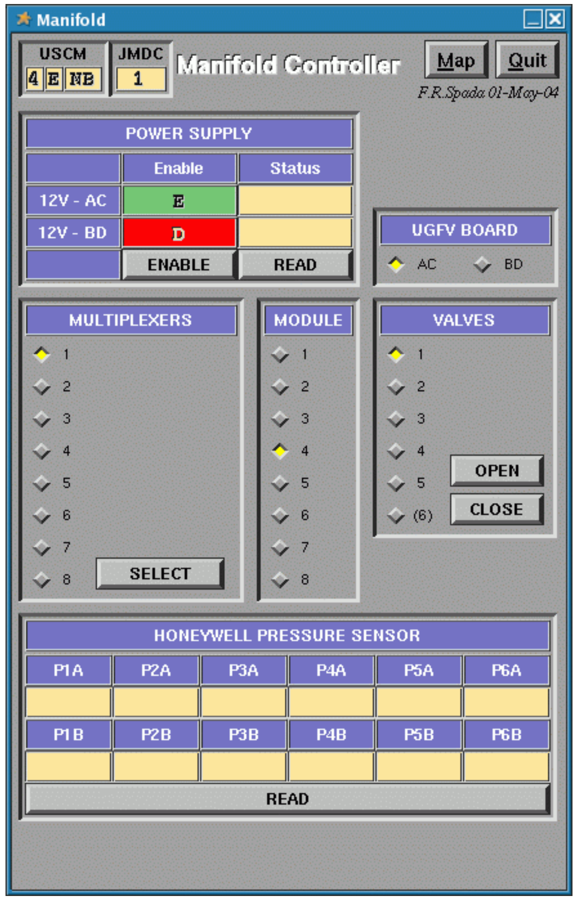


FIGURE 7 – The WINCAN user interface for the manifolds.
(routine: `segments.c`)

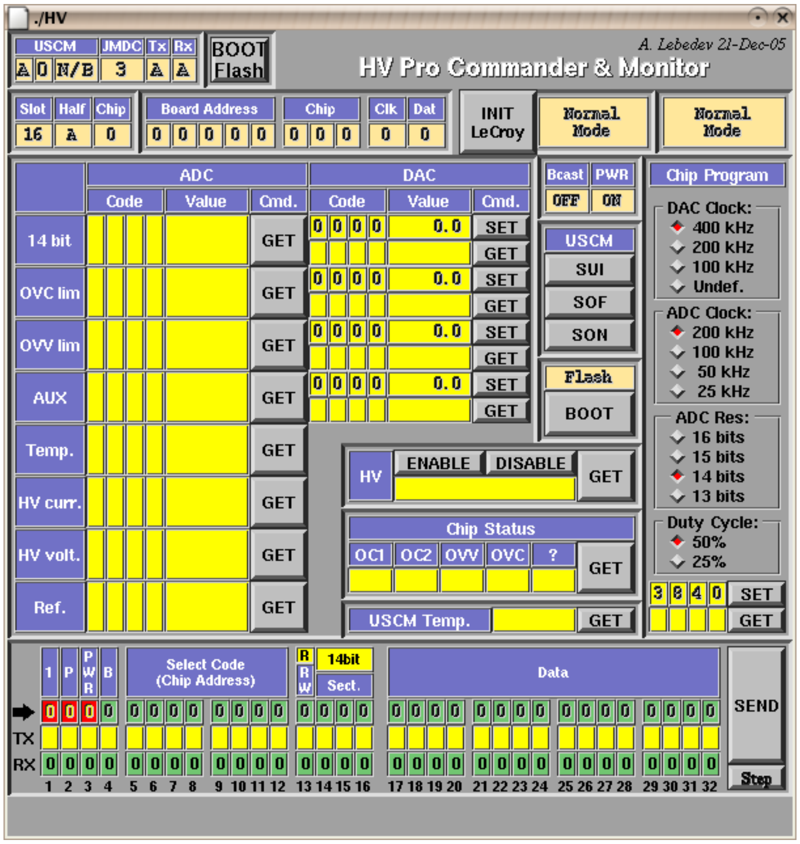


FIGURE 8 – The WINCAN user interface for the high voltage control.
(routine: `HV.c`)

	High Voltage
INIT LeCroy	
Slot	for U-crate only
Half	for U-crate only
Chip	for U-crate only
Board address	for U-crate only, set to 0
Chip	0 to 6, switch between MHV-100 LeCroy chips
Clk	0 to 7, LeCroy clock bus
Dat	0 to 7, LeCroy data bus
Command:	initialize the LeCroy protocol in the bus specified with the *INIT LeCroy* button
ADC	
Command:	select one of 8 ADCs, reads the code and converts it to:
14 bit	HV value
OVC lim	over-current limit value
OVV lim	over-voltage limit value
AUX	not used
Temp	temperature
HV curr	current value
HV volt	voltage value
Ref	reference
DAC	
Command:	*GET/SET* the *Code* and corresponding *Value* for:
14 bit	HV value
OVC lim	over-current limit value
OVV lim	over-voltage limit value
AUX	not used

HV	
Command:	*ENABLE/DISABLE* the HV from the selected chip
Command:	*GET* the HV status
Chip status	
Command:	use *GET* button to check:
OC1	1st overcurrent level
OC2	2nd overcurrent level
OVV	1 = over-voltage, 0 = no over-voltage
OVC	1 = over-current, 0 = no over-current
Chip program	
DAC clock	default: 400 kHz
ADC clock	default: 200 kHz
ADC resolution	default: 14 bit
Duty cycle	default: 50%
Command:	use *SET* button to set the chosen values
Command:	use *GET* button to get the current values

Table 7 – Commands of the WINCAN interface for the high voltage control.

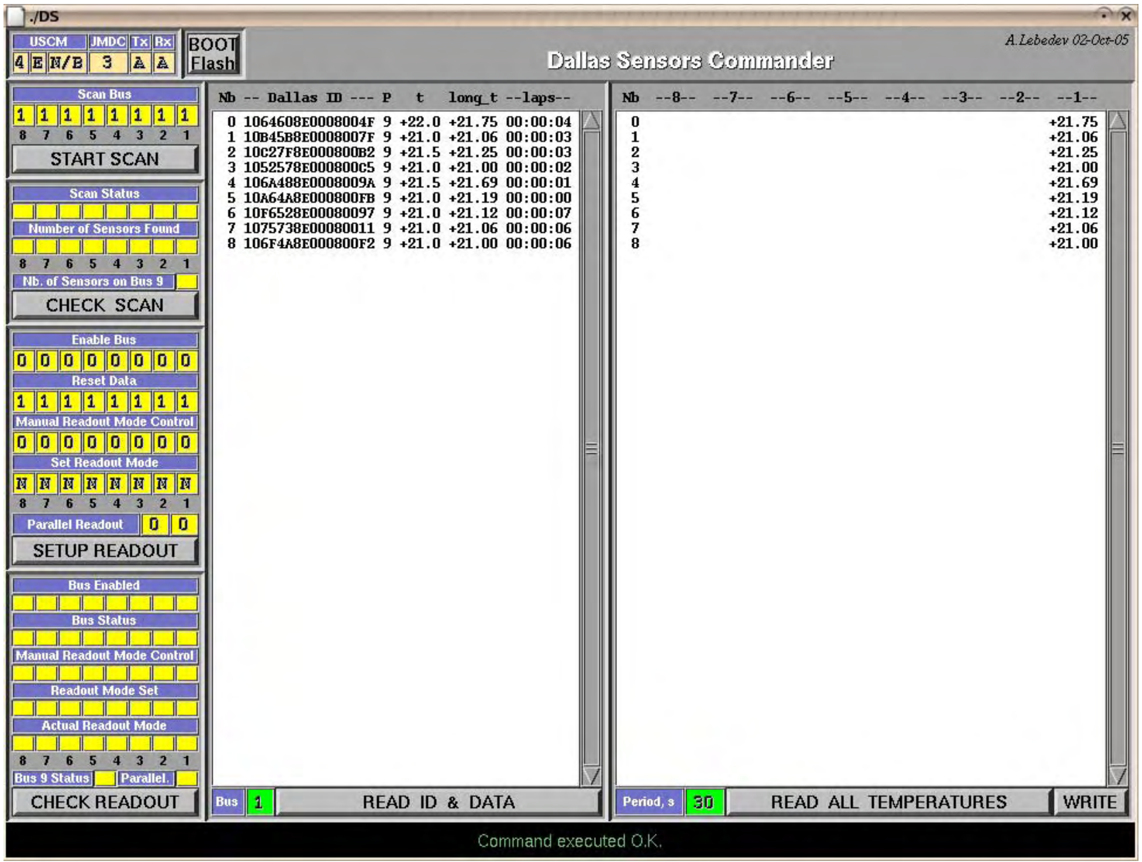


Figure 9 – The WINCAN user interface for the Dallas temperature sensors.
(routine: `DS.c`)

	Dallas
START SCAN	click to scan selected buses (1 = selected)
CHECK SCAN	click to check:
Scan status	0 = completed, 1 = in progress
Number of sensors found	sensors on each scanned bus
SETUP READOUT	click to set:
Enable bus	0 = not enabled, 1 = enabled
Reset data	in a bus (0 = do not reset, 1 = reset)
Manual readout control mode	set manual (0 = do not set, 1 = set)
Set readout mode	N = normal, P = parasitic (does not use VCC line)
Parallel readout	read several Dallas sensors in parallel
CHECK READOUT	click command to check
Buses enabled	0 = not enabled, 1 = enabled
Bus status	0 = not reset, 1 = reset
Manual readout control mode	0 = do not set, 1 = set
Readout mode set	N = normal, P = parasitic (does not use VCC line)
Actual readout mode	N = normal, P = parasitic (does not use VCC line)
READ ID & DATA	readout the sensors on the selected *Bus*
READ ALL TEMPERATURES	
	left-click to readout all the sensors on the scanned buses
	right-click to readout continuously (set *Period*)
WRITE	store temperatures in a log file

Table 8 – Commands of the WINCAN interface for the Dallas sensors control.

directly in the vessel: the intermediate operations are done authomatically. The sequence of operations that determine the ground state of the system has not yet been defined and implementented. For a detailed description see table 9.

FIGURE 10 – The WINCAN mixing procedure interface.
(routine: `mixloop.c`)

4 SENSMON, the online monitor program

WINCAN contains the routine *sensmon*, i.e. the first part of the monitor program. This routine must be run in batch mode during the normal operation of the Gas System. When

Mixing Loop	
MIXING LOOP	
Empty Mix Volume	click to empty the mixing vessel
[open the V4A and V18A, default = 60 s]	
Set Ground State Xe	dummy
Set Ground State CO2	dummy
CO2 Fill	inject carbon dioxide
[1st parameter: n. of shots, default = 2]	
[2nd parameter: opening time of V1A, default = 2 s]	
[3rd parameter: opening time of V2A, default = 4 s]	
[4th parameter: opening time of V3A, default = 60 s]	
Argon Fill	inject Argon
[1st parameter: n. of shots, default = 3]	
[2nd parameter: opening time of V1A, default = 1 s]	
[3rd parameter: opening time of V2A, default = 2 s]	
[4th parameter: opening time of V3A, default = 30 s]	
PRESSURE	
READ	click to read pressures in all GP50 sensors

TABLE 9 – Commands of the WINCAN interface for the mixing loop control.

it is executed, the routine continuously monitors

- pressures from the GP50 pressure sensors (box S, C);
- pressures from the Honeywell pressure sensors (manifolds);
- status of the Marotta MV100 valves (box C);
- temperatures from the Dallas sensors and the corresponding values are written in a log file which is updated every n seconds, where n can be chosen by the user.

The log files can be accessed also offline, if needed.

The second part of the monitor program consists of the routine *sensvisu*, that is based on the Root graphical user interface facilities to allows the graphical display of the system status. The input of *sensvisu* consists of the log files that *sensmon* gives as output. Once every m seconds, where again m can be chosen by the user, *sensvisu* reads the log file (which in the meantime may have been updated by *sensmon*) and visualizes its contents grapically. The flow diagram of the monitor program is shown in figure 11.

The monitor program is started and controlled interactively from a graphical user interface, shown in figure 12. For the sake of easy visualization, the plots concerning pressures, temperature and system status can be displayed separately selecting one or more sets with the corresponding check buttons on the user interface. The Print button allows to print a postscript version of the plots currently displayed. Some examples of how sensmon displays the status of the system are shown in figures 9 and 10.

5 Code availability

The code can be downloaded from the web page http://www.cern.ch/spada/AMS/.

The WINCAN code only needs the XForms [1] libraries to be installed to be compiled and run. The SENSVISU code, i.e. the monitor program, also needs Root [4] to be run.

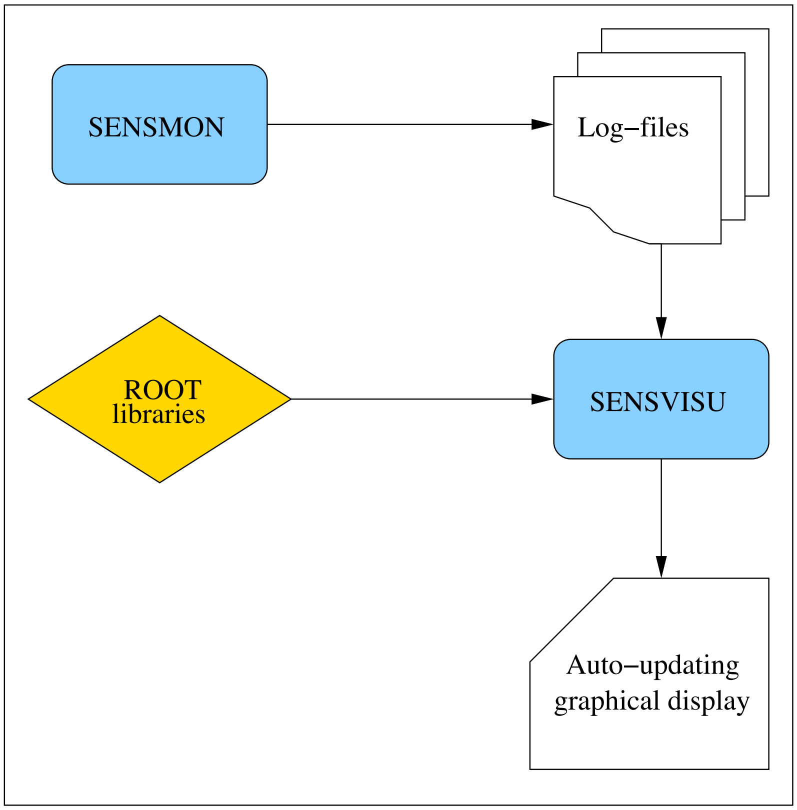


Figure 11 – Scheme of the SENSMON monitor program.

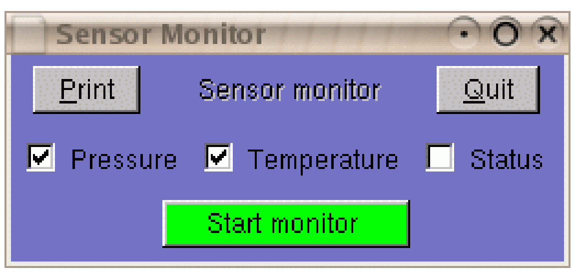


Figure 12 – The SENSMON user interface.

References

[1] http://ams.cern.ch/AMS/Lebedev/XForms/.

[2] http://ams.cern.ch/AMS/Lebedev/My Software/.

[3] A. Bartoloni, B. Borgia, F. Bucci, S. Gentile and F.R. Spada, "*AMS-02 TRD gas Slow-control system specifications*", AMS internal note.

[4] http://root.cern.ch/.