SAM AOM software description

Prepared by: A.Tokovinin, M.Martinez
Last update: July 22, 2011
File: soar/software/aom/aom-sw.odt

This document describes the functionality of the LabView software that controls mechanisms of the SAM AO module – the AOM SW. It is an engineering software created by M. Martinez which is normally not used in the SAM operation, being controlled by the user software. The AOM SW interacts with the user SW by receiving commands, executing them and sending the responses.

1. Location and startup

The AOM SW is installed on the SAM AOM computer (located in the rack on the SOAR platform), IP 139.229.15.32. The AOM computer is a PC running under Linux Fedora core 9. The SW is located in the `/user/aom` directory and is run by the user `aom`. It is normally accessed by the VNC in the work space 4, leaving other work spaces for other SAM SW components that run on the same PC (LGS, LCH, ICSoft).

To get access to the AOM SW, connect by VNC to **139.229.15.32:4**. To start, click once on the icon “SAM aom” in the graphic menu (in the lower-right corner of the screen). This evokes the script `Start_aom` in the home directory. Upon startup, three windows appear:

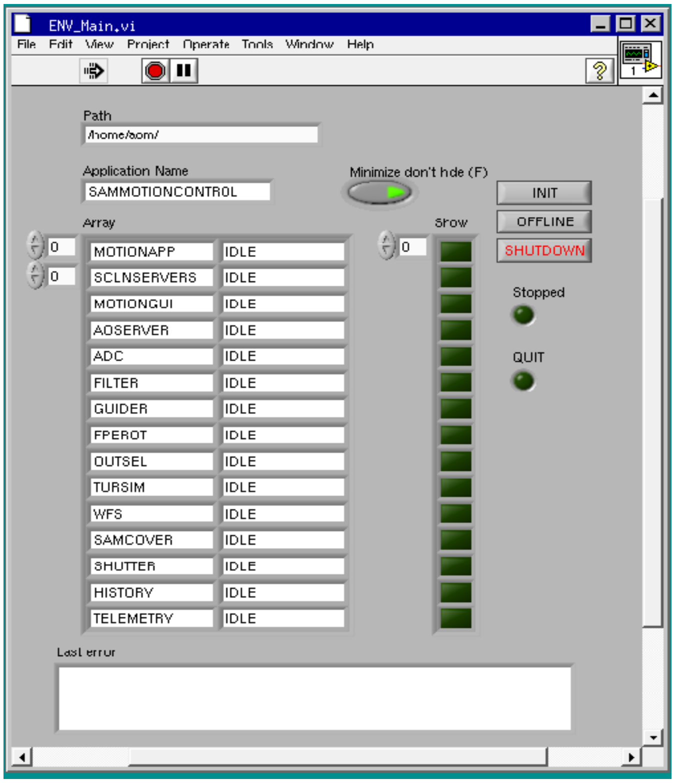
- MOTIONGUI_Main.vi (small window with *Exit* switch)
- Env_Main.vi (grey window)
- MOTIONGUI_Logic.vi (main GUI)

Only the last window is used for control (this is the actual GUI), the first two are for technical use and can be minimized (but don't close them!). The button *Exit* in the GUI should be used for exiting the AOM SW.

2. Structure of the program

The structure of the AOM SW follows the standard adopted at SOAR. The main application is *Env_Main.vi*. It calls other LabView modules, or devices (their code is in the `/user/aom/SAMMOTIONCONTROL/modules/*` directories).

Figure 1 shows the corresponding window. It contains the list of 17 devices (use the arrow-box on the left to scroll through the whole list) and their status. The status can take four possible values: `IDLE, ACTIVE, OFFLINE, SHUTDOWN`. The buttons *Show* are used to open specific GUIs for each device (normally they are not opened). The three buttons on the right, *INIT, OFFLINE,* and *SHUTDOWN*, serve to control the whole application. By going Offline and Init, we can refresh the configuration of the mechanisms (see below).
The GUI (which itself is a device) interacts with other devices by sending commands and processing responses.



Path: /home/aom/

Application Name: SAMMOTIONCONTROL

Array

MOTIONAPP	IDLE
SCLNSERVERS	IDLE
MOTIONGUI	IDLE
AOSERVER	IDLE
ADC	IDLE
FILTER	IDLE
GUIDER	IDLE
FPEROT	IDLE
OUTSEL	IDLE
TURSIM	IDLE
WFS	IDLE
SAMCOVER	IDLE
SHUTTER	IDLE
HISTORY	IDLE
TELEMETRY	IDLE

Fig. 1. The ENV_Main.vi window.

3. Interaction of the AOM SW with hardware and other software

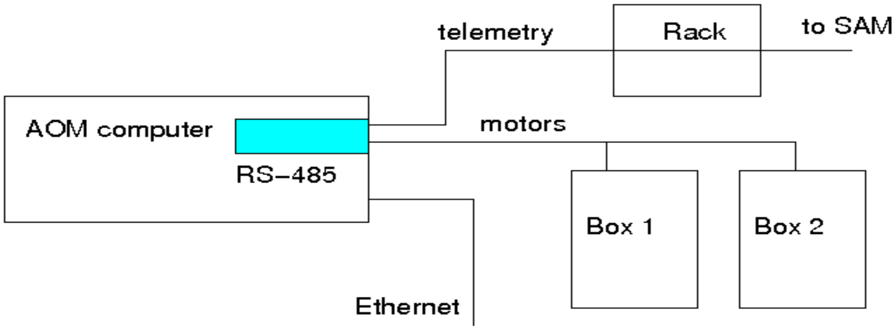


Fig. 2. External connections of the AOM SW.

The AOM SW controls motors and other devices via two RS-485 lines through a serial communication board installed in the AOM computer (another such board is occupied by the LGS motors and laser control). This board uses a generic Linux driver. One RS-485 line is used for telemetry (it goes through the SAM rack to the SAM AOM), another line is connected to the two electronic boxes on the AOM. Each line is actually a bus connected to multiple devices (controllers), recognized by their unique ID numbers. The AOM computer acts as a "master", sending requests to specific devices and getting their responses. A faulty controller can block the communication with all other controllers by occupying the bus. The communication protocol (set of commands and responses) depends on the type of the controller. Four distinct protocols are used by the AOM SW:

- QuickSilver motor controllers, see *http://www.quiksilver.com*
- Allmotion motor controllers, see *http://www.allmotion.com*
- In-house board for servo/DC motors and dimmers (see a document by M. Martinez [3])
- ADAM modules for telemetry, see *http://www.advantech.com*

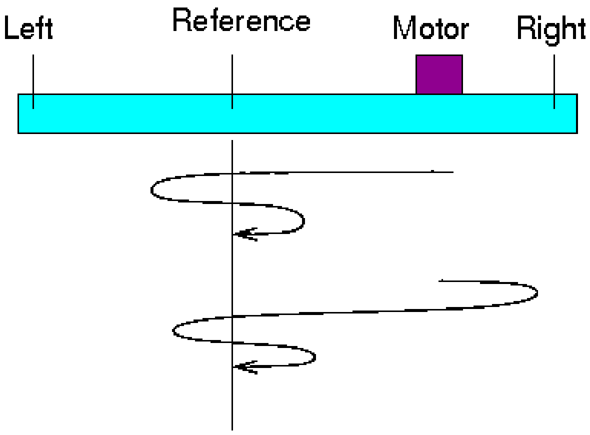


Fig. 3. Schematic diagram of a linear stage with limit switches and reference (or fiducial) sensor. The curves depict motor motion during initialization when the search starts by moving left (reach reference first) or right (reach the limit switch first, reverse motion, then find the reference).

Let us consider a typical case of controlling a linear translation stage with a smart motor. Its position is determined by the incremental encoder which needs to be initialized first. The stage has two limit switches and a reference sensor (also called fiducial) which defines the absolute position (Fig. 3). During initialization, the motor moves in one direction. If it detects the reference sensor, the motor is commanded to go back and to approach the reference again with a slow speed, always from the same direction. If, instead, the limit switch is reached before detecting the reference, the motion is reversed and the search continues. In the case when the initialization fails, a timeout error is reported.

Once the motor is initialized, it can be moved to any desired position, within allowed limits. The speed of motion and acceleration are defined by the configuration file and passed to the motor controller. The positions are measured in absolute units (e.g. millimeters) which are translated into internal controller commands by means of configuration parameters (screw pitch, number of encoder steps per revolution, offset). The state of the mechanism reported in the GUI can be either *ACTIVE* (while moving) or *DONE*. The state is *NOT INITILALIZED* until the mechanism is initialized.

Table 1 lists the mechanisms controlled by the AOM SW and corresponding devices. The column N contains the number of mechanisms, “QS SM” and “PI SM” denote smart motors from Quick Silver and Physik Instrumente, respectively.

The AOM SW communicates with other SAM SW modules by the Ethernet, though sockets. The IP addresses of relevant servers and clients are specified in the configuration files. Actually the AOM SW communicates only with the ICSoft and SAMI (communication with the SOAR TCS is defined, but not used). During normal operation, the SAM AOM is commanded by the ICSoft, while the filter wheel and the shutter are commanded by SAMI. Those commands are passed to the AOM SW which actually executes them.

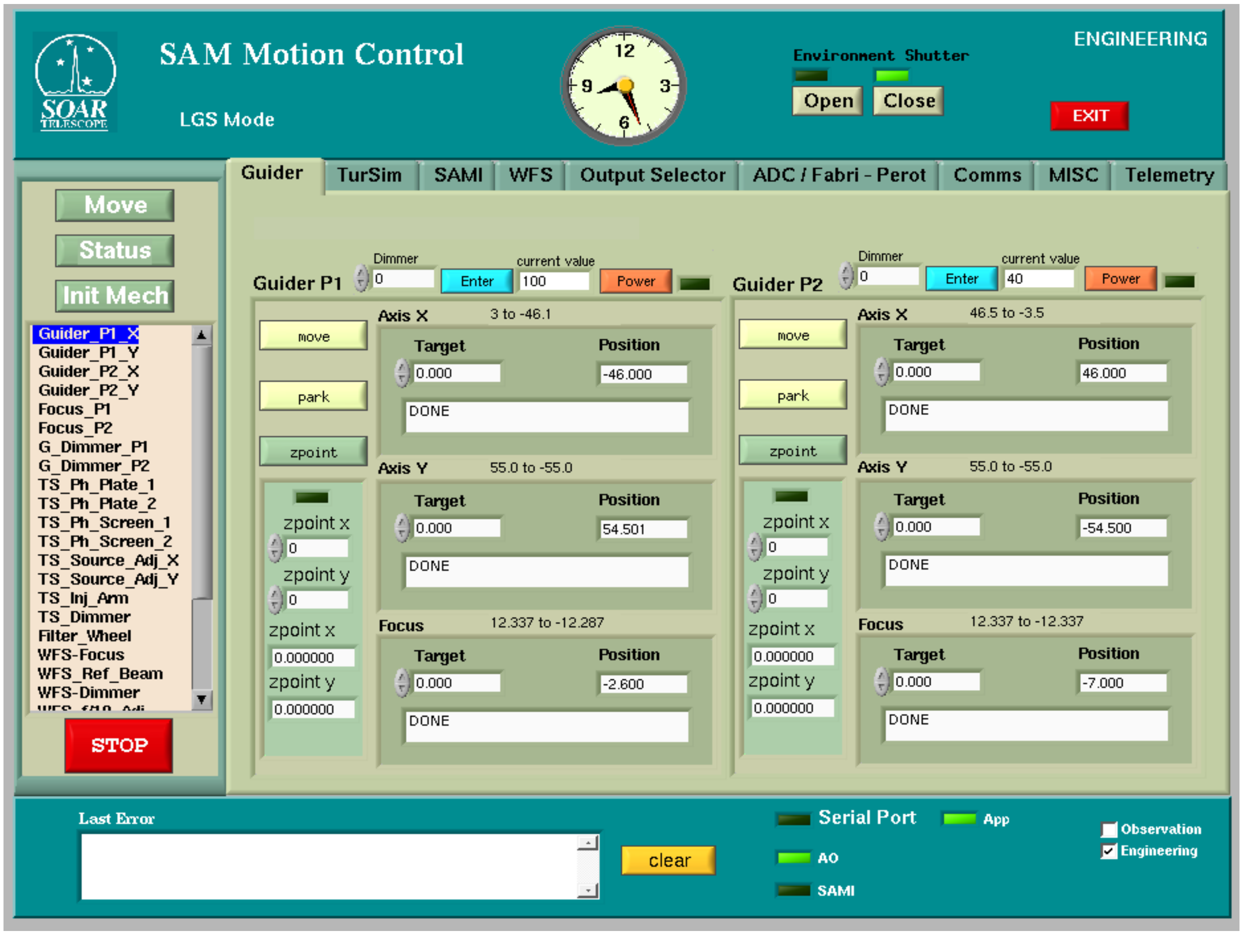
Table 1. **List of AOM devices and controlled mechanisms**

Device	Mechanism(s)	N	Type	What's controlled
GUIDER	X1,X2, Y1, Y2	4	QS SM, linear translation	Position
	Z1, Z2	2	PI SM, linear translation	Position
	Artificial star	2	Dimmer (laser diodes)	Current, on/off
FILTER	Filter wheel	1	DC motor, rotation	7 positions
SHUTTER	Shutter	1	QS SM	open/close
TURSIM	Arm in/out	1	Servo motor	IN/OUT, position
	Phase screen insert	2	Servo motor	IN/OUT
	Phase screen rotation	2	DC motor, rotation	Speed, on/off
	Source X,Y adjust.	2	DC motor, rotation	Speed, on/off, jogs
	Light source	1	Dimmer (LED)	Current, on/off
OUTSEL	Output selector	1	QS SM, linear translation	IN/OUT, position
FPEROT	F-P insertion	1	QS SM, linear translation	IN/OUT
ADC	ADC insertion	1	QS SM, linear translation	IN/OUT
	Prism rotation	2	DC motor	Angle
SAMCOVER	Env. shutter	1	DC motor	open/close
WFS	Focus stage	1	QS SM, linear translation	Position
	F/10 stage	1	PI SM, linear translation	Position
	Acq./reference	1	PI SM, linear translation	IN/OUT, position
	Reference light	1	Dimmer (LED)	Current, on/off
TELEMETRY	WFS Prosilica cam.	1	Power 12V	on/off
	Pockels cell HV	1	Serial control of HV power supply	on/off
	APD1, APD2	2	Power 30V, overlight	on/off, reset

4. The user interface (GUI) and operation

4.1 General description

The screen-shot of the AOM GUI is shown in Fig. 4. The *upper part* contains the window title, clock, control of the SAM environmental shutter, and the *Exit* button. The *left part* is a list (menu) of mechanisms. After selecting a mechanism, it can be initialized or moved to some position with the buttons *InitMech* and *Move*. The motion can be stopped at any moment by the red button *Stop*. The button *Status* refreshes immediately the status of the selected mechanism (otherwise, the status field is refreshed with the polling cycle or upon completion of the current action).



Guider P1

Dimmer 0 | current value 100

Axis X 3 to -46.1

Target 0.000 | Position -46.000

DONE

Axis Y 55.0 to -55.0

Target 0.000 | Position 54.501

DONE

Focus 12.337 to -12.287

Target 0.000 | Position -2.600

DONE

zpoint x 0

zpoint y 0

zpoint x 0.000000

zpoint y 0.000000

Guider P2

Dimmer 0 | current value 40

Axis X 46.5 to -3.5

Target 0.000 | Position 46.000

DONE

Axis Y 55.0 to -55.0

Target 0.000 | Position -54.500

DONE

Focus 12.337 to -12.337

Target 0.000 | Position -7.000

DONE

zpoint x 0

zpoint y 0

zpoint x 0.000000

zpoint y 0.000000

□ Observation

☑ Engineering

Fig. 4. AOM GUI with the guider tab opened.

The *central part* of the GUI contains several tabs (Guider, TurSim, SAMI, WFS, Output Selector, ADC/Fabry-Perot, Comms, Misc, and Telemetry). Each tab corresponds to a group of functionally related motors and dimmers (they may or may not belong in the same device). After selecting a motor in the left-hand menu, the corresponding tab opens automatically. Conversely, opening the tab selects the first relevant mechanism in the left-hand menu.

The *lower part* of the GUI contains a space for error messages with a *Clear* button, indicators of serial communication and connection to other SW modules (AO means ICSoft), and the check-box to select between observation and engineering modes (the mode is also displayed in the upper-right corner of the GUI). In the observation mode, some controls are disabled.

4.2 Guiders

There are two guide probes in SAM, GP1 and GP2. Each probe has three controlled motions X,Y,Z and an artificial (fiducial) light source (laser diode connected by fiber). The GUI (Fig. 4) shows the current position of each motor and its status (DONE or ACTIVE). To move GP1 in X, we enter the desired position (in mm) in the *Target* field of axis X and press the button *Move* in the left part, after selecting the Guider_P1_X motor in the left menu. If we want to move GP1 in X,Y,Z simultaneously, the three target positions must be entered and the button *move* in the guider tab should be pressed instead. The button *park* moves the GP1 or GP2 out of the field, to a pre-defined parking position.

Unlike the remaining motors, the guiders XY motions may have additional offsets, which is in the case of SAM quite confusing (this feature is inherited from other guider applications). The offsets can be introduced by typing values in the fields *zpoint x* and *zpoint y* of each guider (or use arrows) and pressing the button *zpoint*. The offsets are added to the target positions (e.g. the probe moves to +2mm if we send it to +3mm but have an offset -1mm). However, the values reported in the position field are absolute, independent of the offsets. The current offsets are “remembered” in the configuration file. **Always set all offsets to zero!** The functions *park* and *zpoint* are not used in the normal SAM operation (instead, their equivalents exist in the ICSoft).

Before moving the guiders, their motors must be initialized. This is different from other mechanisms in that all 6 motors are initialized sequentially by one command. This action starts by pressing the *InitMech* button after selecting *any* guider motor. The initialization sequence is: X1, X2, Y1, Y2, Z1, Z2. The motion in Z is slow and takes some time to initialize. The allowed motion range in each axis is displayed in the GUI. All motion parameters are determined by the configuration file (see below). While the motion is in progress, further *move* commands generate an error message *Command rejected on state active*.

Another particularity of the guiders is the need to avoid collision between the probes. This is done by the AOM SW (certain motions are forbidden). The collision condition is determined by calculating the distance between the probes in all three coordinates. Actually, only the distance in the XY plane matters, but this more restrictive condition is not currently implemented. There is also a physical anti-collision system which stops the motors in the case when the SW protection fails.

The light sources in each probe can be switched on and off and controlled in intensity on a scale from 0 to 100 percent, with 1% step. Note that zero intensity may still produce some light, so the dimmer power should be turned off when the light source is not needed.

4.3 TurSim

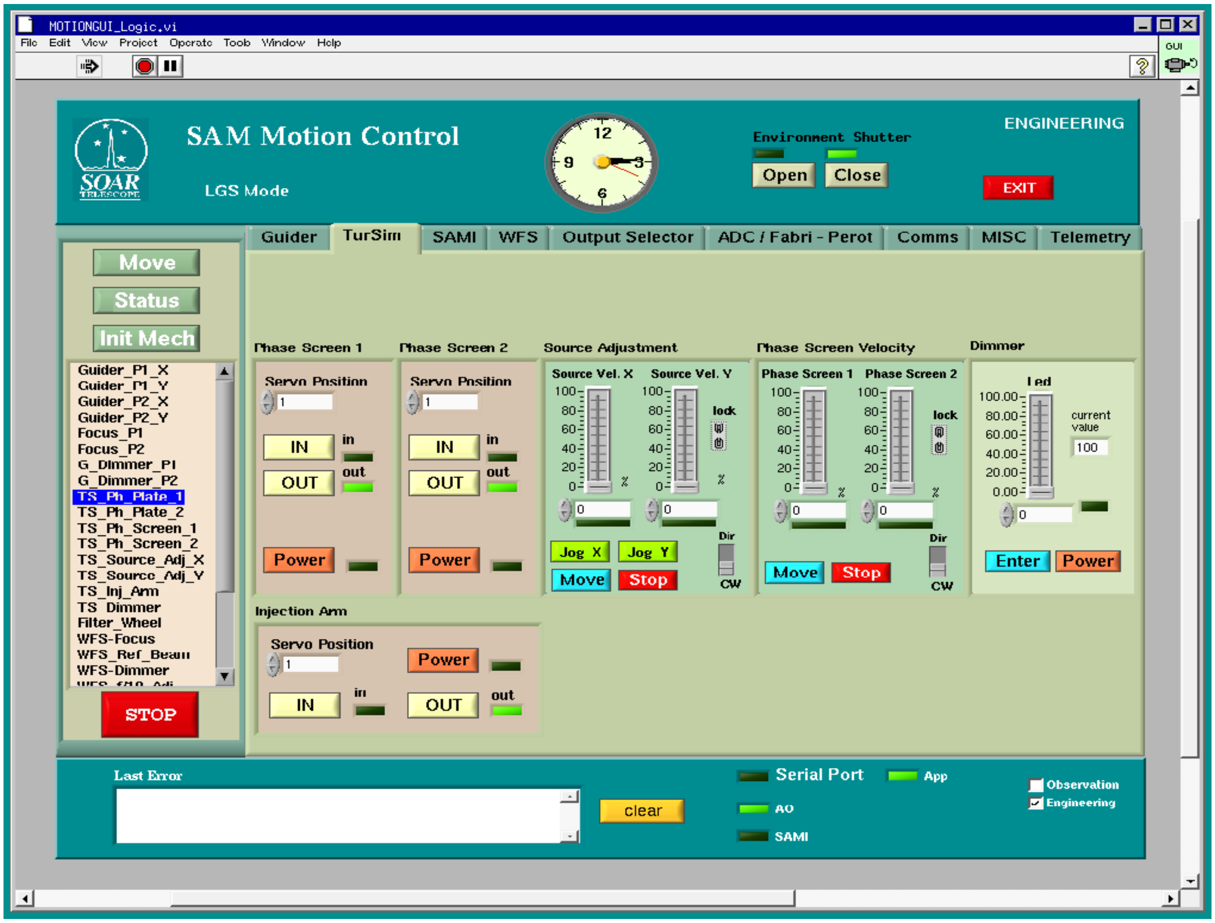


Fig. 5. AOM GUI with TurSim tab opened.

The TurSim tab (Fig. 5) controls motors and dimmers of turbulence simulator. Those motors do not need initialization (they are either servo or DC motors). The functions are grouped in several panels. The two left panels allow us to insert or remove the two phase screens (individually or both) by means of the buttons *IN* and *OUT*. The power to the servo motors is applied only during those motions, to avoid servo jitter. The button *Power* is therefore redundant, but it is functional nevertheless. The position and state of motor power are indicated by the green lights. We can move the motor to some intermediate position (in the engineering mode only: enter the desired Servo position from 0 to 100, select TS_Ph_Screen_1 or ..._2 in the left menu and press *Move*).

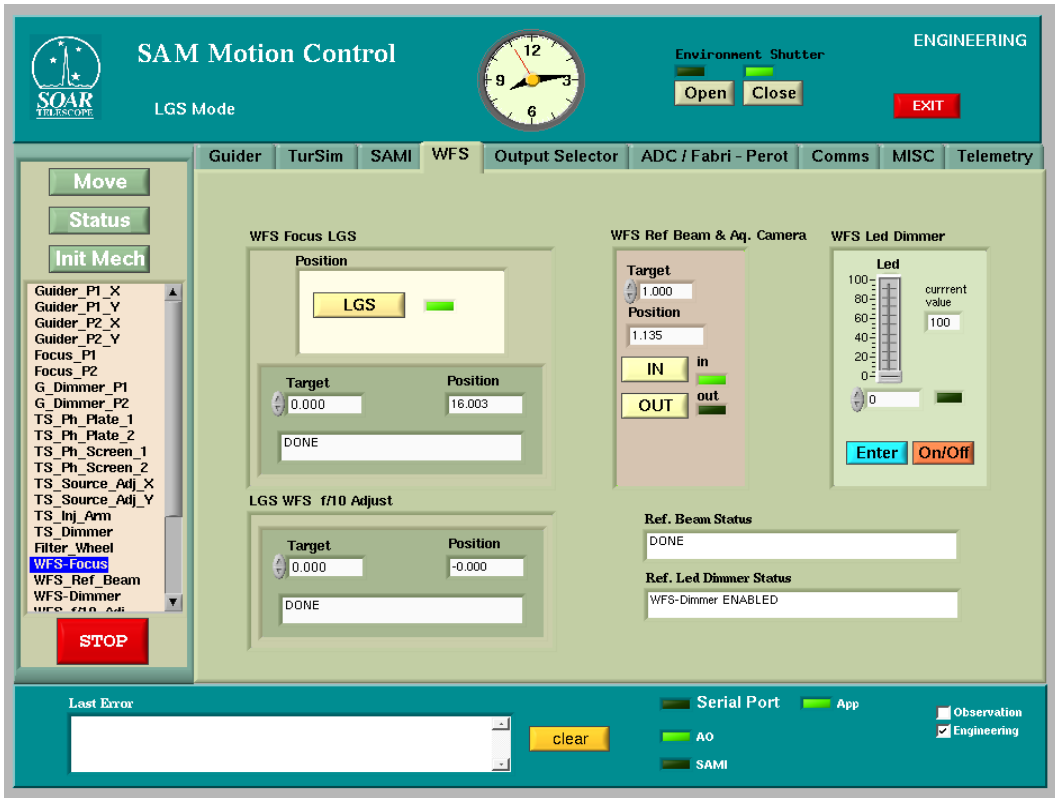
The central panel commands two DC motors that move the TurSim light source in X and Y (for its centering). The motors are connected to the source through a cam, so continued motor motion in one direction causes oscillating source motion. If you want to move X-motor at a constant speed, select the velocity with a slider (or enter the value) and press the *Move* button. Then press *Stop*. Small correction steps, of jogs, are done by pressing the buttons *Jog_X* and *Jog_Y*. The direction of the jogs can be changed, the step size is fixed.

The next panel, *Phase Screen Velocity*, sets the rotation speeds of two phase screens (in percent, from 0 to 100). The direction of rotation can be changed as well. The rightmost panel controls the intensity of the light source in the same manner as the GP artificial lights (intensity in percent and on/off *Power* toggle). Finally, the lower-left panel controls the servo motor that inserts or retracts the injection arm. Here, the power is also applied automatically only during the motion, otherwise the motor is always turned off.

4.4 SAMI

The SAMI tab has two panels (no figure here). The left panel controls the filter wheel: select the position, press *Move* and watch the result. The right panel controls the SAMI shutter. Its switch is normally in the *Automatic* position (shutter slaved to the SAMI Leach controller). By selecting the *Manual* mode, we should be able to control the shutter directly from the AOM SW, but this function is disabled even in the engineering mode.

4.5 WFS



WFS Focus LGS
Position
Target 0.000
Position 16.003
DONE

LGS WFS f/10 Adjust
Target 0.000
Position -0.000
DONE

WFS Ref Beam & Aq. Camera
Target 1.000
Position 1.135

WFS Led Dimmer
currrent value 100
0

Ref. Beam Status
DONE

Ref. Led Dimmer Status
WFS-Dimmer ENABLED

☐ Observation
☑ Engineering

Fig. 6. AOM GUI with the WFS tab opened.

The WFS tab (Fig. 6) controls three linear stages: the main stage *WFS-Focus*, the auxiliary stage with lens *LGS WFS f/10 Adjust*, and the stage that injects the reference source and, at the same time, reflects the field to the WFS acquisition camera. All stages are controlled in the standard way, as in the guiders. In addition, the dimmer and on/off switch for the reference light source are controlled. The button LGS sets the two stages to a specific LGS range defined in the configuration file; it is not normally used.

4.6 Output Selector

This device contains only one linear stage that moves the output selector mirror between two positions, *IN* and *OUT*. It can also be moved to any arbitrary position.

4.7 ADC/Fabry-Perot

This device controls the motion of two identical linear stages, one for the ADC and another for the Fabry-Perot (the Fabry-Perot is actually not installed, but its stage is operational). The rotation of two ADC prisms is also controlled here (they use small DC motors with encoders). The prisms move in the range 0—360 degrees. However, their reported positions are sometimes outside this range (to be corrected).

4.8 Communications

This is a technical tab for monitoring communication with other software. Select the number of server/client in the upper-left panel *Servers and Clients* to see the IPs (the numbering starts at zero) and examine the connection status in the lower-left panel *Who is connected*. The right panel displays current messages and answers. The button *Release* in the lower-right part should disconnect and reconnect the selected link (either SAMI or ICSoft), but this functionality is not implemented.

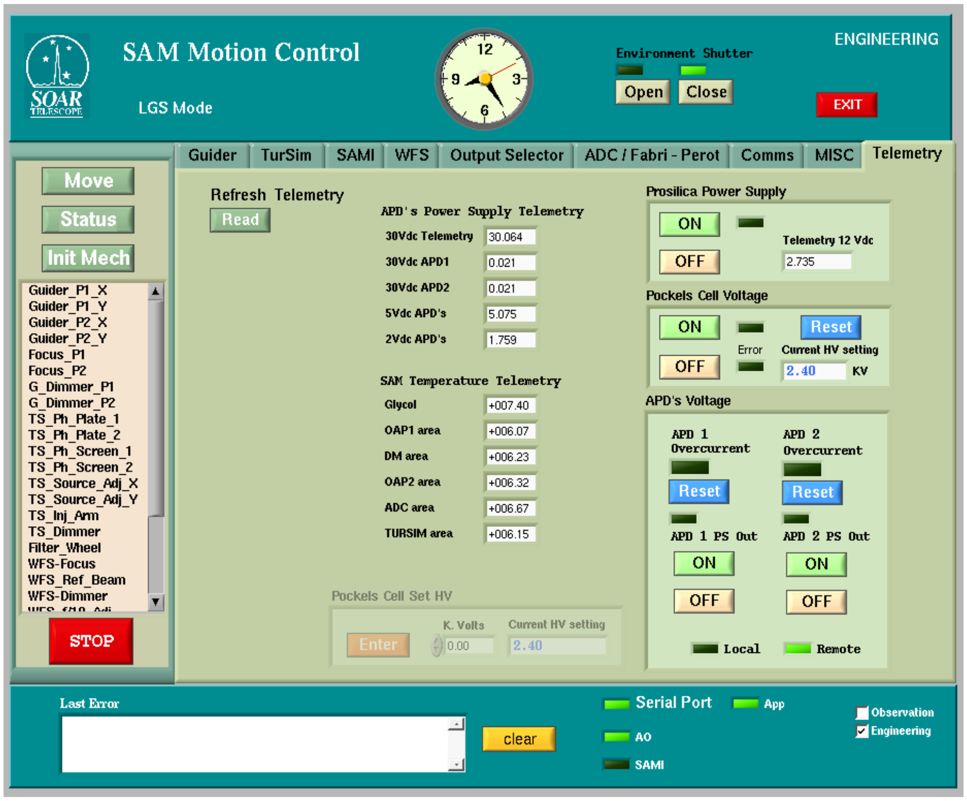
4.9 Miscellaneous

This tab is used to change the polling frequency of the status of mechanisms and telemetry (normally the polling is done every 30s), or to disable the polling (the buttons *ON/OFF*). The AOM is informed of the status of each mechanism either during its controlled motion (by the "state machine") or by polling. The check-box *Check Status Reliability* is used to deactivate the validity control of the polling data (the time stamp of valid data should not be older than the polling time interval). The buttons *Application Release/bind* do not work at present. The right-hand side of this tab (parameter editing) is not used.

4.10 Telemetry

This tab (Fig. 7) displays the telemetry data: voltages of various power supplies (in V) and temperatures (in deg. C). It also has several controls for switching the power to the WFS Acquisition camera (Prosilica) and to the Pockels cell. The regulation of the Pockels cell high voltage is disabled, but its current value is displayed, in kV.

Also, the 30V power to the two APD sensors of the guide probes can be controlled here. To switch it on (for each probe separately), you must press the *ON* button and then toggle the *Reset* button. The lights indicate the power and the over-light condition (the power turns off automatically in the case of excessive light). There is also a *Local/Remote* indicator for the APD power (the corresponding local/remote switch is located on the power supply in the SAM rack). Normally, the APD power is controlled remotely, by the AOM SW.



APD's Power Supply Telemetry

30Vdc Telemetry	30.064
30Vdc APD1	0.021
30Vdc APD2	0.021
5Vdc APD's	5.075
2Vdc APD's	1.759

SAM Temperature Telemetry

Glycol	+007.40
OAP1 area	+006.07
DM area	+006.23
OAP2 area	+006.32
ADC area	+006.67
TURSIM area	+006.15

Telemetry 12 Vdc 2.735

Current HV setting 2.40 KV

K. Volts 0.00

Current HV setting 2.40

Fig. 7. AOM GUI with the Telemetry tab opened.

5. Configuration files

The configuration files used by AOM SW are located in the `/user/aom/config/*/` directories.

1. `SAMMODE/AOM_ModeConfiguration_Default.cfg` defines the default mode of the AOM SW (engineering) and the default mode of the WFS (LGS).
2. `SAMMOTIONCONTROL/ENV_Default.cfg` defines the structure of the AOM SW by listing its main components (environment, devices, servers, managers, translations). This file must not be changed.
3. `SAMMOTIONCONTROL/ENVSCLN_Default.cfg` lists the servers/clients of the SW modules that can communicate with AOM SW: TCS, SAMI, and ICSoft. If some of those IPs are modified, this configuration must be updated and the AOM SW restarted.
4. `SAMPARAMETERS/AOM_Mechanism_Parameters_Default.cfg` is the main configuration file that sets parameters for each mechanism (see below).
5. `SAMPARAMETERS/AOM_Mechanism_Default.cfg` is a list of mechanisms in the left-hand menu of the GUI, not to be changed.
6. `SAMPARAMETERS/AOM_MechanismStatus_Default.cfg` lists data fields of the internal database than contain values for each mechanism, such as position.
7. `SAMPARAMETERS/` also contains 4 other small configuration files: `AOM_Simulated_Mechanism_Default.cfg` lists mechanisms which can be simulated in the software. `AOM_Guider_Zpoint_Default.cfg` is not used, `AOM_Guider_ZpointP1_Default.cfg` and `AOM_Guider_ZpointP2_Default.cfg` memorize the offsets for the guiders (must contain two zeros each).
8. `SAMSTATUS/SAM_TelemetryStatus_Default.cfg` lists the telemetry parameters.

The configuration files are read at the start of the AOM SW. However, the parameters can be updated "on-the-fly", if needed. If the file `SAMPARAMETERS/AOM_Mechanism_Parameters_Default.cfg` is modified while the AOM SW is running, the new values can be read by going offline and then initializing. The corresponding buttons are located in the *Env_Main.vi* window that is not used normally. Changes to other configuration files become effective only after stopping and re-starting the AOM SW.

The main configuration file `AOM_Mechanism_Parameters_Default.cfg` is structured in sections. Each section has a header in square brackets and contains lines "parameter=value". Comments are not allowed (for now). The order of parameters does not matter, only their names. For example, here is the section describing the GP1 X-motion, where the meaning of each parameter is explained after the # sign.

```
[Guider_P1_X]
id=10          # identification of the controller on the RS-485 bus
retries=2      # allowed number of re-tries to establish communication
port=4         # PC port corresponding to the RS-485 communication with the controller
enable=1       # set to zero to disable the mechanism
u-limit=3      # maximum position, mm
l-limit=-46.1 # minimum position, mm
commanded_position=-46.0 # Distance in mm from the reference to the zero position
offset=-46.0                  # same?
screw=1            # screw pitch of the stage, in mm
encoder=8000   # encoder steps per revolution
acceleration=2.5 # acceleration, mm/s2
velocity=5.0      # velocity, mm/s
deceleration=2.5 # deceleration, mm/s2
simulator=0       # set to 1 if the mechanism is simulated in the software
```

```
fiducial=0          # the reference sensor is internal to the stage. Set 1 if the sensor is external
ext. encoder=0      # the encoder is part of the stage. Set 1 if the encoder is external
direction=1         # motion direction corresponding to increasing position
awayfiducialdir=0   # direction to start search the reference
tofiducialdir=1     # direction to approach the reference
backfiducialdir=0   # direction to go away from the reference
tofiducial_lowvel=1 # direction of slow approach to the reference?
Park=-45.0          # park position
timeout=100.0       # time-out to move, seconds
init_timeout=100.0  # time-out to initialize
home_limit=150.00   # time-out ?
tolerance=0.05      # tolerance to reach the commanded position, in mm, before reporting DONE
kill=enable         # not used
force_init=FORCE    # force initialization on Init command even if the motor was already initialized
```

A qualified user may need to change some parameters, for example upper and lower limits, enable/disable mechanism, etc.

6. Errors

List of error/alarm messages:

1. Timeout error when the initialization or motion is not finished in allocated time
2. “rejected on state active” when a new command arrives while the device is still active
3. Communication error (fatal)
4. Motor not initialized
5. Outside range

If the AOM GUI becomes unresponsive (“frozen”), the program has to be killed manually. In a terminal, issue the command `ps -ax | grep labview` to see the process number (PID) and then kill each process with the command `kill -9 PID`.

7. Logging

The AOM SW logs commands and errors. The log-files are located in `/user/aom/logs/SAMMOTIONCONTROL/*/`, where the sub-directories are: `ADC, FILTER, FPEROT, GUIDER, OUTSEL, SAMCOVER, SHUTTER, TELEMETRY, TURSIM, WFS`.

Each directory contains 10 log-files for corresponding mechanisms, the files are named after date, such as `GUIDER_20110719_13h17m19s.log`. The files are “rotated”: the oldest one is deleted so that only 10 latest logs are kept. The contents of such log-file looks like:

```
2011-07-19 16:12:26 : GUIDER COMMAND - MOVE G_Dimmer_P2 0
2011-07-19 16:18:37 : GUIDER COMMAND - MOVEPROBE GUIDERP1 0.000 0.000 0.000
2011-07-19 16:19:21 : GUIDER COMMAND - PARK GUIDERP1
2011-07-19 16:20:45 : GUIDER COMMAND - MOVEPROBE GUIDERP1 -45.000 54.000 0.000
```

The errors are logged with the words “ERROR” or “ALARM” instead of “COMMAND”.

References

1. *The SOAR Communication Library New* (manual, author and year unknown)
2. SAM-AD-02-5400 Motion control: Summary (M.Martinez, Oct. 2 2006)
3. *Motion Control: LSES motor control board communication protocol.* M. Martinez, Jun 25, 2008 [LSES_motor_control_board_protocol.docx]