Introduction

The filter wheel driver software provides an interface to the Programmable Logic Controller (PLC) which controls one or more filter wheels. The software comprises a back-end, which communicates directly with the PLC over a serial link, and the front-end with which the user interacts. There are in fact a number of front-end interfaces available to control and query the filter wheels.

The software is modular, and may be used on its own (to control filter wheels not part of a larger system), or as part of interconnected systems (e.g. the Sutherland High-speed Optical Cameras (SHOC) instrument, which includes filter wheels, a global positioning system (GPS) receiver used for timing information, and a charge-coupled device (CCD) camera.)

A basic command line interface is provided to control the filter wheel, but most users will find it preferable to use the provided web interface. There is also a diagnostic interface, which may be used by technicians for the purposes of trouble-shooting.

The filter wheel software also provides a means for scripting the filter wheel behaviour. This is useful for users whose observation requirements are for a predefined set of steps to be taken, possibly repeatedly. The scripting interface should be particularly useful when the filter wheel is used in conjunction with other instruments, and a particular work-flow needs to be followed.

General Usage

When the filter wheel controller is first switched on, at the start of the night's viewing, it will need to be initialized. This action rotates the wheel until the reference position passes a sensor, and from then on the wheel's position is known. After this the user merely needs to specify the required position, and the PLC controller takes care of rotating the wheel as required.

To check that the wheel is operating as required, the user may check that:

- The initialization status is true.
- The wheel centred status is true.
- When the wheel is at position 1, the *at_reference* status is true.

These are indicated by green lights in the web interface, or status bits displayed in the command-line interface (CLI).

Occasionally (e.g. after a power outage) the PLC may become confused, and a software reset may be required. Both the CLI and web interface allow this.

Command Line Interface

As mentioned, the command line interface (CLI) is the most basic means of interacting with the filter wheel controller.

To start the interface, double-click the `filterwheel_cli` icon (in Windows) or in a bash shell, type `filterwheel_cli.py` (Linux).

The CLI may be invoked with a number of options. The *-h* option presents the user with a list of command line options, and explanations as to their use.

To specify a non-default serial port for the PLC connection, use the `--serial <port>` command (e.g. `--serial /dev/ttyUSB2`).

To specify the log level, use `--log <level>`, where `<level>` is one of `debug, info, warn, critical` or `error`.

To specify the logfile, use `--logfile <file>`. If this option is not specified, the default is to use `filterwheel_cli.log`.

Once the interface has started, the user may view the status, or control a particular wheel.

Entering *S* or *s* displays the status of the PLC. This lists, for each wheel (*A* and *B*):

- whether the wheel has been initialized
- whether it is currently centred
- whether it is currently at the reference position
- whether it is moving
- the current position,
- the id of the wheel (in binary notation).

Other options take the form `<wheel> <option>`. For example:

- `A I` - initialize wheel A
- `A R` - reset wheel A
- `B 5` - move wheel *B* to slot position *5*. The slot positions range from 1 to 8.

Entering *q* or *Q* exits the program.

Web Interface

The web interface (you get there by going to http://<your-instrument-name>.suth.saao.ac.za:5000 in a web browser) is the primary interface for interaction with the filter wheel. The filter wheel control software and web server are started as a service (Windows) or daemon (Linux), and the user should not need to start these. The web server provides a web interface for interaction with the filter wheels.

.

The status (Initialised|At Ref|Centered|Moving) of the each wheel is displayed at the top, below the wheel's name and binary-format ID.

The current position (1) of the wheel is displayed below this. The slot number, and if available, filter name and color are also displayed.

Below this, the next required position (2) may be entered. There are then buttons to move to the required position, or to initialize the wheel.

An "https://topswiki.saao.ac.zaMore..."https://topswiki.saao.ac.za (3) link allows the user to see which filters are installed in each slot (if known), and also allows the user to reset the PLC.

The "https://topswiki.saao.ac.zaPreferences"https://topswiki.saao.ac.za drop-down allows one to choose day-time or night-time modes, as well as whether to disable the tool-tips when one hovers over a button.

▪
More detail on the boxes in the filter wheel GUI interfaces for each filter wheel

▪
Note: the filter name and colour are obtained from a configuration file. This file includes a time for which this information is valid, for each wheel. If there is no information valid for a wheel for the current time, the name and colour are not displayed. This ensures that out-of-date or incorrect information is not displayed.

Diagnostic Interface

The diagnostic interface is invoked by starting the command line interface, using the --diagnostic_mode flag. It automatically sets logging to the most verbose level, and provides greater details on the data strings sent to and received from the PLC. There is also the ability to set various bits in the PLC, to control features not required for day-to-day operation.

Scripting

A scripting interface has been developed, and enables filter wheel operations to be controlled from a script.

The operations are available for each wheel and include status checks and filter wheel control.

Status:

- Is the wheel initialized?
- Is the wheel centred?
- Is wheel at the reference position?
- Is the wheel moving?
- What is the current position?

Control:

- Initialize the wheel.
- Move to a specified position.

So to initialize wheel B, then check that it's at the reference position, move it to position 4 and then check that it is centred, one might write a bash script as follows:

```
#!/bin/bash
wheel=B
port=/dev/ttyUSB0
newpos=4
./fwc --serial $port -w $wheel -a Z
```

```
sleep 4
atref=`./fwc --serial $port -w $wheel -a r`
echo "https://topswiki.saao.ac.zaatref = $atref"https://topswiki.saao.ac.za
if [ $atref -lt "https://topswiki.saao.ac.za1"https://topswiki.saao.ac.za ]
then
    echo "https://topswiki.saao.ac.zaFilter wheel $wheel is not at reference position after ini
    exit -1
fi
echo "https://topswiki.saao.ac.zaMoving wheel to $newpos"https://topswiki.saao.ac.za
./fwc --serial $port -w $wheel -a $newpos
sleep 4
if [ `./fwc --serial $port -w $wheel -a c` -ne 1 ]
then
    echo "https://topswiki.saao.ac.zaFilter wheel $wheel is not centred after move"https://tops
    exit 1
fi
echo "https://topswiki.saao.ac.zaSuccess. Wheel initialized and centred at position $newpos."ht
```