

PMAC Tuning Procedure, With Plotting

Introduction

This procedure describes the low level control of the PMAC environment with PEWIN32, in order to start the system up in lowest level mode (no PLC programs), and in the intermediate mode (PLC programs but PEWIN32 commands). These modes are typically used when testing the motors, setting up the encoder read heads with low speed slews, checking limit switch operations, and tuning the axis PID parameters. Plotting processes are also described, as the quick plot functions in the latest PEWIN 32 version are too buggy to be usable: nthe detailed plot set up is described.

Low Level Start Up

Connect all cables from the cabinet to the base panel, and have them checked. Switch on the cabinet at the rear circuit breaker. The PC should begin to boot. Switch on the DIOP switch on the front panel to enable the encoders. Log into the PC and look for the PEWIN32 icon on the desktop.

With a watcher at the telescope, holding the ESTOP button, run PEWIN32. Disable all default startup PLC programs with ctrl-D (to restart th PLC programs, boot them with “enable PLC1”).

In this low level mode, the drive mappings are:

Channel	Drive	Encoder
1-3	Focus	1-3
8	Time	10MHz input channel 4
9	Azimuth	9-12
13	Elevation	13-16

In low level mode, the chanell numbers for drives are unmapped, such that azimuth axis commands are all to channel 9. Examples are:

```
I922=2360 ; set az jog speed to 1 degree per second
M914=1 ; enable drive amplkifier channel 9 (ESTOPS all off)
```

A typical mistake is to type in “I914=1”, then wonder why the drive won’t operate below 0 counts. Beware the correct use of M variables and I variables.

Tuning parameters are set between Ixx30 and Ixx33 typically. For the azimuth drive, for example:

```
I930 ; Proportional gain, az drive
I931 ; Derivative gain, az drive
I932 ; Velocity feedforward gain, az drive
I933 ; Integral gain, az drive
```

Moving the drives requires open loop current output to the motors (torque output), or closing the servo loops and sending ‘jog’ commands to the drives (speed demand or trapezoidal move command). Typical commands are:

```
M914=1 ; enable azimuth drive amplifier
#9 ; select drive channel 9 for commands
o-15 ; send 15% of max current to azimuth drive, negative direction.
o0 ; stop sending torque current
j/ ; jog at zero speed (ie hold position with servo)
j:1234000; send motor to encoder position 1234000 counts with trapezoidal move
j+ ; jog at speed set by I922, indefinite, positive direction
```

Recommended Notes While Using Low level Commands

It is recommended that the low level operations be logged in note books or official records, with dates. This facilitates error recovery and problem solving. The classic "I914=1" error will be easily resolved if notes are taken.

Limit Switch Testing

The limit switch positions are sensed by the DIOP and mapped into the PMAC variables. Reading Allfiles.pmc with a text editor will allow the limit switch variables to be identified, and put into the PEWIN32 watch window for reference during testing.

Typical variables are M921 and M922 for -ve LIM1 and +ve LIM1 variables. Note that these are defined in reverse by Delta Tau in their PMAC documentation: M921 is the limit switch signal representing the end of positive travel. The only way to keep this in mind may be to see the limit switch names as verbs which show the only drive direction possible once the switch has set the value from 0 to 1.

The limit switches on the azimuth drive are logically compared to the Centre of Rotation (COR) switch, to allow them to show travel of over 360 degrees. Therefore, each set of limit switches can be physically actuated without setting their sense variable to 1, depending on the current position of the COR switch. This takes some getting used to.

Intermediate Level Control

On startup of the PC, the PMAC loads a compiled version of the file **allfiles.pmc**, and runs them as the combined PLC program set. These PLC programs control many of the safety features of the drive system, and coordinate the low level activities for multiple axis operation.

Once these are operating, they may be disabled by **ctrl-D**, and restarted with **enable PLC1**. However, once enabled, the low level parameters used for control of axes are replaced by or copied to higher level variables, which allow higher order functions to be performed, such as homing of axes, standardisation of encoder counts vs position, and error checking on axis operations.

Refer to Space/Projects/StromloIII/1.8m Telescope/ 1.8 Meter Telescope and Enclosure Operation Procedure.doc for details of the typical mapping of drive commands, including homing and error checking.

Axis Tuning

For a stiff, massy structure such as a telescope, the sequence of tuning is normally:

Increase P until instability occurs, then back off

Increase I until following error nulling is fast (typically $0.7 * P$ to $1.2 * P$), but no instability is exhibited at any speed.

Use low, equal values of D and Vff, increasing both together until steady state following error is minimum (usually values of 100 to 300 are used).

Fine tuning by slight imbalance of D and Vff, or backing off P and I together, may improve the stable speed range somewhat. The goal is to achieve the lowest steady state following error over the target range of jog speeds. Acceleration following error is less important, but instability such as "singing" or vibration are unacceptable.

Detail Plotting Using PEWIN32

The plot options of PEWIN 32 are Quick Plotting and Detail Plotting. Sadly, the Quick Plotting functions are so buggy as to be unusable. Detail plotting setup is described here.

There are several setup stages, from selecting motors to gather data from, defining the sampling interval, selecting the low level PMAC variables containing plot data, scaling the data collected, applying text to plot axes, saving the setup file, and then actually clearing a data buffer and collecting data from the variables.

1. In the Motors to Gather button edit box, set the Gather Period to 2252 for one second sampling, or 225 for 0.1 second sampling, etc. Select PMAC variables in order of preference, as plot data parameters. Refer to the Turbo PMAC Software Manual, Chapter 8, page 8-4 for motor register information.
2. Button 2, describes the data description or name for each selected parameter, and allows scaling of the parameter prior to plotting. This can be useful, as the counts for a 4 encoder system will initially be read in as 32 count added values. The calibrated count values must be derived by multiplying the parameter by 1/32 or 0.03125. Note that the use of the 'X-Y' lines in this option appear to be buggy and not to work. For such measurements as following error, commanded – actual is a logical choice, but thankfully, the following error is directly available from the PMAC manual as D:\$x91, where x is different for each motor axis. For channel 9, x=4 (D:\$491), and for channel 13, x=6 (D:\$691).
3. The third button allows the plot options to be configured for each data parameter, and then allows the configured plot file to be saved and recalled. Note that a bug here fails to update the plot options on recalling another plot configuration, but restarting the Plot application updates the displays correctly.
4. Button 4 allows the data buffer to be cleared, but does not allow access to the buffer size. See the Turbo PMAC Software Manual for the buffer size if the default is inadequate.
5. Button 5 starts the buffer filling as configured above. When full, the gathering is ineffective and should be stopped.
6. Button 6 stops data gathering.
7. Button 7 uploads the data buffer to the plotting application, ready for display.
8. Button 8 allows the plotted parameter to be selected from the gathered parameters, then displayed. Repeat steps 4 to 8 as required during testing.