Local SPEC macros at APS beamline 7ID

Donald A. Walko
Advanced Photon Source, Argonne National Laboratory, Argonne, IL 60439
`d-walko@anl.gov`

June 29, 2016

This document describes important and sometimes unique aspects of SPEC operation at beamline 7ID. Primarily, this covers macros that are not part of the standard SPEC distribution. Some of these are custom to 7ID, some are fairly widespread at the APS and other synchrotrons. Not every custom macro is listed here, but I've tried to cover the ones that would commonly be called by users–or at least, the setup and help macros that can lead to further details.

Items that may be particularly of interest to users performing time-resolved pump-probe diffraction experiments are highlighted.

The online manual [1] and the command-line help utility are good sources of information for the standard macros and things of SPEC. We have further documentation of standard macros elsewhere [2].

Starting SPEC

At beamline 7ID, SPEC is generally used only for collecting diffractometer data in hutch 7ID-C. It runs on the linux box "gremlin" from the user account "7idc". These are the geometries currently available:

`fourcC`
: This geometry operates the large Huber in 7ID-C as a four-circle diffractometer. The mu and nu motors must be at their zero positions or else the geometry calculations will be confused. The sample rotates on the phi, chi, and theta axes, and the axis that rotates the detector arm is two-theta.

`psic`
: This geometry operates the big Huber in 7ID-C with all six axes. (The "psic" and "sixc" geometries of SPEC happen to be slightly different.) The sample rotates on the phi, chi, eta, and mu axes, and the axes that rotate the detector arm are delta and nu.

`specC`
: The fourcC, psic, and specC geometries are linked together. This geometry doesn't do anything, but it is a handy place to hide motors and counters that are used sometimes, but not in the present experiment. (Changing an item's geometry is done in `config`.)

`fourcN`
: This geometry operates the small Huber diffractometer on the optical table in the middle of the 7ID-C hutch. It is particularly intended for nanodiffraction experiments, hence the "N."

PyMca

PyMca is software designed at ESRF [3]. It was particularly intended for analysis of multichannel analyzer data, but it is great for plotting and simple analysis of SPEC scans. Use it. Type `pymca` from the command line to get started.

Laser information

Two laser systems may be in use for pump-probe experiments at 7ID.

Ti:sapphire This is the 50-fs pulse, 1 kHz laser housed in the 7ID-E laser lab. The delay of the pulses relative to the x rays is controlled with EPICS-controlled delay generators. In SPEC, a "macro motor" named `delay` controls this time delay. It is not a real motor like phi or chi, but can be moved, scanned, etc., as if it were. It is in units of microseconds for the sake of having the decimal point at a convenient position (the related PV is in units of seconds). The intensity of the laser is adjusted with a waveplate/polarizer combination; the waveplate is rotated with a (real) motor named `wpc` in units of degrees.

Duetto This is the 10- or 100-ps pulse, high rep rate (54 kHz to 6.5 MHz) laser housed in the 7ID-D hutch. In this case, the laser-to-x-ray delay pseudomotor is `deld` and the waveplate for intensity control is `wpd`.

The basic SPEC commands

SPEC is a command-line based program. Therefore it is important to know the correct commands; fortunately, some commands will list the type of parameters needed if you enter the wrong type (or number) of parameters. Actually, most "commands" (and many variables) are actually macros and open to be redefined. The writing and implementation of new macros is quite straightforward; users often write shortcut macros which are combinations of a number of commands. Text files (typically ending in `.mac`) can be filled with commands and are called with `do` or `qdo`. It is also important to note that SPEC is case-sensitive; most commands and variable names are lower-case, while certain special variables and macros are upper-case.

Local/custom macro implementation

"Information" commands, shortcuts, general macros, and macros about macros

`XSD7_MACROS`
: This reloads all of macros in SPEC.

`CLEAN_START`
: This redefines a number of variables to values that are usually good starting points. This is under development.

`do_user_mac` *macro_file.mac*
: This rereads macro definitions from their source. Should be handy if you know the exact macro file you want, but it needs fixing first.

`sh`
: Shell: Go to unix prompt.

`lp` *some command*
: Both performs and sends the output of the SPEC command *some command*, whatever it is, to the printer. This is a very handy way to print out motor positions or diffraction parameters with commands such as `lp wa` or `lp pa`. But don't use it with a scan!

`pplot`, `lpplot`
: Send the most recent scan (using cplot) to the printer with a linear or logarithmic y axis, respectively. But seriously, use pymca instead.

`macrolist`
: Lists all non-standard macros and the source of their definitions.

Macros for setting things and moving things

Energy macros. None of SPEC's options for monochromator macros are used at 7ID. The following macros thus have custom definitions:
`getE`: sets the wavelength (`LAMBDA`) based on the value of the high heatload mono's position, obtained from EPICS.
`moveE` *energy_in_keV*: move mono to a given energy
`Escan` *start finish intervals time [fixQ]*: scan mono (units are keV). The literal "fixQ" means hkl is kept constant during the scan.

`ummv` *mot1 pos1* [*mot2 pos2*] [*mot3 pos3*] [*mot4 pos4*]
: Like umv but moves several motors (up to four) at once.

`check_beam`
: This macro will wait until the beam is ready (that is, shutter open, undulator not very open-gap, current in the ring). If this precedes the scanning commands in a .mac file, then you won't collect a bunch of empty scans if there's a beam loss.

`set_rel_lm`
: This sets a soft limit where two motors can only be so far apart. This may need further testing before being trusted too much.

Macros for orientation

`sor`
: This is a shortcut for `or_swap`. It exchanges the primary and secondary reflections in the orientation matrix.

`setaz2`
: This is a handy macro, only available in psic geometry for now, to input the surface normal direction (or other "special direction") into SPEC. It is like `setaz` but you don't have to remember the sign conventions associated with the angles called *sigma* and *tau*. The procedure is to make your "special direction" collinear with the eta axis. That is, reflect a laser off the sample surface, adjusting chi and phi until the reflected spot doesn't move when eta is rotated. The resulting values of chi and phi are called their "flat" values. In setaz2, you can choose to input flat_chi and flat_phi and the macro will correctly compute the surface normal direction.

Scan macros

xscans
: These are from the ESRF, and are designed for efficient collection of the background intensity. They are like most regular scans except you can increase the step spacing on the ends of the scan with two extra parameters (which, if not

included in your command, are the same as they were, the last time an xscan was performed). For example,
`xascan` *motor start finish intervals time* [*expansion* [*step_ratio*]]
Other macros in this category include `xa2scan` and `xdscan`.

Information will be added for hklextra, hklplane, etc.

`lup2`
A line-up (dscan) that goes to the center of the peak after the scan, if one is found.

`th2th` *tth_start_rel tth_end_rel intervals time*
A shortcut for theta/twotheta scans if you want theta to be exactly half of twotheta. This also works in psic geometry, but the motors are known as delta and eta.

`rlf` *eta_start eta_end intervals time*
A shortcut for vertical reflectivity scans in psic geometry, moving eta and delta.

`rlf_horz` *mu_start mu_end intervals time*
A shortcut for horizontal reflectivity scans in psic geometry, moving mu and nu.

`epics_ascan` *drive_PV read_PV start finish intervals time [positioner_settling_time [counter_settling_time]]*
This is a set of macros which allow scanning of almost arbitrary EPICS input PVs as a regular SPEC scan. Two PVs need to be listed; sometimes the same PV will be used if a separate readback value has not been defined in EPICS. Also available are `epics_dscan`, `epics_d2scan`, and `epics_mesh`.

Pilatus macros

These macros integrate the Pilatus pixel detector into SPEC. Since we gate the detector in pump-probe studies, we are often operating in External Trigger mode rather than Internal mode. Alignment mode is useful for initial alignments.

When counting with SPEC, the number of frames collected by the Pilatus is not set by the SPEC count time in a count command or a scan. Rather, the Pilatus count times are set by the macros below.

`pilatus_help`
Displays the commonly used macros related to Pilatus detector.

`pilatus_setup`
Set up the Pilatus detector. It and its computer must be up and running already.

`pilatus_show`
Displays the current state of the Pilatus detector.

`pilatus_on`
Count with the Pilatus detector.

`pilatus_off`
Stop counting with the Pilatus detector.

`aon`
Start collecting images in alignment mode.

`aoff`
Stop alignment mode.

`intrig` *[exposure_time]*
Switch to Internal mode with *exposure_time* seconds per point.

`extf` *[exposure_time]*
Switch to External Trigger mode with *exposure_time* seconds per point for the femtosecond laser (Ti:sapphire). Specifically, the Pilatus is set up to collect *[exposure_time]**1000 exposures since the laser is operating at 1 kHz.

`extp` *[exposure_time]*
Switch to External Trigger mode with *exposure_time* seconds per point for the picosecond laser (Duetto). Specifically, the Pilatus is set up to collect *[exposure_time]**54000 exposures since the laser often operates at 54 kHz. The rep rate is not adjustable for this macro, yet.

Project startup macros

Here is an extremely useful set of macros that sets up standard directories and files for a project. Their use is recommended. It will, however, be extremely frustrating if you don't know it has been implemented.

`startup_help`
Summarizes the important macros.

`startup_startup`
This is where to begin at the start of your beamtime.

`startup_newproject`
Creates a new project for the current user.

`startup_newsample`
Sets up a new sample in the same project.

`startup_newfile`
Sets up a new file, for the same sample.

`startup_shutdown`
Ends the experiment. Resets paths and filenames to default values.

Attenuators

There are two sets of XIA attenuators on the input beampipe of 7ID-C. Typically, the thickness of the metal foils in each attenuator box doubles from one foil to the next. They can be controlled from the SPEC command line via these commands:

`filters1` *stat1 stat2 stat3 stat4*

`filters2` *stat1 stat2 stat3 stat4*

where *statN* is 1 or 0 for insert or remove filter, respectively. The state of the attenuators is stored in the header of each SPEC scan.

Temperature macros

Two temperature controllers are currently compatible with SPEC at 7ID, a Lakeshore 340 and a Watlow controller. These macros will read and set the temperature, but they will not turn on the heater; that must be done manually in EPICS.

`temperature_help`
: Lists all the temperature-related macros, including `settemp` and `showtemp`.

`temperature_setup`
: Choose a temperature controller.

`save_temperature` *[on/off]*
: Add or remove temperature-related values as columns in the data file.

MCA macros

The following macros can be used to collect data in SPEC from multichannel analyzers such as a Vortex fluorescence detector, a Struck 38xx, or anything which produces an array of data to be saved. The MCAs and regions of interest (ROIs) can be saved in SPEC.

The SIS (Struck Innovative Systeme) 3801 VME board is a scaler that can act as a multichannel analyzer. Counting can be triggered with an external scaler, and an internal clock can advance the channels. Dwell times can be as short as $\sim 2\ \mu$s.

`setup_mca`
: Configure multiple EPICS MCAs in SPEC.

`show_mca`
: Displays the configured MCAs.

`mca_rois`
: Displays the configured ROIs.

`rm_mca`
: Remove one or all MCA, and its associated ROIs.

`rm_roi`
: Remove one or all ROIs.

`MCAscanpt` *any scan*
: Performs *any scan* while collecting and saving MCA data for each point in the scan.

Deadtime macros

If the avalanche photodiode is in photon-counting mode, its efficiency is very limited if count rates are restricted to the linear range. Deadtime corrections will greatly improve the efficiency. These macros assume the detector-arm APD is gated at the rate of the 1-kHz laser and the monitor APD is gated at P0.

`deadtime_on`
: Turn on deadtime corrections for gated APD channels and adds results of the calculations to the SPEC file.

`deadtime_off`
: Clears the deadtime corrections.

Flexible gate

The FlexiGate combines the features of a scaler, delay generator, and logic units. It is used primarily for gating detectors in pump-probe experiments, replacing a rats nest of LEMO cables running between various NIM bins. It is configured in EPICS but the scaler outputs can be read in SPEC.

Acknowledgements

The SPEC macros described in this report were borrowed (stolen) from a variety of sources, some of which include Peter Eng, Mark Rivers, Tom Trainor, Yong Chu, Paul Lyman, Christian Schleputz, Xuesong Jiao, and Jeff Kirchman. I appreciate all of their efforts to improve the ultility of this software.

References

[1] Certified Scientific Software, `http://www.certif.com`

[2] `https://www1.aps.anl.gov/files/download/SECTOR7/7ID/spec_commands.pdf`, `https://www1.aps.anl.gov/files/download/SECTOR7/7ID/spec_basics.pdf`

[3] `http://pymca.sourceforge.net/`