Linux Command Line Guide (EBPG Computer)

Basic Syntax Used In This Guide

`<parameter>`	Required parameter (don't enter the brackets)
`[parameter]`	Optional parameter (again, no brackets)

Commands will appear in **`monospaced fonts`** (mostly).

Linux 101

`ls`	list the contents of a directory (like dir in DOS)
`cd <directory>`	change directories to the given directory
`cd ..`	move up one level from your current directory
`cd ~`	go to the home directory (**`/home/pg`**)
`pwd`	display the current directory
`mkdir <directory>`	create a directory
`rmdir <directory>`	delete a directory (must be empty first)
`cp <file> <destination>`	copy a file to the destination directory or filename
`mv <file> <destination>`	move a file to the destination directory
`mv <file> <newname>`	rename a file (move it to a new name)
`rm <file>`	delete a file
`cat <file>`	display a text file directly in terminal
`less <file>`	display a text file with scroll control ('q' to exit)
`gedit [filename]`	a text editor (graphical)
`vi [filename]`	a text editor (text-only)

- You can get help with the syntax for almost any linux command by typing the command followed by **`-h`** or **`--help`**. You can also type **`man <command>`** to see the UNIX manual entry for the command, which probably has more information than you ever wanted. Hit 'q' to exit the manual display.

- Copying/pasting: in the terminal window, highlight the text you want to copy and middle-click (the mouse wheel) to paste it wherever the cursor is positioned. This is useful if you want to copy a long filename

onto the command line, for example; you can do a **`ls`** command to display the files in the directory, then copy the filename from the output.

- Stopping a process: you can abort almost any process (including EBPG jobs) by hitting **CTRL-C** in the terminal window the process is running in. Closing the terminal window will usually, but not always, kill the process too, but the **CTRL-C** method is more foolproof.

-
    - Killing some pg commands may cause the EBPG software to lock up, which will cause any executed pg command to freeze. If this happens, open up a new terminal window and restart the pg interface using the command **`pg_restart`**.

- Be careful when moving or deleting files! Users on the EBPG system have full access to all other users' files as well as a lot of system files that are vital to tool function.

- Adding the & suffix after a command (e.g. **`gedit&`**) will execute the command in the background. This is useful for commands that open a new window, since it will let you continue to use the terminal you ran the command from while the program window is open (it will otherwise be frozen until the window closes).

- For more useful command line stuff, check out this Bash cheat sheet (PDF).

Loading/Unloading

`subl <position>`	load a sample holder from a given cassette position to the stage. Options for cassette positions are 1 (top shelf) and 2 (bottom shelf). Check the CSYS display to make sure the cassette position or stage you're trying to move a holder to is unoccupied!
`subu <position>`	unload a sample holder from the stage to a given cassette position
`pg select holder [ID]`	tell the system which holder is on the stage. If you know the ID number of the holder you can add it at the end, otherwise you'll be prompted to choose one from the holder list. You HAVE to specify which holder is in the system before you can do any alignment, calibration, or writing!
`load <position> [ID]`	Alias to load a holder from **`<position>`** in the airlock, select holder ID **`[ID]`**, find the calibration mark with **`mvm`**, and measure the current, all in one convenient command! Holder ID will prompt you if left blank; cassette position isn't optional.
`unload <position>`	Unload a sample holder from the stage to **`<position>`** in the cassette, per **`subu`**. Also sets the beam current back to 1nA and displays the **`colchk`** output so you can fill out your run sheet.

Stage Movement

`pg move pos <x,y> [--rel]`	Moves the stage to the given coordinates. If the **`--rel`** flag is specified, the move is relative to the current position; otherwise the stage coordinate system is used. **Shortcut:** **`mvrl <x,y>`** (relative), **`mvab <x,y>`** (absolute)

	The default units of x and y are microns. You can use other units by giving an XY pair like `45.3mm,22.8mm` for example
`pg move mark <x,y> <markname> [--rel]`	Moves the stage to the given coordinates (see `pg move pos`) and attempts to locate a mark of type `markname` there. If it can't find the mark, an error is returned. If `pg move mark` is used without any other arguments, the stage moves to the loaded holder's predefined reference mark location and tries to find the reference mark there. **Shortcut:** `mvm [arguments]` (absolute), `mvmrl [arguments]` (relative)
`findmark <markname>`	Attempt tolocate a mark centered on the SEM screen (same as `pg move mark 0,0 <markname> --rel`)
`pg measure current`	Moves the stage to the loaded holder's Faraday cup and measures the beam current. **Shortcut:** `mea_c` or `mcur`
`pg move home`	Moves the stage motors to their home position, which is useful if the stage is frozen or giving errors. The stage will lose its position reference when it runs this command, so follow it with a `pg move mark` to re-establish stage location
`unlock`	Sets the stage-lock flag to 0, useful if you're getting "stage locked" errors when trying to drive around.

- CSEM can also be used to move the stage, either by double-clicking on the SEM display to center it at the click site or by using the controls to the right of the display.

Beam Control

`pg get apert`	See which aperture is currently switched in **Shortcut:** `aperture`
`pg archive restore beam <beam filename>`	Load a saved beam profile. **Shortcut:** `beamload <beam filename>`
`pg info archive beam`	Display a list of saved beams. **Shortcut:** `beamlist` (displays beams for current aperture only!)
`pg measure current`	Moves the stage to the loaded holder's Faraday cup and measures the beam current. **Shortcut:** `mea_c` or `mcur`

Job Creation and Execution

`ce <username>`	Switch to a user's environment, or create it if it doesn't exist. ALWAYS run this before executing CJOB!
`cjob [cjob file]`	Run CJOB, and optionally load an existing .cjob file. There are different ways to execute

	CJOB, but to ensure that it's pointed to your directories it's best to run it from the command line after first using the **ce** command.
cview [gpf file]	Run CVIEW, optionally loading an existing .gpf file
job <filename.job> [options] <holder ID> <cassette pos.> [x1,y1] [x2,y2] [x3,y3] [x4,y4]	Execute a job file and log the output. **[options]** can be either **-m, -f**, or **-c**, to indicate that the coordinates in the command line are referenced to the holder mark, the faraday cup, or the holder center (respectively). If no option is given, absolute stage coordinates are used. **<holder ID>** is the ID number of the holder with the sample being written **<cassette pos>** is the position of the holder in the cassette (1 is top slot, 2 is bottom slot). **If the holder is already on the stage (which it almost always should be), use 0 here.** **NON-ALIGNED WRITES:** **[x1,y1]** defines the CENTER of the write. If no coordinates are given here, the system uses the holder center. **ALIGNED WRITES:** The coordinates at the end of the command line give the location of the 3 (or 4) global alignment marks. These must be given in the order that they were defined in CJOB (e.g. mark 1 in CJOB comes first). Note that all coordinates in this command have units of microns, which cannot be changed. This is different from every other pg command!
jobx <filename.job> [options] <holder ID> <cassette pos.> [x1,y1] [x2,y2] [x3,y3] [x4,y4]	Same as the **job** command, but will send you an email (to the address in your environment settings) when the job completes or terminates with an error. Useful for long writes when you don't feel like sitting in front of the tool.
blindjob <filename.job> <holder ID> <cassette pos.> [x1,y1] [x2,y2] [x3,y3] [x4,y4]	Same as the **job** command, but turns off the height meter feedback during the write. Instead, it will prompt you for coordinates at the beginning of the write, drive to those coordinates, and perform a single height measurement. It then uses that height reading as the height for the entire write. Useful for transparent substrates and other situations where the height meter can't get consistent readings from the surface of a sample. Note that for this to work, the "height check" option in your CJOB exposure object has to be set to "none." Note also that relative-coordinate switches (-f etc) aren't supported at the moment. Running without height feedback means that the system won't be changing focus to account for height variations. This means you need to be extra careful when leveling the sample if you want your entire pattern to be fully in focus. The depth of focus of the beam is ~1 μm, so ideally the height will change by less than that over your entire write area. As of this writing, **blindjob** is still very much in beta. Talk to Bryan if it doesn't seem to work or you want it to do something it doesn't currently do.

Alignment Marks

pg marker create <type> <tone> <x1,y1> [x2,y2] <markname>	Create an alignment mark definition. **<type>**: Marker shape. **rect** for rectangle, **cross** for cross **<tone>**: Marker tone. **pos** for positive (marker brighter than field in SEM), **neg** for negative (marker dimmer than field) **<x,y>**: For a rectangular mark, **x1** and **y1** are the width and height, in microns. For a cross-shaped mark, **x1** and **y1** are the width and height of the entire cross and **x2** and **y2** are the width/height of the actual bars. **<name>**: The name to save your mark under. Names are limited to six characters and users are encouraged to use the convention **rn20** for a 20 µm negative rectangle, **cp5010** for a 50-µm-wide positive cross with 10-µm-wide lines, etc.
pg marker delete <markname>	Delete a marker definition. There can only be 35 markers defined on the system at one time, so you may have to delete some to make room for new ones. Please be considerate to the other users and delete your own marks before you delete other peoples'!
pg info marker ident *	Display all currently defined markers. **Shortcut:** **marklist**
pg info marker ident <markname> /test=par	Display the mark-location parameter table, with current and default values, for alignment mark <markname>. See the alignment mark troubleshooting section for details. **Shortcut:** **marktable <markname>**
pg marker set <markname> <parameter> <value>	Set a given parameter in the marker table for <markname> to a specific value **<parameter>** One of the marker-table parameters **<value>** The value to give the parameter. **Shortcut:** **markset <markname> <parameter> <value>** * The parameter names, current values, default values, and allowed ranges can be listed using the **marktable <markname>** command * Changing the marker-table settings can sometimes help the system align to marks it is having problems locating. See the troubleshooting section on mark-location failure for details on how to tweak the marker table parameters.
pg marker reset <markname>	Resets all parameters in the marker table for **<markname>** to their default values. *Shortcut:* **markreset <markname>**
pg ebpg save	Save the new mark definitions (and other machine parameters) to the global data file. If you define new marks, you have to run this command before running CJOB if you want the new marks to be visible in CJOB. **Shortcut:** **$pg save**

System Operations

`pg ebpg hotstart`	Restart the EBPG software, without reinitializing the hardware interfaces. Useful if pg commands are freezing or the system is otherwise nonresponsive. This has no effect on hardware, so it's always safe to run. Always use `pg ebpg save` before running this to save the global data though. **Shortcut:** `$pg hot`
`pg ebpg coldstart`	Restart the EBPG software and reinitialize all hardware connections. Takes longer than a hot start but can sometimes fix issues that a hot start won't, so try it if hotstart doesn't fix your problem. Again, completely safe to run any time, but use `pg ebpg save` first to make sure the global data is saved. **Shortcut:** `$pg cold`
`pg ebpg save`	Saves the current system state in the global data file. When the software gets restarted with `hotstart` or `coldstart`, it reloads everything from the most recent global data file, so if you don't run this before restarting you may find some settings are different. **Shortcut:** `$pg save`
`pg ebpg shutdown`	Saves the current system state in the global data file and shut down the software. The only reason you'd ever need this is to reboot the PC, which you should never be doing anyway (see next section). You'll need to do a coldstart to bring everything back online if you run this. **Shortcut:** `$pg shut`
`pg_restart`	Save the global data and perform a hotstart. Basically just runs `$pg shut` then `$pg hot`. Great for lazy people!

Rebooting the PC

The first rule of rebooting the PC is **DON'T REBOOT THE PC** The second rule of rebooting the PC is ***DON'T REBOOT THE PC***

If you find yourself in a situation where the PC has been turned off (should never happen, the system has a UPS) or is completely frozen (should also never happen in linux) though, you may have no choice. Try to contact Bryan before doing anything if this is the case though.

If possible, save the current system conditions by using the `pg ebpg shutdown` command prior to restarting. To restart, use the command in the system menu at the bottom of the screen.

When you start the PC by pushing the power button (or restart it), it will take about five minutes to go through the full Linux boot sequence. Assuming nothing goes wrong, you should eventually reach a login prompt. Log in using the following user information:

Login: `pg` **Password:** `beamwriter`

The login and password are case-sensitive!

There is a roughly 50/50 chance that the right monitor will fail to turn on or will turn on with an extremely low resolution. This appears to be random and the cause of it is unknown (probably display driver issues), but rebooting the system again will usually fix it (or call Bryan, who can fix it with root access).

Once you've logged in you'll need to reconnect the PC to the EBPG's electronics. To do this, open a terminal window and use the `pg ebpg coldstart` (shortcut: `$pg cold`) command. You should see a bunch of status

messages and hopefully no errors after doing this. Once the electronics are started, resume CSEM and CSYS by running them from the desktop icons

The system should now be ready for normal use. If anything goes wrong at any point during this process, stop immediately and contact Bryan (this counts as an emergency).

It's extremely unlikely that you'll ever need to do a full reboot. If the system has lost communication with the electronics (you get a "bus error" message when attempting to run pg commands) or is otherwise freezing or giving errors when you try to run pg commands, reset the link to the electronics with **`pg ebpg hotstart`** (**`$pg hot`**). CSEM and CSYS can always be restarted, and any process (including writes) that needs to be stopped can be killed using **CTRL-C**.

General Tips

- After loading and selecting a holder, you should always immediately do a **`pg move mark`** and **`pg measure current`** command to verify that 1) you have the right holder selected (the mark and cup are where they're expected to be) and 2) the system's mark-location system is functioning and it has a beam down.
- If you run CJOB from the command line, make sure to first use ce to switch to your environment. Otherwise you'll have to drill down through the whole filesystem to get to your home directories. If you run it from the desktop icon, you'll be prompted for your environment name.
- Double-check the locations of your global alignment marks that you measured on the alignment microscope by driving to them (usually using a relative move from the Faraday cup or other reference) and turning on the SEM. Center the mark in the SEM display by double-clicking it and have the system locate the mark (**`findmark <markname>`** when you're directly over it). If location is successful, read the new, more accurate mark location off the SEM coordinates. This has the advantage of also telling you if the system is having problems with your marks before you start a write.
- You can always read the current absolute coordinates of the stage off of the status panel in CSEM.
- Be aware that the center of the CSEM SEM-mode viewscreen does not correspond to the center of the scan field! To find the "true" center, do a pg move mark, click "calibrate center", and drag the crosshairs so they're centered over the mark. CSEM will now treat the crosshair location as the center. Since CSEM doesn't save settings, this has to be redone each time it is closed and re-run .
- Many commands have short forms not listed in this guide. See the 'mnemonics' document on the desktop or type **`mnemonics`** at a prompt for a full list.
- Any **`pg`** command can be shortened as long as it isn't ambiguous. For example, **`pg mea cur`** will work the same as **`pg measure current`**.
- If you have any questions or are unsure about something, always feel free to email or call Bryan during reasonable hours (type **`ohno`** to get his contact information). If something seems like an emergency situation (e.g. alarms in CSYS, holder stuck between the chamber and loadlock, column on fire) call immediately no matter what.

► Attachments

▼ Attachments

Topic attachments

I	Attachment	Action	Size	Date	Who	Comment
pdf	bash_cheat_sheet.pdf	manage	104.9 K	27 Jul 2015 - 15:00	BryanCord	Bash cheat sheet