

Running Sigma16 Cheatsheet

February 7, 2019

WARNING: THIS GUIDE COMES WITH NO GUARANTEES. IT IS MEANT TO HELP YOU RUN SIGMA16 AT YOUR OWN MACHINE BUT PLEASE DO NOT, I REPEAT DO NOT FOLLOW IT BLINDLY AND EXPECT IT TO BE A RECIPE FOR SUCCESS. ALSO, THIS GUIDE IS RATHER SARCASTIC, PLEASE BEAR THAT IN MIND WHEN PROCEEDING. I DO NOT MEAN TO MAKE FUN OF YOU, BUT IT'S JUST MORE ENTERTAINING IF I DO SO!

1 Introduction

If you've got here you are either very lost, own a mac or just want to get that thing going. Perhaps because you've only got a few hours to submit your AX (Oops). Fear not, we are here to help. Follow the instructions below on how to get Sigma16 setup on your own machine.

2 Running it the easy (less proper) way. Running with Wine.

2.0 Windows

What are you doing here people... just double click the extracted Sigma16.exe thing and follow the instructions. Go to the other guides!!! Jeez, now that we've got them out of the way let's proceed.

2.1 Linux-generic

1. Open that scary dark thing called 'the terminal' and type:

```
sudo apt-get install wine-stable
```

(because you do not want unstable thing laying around your PC. The person in front is already enough)

2. Make sure you've installed Wine. Type

`wine --version`

and you should see a version number appear if previous step went well or a crazy error message if you've messed something up.

3. Go to the directory where you extracted the sigma files using the terminal (the place where your `sigma16.exe` or `sigma16.exe.exe` is located)

4. In that directory type

`wine Sigma16.exe`

or

`wine Sigma16.exe.exe`

depending on which executable file you've got

5. That is it you're all done. Wasn't so bad was it?

2.1.1 Mac (OSX)

0. Ok, since mac is very secure (weird) and everything needs to be approved it is a bad env for development, unless you get some package manager like homebrew. If you do not already have homebrew run

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

in the terminal and follow the instructions.

1. Then install wine, using brew. Run

`brew install wine`

2. Navigate to the folder where you extracted your sigma files. The one with `Sigma16.exe` or `Sigma16.exe.exe`

3. Run

`wine Sigma16.exe`

or

`wine Sigma16.exe.exe`

depending on which executable you've got

4. Enjoy the sigma experience!

2.2 End of the section

Disclaimer: Congratulations! You are now officially running Sigma16. Please note that the product (both Wine and Sigma16 lol, but I've only participated in the development in one of them so do not blame me) is buggy and this guide takes no responsibility for any potential damage. Not that you are expected to receive some, but I'm just setting my insurance here.

Q: What are the pros and cons of running it with wine?

A: There are many reasons why you *do not* want to use wine in general. A good rule of thumb is use wine as a last resort. For you, a reason to not use wine might be that all your peers that did not use it, might have something to mock you about.

Pros:

- It's relatively easy to set it up
- You do not necessarily need the knowledge that full computer geeks have (wait for the 3rd moon in November when Pluto is casting shadow over one of it's moons then use an enhanced lithium crystal in order to deflect the right amount of light on your circuit so the build finally completes with no errors), when they prefer to compile everything from scratch.
- Even a Windows user will not be scared when they see the result of your execution (This can also be a Con)

Cons:

- It's just a bad way to do it.
- Essentially what you are doing is "emulating" a Windows environment and thus might experience slower response rate
- Because of the emulation and how sigma saves things, it might be a bit trickier to find where your saved item is (unless you saved it somewhere you knew was good), In any case if your all important item (AX) is missing, go see the troubleshoot section.

3 Running it the proper comp. sci. person (proper) way. Install it from source.

Disclaimer: Here be DRAGONS!!!! Proceed at your own risk. This sounds very scary so be warned you are gonna have to use 'the terminal' again, use a real programming language and possibly edit some config files! Shocking, if you've got weak nerves and have not yelled at your screen before, you probably do not want to continue reading this section. Ok enough fireside talk let's get to business.

3.1 Linux

1. Open the terminal. This should really be your habit by now... We'll only make use of this tool for the tutorial

2. Check if you've already got Haskell on your machine. Run

```
ghc --version
```

If you see a version number make sure it's 8.0.2 or newer. If you see a message saying you do not have Haskell on your machine... How dare you, after all the university came up with the thing, surely you should be a Haskell expert studying in university of Glasgow.

3. Install haskell-platform if you do not already have one. If you do, proceed to step 3. Run

```
sudo apt-get install haskell-platform.
```

4. Make sure your version of haskell is at least 8.0.2 (Run step 2 again) if it isn't bad times you are *doomed*, close your PC and run away... Or just go to the Troubleshoot section and *seriously* consider upgrading your linux-distribution -.- .

5. In your terminal navigate to the folder where you extracted your sigma files e.g /home/USER/Sigma16-2.4.7

6. Run

```
cabal update
```

If you do not have cabal at this point go to the troubleshoot section.

7. Run

```
cabal install --only-dependencies
```

8. Run

```
cp app/Main.hs src/haskell/Main.hs
```

9. Run

```
ghci
```

(at this point you should see the glorious haskell compiler's interpreter in action)

10. The next steps assume you are inside the ghci prompt

11. Run

```
:load Main
```

If at that point you see `src/haskell/SysEnv` error, you've messed up. Go to the troubleshoot section...

12. Run

```
main
```

13. That's it you've successfully interpreted Sigma16 on your own!!! Well... I'll just pretend I wasn't here. Now follow the onscreen instructions, open your browser and go to `http://127.0.0.1:8023`

14. If the page is constantly refreshing after you've done so go to the troubleshoot section

3.2 Mac (OSX)

You are in luck, since Brew decided to not support Haskell-platform anymore, because apple provided your favorite drag-n-drop installers you do not have to use 'The terminal' for the whole tutorial (just most of it).

1. Just follow the instructions under `https://www.haskell.org/platform/mac.html` to get your Haskell-environment going and follow the rest of the steps
2. Now that you have Haskell on your machine (hopefully a correct version) go to the linux section and follow the instructions starting from step 4. Turns out Linux and Mac are not all that different, who would have thought? Maybe it has to do something with the Unix kernel.

4 Troubleshoot

Note: If you do not see your problem here it might be worth to report it, or you might have hit an older version of the document. To check whether the latter is correct you can go to: `https://github.com/janev94/Sigma16-Extra/` and check whether a new version was submitted recently, or I've given up support for the document altogether :(

Issue: *I do not know how to open that scary thing called 'The terminal'?* Before you judge me for putting this up here - it's a true story, I am not gonna call out names, but some of you have asked me that in the past.

Resolution:

Linux: Now I am judging you... Usually, the terminal is bound to `ctrl+shift+T`, so try that if it is not, press your windows key and type 'terminal'. If that does not help as well, well use Google and find out how to open the terminal for your distribution of Linux.

Mac (OSX): Press Control + Option + Shift + T, if that does not do it find the icon of the terminal in your apps (looks like a black square, you've never clicked it before). If neither of these methods work, resort to Google.

Issue: *I do not know how to navigate to a folder in the terminal.*

Assuming you are in your home directory (the one that pops-up by default), use the **cd** command to navigate to a different folder. For advanced users, or adventure-seekers you can combine **cd** with a back-slash in order to change multiple directories (amazing stuff). For example, if you extracted your sigma in a directory called Sigma16-X in your Downloads folder you can go there, by running:

```
cd Downloads\Sigma16-X
```

from your home directory.

Tip for Mac users:

You can navigate to the directory you want to open your terminal window in with Finder, then click on the finder menu, go to services and select 'New Terminal at folder'.

Resolution:

Issue:

My haskell is 7.X.X, HELP!

I do not have Cabal, what should I do?

Resolution:

Do the ghcup magic. Cast spells, make potions, lift curses and pray for the best.

1. Go to your home directory

2. Run

```
( mkdir -p ~/.ghcup/bin && curl https://raw.githubusercontent.com/haskell/ghcup/master/ghcup > ~/.ghcup/bin/ghcup && chmod +x ~/.ghcup/bin/ghcup ) && echo "Success"
```

3. If you do not see "Success" printed at the end... You're in some deep trouble contact me or John

4. Install haskell 8.0.2. Run

```
ghcup install 8.0.2
```

5. Set 8.0.2 as your default haskell compiler. Run

```
ghcup set 8.0.2
```

6. Install cabal. Run

```
ghcup install-cabal
```

7. update the cabal-install. Run

```
cabal new-install cabal-install
```

8. Continue from step 3 in the linux section.

Issue:

*Web page is constantly **refreshing** I cannot see my Sigma in the browser.*

Resolution:

Well, well, well. Look who's back here. Now there, there, we'll help you again. Most likely, you've encountered an issue where your sigma misinterprets your data directory that is ok, we'll fix it.

1. Go to the folder where your extracted sigma is.
2. Run

```
EXPORT SIGMA16='pwd'
```

NOTE: The characters around pwd are called backticks! Make sure you use those and not quotes!

3. You might consider adding SIGMA16 to your .bashrc, unless you want to do step2 everytime there's this issue.

Issue:

*I've messed up :(. I see **src/haskell/SysEnv error** whenever I try to load Main.*

Resolution:

It's ok, probably your cabal is outdated. Just follow the instructions under "*My Haskell is 7.X.X, HELP!*"

Issue: *My Sigma does not do anything when I press "Assemble"*

Resolution: You might be experiencing one of the most rare 'features' of Sigma. To verify that this is the case open the terminal(console) window and check whether you get some message for invalid ascii characters. If you do, then you are one of the few chosen ones that get to have the full Sigma 16 experience! But seriously, it is a weird bug we're trying to trace down (unsuccessfully so far as you can see), but fear not, we've got the work-around nailed down! So what you need to do is:

1. Copy all the code you currently have on your clipboard (or save it to a file)
2. Press the 'Example' button
3. Go the the Assembler and press Assemble

4. Now that your sigma is all patched-up, simply delete the example and replace it with your code, which was not willing to assemble

Issue: *Sigma complains that `cmpgt/cmplt/cmpeg/jumpr/jumpr` are invalid instructions*

Resolution:

Now, this is usually version dependent. For example, if you are running 2.3.X you will experience Sigma pretending not to know what these instructions are (How dare you Sigma, the students know you are fooling them around). So your two options now are either:

1. Call your tutor and ask them to explain you how the `cmp/jumpr/jumprgt etc` instructions work (Oh no, learning more stuff!)
2. Go to the Moodle page and download a version that *does* understand these instructions, e.g. 2.4.X and use that to assemble your code.

5 Final Words

If you've found that useful, or have any critiques, comments or difficulties, feel free to contact me at m.yanev.1@research.gla.ac.uk. You know, just to say Thanks or Your guide was B*. Whatever you want, I am lonely and *definitely* do not get a lot of email, but seriously contact me, I'll try and help you.

Note: I know LaTeX was probably a bad choice for the 'prettier' document. If you have found any troubles copy-pasting from this document, let me know and I might consider migrating it to something more clipboard-friendly.