

Fall 2012 ECE498 Linux Assembly Language – Homework 3

Due: 18 December 2012, 6pm

This homework involves some SNES programming. This is meant to be a simple and fun homework; you should concentrate on Project 1 rather than spending too much time on this.

1. You will need to build the ROM image. The tools to do this (the cc65 assembler / linker) are available on the ARM pandaboard machine.

- Log into the ARM pandaboard machine (in umelst that's an L):

```
ssh username@vincent-weaver-1.umelst.maine.edu
```

Where “username” is the username on the slip of paper I gave out in class (if you missed class please stop by my office for your account) and the password is also from the slip of paper.

- Download the code from:

```
http://www.eece.maine.edu/~vweaver/classes/ece498asm_2012f/hw3_code.tar.gz
```

You can use `wget` on the ARM machine to do this:

```
wget http://www.eece.maine.edu/~vweaver/classes/ece498asm_2012f/hw3_code.tar.gz
```

- Uncompress/unpack it with the command `tar -xzvf hw3_code.tar.gz`
- Change into the `hw2_code` directory `cd hw3_code`
- Run `make` to build the code.

2. The “sprite.sfc” (sfc = super fami-com, the Japanese name for the SNES) file is created. You can load this into an emulator. You can install an emulator on any local machine you have; `snes9x`, `zsnes`, and `bsnes` should all work. If you cannot do this, `bsnes` is installed on the pandaboard. You can in theory `ssh` in with the `ssh -Y` option to enable X11 forwarding and run `bsnes` from there, but it will likely be slow.
3. Run the “sprite.sfc” example. It should say Hello World on an ugly red background, with a sprite going back and forth. If you press and hold a joystick key (emulated by enter, spacebar, etc) the sprite should stop moving.
4. Part 1 is to change the “HELLO_WORLD” message to something of your choosing.
5. Part 2 is to make the sprite move vertically as well as horizontally. The code that moves the sprite is in the `VBlank` routine. You can duplicate this code. In addition to writing the X value to address \$0000 write the Y value to \$0001.
6. That's all! When done run “make submit” and send me the resulting file.
7. Feel free to do additional changes, but they won't be counted extra. This can include things like changing the sprite image (it's generated from “sphere.pcx”), or changing the background tile or color, or more.