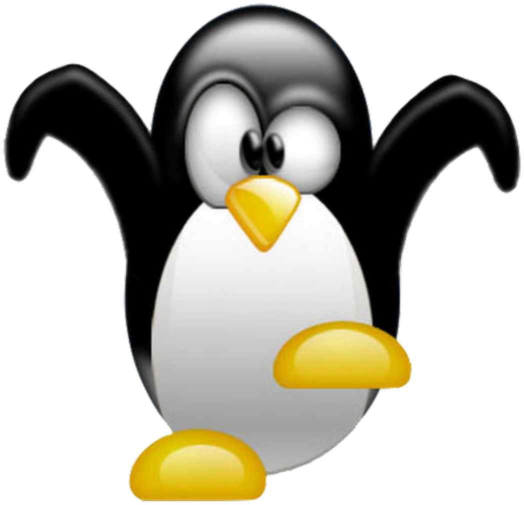
Managing Linux with the Embedded Perspective

Exercise 1

Joseph Hotchkiss

August 19, 2025



Introduction

Before we can investigate developing software for embedded systems, we will need to setup our development environment. In order to do this, we will need to be running Linux or at least a POSIX compliant shell. If you are running Linux natively, congratulations you are ready to proceed. If you have a Mac, you can either use a VM(Not really recommended) or you can use the Mac's terminal app (Or your preferred terminal emulator) to carry out your exercises. If you are on Windows, you can use a VM, but I would think it is really much better yo install WSL (Windows Subsystem for Linux). In structions can be found from https://learn.microsoft.com/en-us/windows/wsl/install

Tools

For this course we will need a fair few tools. The tools required:

1. C/C++ Compiler and Libraries
2. A terminal Text editor
3. GDB
4. make
5. CMake

One of your most used tools with be your shell. You are free to choose a POSIX compliant Shell for this course. Bash or zsh are good options. You can decide what you want to use. Once you have decided on a shell it would be a nice idea to customize it to better work for you. You can customize some behaviors of your shell by using a tool like Oh-my-bash https://github.com/ohmybash/oh-my-bash or oh-my-zsh https://ohmyz.sh/. You can further customize your shell, by using ohmyposh https://ohmyposh.dev/ for all shells, or powerlevel10k for https://github.com/romkatv/powerlevel10k zsh. The idea here is to make the shell do some work for you. You might be required to add some nerd font https://www.nerdfonts.com/ to get powerlevel10k or ohmyposh to render correctly.

Depending on your choice of text editor, you may need to do some configuring. For example if you use Vim or Neovim, this is a pretty bare bones experience without customizing it. Kickstart.nvim is a pretty good starting point for setting up Nvim for yourself. There are a plethora of resources to get started with vim/Nvim motions but vimtutor is a fine way to get started https://vimschool.netlify.app/introduction/

vimtutor/. Once you have finished, test things out. Get familiar with the features that you have enabled and then write a shell script called internetaddress.sh that gets the ip address of your pc. If you invoke internetaddress -v it will output your IP address, subnetmask, and broadcast address. You will then need to make the script callable anywhere(as shown in the example in class).

Documenting

Document all the steps that you have taken to achieve the requirements of this exercise. Screenshots are a great way to demonstrate that things are working.

Submission

Take your time and setup a terminal to your liking. Configuring the terminal makes it more pleasant to use. Submit your document to Oma once you have completed your setup.