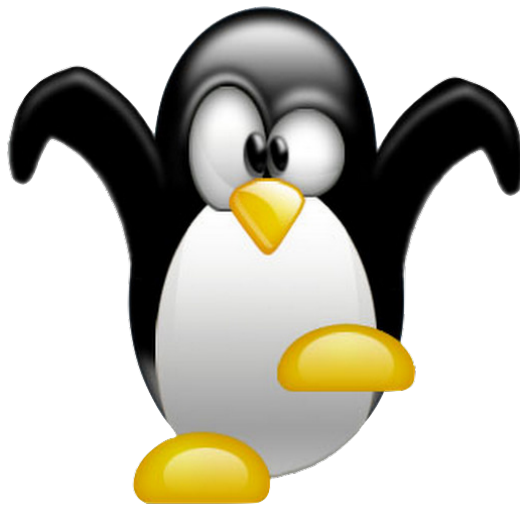


Managing Linux with the Embedded Perspective

Exercise 4(CMake)

Joseph Hotchkiss

January 16, 2025



Introduction

Exercise 2 required you to write a C program with multiple files. You were asked to compile and link the code with from the terminal and you were supposed to use the different compiler and linker flags. You have also used GNUmake utility to build this with the same flags.

CMake

Use CMake to generate files so that you can build your unique random project. Setup your project as I did in the lecture. Have a separate directory for your .c files, a separate directory for your .h files, and a separate directory for your Build output from CMake.

Adding options and Settings

Once your CMake workflow is working and you are able to build your code let's add some things to the CMakeLists.txt file

make sure that CMake is using c 17 standard and that your cmakeLists.txt file contains required targets to be able to use CMake to create a makefile to build your application.

Make sure to add your compiler flags from the previous exercises

Take a look at the CMake docs and see if there is anything else that you might want to experiment with

Please note that we are only brushing the surface of what is possible with CMake, but hopefully this gives you an idea and you will be able to see the use of this tool.

Return

Return your cmakeLists.txt file to oma.