

Mplab

author: andrew lycas

alright so now u should have a basic understanding of the C language and some of the syntax. If you feel ur not ready.. then dont read on.. but if u think u can move on.. well then this is the article to read.

For robotics we will be using the program mplab and the mcc18 compiler. Mplab is a IDE, or an integrated development enviornment. Dev-C++ was an IDE, and it used the gcc compiler. Due to that right now i do not have access to mcc18, i cannot post it on the website.. but eventually(ill post it on the homepage of the site) i will have the CD posted. (note: when it is posted, it is only to be used for robotics and learning purposes, any other use of it will ont be the fault of Andrew Lycas nor of Langley Robotics at all, so please.. if ur from another team, or not even in first, i am flattered that ur reading this.. but dont download the compiler.. its not legal)

Go ahead and download and unzip the cd folder.

So, just follow all of the instructions to install it, you should be able to install stuff by now..

Also head over to <http://www.ifrobotics.com/docs/frc-code-2-28-2006.zip> and unzip/save it some where..

alright so once mplab is installed...open it up

then go to file->open workspace and navigate to where you have FRC\_DEFAULT saved, and open up the file that it shows, which is FrcCode.mcp .

So on your left is the listing of all of the files in this project. It is including all of the header files or the libraries, the source files, which are what we have been doing before, and other things various critical files. Most of these files we do not even touch, many are there because it is required for FIRST to be able to shut it off and such at the competition.

Some of the files we will be editing later:

user\_routines.c

user\_routines\_fast.c

in those file, the code for all of the autonomous code, and all of the user control code. By say that those are the only two files we will edit, i am not saying that it is not good to look throught the rest of the files.. If you get the chance, look thorough some of the files... read the comments, see how much of it makes sense to you. Also look at some of the header files. To be a good programmer you should know how all of the code works, rather than just the part that you can edit. Make sure tho to not edit anything that says DO NOT EDIT.. kinda stupid to remind you but seriously if anything is changed that it says to not edit... we're screwed.. so yea.

Next lesson i will get into pwms, and spikes : wtfak are they and how do u controll them.