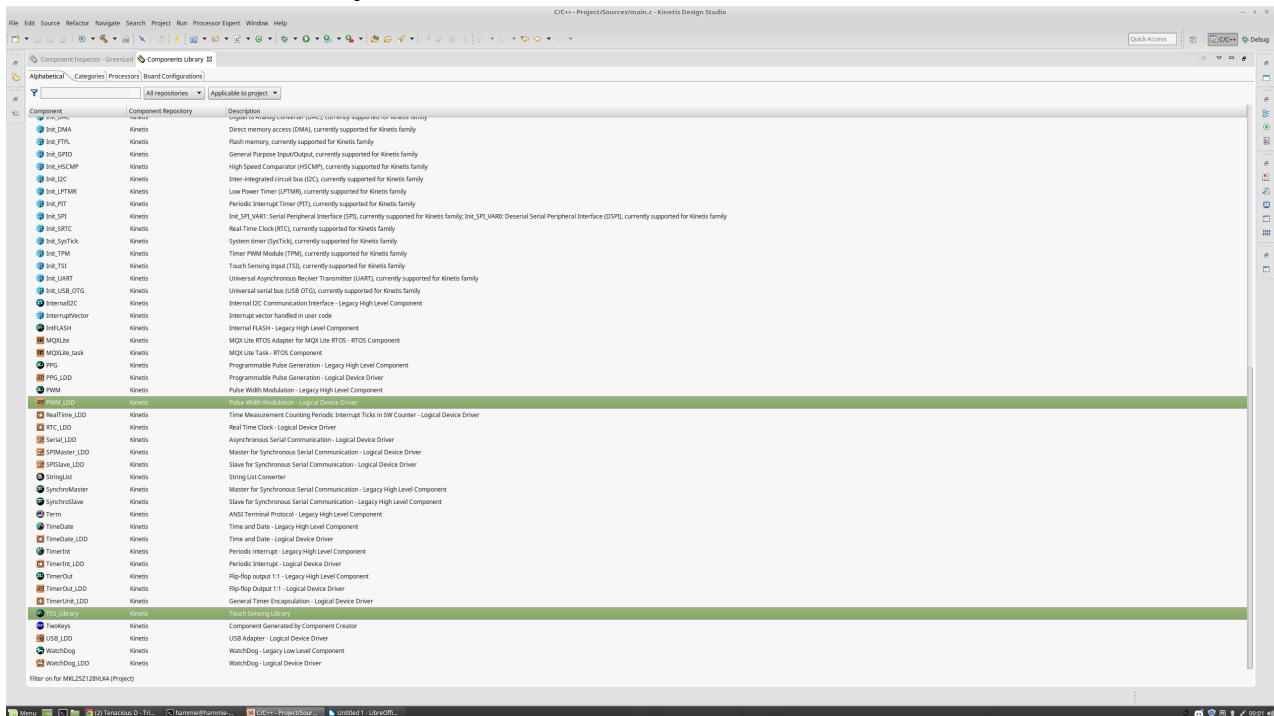


Set up PWM in processor expert.

Skip the first bits if you're familliar with processor expert (usefull stuff at the back)

To begin create a file making sure to include processor expert.

Navigate to processor expert components library (The big IC logo) and add 3 **PWM_LDD** modules and 1 **TSS_Library** .



Now open up any of the PWM libraries and input the values in the table below

When doing your second or third it will ask you if you would like to use a pre existing timer or make a new one!! **Be sure to have the red and green on the same timer but the blue on a separate one!!**

Pin	Period	Pulse Width
PTB18(Red)	500Hz	1ms
PTB19(Green)	500Hz	1ms
PTD1(Blue)	500Hz	1ms

Now the Touch slider:

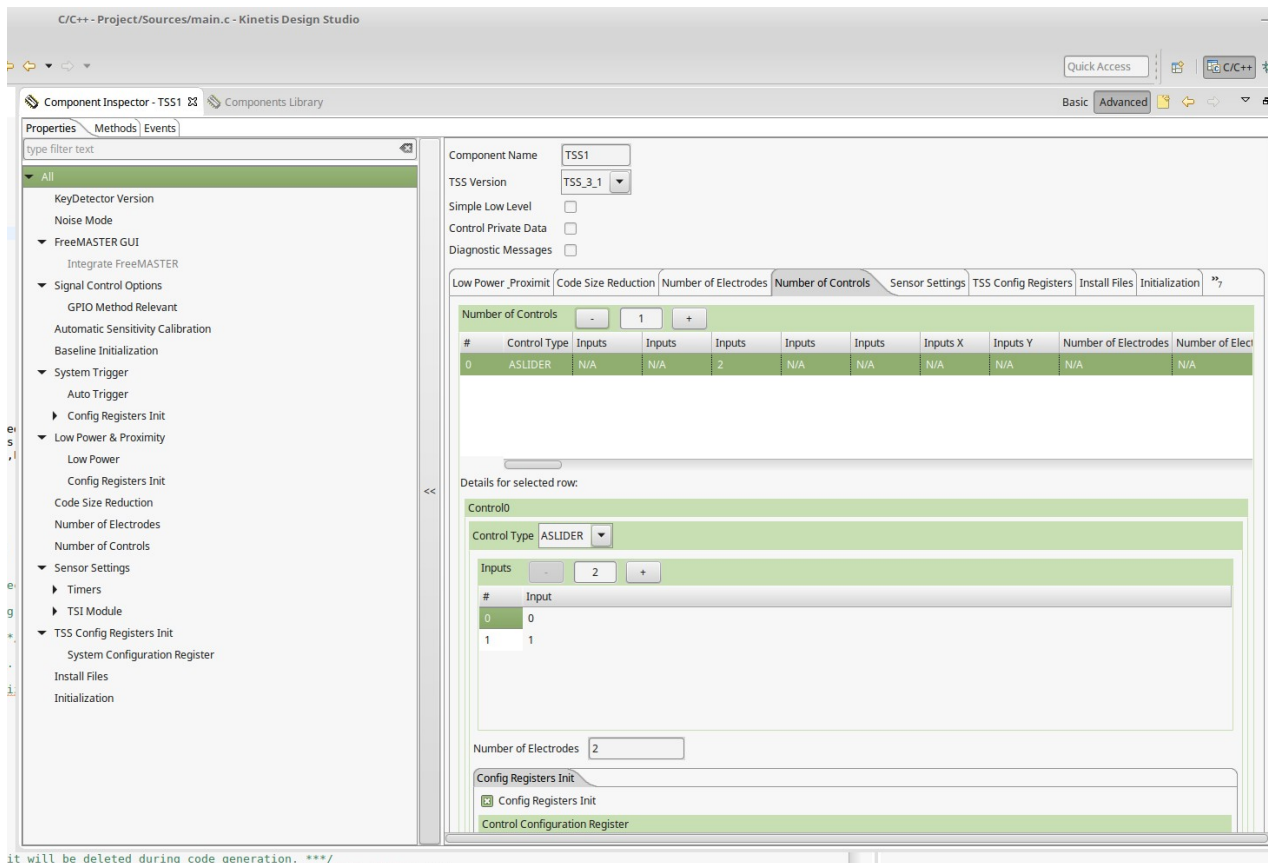
Open it up in component inspector and first of all go to the **initialisation** tab and **check both boxes**
Then go to **number of electrodes** and set it to **two**:

Electrode0 – PTB16/Channel9

Electrode1 - PTB17/Channel1

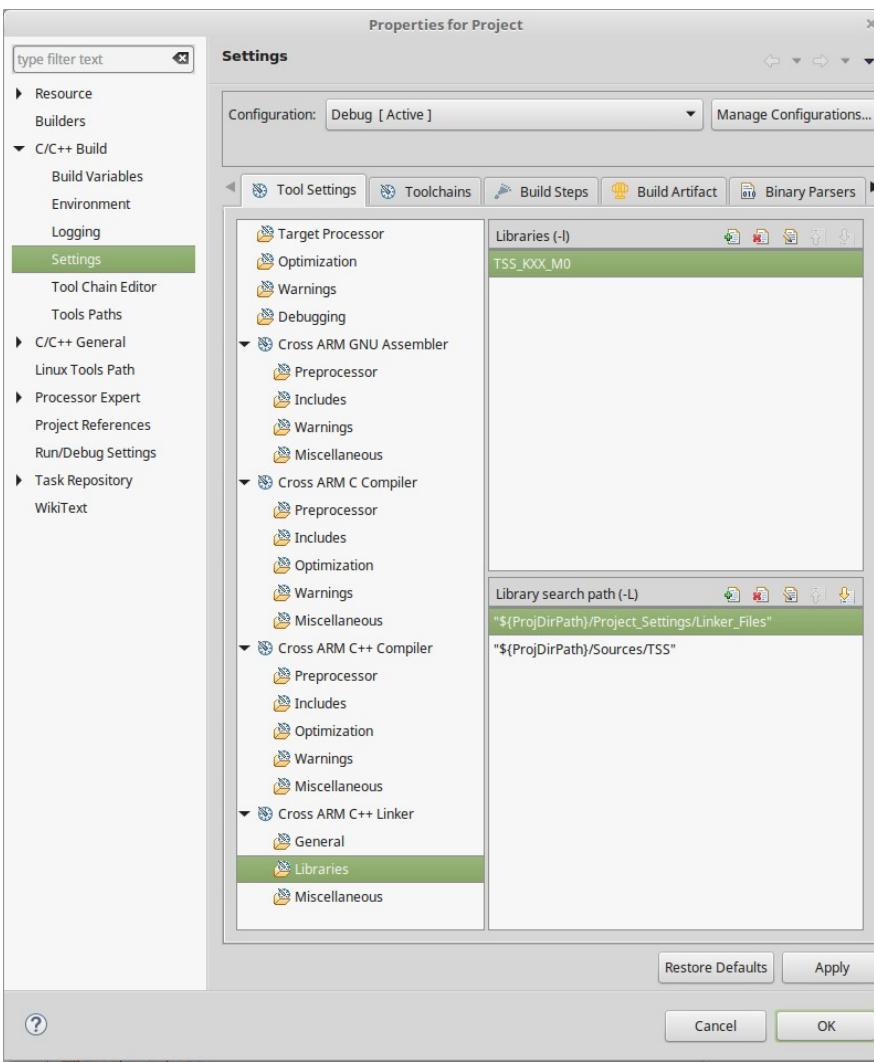
Now go to **number of controls** and set to 1

set **control 0** to **ASLIDER** for control type with **2 inputs** and change these inputs to **0** and **1** respectively (unfortunatley kinetisis bugged and you will have to restart to do this)



Touch slider fixes:

Now go to **project** in the navbar at the top of the page and select **properties** (you may have to be in the c/c++ view to do so) under **c/c++ build** select **settings** and then **libraries** under **cross ARM c++ Linker** and add



Library: TSS_KXX_M0

Library path: "\${ProjDirPath}/Sources/TSS" as shown

Finally go to optimisation under CrossARM C Compiler and change the language standard to **Toolchain Default**.

Now you can program!!!

Useful things:

name_SetRatio16(name_DeviceData,Value);

This is what you will use to set your PWM value to your LED where value is(0-0xFFFF(65535)) and name is what you called your pwm module.

_asm("NOP");

This strange piece of code will pause your program for 1µs meaning 1000 is 1 millisecond (theoretically) so make your own delay function using it.

static void name(int input) {}

Declare these outside your main to be able to constantly call something which cleans up your code and makes it re-useable. Simply write why you want to repeat do (such as a delay) inbetween the brackets. If you wanted this to return something you can change void for int or char or whatever you want to return. The int input in the brackets means when this function is called it asks for a value (like delay(500)) and this value is locally stored meaning you can access it only within this static void. (use these for good programming marks).

TSS_Task();

This simply asks the touch sensor what its value is and stores it in TSS1_cKey0.**Position** so call that whenever you want to refresh the sensor and then call TSS1_cKey0.**Position** when you want to check its value.

Hint!

Don't forget arrays exist.

Good luck and happy hacking!