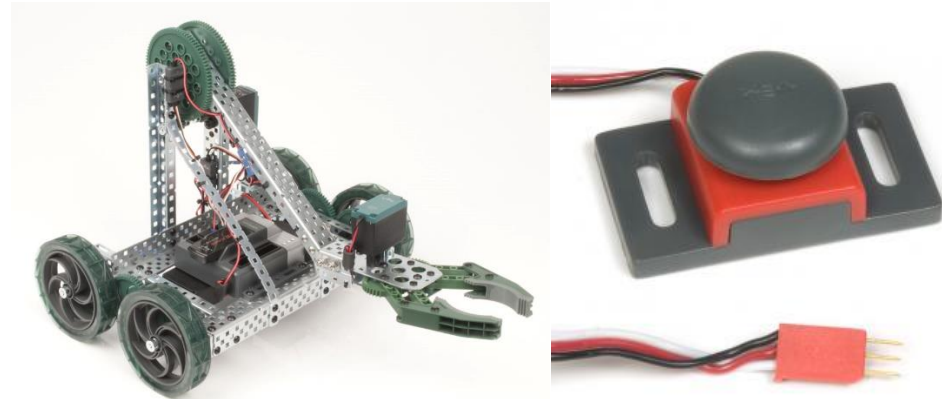


Marist School Robotics Teams
Directions for Writing TouchStop Function
ClawBot Training Robot



Description:

These directions will take you through writing a function to read the digital Touch sensor. The algorithm for most sensor based robotics is:

0. Set up any variables or initialize sensors (Not done with this function)
1. Start an action. (Turn on motors . . .)
2. Wait for a condition to be true. Measure the sensor values.
3. When condition is met – stop the action from step 1.

While these directions work with the touch sensor, the algorithm above can be applied to many different types of sensors (Light, Potentiometer, Sonar, Encoders).

Touch Sensor: Digital Sensor (Returns 0 or 1)

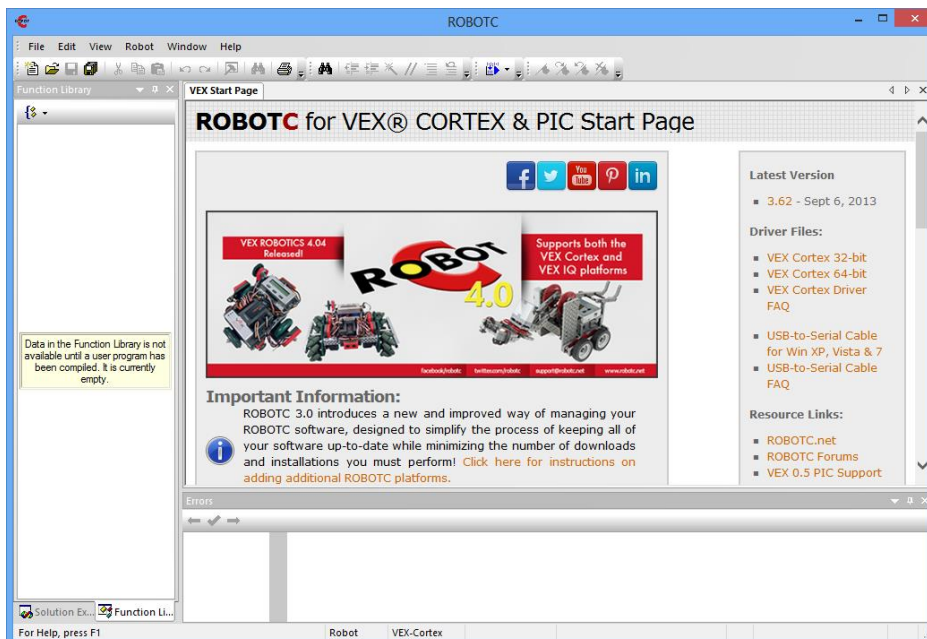


Digital sensor
Plugs into any dgtl port
(Black Cable Out)
Returns Boolean value or short values of 0 or 1.

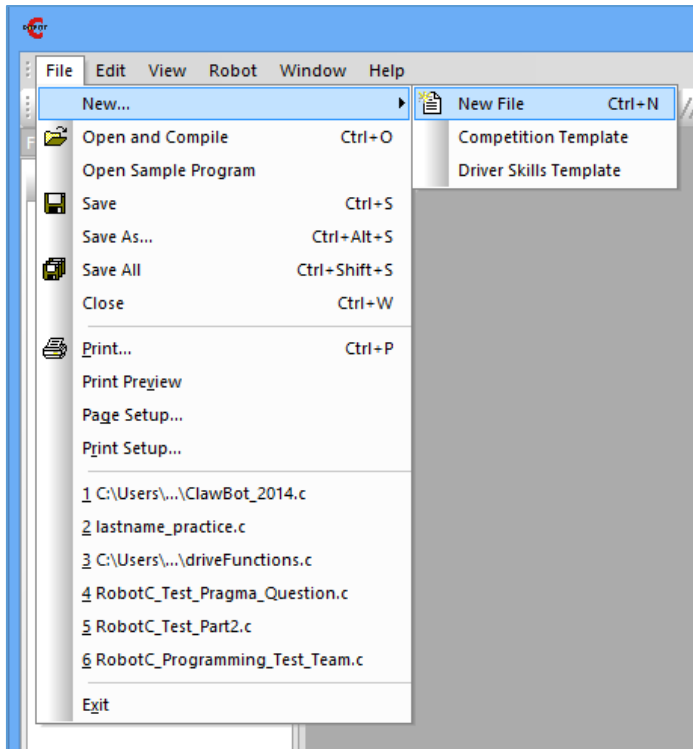
```
bool touched = SensorValue[touch];
```

Process:

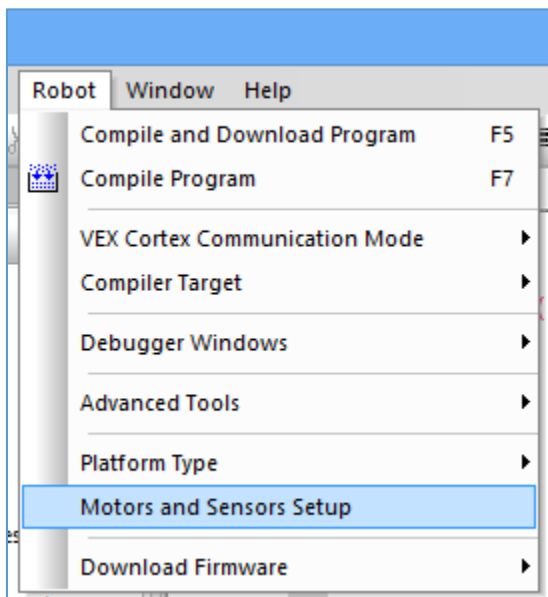
1. Start RobotC on your computer. (You may download RobotC at http://downloads.robotc.net/vex/ROBOTCforCortexPIC_362.exe)



2. Select “File – New - NewFile” from the Menu Bar. This creates a new RobotC program.



3. We now need to set up the Motors and Sensors for the Robot. We will use the “Motors and Sensors” setup to set the Pragma Code for the program. Select “Robot – Motors and Sensors Setup” from the Menubar.



4. Click on the “Motors” tab and label the motors for the following:
- port1: right: 393
 - port8: claw: 393
 - port9: arm: 393
 - port 10: left: 393: reversed

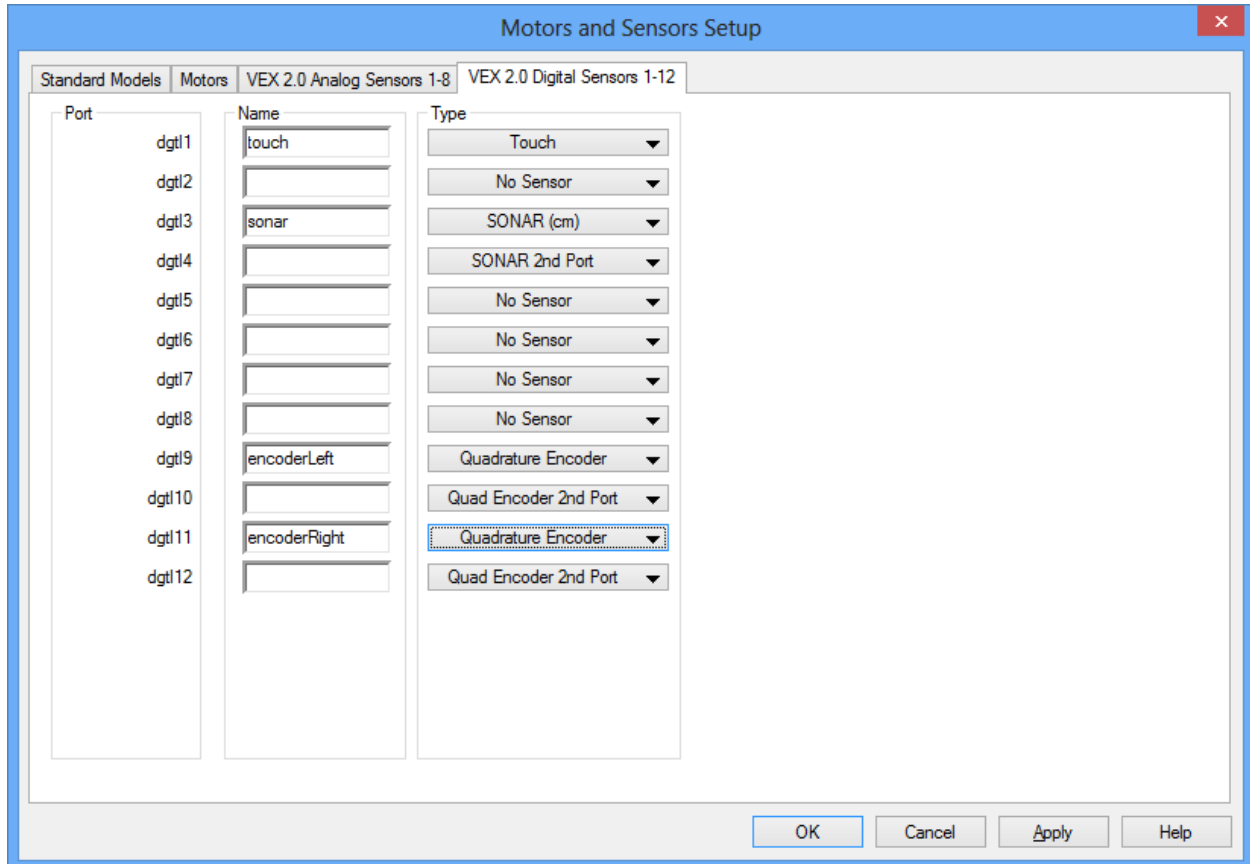
The screenshot shows the 'Motors and Sensors Setup' dialog box with the 'Motors' tab selected. The dialog is divided into several columns: Port, Name, Type, Reversed, Encoder Port, and PID Control. The configuration for each port is as follows:

Port	Name	Type	Reversed	Encoder Port	PID Control
port1	right	VEX 393 Motor	<input type="checkbox"/>	None	<input type="checkbox"/>
port2		No motor			
port3		No motor			
port4		No motor			
port5		No motor			
port6		No motor			
port7		No motor			
port8	claw	VEX 393 Motor	<input type="checkbox"/>	None	<input type="checkbox"/>
port9	arm	VEX 393 Motor	<input type="checkbox"/>	None	<input type="checkbox"/>
port10	left	VEX 393 Motor	<input checked="" type="checkbox"/>	None	<input type="checkbox"/>

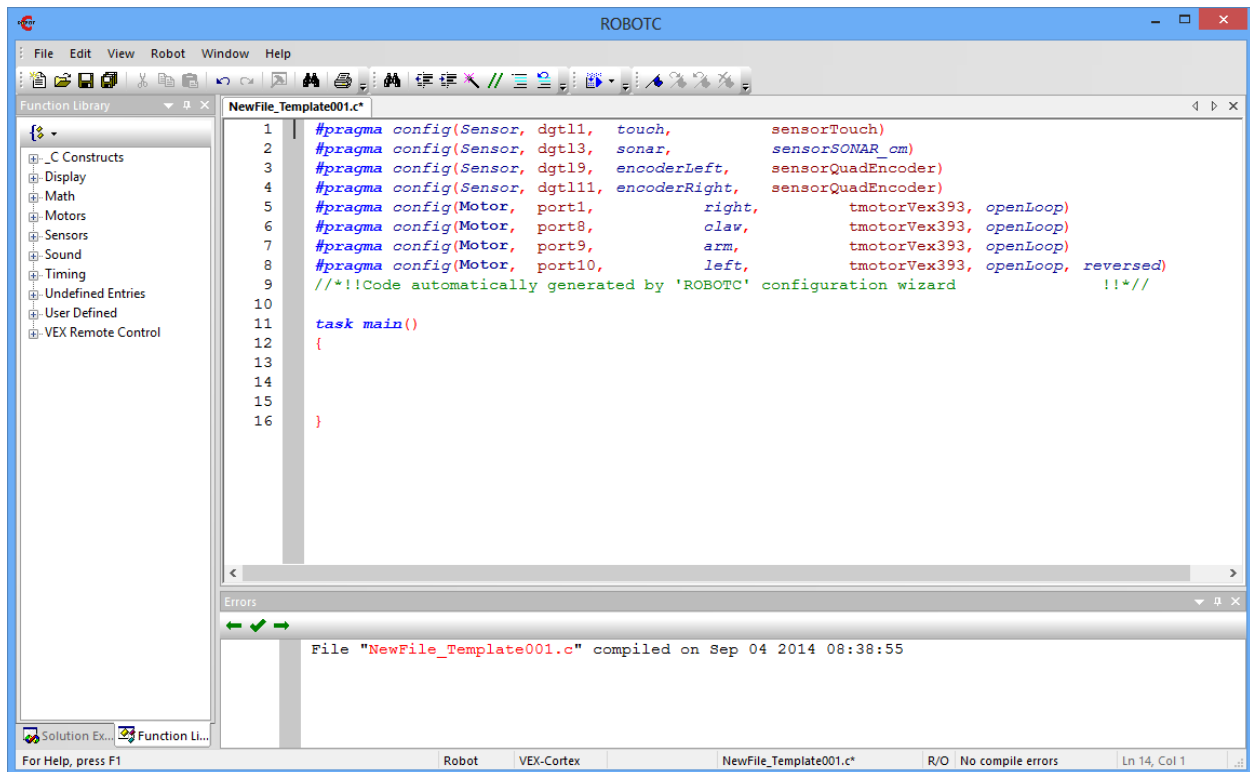
At the bottom of the dialog, there are four buttons: OK, Cancel, Apply, and Help.

5. Click on the VEX 2.0 Digital Sensors 1-12 tab and set up the sensors:

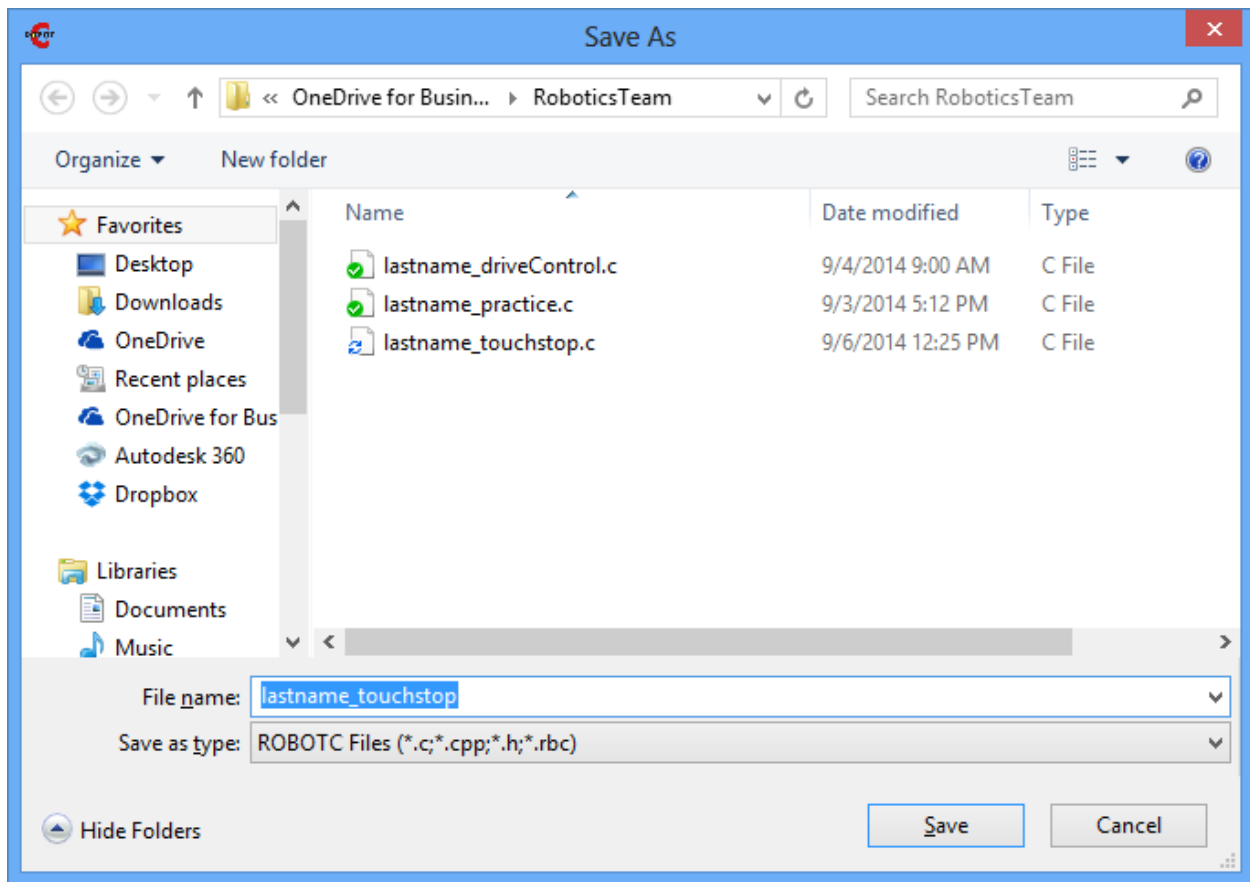
- dgtl1: touch: Touch
- dgtl3: sonar: SONAR (cm)
- dgtl9: encoderLeft: Quadrature Encoder
- dgtl11: encoderRight: Quadrature Encoder



6. Click 'Apply' and 'OK'. Your code should look like this:



7. Select “File – Save” and save your program to your OneDrive for Business as “lastname_touchStop”



8. We now need to write the function for touch stop. Begin to define the function as shown below. Note the comments describing each portion of the sensor algorithm.

```
10
11 // Touch Stop Function
12 void touchStop(int p) { // p is for power of motors
13     // Step 1: Start an action: Motors on for p
14     motor[right] = p;
15     motor[left] = p;
16
17 } // end touchStop
18
19 task main()
20 {
21
22
23
24 }
```

9. Write the code inside the function for Step 2: Wait for condition. Note the use of the 'while' statement.

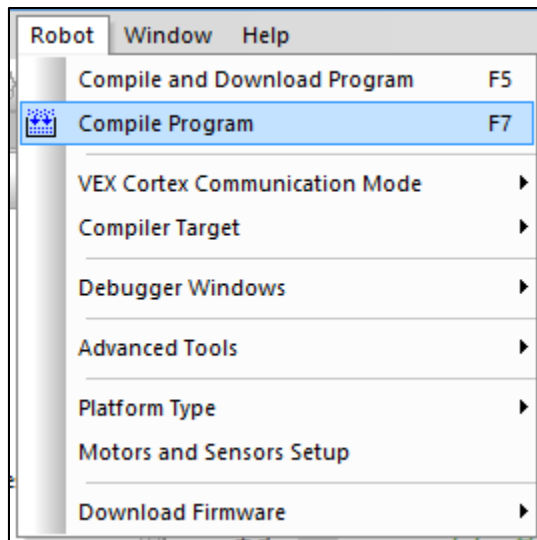
```
10
11 // Touch Stop Function
12 void touchStop(int p) { // p is for power of motors
13 // Step 1: Start an action: Motors on for p
14 motor[right] = p;
15 motor[left] = p;
16
17 // Step 2: Use while to react to touch
18 while (SensorValue[touch] == 0) {
19 // Do nothing here . . waiting for touch
20 }
21
22 } // end touchStop
23
```

10. Finish the function with Step 3; stopping the motors after the touch sensor completes the while loop.

```
11 // Touch Stop Function
12 void touchStop(int p) { // p is for power of motors
13 // Step 1: Start an action: Motors on for p
14 motor[right] = p;
15 motor[left] = p;
16
17 // Step 2: Use while to react to touch
18 while (SensorValue[touch] == 0) {
19 // Do nothing here . . waiting for touch
20 }
21
22 // Step 3: Stop the motors after loop
23 motor[right] = 0;
24 motor[left] = 0;
25
26 } // end touchStop
27
```

11. Save your work by selecting 'File – Save'.

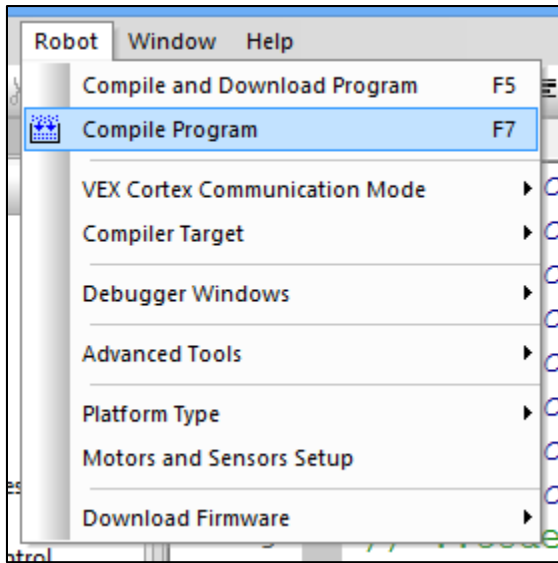
12. Select 'Robot – Compile Program' to compile and check for errors.



13. We need to test the touchStop() function. In the task main() call the touchStop() function with a power level of 75.

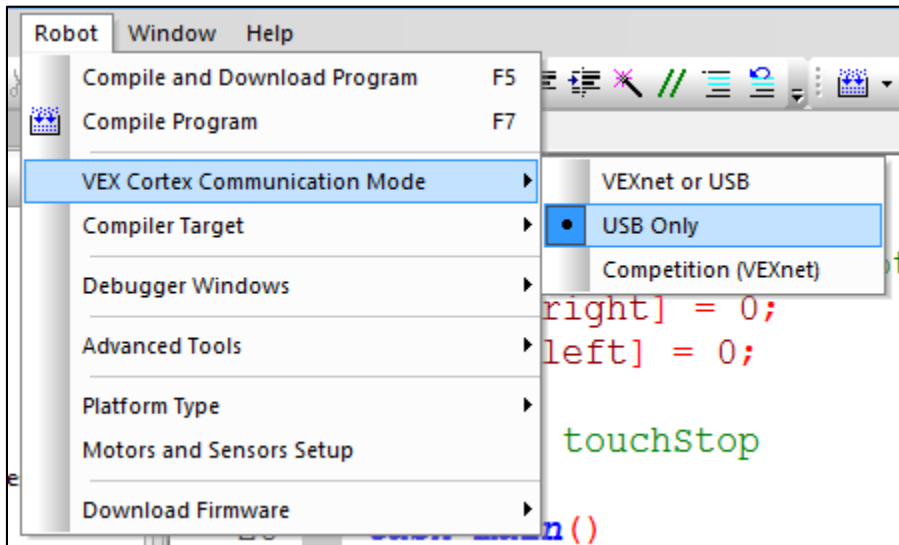
```
27  
28     task main()  
29     {  
30         // Test the Touchstop function  
31         touchStop(75); // Motors at 75 power  
32  
33     }  
34  
35
```

13. To check for errors, select “Robot-Compile Program” from the Menu Bar.

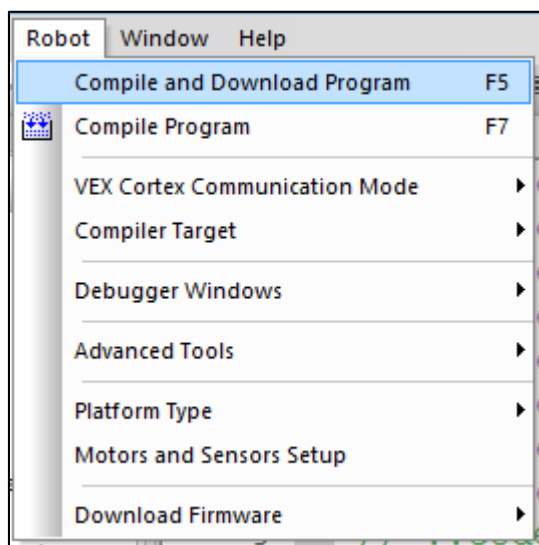


14. If the program compiles – you will not have errors (red x marks next the numbers). Now it is time to load the program in the Robot.

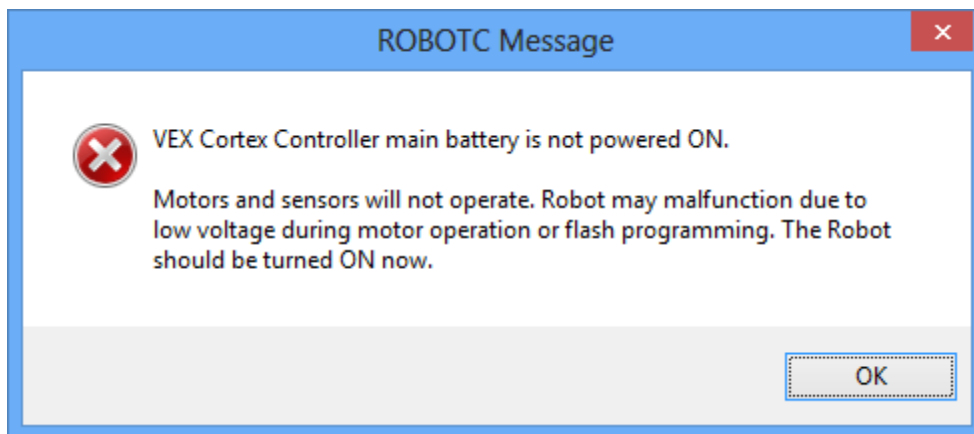
15. First, set the communication mode for the Robot to be “USB Only” by going to ‘Robot – VEX Cortex Communication Mode – USB Only’ on the Menu bar. This allows the robot to run the program without having to plug into the computer or a Vex Remote.



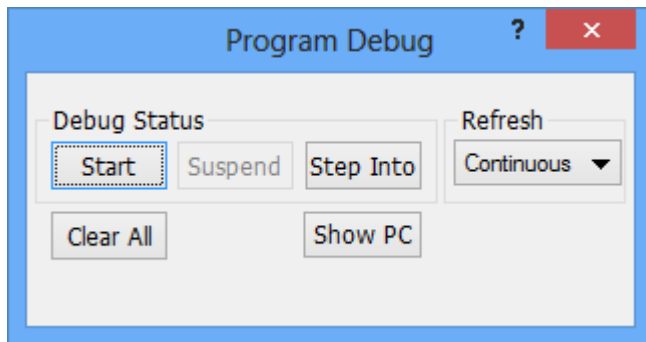
16. Take your ClawBot and make sure you have the following:
 - ClawBot
 - Orange USB Cable
 - Blue Battery Pack (7.2 Volts, 3000 mAH)
 - Confirm that the Touch Sensor is plugged into Digital Port 1
17. Battery into the VEX Cortex brick but do not turn on.
18. Plug the orange USB cable into the Cortex and into your computer.
19. Select “Robot – Compile and Download Program” from the Menu Bar.



20. You might get a message about the Robot not being switched on. Click OK at this message:



21. The Program Debug window will open. To test the robot at this stage, hold the robot in the air and be careful not to have wires in fingers in the gears. (Propping the Robot on a stack of books works well for 'bench' testing). Turn the robot on. Press the 'Start' button icon on the 'Program Debug' window to start the program. The robot wheels should spin until you press the touch sensor plugged into digital port 1.



22. To test the robot without the orange cable; Close the 'Program Debug' window. Turn the robot off. Unplug the robot from the computer. Place the robot in an open space and turn the robot on. It should run the program directly. Press the touch sensor to stop the program.

23. If the robot moves forward until the touch sensor is touched – the robot and function passed the touch sensor test!