**Set-up Timer for Sonar Lab Notes**

This is an outline for the Sonar Lab 10. We will use the Enhanced Capture Timer (ECT) Module data sheet, TIC/TOC Timer Lecture PPT, and the CodeWarrior MC9S12XEP100.h file.

Write the Sonar code and make sure it compiles before adding the LCD files.

**Set up I/O PORTS:**

DDRT = 0x40 // our only output: the speaker is at bit 6 of timer.

**Initialize the timer for Input Captures on rising edge.**

1. Write the command to enable the timer.

2. Set bits (Channels) 6 & 7 as an input capture bits (channels).

3. Setup the above captures for rising edge inputs.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| EDG7B | EDG7A | EDG6B | EDG6A | EDG5B | EDG5A | EDG4B | EDG4A |

OR: set/clear them all at once:

1. will need to slow down the timer clock speed by writing to the prescale bits in

register ECT\_TSCR2 (data sheet page: 172) I will show you this in class

**End of Timer Initialize**

**Sonar: inside the main.c for(;;) loop or your own for loop, while(1) etc..**

1. **Make sure** sonar is not running. Turn it off at bit 6.
2. **Check for pushButton** press. If high keep waiting. Use a while loop.
3. **Just after the** while loop put in a debounce delay for 10ms. Can use the LCD delay.

Delay(20);

1. **Wait for pushButton relase**. If low keep waiting. A release will result in a one. So we wait till a one.
2. **Once we have gotten** the press and release, then start the Sonar time. You should call a function.

SonarStart();

1. **Reset counter** bits 7 and 6 by clearing the bits 7 and 6 capture flags
2. **Turn on sonar speaker to send out single (bit 6)**

End of SonarStart function

1. **Now we can check TC6 and TC7** and get the raw delta time. Call a function to do the checking and return the raw delta time. SonarCheck( );
2. Wait for echo at bit 7 to return (go low)
3. Read the counts for the bits 6 & 7 capture

**Check if timer turned over during** capture and make corrections. 0 > 0xFFFF.

if start > end // end < start either way works.

1. Return the raw delta time.

Next is the most difficult: taking the raw count and using integer algebra scaling it to Feet and Inches.

Once that is done we can use the string print function *sprintf()* to format a string and then send the formatted string to our LCD printString function.

So, that is a general outline: Thursday, how to set the timing and how to scale and format our raw Count