Timers, Counters, and Pulse Width Modulation

Objectives

The purpose of this lab is to become familiar with the 68HC11 on chip timer/counter system. In part 1 of this lab, you will program the Real Time Interrupt system. In part 2, you will learn to use the compare register(s) to generate a waveform of known duty cycle and in part 3 use that knowledge to implement a Pulse Width Modulator. Lastly, you will use part 2 to implement a time-keeping function and display how long the Motor Control program has been running.

Turn-In Requirements:

1. Pages 2, 3, 4 and answers to the 4 questions as specified in the lab policies handout.

References


Equipment for this lab:

- 68HC11 trainer kit, to include 68HC11 EVBU and prototyping strips.
- IBM compatible PC to connect to the trainer kit via an RS-232 serial cable.
- Agilent 54621D oscilloscope.
- Floppy disk provided by student.
*In completing this lab, I have abstained from any form of academic dishonesty or deception (e.g. cheating, lying, stealing or plagiarism). This work is entirely of my own origin (beyond the code that was supplied to me by the instructor).

Name: __________________________            Signature: ___________________________

## Laboratory Exercise

### Notes:

- Make sure your development EVBU board is connected to power (green LED on board ON) and the serial port of the EVBU is connected to the serial port of your development PC containing the AXIDE software, configured to the correct port at the correct baud rate, etc.

- Be sure to bring your control.RTF file from 68HC11 lab #2.

- The lab TA has a copy of the .s19 files for each part of the lab. The TA will demonstrate the waveforms to you at the beginning of the class.

### Part 1

In this part, you will program the Real Time Interrupt (RTI) system to generate interrupts at a known rate, and use the interrupts to generate a waveform of known frequency. Please read pages 382 - 385 of the 68HC11 reference manual for a detailed description of the RTI system. The bits “RTR1” and “RTR0” in register “PACTL” determine the pre-scaler for the clock input, and the bit “RTII” in register “TMSK2” enables the RTI interrupts. The bit “RTIF” in register “TFLG2” indicates that a RTI has occurred. Note that the clock oscillator frequency of the Axiom board is 8.0 MHz.

Write a program to generate real time interrupts at 32.7 mS intervals and create a square wave of 15.25Hz, 50% duty cycle using the interrupt service routine at PORT D. Use the Agilent 54621D oscilloscope to monitor the waveform.

Fill in the values of the “PACTL” and “TMSK2” registers below. Disable all the other interrupts except RTI. In addition to enabling Real Time Interrupts, you also need to globally enable all the maskable interrupts.

```
PACTL $1026
  DDRA7 | PAEN | PAMOD | PEDGE | 0   | 0   | RTR1 | RTR0
  X     | X    | X     | X     | 0   | 0   | X    | X

TMSK2 $1024
  TOI  | RTII | PAOVI | PAII | 0   | 0   | PR1  | PR0
  0    | X    | X     | X    | 0   | X   | X    | X
```

Please note that the interrupt vectors are re-mapped in the Axiom board to lower RAM locations. You can find these locations in page 23 of the “CME11E9-EVBU development board” documentation under “Buffalo Monitor Interrupt Jump Table.” For example, RTI is relocated at $00EB - $00ED, and you should put a “jmp int_RTI” at this location. (“int_RTI” is the name of the Real Time Interrupt service routine.)

Demonstrate to the TA that your program is working.
Part 2

In this part you use the output-compare function of the 68HC11 to generate waveforms of a known duty cycle. Please read pages 372 - 373 of the 68HC11 reference manual for a detailed description of the output-compare function.

An output compare is generated when the free running counter of the 68HC11 matches the value of “TOCx” counter. This signal can be used to generate an interrupt if the corresponding interrupt enable bit in the register “TMSK1” is set. The “OMx” and “OLx” bits in register “TCTL1” determine the state of output pin “OCx” when the free running counter matches the “TOCx” counter. Refer to Table 10-6 on page 413 of the 68HC11 reference manual for details.

Program the output-compare channel 3 (OC3) to generate a waveform of the following characteristics at pin PA5/OC3.

<table>
<thead>
<tr>
<th></th>
<th>High</th>
<th>Low</th>
<th>Period</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.5</td>
<td>1.0</td>
<td>2.5</td>
</tr>
</tbody>
</table>

Duty Cycle = High/Period = 60%

You need to program the “TMSK1” and “TCTL1” registers with the appropriate values to generate the waveform. Enable the Output Compare 3 Interrupt in the “TMSK1” register, and disable all other interrupt enable bits in the register. Program the appropriate bits in the “TCTL1” register to generate the required waveform at the PA5/OC3 pin.

TCTL1  $1020

<table>
<thead>
<tr>
<th>OM2</th>
<th>OL2</th>
<th>OM3</th>
<th>OL3</th>
<th>OM4</th>
<th>OL4</th>
<th>OM5</th>
<th>OL5</th>
</tr>
</thead>
</table>

TMSK1  $1022

<table>
<thead>
<tr>
<th>OC1I</th>
<th>OC2I</th>
<th>OC3I</th>
<th>OC4I</th>
<th>OC5I</th>
<th>IC1I</th>
<th>IC2I</th>
<th>IC3I</th>
</tr>
</thead>
</table>

The following method to generate the waveform is suggested.

- Allocate 3 memory locations for High, Low and the Period.
- Initialize these locations to the appropriate values.
- Enable Output Compare 3 Interrupts.
- Within the interrupt service routine, change the value of the “TOC3” counter, so that the next compare occurs at the appropriate time.

Part 3
In this part, you will generate a PWM signal by changing the duty cycle of the waveform generated in part 2. This concept of PWM will later be used to control the speed of the DC Motor using the interface that was developed in the first two labs. Use the Real Time Interrupt generated in part 1 to modify the duty cycle of the waveform generated in part 2 from 5% to 95% in increments of 1%. That is, modify the values of “High” and “Low” inside the Real Time Interrupt service routine. When the duty cycle reaches 95%, then reset it back to 5%. The Period of the waveform doesn’t change. Note that both interrupt service routines are accessing the same data (High, Low, Period) and appropriate precautions should be taken when modifying shared data. Make sure the duty cycle increases from 5% to 95% (not decreases from 95% to 5%).

T. A. Initials:______________________

Part 4

In the last part of this lab, you will create a timer that will count the number of minutes and seconds that a program has been running (well, at least since the interrupts were enable, that is). Any time the program updates in the “Display Mode” or returns to it (e.g. after the approximation type is entered), the number of minutes and seconds that the program has been running should be updated on the fourth line of the LCD display. The time should not appear in the other modes. Your program should be able to display at least 10’s of minutes properly and should have two fixed places for the number of seconds (e.g. the time display should appear as “24:06” if the program has been running for 24 minutes and 6 seconds). Below is a suggested set of steps for developing the timer:

- Generate an interrupt at a known rate (e.g. 20 ms).
- Keep a counter that counts up to the number of interrupts equivalent to one second.
- Increment the number of seconds at that point and reset the counter.
- When the number of seconds reaches 60, increment the number of minutes and reset the seconds.
- Append a subroutine that can be called that displays the current minute and second values and call it from the appropriate portions of the main program loop.

Demonstrate the functional time-keeping part of the motor control interface.

T. A. Initials:______________________

Questions

1. What are maskable and unmaskable interrupts? Give two examples of each.

2. Where are the vector addresses for the following interrupts:
   (a) Real Time Interrupt
   (b) Timer Output Compare 2
   (c) Pulse Accumulator Overflow
   (d) Software Interrupts

List them according to increasing priority (i.e. the lowest priority interrupt first and the highest priority interrupt last).

3. What values are necessary (High, Low, Period) to generate a 20% duty cycle, 800 Hz waveform in Part 2?

4. Describe one important use of Pulse Width Modulation.