University of Manitoba
Midterm

Winter 2012

COMPUTER SCIENCE

Real-time Systems

Date: Wednesday, 7th March 2012
Time: 10:30 - 11:20
Room: EITC E2-304, University of Manitoba
(Time allowed: 50 Minutes)

NOTE: Attempt all questions.
This is a closed book examination.
Use of non-programmable calculators is permitted.
Use of any other electronic equipment is strictly forbidden.
Show your work to receive full marks.

SURNAME:

FORENAME(S):

STUDENT ID:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>15</td>
<td>20</td>
<td>50</td>
</tr>
</tbody>
</table>

CONTINUED
Section A: Timer Interrupts

1. To be able to implement more accurate delay loops, you want to generate a 50 Hz system tick on Timer0. That is every 0.02 second, the value of the unsigned 16 bit variable systicks is incremented. If it is impossible to generate an exact 50Hz timer signal, then generate one as closely as possible to 50Hz.

Assume that you are using the Taibotics educational robot as used in class. The Taibotics robot’s controller is an ATMega128 running at 16 MHz.

Only change the bits necessary to start your counter. Leave all other bits unchanged. Show the correct value for all the registers shown in the answer box. Use 0,1,X to indicate that a bit must be written as 0, 1, or should remain unchanged.

The current state of the bits are unknown.

If it is impossible to show the register settings, then say so in your answer and explain why.

[8 marks]

<table>
<thead>
<tr>
<th>Register</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCCR0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>OCR0</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>TIMSK0</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>TIFR0</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Prescaler is 256, see next question.
2. Show the interrupt service routine necessary to generate an interrupt as closely as possible to 50Hz. Also show the necessary to code to manage the variable \texttt{systicks}. [7 marks]

```c
/* Variable definitions */
volatile uint16_t systicks = 0;

/* 16MHz/50 = 320000 ticks  
320000 = 256 * 1250 , Largest prescalar that divides evenly  
1250=125*10 or better 250*5,  
Count from 256-250 five times, then incr. systick */

ISR( TIMER0_OVF_vector ) {
    static uint8_t count = 0;

    if (count++ > 5 )
    {
        systicks++;
        count = 0;
    }
    T0CNT = 256-250;
    systicks++;
}
```

Section B: Sound Generation

3. The following code is used to generate a variable frequency and amplitude sound wave on the speaker of the Taibotics Educational Robot.

```c
void startSineWaveInverted( void ) {
    DDRE = DDRE | (1 << PINE3);
    PORTE = PORTE & ~( 1 << PINE3);

    TCCR3A = (1 << COM3A0);
    TCCR3B = (1 << WGM33) | (1 << WGM32) | (1 << CS31);
    TCCR3C = 0;
}
```
ICR3H = 0x60;
ICR3L = 0x00;

OCR3AH = 0x1F;
OCR3AL = 0x00;
}

Show the generated waveform on PORTE, Bit 3 and the frequency of the generated wave.
If it is impossible to show the waveform or calculate the frequency, then say so in your answer and explain why.

[15 marks]

Frequency of the sine wave is: 40.69Hz
16000000/8=2000000 ticks per second
0x6000=24576 ticks for one half wave
2000000/(2*(24576 + 1))=40.69 Hz

Section C: Scheduling

4. Given is the following task set for a processor

<table>
<thead>
<tr>
<th>Name</th>
<th>Priority</th>
<th>WCET</th>
<th>Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>Super High</td>
<td>10ms</td>
<td>40ms</td>
</tr>
<tr>
<td>T2</td>
<td>High</td>
<td>10ms</td>
<td>80ms</td>
</tr>
<tr>
<td>T3</td>
<td>Medium</td>
<td>20ms</td>
<td>60ms</td>
</tr>
<tr>
<td>T4</td>
<td>Super High</td>
<td>20ms</td>
<td>80ms</td>
</tr>
</tbody>
</table>

Show the performance of the three scheduling algorithms (fixed priority scheduling (FPS), earliest
deadline first (EDF), and least laxity scheduling (LLS) as described in lectures.

In case of a tie, prefer the task with the lowest task number.
Complete the entire task set and indicate any missed deadlines.
If it is impossible to show the schedule, then say so in your answer and explain why.

[20 marks]
Additional work pages