ECE/CS 5780/6780: Embedded System Design

Scott R. Little

Midterm 2 Solution

Grading Information

- Exam grades will be final on 04/16/2008.
- Please discuss any questions about the grading of a question with the person who graded that question.
- Question 1: Scott
- Question 2: Scott
- Question 3: Steve
- Question 4: Anthony

Statistics

- Class average is 64.

Approximate grade breakdown

- This will give you an idea of how you scored although 5780 and 6780 are graded on different curves.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>74-80</td>
<td>4</td>
</tr>
<tr>
<td>69-73</td>
<td>9</td>
</tr>
<tr>
<td>64-68</td>
<td>10</td>
</tr>
<tr>
<td>60-64</td>
<td>9</td>
</tr>
<tr>
<td>55-59</td>
<td>6</td>
</tr>
<tr>
<td>50-54</td>
<td>4</td>
</tr>
<tr>
<td>45-49</td>
<td>2</td>
</tr>
<tr>
<td>Below 45</td>
<td>2</td>
</tr>
</tbody>
</table>
**Question 1a - 1st Option**

- 4 points.
  - Read input capture time.
  - Setup for the next capture.
  - Clear the flag register.

**Question 1a - 2nd Option**

- 4 points.
  - Set TIOS for input capture.
  - Set DDRT to be input for the correct bits.
  - Set TCTL3/4 to setup the edge triggering.
  - Set TIE for interrupts if desired.
  - Set TSCR1/2 for an appropriate TCNT period.
  - Initially clear TFLG1.

**Question 1b**

- 6 points.
  - Finish the current instruction.
  - Push the registers (A,B,X,Y,CC,PC,SP) on the stack.
  - Disable interrupts.
  - Execute the ISR.
  - Pop the registers from the stack.
  - Begin executing the user code again.

**Question 2a**

- 10 points.
  - Non-blocking scheduler.
  - Key: - normal code, * semaphore code, @ waiting for the semaphore

Key:

- Thread T1 T2 T3
- Clock time 20ms 40ms 60ms 80ms 100ms 120ms 140ms 160ms 180ms 200ms 220ms 240ms 260ms 280ms 300ms 320ms 340ms
Question 2b

- 10 points.
- Blocking scheduler.
- Key: - normal code, * semaphore code, # put onto blocking queue

<table>
<thead>
<tr>
<th>Thread</th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T1*</th>
<th>T2*</th>
<th>T3*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock time</td>
<td>20ms</td>
<td>40ms</td>
<td>60ms</td>
<td>80ms</td>
<td>100ms</td>
<td>120ms</td>
</tr>
<tr>
<td>T1</td>
<td>100ms</td>
<td>200ms</td>
<td>210ms</td>
<td>210ms</td>
<td>220ms</td>
<td>220ms</td>
</tr>
</tbody>
</table>

Question 3

- 30 points.
  - Each part was worth 10 points.
  - Reasonable solutions were given full credit.

Question 4a

- 10 points.

```java
accessMutex.wait()
<critical section for modification>
accessMutex.signal()
```

Question 4b

- 10 points.

```java
readMutex.wait()
readers++
if readers == 1:
    accessMutex.wait()
readMutex.signal()
<critical section for reading>
readMutex.wait()
readers--
if readers == 0:
    accessMutex.signal()
readMutex.signal()
```