EXAM
(Tentamen)

TDDI11
Embedded Software

2017-08-16 08:00-12:00

On-call (jour):
Ahmed Rezine, 013 - 28 1938

Admitted material:

- Dictionary from English to another language

General instructions:

- The assignments are **not ordered** according to difficulty.
- You may answer in either English or Swedish.
- Read all assignments carefully and completely before you begin.
- Use a new sheet for each assignment and use only one side.
- Before you hand in, order the sheets according to assignment, number each sheet, and fill in AID-number, date, course code and exam code at the top of the page.
- Write clearly. Unreadable text will be ignored.
- Be precise in your statements.
- **Motivate** clearly all statements and reasoning.
- **Explain** calculations and solution procedures.
- If in doubt about the question, write down your interpretation and assumptions.
- Grading: U, 3, 4, 5. The **preliminary** grading thresholds for p points are:

$0 \leq p < 20$:	U
$20 \leq p < 30$:	3
$30 \leq p < 35$:	4
$35 \leq p \leq 40$:	5

Good Luck!

Question 1, multiple choice. (10 points)
Use the answer sheet at the end of the exam.
No motivation or explanation is required for this question.
Zero or more statements may be correct for each question.
For each question, only tick the correct statement(s) if any.

1a) Which of the following statements is / are correct. Direct Memory Access (DMA) is:

1. Another name for polling
2. Wastes less CPU cycles for I/O processing.
3. Requires less hardware support than interrupt based I/O processing.

1b) Which of the following statements is / are correct?

1. An embedded system is always a real time system.
2. An embedded system cannot be real time.
3. A real time system can be embedded.

1c) What will be the output from the following C program?

```
#include <stdio.h>
int main() {
      unsigned long int a = 10;
      unsigned long int *b = &a;
      unsigned long int *c = b;
      a = 12;
      printf("%lu %lu \n", a, *c);
}
```

1. 12 10
2. 12 12
3. 10 10

1d) What will be the output of the following C program?

```
#include <stdio.h>
int main() {
      printf("%d \n", (2 || 0) && 1);
}
```

1. 0
2. 1
3. 2

1e) What will be the output of the following C program?

```
#include <stdio.h>
int main() {
      printf("%d \n", (2 | 0) & 1);
}
```

1. 0
2. 1
3. 2

1f) Which of the following statements is / are correct?

1. The foreground/background model requires extensive support from an underlying operating system.
2. The foreground/background model is not suitable for improving response time in complex and large systems.
3. The foreground/background model is suitable for improving scalability and maintainability of complex and large systems.

1g) Which of the following statements is/are correct. A Moore state machine can be used to model an embedded system, and

1. is a finite state machine where loops may occur (transitions going back to the same state).
2. is a finite state machine that associates outputs to states.
3. is a finite state machine that associates outputs to transitions.

1h) Which of the following statements is/are correct. Shared memory based concurrent software:

1. Can involve threads.
2. May result in deadlocks.
3. May result in race conditions.

1i) Which of the following statements is/are correct. An over-the-wall design process

1. Tend to isolate teams from each others
2. Cuts on product costs and time to market by improving the communication between the different teams.
3. Is another name for concurrent engineering.

1j) Which of the following statements is / are correct. An AND-group is used in a state machine

1. To represent the fact that the system state is composed of several simultaneous sub-states.
2. To describe a non-deterministic choice between two outgoing transitions.
3. To describe a non-deterministic choice between two incoming transitions.

Question 2. (5 points)

Consider a system that periodically reads 4 consecutive bytes representing an unsigned 4-bytes integer. The 4 bytes are read from a buffer “char in_buffer[4]”. The 4-bytes integer is stored in big-endian order. The processor uses little-endian order.

a) Provide a C-like pseudocode for a program that fills a 4-bytes buffer “char out_buffer[4]” with the same content as the one in in_buffer (i.e., one 4-bytes integer) but where each integer is now stored in little-endian order. (3pt)
b) Explain how you do if in_buffer was written in little-endian and the out_buffer had to be written in big-endian order. (2pt)

Question 3. (5 points)
There are three main approaches to I/O processing (interrupt based is one of them). Explain briefly each of them and compare them in terms of what kind of platform support is needed for each one of them.

Question 4. (5 points)
Consider a task set with two periodic tasks: Task 1 with period T1= 5 and execution time C1= 2 and Task 2 with period T2= 7 and execution time C2= 4. Both tasks are to be run on the same processor using some scheduling algorithm.

1. Give the processor utilization ratio in case the two tasks are scheduled (1pt)
2. What would the priority of the tasks be if RMS is used (1pt)
3. Assume pre-emptive RMS is used. Can the tasks be scheduled? Explain using a diagram (1 pt).
4. Can pre-emptive EDF schedule the tasks? Explain using a diagram (2pt).

Question 5. (5 points)
Give a three-states Mealy machine that takes sequences of 0s and 1s as input. The machine should output 0 unless it witnesses two consecutive inputs with the same value (i.e., two consecutive 0s or two consecutive 1s), in which case it outputs 1. Possible runs of your solution:

Input sequence	Output sequence
`0000...`	`0111...`
`1111...`	`0111...`
`1010...`	`0000...`
`0011...`	`0101...`

Question 6. (5 points)
Describe the sequence of events that occur when a key is pressed and read into memory when using interrupt-driven I/O

Question 7. (2 points)
The following macros is meant to double a number. This macros is not well written. Explain what problems may occur if used as currently written and rewrite it to solve these problems.

```
#define Twice(x) 2*x
```

Question 8. (3 points)
Write a C-like pseudocode method “swap” that takes as input a 16 bits unsigned int and returns the result of swaping each even bit with the odd bit following it. For example, the argument “10 01 10 01 10 01 10 01” would result in swap returning “01 10 01 10 01 10 01 10” while the argument “10 00 10 00 10 00 10 11” would result in swap returning “01 00 01 00 01 00 01 11”.

Answer sheet for question 1. Please hand this paper in together with the answers for the other questions (numbered and with AID number).

1a)	( ) 1	( ) 2	( ) 3
1b)	( ) 1	( ) 2	( ) 3
1c)	( ) 1	( ) 2	( ) 3
1d)	( ) 1	( ) 2	( ) 3
1e)	( ) 1	( ) 2	( ) 3
1f)	( ) 1	( ) 2	( ) 3
1g)	( ) 1	( ) 2	( ) 3
1h)	( ) 1	( ) 2	( ) 3
1i)	( ) 1	( ) 2	( ) 3
1j)	( ) 1	( ) 2	( ) 3