Försättsblad till skriftlig tentamen vid Linköpings universitet

Datum för tentamen	2019-08-21
Sal (2)	TER1(5) TERE(1)
Tid	8-12
Utb. kod	TDDI11
Modul	TEN2
Utb. kodnamn/benämning **Modulnamn/benämning**	Programmering av inbyggda system Skriftlig tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	8
Jour/Kursansvarig Ange vem som besöker salen	Ahmed Rezine
Telefon under skrivtiden	013 28 1938
Besöker salen ca klockan	9:00
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Veronica Kindeland Gunnarsson veronica.kindeland.gunnarsson@liu.se 5634
Tillåtna hjälpmedel	Ordbok engelska-annat språk.
Övrigt	
Antal exemplar i påsen	

Försättsblad till skriftlig tentamen vid Linköpings universitet

Datum för tentamen	2019-08-21
Sal (2)	TER1(5) TERE(1)
Tid	8-12
Utb. kod	TDDI11
Modul	TEN2
Utb. kodnamn/benämning **Modulnamn/benämning**	Programmering av inbyggda system Skriftlig tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	8
Jour/Kursansvarig Ange vem som besöker salen	Ahmed Rezine
Telefon under skrivtiden	013 28 1938
Besöker salen ca klockan	9:00
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Veronica Kindeland Gunnarsson veronica.kindeland.gunnarsson@liu.se 5634
Tillåtna hjälpmedel	Ordbok engelska-annat språk.
Övrigt	
Antal exemplar i påsen	

EXAM
(Tentamen)

TDDI11
Embedded Software

2019-08-21 kl: 08-12

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
$20 \leq p \leq 30$:	3
$31 \leq p \leq 35$:	4
$36 \leq p \leq 40$:	5

Question 1, multiple choice. (10 points)
Use the answer sheet at the end of the exam. No motivation or explanation is required for this 10-points question. **Zero or more statements may be correct for each question.** Tick each statement if and only if it is correct. Ticking a wrong statement or missing to tick a correct statement gives 0 points for that question.

1a) Memory-mapped I/O communication with peripherals

1. Uses usual memory assembly instructions to communicate with peripherals.
2. Uses special I/O instructions to access memory and communicate with peripherals.
3. Is UART based.

1b) A UART bus:

1. Is Parallel.
2. Can make use of parity bits.
3. Can only be used between two components.

1c) A soft-real-time embedded systems

1. Uses software instead of hardware as in hard-real-time systems.
2. Requires each computation to be carried out as soon as possible.
3. Correctness does not require meeting all deadlines.

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

```
int main() {
        int a[5];
        for(int i=0; i < 5; ++i)
          a[i]= i + 2;
        printf("%d \n", *(a + 3));
}
```

1. 0
2. 3
3. 5

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

```
int main() {
        printf("%d \n", (2+1) && 6);
}
```

1. 1
2. 2
3. 3

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

```
int main() {
        printf("%d \n", (1 | 3) & 7);
}
```

1. 1
2. 3
3. 4

1g) Which of the following statements are correct?

1. The foreground/background model requires a real-time operating system.
2. The foreground/background is well suited for complex real-time systems that are often updated.
3. The foreground/background model has problems with scalability and maintainability.

1h) A finite state machine capturing the program controlling an embedded system…

1. should have an initial state
2. should be deterministic
3. should have a final state

1i) Some advantages of using a message passing approach to concurrency are that …

1. It allows fast access to shared memory.
2. It solves the deadlock problem.
3. No need for mutexes to protect individual reads and writes to shared variables.

1j) Concurrent engineering:

1. Makes several teams compete to deliver the best product.
2. Build cross functional teams to ease information sharing.
3. Tends to result in higher costs and longer times-to-market than "over-the-wall" engineering.

Question 2. (4 points)

Write a C function "unsigned convert(unsigned num){…}" that returns the little endian value corresponding to a 4 bytes unsigned big-endian number "num".

Question 3. (5 points)

Explain **the steps** involved when I/O programming using interrupt and when using polling. What is the difference between the two of them in terms of **required CPU cycles** and **hardware support**? **Explain**.

Question 4. (6 points)

Consider a task set with three periodic tasks: Task A with period TA= 3 and execution time CA= 1, Task B with period TB= 6 and execution time CB= 2, and task C with period TC=9 and execution time CC=3. All three tasks are to run on the same processor using some scheduling algorithm.

1. Give the processor utilization ratio in case the tasks are scheduled (1pt)
2. Which task would get the highest priority if Rate Monotonic Scheduling (RMS) is used (1pt)
3. Can the tasks be scheduled using preemptive RMS? Explain with a Gantt diagram (2pt).
4. Can the tasks be scheduled using preemptive Earliest Deadline First (EDF)? Explain using a Gantt diagram (2pt).

Question 5. (5 points)
Give a Mealy machine (outputs associated to transitions, not states) that takes sequences of 0s and 1s as input. The machine should output 1 when it finished reading a non-empty sequence of ones that is of a length that is divisible by 4 (i.e. sequence of 4, 8, 12, 16, ... of consecutive ones followed by a zero). It should output 0 otherwise. Possible runs of your solution:

Input sequence	Output sequence
011110000...	000001000...
110111100...	000000010...
111111110...	000000001...
111111100...	000000000...

Question 6. (5 points)
Describe the sequence of events that occur when a CPU requests a DMA (Direct Memory Access) controller to buffer data, e.g., disk data, to memory. Describe the main advantages compared to interrupt based I/O.

Question 7. (2 points)
The following macros is meant to subtract two numbers. This macro is not well written. We still want to use a similar macro for computing the difference of two numbers. Give an example that would compile but not give the intended result. Rewrite the macro to solve the problem you identified.

```
#define diff(x,y) (x - y)
```

Question 8. (3 points)
In this question, you can use bit-level operators: e.g., "&" (and), "|" (or), "<<" (shift left), ">>" (shift right), etc. You are not allowed to use loops, additions, divisions, subtractions or multiplications. Write a C function "int is_8_multiple(unsigned short x)" that returns 1 if the 16-bits unsigned x is a multiple of 8 and 0 otherwise.

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