



**KTH Microelectronics
and Information Technology**

WRITTEN EXAM

2B1445 Embedded Systems

2005-08-22, 9.00-13.00

Material

The student is allowed to use the following material in the exam:

- ARM & Thumb Instruction Set Quick Reference Manual
- Calculator

- Write your name on top of each paper.
- Do not use a paper for more than one task.
- Write only on the front side of each paper.
- Motivate your answers. A correct answer with a wrong or no motivation can result in 0 points!
- The exam consists of 6 tasks on 4 pages, which together give 36 points.
- 18 points are required to pass the exam!

GOOD LUCK!

1. Give an answer to the following questions.
 - (a) (3 POINTS) Describe how *semaphores* can be used to protect a shared resource in a multi-tasking environment? How do semaphores work?
 - (b) (3 POINTS) Explain the principles of polling and interrupt. What are the advantages and disadvantages of both techniques?
 - (c) (2 POINTS) Can caches always be used to their full advantage in an embedded system? Motivate your answer!
2. (4 POINTS) Write the ARM assembly code needed for the following function. Follow the *ARM procedure call standard* (APCS) so that it can be called from a C program. The variables *i*, *x* shall be local to this function and thus be located in registers.

```
f(int a, b, c)
{
    int i, x;

    x = 0;

    for(i = 1; i < a; i++)
    {
        if (i <= b)
            x = x + c;
    }
    return x;
}
```

3. (5 POINTS) A small ARM program is divided into three source files.

- Source-File 1

```
GLOBAL A
AREA inst_A, CODE
ENTRY
EXTERN num1
EXTERN num2
EXTERN X
A    ADR r1, num1
     LDR r2, [r1]
     ADR r1, num2
     LDR r3, [r1]
     MOV r4, #1
     BL X
```

```

D      B   D
      END

```

- Source-File 2

```

      GLOBAL Y
      AREA inst_B , CODE
X     CMP r4 , r3
      BGE Y
      MOV r7 , #1
      ADD r4 , r4 , r7
      B   X
Y     MOV pc , lr

```

- Source-File 3

```

      GLOBAL num1 , num2

      AREA num_A , DATA
num1  DCD 1 ,2 ,3 ,4 ,5 ,6 ,7 ,8
num2  DCD 3

```

- Determine for each source file the symbol table that is generated by the assembler. Assume a relative start address of 0.
 - Does the linker have sufficient information to generate an executable program. Motivate!
 - How many instructions will the program run until it enters for the first time label D (include that instruction in your calculation) assuming the program starts at label A?
 - What is the value in r4 after the program has run to label D?
4. (4 POINTS) Draw the control and data flow graph (CDFG) for the following code fragment and determine the longest path and shortest path and give their execution time. You can assume that all statements and branches are executed in one time unit and that all branch directions are equally probable.

```

if ( a > 1 ) {
  for( i = 0; i < 10; i++ ) {
    if ( b > 0 ){
      w[ i ] = x[ i ] - y[ i ];
      u[ i ] = u[ i ] + 2;
    }
    else {
      w[ i ] = x[ i ] + y[ i ];
    }
  }
}

```

```

    }
  }
else {
    w[ i ] = 0;
    u[ i ] = 0;
  }
}

```

5. (8 POINTS) A small *word-addressable* embedded computer system has a main memory consisting of 1024 32-bit words. It also has a small data cache capable of holding eight 32-bit words, where each cache block consists of only two words. When a given program is executed the processor reads data from the following sequence of hex addresses:

0x000, 0x002, 0x003, 0x0C1, 0x000, 0x001, 0x0C2, 0x1D4, 0x1D5.

This pattern is repeated four times.

- (a) Consider a 2-way set-associative cache. The cache uses the following replacement policy: If a cache line is full, the least recent used block is replaced in the cache.
 - i. How many bits are used for the tag, block, and offset fields for the representation of a memory address?
 - ii. Show the contents of the cache at the end of the execution. Assume the cache is empty at the start of the program.
 - iii. Compute the hit-rate for this example. Assume that the cache is initially empty.
 - (b) Does a direct-mapped cache capable of holding eight 32-bit words achieve the same performance with respect to hit-rate? Give a convincing motivation.
6. (7 POINTS) Assume three periodic processes¹ $P_1 = (1, 4, 4)$, $P_2 = (1, 2, 2)$ and $P_3 = (1, 5, 5)$, which shall be scheduled on a single processor.
- (a) Give the processor utilisation.
 - (b) Can the rate-monotonic algorithm produce a feasible schedule? In such a case, give a schedule.
 - (c) Can the earliest-deadline-first algorithm produce a feasible schedule? In such a case, give a schedule.
 - (d) What are the advantages of rate-monotonic scheduling?

¹A process P is defined as (c, p, d) , where c is the execution time, p the period and d the deadline.