hDepartment of Electrical and Computer Engineering
Dalhousie University

ECED 3403 – Computer Architecture

Final Examination – Sample questions

9 August 2024

Instructions

The following examination is open book and open notes.

Laptops and other computing devices are permitted on the understanding they will only be used for access to course notes. Any other use is prohibited.

Write your name and student number on each examination booklet used.

Read over the entire examination before attempting any questions. Questions may be answered in any order; however, please label each answer with its section and question number. Give examples using XM23 where and when necessary. State any assumptions. **Neatness and brevity would be appreciated.**

The instructor will **not** answer questions during the examination.

Time allotted for this examination is 180 minutes (three hours). The examination consists of four (4) pages on two sheets. The examination can be kept.

Section 1

The following section contains 20 one-point questions.

1. How are the fields of a C/C++ structure accessed in X-Makina?

Section 2

The following section contains 10 short answer questions, each is worth 2 points.

1. What could possibly go wrong with the following "optimized" checksum decoder? Give two reasons.

```
chksum = 0;

while (chksum != 0xFF)
{
        chksum += *msg;
        /* Display char and checksum */
        printf("%c (%02x)  %02x  \n", *msg, BYTE(*msg), chksum);
        /* Next char */
        msg++;
        len--;
}

if (chksum == 0xff)
   printf("Msg correct");
else
   printf("Chksum error!");
```

Section 3

The following section contains five questions, each is worth 4 points.

1. Some of XM23's constants are seldom, if ever, used. A better approach would be to allow the programmer to assign a value to a constant at run time. Assume a new constant table that allows four programmable constants:

R/C		SRC
0	**1**	**Encoding (bits 3-5)**
Register	**Constant**	
R0	0	000
R1	1	001
R2	2	010
R3	-1	011
R4	Programmable	100
R5/LR		101
R6/SP		110
R7/PC		111

The constant is to be taken from a register (allowing it to be 16-bits long) and assigned to one of the four programmable constants (4, 5, 6, or 7). For example, if R2 had the value `#1234` and was assigned to location 7, the instruction would take R2 and store it as the eighth constant. The constant table would have the value `#1234` as the eighth constant and accessed as `RC=1` and `SRC=111`. Constants `000` to `011` cannot be changed.

Propose a new instruction that would assign a value to a constant. Specify the opcode and operands required.