CARLETON UNIVERSITY
School of Computer Science

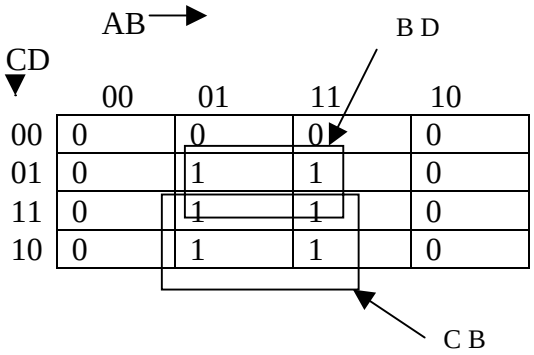
COMP 2003A Computer Organization

Fall 2005

Mid-term Examination Solution Key

Question 1 [5 marks]

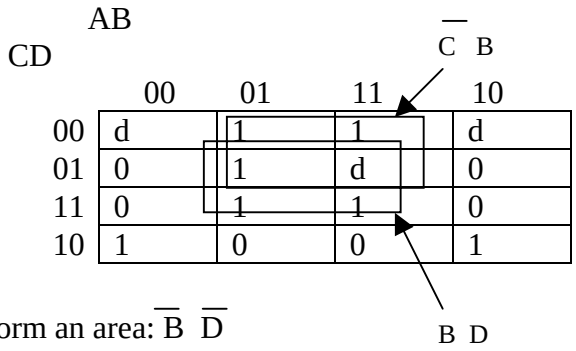
a)



Marking scheme: 1 mark for each term

Answer: **B D + C B**

b)



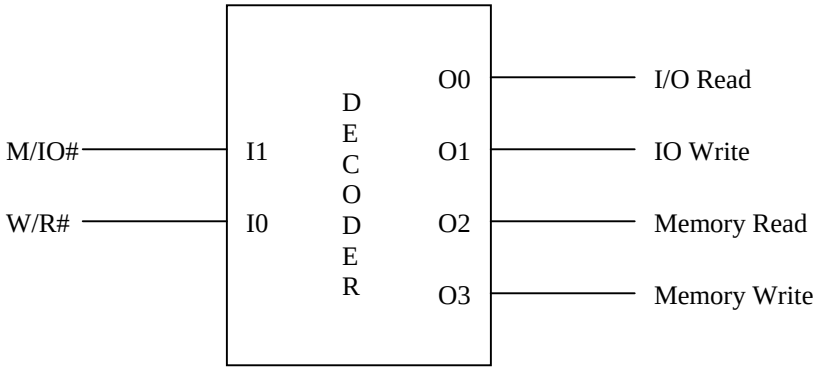
Four corner cells form an area: $\overline{B}\ \overline{D}$

Marking scheme: 1 mark for each term

Answer: $\overline{B}\ \ \overline{D} + \overline{C}\ B + B\ D$

Question 2

(a)



2-to-4 Decoder

Marking scheme:

Using the decoder and showing the connections – 2 marks
Correct connections – 1 mark
(for example, if we switch the two inputs---M/IO# to I0 and W/R# to I1 --- the output labeling should be changed)

(b) Convert the following logical addresses to physical addresses. All numbers are in hexadecimal. Assume the real address mode of the Pentium processor. **(2 marks)**

```
(i)    1345:A251
(ii)   A1B3:D011
```

(i)

```
 13450
  A251
------
 1D6A1  ← Physical address
```

(ii)

```
 A1B30
  D011
------
 AEB41  ← Physical address
```

Marking scheme: 1 mark each

Question 3

(a)

This limitation is due to the fact that Pentium has *six segment registers*.
Marking scheme: No partial marks (0 or 1.5 marks only)

(b)

This restriction is due to the fact that Pentium segment registers can hold *only 16-bit values*.
Marking scheme: No partial marks (0 or 1.5 marks only)

Question 4

Marking scheme:

Four levels: 0, 1, 2, or 3 marks will be given, depending on the answer.

(a)

Computer systems are no longer limited to number crunching. Modern computer systems are more complex and these systems run a variety of different applications (3D rendering, string processing, number crunching, and so on). Performance measured for one type of application may be inappropriate for some other application. Thus, it is important to measure performance of various components for different types of applications.

(b)

RISC processors use the load/store architecture, which assumes that the operands required by most instructions are in the internal registers. Load and store instructions are the only exceptions. These instructions move data between memory and registers. If we have few registers, we cannot keep the operands and results that can be used by other instructions (we will be overwriting them frequently with data from memory). This does not exploit the basic feature of the load/store architecture. If we have more registers, we can keep the data longer in the registers (e.g., result produced by an arithmetic instruction that is required by another instruction), which reduces the number of memory accesses. Otherwise, we will be reading and writing data using the load and store instructions (lose the main advantage of the load/store architecture).

(c)

Registers are used as a processor's scratchpad to store data and instructions temporarily. Because accessing data stored in the registers is faster than going to the memory, optimized code tends to put most-often accessed data in processor registers. Obviously, we would like to have as many registers as possible, the more the better.

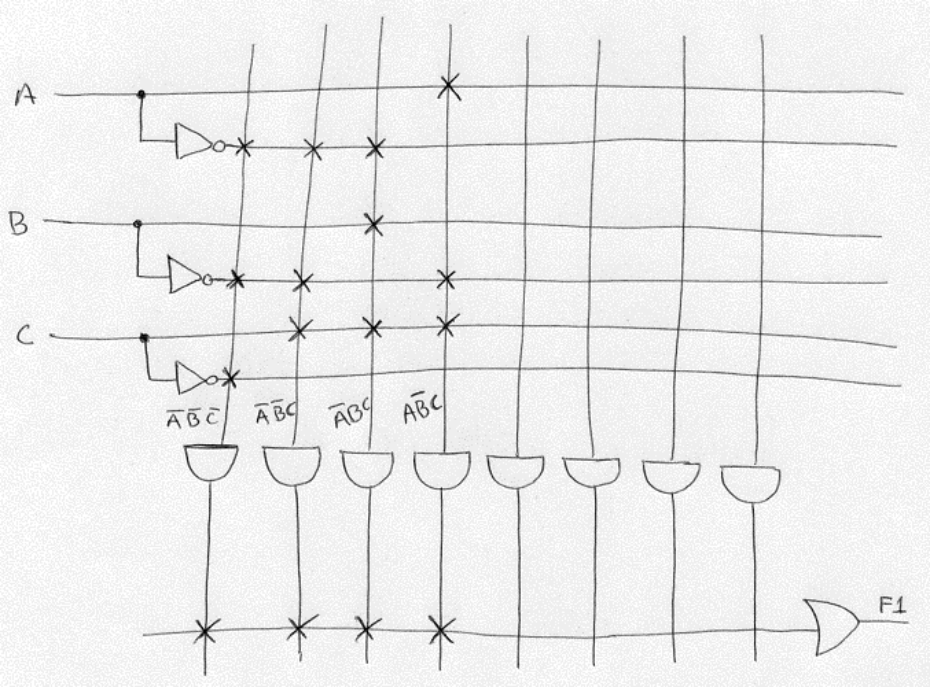
Some of the registers contain special values. For example, all processors have a register called the *program counter* (PC). The PC register maintains a marker to the instruction that the processor is supposed to execute next. Some processors refer to the PC register as the instruction pointer (IP) register. There is also an *instruction register* (IR) that keeps the instruction currently being executed. Although some of these registers are not available, the programmer can use most processor registers.

Question 5

(a)

Marking scheme:

Basic PLA – 1 mark
AND array connections – 2 marks
OR array connection – 1 mark



(b)

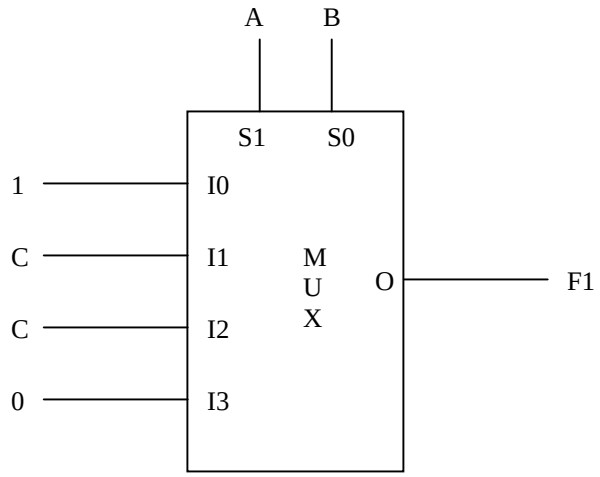
A	B	F1	
0	0	1	← 0.5 mark
0	1	C	← 1 mark
1	0	C	← 1 mark
1	1	0	← 0.5 mark

(c)

Marking scheme:

Input and output connections: 1 mark

Control (i.e., selection) inputs: 1 mark



Question 6

Answer: Propagation delay for the ripple counter circuit (Figure 1): 30 ns

Answer: Propagation delay for the synchronous counter circuit (Figure 2): 15 ns

Marking scheme: 2 marks each

Work space: Here explain how you computed the answers given on the previous page.

Explanation for Figure 1

The longest path (for the clock to propagate) involves three JK flip-flops to produce the Q2 output. Thus, the propagation delay for the whole circuit is 3 * 10 ns = 30 ns.

Explanation for Figure 2

The longest path (for the clock to propagate) involves one AND gate and one JK flip-flop. Thus, the propagation delay for the whole circuit is 5 ns + 10 ns = 15 ns.