CS2100 Computer Organization

AY2024/25 Semester 2

Assignment 2

(Deadline: 17 March 2025, Monday, 1pm)

ANSWERS

Instructions

1. There are **5** questions in this assignment, with a total of 40 marks.

2. This assignment is due on **Monday, 17 March 2025, 1 pm.** The late submissions will incur penalties: a 10% penalty for submissions up to one day late (e.g., submitting at 10.20pm for a 10pm deadline), a 20% penalty for submissions up to two days late, and no grades will be given for assignments submitted more than two days late.

3. Enter your answers into **Canvas > Assignment 2**.

4. Please read and follow the instructions on how to format your answers. This is important as your answers will be autograded, so any answer that departs from the specified format will be graded as incorrect.

5. You should do these assignments on your own. Do not discuss the assignment questions with others.

6. Please post on QnA "Assignments" topic if you have any queries on this assignment.

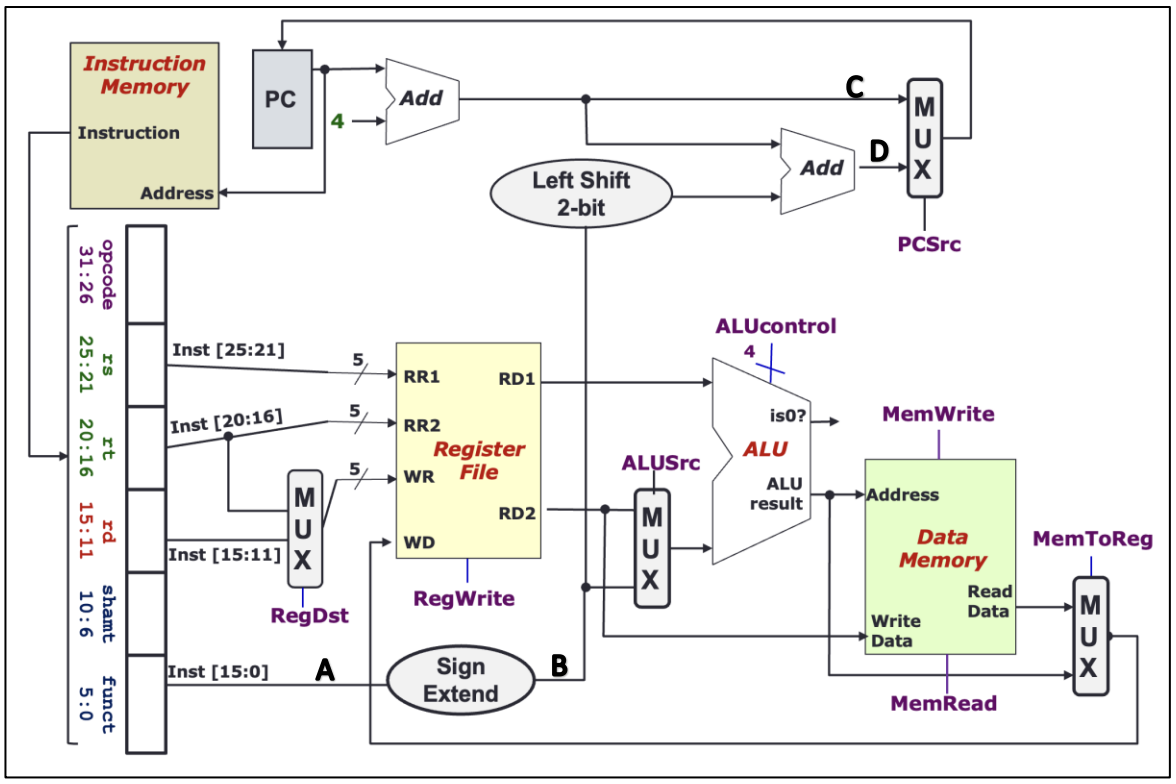
Part I: MIPS Datapath (20 marks)

Question 1. (20 marks)

Consider the MIPS Datapath with the initial values of the registers and memory as shown below. Values preceded by 0x are in hexadecimal.

MIPS Registers	
R0 (r0) = 0x00000000	R1 (at) = 0x00002000
R2 (v0) = 0x00000001	R3 (v1) = 0x0000000a
R4 (a0) = 0x00000005	R5 (a1) = 0x7ffff000
R6 (a2) = 0x7ffff004	R7 (a3) = 0x000000b0
R8 (t0) = 0x00000001	R9 (t1) = 0x00000c00
R10 (t2) = 0x0000c000	R11 (t3) = 0xfffffff0
R12 (t4) = 0xf0000000	R13 (t5) = 0x00000fff
R14 (t6) = 0x00006200	R15 (t7) = 0x00000e00
R16 (s0) = 0x00300000	R17 (s1) = 0x00000c00
R18 (s2) = 0x00040200	R19 (s3) = 0x00011000
R20 (s4) = 0x00030200	R21 (s5) = 0x00000000
R22 (s6) = 0x00055000	R23 (s7) = 0xf0000000
R24 (t8) = 0x00000005	R25 (t9) = 0x0000d000
R26 (k0) = 0x00000000	R27 (k1) = 0x00000000
R28 (gp) = 0x10008000	R29 (sp) = 0x7fffeff4
R30 (s8) = 0x1000000f	R31 (ra) = 0x00400018

Address	Memory
0xFFFFFF94	100
0xFFFFFF98	200
0xFFFFFF9C	300
0xFFFFFFA0	400
0xFFFFFFA4	500



The current PC value and the address of the first instruction is 0x**00001024. T**he following instructions are being executed one by one:

```
          sw   $s6, -100($s5)    # instruction 1
          addi $s2, $s3, 0x20    # instruction 2
          beq  $s6, $s3, target  # instruction 3
          addi $s4, $s5, 0x40    # instruction 4
   target:
```

Fill in the values of the fields in the table below. You have to fill column 2 after executing instruction 1 and column 3 after executing instruction 3. For rows marked with *, your answers must follow the base given (0b for binary, 0x for hexadecimal), and you must write all the required digits with A to F in capital letters, or it will be graded as wrong even if the value is correct. You do NOT need to include the prefix 0b or 0x in your answers. For rows without *, the answers are to be in decimal, and you are not to include leading zeroes in your decimal answers. If a control signal is a don't-care value, enter X.

Fields	Values after executing Instruction 1	Values after executing Instruction 3
RegDst		
MemRead		
MemWrite		
ALUcontrol*	**0b**	**0b**
RegWrite		
Instruction[31-26]*	**0b**	**0b**
Instruction[25-21]*	**0b**	**0b**
Instruction[20-16]*	**0b**	**0b**
Instruction[15-0]*	**0x**	**0x**
A*	**0x**	**0x**
B*	**0x**	**0x**
RD1*	**0x**	**0x**
RD2*	**0x**	**0x**
ALU Result*	**0x**	**0x**
ALUSrc		
RR1		
RR2		
MemToReg		
C*	**0x**	**0x**
D*	**0x**	**0x**

Answers: The instructions are decoded as below for you reference. Students are not expected to provide this solution or working. They just need to fill the values in the table below.

sw $s6, -100($s5) is decoded as:

101011 Op	10101 Rs	10110 Rt	1111 1111 1001 1100 -100 in 2s Comp, 16bit rep

beq $s6, $s3, target is decoded as:

000100 Op	10110 Rs	10011 Rt	0000 0000 0000 0001 target = 1 in 2s Comp, 16bit rep

Fields	Values for Instruction 1	Values for Instruction 3
RegDst	X	X
MemRead	0	0
MemWrite	1	0
ALUcontrol*	0b0010	0b0110
RegWrite	0	0
Instruction[31-26]*	0b101011	0b000100
Instruction[25-21]*	0b10101	0b10110
Instruction[20-16]*	0b10110	0b10011
Instruction[15-0]*	0xFF9C	0x0001
A*	0xFF9C	0x0001
B*	0xFFFFFF9C	0x00000001
RD1*	0x00000000	0x00055000
RD2*	0x00055000	0x00011000
ALU Result*	0xFFFFFF9C	0x00044000
ALUSrc	1	0
RR1	21	22
RR2	22	19
MemToReg	X	X
C*	0x00001028	0x00001030
D*	0x00000E98	0x00001034

Part II. Boolean algebra, logic circuits and simplification (20 marks)

Note that in general, unless otherwise stated, complemented literals are not available. Constants 0 and 1 are always available, and they are considered (degenerated form of) SOP and POS expressions.

Remember to write · for the AND operation, or mark will be deducted. If you are typing your answers, you may use the full stop (.) for AND and the single quote (') for complement. Do not use other alternative symbols (such as $\sim A$, $\neg A$, $\bar{A}$, etc. for the complement of A, write A' instead).

Unless otherwise stated, workings are not required.

The above instructions apply for subsequent assignment, midterm test and the final exam.

Note that no partial credits will be awarded for incorrect answers for most of the questions.

Question 2 (Total: 6 marks)
Given a 5-variable Boolean function $F(A, B, C, D, E)$.

(a) Let x and y be any two distinct integers in the range 0 through 31. What is the result of (minterm x AND minterm y)? (1 mark)

(b) Pick any two distinct integers x and y and show clearly the working on how you derive your answer for part (a) above. You are to cite the laws/theorems used. (2 marks)

(c) How many minterms does a 2-literal product term in F have? (1 mark)

(d) Suppose $F(A, B, C, D, E) = C \cdot D + A' \cdot C \cdot E + A \cdot D \cdot E$. Write $F(A, B, C, D, E)$ in the $\sum m$ notation. You are to list out the minterms in increasing order, enclosed in a pair of parentheses, separated by comma, with no space and no other punctuation. For example, if the answer is $\sum m\,(3,5,12)$, you are to enter (3,5,12). No working is needed. (2 marks)

Answers

(a) 0.

(b) Let $x = 30$ and $y = 31$. Then minterm 1 = $A \cdot B \cdot C \cdot D \cdot E'$ and minterm 2 = $A \cdot B \cdot C \cdot D \cdot E$.

Hence $(A \cdot B \cdot C \cdot D \cdot E') \cdot (A \cdot B \cdot C \cdot D \cdot E)$
$= (A \cdot A \cdot B \cdot B \cdot C \cdot C \cdot D \cdot D \cdot E' \cdot E)$ (by a series of associative law and commutative law)
$= (A \cdot A \cdot B \cdot B \cdot C \cdot C \cdot D \cdot D \cdot 0)$ (as $E' \cdot E = 0$ by the complement law)
$= 0$ (by the zero element theorem)

General explanation: Since the 2 minterms are distinct, there must be at least one pair of literals in them that are complementary, say, X' and X. Since $X' \cdot X = 0$ by the complement law, the product of these two minterms will bo 0.

(c) 8.

(d) $F(A, B, C, D, E) = \sum m\,(5,6,7,13,14,15,19,22,23,27,30,31)$.

$C \cdot D = --11- = \Sigma m(6,7,14,15,22,23,30,31)$
$A' \cdot C \cdot E = 0-1-1 = \Sigma m(5,7,13,15)$
$A \cdot D \cdot E = 1--11 = \Sigma m(19,23,27,31)$

Question 3 (Total: 2 marks)

The following Boolean functions F, G, H on 4 variables a, b, c, d are given in sum-of-minterms ($\sum m$) notation or product-of-maxterms ($\prod M$) notation:

$$F(a,b,c,d) = \sum m(1,2,3,4,5,6,7)$$
$$G(a,b,c,d) = \sum m(0,1,4,5,11,13,15)$$
$$H(a,b,c,d) = \prod M(0,2,4,6,8,11,15)$$

Find (a) $F' \cdot G$; and (b) $G \oplus H'$, where $\oplus$ is the exclusive-or (XOR) operation. Write your answers in the $\sum m$ notation. You are to list out the minterms in increasing order, enclosed in a pair of parentheses, separated by comma, with no space and no other punctuation. For example, if the answer is $\sum m\,(3,5,12)$, you are to enter (3,5,12). No working is needed.

Answers

(a) $F'(a,b,c,d) = \sum m(0,8-15); G(a,b,c,d) = \sum m(0,1,4,5,11,13,15)$

$F' \cdot G = \sum \boldsymbol{m(0,11,13,15)}$. [1 mark]

(b) $G(a,b,c,d) = \sum m(0,1,4,5,11,13,15); H'(a,b,c,d) = \sum m(0,2,4,6,8,11,15)$

$G \oplus H' = \sum \boldsymbol{m(1,2,5,6,8,13)}$. [1 mark]

Question 4 (Total: 7 marks)

Given the 4-variable Boolean function $Z(A,B,C,D) = \sum m(1,2,3,7,9,15) + \sum x(0,5,6,11,12)$ where m are the minterms and x are the don't-cares.

(a) How many prime implicants (PIs) are there on the K-map of Z? [1 mark]

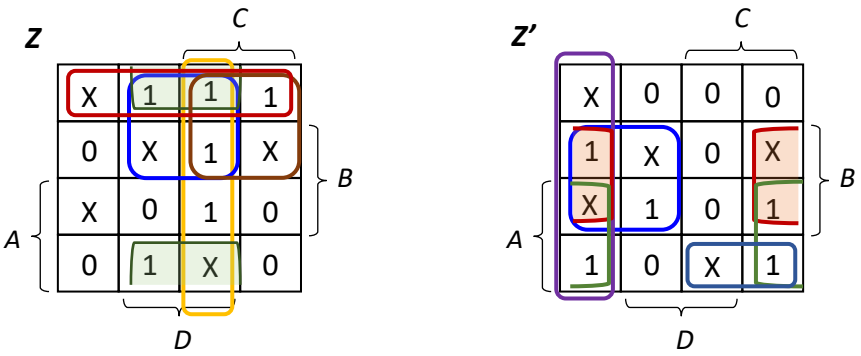
(b) How many essential prime implicants (EPIs) are there on the K-map of Z? [1 mark]

(c) How many distinct simplified SOP expressions are there for Z? [1 mark]

(d) Write out one simplified SOP expression for Z. You are to (1) write the literals with a product term in alphabetical order (eg: $P \cdot Q' \cdot R$ instead of $R \cdot Q' \cdot P$ or $P \cdot R \cdot Q'$), and (2) write the essential prime implicants before the non-essential ones. The order among the essential prime implicants and the order among the non-essential prime implicants do not matter. You have to use the dot for AND, plus for OR, single quote for negation, with no space, no unnecessary parentheses and no symbols other than dot, plus, single quote and parentheses. For example, if the answer is $B \cdot A + C' \cdot D$, you are to enter A.B+C'.D [2 marks]

(e) Write out one simplified POS expression for Z. The order of literals in a sum term must be in alphabetical order (eg: $U + V' + W'$ instead of $W' + U + V'$ or $U + W' + V'$). You have to use the dot for AND, plus for OR, single quote for negation, with no space, no unnecessary parentheses and no symbols other than dot, plus, single quote and parentheses. For example, if the answer is $(B + A) \cdot (C' + D)$, you are to enter (A+B).(C'+D) [2 marks]

Answers:



(a) 5 PIs. (They are: $A' \cdot B'$, $A' \cdot D$, $C \cdot D$, $B' \cdot D$ and $A' \cdot C$.)

(b) 2 EPIs. (They are: $C \cdot D$ and $B' \cdot D$.)

(c) 2.

(d) $B' \cdot D + C \cdot D + A' \cdot B'$ or $B' \cdot D + C \cdot D + A' \cdot C$ (The EPIs $B' \cdot D$ and $C \cdot D$ can be swapped, but they must be placed before the non-EPI according to the instruction.)

(e) $(B' + C) \cdot (A' + D)$ or $(A' + D) \cdot (B' + C)$
Working: $Z' = B \cdot C' + A \cdot D'$

Question 5 (Total: 5 marks)
The 2421 code for the ten decimal digits 0 to 9 is shown in the table below:

Decimal digit	0	1	2	3	4	5	6	7	8	9
2421 code	0000	0001	0010	0011	0100	1011	1100	1101	1110	1111

Parity bit is an error-detection scheme. In the even parity bit scheme, the parity bit is assigned 0 if the number of 1's in the data bits is an even number, or assigned 1 if the number of 1's in the data bits is an odd number.

You are to design a circuit that takes in a 4-bit 2421 code *ABCD* and generates two outputs *V* and *P*: output *V* is 1 if the input is a valid 2421 code, or 0 otherwise; output *P* is the parity bit.

For simplified SOP expressions, you are to (1) write the literals with a product term in alphabetical order (eg: $P \cdot Q' \cdot R$ instead of $R \cdot Q' \cdot P$ or $P \cdot R \cdot Q'$), and (2) write the essential prime implicants before the non-essential ones. The order among the essential prime implicants and the order among the non-essential prime implicants do not matter. You have to use the dot for AND, plus for OR, single quote for negation, with no space, no unnecessary parentheses and no symbols other than dot, plus, single quote and parentheses.

(a) How many simplified SOP expressions are there for *V*? [1 mark]

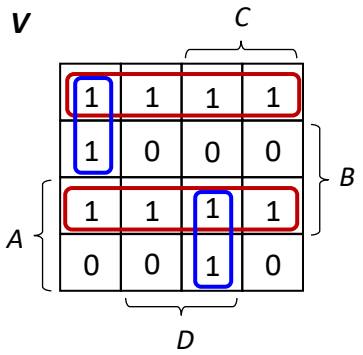
(b) Write out one simplified SOP expression for *V*. [1 mark]

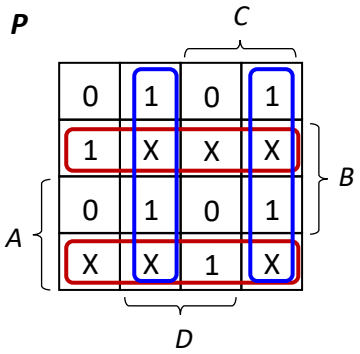
(c) Write out the simplified SOP expression for *P*. [2 marks]

(d) Write an alternative expression for *P* which can be implemented with at most 3 logic gates. (Logic gates allowed are inverters, AND, OR, NAND, NOR, XOR and XNOR. Except for inverters, all logic gates are 2-input gates.) Use dot for AND, plus for OR and single quote for negation. Type the words NAND, NOR, XOR and XNOR if they used, for example: (A NOR B)+(C NAND D). You may type a space before and after the words NAND, NOR, XOR and XNOR, but there should be no space in the rest of your answer. [1 mark]

Answers:

A	B	C	D	V	P
0	0	0	0	1	0
0	0	0	1	1	1
0	0	1	0	1	1
0	0	1	1	1	0
0	1	0	0	1	1
0	1	0	1	0	X
0	1	1	0	0	X
0	1	1	1	0	X
1	0	0	0	0	X
1	0	0	1	0	X
1	0	1	0	0	X
1	0	1	1	1	1
1	1	0	0	1	0
1	1	0	1	1	1
1	1	1	0	1	1
1	1	1	1	1	0





(a) 4

(b) $V = A'\cdot B' + A\cdot B + A'\cdot C'\cdot D' + A\cdot C\cdot D$ or $A'\cdot B' + A\cdot B + A'\cdot C'\cdot D' + B'\cdot C\cdot D$ or

or $A'\cdot B' + A\cdot B + B\cdot C'\cdot D' + A\cdot C\cdot D$ or $A'\cdot B' + A\cdot B + B\cdot C'\cdot D' + B'\cdot C\cdot D$

($A'\cdot B'$ and $A\cdot B$ are the EPIs, so they must be placed before the non-EPIs according to the instruction.)

(c) $P = A'\cdot B + A\cdot B' + C'\cdot D + C\cdot D'$

As all the prime implicants are essential, they can be in any order.

This question is to test students' usage of don't cares.

(d) $P = (A \oplus B) + (C \oplus D)$ or $P = (A \oplus B) \oplus (C \oplus D)$

=== END OF PAPER ===