INSTRUCTIONS TO CANDIDATES

1. This examination paper consists of ELEVEN (11) questions and comprises EIGHT (8) printed pages.

2. This is a CLOSED BOOK examination. Two handwritten A4 reference sheets are allowed. Calculators are not allowed.

3. Answer all questions.

4. Write your answers in the ANSWER BOOKLET provided.

5. Fill in your Matriculation Number with a pen, clearly on odd-numbered pages of your ANSWER BOOKLET.

6. You may use pencil to write your answers.

7. You are to submit only the ANSWER BOOKLET and no other document.
Questions 1 - 6: Each question has only one correct answer. Write your answers in the boxes provided in the Answer Booklet. One mark is awarded for a correct answer and no penalty for wrong answer.

1. Given the following circuit with a half adder and a 2×4 decoder with 1-enable and active-high outputs, what is the Boolean function $F(P,Q,R)$?

![Circuit Diagram]

A. $\Sigma m(3, 5)$  
B. $\Sigma m(5, 6)$  
C. $\Sigma m(2, 4, 6)$  
D. $\Sigma m(1, 3, 5)$  
E. None of the above

2. Which of the following Boolean functions can be implemented using a single 4-bit magnitude comparator (whose block diagram is shown below) without any additional logic gates?

i. $F(A,B,C,D) = \Sigma m(0, 1, 2, 3, 4, 5)$  
ii. $G(A,B,C,D) = \Sigma m(11)$  
iii. $H(A,B,C,D) = \Sigma m(0, 3, 12, 15)$  
iv. $J(A,B,C,D) = \Sigma m(0, 6, 9, 15)$  
v. $K(A,B,C,D) = \Sigma m(2, 8)$

![Comparator Diagram]

A. Only (i), (ii) and (iii)  
B. Only (i), (ii) and (iv)  
C. Only (ii), (iii) and (iv)  
D. Only (i), (ii), (iii) and (iv)  
E. Only (ii), (iii), (iv) and (v)
3. Assuming odd parity scheme, the following Hamming code, including data bits and parity bits, is received with a single-bit error:

```
0 0 1 1 0 0 1 1 0 1 1 0
```

Which of the following is the corrected code with the corrected bit underlined?

A. 0 1 1 0 0 1 1 0 1 1 0
B. 0 0 1 0 0 1 1 0 1 1 0
C. 0 0 1 1 0 1 1 0 1 1 0
D. 0 0 1 0 0 1 1 0 0 1 0
E. 0 0 1 0 0 1 1 0 1 1

4. Suppose Intel announced that they have a new floating-point execution engine that reduces the CPI for floating point instruction by 50%. Which of the following statements is/are TRUE if program X contains 50% of floating point instructions?

i. Total execution time for program X is reduced by 25%.
ii. Average CPI for program X is reduced by 25%.
iii. The instruction count for program X is reduced by 25%.

A. Only (i) is TRUE.
B. Only (i) and (ii) are TRUE.
C. Only (ii) and (iii) are TRUE.
D. (i), (ii) and (iii) are all TRUE.
E. None of the above.

5. Suppose register $s0$ contains a random 32-bit value. Which of the following MIPS statements can clear the content of register $s0$?

i. xor $s0$, $s0$, $s0$
ii. andi $s0$, $s0$, 0
iii. nor $t0$, $s0$, $s0$
   nor $s0$, $s0$, $t0$

A. Only (i) can clear the content of $s0$.
B. Only (i) and (ii) can clear the content of $s0$.
C. Only (ii) and (iii) can clear the content of $s0$.
D. (i), (ii) and (iii) can all clear the content of $s0$.
E. None of the above.
6. Suppose $B$ is the binary encoding for the instruction “addi $s0, s0, 0$.” Given $A$ is the binary encoding of an unknown MIPS instruction, which of the following statements is/are TRUE?

   i. If $2^{15} < A - B < 2^{16}$, then $A$ is an addi instruction that decreases the value in $s0$.

   ii. If $0 \leq A - B < 2^{16}$, then $A$ is an addi instruction that writes into $s0$.

   iii. If $2^{16} < A - B < 2^{21}$, then $A$ is an addi instruction that has a different source register.

A. Only (i) is TRUE.
B. Only (i) and (ii) are TRUE.
C. Only (ii) and (iii) are TRUE.
D. (i), (ii) and (iii) are all TRUE.
E. None of the above.
7. [9 marks]

The first 5 code values in the self-complementing 4221 decimal code are: 0000, 0001, 0010, 0011 and 0110, representing the decimal digits 0, 1, 2, 3, and 4 respectively. We want to implement a sequential circuit that cycles through the 10 (ten) code values of the 4221 decimal code.

a. Implement the sequential circuit using only \( T \) flip-flops, named \( A, B, C \) and \( D \). Write out the simplified SOP expressions for all the flip-flop inputs. If there are alternative simplified SOP expressions, you need only write out one. You do not need to draw the logic diagram. [4 marks]

b. Complete the given state diagram on the answer booklet, by filling in the state values in decimal. For each of the 6 unused states, indicate its next state. [4 marks]

c. Is this sequential circuit, if correctly implemented using the simplified SOP expressions for all the flip-flops inputs, a self-correcting one and why? [1 mark]

8. [8 marks]

a. You want to construct a circuit that takes in a 4-bit unsigned binary number \( ABCD \) and outputs a 4-bit unsigned binary number \( EFGH \) where \( EFGH = (ABCD + 1) / 2 \). Note that the division is an integer division. For example, if \( ABCD = 0110 \) (or 6 in decimal), then \( EFGH = 0011 \) (or 3 in decimal). If \( ABCD = 1101 \) (or 13 in decimal), then \( EFGH = 0111 \) (or 7 in decimal).

Construct the above circuit using a single 4-bit parallel adder and at most one logic gate with no restriction on its fan-in. [4 marks]

b. You want to construct a 4221-to-8421 decimal code converter. The 4221 decimal code is as given in question 7. The converter takes in a 4-bit 4221 decimal code \( PQRS \) and generates the corresponding 4-bit 8421 decimal code \( WXYZ \).

Let’s call the circuit you created in part (a) above the A1H (Add-1-then-Half) device, represented by the block diagram below. Implement your 4221-to-8421 decimal code converter using this A1H device with the fewest number of additional logic gates. [4 marks]
9. [5 marks]
The mysterious MIPS code below assumes that $s0 is a 31-bit binary sequence (i.e. the Most Significant Bit of $s0 is assumed to be zero at the start of the code).

```
add  $t0, $s0, $zero  # make a copy of $s0 in $t0
lui  $t1, 0x8000
lp: beq  $t0, $zero, e
    andi $t2, $t0, 1
    beq  $t2, $zero, s
    xor  $s0, $s0, $t1
s:  srl  $t0, $t0, 1
    j lp
e:
```

a) For each of the following initial values in register $s0, give the hexadecimal value for register $s0 at the end of the code.

i. Register $s0 is initialized to decimal 31 at the beginning of the code. [1 mark]

ii. Register $s0 is initialized to hexadecimal 0xAAAAAAAA at the beginning of the code. [1 mark]

b) Explain the purpose of the code in one sentence (2 marks) or if you can link the purpose to a topic taught in the first half of this course, you will get the full credit. [3 marks]
10. [11 marks]
Mr. Noobie has coded a simple student information management system. He stores the matriculation number and student’s name in two separate arrays: MatricArray and NameArray. Each matriculation number takes up 8 bytes, while each student’s name takes up 24 bytes. Mr. Noobie made sure that information belonging to the same student is stored at the same index in the two arrays.

Consider the following pseudo-code for locating and printing the student information based on a given student’s surname (i.e. the family name), S:

```
i = 0
while ( i < N ) {
    if ( NameArray[i] has the same surname as S ) {
        print NameArray[i]
        print MatricArray[i]
    }
    i++
}
```

Suppose the code is executed on a machine with fully associative cache with a total size of 2048 bytes. There are a total of 64 cache blocks. For simplicity, you may assume:

- Both arrays start on a memory word boundary.
- Both arrays are not sorted.
- Checking the surname requires to read the whole name.
- There are a total of N students.

a) If there are a total of 20 students with surname S, describe the best-case scenario in terms of how the items are located. [2 marks]

b) Give the number of cache misses for (a). [2 marks]

c) Describe the worst-case scenario in terms of how the items are located. [2 marks]

d) Give the number of cache misses for (c). [2 marks]

e) Instead of two separate arrays, suppose we merge the arrays such that the matriculation number is followed by the name of that student (i.e. an array of objects/structures). Give the best-case cache misses and the worst-case cache misses. [3 marks]
11. [11 marks]
Study the following MIPS code:

```mips
addi $s0, $zero, 0
lw $t1, 0($t0)
loop:
    beq $t1, $zero, end
    add $s0, $s0, $t1       # I1 (for part d)
    addi $t0, $t0, 4         # I2 (for part d)
    lw $t1, 0($t0)         # I3 (for part d)
    beq $zero, $zero, loop  # to simulate a jump
end:
```

The code processes array $A$ and produces a result in $s0$. Register $t0$ points to the first array element initially. Suppose the array $A$ contains

\{9, 6, 3, 0, -3, -6, -9\}

Given a pipeline machine with 5 stages, answer the following:

a) If there is no mechanism for handling data and control hazards, what is the total number of cycles needed by the code? [2 marks]

b) If we use data forwarding but no control hazard mechanism, what is the total cycle needed? [2 marks]

c) If we use data forwarding and early branch, what is the total cycle needed? [3 marks]

d) Suppose data forwarding and delayed branching are used and the number of delay slot is 1. Modify the code so that the two branches work properly. For your convenience, use the instruction number I1, I2, I3 if you are simply reordering them. You only need to write the instruction if you are modifying it in anyway. [4 marks]