Final Exam

Started: Dec 12 at 10:54am

Quiz Instructions

ETHICS: All students that are found to have shared their work will receive a score of 0 for all the question. Other actions are also possible.

About this exam:

1. You have **90 minutes** to complete this test.
2. **The exam will be available only on Saturday December 12th, and only from 9 AM to 9 PM.**
3. **The exam Access Code is:** carnival
4. **You must complete and submit your answers before 9 PM Saturday December 12th.**
5. You must provide your answer for each question in the space provided after that question. Use the editing tools provided by canvas to provide the required answer. For example, if a table is requested, then use use the "table" tool.
6. You can submit only once. So wait until you have provided the answer to all question before you click on submit button.
7. As mentioned in the rubric below, only the answer that is completely correct will receive the 2 out of 2, or 1 out of 1. Therefore make sure to provide a completely correct answer.
8. If you have any technical difficulty you may call me 858-208-8593
9. As always, no late submission will be accepted.

The rubric for this exam:

Questions that have 2 points:

2 points: If the answer is completely correct
1 point: If the answer is not completely correct
0 point: Incorrect or no answer

Questions that have 1 point:

1 point: If the answer is completely correct
0 point: If the answer is not completely correct, or it is incorrect, or no answer

Note:

All students that are found to have shared their work will receive a score of 0 for all the question. Other actions are also possible.

Questions
✓ Question 1
✓ Question 2
✓ Question 3
✓ Question 4
✓ Question 5
✓ Question 6

Time Running: Hide
15 Minutes, 12 Seconds

Question 1 — 1 pts

Use the Truth Table to show that:
X = (X **AND** Y) **OR** (X **AND NOT** Y)

the requested truth table must clearly show that the above statement is is true, that is: X = X.Y + X.Y'

NOTE: Make sure to use the above editing and formatting tools (for example **table**) to provide your answer.

HTML Editor

12pt Paragraph

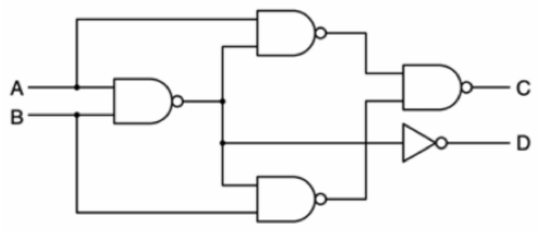
X	Y	X AND Y	X AND NOT Y	(X AND Y) OR (X AND NOT Y)
0	0	0	0	0
0	1	0	0	0
1	0	0	1	1
1	1	1	0	1

table » tbody » tr » td

37 words

Question 2 — 2 pts

What does this circuit do? What is C and D?



HTML Editor

With the four NAND gates, C outputs A **XOR** B, and D simply outputs A **AND** B because it just negates the first NAND gate that takes the input which is equalls to an AND gate of A and B

Summary:

C = A **XOR** B = (A **OR NOT** (A **AND** B)) **AND** ((A **AND** B) **OR NOT** B)

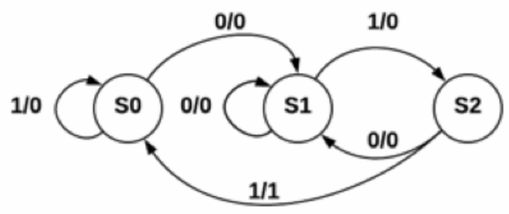
D = A **AND** B

p

65 words

Question 3 — 1 pts

1. Describe in words what the state machine in this figure does.



Finite State Diagram (Mealy Machine)

HTML Editor

This state machine counts the number of times it reads "011"

state S0 checks if it reads 0, if so, move on to state S1, else, stay in S0.

state S1 checks if it reads 1, if so, move on to state S2, else stay in S1.

state S2 checks if it reads 1, if so, move back to state S0, else, back to state S1.

Example:

given an input of:

0010011100

the output will be:

0010010000

p

77 words

Question 4

2 pts

Assume X and Y are stored in R0 and R1 respectively, write a C code that shows what the following ARM code does.

```
        MOV R0, #10
        MOV R1, #60
one:    CMP R0, R1
        BEQ three
        BLT two
        SUB R0, R0, R1
        B one
two:
        SUB R1, R1, R0
        B one
three:
        B three
```

HTML Editor

```
#include <stdio.h>

int main() {
    int r0 = 10;          // MOV R0, #10
    int r1 = 60;          // MOV R1, #60

    one: if (r0 == r1) { // CMP R0, R1
         goto three;     // BEQ three
      if (r0 < r1)
        goto two;        // BLT two
    } else {
        r0 = r0 - r1;    // SUB R0, R0, R1
        goto one;        // B one
    }
    two: r1 = r1 - r0;   // SUB R1, R1, R0
      goto one;          // B one
    three:
        goto three;      // B three
}
```

pre

69 words

Question 5

2 pts

For the following code assume we have an array called **myNumbers**, and the base address of **myNumbers** is stored in R8. write a C code that shows what the following ARM code does.

```
    MOV R1, #5
    MOV R2, #2
    MOV r3, #8
    LDR R4, [R8, R2, LSL #2]
    ADD R4, R4, R1
    STR R4, [R8, R3, LSL #2]
```

HTML Editor

```
int r1 = 5;             // MOV R1, #5
int r2 = 2;             // MOV R2, #2
int r3 = 8              // MOV R3, #8
int r4 = myNumbers[2];  // LDR R4, [R8, R2, LSL #2
r4 = r4 + r1;           // ADD R4, R4, R1
myNumbers[8] = r4;      // STR R4, [R8, R3, LSL #2]
```

pre

51 words

Question 6 — 2 pts

For the following code:

1. Describe what the following code is doing (for example, as a series of C statements).
2. Provide comments to replace comments 1-7, your comments should be in plain English and describe what the code is doing.
3. When executed, what will this code print?

Note: You may use your Raspberry Pi account to execute this code

```
.global main
.func main
main:

      MOV R4, #10 @ Comment 1
      MOV R5, #5  @ Comment 2
      CMP R4, R5  @ Comment 3
      BLE else    @ Comment 4
      MOV R6, R4  @ Comment 5
      B endif     @ Comment 6
else: MOV R6, R5  @ Comment 7
endif:
      MOV R1, R6
      LDR r0, =string
      BL printf
      B exit
exit:
      MOV R7, #1
      SWI 0
.data
string: .asciz "The Answer is: %d\n"
```

HTML Editor

12pt Preformatted

1.

```
int main() {
    int r4 = 10;        // MOV R4, #10
    int r5 = 5;         // MOV R5, 5
    int r6;

    if (r4 <= R5) {     // CMP R4, R5
        goto else;      // BLE else
    } else {
        r6 = r5;        // MOV R6, R5
        goto endif;
    }

    else: int r6 = r5  // else: MOV R6, R5

    endif: r1 = r6;
      printf("The Answers is: %d\n", r1);  // BL printf
      goto exit:

    exit: return 1;     // SWI 0
}
```

2.

```
.global main
.func main
main:

      MOV R4, #10 @ Move immediate int value 10 to register 4
      MOV R5, #5  @ Move immediate int value 5 to register 5
      CMP R4, R5  @ Compare register 4 and 5
      BLE else    @ If register 4 is less than or equal to 5: jump to else label
      MOV R6, R4  @ Move value in register 4 into register 6
      B endif     @ Jump to endif label
else: MOV R6, R5  @ Move value in register 5 into register 6
endif:
      MOV R1, R6
      LDR r0, =string
      BL printf
      B exit
exit:
      MOV R7, #1
      SWI 0
.data
string: .asciz "The Answer is: %d\n"
```

3. OUTPUT:

```
The Answer is: 10
```

pre 187 words

Quiz saved at 12:09pm Submit Quiz