Exam Simulation - Solutions

Resources

The following simulation exam paper:

1. If a floating point encoding uses sign, three bit mantissa and four bit exponent (2's complement). What are the extremes of the interval in which the numbers are encoded? There are no hidden bits and the mantissa is shifted until the first significant bit is to the right of the radial point.

Solution

sign	m	m	m	exp	exp	exp	exp

The sign can be 0 or 1 (+ve or -ve)

The mantissa is 3 bits and represents the fractional component of the number (shifted so that the first non-zero bit is to the right of the radix). For this reason, the smallest value it can take is 100 (representing 0*b*0.100) and the largest is 111 (representing 0*b*0.111).

The exponent is 4 bits in 2's complement so the range is $\left[-2^3\,;\,2^3-1\right]$ r [-8,7]. The represented numbers closest to zero will have the most negative exponent (i.e. -8) and those furthest from zero will have the largest exponent (i.e. 7).

Given this, the range of negative numbers is:

$$\left[-0b0.111\times 2^7\,;\,-0b0.1\times 2^{-8}\right]=\left[-0.875\times 2^7;\,-0.5\times 2^{-8}\right]$$

And the range of positive numbers is:

$$\left[0b0.1\times 2^{-8};\,0b0.111\times 2^7\right]=\left[0.5\times 2^{-8};\,0.875\times 2^7\right]$$

2. If you encode a set of 36 symbols with 6 bits, how many combinations are unused?

Solution

28

6 bits encode 2^6 combinations = 64 possible combinations.

If you only have 36 symbols to encode, then 64-36 = 28 combinations are unused

3. What is the equivalent Boolean expression of $(x+y)(x+z)$?

Solution

$x+yz$

Derivation:

$$(x+y)(x+z)=xx+xz+yx+yz$$

$$=x+xz+yx+yz$$

$$=x(1+z)+yx+yz$$

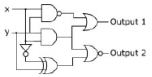
$$=x+yx+yz$$

$$=x(1+y)+yz$$

$$=x+yz$$

NOTE: The term (1+x) will always evaluate as 1

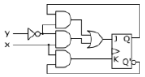
4. What is the value of the outputs when x = 1 and y = 1?



Solution

Output 1 = 1, Output 2 = 0

5. What is the value of Q at time $t+1$ if $X = 0, Y = 0 \; and \; Q = 1 \; at \; time \; t$?



Solution

Q = 1

6. Which registers are modified in the ALU execution stage?

Solution

Status registers

7. Which sequence of stack instructions will you use to rotate the content of registers `R0`, `R1` and `R2`. That is, `R2` takes the value of `R1`, `R1` the value of `R0` and `R0` the value of `R2` as if they were assigned simultaneously?

Solution

```
push R0
push R1
push R2
pop R0
pop R2
pop R1
```

8. The operation code in one instruction format in the AVR architecture has 4 bits. How many possible types of instructions (considering only the op code) can be encoded?

Solution

$2^4 = 16$

9. Consider the following data section in an AVR assembly program:

```
.section .data
d1:     .byte 0x20, 0x21, 0x22, 0x23
        .string "MSG"
```

What is the distance between the address of the last element of the string and the address `d1`?

Solution

7

10. What is the effect of the AVR instruction `STS dst, R20`?

Solution

It stores the content of register `R20` in memory in the position `dst`.

11. What is the effect of the AVR instruction `STD Y + 60, R31`

Solution

It stores the content of register `R31` in the position in memory pointed to by `Y + 60`.

12. If a function receives three 1 byte parameters, returns a 1 byte result and the return address occupies three bytes (no temporary variables), what is the size of the activation block?

Solution

7 bytes.

13. A lock has a numeric keyboard with 10 digits and an extra key with letter G (Go). The lock is programmed to detect the combination G1234 to enter. Design the FSM that detects this combination. How many inputs/outputs and states does it have?

Solution

Inputs: 4 bits to encode the 11 possible input symbol combinations.

Outputs: 1, the lock is open or not.

Transitions: From any state, the right digit advances in the sequence towards the open state. From the open state, G takes you one step, any other symbol, back to initial state. From any of the other states, incorrect symbol takes you to initial state.

14. If you are given the FSM, how would you obtain the circuit to implement it?

Solution

1. Write the truth table from the inputs and state bits, to future state bits.
2. Implement each of the outputs with a circuit.
3. Connect the outputs of the function to a D flip flop.
4. Connect the output of the latch to the inputs of the circuit.

15. Suppose we have the following definition of an array of 8 bit integers.

```
int tbl[7] = {10, 2, 10, 3, 10, 4, 10};
```

Suppose that the size of the array is stored also as an 8 bit integer in the first position of the array (similar to Java arrays).

Write an AVR assembly program that counts the number of times the number 10 appears in the array and leaves the result in `R24`. You may assume the array is declared with a label `tbl`.

Solution

```
;;; Data definitions go here
.section .data
tbl:    .byte 7, 10, 2, 10, 3, 10, 4, 10

;;; Code definition goes here
.section .text
        .global asm_function

asm_function:
        CLR R25
        LDI R27, hi8(tbl)
        LDI R26, lo8(tbl)

        CLR R18             ; counter
        CLR R24             ; result

        LD R19, X+          ; Get the size

loop:
        CP R18, R19         ; Compare, and if equal terminate.
        BREQ done

        LD R20, X+          ; Load the array element
        CPI R20, 10
        BRNE loopincrease   ; If different skip count increase

        INC R24             ; Increase the result
loopincrease:
        INC R18             ; Increase the count
        JMP loop

done:
        RET

.end
```

16. Write a program for Arduino that makes an led on pin 3 blink every second following the signal read from pin 4 and sends through the Serial Monitor a message indicating the operation:

Solution

```
int sensor = 4;
int led = 3;

void setup()
{
    pinMode(sensor, INPUT);
    pinMode(led, OUTPUT);
    Serial.begin(9600);
}

void loop()
{
    if(digitalRead(sensor))
    {
        digitalWrite(led, HIGH);
        Serial.println("Led is high");
    } else
        digitalWrite(led, LOW);
        Serial.println("Led is low");
    }
    delay(1000);
}
```

Workplan

1. Read the entire exam one time.
2. Solve the exam questions in groups and comment with peers
3. Annotate the blocks that you clearly need additional study
4. Divide the exam into blocks.

Assessment

Solve each exam block individually and comment the solutions with your class mates or tutor the solutions.