

# ELEC1601/ELEC9601 Final Exam

ⓘ This is a preview of the published version of the quiz.

Started: Nov 10 at 20:48

## Quiz Instructions

---

### Question 1

1 pts

A computer encodes integers that lie in the range [ -277; 289 ] using a signed-magnitude number scheme.

How many bits are required to represent these numbers?

### Question 2

1 pts

A computer system has  $2^{24}$  bytes of memory, where addresses are encoded using 12 bits; how many bytes are there per memory cell?

### Question 3

1 pts

Complete the truth table for the following boolean function:

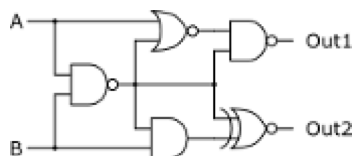
$$A(B' + C) + BC$$

A	B	C	out
0	0	0	<input type="text"/>
0	0	1	<input type="text"/>
0	1	0	<input type="text"/>
0	1	1	<input type="text"/>
1	0	0	<input type="text"/>
1	0	1	<input type="text"/>
1	1	0	<input type="text"/>
1	1	1	<input type="text"/>

#### Question 4

1 pts

Study the following combinational digital circuit:

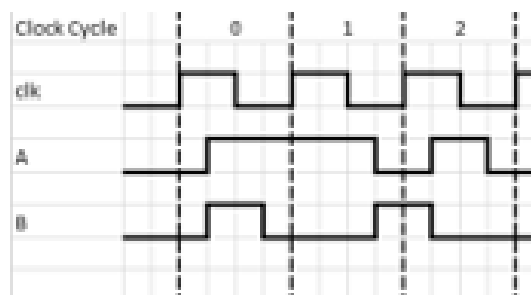
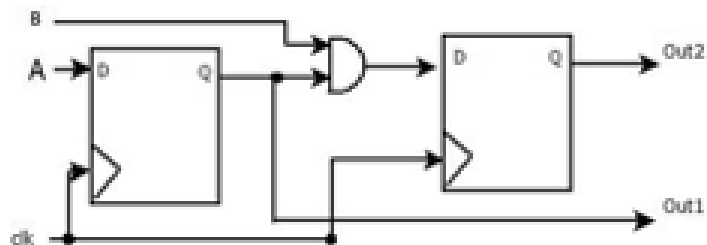


What is the value of Out2 when A = 0 and B = 1?

### Question 5

1 pts

Study the following circuit and timing diagram:



What is the values for Out1 and Out2, during clock cycle 2 just after the falling clock edge? Assume gates have negligible delay

Out1 =

Out2 =

### Question 6

1 pts

Select all registers in the AVR architecture that are changed over the course of the following instruction:

LD R28, -X

You can assume:

- R28 initially has the value 4,
- The address referenced by X contains the value 1
- The address referenced by X+1 contains the value 0
- The address referenced by X-1 contains the value 2

☐ status register

☐ instruction register

☐ 'X' Register

☐ program counter

☐ 'Y' Register

☐ R1 in register file

## Question 7

1 pts

Suppose there are two computer systems:

System A has a 3 stage pipeline, where each stage lasts 50ns,

System B has a 5 stage pipeline, where each stage lasts 30ns.

Assume both systems receive their first instruction at the same time, which system will be the first to complete 5 instructions, assuming no pipeline stalls?

☐ Not enough information to answer question

☐ System A

☐ System B

☐ Both finish at same time

**Question 8****1 pts**

A computer system has the value 0x335 in its stack pointer. The stack grows (when you push data) towards lower memory positions. Suppose the system executes the sequence of instructions:

- PUSH R4
- PUSH R5
- POP R1
- POP R2
- POP R3

What is the value of the stack pointer after this sequence?

(If it cannot be computed given the above information, enter the value 0)

**Question 9****1 pts**

Consider the 16 bit encoding used in the AVR architecture shown in the following figure:

Operation:

- (i) Rd - K

Syntax:

- (i) CPI Rd,K

Operands:

$16 \leq d \leq 31, 0 \leq K \leq 255$

Program Counter:

$PC \leftarrow PC + 1$

16-bit Opcode:



Assume the following values:

R5 = 5

R21 = 85

The instruction being executed (in machine code is): 0011 0000 0101 0101

After the instruction:

R3 has the value:  (write this as a decimal value)

R19 has the value:  (write this as a decimal value)

The Z flag in the status register has the value:

### Question 10

1 pts

Suppose the following command is executed:

`jmp someSubroutine`

Assume that the address of someSubroutine is held in the 'X' Register

Select all registers in the AVR architecture that are modified during this instruction:

☐ Stack Pointer

☐ Status Register

☐ 'X' Register

☐ Instruction Register

☐ Program Counter

☐ R1 in register file

### Question 11

1 pts

Consider the following definitions in an assembly program:

- d1: .byte 0, 1, 2, 3

If d1 is located in address 0xC33D, what is the value stored in that position after executing the following instructions?

- LDI R27, hi8(d1)
- LDI R26, lo8(d1)
- LDI R19, hi8(d1)
- LDI R18, lo8(d1)
- ST -X, R18

## Question 12

1 pts

An AVR assembly program defines the following variables and labels:

```
.section .data
```

```
D1: .byte 3, 2
```

```
D2: .byte 10, 6, 5, 4
```

What is the result if we subtract the address number of label D1 from the address number of label D2

## Question 13

1 pts

How many bytes are reserved in memory if an AVR assembly program has the following data section:

.section .data

.ascii "this question is hard"

.space 70, 10

### Question 14

1 pts

What is the decimal value (base 10) held in R9 after the following sequence of instructions?

LDI R18, 0x287

MOV R9, R18

ADD R9, R18

### Question 15

1 pts

Consider the AVR assembly instruction:

LDI R31 hi8(label)

LDI R30 lo8(label)

LD R12, Z+

(label has been defined in the data section).

Where is the microprocessor getting the operand loaded in R30?



- ☐ From the data memory
- ☐ From one field of the instruction
- ☐ From the register file.
- ☐ From the program memory

### Question 16

1 pts

Register X contains an address with value 0x012F.

The values in Data memory at addresses 0x012E, 0x12F and 0x0130 are 0x02, 0x05 and 0x03 respectively.

What is the value of Register X after the following instruction?

LD R12, X+

(write the base10 equivalent)

### Question 17

1 pts

Suppose a program invokes a subroutine.

The subroutine requires 8 input parameters and 7 global variables to operate.

Assume the parameters are passed to the subroutine and result returned through the stack. The result is 1 byte.

What is the offset (in terms of number of bytes) between the stack pointer and the result after returning from the subroutine?

### Question 18

1 pts

Study the following program

It was generated by the compiler avr-gcc that uses the AVR libc library, so obeys the convention for register management

```
LDI R5, 5
LDI R20, 6
LDI R21, 7
PUSH R21
CALL my_subroutine ; Call the subroutine
POP R0
```

The result of the subroutine call was the integer 0x5123.

State the values (in base 10) in each of the following registers after the end of this section of code. If unknown, type UNKNOWN.

R5:

R20:

R21:

R24:

R25:

### Question 19

1 pts

Suppose a subroutine receives 4 1 byte parameters, returns a 1 byte result, and the return address occupies 3 bytes. Internally, the subroutine needs 3 local register. The

subroutine returns a 1 byte result via the stack.

The program was generated by the compiler avr-gcc that uses the AVR libc library, so obeys the convention for register management

Once in the subroutine, at the first instruction, what is the required displacement (in bytes) from the top of the stack to the memory location where the result will be stored?

## Question 20

1 pts

Study the following code (you can assume ". . ." refers to code that is not shown

```
main:
    ...
    LDS R25, x
    CALL subroutine2
    ...
    PUSH R25,
    CALL subroutine1
    POP R0,
    ...
subroutine1:
    ...
    MOV R25, R8
    CALL subroutine2
    ...
    MOV R24, R7    ; result
    RET
subroutine2:
    ...
    RET
```

How many calls to subroutine 2 are made in this code fragment (you can assume that there all subroutine calls are shown in this code?)

## Question 21

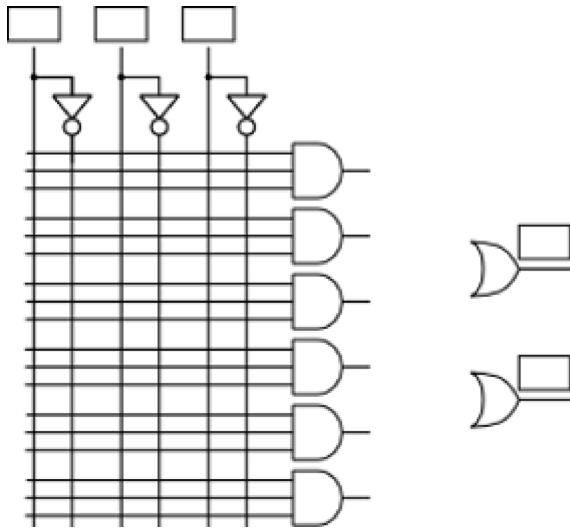
1 pts

Study the following Boolean Expressions:

$$F = ((ABC)' (A + BC'))'$$

$$G = B' + C$$

Show how this could be implemented on the following device:



Draw your answer, take a picture and upload it

Upload

Choose a File

## Question 22

1 pts

Imagine a door lock that is password protected. Its input is a two-bit unsigned number, and a reset button. The user can only press one input at a time.

To open the door, a user must first press reset. Three consecutive inputs from the next four inputs must contain the sequence 1, 3, 0. The door only ever opens after receiving the 4th input.

(For example, 'reset, '1', '3', '0', '0' or 'reset, '1', '3', '0', '1' or 'reset, '3', '1', '3', '0' are all valid inputs to open the door.)

(For example, 'reset, '1', '0', '3', '0' or 'reset, '1', '3', '3', '0' do not open the door.)

After the door has been unlocked, after 30 seconds it is automatically re-locked.

1. Draw the state transition diagram for the finite state machine (FSM) encoding such a system. It must include the value of the inputs and output (door unlocked or door locked). Try to use as few states as possible.

Take a picture of your final drawing and upload this to Canvas

Upload

## Question 23

1 pts


Consider the following fragment of code written in a high level programming language:

```
int data[5] = {15, 25, 36, 12, 50};
int result = 0;
int count = 0;
for (count = 0; count <= 4; count++) {
{
    if (data[count] > result)
        result = result+data[count];
    else
        result = my_function(data[count]);
    count = count + 1;
}
```

Write a sequence of AVR assembly instructions equivalent to the previous code. You can assume my\_function1 and my\_function2 are written elsewhere. You do not need to write this, but you can access it by calling the subroutine.

```
.segment .data
data: .byte 15, 25, 36, 12, 50
result: .space 1, 0
count: .byte 0
```

Edit View Insert Format Tools Table

12pt ▾ Paragraph ▾ | **B** *I* U A ▾  ▾ T<sup>2</sup> ▾ |

 ▾  ▾  ▾ |  ▾ |  ▾  ▾  ▾ |   ▾  

p



0 words

</>



## Question 24

1 pts


Complete the following program that monitors the status of a sensor connected to digital PIN 5 (copy, then fill in gaps. Note that you do not need to fill in every gap). The program should achieve the following:











- When the sensor is active, the program makes an LED connected to PIN 9 flash off and then on for half a second each and sends the message SENSOR UP through the serial communication link at 9600bps
- When the sensor is not active, the LED is left on and the message SENSOR DOWN is sent to the serial communication link.

```
//Global variable declarations
bool sensor_on = false;
(gap 1) .....
(gap 2) .....
void setup()
{
  (gap 3) .....
  (gap 4) .....
  (gap 5) .....
}
void loop ()
{
  (gap 6) if (.....)
  {
    (gap 7) .....
    (gap 8) .....
    (gap 9) .....
    (gap 10) .....
```


```
}  
else  
{  
  (gap 11) .....  
  (gap 12) .....  
  (gap 13) .....  
  (gap 14) .....  
  (gap 15) .....  
  (gap 16) .....  
  (gap 17) .....  
  (gap 18) .....  
}  
}
```

Edit View Insert Format Tools Table

12pt ▾ Paragraph ▾ | **B** *I* U A ▾  ▾  $T^2$  ▾ |

 ▾  ▾  ▾ |  ▾ |  ▾  ▾  ▾ |   ▾  $\sqrt{x}$  

p

  | 0 words |   

Not saved

Submit Quiz