**1 AVR architecture**

**1.1 Which of the following is placed in the CPU of the ATmega2560? (If yes in parenthesis that means it has it)**

· The 32 general purpose registers? (Yes)

· The SRAM

· The ALU (Yes)

· The input/output Pin/Ports

· The PC (program Counter) (Yes)

· The program storage (flash memory)

· The Status register (Yes)

**1.2 What is the Stack and what is its address?**

Stack is temporary data storage used for storing local variables and for storing return addresses, it also has a stack pointer register which always points to the top of the stack. The stack pointer is incremented by three for ATmega2560 when data is popped from the stack with return from subroutine RET or return from interrupt RETI. It decrements by 3 when return address is pushed onto the stack with subroutine call or interrupt.

The ATmega2560 addressing is at 32K/64K/128K program memory locations.

**2 Delay calculations**

**2.1 How many clocks-cycles does the delay “function” take? And given a 16MHz clock. How long is the delay? Show your calculations:**

**delay:**  
ldi r16, 201  
l1:  
nop  
nop  
dec r16  
brne l1

Calculation: 1+(1+1+1+2)x200+1+1+1+1 = 1005

**2.2 How many clocks-cycles does the delay “function” take? Show your calculations:**

**delay:**

ldi r17, 128  
l2:  
ldi r16, 100  
l1:  
nop  
nop  
dec r16  
brne l1  
nop  
dec r17  
brne l2

Calculation: 1+(1+(1+1+1+2)x100-1+1+1+2)x128-1 = 64512

**2.3 How many clocks-cycles does the delay “function” take? And given a 16MHz clock. How long is the delay? Show your calculations:**

**delay:**

ldi r18, 7  
l3:  
ldi r17, 55  
l2:  
ldi r16, 222  
l1:  
dec r16  
brne l1  
nop  
dec r17  
brne l2  
dec r18  
brne l3

Calculation: 1+(1+(1+(1+2)x222-1+1+1+2)x55-1+1+2)x7-1 = 257971 cycles

16MHz = 16000000

257971/16000000 = 0.0161 second’s delay

**3 Write an assembly program that makes the test LED blink with approximately 10 Hz**

LDI R16, 2

loop:

LDI R17, 254

l2:

LDI R18, 225

l3:

NOP

NOP

NOP

NOP

DEC R18

BRNE l3

DEC R17

BRNE l2

DEC R16

BRNE loop

NOP  
NOP

**4 Write a program that add two values. Send the result to PORTB**

The two values can simply be hardcoded in the software from the beginning.

LDI R16, 0x50

LDI R17, 0x90

ADC R16, R17

OUT PORTB, R16

**5 Write a program that multiplies two unsigned 8-bit values, which are stored in register 17 and register 18. Send the least significant byte of the result to port A and the most significant byte of the result to port B.**

LDI R17, 0x26

LDI R18, 0x75

MUL R17, R18

OUT PORTB, R1

OUT PORTA, R0

**6 Find the contents of R20 after each of the following is executed. Start by doing it by hand, and then test your result using the simulator.**

**6.1**

LDI R20, 0x56

SWAP R20

CLC

ROR R20

ROR R20

R20 = 10011001

**6.2**

LDI R20, 0x39

SEC

ROL

ROL

R20 = 11100110

**6.3**

CLC

LDI R20, 0x4D

SWAP R20

ROL R20

ASR R20

R20 = 11010110

**6.4**

CP R20, R20

LDI R20, 0x7A

ROR R20

R20 = 00111101

**7 Switch**

**7.1 What the purpose of the build in PullupResistor?**

The purpose of the built in Pullup resistor is to set a default value to PINs

**7.2 How do you enable the building Pullupresistor?**

By setting the PORTX to 1.

**7.3 Write a program that turns a LED ON and OFF according to the value of a switch. The switch should be connected to PA0 and the LED to PB5.**

LDI R16, 0b00100000; on

LDI R17, 0; off

OUT DDRB, R16; LED is an Output

OUT DDRA, R17; Switch is an Input

ever:

SBIC PINA, 0

OUT PORTB, R16

SBIS PINA, 0

OUT PORTB, R17

JMP ever