Q4. Subtract contents of locations 0x14C to 0x153 from contents of the corresponding memory location from 0x154

A. if borrow = 1 store result in 154 + 15

B. if borrow = 0 " " 154 + 30

y →	0x14C	0x14D	0x14E	0x14F	0x150	0x151	0x152	0x153
x →	0x154	0x155	0x156	0x157	0x158	0x159	0x15A	0x15B

Counter → 8

~~LDI yH, 0x14D~~

~~LDI yH, 0x01 ; higher y~~

~~LDI ZL, 0x154~~

~~LDI ZH, 0x01 ; higher x~~

~~LDI R16, 0x08 ; counter~~

~~LDI R20, 0 ; result of SUB~~

Cancelled

again:

~~SUB ZL, yL~~

~~DEC R16~~

~~BRNE again~~

~~BREQ H...~~

~~... ; borrow = 1~~

~~STS 15, yL~~

~~RJMP~~

answer Q4 A.

```
LDI YL , 0x40
LDI YH , 0x01

LDI ZL , 0x54
LDI ZH , 0x01

LDI R16 , 0x08

again :  LD R20 , Y+
         LD R21 , Z+
         SUB R20 , R21
         BRCS L1
L2:      ST Z+30 , R20     ; borrow =0
         DEC R16
         RJMP again

L1 :     ST Z+15 , R20     ; borrow =1
L2:      DEC R16
         BRNE again
```

12 1/2

بهذه الطريقة مستكرر الـ loop forever الغرض RJMP L2

Q4. B get data from port B , and Multiplied it by 2 send it at port D if result correct.

```
LDI R16 , 0
OUT DDRB , R16
LDI R16 , 0xFF
OUT DDRD , R16
IN R18 , PINB
LSL R18
BRCC L1
RJMP end
L1: OUT R18 , PORTD
End:
```

9

بهذه الطريقة هينزل ينفذ L1 حتى لو لم يتحقق الشرط

Consider the instructions below and complete the table [16 marks]

16

```
0x0000          LDI     R20, 0xE5
0x0001          LDI     R21, 0x87
0x0002          CALL    CALCLATION
.= 0x0004 Here: RJMP    Here

0x0005 CALCLATION: PUSH  R21
0x0006             PUSH  R20
0x0007             ADD   R20, R21
0x0008             SUBI  R21, 0xA4
0x0009             POP   R10 ——→ R10 = (stack)
0x000A             POP   R11 ——→ R11 = (stack)
0x000B             RET
```

After the execution of:	Contents of some registers											Stack
	R20	R21	R10	R11	SP	Z	C	H	S	V	N	
LDI R20, 0xE5	0xE5	0	0	0	0x8FF	0	0	0	0	0	0	
LDI R21, 0x87	0xE5	0x87	0	0	0x8FF	0	0	0	0	0	0	
CALL CALCLATION	0xE5	0x87	0	0	0x8FD	0	0	0	0	0	0	CH CC
PUSH R21	0xE5	0x87	0	0	0x8FC	0	0	0	0	0	0	87 CH CC
PUSH R20	0xE5	0x87	0	0	0x8FB	0	0	0	0	0	0	E5 87 CH CC
ADD R20, R21	0x6C	0x87	0	0	0x8FB	0	1	0	1	1	0	E5 87 CH CC
SUBI R21, 0xA4	0x6C	0xE3	0	0	0x8FB	0	1	0	1	0	1	E5 87 CH CC
pop R10	0x6C	0xE3	0xE5	0	0x8FC	0	1	0	1	0	1	87 CH CC
pop R11	0x6C	0xE3	0xE5	0x87	0x8FD	0	1	0	1	0	1	CH CC
RET	0x6C	0xE3	0xE5	0x87	0x8FF	0	1	0	1	0	1	

Good Luck
Dr. Huda Jomma

Correct

1- After a write operation in the EEPROM the user should clear EEPE bit. (15 marks)

EEPE cleard automatically by hardware

2- SBI and SBIS instructions can be used for any bits of locations 0x00-0x60 of the flash memory.

used with 32 GPIO (0x00 to 0x1F) of I/O memory that is bit accessible and addressible

3- On the ATmega328P microcontroller, Ports B & D both can be used as input to digital and analog signals.

port C can used analoge and digital

4- LDI R20, 0x0D ← 0000 1101 (to toggle)
STS EICRA, R20

The above instructions will make the external interrupt connected to INT0 pin is activated on falling edge and INT1 is activated on any logical change.

INT0 activated on any logical change

INT1 activated on rising edge

5- LDI R16, 0xCC 1100 1100
OUT DDRD, R16

The above instructions will make portD as input port.

make pins 2,3,6,7 in portD as output

and rest of pins (0,1,4,5) as input

6- The interrupt execution response for enabled interrupts is 5 clock cycles maximum, one for pushing the PC onto the stack, one to the decrement SP by 3, one to set the I-bit, and one to execute the JMP instruction.

response for enabled is 4 clock cycles minimum, 1 clock for pushing PC to stack, decrement SP by 2 and set I bit in SREG, and 3 clocks for jump instruction 13/4

7- RJMP and JMP instructions are both 16 bits instruction.

RJMP is 16 bits while JMP is 32 bits instruction

8- When two unsigned numbers are added, the overflow flag indicates an overflow.

» » the carry flag indicates an overflow

9- We can perform the "COM R20" instruction using the instruction SUB 0, R20 13/4

we can perform using SUB, R0, R20

where is R19 = 0xFF

10- LD X-, R15 LD X-, R16 ; load from data: X

11- SBCI 320, R9 SBCI R9, 0x32 ; range 0xFF 3.14

12- STS XL, 0xFA2 STS 0xFA2, XL

(4marks)

Q3) Complete the table shown below.

Instruction	Addressing mode	Number of bits					
		Opcode	GPRs	immediate	pointers	Displacement	I/O
OR R5, R6	Direct, two Registers addressing mode	6 bits	10 bits	\	\	\	\
ROR ZL	Indirect, Single register addressing mode	16 bits	\	\	16 bits	\	\
ANDI R22, 0x38	Direct immediate addressing mode	4 bits	4 bits	8 bits	\	\	\

(24 marks)

Q4) Answer the following

1a) Write assembly program that subtracts the contents of memory locations 0x14C – 0x153 from the contents of the corresponding memory locations started at 0x154.

- If there is a borrow store the results on the memory locations started at the location away 15 from the memory location 0x154.
- If there is not a borrow store the results on the memory locations started at the location away 30 from the memory location 0x154.

(14 marks)

b) Write assembly program that gets data from Port B and then multiplied it by 2 (don't use MUL instruction).

- If the multiplication result is correct send it to Port D.

(10 marks)