به نام خدا

حدیث غفوری 9825413

سوال 1

```
;timer clock = 8MHz / 64 = 125kHz => timer period = 1/125 kHz = 8us
;timer value = 1920us / 8us = 240


;Timer2 delay
DELAY:LDI   R20,-240        ;R20=0x10
      OUT   TCNT2,R20       ;load Timer2
      LDI   R20,0x04
      OUT   TCCR2,R20       ;Timer2 , normal mode,int clk,prescaler 64
AGAIN:IN    R20,TIFR        ;read TIFR
      SBRS  R20,TOV2        ;if TOV2 is set skip next instructin
      RJMP  AGAIN
      LDI   R20,0x0
      OUT   TCCR2,R20       ;stop Timer2
      LDI   R20, 1<<TOV2
      OUT   TIFR,R20        ;clear TOV2 flag
      RET
```

سوال 2

```
;we should use prescaler = 256 since we cannot use a decimal point
;to wait 250clocks we should load OCR2 with 250-1 = 249
;  TIMER2 DELAY

DELAY:LDI   R20,0
      OUT   TCNT2,R20       ;TCNT2 =0
      LDI   R20,249
      OUT   OCR2,R20        ;OCR2 = 249
      LDI   R20,OxOE
      OUT   TCCR2,R20       ;Timer2 , CTC mode, prescaler = 256

AGAIN:IN    R20,TIFR        ;read TIFR
      SBRS  R20,OCF2        ;if OCF2 is set skip next inst.
      RJMP  AGAIN
      LDI   R20,0x0
      OUT   TCCR2,R20       ;stop Timer2
      LDI   R20,1<<OCF2
      OUT   TIFR,R20        ;clear OCF2 flag
      RET
```

سوال 3

```c
#include <avr/io.h>
int main(void)
{

    PORTB = 0x01; // activate pull-up of PB0
    DDRC = 0xFF;  // PORTC is output
    DDRD = 0xFF;  // PORTD is output

    TCCR1A = 0x00; // output clock source
    TCCR1B = 0x07; // output clock source

    TCNT1H = 0x00; // set count to 0
    TCNT1L = 0x00; // set count to 0

    while (1)
    {
        do
        {
            PORTC = TCNT1L;
            PORTD = TCNT1H;

        } while ((TIFR & (0x1 << TOV1)) == 0); // wait for TOV1

        TIFR = 0x1 << TOV1; // clear TOV1
    }
}
```

سوال 4

```c
#include <avr/io.h>
#include <avr/interrupt.h>

int main(void)
{

    DDRB |= 0x20; // DDRB.5=output
    TCNT0 = -32;  // timer value for 4us
    TCCR0 = 0x01; // normal mode ,int clk,no prescaler
    TIMSK = (1 << TOIE0);
    sei();

    DDRC = 0;    // PORTC is input
    DDRD = 0xFF; // PORTC is output

    while (1)
    {

        PORTD = PINC;
    }
}

ISR(TIMER0_OVF_vect)        //ISR for timer0 overflow
{
    TCNT0 = -32;
    PORTB ^= 0x20; // toggle PORTB.5
}
```