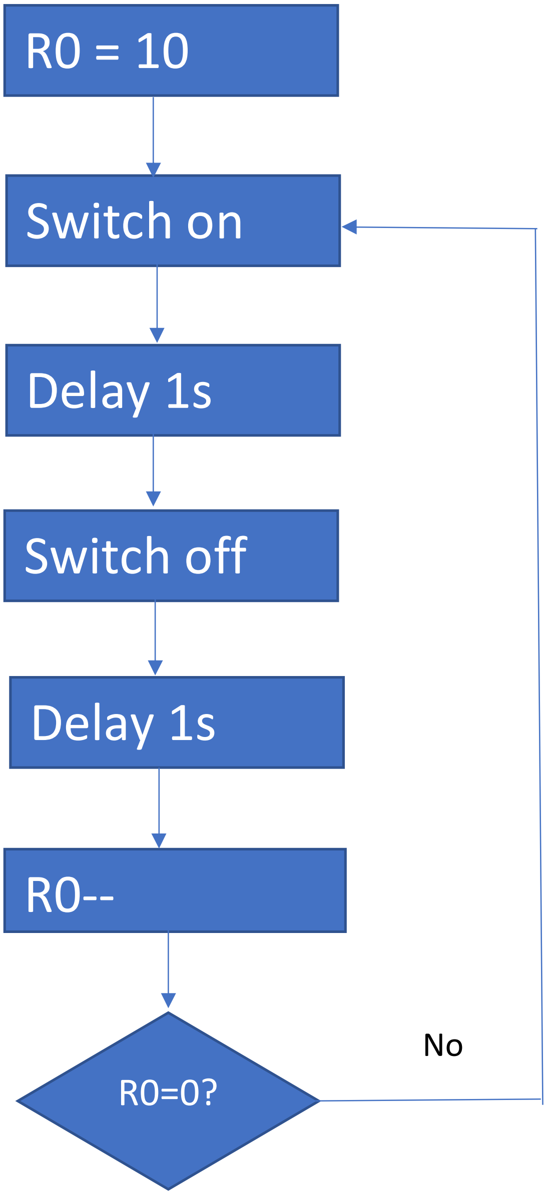
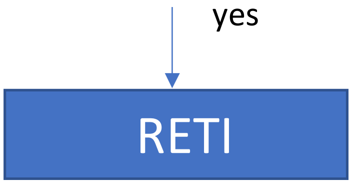
Interrupts and Timers in Atmel AVR Atmega

Aim : To write an ISR to switch on LED for few seconds and then switch off. Program should be written in assembly and c language.

Question : To make the led blink with 1s intervals of ON and OFF for 10 times on interruption.

Solution : Flowchart for interrupt subroutine





Code : Assembly language

```
.org 0x0000
rjmp reset

.org 0x0002
rjmp int1_ISR

reset:
                         ;Loading stack pointer address
         LDI R16,$70
         OUT SPL,R16
         LDI R16,$00
         OUT SPH,R16

         LDI R16, $01     ; Interface port B pin0 to be output
         OUT DDRB, R16    ;so to view LED blinking

         LDI R16,$08
         OUT DDRD,R16

         LDI R16, $00     ;Set MCUCR register to enable low level interrupt
         OUT MCUCR,R16

         LDI R16, $80     ;Set GICR register to enable interrupt 1
         OUT GICR,R16

         LDI R16, $00
         OUT PORTB, R16

         LDI R16, $08
         OUT PORTD, R16
         LDI R16, $00
         OUT DDRD, R16

         SEI
ind_loop:rjmp ind_loop

int1_ISR:

        IN R16,SREG
        PUSH R16

        LDI R16,0x0A
        MOV R0,R16

                           ;Modify below loops to make LED blink for 1 sec
        LDI R16,0x00
        OUT PORTB,R16
```

```
C1:     LDI R16, $01        ;setting PORTB
        OUT PORTB,R16

;To achieve 1s delay = 1[clock time period] x (250 x 5[Clock cycles])[First loop] x
200[2nd loop] x 4[3rd loop]
;250x5 x 200 x 4 = 1,000,000 clock cycles

        LDI R18,4
A3:     LDI R16,200
A2:     LDI R17,250
A1:     NOP
        NOP                  ;delay of one sec
        DEC R17
        BRNE A1
        DEC R16
        BRNE A2
        DEC R18
        BRNE A3

        LDI R16, $00
        OUT PORTB,R16        ;clearing PORTB

        LDI R18,4
A3:     LDI R16,200
A2:     LDI R17,250
A1:     NOP
        NOP                  ;delay of one sec
        DEC R17
        BRNE A1
        DEC R16
        BRNE A2
        DEC R18
        BRNE A3


        DEC R0
        BRNE C1

        POP SREG
        RETI
```

Code : C language

```
#define F_CPU 1000000  // clock frequency

#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

ISR (INT1_vect)
{
       int i;
       for (i=1;i<=10;i++) // for 10 times LED blink

       {
              PORTB=1;
              _delay_ms(1000);   // delay of 1 sec
              PORTB=0;
              _delay_ms(1000);

       }
}


int main(void)
{
       //Set the input/output pins appropriately
       //To enable interrupt and port interfacing
       //For LED to blink
                 //Set appropriate data direction for D
       DDRB=1;   //Make PB0 as output
       MCUCR=0;  //Set MCUCR to level triggered
       GICR=128; //Enable interrupt 1
       DDRD=8;
       PORTD=8;
       DDRD=0;
       sei();    // global interrupt flag

       while (1) //wait
       {

       }
}
```

Inferences:

- Programming in c language has reduced the logical complexity when compared to assembly language.
- Learned about external interrupt programming.
- Practical understanding of how the chips are used by an example based on LEDs.

Name : Dharahas H

Rollno. : EE20B04!