## Objective: AVR Microcontroller I/O Programming In C.

**Task-1 :-**  Let us assume that 8 LEDs are connected to pins of PORTB. Write an AVR C program that shows count from 0 to FFH on the LEDs.

**Code :-**

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <util/delay.h>

int main(void) {

DDRB = 0xFF;

while (1)

{

for (int i = 0; i <= 255; i++)

{

PORTB = i;

\_delay\_ms(50);

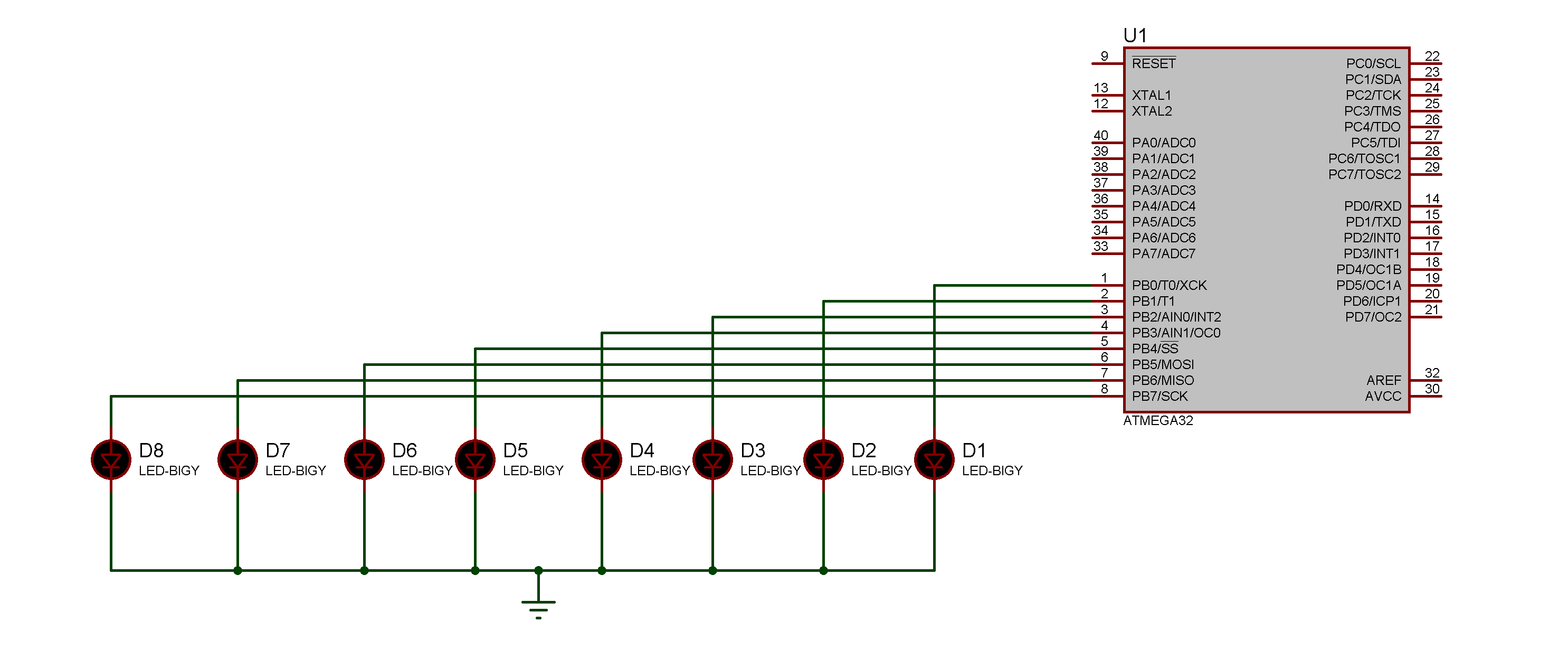
}

}

return 0;

}

**Circuit :-**



**Task-2 :-**  Configure portB as input port, configure portC as output port. Write an AVR C program to get a byte from PortB and then send it to PortC.

**Code :-**

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <util/delay.h>

int main(void) {

DDRB = 0x00;

DDRC = 0xFF;

while (1) {

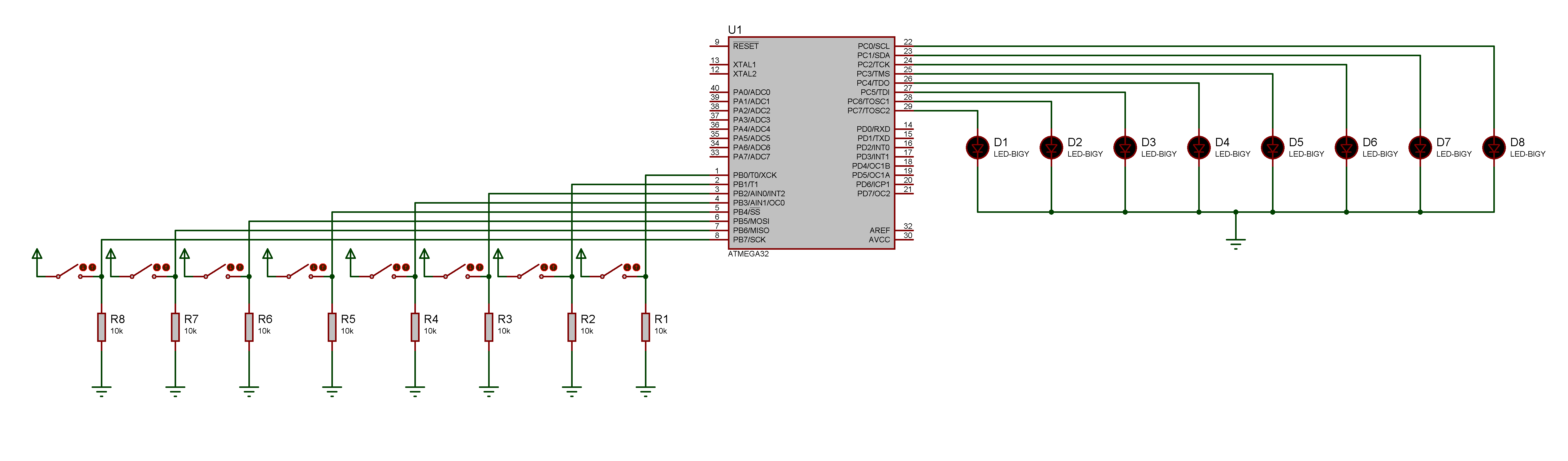
PORTC = PINB ;

}

return 0;

}

**Circuit :-**



**Task-3 :-**   Write an AVR C program to get a byte of data from PORTC. If it is less than 200, send it to PortB, otherwise send it to Port D.

**Code :-**

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <util/delay.h>

int main(void) {

DDRB = 0xFF;

DDRC = 0x00;

DDRD = 0xFF;

while (1) {

if(PINC < 200) {

PORTB = PINC;

PORTD = 0x00;

}

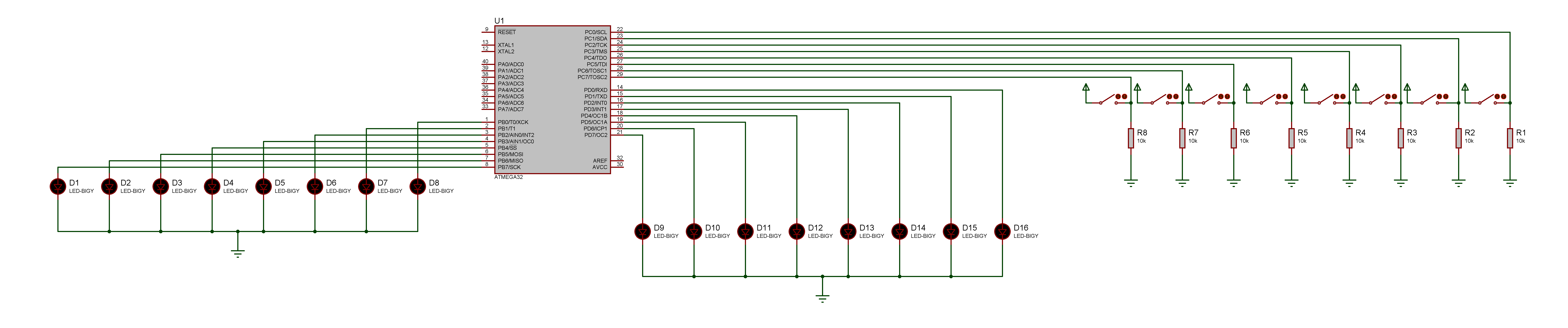
else {

PORTD = PINC;

PORTB = 0x00; }}

return 0; }

**Circuit :-**



**Task-4 :-**  Write an AVR C program to toggle only bit 7 of PortB continuously without disturbing the rest of the pins of PortB.

**Code :-**

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <util/delay.h>

int main(void) {

DDRB = 0xFF;

PORTB = 0xFF;

while (1) {

PORTB = PORTB ^ 0x80 ;

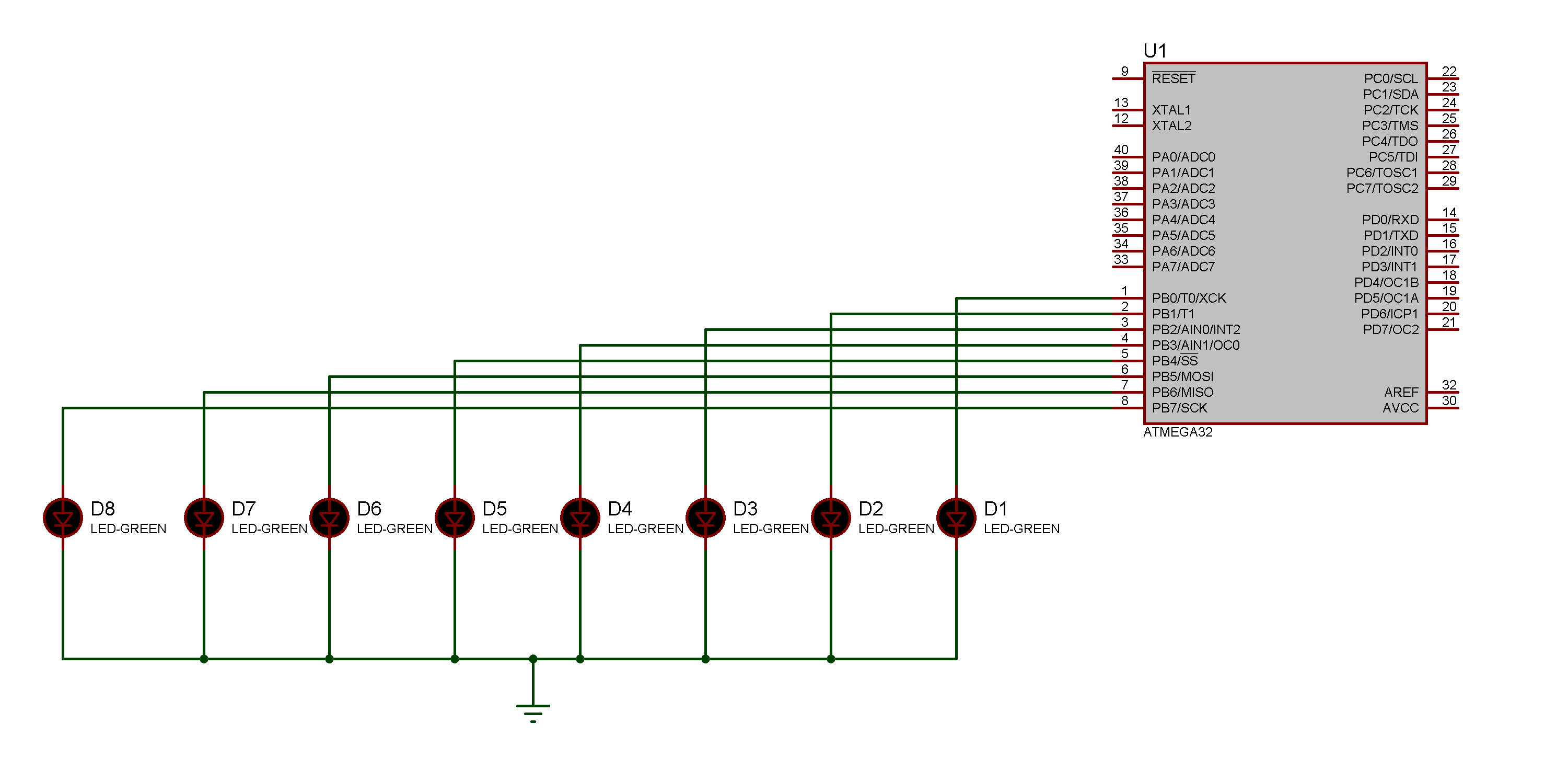
\_delay\_ms(10);

}

return 0;

}

**Circuit :-**



**Task-5 :-**  Write an AVR C program to monitor bit 5 of port C. If it is high, send FFH to PortB , otherwise send 00H to PortB.

**Code :-**

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <util/delay.h>

int main(void) {

DDRC = 0x00;

DDRB = 0xFF;

while (1) {

if(PINC == 0x20) {

PORTB = 0xFF ;

} else {

PORTB = 0x00 ;

} }

return 0;

}

**Circuit :-**

