Aditi Tapariya
U21EE081

# MICROCONTROLLER AND MICROPROCESSOR LAB

## EXPERIMENT 7

**AIM**: Write an embedded C program to toggle the port pin with software delay.

**SOFTWARE USED**: Keil uVision5

**Question-1:** Blink all the LEDs connected to port P1 with regular delay.

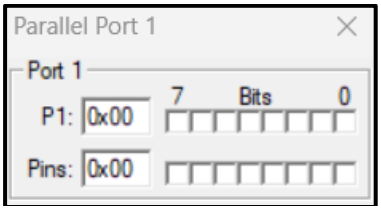
**Code**:

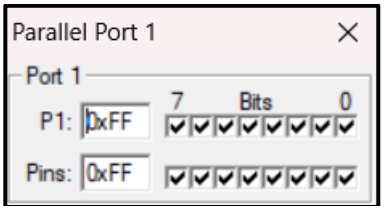
```
#include<reg51.h>
void delay(int);
void main ()
{
        p1=0x00;
        while (1)
        {
                p1=0xff;
                delay (500);
                p1=0x00;
                delay (500);
        }
}

void delay (int t) {
        unsigned int i, j;
        for (i=0; i<t; i++) {
                for (j=0; j<t; j++) {
                        ;
                }
        }
}
```

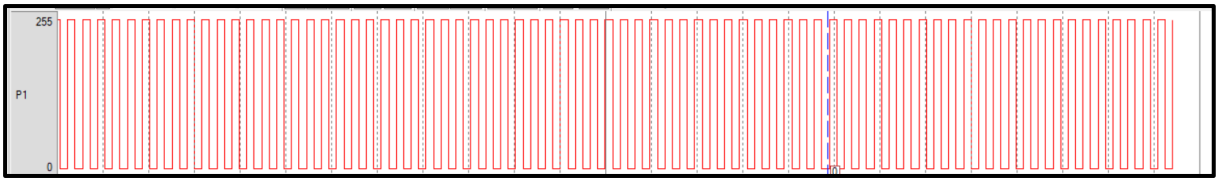
**Algorithm**:

1. Initialize port P1 as output.
2. Enter an infinite loop.
3. Set all LEDs connected to port P1 on.
4. Call the delay function with a parameter of 500.
5. Turn off all LEDs connected to port P1.
6. Call the delay function with a parameter of 500.
7. Repeat steps 3-6 indefinitely.

**Result**:






**Conclusion:**
The code blinks LEDs connected to port P1 by turning them on and off alternately with a delay of 500 milliseconds, executing this sequence continuously.

**Question 2:** Blink alternative LED connected to port P1 with regular delay.

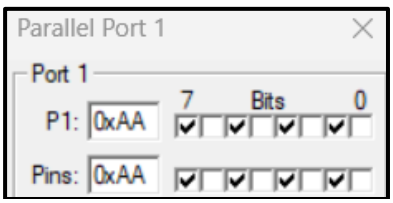
**Code**:

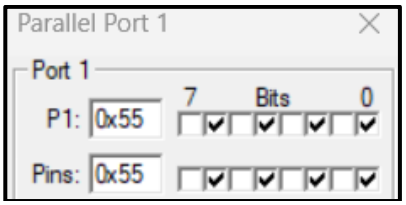
```
#include<reg51.h>

void delay(int);

void main()
{
	P1=0x00;
	while(1)
	{
		P1=0xaa;
		delay(500);
		P1=0x55;
		delay(500);
	}
}

void delay(int t){
	unsigned int i,j;
	for(i=0;i<t;i++){
		for(j=0;j<t;j++){
			;
		}
	}
}
```

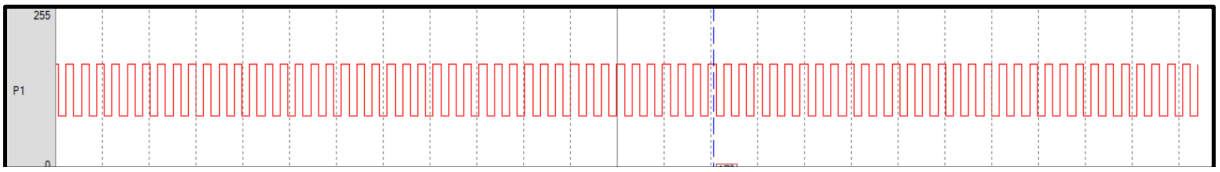
**Algorithm**:

1. Initialize port P1 as output.
2. Enter an infinite loop.
3. Set alternate LEDs connected to port P1 by assigning the value 0xAA (10101010 in binary).
4. Call the delay function with a parameter of 500.
5. Set alternate LEDs connected to port P1 off by assigning the value 0x55 (01010101 in binary).
6. Call the delay function with a parameter of 500.
7. Repeat steps 3-6 indefinitely.

**Result**:




**Conclusion:**
The code alternately blinks LEDs connected to port P1 by setting every other LED on and off, repeating this pattern with a delay of 500 milliseconds continuously.

**Question-3:** Shift the blinking of the LED connected to port p1 from left to right by keeping the previous LED off.

**Code**:

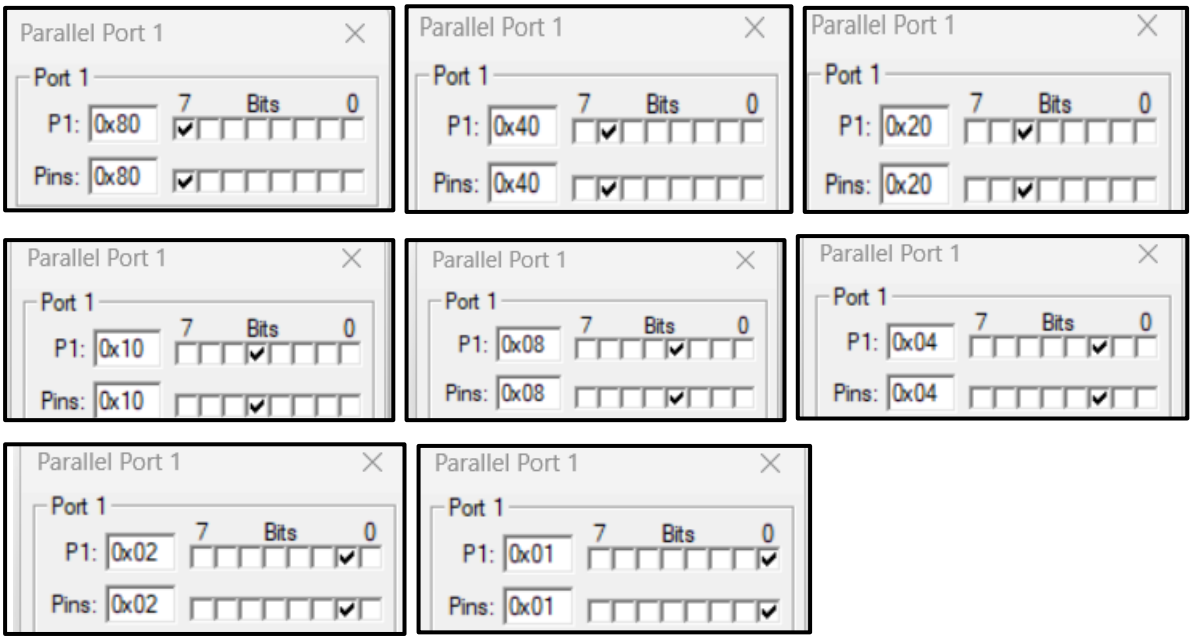
```
#include<reg51.h>
void delay(int);
void main()
{
	P1=0x80;
	while(1)
	{
		if(P1==0x00){
			P1=0x80;
		}
		delay(500);
		P1=P1>>1;
	}
}
void delay(int t){
	unsigned int i,j;
	for(i=0;i<t;i++){
		for(j=0;j<t;j++){
			;
		}
	}
}
```

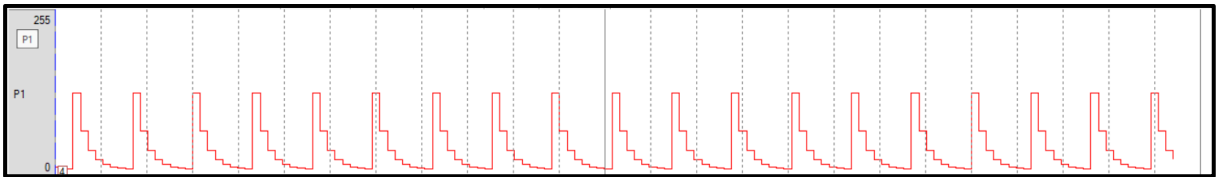
**Algorithm**:

1. Initialize port P1 with the value 0x80 to turn on the rightmost LED.
2. Enter an infinite loop.
3. Check if all LEDs are turned off (P1 == 0x00).
4. If all LEDs are off, reset the pattern to start from the rightmost LED (P1 = 0x80).
5. Call the delay function with a parameter of 500.

6. Right shift the value of port P1 to shift the LED pattern one position to the right.
7. Repeat steps 3-6 indefinitely.

**Result**:




**Conclusion:**
The code shifts the blinking LED connected to port P1 from left to right, ensuring the previous LED is turned off before lighting up the next one, with a delay of 500 milliseconds.

**Question-4:** Shift the blinking of the LED connected to port P1 from right to left by keeping the previous LED off.

**Code**:

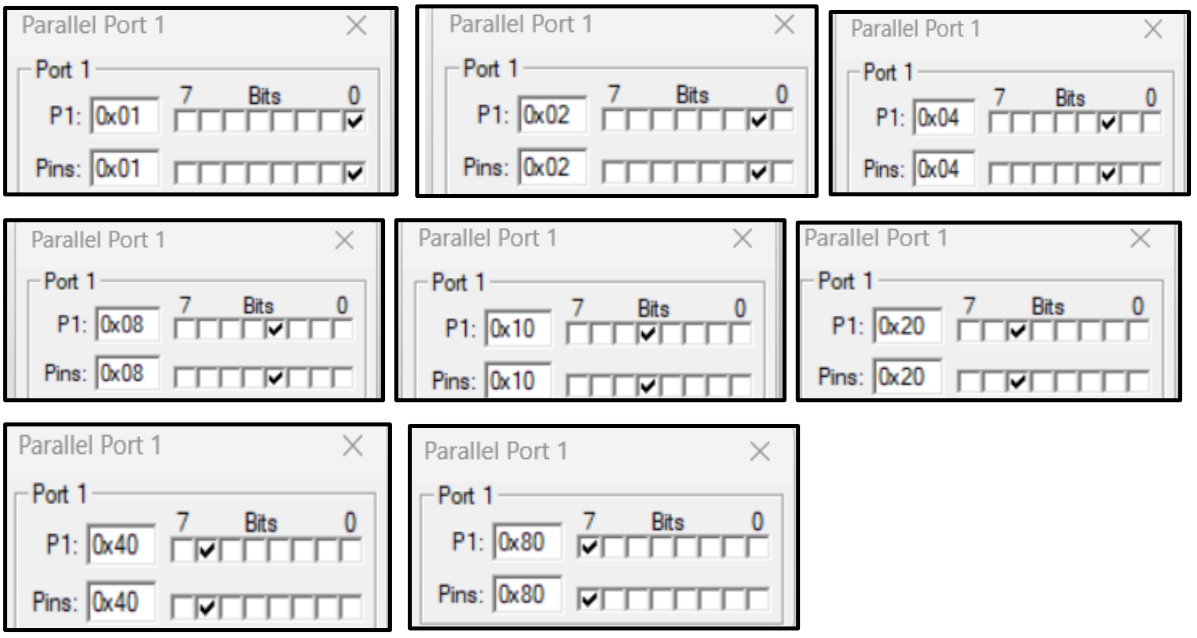
```
#include<reg51.h>
void delay(int);
void main()
{
        P1=0x01;
        while(1)
        {
                if(P1==0x00){
                        P1=0x01;
                }
                delay(500);
                P1=P1<<1;
        }
}
```

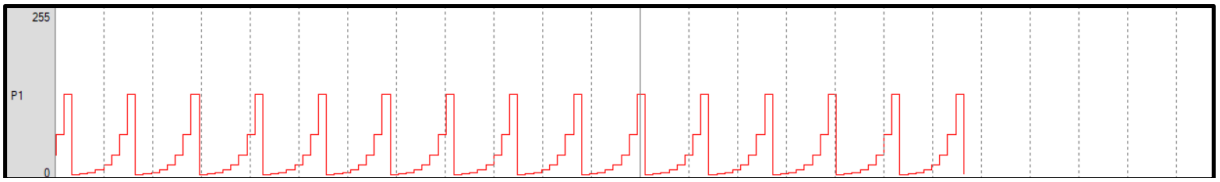
```
void delay(int t){
        unsigned int i,j;
        for(i=0;i<t;i++){
                for(j=0;j<t;j++){
                        ;
                }
        }
}
```

**Algorithm**:

1. Initialize port P1 with the value 0x01 to turn on the leftmost LED.
2. Enter an infinite loop.
3. Check if all LEDs are turned off (P1 == 0x00).
4. If all LEDs are off, reset the pattern to start from the leftmost LED (P1 = 0x01).
5. Call the delay function with a parameter of 500.
6. Left shift the value of port P1 to shift the LED pattern one position to the left.
7. Repeat steps 3-6 indefinitely.

**Result**:




**Conclusion:**
The code shifts the blinking LED connected to port P1 from right to left, ensuring the previous LED is turned off before lighting up the next one, with a delay of 500 milliseconds.

**<u>Question-5</u>:** Shift the blinking of the LED connected to port p1 from left to right by keeping the previous LED on.

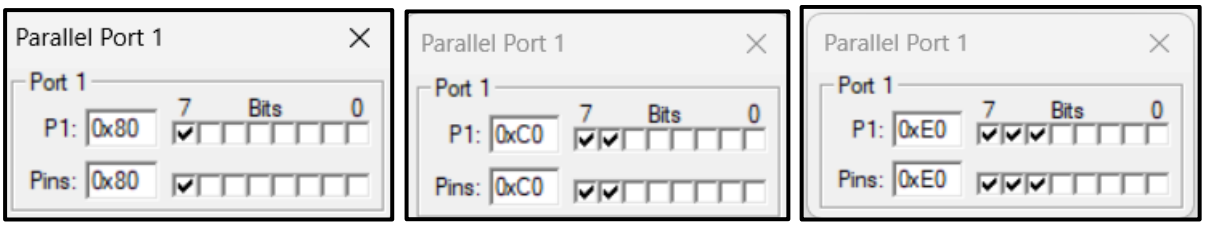
**Code**:

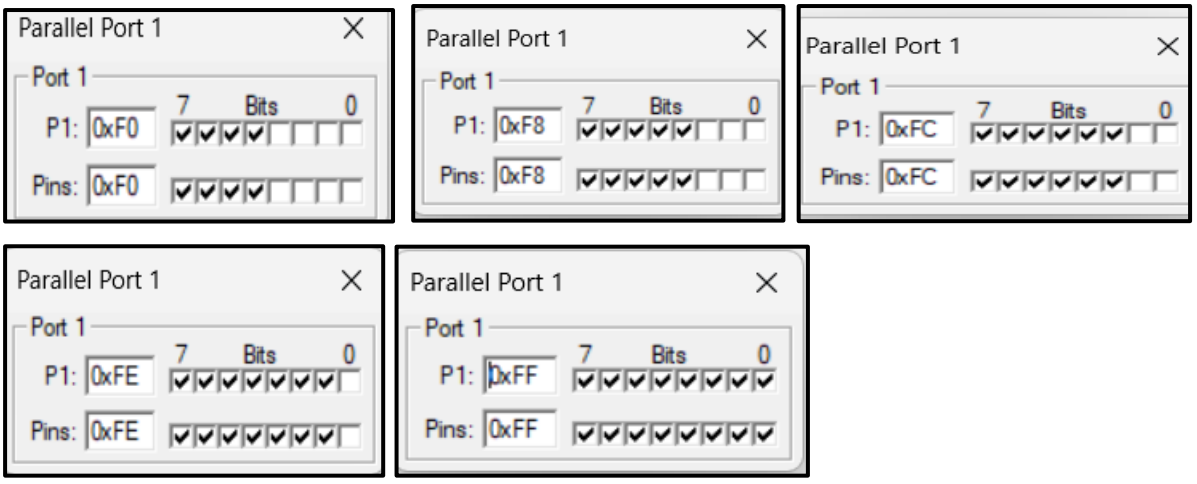
```
#include<reg51.h>
void delay(int);
unsigned int x;
void main()
{
	P1=0x80;
	x=P1;
	while(1)
	{
		delay(500);
		if(P1==0xff){
			P1=0x80;
		}
		else{
			P1=x|P1>>1;
		}
		x=P1;
	}
}
void delay(int t){
	unsigned int i,j;
	for(i=0;i<t;i++){
		for(j=0;j<t;j++){
			;
		}
	}
}
```

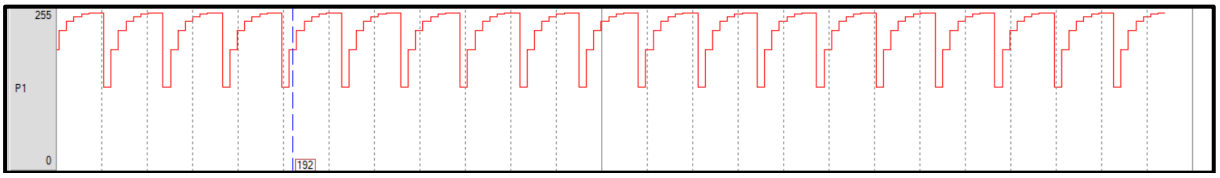
**Algorithm**:

1. Initialize port P1 with the value 0x80 to turn on the rightmost LED.
2. Initialize variable x with the value of port P1.
3. Enter an infinite loop.
4. Call the delay function with a parameter of 500.
5. Check if all LEDs are turned on (P1 == 0xFF).
6. If all LEDs are on, reset the pattern to start from the rightmost LED (P1 = 0x80).
7. Otherwise, shift the LED pattern one position to the right while keeping the previous LED on (P1 = x | P1 >> 1).
8. Update variable x with the current value of port P1.
9. Repeat steps 4-8 indefinitely.

**Result**:






**Conclusion:**
The code shifts the blinking LED connected to port P1 from left to right, ensuring the previous LED remains on, with a delay of 500 milliseconds, continuously looping.

**Question-6:** Shift the blinking of the LED connected to port P1 from right to left by keeping the previous LED on.

**Code**:

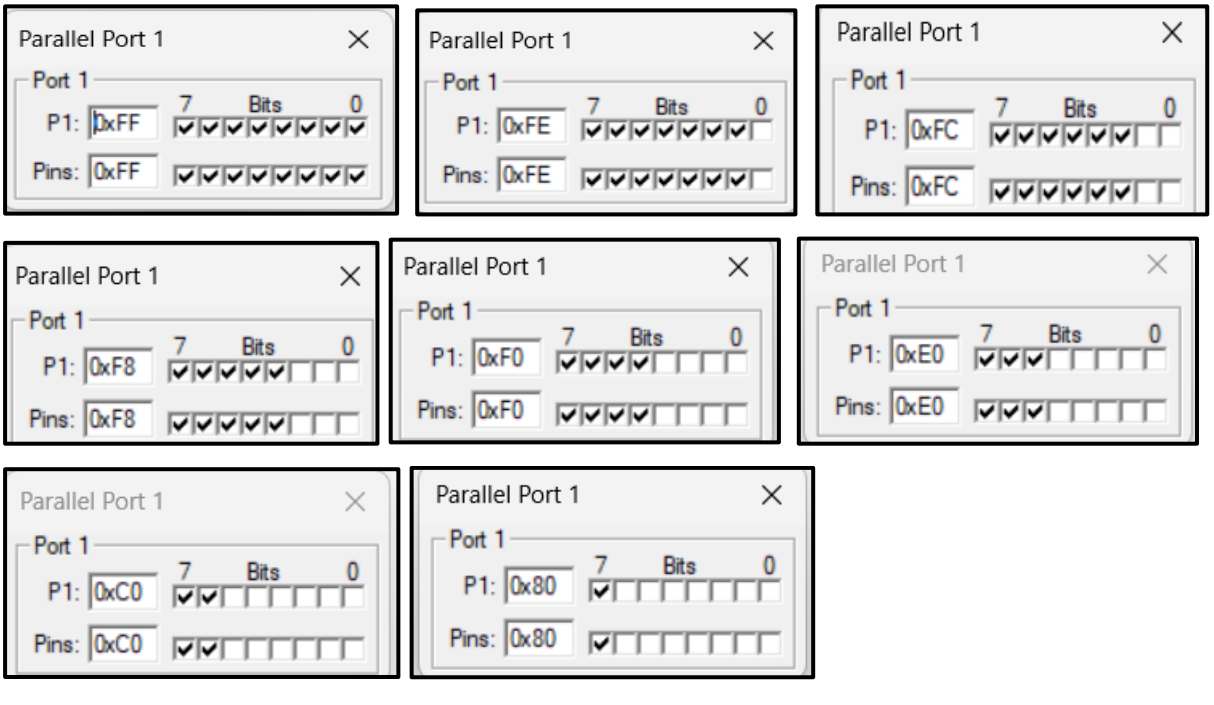
```
#include<reg51.h>
void delay(int);
unsigned int x;
void main()
{
        P1=0x01;
        x=P1;
        while(1)
        {
                delay(500);
                if(P1==0xff){
                        P1=0x01;
                }
                else{
                        P1=x|P1<<1;
                }
                x=P1;
        }
}
void delay(int t){
        unsigned int i,j;
        for(i=0;i<t;i++){
```

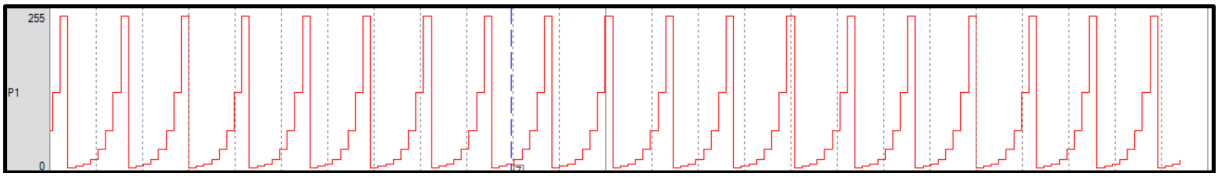
```
                for(j=0;j<t;j++){
                        ;
                }
        }
}
```

**Algorithm**:

1. Initialize port P1 with the value 0x01 to turn on the leftmost LED.
2. Initialize variable x with the value of port P1.
3. Enter an infinite loop.
4. Call the delay function with a parameter of 500.
5. Check if all LEDs are turned on (P1 == 0xFF).
6. If all LEDs are on, reset the pattern to start from the leftmost LED (P1 = 0x01).
7. Otherwise, shift the LED pattern one position to the left while keeping the previous LED on (P1 = x | P1 << 1).
8. Update variable x with the current value of port P1.
9. Repeat steps 4-8 indefinitely.

**Result**:




**Conclusion:**

The code shifts the blinking LED connected to port P1 from right to left, ensuring the previous LED remains on, with a delay of 500 milliseconds, continuously looping.