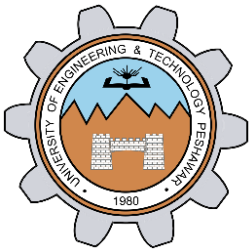
MICROPROCESSOR BASED SYSTEM DESIGN

TASK 6



Spring 2021

CSE307 MBSD

Submitted by: **Shah Raza**

Registration No. : **18PWCSE1658**

Class Section: **B**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Student Signature: _____

Submitted to:

Dr. Bilal Habib

Friday, June 19, 2021

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Task:

A. Generate a Knight rider pattern on P1 as shown below in figure 1.
B. By default, the pattern shifts left by l position after 1 sec.
C. When the pattern reaches the end point, it restarts as shown in figure 2 and 3.
D. Only one LED is ON during any time.
E. LED is ON only for 1 sec.
F. When a user presses a button at P3.2 then pattern changes the direction from left-moving to right-moving.
G. The hexadecimal equivalent of pattern is displayed on two seven segments connected to P2.
H. Use timers to generate the required 1 sec delay.
I. Program only in C
J. Whenever a user press P3.2, the pattern toggles from left moving to the right moving knight rider.
K. Use External interrupt (INT0) to detect button press.
L. Timer should have higher priority.

Problem Analysis:

We need a delay of 1 sec = 1000 msec
65.535ms is the max delay we can create, so to attain a delay of 1000ms, we should create a delay of 50ms and run it 20 times.

TMOD:

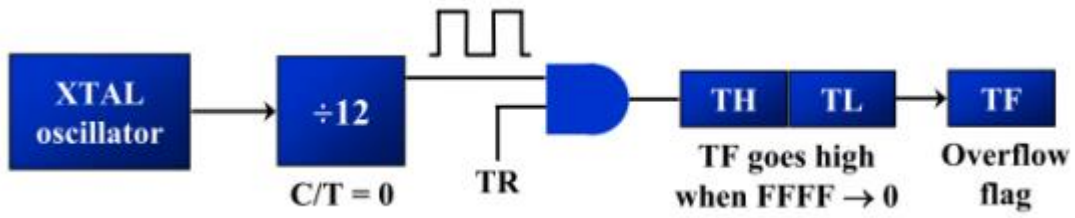
	Timer1				Timer0				
Gate	C/T	M1	M0		Gate	C/T	M1	M0	
0	0	0	0	0	0	0	1		(Hex= 1)

Timer 0: Used as Timer with mode 1

IE:

EA	--	--	ES	ET1	EX1	ET0	EX0	
1	0	0	0	0	0	1	1	(Hex = 83)

Timer 0 (Used as timer in Mode 1):



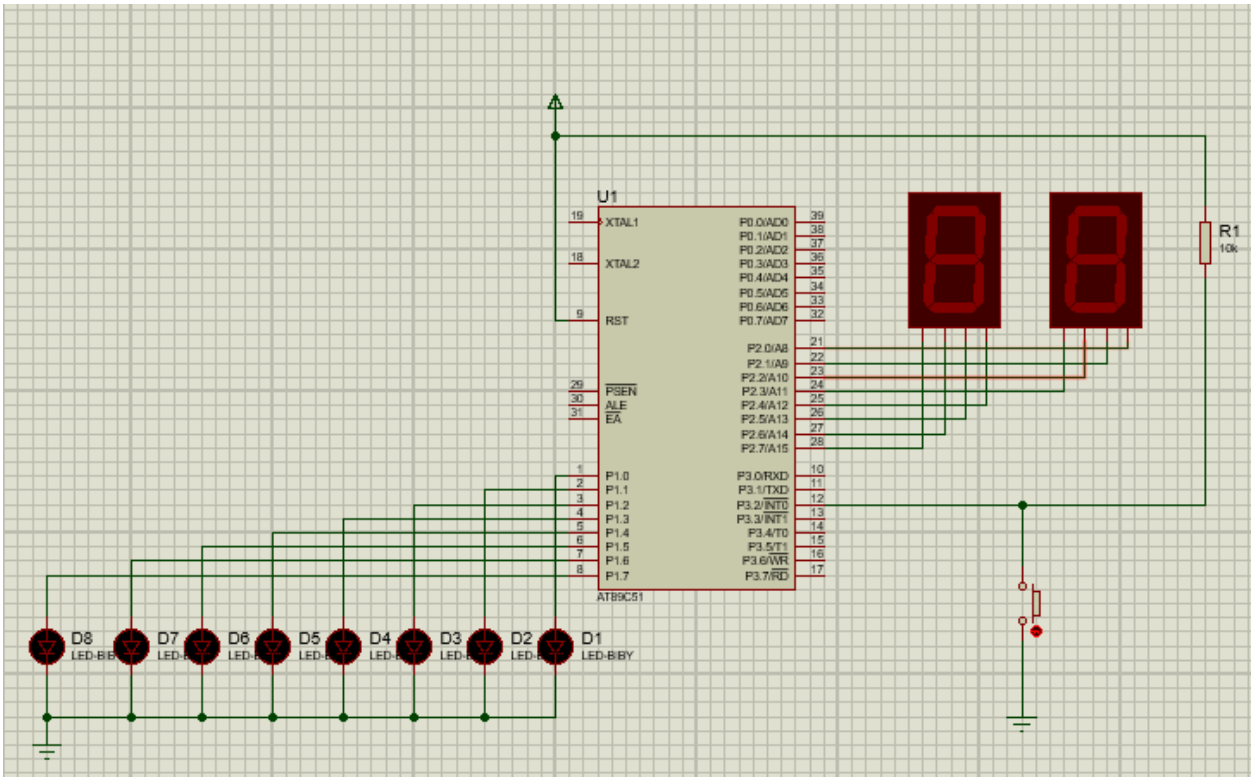
Code:

```
#include <reg51.h>
#include <stdio.h>

unsigned int index = 0;
int x = 0;

int Array[8] = {0x1,0x2,0x4,0x8,0x10,0x20,0x40,0x80};

int forward = 1; //if 1 then move to left else move to right

void Start_Timer()
{
    TR0 = 1;   //start timer0
}

void ExternInt0() interrupt 0
{
    forward = !(forward);       //Toggle forward
}

void timer0() interrupt 1
{
  x++;
  if(x==20)     //if 1 sec delay completed
  {
    x=0;
    if(forward)
          index++;
    else
          index--;
    index%=8;
  }
   //Create a delay of 50 msec
   TH0 = 0x3C;
   TL0 = 0xAF;

}

void Init()
{
    TMOD =  0x1;  //Use timer 0 in mode 1
    //Create a delay of 50 msec
    TH0 = 0x3C;
    TL0 = 0xAF;
```

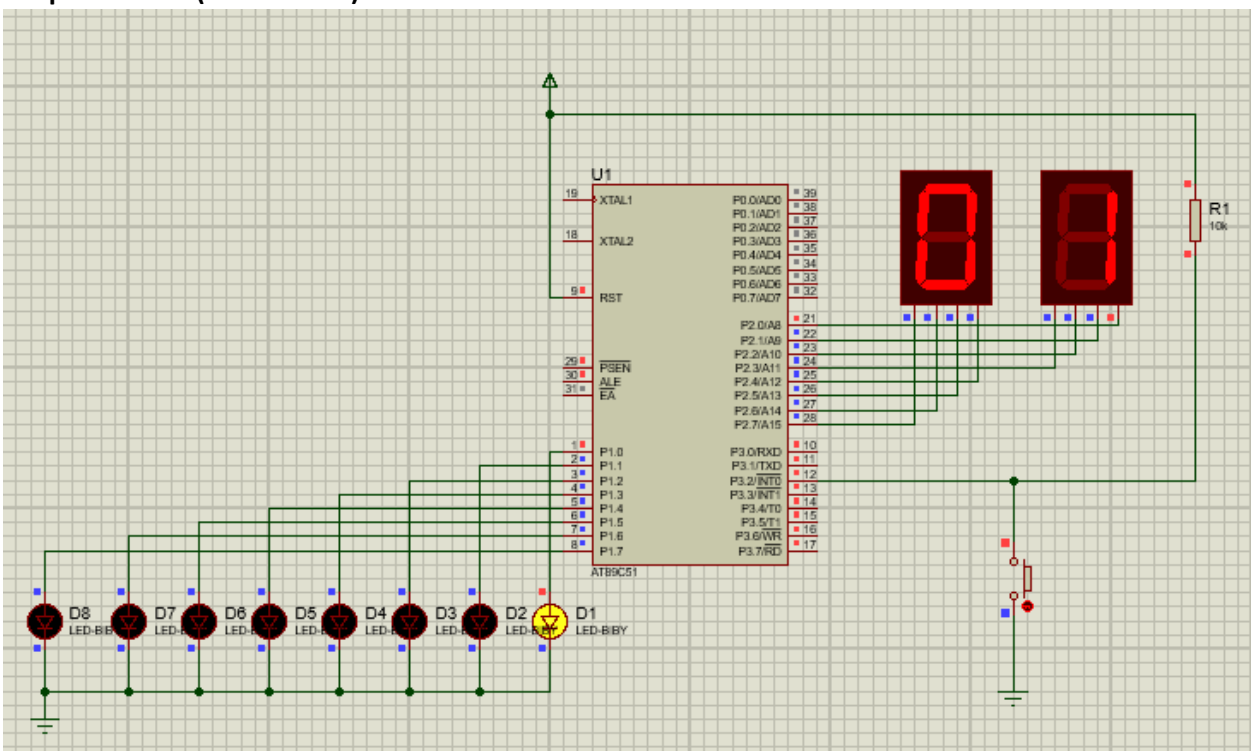
```
    IT0 = 1;    //Make the external interrupt 0 edge triggered


    EA = 1;     //Enable global interrupt
    EX0 = 1;    //Enable External interrupt 0
    ET0 = 1;    //Enable Timer0 interrupt

    PT0 = 1;    //Give higher priority to timer0

}

void main(void)
 {
   Init();
   Start_Timer();
   while (1)
   {
                P1 = Array[index];
                P2 = Array[index];
    }
 }
```

Output / Graphs / Plots / Results:

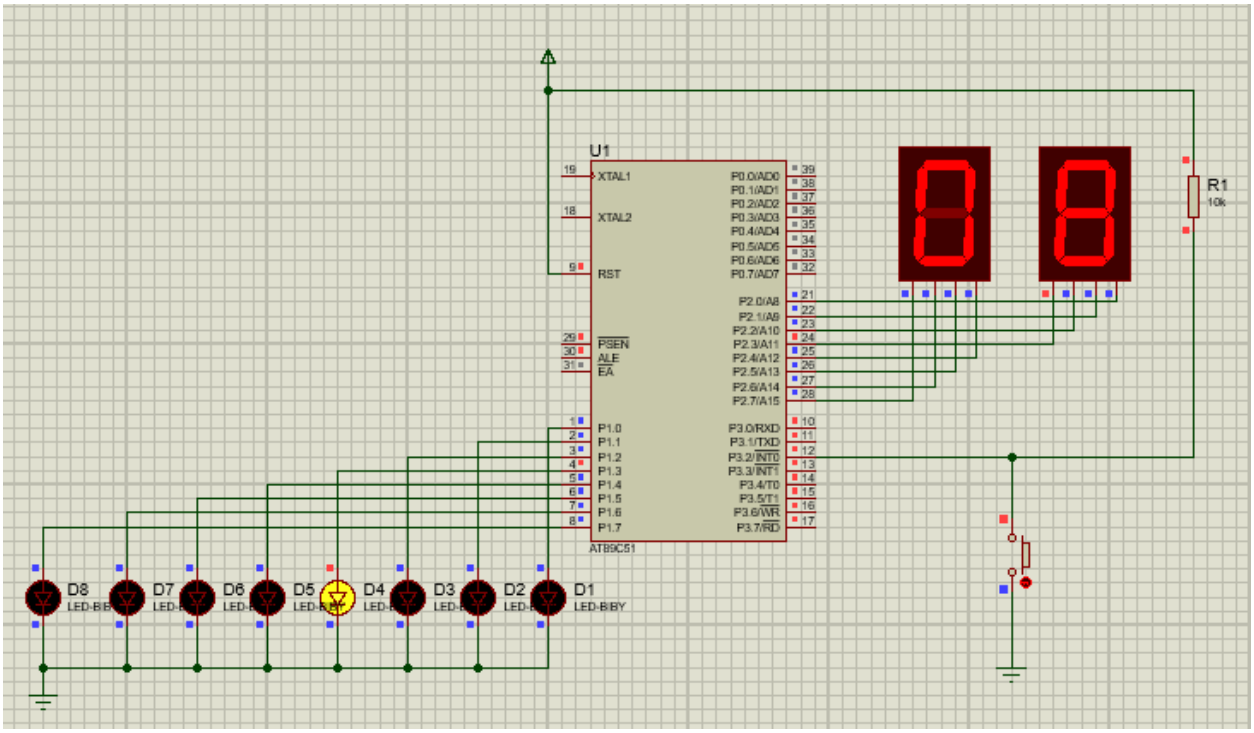
Schematic:



Output at 0 sec (Forward = 1):



Output at 3 sec (Forward = 1):



Output at 4 sec (Forward = 0):

Now the button is pressed and forward becomes 0 meaning that the order will be in reverse now.