

Traffic Control System For a 4 Way Junction Road Using an 8051 Microcontroller

Introduction:

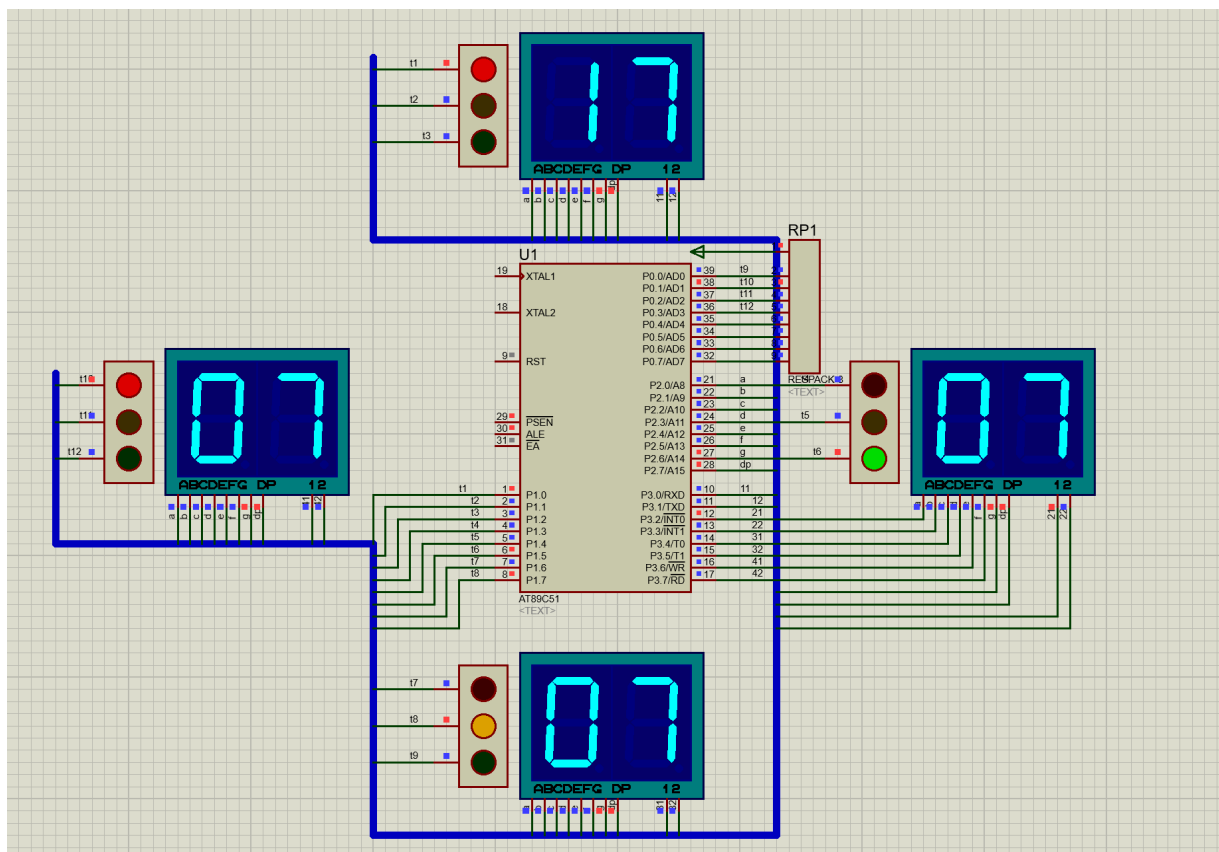
The 8051 Microcontroller Based Traffic Control System is an intelligent traffic control system designed to efficiently manage traffic flow at a 4-way junction. By leveraging the capabilities of the 8051 microcontroller, this system coordinates the timing and sequencing of traffic signals, ensuring smooth and safe traffic movement at the junction.

The system operates in a continuous cycle where, at any moment, one signal is green for a specified duration, the next signal in a clockwise direction is yellow, and the remaining signals are red. This setup ensures that each direction receives an equal amount of green light time, with transitions occurring uniformly across all pathways.

Components Used:

1. 1x 8051 Microcontroller
2. 4x 7 Segment LCD Displays
3. 4x Traffic Lights LED module
4. 1x Respack-8

Simulation:



Code:

```
ORG 00H
LJMP MAIN
ORG 300H
TBL:  DB 0C0H,0F9H,0A4H,0B0H,99H,92H,82H,0F8H,80H,90H    ;7seg data for comm.
anode type
ORG 30H
```

```
MAIN:  MOV P2,#00H
      MOV P3,#00H
      ACALL FRONT
      MOV DPTR,#TBL
      CLR A
      MOV 40H,#10
      MOV 43H,#10
      MOV 46H,#20
      MOV 49H,#20
      MOV R0,#35
      MOV R6,#30
      MOV R7,#40
```

```
X1:   MOV A,40H
      MOV B,#10
      DIV AB
      MOV 41H,A
      MOV 42H,B
```

```
A1:   SETB P3.0
      CLR P3.1
      MOV A,41H
      MOVC A,@A+DPTR
      MOV P2,A
      ACALL DELAY
      MOV P3,#00H
      SETB P3.1
      CLR P3.0
      MOV A,42H
      MOVC A,@A+DPTR
      MOV P2,A
      ACALL DELAY
      MOV P3,#00H
```

```
SJMP X3
X2:  SJMP X1
X3:  MOV A,43H
      MOV B,#10
      DIV AB
      MOV 44H,A
      MOV 45H,B
      SETB P3.2
      CLR P3.3
      MOV A,44H
      MOVC A,@A+DPTR
      MOV P2,A
      ACALL DELAY
      MOV P3,#00H
      SETB P3.3
      CLR P3.2
      MOV A,45H
      MOVC A,@A+DPTR
      MOV P2,A
      ACALL DELAY
      MOV P3,#00H
```

```
MOV A,46H
MOV B,#10
DIV AB
MOV 47H,A
MOV 48H,B
SETB P3.4
CLR P3.5
MOV A,47H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H
SETB P3.5
CLR P3.4
MOV A,48H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H
```

```
MOV A,49H
MOV B,#10
DIV AB
MOV 50H,A
MOV 51H,B
SETB P3.6
CLR P3.7
MOV A,50H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H
SETB P3.7
CLR P3.6
MOV A,51H
MOVC A,@A+DPTR
MOV P2,A
ACALL DELAY
MOV P3,#00H
```

```
DJNZ R0,X2
MOV R0,#35
```

```
DJNZ 40H,Q1
MOV 40H,#20
```

```
Q1:  DJNZ 43H,Q2
      MOV 43H,#10
      ACALL RIGHT
```

```
Q2:  DJNZ 46H,Q3
      MOV 43H,#20
      MOV 46H,#10
```

```
Q3:  DJNZ 49H,Q4
      MOV 49H,#10
      ACALL BACK
```

```
Q4:  DJNZ R6,X4
      ACALL LEFT
      MOV 40H,#10
```

```
MOV 43H,#10
MOV 46H,#30
```

```
X4:  DJNZ R7,L1
      LJMP MAIN
L1:   LJMP X1
```

```
DELAY: MOV R4,#5
H2:    MOV R5,#0FFH
H1:    DJNZ R5,H1
      DJNZ R4,H2
      RET
```

```
FRONT: MOV P1,#54H
      MOV P0,#02H
      RET
```

```
RIGHT: MOV P1,#0A1H
      MOV P0,#02H
      RET
```

```
BACK:  MOV P1,#09H
      MOV P0,#05H
      RET
```

```
LEFT:  MOV P1,#4AH
      MOV P0,#08H
      RET
```