2 Way Traffic

HAGAR MOHAMED

SHOUBRA FACULTY OF ENGINEERING | Department of Electronics and Computer science

INTRO

This report presents the design and implementation of a two-way traffic light controller using a PIC16F877A microcontroller. The system aims to optimize traffic flow through precise timing and sequence control. Hardware and software components, as well as performance evaluation, are detailed.\

HARDWARE COMPONENTS
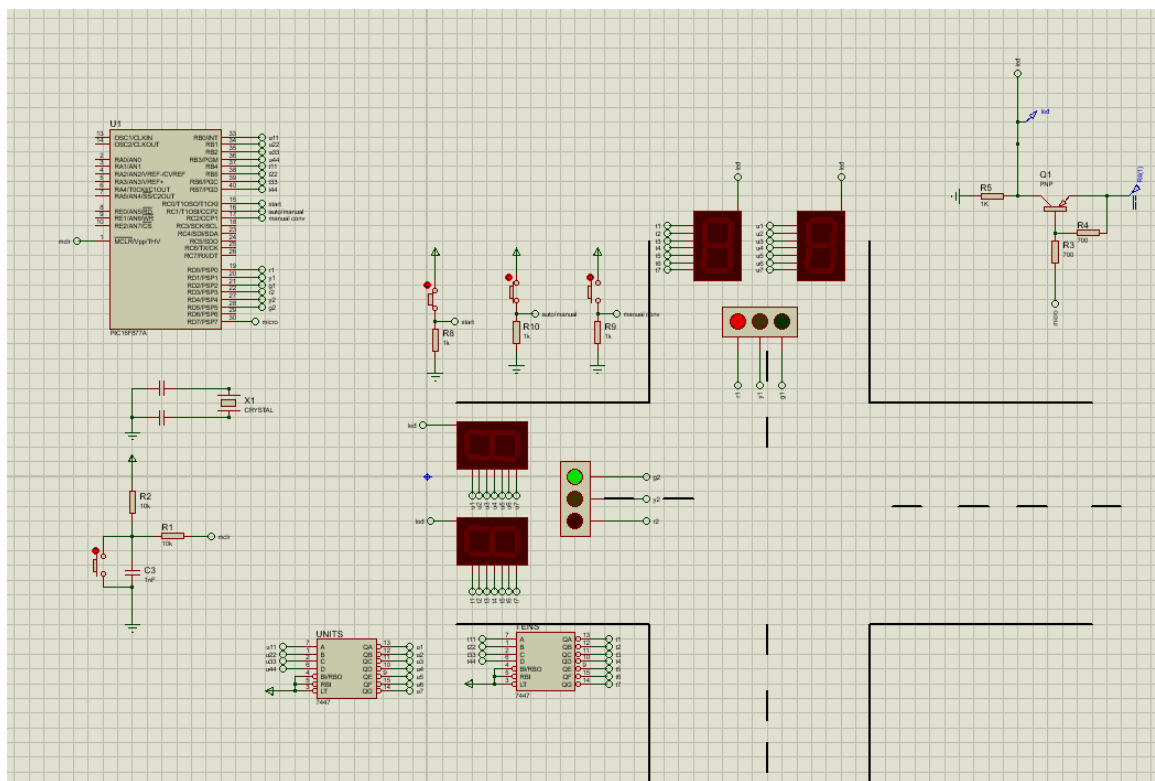
SOFTWARE

```c
void main(void) {
    char num15=21;
    char num20=32;
    char num3=3;
    char M=0;
    TRISC=0xff;
    TRISB=0x0;
    TRISD=0X0;
    RD0=0;RD1=0;RD2=0;RD3=0;RD4=0;RD5=0;RD6=0;RD7=1;


    while(RC0==1 && RC1==0){ //RC0==STTART
        RD7=0; //LED DISPLAY IS ON

        for(char i=0 ;i<=15;i++) { RD3=1;
                              if(i<13){RD2=1;}else RD2=0;
                              if(i>=13){RD0=1;}else RD0=0;
                              if(RC1==1){RD7=1;break;} ;
                              PORTB=num15;  num15--;  __delay_ms(200); if(num15==15){  num15-=6;}; }  RD3=0;    num15=21;

        for(char j=0 ; j<=3;j++) { RD4=1;
                              if(RC1==1){RD7=1;break;};
                              PORTB=num3;    num3--;    __delay_ms(200);}  RD4=0;           num3=3;


        for(char k=0;k <=20;k++) { RD5=1;
                              if(k>17){RD0=0;RD1=1;};
                              if(RC1==1){RD7=1;break;};
                              PORTB=num20;    __delay_ms(200); num20--; if(num20==31 || num20==15){ num20-=6;}  }RD5=0;RD1=0; num20=32;
    }



    while(RC1==1){
             if(RC2==1){M++;}

        if (M>1)M=0;
             if (M==0){
                    RD0=1; RD3=0;
                    RD2=0; RD5=1;
                    RD1=0; RD4=0;__delay_ms(80);
                       }


        if (M==1){
                    RD0=0; RD3=1;
                    RD2=1; RD5=0;
                    RD1=0; RD4=0;
                    __delay_ms(80);
                  }
```

FLOWCHART

void main(void)

char num15=21

char num20=32

char num3=3

char M=0

TRISC=0xff

TRISB=0x0

TRISD=0X0

RD0=0

RD1=0

RD2=0

RD3=0

RD4=0

RD5=0

RD6=0

RD7=1

cha

RC0==STTART

RC0==1 && RC1==0

True

RD7=0

char i=0

i<=15

True

RD3=1

i<13

False — RD2=0

True — RD2=1

i>=13

False — RD0=0

True — RD0=1

RC1==1

True — RD7=1

False — PORTB=num15

RD3=0

num15=21

cha

num15--

__delay_ms(200)

num15==15

True — num15-=6

False

i++

False

Flowchart (left diagram)

```
char j=0
   ↓
j<=3 ──False──────────────────────────┐
   │ True                             │
RD4=1                                 │
   ↓                                  │
RC1==1 ──True──> RD7=1                │
   │ False              ↓             │
PORTB=num3          RD4=0             │
   ↓                    ↓             │
num3--              num3=3            │
   ↓                    ↓             │
__delay_ms(200)     char k=0         │
   │                    ↓             │
   │              k <=20 ──False──┐   │
   │                 │ True       │   │
   │              RD5=1           │   │
   │                 ↓            │   │
   │              k>17 ──True──> RD0=0
   │                 │ False       ↓
   │              RC1==1 ───────> RD1=1
   │                 │  │
   │            True │  │ False
   │              RD7=1  PORTB=num20
   │                 ↓      ↓
   │              RD5=0  __delay_ms(200)
   │                 ↓      ↓
   │              RD1=0  num20--
   │                 ↓      ↓
   │              num20=32  num20==31 || num20==15
   │                 ↓       │ True        │ False
   │               cha    num20-=6       k++
   │
   └──> j++
```

num20-=6 ──True──> (from num20==31 || num20==15)
k++ ──False

cha ──> (red oval) ──> (red diamond end)

Flowchart (right diagram)

```
(top junction)
   ↓
RC1==1 ──True──> RC2==1 ──True──> M++
   │ False          │ False        │
  cha               └──────────> M>1
   ↓                              │ True      │ False
(red diamond)     (red oval)    M=0          │
                                  ↓           │
                              M==0 <──────────┘
                               │ True
                              RD0=1
                               ↓
                              RD3=0
                               ↓
                              RD2=0
                               ↓
                              RD5=1
                               ↓
                              RD1=0
                               ↓
                              RD4=0
                               ↓
                              __delay_ms(80)
                               ↓
                              M==1 ──True──> RD0=0
                               │ False         ↓
                               │             RD3=1
                               │               ↓
                               │             RD2=1
                               │               ↓
                               │             RD5=0
                               │               ↓
                               │             RD1=0
                               │               ↓
                               │             RD4=0
                               │               ↓
                               │            __delay_ms(80)
```