EMBEDDED SYSTEMS

TRAFFIC LIGHT

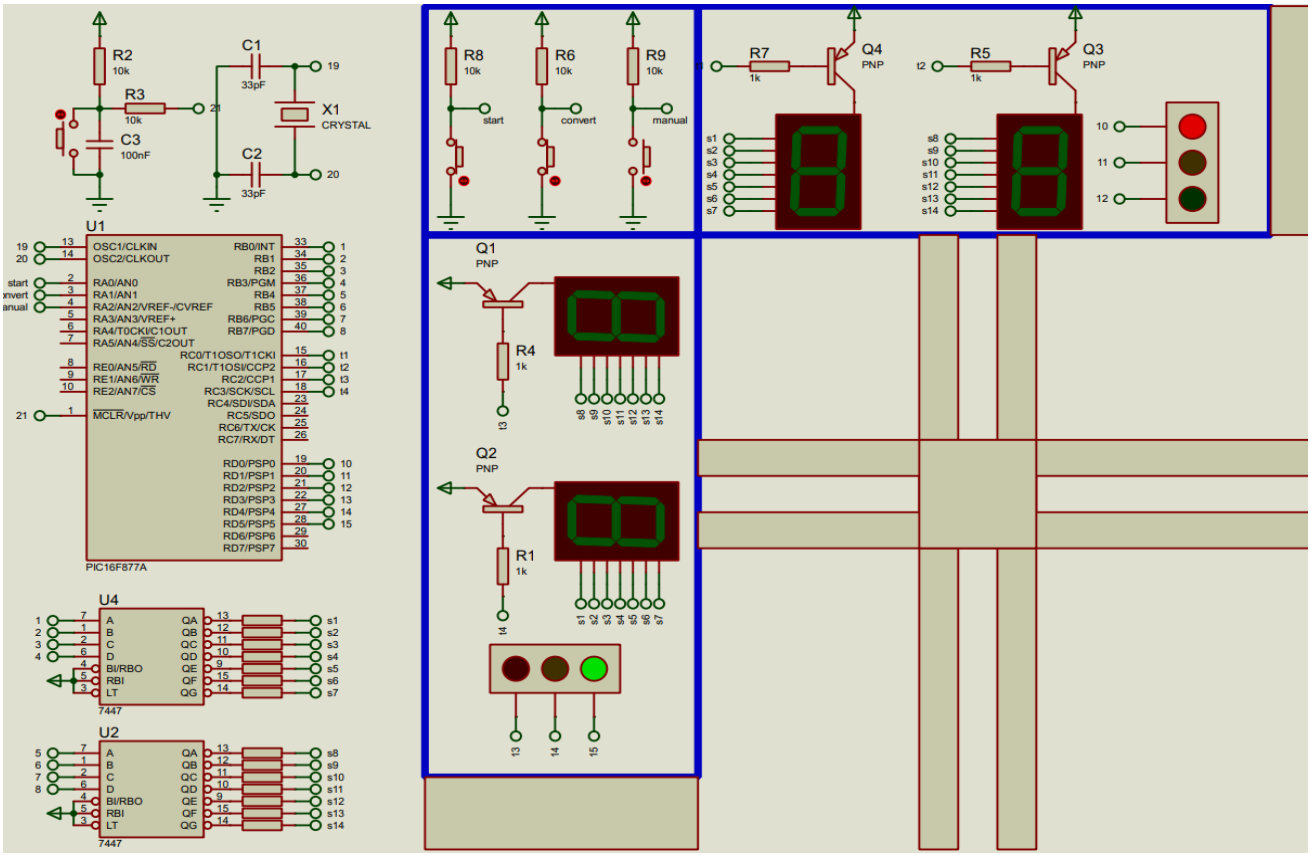
Presented By

KEROLOS HANY SALAMA

SEC:3 B.N:41

Circuit:

1. **Objective:** Develop a traffic light control system using the PIC16F877A microcontroller, designed to operate in both automatic and manual modes. The system will manage traffic lights at an intersection with two streets: West and South.
2. **Modes of Operation:**
    - Automatic Mode: The traffic lights switch according to predefined timing:
    - West Street: 15 seconds Red, 3 seconds Yellow, 20 seconds Green.
    - South Street: 23 seconds Red, 3 seconds Yellow, 12 seconds Green
    - Manual Mode: Allows manual switching between streets and toggling the traffic lights if needed.
3. **Control Mechanism:**
    - **Three switches are used:** 1st switch for start, 2nd for toggling between Manual and Automatic modes, 3rd for selecting between the two streets in Manual mode.
4. **Timing in Manual Mode:** Even in Manual mode, a 3-second Yellow light period is required.
5. **Display:** Use 7-segment displays at each corner of the intersection to show the remaining time. BJTs and a 7447 IC are recommended to manage these displays and reduce the number of microcontroller pins used.



Code:

```
signed char i = 0, flag = 0, count = 0;
void counter(signed char i)
{
    portc = 0;
    portb = (i / 10) + (i % 10) * 16;
    delay_ms(500);
}
void toggle(void)
{
    if (porta.b2 == 0)
    {
        flag = ~flag;
        delay_ms(500);
    }
}
void west(void)
{
    portd.b0 = 1;
    for (i = 15;; i--)
    {
        if (porta.b1 == 0)
            break;
        if (i < 0)
        {
            portd.b4 = 0;
            break;
        }
        if (i <= 3)
        {
            portd.b5 = 0;
            portd.b4 = 1;
        }
        else
            portd.b5 = 1;
        counter(i);
    }
    portd.b0 = 0;     portd.b5 = 0; }
```

```
void south(void)
{
    portd.b3 = 1;
    for (i = 23; i >= 0; i--)
    {
        if (porta.b1 == 0)
            break;
        if (i < 0)
        {
            portd.b1 = 0;
            break;
        }
        if (i <= 3)
        {
            portd.b1 = 1;
            portd.b2 = 0;
        }
        else
            portd.b2 = 1;
        counter(i);
    }
    portd.b2 = 0;
    portd.b3 = 0;
}
void automatic(void)
{
    while (porta.b1 == 1)
    {
        portb = 0x00;
        portd = 0x00;
        west();
        south();
    }
}
```

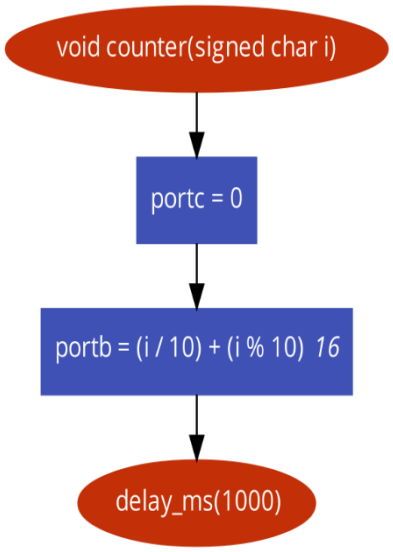
```
void manual(void)
{
    while (porta.b1 == 0)
    {
        portd = 0x00;
        portb = 0x00;
        portc = 15;
        while (flag == 0)
        {
            portd.b2 = 0;
            portd.b3 = 0;
            portd.b5 = 1;
            for (i = 0; i < 3; i++)
            {
                portd.b1 = 1;
                delay_ms(500);
                portd.b1 = 0;
                delay_ms(500);

                toggle();
                if (flag == 1)
                    break;
                if (porta.b1)
                    break;
            }
            while (1)
            {
                portd.b0 = 1;

                toggle();
                if (flag)
                    break;
                if (porta.b1)
                    break;
            }

            if (porta.b1)
                break;
        }
```

```
        while (flag)
        {
            portd.b0 = 0;
            portd.b5 = 0;

            portd.b2 = 1;
            for (i = 0; i < 3; i++)
            {
                portd.b4 = 1;
                delay_ms(500);
                portd.b4 = 0;
                delay_ms(500);

                toggle();
                if (flag == 0)
                    break;
                if (porta.b1)
                    break;
            }
            while (1)
            {
                portd.b3 = 1;

                toggle();
                if (flag == 0)
                    break;
                if (porta.b1)
                    break;
            }

            if (porta.b1)
                break;
        }
    }
}
```

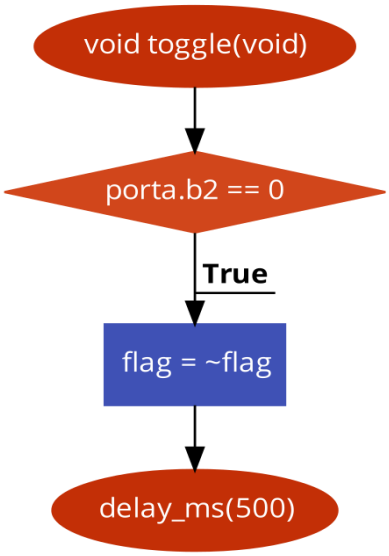
```
/*
Code Overview:
      • automatic() Function: Handles traffic light timing and
      transitions in Automatic mode.
      • manual() Function: Controls the traffic lights in Manual mode,
      allowing manual toggling of lights.
      • south() and west() functions: they control the leds in traffic
      light
      • counter() function: control  7-segment
      • Main Loop: Continuously checks mode and executes corresponding
      functions.
*/

void main()
{
    adcon1 = 0x07;
    trisb = 0x00;
    portb = 0x00;
    trisc = 0x00;
    portc = 15;
    trisd = 0x00;
    portd = 0x00;
    trisa = 7;
    while (porta.b0)
        ;
    while (1)
    {
        automatic();
        manual();
    }
}
```

FLOW CHART:

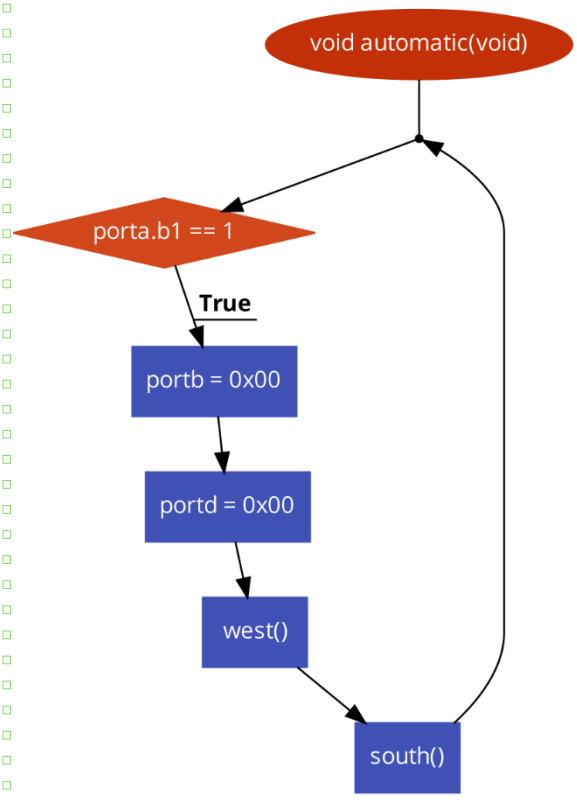
A) Counter function:



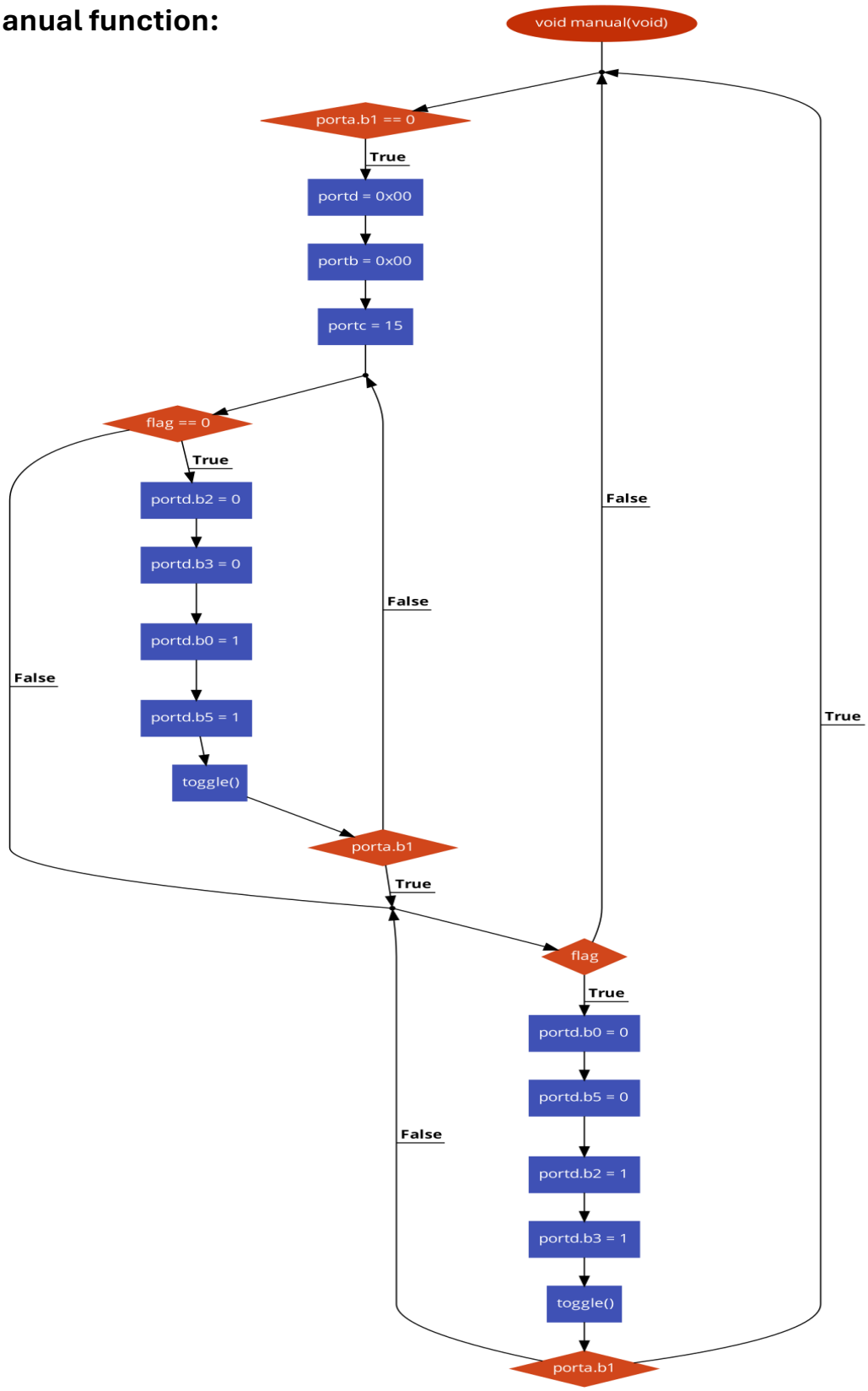
B) Toggle function:



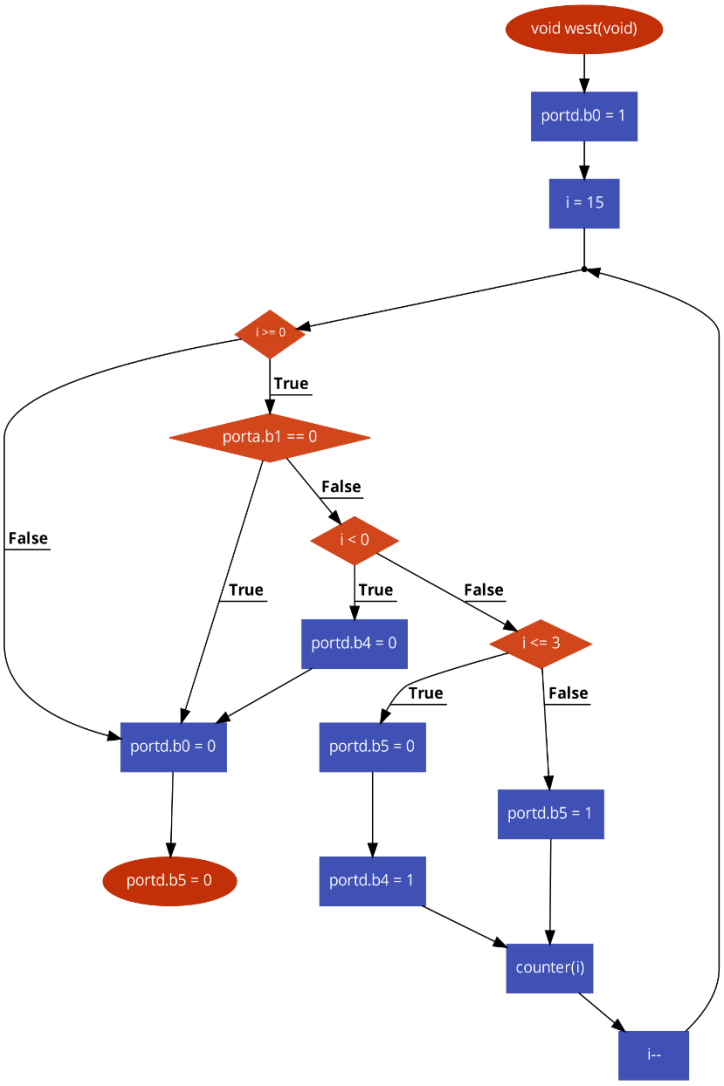
C) Automatic function:



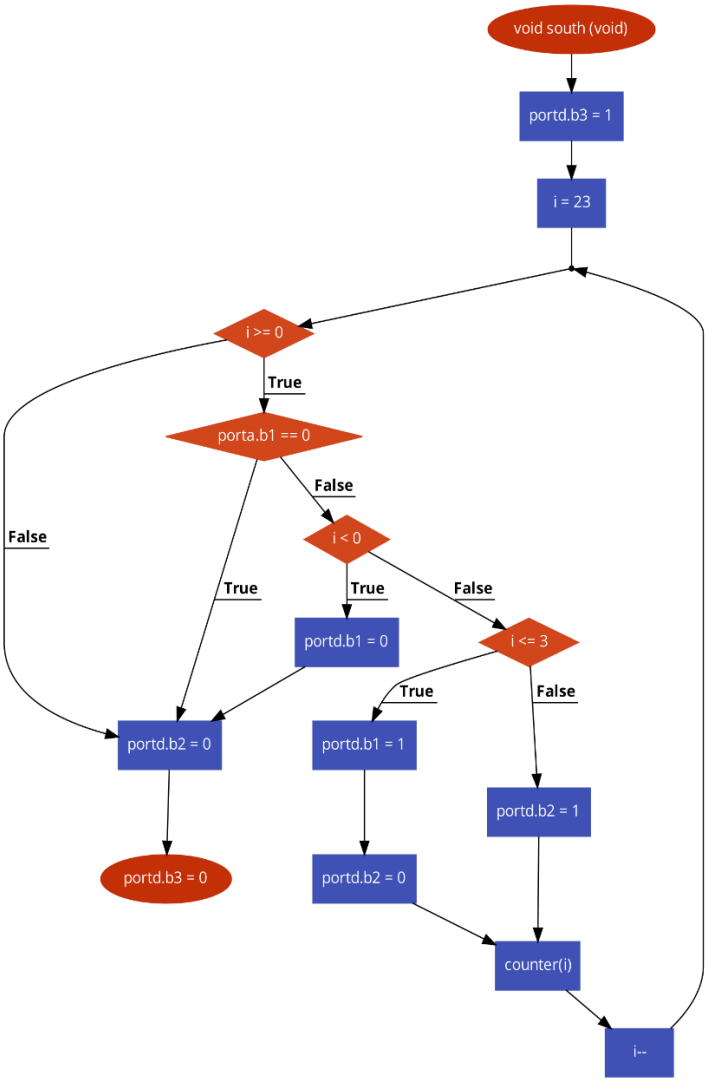
D) Manual function:



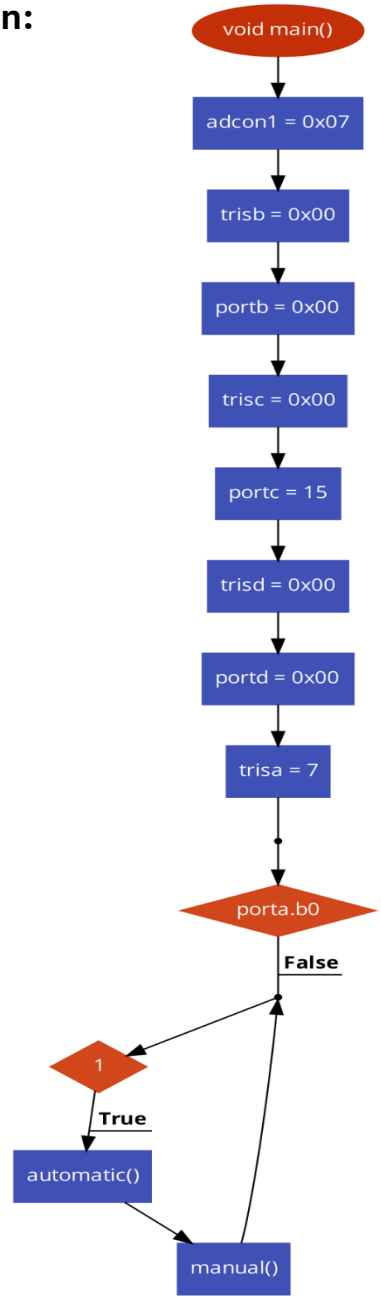
E) West function:



F) South function:



G) main function:



YouTube link: https://youtu.be/BWmRV7l0t_g?si=yr7zQFS-X2noEK6T

GitHub link: https://github.com/kerolosHani4/traffic-_light