DROWSY DRIVER DETECTION WITH 8051 MICROCONTROLLER

1. Introduction

Proteus Software:

Proteus is a comprehensive suite of electronic design automation (EDA) tools used for circuit simulation, PCB design, and microcontroller simulation. It is widely used by engineers, designers, and hobbyists for designing and testing circuits before implementing them in hardware. Developed by Lab centre Electronics, Proteus offers a range of features that make it a powerful tool for both education and professional projects.

Keil µvision:

Keil µVision, often referred to as Keil µVision or simply Keil, is an integrateddevelopment environment (IDE) specifically designed for embedded software development. Developed by Keil, a division of ARM Holdings, it is widely used for programming and debugging microcontrollers, especially those based on ARM Cortex-M, ARM7, ARM9, and other ARM architecture processors. The Keil µVision environment combines project management, code editing, compilation, and debugging in a single application, making it a powerful tool for embedded developers.

- **DROWSY DRIVER DETECTION :**

Drowsy Driver detection system designed using 8051 microcontroller,Eye blink sensors, 16*2Lcd , DC motor, buzzer and driver IC L293D. This application designed for driver who do mistake while driving vehicles even they are sleeply or drowsy. This eye blink sensor help and protects humans as well as vehicles.Whenever driver close eyes vehicle will stopped .This application source code developed using Embedded C Language in keil IDE and Simulated with proteus simulation tool.

8051 Microcontroller:

The 8051 microcontroller, developed by Intel in 1980, is a widely used 8-bit microcontroller renowned for its simplicity, versatility, and extensive support in embedded systems.

a. **CPU**: Handles arithmetic and logical operations, and controls the flow of instructions.
b. **Memory**:
    1. 4KB ROM for program storage.
    2. 128 bytes RAM for data storage and temporary variables.
c. **I/O Ports**: Four 8-bit bidirectional ports (P0 to P3) for interfacing with external devices.
d. **Timers/Counters**: Two 16-bit timers (Timer 0 and Timer 1) for timing operations and event counting.
e. **Serial Communication**: Built-in UART for serial communication with external devices.
f. **Interrupts**: Five interrupt sources for handling asynchronous events.
g. **Oscillator and Clock**: Typically uses a 12 MHz crystal oscillator to generate the clock signal.

2. Circuit Diagram:

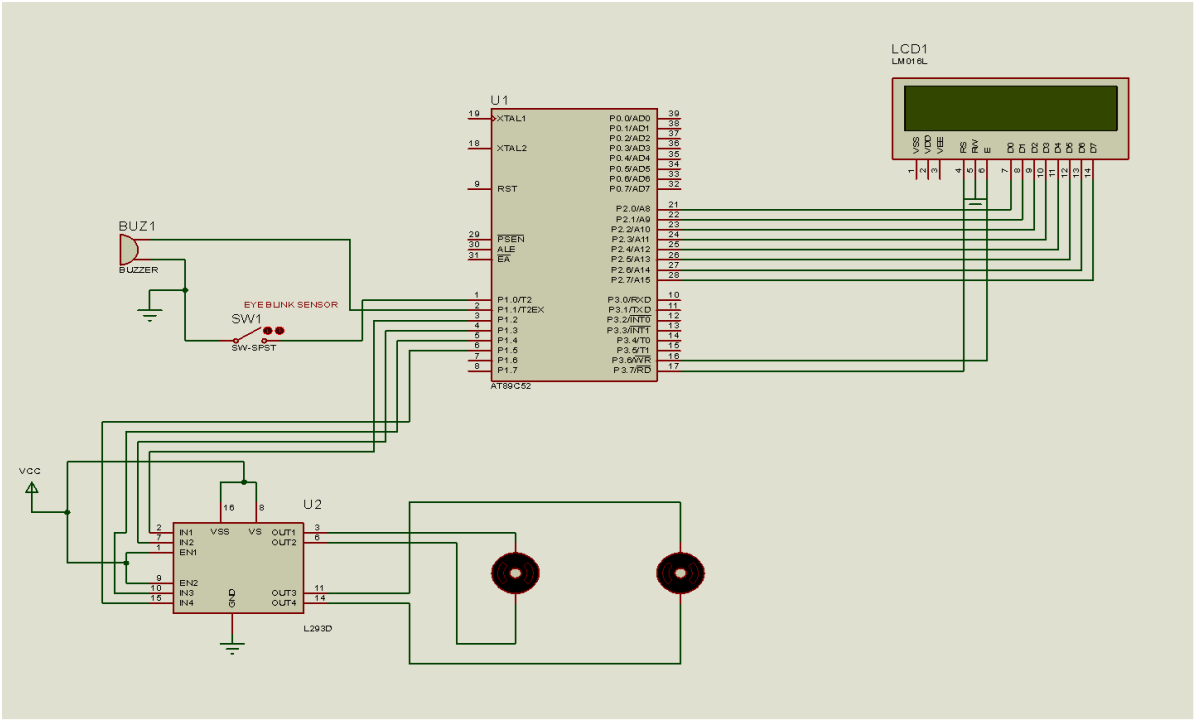


Figure 1: Circuit Diagram

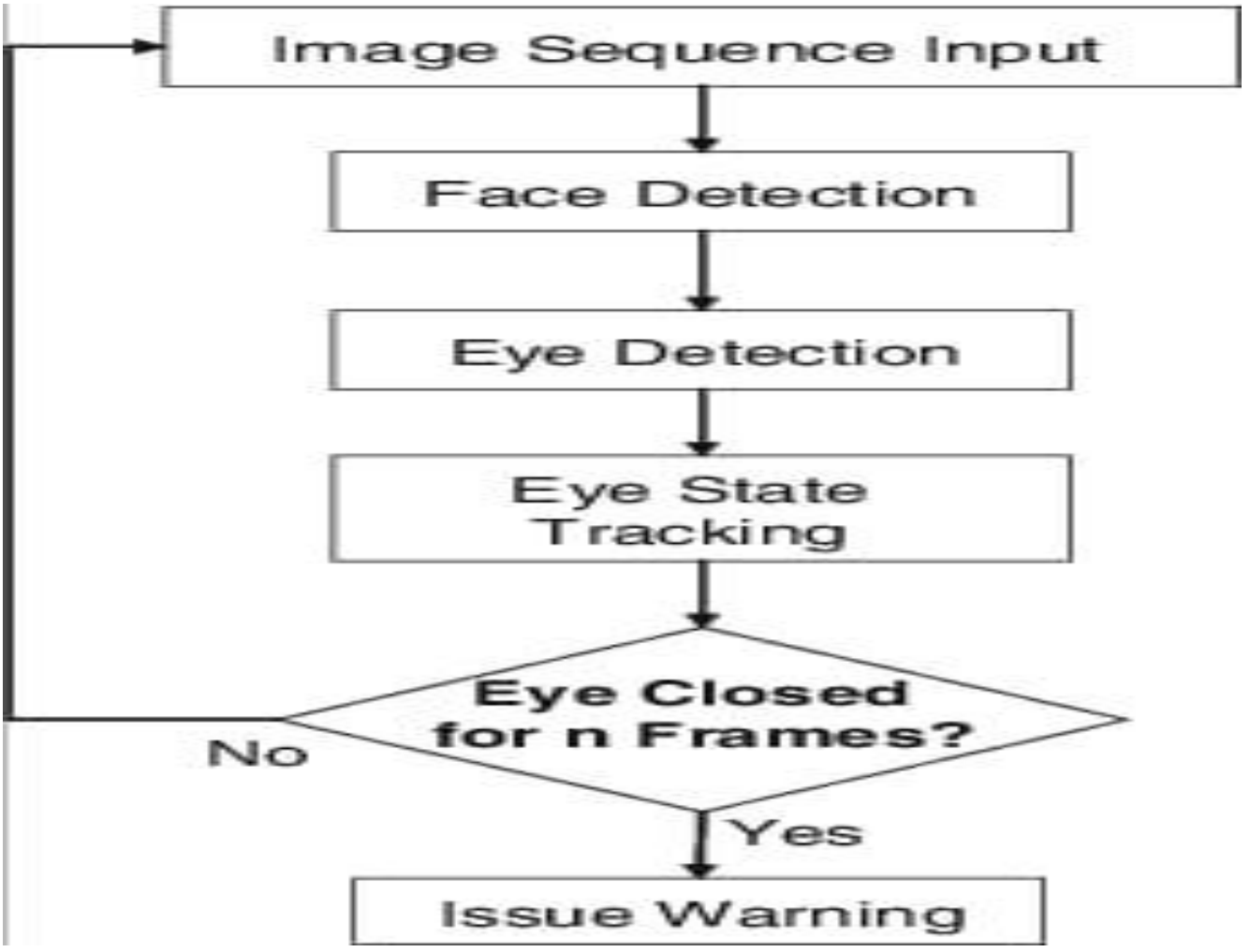
3. Explanation:

A Drowsy driver detection system is a safety technology designed to monitor a driver's alertness level and detect signs of fatigue or sleepiness while driving. It's goal is to to prevent accidents caused by a tired or inattentive driver by issuing warning or taking corrective actions.

A eye blinked sensor monitors the driver's face and eyes. Once the sensor is changing it's function to logic zero, then motors off and buzzer count high and we will get message on screen is driver is sleeping.

When the sensor logic is one, the motor is keep on runnimg and buzzer is off state text message on screen is driver is not sleeping.

4. Algorithm / Flowchart:



5. Program:

```
#include <reg51.h>
sbit rs =P3^7;      //**LCD REGISTER PIN AT PORT-3 SEVENTH PIN
sbit en =P3^6;      //**LCD ENABLE PIN AT PORT-3 SIXTH PIN
sbit EyeBlinkSensor = P1^0;
sbit Buzzer=P1^1;   //Buzzer for alerting Purpose
sbit M11 = P1^2;
sbit M12 = P1^3;
sbit M21 = P1^4;
sbit M22 = P1^5;
//*DELAY FUNCTION//
void delay (unsigned char a)
{
  int i,j;
  for(i=0;i<a;i++)
  for(j=0;j<=1000;j++);
}
void Cmd (unsigned char a)
{
  rs=0;
  P2=a;
  en=1;
  delay(10);
  en=0;
}
void Data (unsigned char a)
{
```

```
  rs=0;
  P2=a;
  en=1;
  delay(10);
  en=0;
}
void String (unsigned char *p)
{
  while (*p)
  Data (*p++);
}
void main()
{
   M11 = 0;
   M12 = 0;
   M21 =0;
   M22=0;
   Buzzer=0;
   Cmd(0x38);   //COMMAND TO INNITILIZE THE LCD
   Cmd(0x0E);
   Cmd(0x80);
   String(" DROWSY DRIVER");
   Cmd(0xc0);
   String("DETECTING SYSTEM ");
   delay(500);
   Cmd(0x01);
while (1)
{
```

```
  if (EyeBlinkSensor == 0)


  {
     Buzzer=1;
      M11= 0;
      M12 = 0;
      M21 =0;
      M22=0;
     Cmd(0x80);
     String("DRIVER IS")
     Cmd(0xc0);
     String("SLEEPING...");
      else
      Buzzer=0;
      M11 = 1;
      M12 = 0;
      M21 =1;
      M22=0;
      Cmd(0x80);
      String(" DRIVER IS NOT ");
      Cmd(0xc0);
      String("SLEEPING");                                                    }
      }
   }
}
```

6. Applications:

1.Passenger Vehicles :

Embedded in modern cars by manufacturers (e.g., Tesla, BMW, Volvo).

Helps detect driver fatigue and alerts the driver before drowsiness leads to unsafe driving.

Promotes safer long-distance travel, especially at night or on highways.

2. Commercial and Fleet Vehicles :

Used in trucks, buses, and delivery vehicles to prevent fatigue-related crashes.

Important for logistics and transportation companies to: Ensure driver safety. Reduce liability.

Lower insurance and Improve compliance with regulations on driver work hours.

3. Public Transport :

Installed in buses, trains, and taxis to monitor operator alertness.

Enhances passenger safety by ensuring drivers are awake and focused.

4. Ride-Sharing and Taxi Services :

Used in platforms like Uber and Lyft (or in vehicles operated by their drivers) to monitor long shifts.

Encourages breaks and rest for drivers working extended hours.

5. Aviation and Rail Industries :

In aircraft cockpits or locomotive cabins, monitoring systems help detect pilot or operator fatigue.

Enhances safety in high-stakes, high-concentration roles.

7. Conclusion

- Drowsy driver detection systems play a crucial role in enhancing road safety by identifying signs of fatigue and preventing accidents caused by driver inattention or sleepiness. By utilizing technologies such as facial recognition, physiological sensors, and vehicle behavior analysis, these systems can accurately monitor driver alertness in real time.

8. References :

1. Muhammad Ali Mazidi and Janice Gillespie Mazidi and Rollind. The "8051 Microcontroller and Embedded Systems – Using Assembly and C", Mckinlay; Phi, 2006/ Pearson, 2006.