Alexandria university

Faculty of Engineering

Electrical Power and Machines

1st Term

LINE FOLLOWER ROBOT

Microprocessor Project

Team Members:

ID	Name
21010095	أحمد شريف أحمد غزال
20010557	رحمة إسماعيل محمد طعيمة
21010896	عمرو إسماعيل أحمد الجندى
21010972	كريم إبراهيم محمد السيد
21011036	مازن محمود شعبان محمد
21011159	محمد صلاح مختار إبراهيم
21011314	مروان يسري محمود طه
21011347	مصطفى أيمن مصطفى عزت
19015641	دينا أحمد حسين فوزي يوسف

Table of Contents

Introduction:

Robots stand as intricately designed machines, purpose-built to carry out specific and repetitive tasks with a level of speed and precision that surpasses human capabilities. The orchestration of these tasks primarily relies on advanced programming, manipulators, and an array of sensors.

Within the realm of robotics, the sensor-based black line follower robots emerge as fundamental entities adept at seamlessly tracking black lines against a contrasting white background, or vice versa. This particular robotic application showcases versatility, finding application in a spectrum of industrial and domestic scenarios. From the efficient transportation of goods to the meticulous task of floor cleaning, and from providing delivery services to serving general transportation needs, these robots demonstrate their adaptability across diverse domains.

This paper unfolds with a focus on presenting a circuit design that embodies simplicity and cost-effectiveness for a black line follower robot. The subsequent sections delve into the intricacies of its practical implementation, providing insights into the mechanisms and components that contribute to its functionality. Through this exploration, the aim is to contribute to the understanding of the fundamental concepts behind black line follower robots, laying the groundwork for further advancements in this domain.

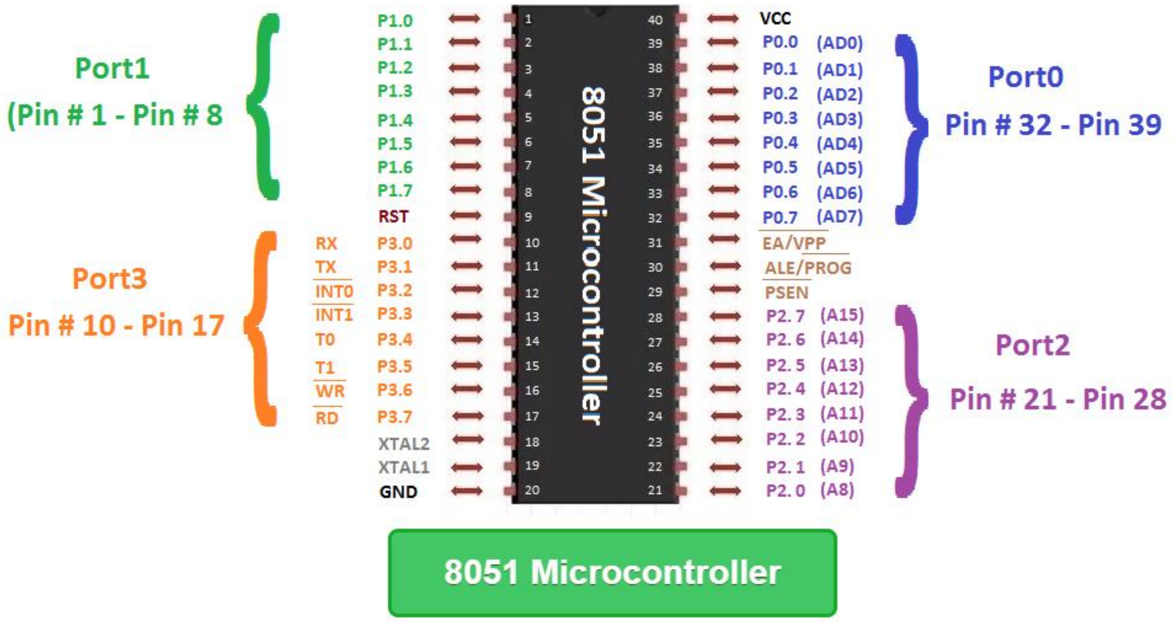
Objective:

1. **Assemble a Line Follower Robot with Enhanced Mobility:**
    - Fabricate a robust line follower robot with a carefully designed stable base to ensure durability and stability during operation.
    - Incorporate two driven wheels to propel the robot forward and one free wheel to enhance maneuverability, allowing for effective turns and directional changes.
    - Integrate specialized sensors strategically placed for efficient line detection, contributing to the robot's ability to precisely follow designated paths.
2. **Programming the 8051 Microcontroller Using Assembly Language:**
    - Employ Assembly language to program the 8051 microcontroller, utilizing a low-level programming paradigm to harness the microcontroller's full computational capacity.
    - Develop a program that interfaces seamlessly with the integrated sensors, enabling the microcontroller to receive and process real-time data for informed decision-making during the robot's navigation.
3. **Implementation of an Advanced Control Algorithm:**
    - Design and implement a sophisticated control algorithm that optimizes the robot's navigation capabilities.
    - Fine-tune the control algorithm to facilitate accurate and responsive line-following, taking into account variations in line patterns and environmental conditions.
    - Ensure the adaptability of the robot, allowing it to traverse a black line on a white surface with precision, demonstrating a high level of autonomy in following intricate and dynamic paths.

By accomplishing these comprehensive objectives, this project aims to showcase an integration of mechanical and electronic elements, emphasizing not only the assembly of a physically robust robot but also the intricate programming required to achieve intelligent and precise line-following behavior. Through this holistic approach, the project seeks to contribute to the advancement of autonomous robotic systems with practical applications in diverse environments.

Components Used:

1. 8051 Micro-Controller



The 8051 microcontroller is a widely used and historically significant microcontroller in the realm of embedded systems. Introduced by Intel in 1980, the 8051 has since become a staple in various applications due to its versatility and reliability. Here are key characteristics and features of the 8051 microcontroller:

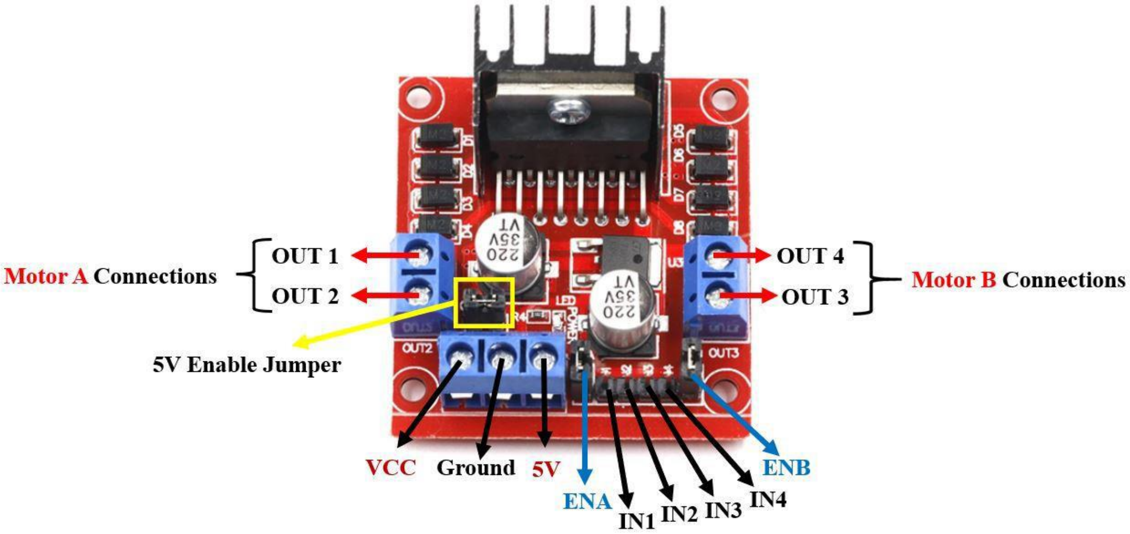
1. **Architecture:**
   - The 8051 microcontroller features a Harvard architecture with separate program and data memory spaces. It has an 8-bit processor and a rich set of integrated peripherals.
2. **Memory:**
   - The 8051 typically includes on-chip ROM (Read-Only Memory) for program storage and RAM (Random Access Memory) for data storage. The amount of memory can vary depending on the specific variant.
3. **Peripherals:**
   - It incorporates a variety of built-in peripherals such as timers/counters, serial communication ports, and interrupt controllers. These peripherals enhance its capability to interface with external devices.
4. **I/O Ports:**
   - The 8051 microcontroller typically provides multiple general-purpose I/O (Input/Output) ports, allowing it to interact with the external environment by receiving inputs and generating outputs.
5. **Clock Speed:**
   - The clock speed of the 8051 microcontroller varies, but it commonly operates in the range of a few megahertz, providing a balance between performance and power consumption.
6. **Instruction Set:**
   - The 8051 microcontroller has a well-defined and compact instruction set, making it efficient for programming in assembly language. This feature is particularly valuable for low-level programming and real-time applications.
7. **Applications:**
   - Due to its reliability and ease of use, the 8051 microcontroller has found applications in a wide range of embedded systems. It is commonly employed in industrial automation, automotive control systems, home appliances, and various electronic gadgets.

2. DC Motors

DC Motors Features:

1. **Power Supply:**
    - Operates on DC power supply directly or can be adapted for use with AC line voltage through a rectifier.
2. **Operating Speeds:**
    - Typically runs at speeds ranging from 1,000 to 5,000 revolutions per minute (rpm), providing flexibility for various applications.
3. **Efficiency:**
    - Exhibits an efficiency rate of 60-75%, ensuring effective energy utilization in diverse operational scenarios.
4. **Starting Torque:**
    - Possesses high starting torque, making it suitable for applications requiring a robust initial rotational force.
5. **No-Load Speeds:**
    - Demonstrates low no-load speeds, allowing for precise control over the motor's rotational behavior.
6. **Speed Regulation:**
    - Features inexpensive controls for speed regulation, offering a cost-effective solution for adjusting motor speeds as needed.
7. **Cost-Effective:**
    - Boasts a low-cost profile, making DC motors economical for a wide range of applications.
8. **Compatibility with Gear Reducers:**
    - Easily pairs with gear reducers, enhancing its adaptability and suitability for applications requiring torque multiplication or speed reduction.

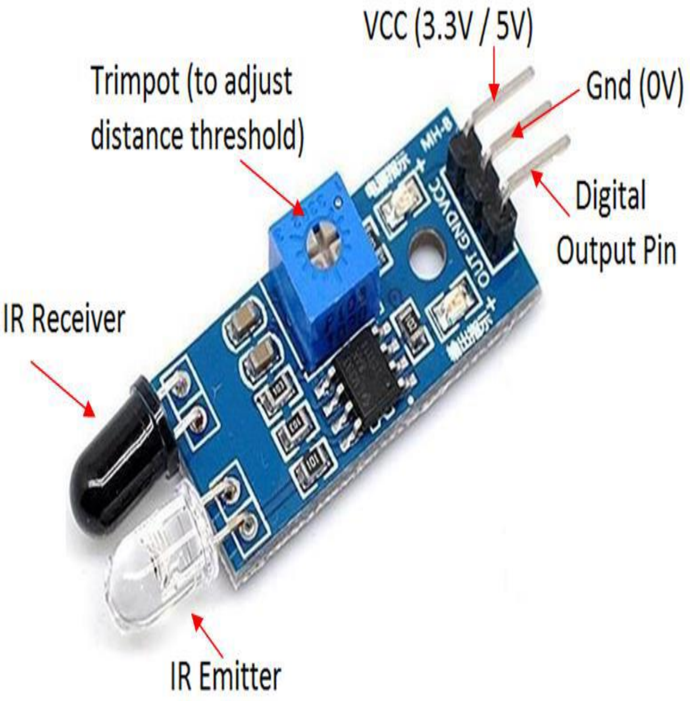
3. Motor Driver Module (L298)



Motor Driver Features:

1. **Dual H-Bridge Configuration:**
   - Incorporates a dual H-bridge configuration, allowing independent control of two DC motors or a single stepper motor.
2. **Operating Voltage:**
   - Supports a wide operating voltage range, making it compatible with various power supply configurations.
3. **Current Handling Capacity:**
   - Exhibits a high current-handling capacity, enabling it to drive motors with substantial power requirements.
4. **Built-in Protection Diodes:**
   - Includes built-in protection diodes, safeguarding the module and connected components from voltage spikes generated during motor operation.
5. **PWM Speed Control:**
   - Facilitates Pulse Width Modulation (PWM) speed control, offering a mechanism for precise regulation of motor speed.
6. **Enable and Disable Functionality:**
   - Provides enable and disable functionality for each motor channel, allowing for selective motor control based on application requirements.
7. **Versatile Control Inputs:**
   - Accepts a range of control inputs, making it compatible with various microcontrollers, Arduino boards, or other control systems.
8. **Compact and Durable Design:**
   - Features a compact design for ease of integration into projects and a durable construction for reliable performance in diverse applications.
9. **Overcurrent Protection:**
   - Incorporates overcurrent protection mechanisms to prevent damage to the module and connected motors in case of excessive current draw.
10. **Ease of Wiring:**
    - Simplifies wiring with clearly labeled terminal blocks, reducing the likelihood of errors during setup.

4. IR Sensors



Principle of Operation for Infrared (IR) Sensors:

Active infrared sensors function by both emitting and detecting infrared radiation. Comprising two essential components, these sensors consist of a light-emitting diode (LED) and a receiver.

As an object approaches the sensor, the infrared light emitted by the LED reflects off the object's surface and is subsequently detected by the receiver.

IR Sensor Features:

- Operates at a voltage of 5VDC.
- Supports I/O pins at 3.3V and 5V.
- Equipped with a mounting hole for convenient installation.
- Demonstrates a sensing range of up to 20 centimeters.
- Requires a supply current of 20mA.
- Offers an adjustable sensing range.
- Incorporates a fixed ambient light sensor.

IR Sensor Applications:

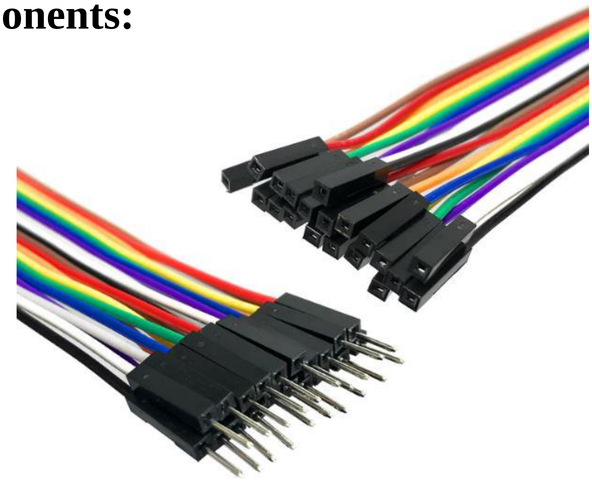
- Proximity sensor
- Item counter
- Radiation thermometers
- Human body detection

Datasheet of IR

https://pdf1.alldatasheet.com/datasheet-pdf/view/3068/MOTOROLA/LM393.html

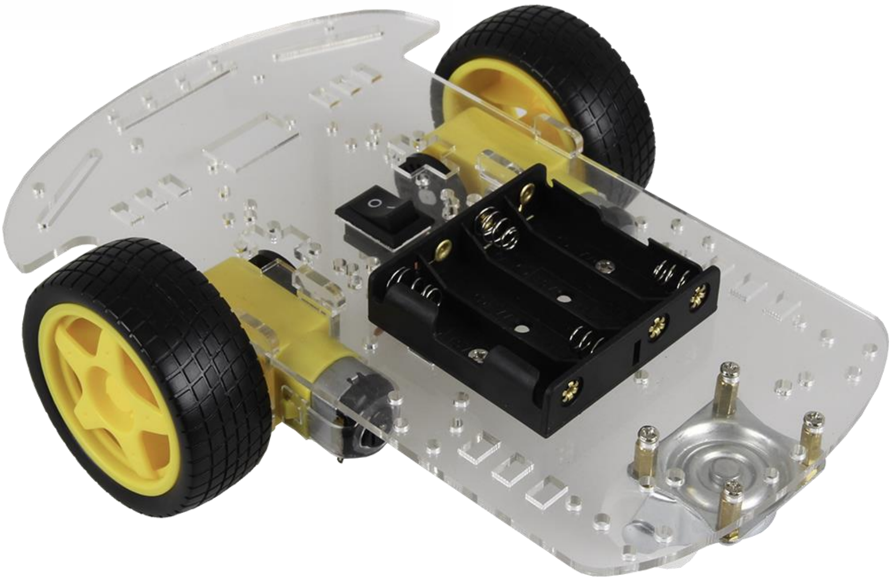
5. Other components:

1- Jumpers



2- Batteries

3- Robot chassis



4- Wheels

5- Battery holder

Procedures:

Building a Line Follower Robot with 8051 Microcontroller and L298 Motor Driver Module

1. **Gather Components:** Collect all necessary components, including the 8051 microcontroller, L298 Motor Driver Module, DC motors, chassis, wheels, sensors, and power supply.

2. **Design the Robot Chassis:** Assemble the robot chassis by attaching the motors and wheels. Ensure a stable base for the components.

3. **Connect the Motors to the L298 Module:** Wire the DC motors to the L298 Motor Driver Module following the module's datasheet. Connect the motor outputs to the designated terminals on the module.

4. **Connect Sensors to the 8051 Microcontroller:** Integrate the line-following sensors with the 8051 microcontroller. Establish connections based on the microcontroller's pin configuration and the sensor requirements.

5. **Connect L298 Module to the 8051 Microcontroller:** Establish connections between the 8051 microcontroller and the L298 Motor Driver Module. Use appropriate pins for motor control and communication.

6. **Program the 8051 Microcontroller:** Write an Assembly language program for the 8051 microcontroller to interpret sensor data and control the motors. Implement logic for line-following based on the input received from the sensors.

7. **Compile and Load the Program:** Compile the Assembly program and load it onto the 8051 microcontroller. Ensure that the code addresses the specifics of your robot's design and sensor layout.

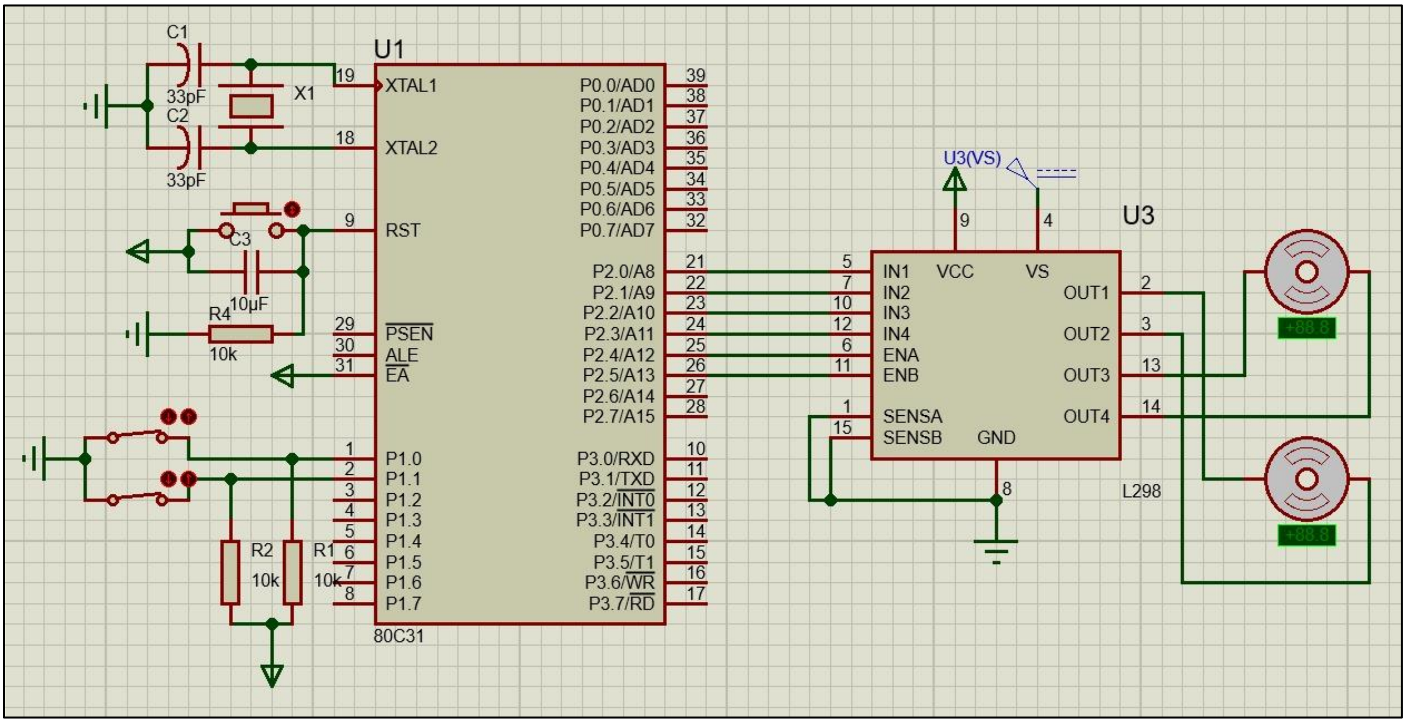
8. **Test the Line Following Algorithm:** Power up the robot and test its line-following capabilities on a white surface with a black line or vice versa. Tweak the algorithm and sensor calibration as needed for optimal performance.

9. **Integrate Power Supply:** Connect a suitable power supply to the robot, ensuring that it meets the voltage and current requirements of both the 8051 microcontroller and the motors.

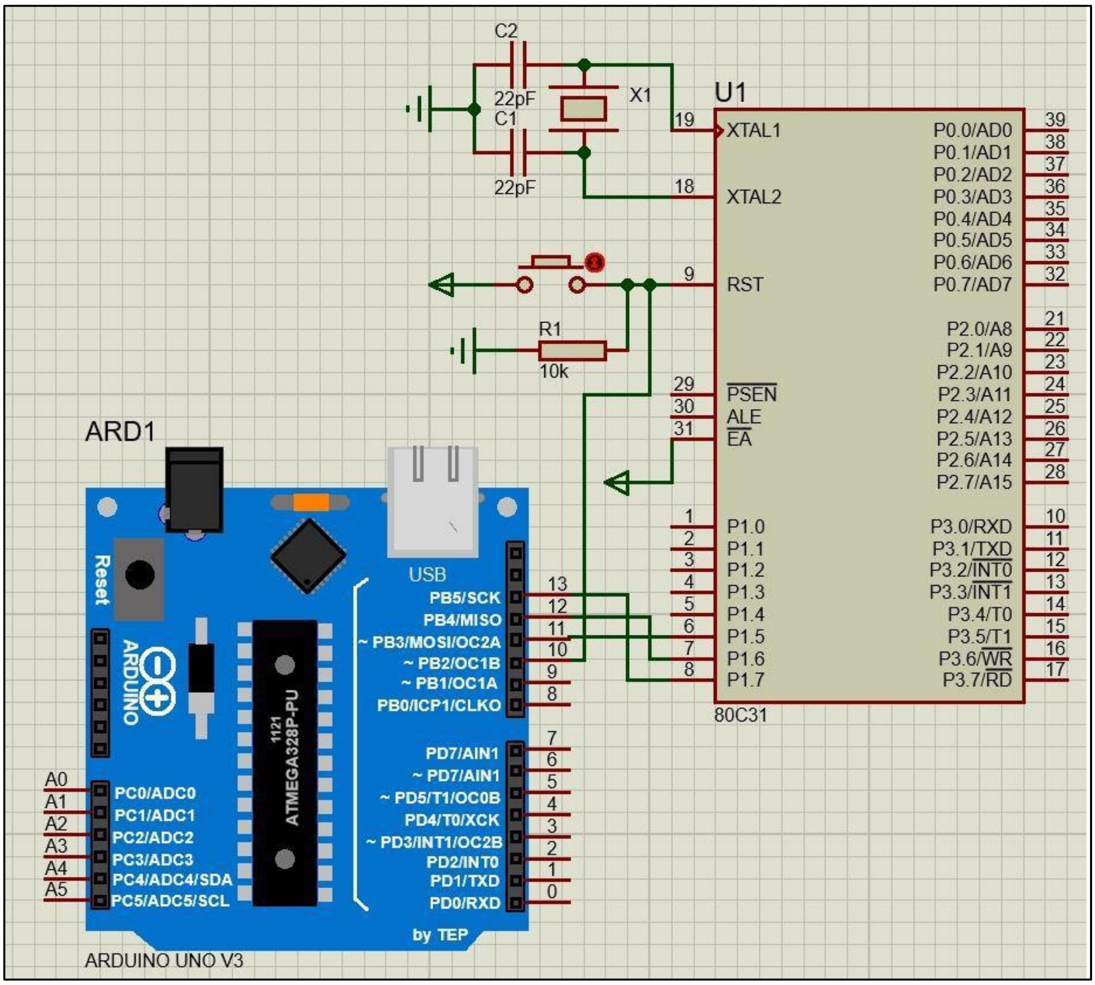
10. **Fine-Tune and Optimize:** Fine-tune the control algorithm, adjusting parameters as necessary to optimize the robot's responsiveness and accuracy in following the designated line.

11. **Document and Present:** Document the entire process, including circuit diagrams, code snippets, and any modifications made during testing. Present your findings and the functionality of your line follower robot.

Circuit Design:



Connection with Arduino Uno as a programmer which is used to program (upload the code into) the 8051 microcontroller as in the circuit:



Code:

```
ORG 0000H
AJMP MAIN
ORG 000BH  ; Timer Interrupt ISR, When TCON.5 is set, this ISR is executed
ACALL PWM
RETI

ORG 0030H
MAIN:
MOV P1,#0FFH      ;INIT as INPUT pins
MOV P2,#00H             ;INIT as OUTPUT pins
MOV R7,#0FFH      ;R7 is responsible for switching ON and OFF
MOV TMOD, #01H    ;Setting Timer Mode to 1
MOV IE,#82H       ;Enabling Timer0 Interrupt and global interrupt
MOV TH0,#00H      ;INIT TH0
MOV TL0,#00H      ;INIT TL0
CLR P2.0
CLR P2.2
SETB TCON.4       ;Starting the timer
MAINLOOP:
JNB P1.0,StraightOrRight
JNB P1.1,TurnLeft
STOP: ;Stops both motors
CLR P2.1
CLR P2.3
AJMP MAINLOOP

TurnLeft: ;Stops left motor and starts right motor
CLR P2.1
SETB P2.3
AJMP MAINLOOP
```

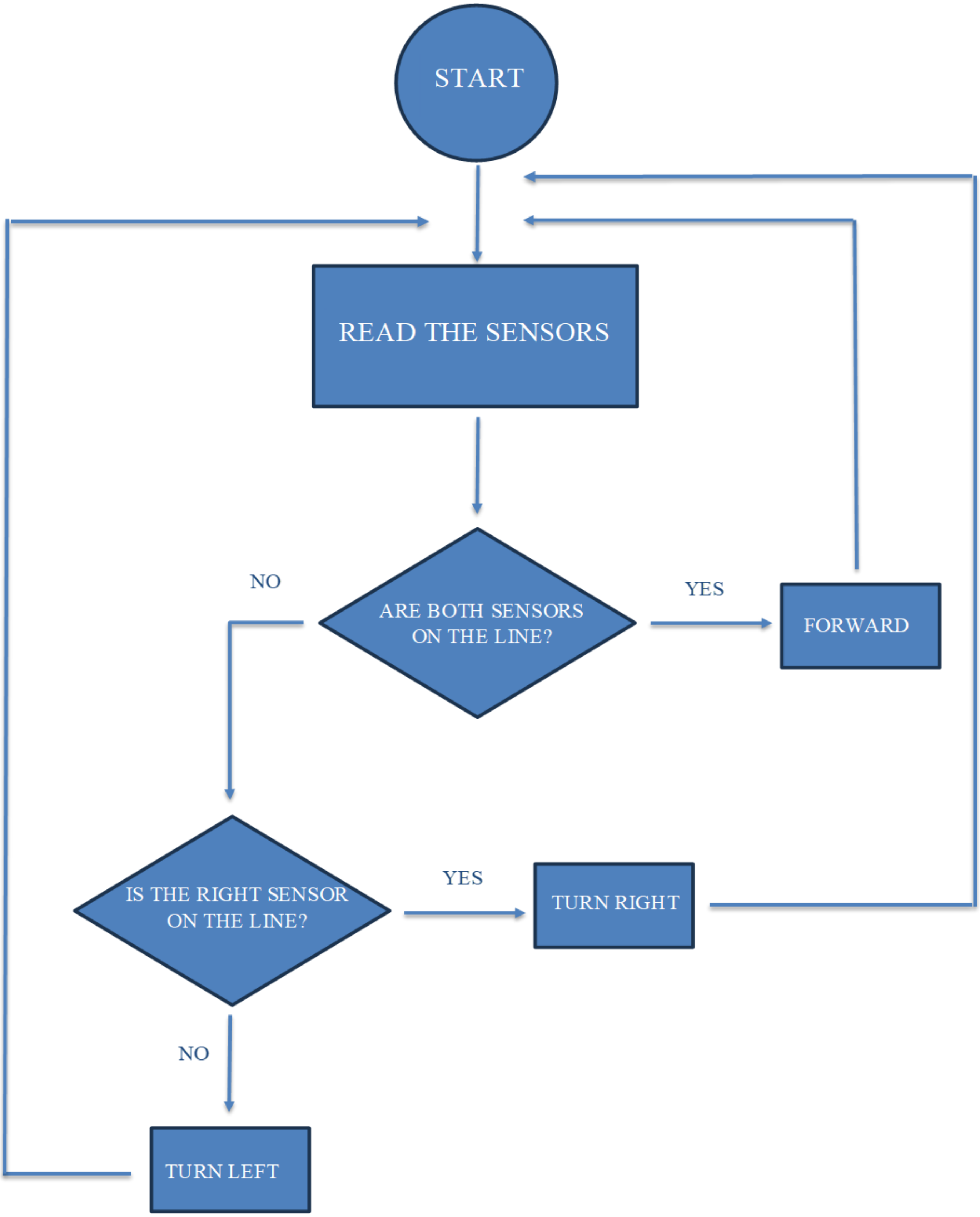
```
StraightOrRight:
JNB P1.1,Straight
TurnRight: ;Stops right motor and starts left motor
SETB P2.1
CLR P2.3
AJMP MAINLOOP

Straight: ;Starts both motors
SETB P2.1
SETB P2.3
AJMP MAINLOOP

ORG 0400H
PWM:
CJNE R7,#0FFH,OFFF ;Checks value of R7 and goes to ON/OFF label accordingly
ONN:            ;To set the Delay for ON time
MOV R7,#00H
SETB P2.4
SETB P2.5
MOV TH0,#0F8H
MOV TL0,#30H
RET
OFFF:           ;To set the delay for OFF time
MOV R7,#0FFH
CLR P2.4
CLR P2.5
MOV TH0,#0F4H
MOV TL0,#48H
RET

END
```

Flow Chart:


START
READ THE SENSORS
NO
ARE BOTH SENSORS ON THE LINE?
YES
FORWARD
IS THE RIGHT SENSOR ON THE LINE?
YES
TURN RIGHT
NO
TURN LEFT