جامعة الإسكندرية
ALEXANDRIA UNIVERSITY
كلية الهندسة
Faculty of Engineering

Automatic Control Project

	الاسم	الرقم
1	محمد أشرف عبدالرحمن إبراهيم	17011478
2	أحمد طارق إدريس التابعي	21010105
3	عبدالرحمن أحمد عزت	20010782
4	محمود أحمد سعد	20011774
5	عبدالرحمن محمود عبدالفتاح القصبي	21010782
6	نور عاطف أحمد	20012132
7	يوسف محروس سليم	21011611
9	علي أسامة عبدالعزيز	20010932
8	سلمى إبراهيم السيد	21010613
10	سمر خالد خميس	19015793
11	عبدالله السيد عبدالقادر	20010881

• QR Codes

Scan to enter control page Or Press Here if using PDF

Please Note that you must connect to the robot Wi-Fi Network (**Sonic**) and scan QR Code on first page to enter

You may Have To Disable Mobile Data First to connect properly

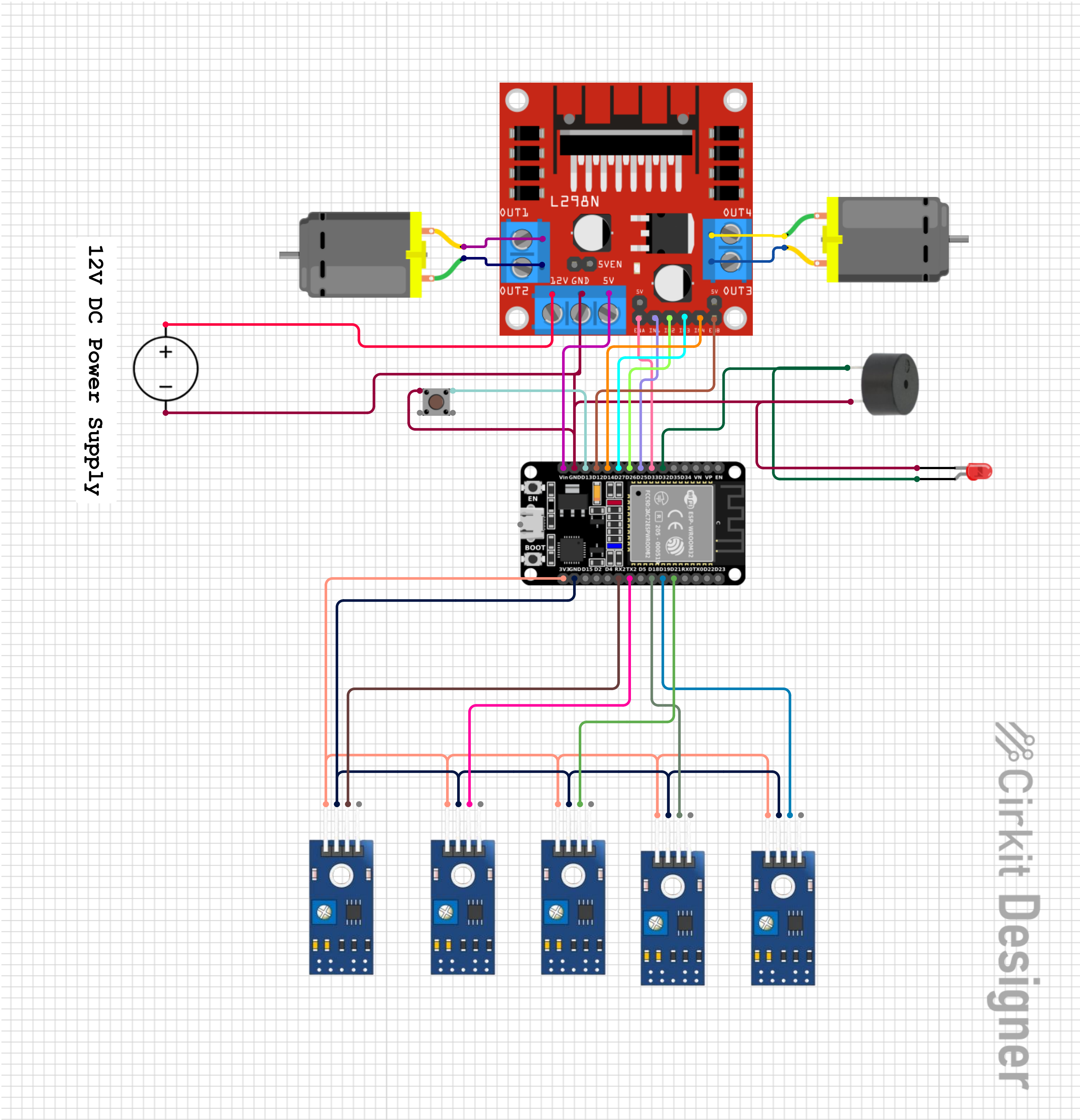
Scan To get the full code and user manual from GitHub

Scan to open circuit Diagram on CirkitDesigner

Or press here if using PDF

Scan To get Videos Page or click Here if using PDF

• Circuit Diagram



Components Used

• DC Mototrs

Specs

1. Controlled Speed for Better Sensor Response

600 RPM is a moderate speed, allowing your minicar to move steadily without overwhelming the TCRT sensor.

If the car moves too fast, the infrared sensor may not detect lines or obstacles in time, causing poor performance or crashes.

2. Good Torque Balance

DC motors at lower RPMs generally offer higher torque, which is useful for:

- Smoother starts and stops.
- Moving over uneven surfaces.
- Turning or maneuvering at low speeds, which is ideal for sensor-based navigation.

3. Voltage Compatibility

12V is common and easy to supply, especially with small battery packs.

Many microcontrollers and motor drivers used in hobby robotics (like L298N or TB6612FNG) are designed to work well with 12V motors.

4. Power Efficiency

12V motors tend to be more efficient than lower-voltage ones, especially at moderate loads.

This helps extend battery life, which is critical in small mobile robots.

5. Fine Control with PWM

At 600 RPM, using Pulse Width Modulation (PWM) for speed control gives you finer control compared to very high RPM motors, helping you:

- Adjust speed based on sensor feedback.
- Improve line-following accuracy.

• Micro Controller (ESP32)

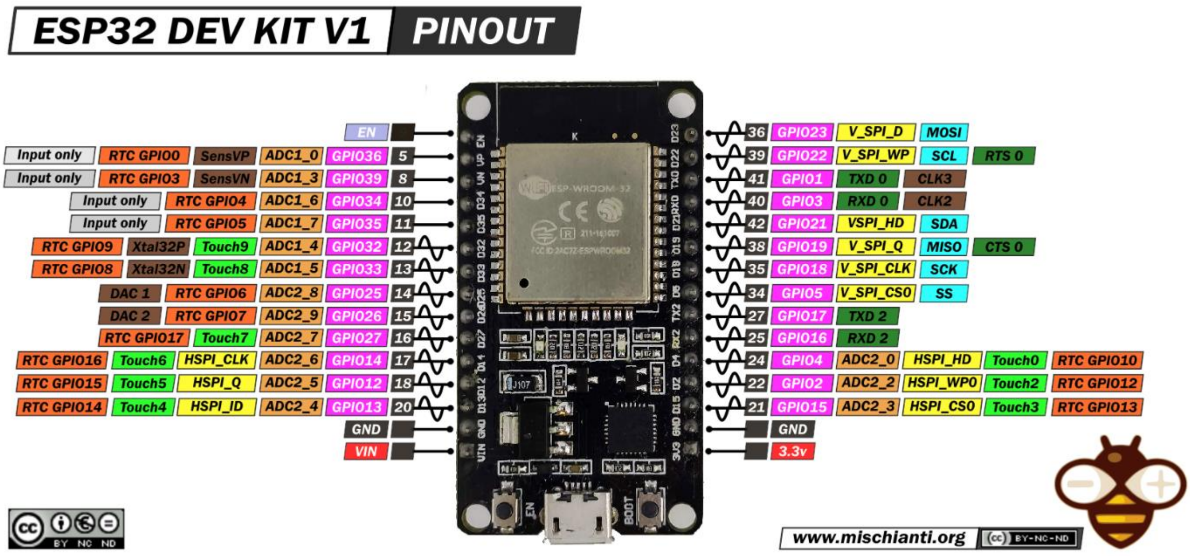
1- Overview

Arduino: An open-source microcontroller platform with a simple IDE and extensive community support. Commonly uses ATmega328 (Uno) or ATmega2560 (Mega).
- ESP (ESP8266/ESP32): Developed by Espressif Systems, these are low-cost Wi-Fi and Bluetooth-enabled microcontrollers

2 – Why we chose ESP32?

Feature	Arduino	ESP (ESP8266/ESP32)
Ease of Use	Beginner-friendly, intuitive IDE	Slightly steeper learning curve
Community Support	Very large and established community	Growing community, strong support
I/O Pins	Good number for general applications	Fewer on ESP8266; ESP32 has more
Connectivity	Requires modules for Wi-Fi/Bluetooth	Built-in Wi-Fi (ESP8266/32), Bluetooth (ESP32)
Processing Power	16 MHz clock (Uno) 8 Bits Processor	Up to 240 MHz (ESP32) 16 Bits Processor
Price	Inexpensive	Also very affordable for its features
Internet of Things	Needs extra hardware for connectivity	Better suited with built-in connectivity
Speed & Memory	Slower and less RAM	Faster but can be overkill for basic tasks
Power Consumption	Less optimized for low power	ESP32 can consume more power
IDE Integration	Limited features in Arduino IDE	Better support in ESP-IDF/PlatformIO
3.3V vs 5V	Most are 5V logic	ESP uses 3.3V (needs level shifting)

• Micro Controller (ESP32)



The image above shows the pinouts of the board

Left is the ESP32 Board, and the above is the devkit that implements esp32 board with other components for easier usage and programming, like voltage regulator and USB to UART IC for programming through USB

We used the Wi-Fi capability of this board to implement a GUI for controlling the car remotely through the web UI, We also wanted to implement the Bluetooth capability to make wireless serial monitor but unfortunately it is not possible to use both options simultaneously because they share the same antenna but however the GUI is enough for good control for now

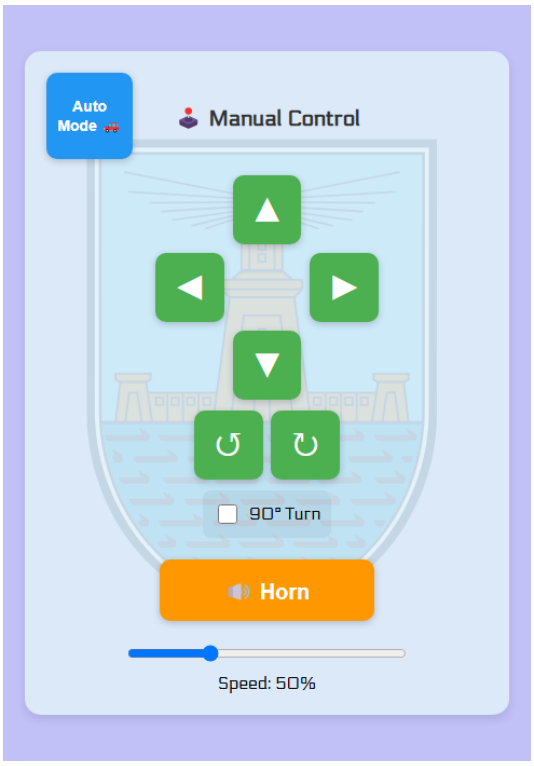
The GUI adds the capability of both automatic path following using sensors or manual control that can be , with various options of changing speed, PID variables, and some delays in the system, either from mobile phones or computers, you can also use keyboard for control in manual mode

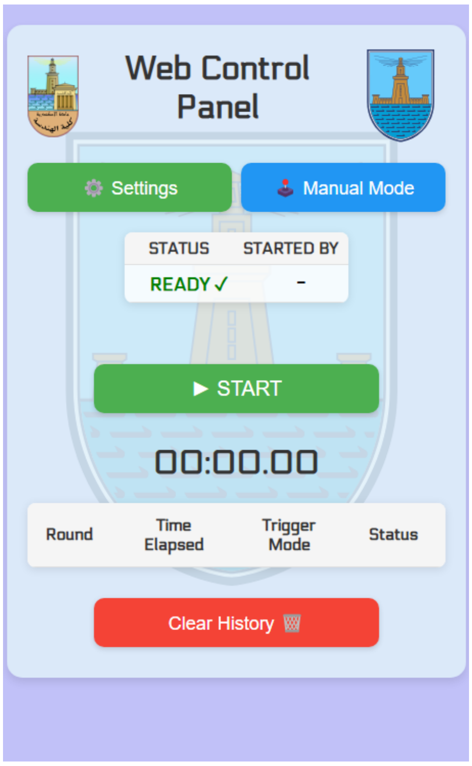
• Micro Controller (ESP32)

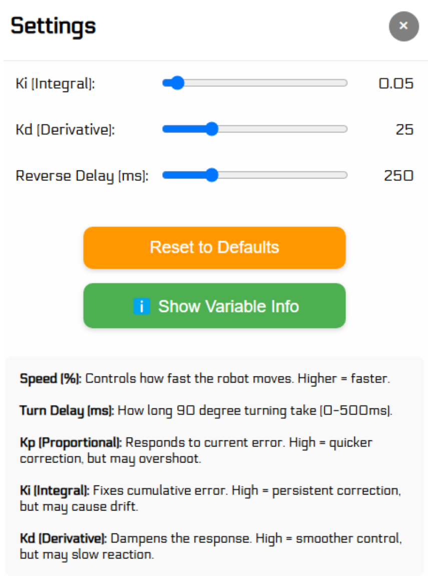
Screenshots of the control UI

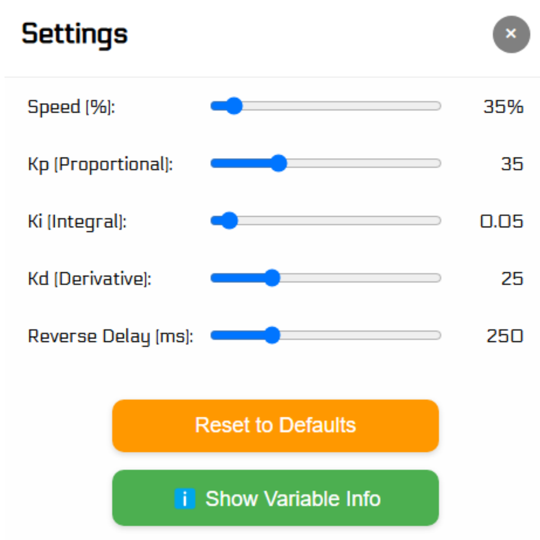
Please Note that you must connect to the robot Wi-Fi Network (**Sonic**) and scan QR Code on first page to enter

Or manually enter the IP address http://192.168.4.1







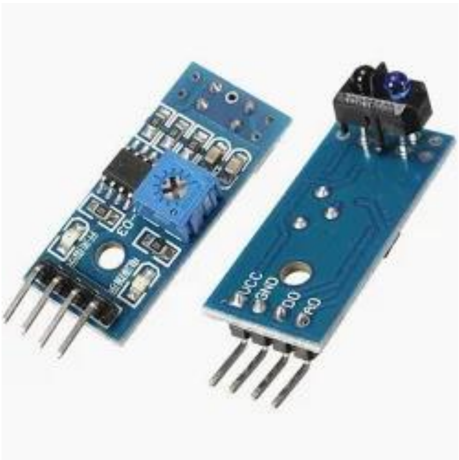


• IR Sensors (TCRT5000)

Our innovative electric car project relies on accurately following a defined path marked by two black lines on a white surface. To achieve this crucial task, the "eyes" of our robot car are a set of infrareds (IR) sensors. After careful consideration and experimentation, we've chosen the **TCRT5000 reflective optical sensor** as our primary line detection tool, and we've opted for a configuration of **five sensors** strategically positioned at the front of our vehicle.

This article will delve into the rationale behind these choices, highlighting the advantages of the TCRT5000 over other sensor types, explaining how the increased sensor count contributes to superior path-following accuracy, and providing a detailed look at the TCRT5000 itself.

Understanding the TCRT5000 IR Sensor



The TCRT5000 is a reflective optical sensor commonly used for line following and object detection. Here's a breakdown of its components, function, working mechanism, and options:

Components:

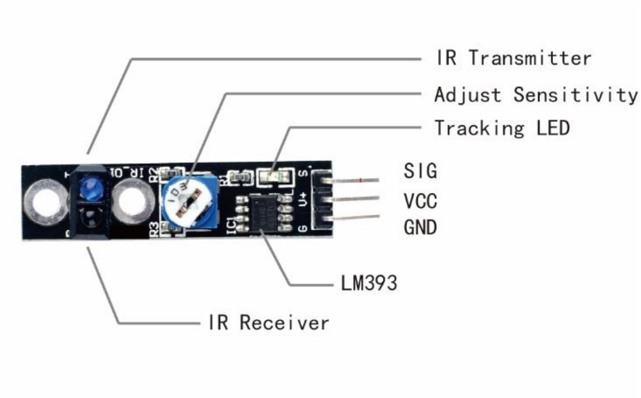
The TCRT5000 essentially consists of two main parts integrated into a single package:

1. **Infrared (IR) Emitting Diode (LED):** This component emits infrared light at a specific wavelength (typically around 950 nm), which is invisible to the human eye.

2. **NPN Phototransistor:** This is a light-sensitive transistor. When infrared light of the appropriate wavelength strikes the phototransistor, it starts to conduct, allowing current to flow from the collector to the emitter. The amount of current that flows is proportional to the intensity of the received IR light.

• IR Sensors (TCRT5000)

Many TCRT5000 modules also include additional components on a small circuit board:



- **Current Limiting Resistor for the IR LED:** This resistor is crucial to prevent excessive current from flowing through the IR LED, which could damage it.

- **Pull-up Resistor for the Phototransistor:** When the phototransistor is not receiving reflected IR light (or very little), it acts like an open switch. A pull-up resistor connected to the collector ensures that the output signal is pulled to a high voltage level. When IR light is received and the phototransistor conducts, it pulls the output voltage down.

- **Potentiometer (on some modules):** This variable resistor allows you to adjust the sensitivity of the sensor's digital output by setting a threshold voltage for a comparator.

- **Comparator (e.g., LM393 on some modules):** This integrated circuit compares the analog voltage from the phototransistor with a reference voltage (often set by the potentiometer). It outputs a digital signal (high or low) based on this comparison, making it easy to interface with a microcontroller.
- **Power LED:** Indicates if the module is receiving power.

- **Output LED:** Indicates the state of the digital output (e.g., lit when a black line is detected).

Working Mechanism:

1. **Emission:** The IR LED continuously emits infrared light.

2. **Reflection:** When this infrared light encounters a surface, some of it is reflected back towards the sensor. The amount of reflection depends on the surface's properties, particularly its color and reflectivity. White surfaces reflect significantly more IR light than black surfaces, which tend to absorb it.

3. **Detection:** The phototransistor detects the intensity of the reflected infrared light. A stronger reflection results in a higher current flow through the phototransistor.

• IR Sensors (TCRT5000)

4. **Output:**

   - **Analog Output (if available):** The voltage at the collector of the phototransistor (with the pull-up resistor) provides an analog signal proportional to the amount of reflected IR light. Higher reflection (e.g., from a white surface) typically results in a lower voltage at the output, while lower reflection (e.g., from a black surface) results in a higher voltage.

   - **Digital Output (if using a comparator):** The comparator compares the analog voltage from the phototransistor to a set threshold. If the analog voltage is above or below the threshold, the comparator outputs a digital high or low signal, indicating whether a black line (or an object with low reflectivity) is detected. The potentiometer allows you to adjust this detection threshold.

Options and Considerations:

- **Analog vs. Digital Output:** As mentioned, some TCRT5000 modules provide both analog and digital outputs, offering flexibility in how you process the sensor data. Analog output provides more nuanced information about the reflectivity, while digital output offers simple black/white detection.

- **Breakout Boards:** The TCRT5000 is often available on breakout boards that include the necessary resistors, a potentiometer for sensitivity adjustment, and a comparator for digital output. These boards simplify interfacing with microcontrollers.

- **Sensitivity Adjustment:** The ability to adjust the sensitivity (usually via a potentiometer on the module) is crucial for calibrating the sensor to reliably detect the black lines on your specific white surface under your lighting conditions.

- **Operating Distance:** The TCRT5000 has a typical operating distance of a few millimeters (e.g., 0.2 mm to 15 mm, with an optimal range around 2.5 mm). You need to ensure your sensor mounting maintains this distance from the tracking surface for reliable operation.

- **Daylight Filtering:** The package of the TCRT5000 often includes a filter to minimize interference from visible light, improving its performance under different lighting conditions.

• IR Sensors (TCRT5000)

Why the TCRT5000? A Superior Choice for Line Following

When selecting a sensor for line following, several options are available, each with its own characteristics. We considered alternatives such as:

- **Simple Photoresistors:** These light-dependent resistors change their resistance based on the intensity of light. While inexpensive, they often lack the precision and focused detection needed for reliable line following. Ambient light variations can significantly affect their readings, making it difficult to consistently differentiate between the black line and the white surface.

- **Ultrasonic Sensors:** These sensors work by emitting sound waves and measuring the time it takes for the echo to return. They are excellent for distance measurement and obstacle avoidance but are not well-suited for detecting thin lines on a surface. Their detection area is broader, making it hard to pinpoint the edge of a line accurately.

- **Color Sensors:** These sensors can identify different colors. While seemingly ideal for detecting black lines on a white surface, they can be more complex and expensive than IR sensors. They might also be affected by variations in the shade of black or the reflectivity of the white surface.

- **QTR Reflectance Sensors (e.g., QTR-8RC):** These sensors, often from Pololu, are specifically designed for line following and often come in arrays. They can be very precise and offer both digital and analog outputs. However, individual TCRT5000 sensors and their breakout boards can be more readily available and cost-effective for specific sensor arrangements. The adjustable sensitivity on many TCRT5000 modules can also be advantageous.

- **IR Proximity Sensors (e.g., Sharp GP2Y0A Series):** While using IR light, these sensors are designed for longer-range distance measurement and have a broader detection beam, making them less suitable for precise line detection compared to the TCRT5000's focused reflective sensing.

- **Discrete IR LED and Phototransistor Pairs:** While offering flexibility, these require careful alignment and may be more susceptible to ambient light interference compared to the integrated and filtered design of the TCRT5000.

• IR Sensors (TCRT5000)

The Power of Five: Enhanced Accuracy in Path Following

Our initial experimentation with three IR sensors provided a basic level of line detection. However, we observed that with only three points of reference, the car could sometimes oscillate or lose its way, especially when encountering curves or slight irregularities in the path. This led us to explore the benefits of increasing the sensor count to five.

Here's how using five sensors significantly improves the accuracy of our electric car's path following:

1. **Increased Resolution of Path Detection:** Five sensors provide a finer "grain" of information about the car's position relative to the black lines. Instead of just a left, center, and right reading, we now have a more detailed profile of where the lines are in relation to the car's front.

2. **More Granular Error Detection:** With five sensors, we can detect even subtle deviations from the center of the path much earlier. For instance, if only one of the outermost sensors starts to detect the black line, we can initiate a small corrective action before the car drifts too far.

3. **Improved Handling of Curves:** When the car approaches a curve, the outer sensors will start to detect the line before the inner sensors. This staggered detection provides an early indication of the upcoming turn, allowing the control system to initiate a smoother and more controlled steering response. With only three sensors, this anticipation is less precise.

4. **Enhanced Stability:** The additional sensor readings provide more data points for our control algorithm to work with. This allows for more sophisticated control strategies that can dampen oscillations and maintain a more stable trajectory along the path. The system can better differentiate between a sharp turn and a minor wobble.

5. **Redundancy and Robustness:** While not the primary reason for the increase, having more sensors can also offer a degree of redundancy. If one sensor malfunctions or becomes temporarily obscured, the other four can still provide enough information for the car to continue following the path, albeit potentially with slightly reduced accuracy.

By understanding these aspects of the TCRT5000 and considering the advantages of using five sensors, your electric car project is well-equipped for accurate and reliable line following.

• Auxillary Parts

Buzzer: Used for giving sound feedback for different operating Modes

Red LED: Used For giving visual feedback and it is connected parallel with the buzzer

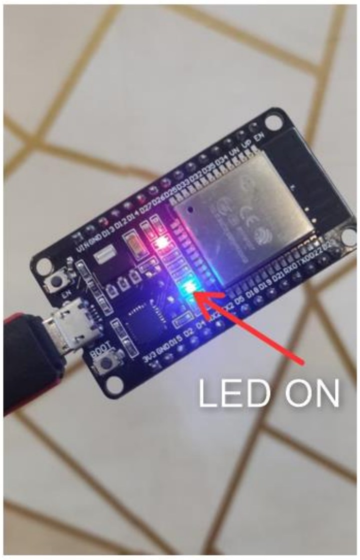
Push Button: Used for starting or stopping the robot car using digital signal instead of physical switch, it works alongside the web UI remote start button

Built in LED: shows different operating modes In Blue light

Running (Fast blink)

Manual Mode (Slow blink)

Stopped/Ready (Off)



PID

Control

• PID Control

Modeling Equations

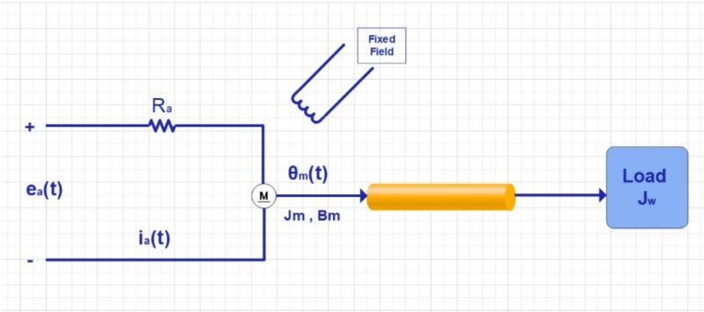
$$\mathrm{ei(t)} - \mathrm{eb(t)} = \mathrm{R\,i(t)} + \mathrm{L}\frac{di(t)}{dt}$$

$$\mathrm{eb(t)} = \mathrm{Kv}\,\dot{\theta}m(t)$$

$$Tm(t) = Kt\,i(t)$$

$$Tm(t) - Tw(t) = Jm\,\ddot{\theta}m(t) + Bm\,\dot{\theta}m(t)$$

$$Tw(t) = Jw\,\ddot{\theta}m + \mathrm{\mu Mgr}$$



Taking Laplace Transform

$$\mathrm{Ei(s)} - \mathrm{Eb(s)} = \mathrm{R\,I(s)} + \mathrm{L}\,S\,\mathrm{I(s)} = [R + LS]\,I(S)$$

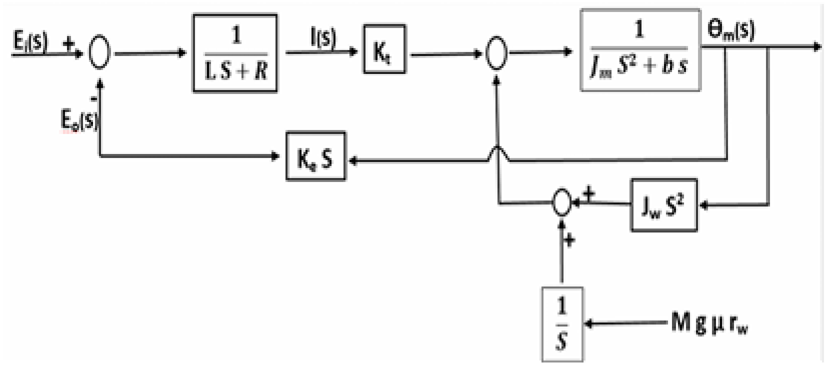
$$\mathrm{Eb(s)} = \mathrm{Kv\,S\,\Theta m}(s)$$

$$Tm(s) = Kt\,I(s)$$

$$Tm(s) - Tw(s) = Jm\,S^2\,\theta m(s) + Bm\,S\,\mathrm{\Theta m}(s) = [Jm\,S^2 + Bm\,S]\,\theta m(s)$$

$$Tw(s) = Jw\,S^2\,\theta m(s) + \frac{\mathrm{\mu Mgr}}{\mathrm{S}}$$

Block Diagram



Forward Path

$$P1 = \frac{Kt}{[R + LS]\,[Jm\,S^2 + Bm\,S]}$$

Loops

$$L1 = \frac{-Kt\,Kv\,S}{[R + LS]\,[Jm\,S^2 + Bm\,S]}$$

$$L2 = \frac{-Jw\,S^2}{[Jm\,S^2 + Bm\,S]}$$

Non touching loops

All of them are touching.

Δ

$\Delta = 1 - \sum L = 1 - (L1 + L2)$

$\Delta = 1 + \frac{1}{[Jm\, S^2 + Bm\, S]}\left[\frac{Kt\, Kv\, S}{[R + LS]} + Jw\, S^2\right]$

$\Delta\, 1 = 1$

T.F

$$T.F = \frac{P1\,\Delta\,1}{\Delta}$$

$$T.F = \frac{Kt}{[R + LS]\,[Jm\, S^2 + Bm\, S]} * \frac{1}{1 + \frac{1}{[Jm\, S^2 + Bm\, S]}\left[\frac{Kt\, Kv\, S}{[R + LS]} + Jw\, S^2\right]}$$

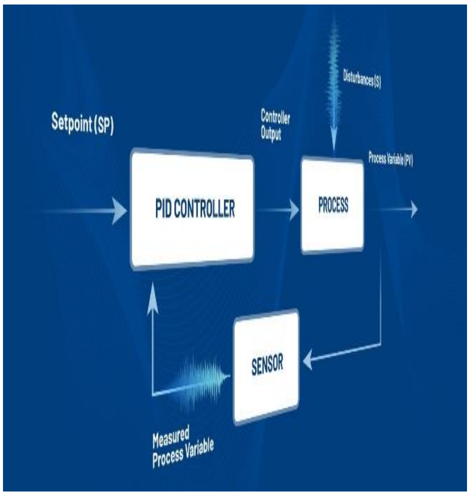
$$T.F = \frac{Kt}{[R + LS]\,[Jm\, S^2 + Bm\, S] + \left[\,Kt\, Kv\, S + Jw\, S^2\,[R + LS]\right]}$$

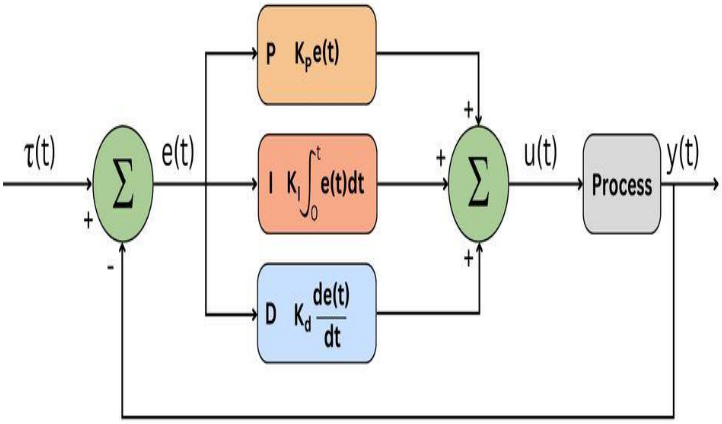
Controller Design

PID Controller

PID (proportional-integral-derivative) control was first introduced around 70 years ago and has since become a mainstay in engineering applications. The proportional, integral, and derivative (PID) control approach is a frequently used feedback control mechanism that is based on these three essential components. This control technique is essential for resolving complicated dynamics issues in a range of engineering systems because of its simplicity, flexibility, and efficacy. PID control is a commonly used industrial automation technique in manufacturing, production processes, and other industrial applications. PID control is now used in industrial automation, although there are still issues with it, including the need for complicated systems, performance optimization, and disturbance immunity

• PID Control





- Mathematical Representation of PID Controller

$$u(t) = K_p . e(t) + k_i \int_0^t e(t)\ dt + K_d \frac{de(t)}{dt}$$

$$G_c(s) = K_p + \frac{K_i}{s} + K_d \times s$$

where u(t) is the controller's output, e(t) is the current error. and Kp , Ki, Kd are the proportional, integral, and derivative tuning parameters respectively

<u>Principle of PID controller</u>

A PID controller (Proportional-Integral-Derivative controller) is a popular control algorithm used in robotics, including line-following robots, to maintain a desired path by minimizing the error between a setpoint (the line) and the robot's actual position

The utilization of PID control comes in handy when the developer does not fully understand the structure and parameters of the system of study, or when there is no precise mathematical model and other control theories may be difficult to apply, a time when the structure and parameters of the system's controllers must be determined by experience and field debugging. In practice, variants of PID control are also used, the classical ones being PI and PD control

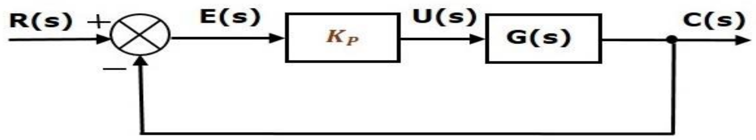
• PID Control

Proportional control

According to the expression, the proportional control is proportional to the current control error

$$u(t) = K_p\, e(t)$$

$$G_c(s) = K_p$$



If the proportional value is high, it results in a larger output for the same error. However, an excessively high proportional gain can lead to system instability. Conversely, a low proportional gain yields a smaller output for the same error, making the controller not too sensitive. Yet, if the value of proportional is too small, the support signal is likely to be insufficiently large to correct for interference effects

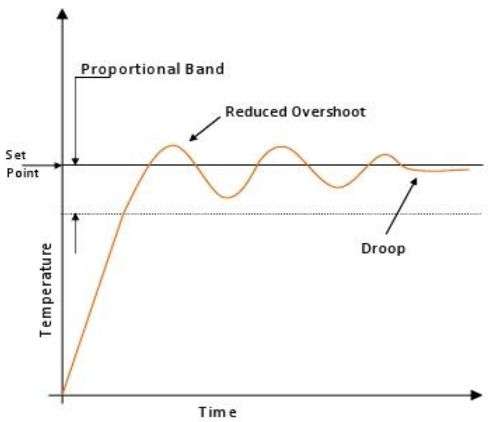
Integral control

Integral control is positively dependent on the integral of the set of control uncertainties

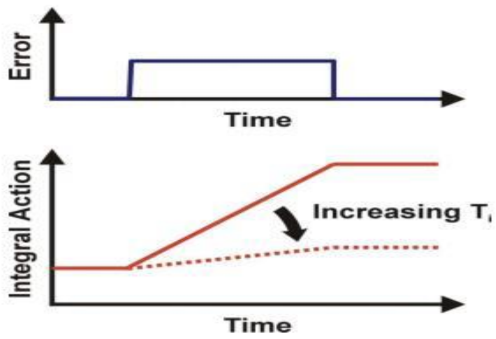
Integral control reduces the time to converge to setpoint and eliminates some of the steady state error. The higher the integrated charge gain, the shorter the time to converge to setpoint, but since the integral control cumulates whole previous errors, it may result in an overshoot of the return value.

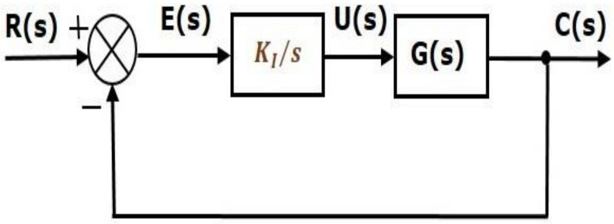
$$u(t) = k_i \int_0^t e(t)\ dt$$

$$G_c(s) = \frac{K_i}{s}$$



• PID Control





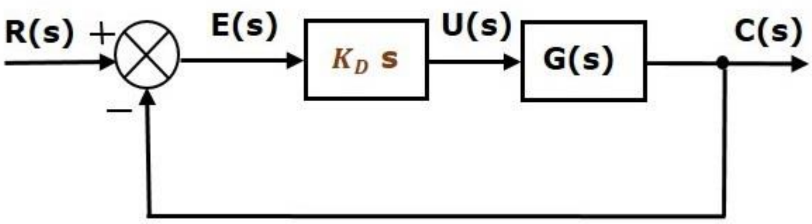
Integral control ensures that any persistent, small errors accumulate and result in corrective actions, driving the steady-state error to zero. However, excessive integral action can lead to slow response and oscillations.

<u>Derivative control</u>

Differential control improves tuning time and system stability , dampens oscillations and reduces overshoot. However, derivative control is sensitive to noise, which can limit its practical application

$$u(t) = K_d \frac{de(t)}{dt}$$

$$G_c(s) = K_d \times s$$



• PID Control

- **The aim of PID control system** in this project. It continuously reads **five line sensors** to calculate an error based on their weighted readings, then applies this error to adjust motor speeds and keep the robot centered on the line. Additionally, the code incorporates specialized functions to **detect and handle 90-degree turns** by temporarily stopping, backing up, and performing detection sweeps (left and right), and it also identifies the **end of the track** when all sensors are high.
- **From this the best choose is to use PID Controller** is a versatile and powerful tool in control systems of robot offering a structured approach to minimizing error and optimizing system performance