**University of Puerto Rico**
**Mayagüez Campus**
**Department of Mechanical Engineering**

# *Final Project:*
# Autonomous Race Car
# *(5/15/2025)*

Group 2

| | |
|---|---|
| Kendall Soto Vázquez | 802-20-0725 |
| Gabriela M. González Rodríguez | 840-20-5313 |
| David E. Candelario Recabarren | 842-20-0685 |

INME4237-016

Inst. Steven Mora Navarro

5/15/2025

## TABLE OF CONTENT

## TABLE OF SYMBOLS

| Symbol | Units | Name |
|---|---|---|
| Ω | Ohm | Ohmage |
| V | Volts | Voltage |
| cm | Centimeter | Centimeter |

## INTRODUCTION

This project focuses on the design, assembly, and programming of an autonomous Arduino-based vehicle equipped with multiple ultrasonic sensors for obstacle detection and basic navigation capabilities. The primary objective of the project is to implement sensor-based control logic using an Arduino microcontroller to achieve forward motion, environmental awareness, and directional decision-making based on proximity data. Through this process, students become familiar with real-time sensor integration, motor control, and embedded systems logic.

The motivation behind this project stems from the growing relevance of autonomous systems in modern engineering applications, particularly in areas such as robotics, automated delivery systems, and smart mobility solutions. Hands-on experience with microcontrollers, sensors, and actuators allows students to bridge the gap between theoretical coursework and real-world problem solving. This project presents an accessible platform to explore foundational concepts such as pulse-width modulation (PWM), servo positioning, and ultrasonic distance measurement.

Historically, the use of autonomous vehicles in research and industry has evolved significantly since the development of early robotic platforms in the 20th century. Pioneering work in mobile robotics, such as the Stanford Cart and Carnegie Mellon’s NavLab series, laid the groundwork for today’s advanced autonomous technologies. While the present project is limited in scale, it reflects the same fundamental engineering principles used in more complex applications, offering a valuable learning opportunity within an educational laboratory setting.

## THEORETICAL BACKGROUND

The autonomous race car designed should be able to detect and avoid obstacles, in this case, wooden obstacles, while maintaining a competitive speed. To achieve this, it is necessary to implement three ultrasonic sensors, one infrared sensor, two DC motors, a servo motor, a potentiometer, and all the necessary components to build the circuits, such as a breadboard, eight resistors, jumper wires, and eight transistors. An Arduino microcontroller is also necessary to read the outputs from the sensors. Lastly, batteries are necessary for the system to work. Eight AA Lithium batteries to power the two DC motors, and six AA Lithium batteries to power the servo and the Arduino, providing 9V.

I. Breadboard

A breadboard is a reusable platform for prototyping electronic circuits without soldering. It consists of a plastic base with a grid of holes, where metal strips underneath establish electrical connections [1]. Breadboards have two main sections: power rails for voltage and ground distribution, and terminal strips for placing components. The terminal area is split by a central gap to fit integrated circuits. Figure 1 shows a typical breadboard layout.

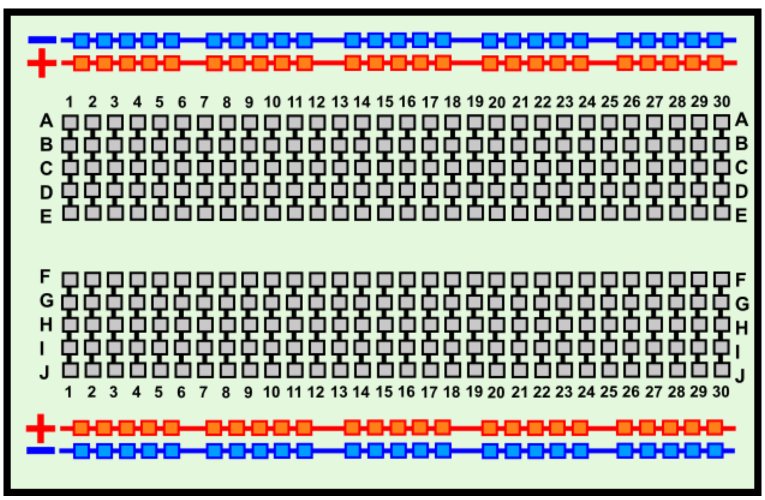



**Figure 1.** Breadboard [2]

II. Electrical Resistor

A resistor is a passive electrical component that limits current flow in a circuit by converting electrical energy into heat [1]. It follows Ohm's Law, where voltage across the resistor is proportional to the current passing through it. Resistors come in various types, such as fixed, variable, and thermistors, each suited for different applications. Their resistance is measured in ohms (Ω) and is determined by material composition and dimensions. Figure 2 shows a common axial-lead resistor.

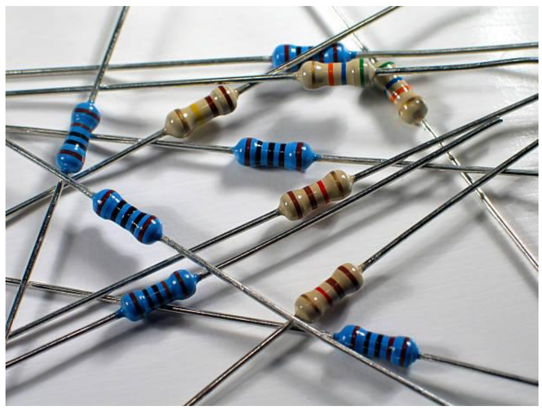


**Figure 2.** Axial-lead Resistor [3]

III. Jumper Wires

Jumper wires are cables with a connector at both ends, used for connecting components in a breadboard without the need to weld [4].

**Figure 3.** Jumper Wires [5]

IV. Transistors

A transistor is a semiconductor used to regulate or control the flow of voltage or current. The main types of transistors are the Field Effect Transistor (FET) and the Bipolar Junction Transistor (BJT). A BJT is the one used for this project. This transistor has three terminals, they are Collector, Base, and Emitter. Within the BJT, there are two types, the NPN and the PNP. An NPN transistor is normally open, and a PNP transistor is normally closed. In this case, the NPN and PNP were both used. An NPN transistor produces current flow from the Collector to the Emitter, while in a PNP transistor the current flows from Emitter to Collector [6].

**Figure 4.** Transistor [7]

V. Batteries

A battery is a device that converts the chemical energy stored into electrical energy. The flow of electrons from one material to another is what provides the electrical current. The potential difference between the ends of the battery determines the amount of voltage the battery will provide. A battery transfers more energy the higher the voltage is, this is due to the fact that more current flows around the circuit. There are different types of batteries; to select the correct one for the application a number of factors must be taken into consideration, such as the voltage and the capacity [8].


**Figure 5.** Batteries [9]

VI. Pulse Width Modulation

Pulse Width Modulation (PWM) is a technique used to obtain an analog signal while using a digital signal. This can be achieved by alternating the signal between a high and a low state. In an Arduino, it is necessary to use the function analogWrite() to control the duty cycle of the signal [10].

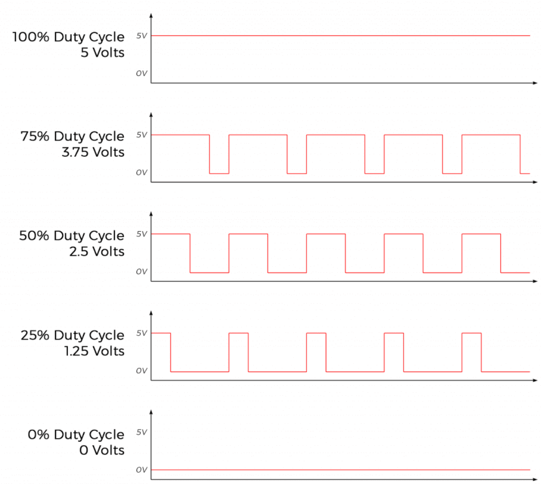



**Figure 6.** Duty Cycle Using PWM Function [11]

VII. Servo Motors

A servo motor is a motor designed to rotate with a high level of precision. They are commonly used to rotate an object at a specific angle. Inside, they contain an arrangement of gears that allows a high torque [12]. The function of the servo motor for this project is to control the steering of the car.


**Figure 7.** Servo Motor [13]

VIII. Direct Current Motors

As its name suggests, this motor works with direct current, turning the electrical energy from the current to mechanical energy. These motors use the magnetic fields that occur as a result of the electrical currents that are generated, powering the movement of a rotor located in the shaft [14].

The DC motors will enable the car to move, their speed and direction (forward, or backwards) is controlled with the help of an H-bridge.

**Figure 8.** DC Motor [15]

IX. Potentiometer

A potentiometer is a variable resistor that allows for adjustable voltage division in a circuit [16]. It consists of a resistive track and a movable wiper that changes resistance as it rotates or slides. Potentiometers are commonly used for tuning and control applications, such as volume knobs in audio devices and sensor calibration. Figure 9 shows a typical rotary potentiometer. By adjusting the wiper position, the output voltage can be varied, making potentiometers essential in circuits requiring precise control of electrical signals.

In this case, the potentiometer will be used to find the center position of the servo motor.

**Figure 9.** Potentiometer [17]

X. Arduino Controller

An Arduino controller is a microcontroller board used for prototyping and automation. It can power and control components on a breadboard by supplying 5V or 3.3V through its power pins. The board features multiple input/output (I/O) pins, allowing it to interact with sensors, actuators, and other circuit elements. Figure 10 shows a typical Arduino board. By connecting the Arduino's power and ground pins to the breadboard's rails, it can serve as a voltage source while also executing programmed instructions for circuit control.

In this case, the Arduino's main function is to obtain data from the sensors and control the two DC motors (move forward, or backwards) and the servo motor for steering.

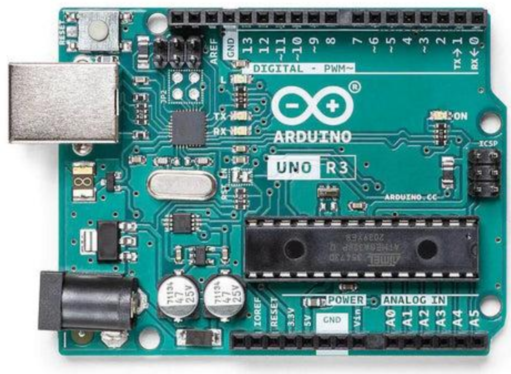



**Figure 10.** Arduino Microcontroller [18]

XI. Ultrasonic Sensor

An ultrasonic sensor calculates distance by emitting sound waves and receiving the echo. These sensors are less susceptible to interferences, such as gases, or smoke [19]. The ultrasonic sensors measure the time it takes for the echo to return after it reflected on an obstacle. This time (microseconds) is then converted to distance, in this case, to centimeters. This was achieved by performing a calibration for the sensor, which will be explained in detail in Part V.


**Figure 11.** Ultrasonic Sensor [20]

XII. Infrared Sensor

Infrared sensors emit light in order to determine if there is an object. This is possible due to the fact that these sensors measure the heat of the object. It has an emmiter and a detector. The emmiter is an IR Light Emitting Diode (LED), and the detector is an IR photodiode [21].

In this project, the infrared sensor is placed at the rear of the car. Its function is to detect if there is an obstacle at the back of the car. If it detecs an obstacle, the motors stop. This is implemented in case that when the car is reversing there is an object too close, making the motors stop and avoiding a crash.

**Figure 12.** Infrared Sensor [22]

## EXPERIMENTAL METHODS

The first step was to select and acquire all the necessary components to successfully build the autonomous race car. After this, it was necessary to design a chassis for the car. The CAD of the chassis was done using SolidWorks, after this, it was 3D printed. Figures 13-16 show the CAD of the car. This design included a mechanism which attached to the servo motor and the wheels will steer the car.

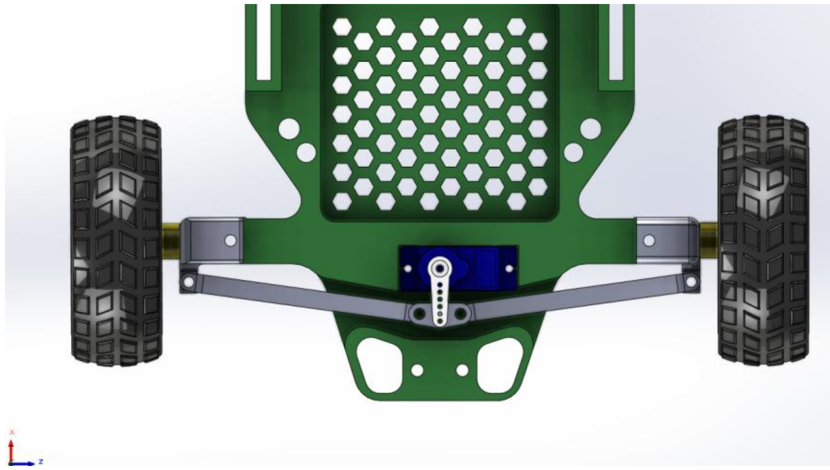


**Figure 13**. Car steering system

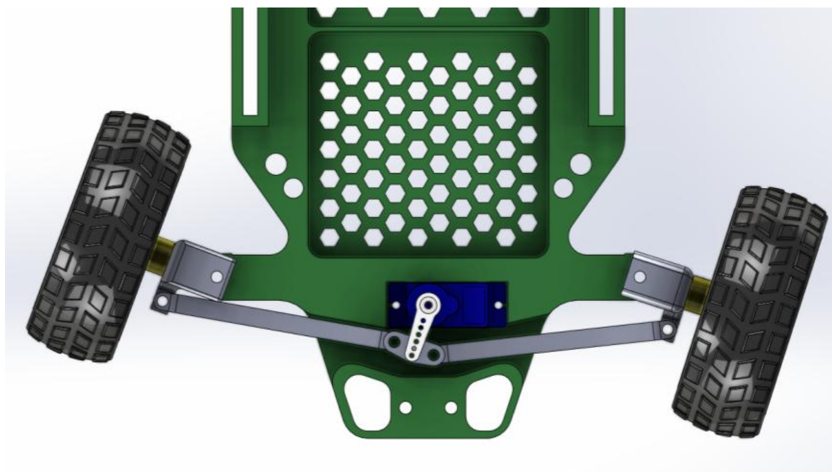


**Figure 14**. Car steering at maximum angle

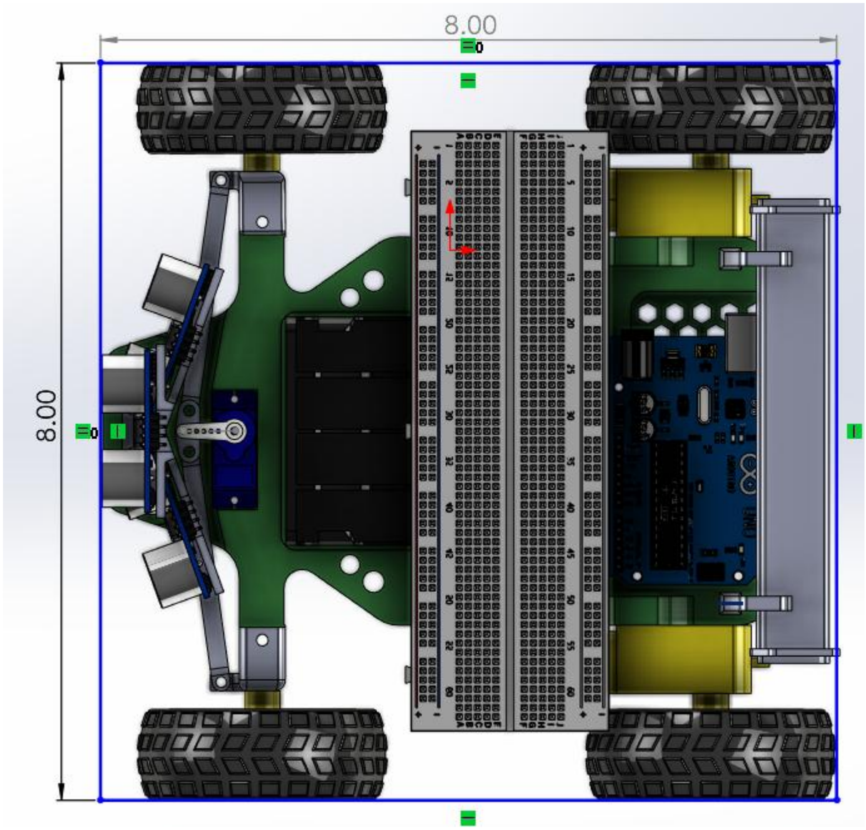



**Figure 15.** CAD of Car Showing Measurement of the Whole System

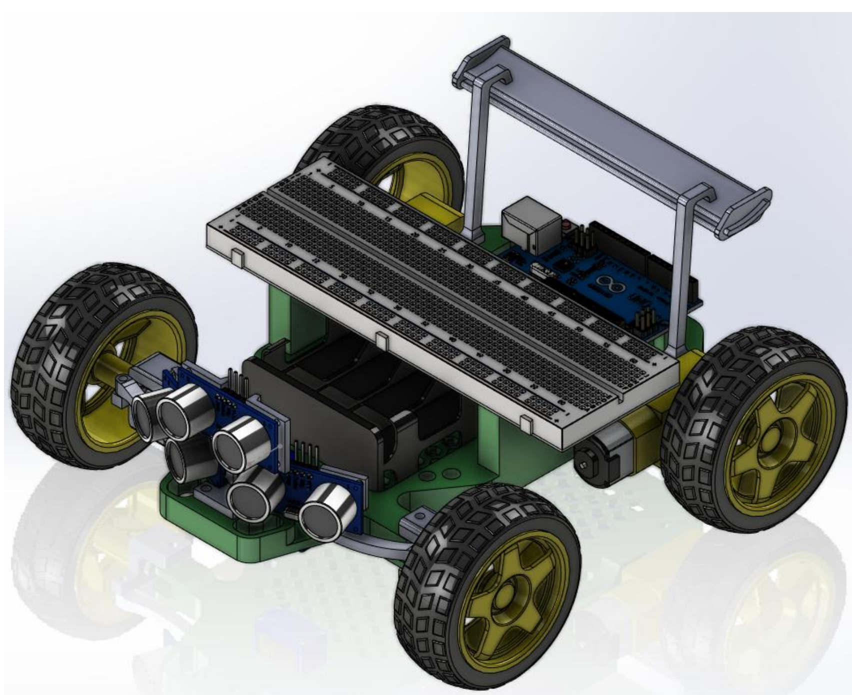


**Figure 16.** Final CAD of Race Car

The second step was to build two H-bridges to control the speed and direction (forward or backwards) of the two DC motors. Each H-bridge consists of four transistors (two TIP120 NPN and two TIP125 PNP), four resistors of 1kΩ, multiple jumper wires, and 12V power supply. To build the first H-bridge the two TIP120 NPN were placed at the top and the two TIP125 PNP were placed at the bottom. For the normally closed transistors, the bases were connected to resistors and microcontroller digital pins, the collectors were attached to the motor terminals, and the emitters were linked to the power supply. For the normally open transistors, the bases were connected to the same microcontroller pins as the normally closed transistors, the collectors were connected to the same motor terminals, and the emitters were connected to ground. This configuration allows the current to flow in alternate paths, making possible for the motors to move forward and backwards. This is controlled with the microcontroller by switching the rotation of the motor depending on the data acquired by the sensors. Figure 17 shows a schematic of the H-bridge. The selected pins for each motor were: digital pins 2 and 3 for the left motor, and digital pins 12 and 13 for the right motor.

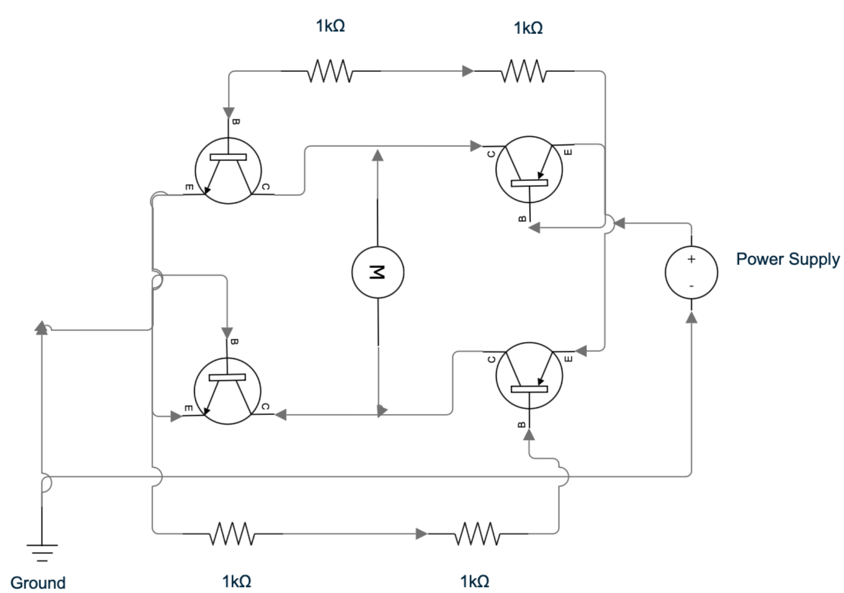



**Figure 17.** Schematic of H-bridge

The third step was to connect the servo motor. This was achieved by connecting the brown wire to the ground, the red wire to the power supply, and the orange cable to digital pin 6. The servo motor shares its power supply with the Arduino. It was also necessary to find the positions at 0°, 90°, and 180° of the servo motor. Figure 18 shows a schematic of the connections for the servo motor.

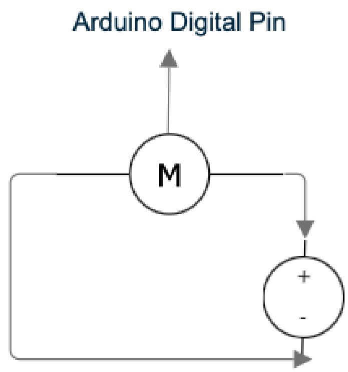



**Figure 18.** Schematic of Servo Motor

The fourth step was to connect the potentiometer. This was done by connecting the potentiometer's first lead to the power common node and the second lead to the common ground. The wiper lead was connected to analog pin A0. Figure 19 shows the schematic for the connection of the potentiometer.

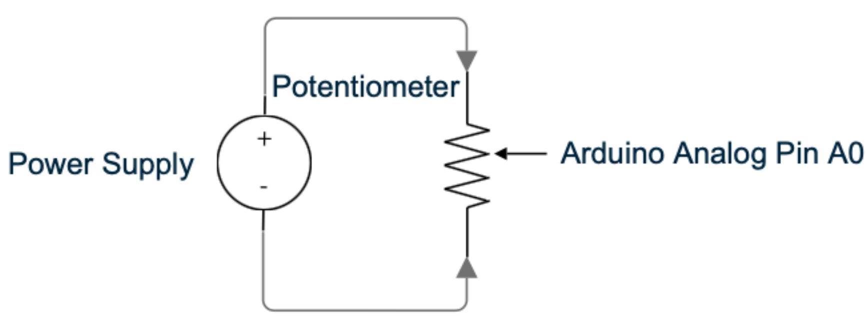



**Figure 19.** Schematic of Potentiometer

The fifth step was to connect the three ultrasonic sensors, which will detect the wooden obstacles placed in the track, which is in the first floor hall of Luchetti Building. Four jumper wires were used for each sensor. The red wires were connected to the common positive node, the green wires were connected to the common negative node, the white wires (trigger) were connected to the same node and then another cable was used to connect them to the microcontroller to the digital pin 8, and the blue wires (echo) were connected to different nodes and then connected to the microcontroller to the digital pins 7, 9, and 11. Each sensor was then calibrated (refer to Part V for details on the calibration). Figure 20 shows the schematic for the ultrasonic sensors.

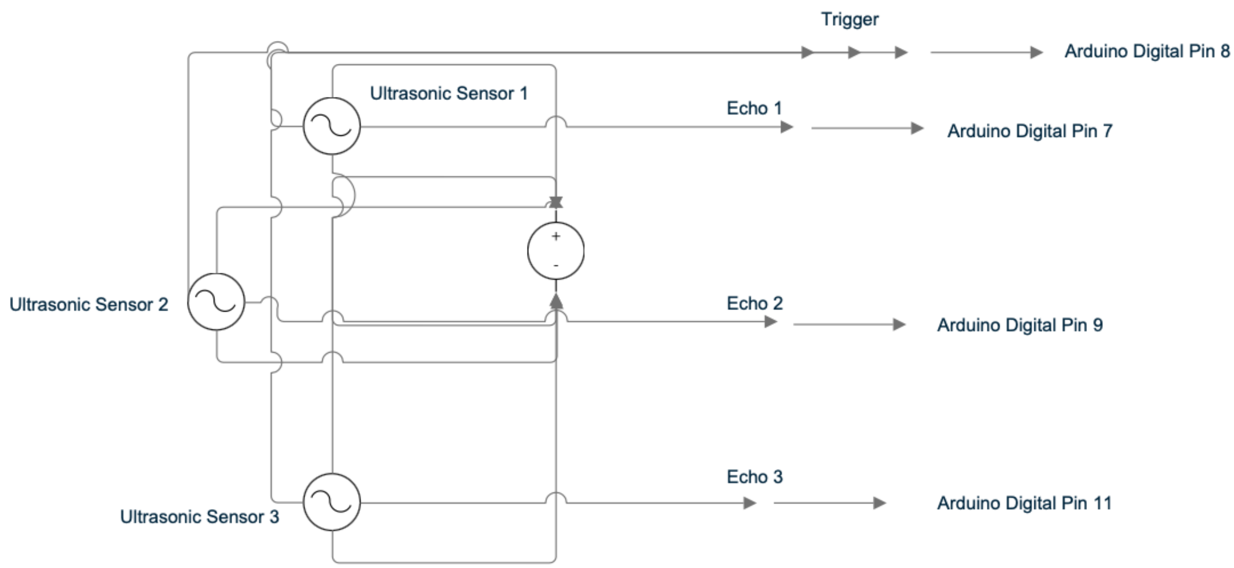



**Figure 20.** Schematic of Ultrasonic Sensors

Lastly, the infrared sensor was connected. Three jumper wires were used to connect the sensor. The first cable was connected to the common positive node, the second cable was connected to the negative common node, and the last cable was connected to the analog pin A2. Figure 21 shows the schematic for the infrared sensor.

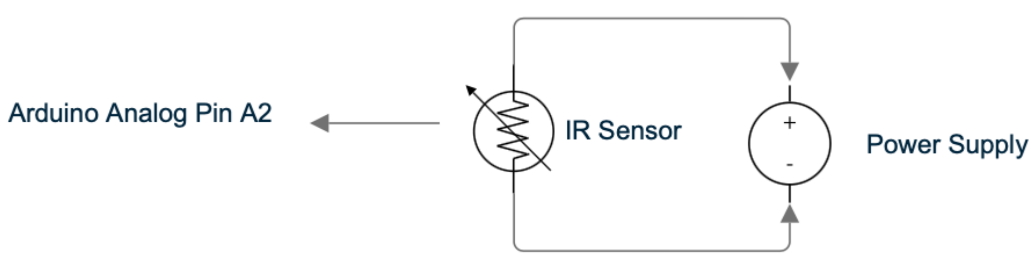



**Figure 21.** Schematic of Infrared Sensor

After realizing all the connections and the necessary calibrations, the complete schematic is shown in Figure 22, and the final product is shown in Figure 23.

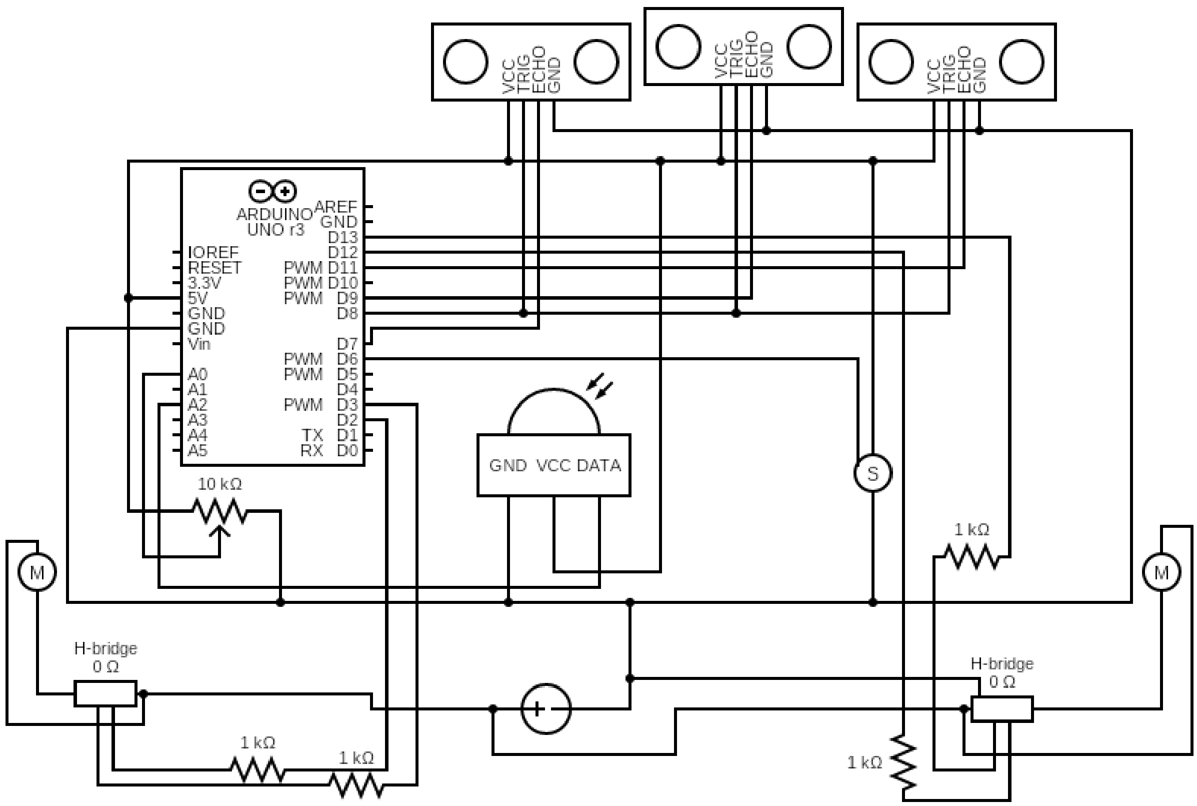



**Figure 22.** Schematic of Autonomous Race Car Connections

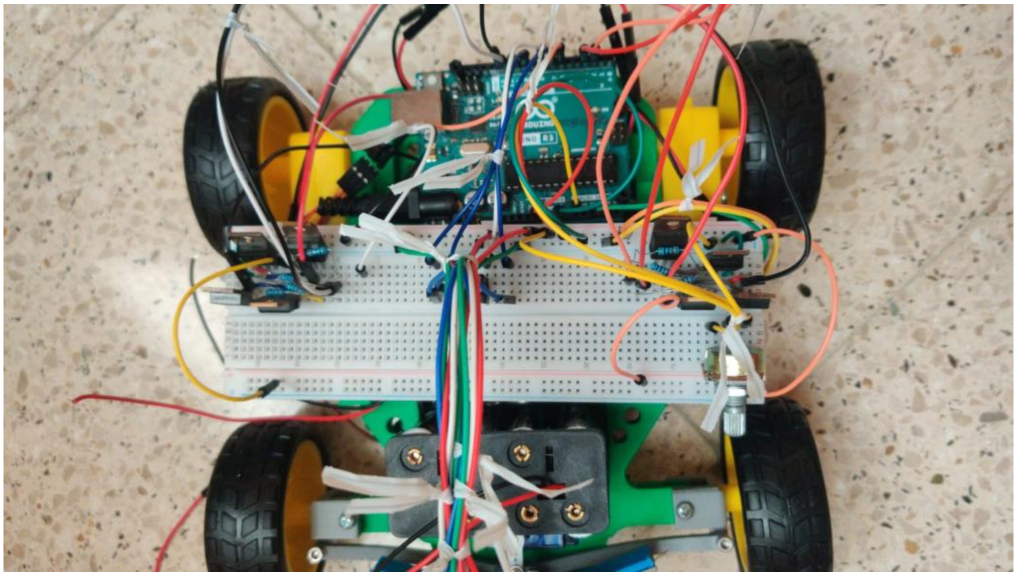


**Figure 23.** Autonomous Race Car Connections

## RESULTS & DISCUSSION

### PART A. CALIBRATION

For the ultrasonic sensors, the standard used for calibration was a known distance measured with a caliper. The process for the calibration of these sensors was done separately for each sensor. The process began by making a simple code to obtain the readings from the sensors. Then, with a caliper a distance was measured, and an object was placed. The values measured with the caliper and the ones provided by the code were annotated, and using excel a plot was developed. Then, with the trendline obtained from the plot, it was possible to obtain the equation to convert the microseconds that the sensor reads, to a distance (centimeters). With this equation in the Arduino code, it will be possible to determine the distance of an object and use this to determine to which side the car should steer.

**Table 1.** Calibration for Ultrasonic Sensor 1

| **Calibration Sensor for Ultrasonic Sensor 1 (Left)** | |
|---|---|
| **Real Measurements [cm]** | **Code Readings [μs]** |
| 2 | 174 |
| 3 | 204 |
| 4 | 222 |
| 5 | 270 |
| 6 | 351 |
| 7 | 422 |
| 8 | 479 |

**Table 2.** Calibration for Ultrasonic Sensor 2

| **Calibration Sensor for Ultrasonic Sensor 2 (Center)** | |
|---|---|
| **Real Measurements [cm]** | **Code Readings [μs]** |
| **2** | 144 |
| **3** | 160 |
| **4** | 212 |
| **5** | 258 |
| **6** | 344 |
| **7** | 382 |
| **8** | 401 |

**Table 3.** Calibration for Ultrasonic Sensor 3

| Calibration Sensor for Ultrasonic Sensor 3 (Right) | |
|---|---|
| **Real Measurements [cm]** | **Code Readings [μs]** |
| **2** | 136 |
| **3** | 155 |
| **4** | 212 |
| **5** | 263 |
| **6** | 345 |
| **7** | 383 |
| **8** | 402 |

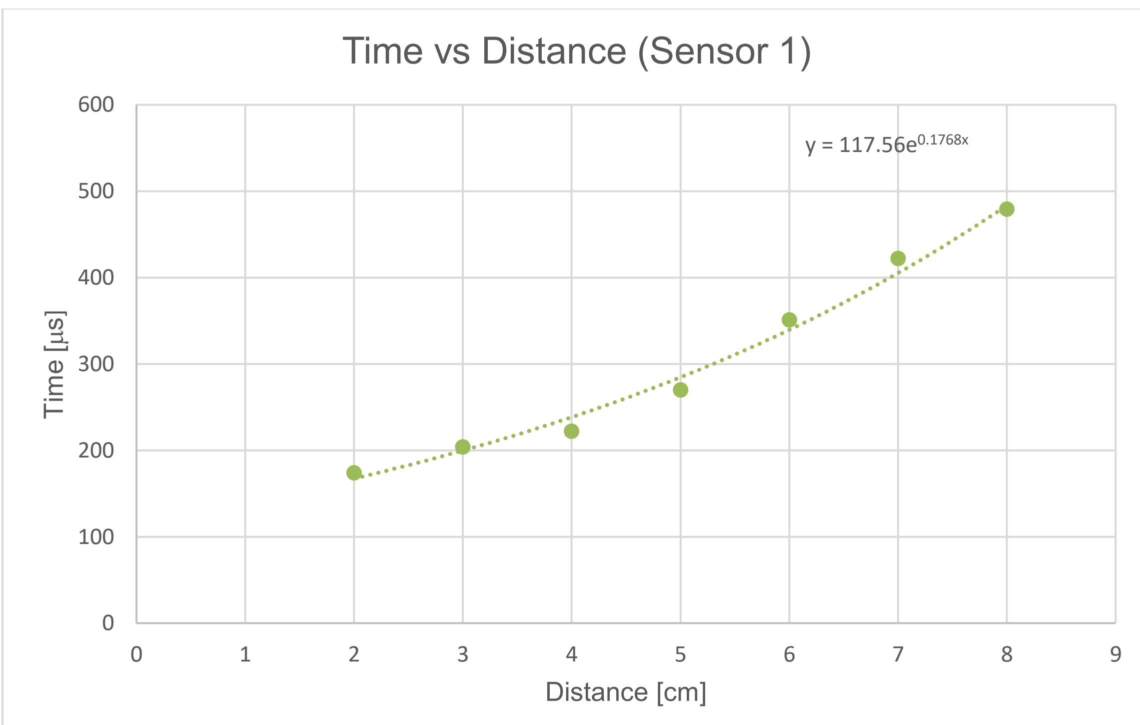



**Figure 24.** Time (μs) vs Distance (cm) for Sensor 1

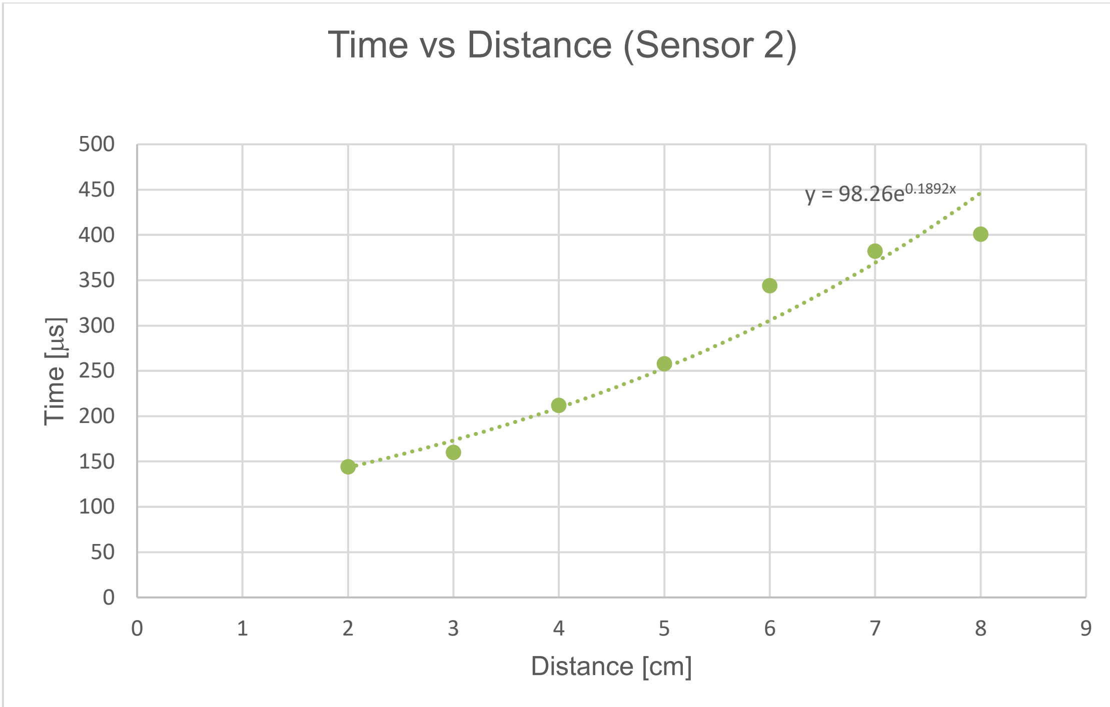



**Figure 25.** Time (µs) vs Distance (cm) for Sensor 2

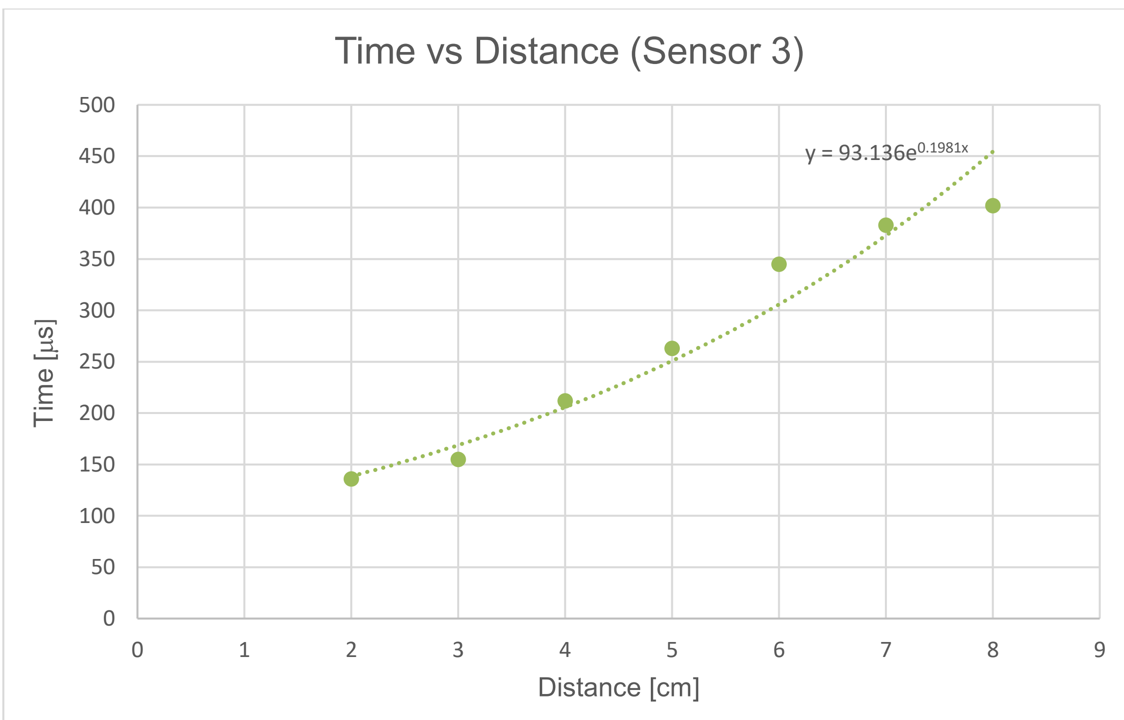



**Figure 26.** Time (µs) vs Distance (cm) for Sensor 3

After obtaining the trendline for the three ultrasonic sensors, finding the inverse and dividing by two, the equations obtained are presented in Table 4 and an example of the calculation is shown below.

$$Calibration\ Sensor\ 1 = \frac{117.56^{-1}}{2}$$

$$Calibration\ Sensor\ 1 = 0.004253$$

**Table 4.** Calibration Equations for Ultrasonic Sensors

| **Calibration Equations for Ultrasonic Sensors** | |
|---|---|
| Sensor 1 (Left) | 0.004253 |
| Sensor 2 (Center) | 0.005089 |
| Sensor 3 (Right) | 0.005368 |
| Average | 0.004903 |

An average of the values was calculated, resulting in 0.004903. This value was then placed in the code to convert the readings of the ultrasonic sensors from microseconds to centimeters.

A linearity error analysis was performed to quantify the deviation between the sensor readings and the real values. To achieve this, it was necessary to use the Method of Least Squares. The results for the Method of Least Squares and the linearity error are shown in Tables 5-13 and an example of the calculations are shown below.

**Table 5.** Method of Least Squares for Sensor 1

| Method of Least Squares Sensor 1 | | | |
|---|---|---|---|
| **x** | **y** | **xy** | **$x^2$** |
| 174 | 2 | 348 | 30276 |
| 204 | 3 | 612 | 41616 |
| 222 | 4 | 888 | 49284 |
| 270 | 5 | 1350 | 72900 |
| 351 | 6 | 2106 | 123201 |
| 422 | 7 | 2954 | 178084 |
| 479 | 8 | 3832 | 229441 |
| **Sum** | | | |
| 2122 | 35 | 12090 | 724802 |
| **Sum of x squared** | | | |
| 4502884 | | | |

$$a_0 = \frac{\sum x_i \sum x_i y_i - \sum x_i^2 \sum y_i}{(\sum x_i)^2 - N \sum x_i^2}$$

$$a_0 = \frac{(2122 \times 12{,}090) - (724{,}802 \times 35)}{4502884 - (7 \times 724{,}802)}$$

$$a_0 = -0.5027$$

$$a_1 = \frac{\sum x_i \sum y_i - N \sum x_i y_i}{(\sum x_i)^2 - N \sum x_i^2}$$

$$a_1 = \frac{(2122 \times 35) - 7(12{,}090)}{4502884 - (7 \times 724{,}802)}$$

$$a_1 = 0.0181$$

$$y_L = a_0 + a_1 x$$

$$y_{L\_s1} = -0.5027 + 0.0181x$$

**Table 6.** Values for $y_L$ for Sensor 1

| $y_L$ |
|---|
| 2.65577418 |
| 3.20033992 |
| 3.52707935 |
| 4.39838453 |
| 5.868712 |
| 7.15751757 |
| 8.19219246 |

**Table 7.** Values for Linearity Error Sensor 1

| Linearity Error | | |
|---|---|---|
| $y_L$ | $y_{exp}$ | $e_L$ |
| 2.65577418 | 2 | 0.65577418 |
| 3.20033992 | 3 | 0.20033992 |
| 3.52707935 | 4 | 0.47292065 |
| 4.39838453 | 5 | 0.60161547 |
| 5.868712 | 6 | 0.131288 |
| 7.15751757 | 7 | 0.15751757 |
| 8.19219246 | 8 | 0.19219246 |
| $\%e_L$ | | |
| **10.92956973** | | |

$$\%e_L = \frac{e_{L,max}}{y_{max} - y_{min}} \times 100\%$$

$$\%e_L = \frac{0.6557}{8 - 2} \times 100\%$$

$$\%e_L = 10.92\%$$

**Table 8.** Method of Least Squares for Sensor 2

| Method of Least Squares Sensor 2 | | | |
|---|---|---|---|
| x | y | xy | $x^2$ |
| 144 | 2 | 288 | 20736 |
| 160 | 3 | 480 | 25600 |
| 212 | 4 | 848 | 44944 |
| 258 | 5 | 1290 | 66564 |
| 344 | 6 | 2064 | 118336 |
| 382 | 7 | 2674 | 145924 |
| 401 | 8 | 3208 | 160801 |
| **Sum** | | | |
| 1901 | 35 | 10852 | 582905 |
| **Sum of x squared** | | | |
| 3613801 | | | |

$$y_L = a_0 + a_1 x$$

$$y_{L\_s2} = -0.4886 + 0.0202x$$

**Table 9.** Values for $y_L$ for Sensor 2

| $y_L$ |
|---|
| 2.42168631 |
| 2.74505824 |
| 3.79601701 |
| 4.72571131 |
| 6.46383543 |
| 7.23184377 |
| 7.61584793 |

**Table 10.** Values for Linearity Error Sensor 2

| Linearity Error | | |
|---|---|---|
| $y_L$ | $y_{exp}$ | $e_L$ |
| 2.42168631 | 2 | 0.42168631 |
| 2.74505824 | 3 | 0.25494176 |
| 3.79601701 | 4 | 0.20398299 |
| 4.72571131 | 5 | 0.27428869 |
| 6.46383543 | 6 | 0.46383543 |
| 7.23184377 | 7 | 0.23184377 |
| 7.61584793 | 8 | 0.38415207 |
| $\%e_L$ | | |
| **7.730590554** | | |

**Table 11.** Method of Least Squares for Sensor 3

| Method of Least Squares Sensor 3 | | | |
|---|---|---|---|
| x | y | xy | $x^2$ |
| 136 | 2 | 272 | 18496 |
| 155 | 3 | 465 | 24025 |
| 212 | 4 | 848 | 44944 |
| 263 | 5 | 1315 | 69169 |
| 345 | 6 | 2070 | 119025 |
| 383 | 7 | 2681 | 146689 |
| 402 | 8 | 3216 | 161604 |
| **Sum** | | | |
| **1896** | 35 | 10867 | 583952 |
| **Sum of x squared** | | | |
| **3594816** | | | |

$$y_L = a_0 + a_1 x$$

$$y_{L\_s3} = -0.3358 + 0.0197x$$

**Table 12.** Values for $y_L$ for Sensor 3

| $y_L$ |
|---|
| 2.34334318 |
| 2.71763911 |
| 3.8405269 |
| 4.84521597 |
| 6.4605984 |
| 7.20919026 |
| 7.58348619 |

**Table 13.** Values for Linearity Error Sensor 3

| Linearity Error | | |
|---|---|---|
| $y_L$ | $y_{exp}$ | $e_L$ |
| 2.34334318 | 2 | 0.34334318 |
| 2.71763911 | 3 | 0.28236089 |
| 3.8405269 | 4 | 0.1594731 |
| 4.84521597 | 5 | 0.15478403 |
| 6.4605984 | 6 | 0.4605984 |
| 7.20919026 | 7 | 0.20919026 |
| 7.58348619 | 8 | 0.41651381 |
| $\%e_L$ | | |
| **7.676639992** | | |

The results obtained for the linearity error were 10.92% for sensor 1(left), 7.73% for sensor 2 (center), and 7.67% for sensor 3 (right). For a low-cost sensor as HC-SR04, these values are within the acceptable range. These sensors, paired with a code that takes this into consideration, are ideal for the purpose of this project, which is to avoid objects while maintaining a competitive speed.

Some of the possible limitations, that could have impacted the results were that the ultrasonic sensors can produce inconsistent readings due to the reflectivity of the surface and enviromental noises. Another factor could be the smoothness or roughness of the floor. When testing the car in a smooth floor, this caused the wheels to slip, due to a reduced traction. On the other hand, on a rougher surface the car had better traction.

## PART B. BATTERIES

The selection of the batteries was an important step, due to the fact that if the wrong ones were chosen, it will be difficult for the system to work. The first batteries chosen were AA Alkaline. During the trials performed, it was noticed that the batteries discharged too fast, and that one of the transistors got hot. After researching, it was concluded that the batteries needed were AA Lithium. These batteries have a lower resistance, and weight less. After this change, it was noticed that the transistor did not get too hot and that they lasted longer. Also, at first, the Arduino was powered by a 9V battery, but it was noticed that it discharged too quickly. This led to deciding it was a better option for the servo motor and the microcontroller to share the power supply.  The distribution of the batteries in the system is shown in Table 14.

**Table 14.** Battery Distribution for System

| Battery Distribution | | | |
|---|---|---|---|
| **Component** | **Voltage Required [V]** | **Voltage Supplied [V]** | **Batteries** |
| DC Motors | 6 V | 12V | 8 AA Lithium |
| Servo Motor | 4.8V-6V | 9V | 6 AA Lithium |
| Arduino Microcontroller | 2.7V-5.5V | 9V | 6 AA Lithium |

## PART C. ARDUINO CODE

### Code Used to Calibrate the Three Ultrasonic Sensors and the Infrared Sensor

The code used to calibrate the three ultrasonic sensors, and the infrared sensor works by acquiring data from the sensors. The *readUltrasonic()* function sends a trigger signal to the sensor and measures the time it takes for the echo to return. It returns a value in microseconds. For the infrared sensor, it will return a value of 4 if there isn't  a close object. On the contrary, if it does detect an object, it will return a 0.

```
// Pin assignments
const int trigPin = 8;
const int echoLeft = 7;
```

```
const int echoCenter = 9;
const int echoRight = 11;
const int irPin = A2;

void setup() {
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoLeft, INPUT);
  pinMode(echoCenter, INPUT);
  pinMode(echoRight, INPUT);
  pinMode(irPin, INPUT);

  // Print header
  Serial.println("Left(cm),Center(cm),Right(cm),IR_Value");
}
void loop() {
  long distanceLeft = readUltrasonic(echoLeft);
  delay(50);
  long distanceCenter = readUltrasonic(echoCenter);
  delay(50);
  long distanceRight = readUltrasonic(echoRight);
  delay(50);

  int irVal = (analogRead(irPin)* 5.0)/1023.0;

  // Print data
  Serial.print(distanceLeft);
  Serial.print(",");
  Serial.print(distanceCenter);
  Serial.print(",");
  Serial.print(distanceRight);
  Serial.print(",");
  Serial.println(irVal);

  delay(200); // Wait before next reading
}

long readUltrasonic(int echoPin) {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  long duration = pulseIn(echoPin, HIGH, 30000); // 30 ms
  long distance = duration;
  return distance;
}
```

## Main Code for the Autonomous Race Car

This code sends a pulse through the trigger which will be received by the respective echo of each sensor, and it will convert it to a distance in centimeters. These readings will help the racing car to steer away from the objects it encounters. If one of the ultrasonic sensors detects an object too close, the car will reverse. If the car reverses three times within a time window of 10 seconds, it will perform a longer reverse. With the help of a potentiometer, it will be possible to set the center position of the servo motor. Additionally, it has an infrared sensor at the back of the car that will act as a safety stop, meaning that if it detects an object is very close, it will stop the motors.

```
// Pin definitions
const int trigPin = 8;
const int echoLeft = 7;
const int echoCenter = 9;
const int echoRight = 11;

const int motorLeft1 = 2;
const int motorLeft2 = 3;
const int motorRight1 = 12;
const int motorRight2 = 13;

const int servoPin = 6;
const int potentiometerPin = A0;
const int irPin = A2;

// Steering and timing constants
const int steer_threshold = 50;    //if object is within 50 cm, it will start steering
const int reverse_threshold = 7;     // if object is within 7 cm, it will trigger reverse
const int reverse_delay = 1500;  // car will reverse for 1500 ms
const int long_reverse_delay = 2000;  // car will reverse for 2000 ms when stuck
const int stuck_time = 10000; // it will detect the car is stuck after a 10,000 ms window
const int stuck_count = 3;     // if it reverses 3 times within the stuck window, it will do a long reverse
const int assist_pulse_ms = 20;    // motor-pulse assist duration
const int turn_offset = 700;   // microseconds offset from center for turns
const int min_pulse = 1000;  // limits
const int max_pulse = 2000;

unsigned long reverseTimes[stuck_count];
int reverseIdx = 0;
void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoLeft, INPUT);
```

```
  pinMode(echoCenter, INPUT);
  pinMode(echoRight, INPUT);

  pinMode(motorLeft1, OUTPUT);
  pinMode(motorLeft2, OUTPUT);
  pinMode(motorRight1, OUTPUT);
  pinMode(motorRight2, OUTPUT);

  pinMode(servoPin, OUTPUT);
  pinMode(irPin, INPUT);

  Serial.begin(9600);
}

// Read one ultrasonic sensor (shared trigger)
long readUltra(int echoPin) {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delay(60);
  digitalWrite(trigPin, LOW);
  long d = pulseIn(echoPin, HIGH, 30000);
  return d ? (d * 0.004903 / 2) : 999;
}

void moveForward() {
  digitalWrite(motorLeft1, HIGH);
  digitalWrite(motorLeft2, LOW);
  digitalWrite(motorRight1, HIGH);
  digitalWrite(motorRight2, LOW);
}
void moveBackward() {
  digitalWrite(motorLeft1, LOW);
  digitalWrite(motorLeft2, HIGH);
  digitalWrite(motorRight1, LOW);
  digitalWrite(motorRight2, HIGH);
}
void stopMotors() {
  digitalWrite(motorLeft1, LOW);
  digitalWrite(motorLeft2, LOW);
  digitalWrite(motorRight1, LOW);
  digitalWrite(motorRight2, LOW);
}

// Briefly pulse off the left motor to assist turning right
void pulseLeftAssist() {
  digitalWrite(motorLeft1, LOW);
  digitalWrite(motorLeft2, LOW);
```

```
  delay(assist_pulse_ms);

  digitalWrite(motorLeft1, HIGH);
  digitalWrite(motorLeft2, LOW);
}
// Briefly pulse off the right motor to assist turning left
void pulseRightAssist() {
  digitalWrite(motorRight1, LOW);
  digitalWrite(motorRight2, LOW);
  delay(assist_pulse_ms);

  digitalWrite(motorRight1, HIGH);
  digitalWrite(motorRight2, LOW);
}

// Record a reverse timestamp for stuck detection
void logReverse() {
  reverseTimes[reverseIdx] = millis();
  reverseIdx = (reverseIdx + 1) % stuck_count;
}
// Returns true if we've reversed stuck_count times within stuck_window
bool isStuck() {
  unsigned long now = millis();
  for (int i = 0; i < stuck_count; i++) {
    if (now - reverseTimes[i] > stuck_time) return false;
  }
  return true;
}

// Send one servo PWM pulse
void servoPulse(int pulseUs) {
  digitalWrite(servoPin, HIGH);
  delayMicroseconds(pulseUs);
  digitalWrite(servoPin, LOW);
  delay(20);
}

// Steering based on left/right distances
void steer(int distanceLeft, int distanceRight) {
  // Read pot to set center
  int potentiometer = analogRead(potentiometerPin);
  int center = map(potentiometer, 0, 1023, min_pulse, max_pulse);

  int pulse = center;
  // If left obstacle, turn away to right
  if (distanceLeft < steer_threshold && distanceLeft < distanceRight) {
    pulse = constrain(center - turn_offset, min_pulse, max_pulse);
    pulseRightAssist();
```

```
  }
  // If right obstacle, turn away to left
  else if (distanceRight < steer_threshold && distanceRight < distanceLeft) {
    pulse = constrain(center + turn_offset, min_pulse, max_pulse);
    pulseLeftAssist();
  }

  servoPulse(pulse);
}

void loop() {
  long distanceLeft = readUltra(echoLeft);
  delay(50);
  long distanceCenter = readUltra(echoCenter);
  delay(50);
  long distanceRight = readUltra(echoRight);
  delay(50);

  bool ir_sensor = (digitalRead(irPin) == LOW);

  Serial.print("Left:"); Serial.print(distanceLeft);
  Serial.print(" Center:"); Serial.print(distanceCenter);
  Serial.print(" Right:"); Serial.print(distanceRight);
  Serial.print(" IR:"); Serial.println(ir_sensor);
  delay(50);

  // IR sensor safety stop
  if (ir_sensor) {
    stopMotors();
    return;
  }

  // if any sensor is too close to an object it will reverse
  if (distanceLeft < reverse_threshold || distanceCenter < reverse_threshold || distanceRight < reverse_threshold) {
    logReverse();
    moveBackward();
    delay(reverse_delay);

    // If the car gets stuck, it will do a longer reverse
    if (isStuck()) {
      moveBackward();
      delay(long_reverse_delay);
      // clear history
      for (int i = 0; i < stuck_count; i++) reverseTimes[i] = 0;
    }

    stopMotors();
    delay(100);
```

```
    return;
  }

  // forward and steering
  steer(distanceLeft, distanceRight);
  moveForward();
}
```

## Conclusion

The objective of this project was to build an autonomous race car. This was achieved by creating the chasis of the car in SolidWorks, 3D printing it, and assembling two H-bridges to contol the motors, a servo motor to control the steering, three ultrasonic sensors to detect objects, and an infrared sensor to act as a safety stop. For the sensors to be as accurate as possible, it was necessary to calibrate them and to perform a linearity error analysis. The resulting linearity errors were 10.92% for the left sensor, 7.73% for the center sensor, and 7.67% for the right sensor. This errors are considered acceptable for a low-cost sensor and with some implementations to the code, they serve their purpose.

These sensors allowed the car to detect close objects and adjust its steering away from them. It was also programmed to reverse in case its not able to swerve away from the object, and if the sensors detect that the car reversed three times in a specified window of time, it will do a longer reverse. Even though the objectives were met, there were some complications such as the inconsistent readings of the sensors at some points, traction issues due to a smooth floor, and the fact that when using Alkaline batteries, they discharged faster and made one of the transistors to heat. Despite these issues, the autonomous race car functioned as expected.

## References

[1] M. Banzi and M. Shiloh, Getting Started with Arduino: The Open Source Electronics Prototyping Platform, 2nd ed. Sebastopol, CA: O'Reilly Media, 2011.

[2] “Using the electrical breadboard - breadboard UK,” RUSH PCB Ltd, https://www.rushpcb.co.uk/using-the-electrical-breadboard/ (accessed May 10, 2025).

[3] “6,200+ electrical resistance stock photos, pictures & Royalty-free images,” iStock, https://www.istockphoto.com/photos/electrical-resistance (accessed May 10, 2025).

[4] “Jumper Wires – an introduction, applications, and working principles.,” Electronic, https://electronicssecret.wordpress.com/2022/07/14/jumper-wires-an-introduction-applications-and-working-principles/ (accessed May 10, 2025).

[5] “20cm male to female jumper cable wire for Arduino - 10pcs,” Flyrobo, https://www.flyrobo.in/10cm_male_to_female_jumper_cable_wire_for_arduino (accessed May 10, 2025).

[6] C. Hashemi-Pour and R. Awati, “What is a transistor and how does it work? Definition from TechTarget,” WhatIs, https://www.techtarget.com/whatis/definition/transistor (accessed May 10, 2025).

[7] “Low Voltage PNP Power Darlington Transistor (TIP127),” Voltaat, https://www.voltaat.com/products/low-voltage-pnp-power-darlington-transistor-tip127 (accessed May 10, 2025).

[8] Mischa, “How A battery works,” Australian Academy of Science, https://www.science.org.au/curious/technology-future/batteries (accessed May 10, 2025).

[9] P. McCord, Batteries, http://chembook.org/page-nonav.php?chnum=7&sect=9 (accessed May 10, 2025).

[10] “Pulse width modulation,” Pulse Width Modulation - an overview | ScienceDirect Topics, https://www.sciencedirect.com/topics/engineering/pulse-width-modulation#:~:text=Pulse%2Dwidth%20modulation%20is%20a,parameters—frequency%20and%20duty%20cycle. (accessed May 10, 2025).

[11] G. Lambert, “How to build a pulse width modulation signal generator,” Circuit Basics, https://www.circuitbasics.com/pulse-width-modulation/ (accessed May 10, 2025).

[12] “What is a servo motor? - understanding basics of servo motor working,” Servo Motor Basics, Working Principle & Theory, https://circuitdigest.com/article/servo-motor-working-and-basics (accessed May 10, 2025).

[13] “Servo motor interfacing with LPC2148,” ARM7-LPC2148, https://www.electronicwings.com/arm7/servo-motor-interfacing-with-lpc2148

(accessed May 10, 2025).

[14] “Everything you need to know about DC Motors,” RS, https://uk.rs-online.com/web/content/discovery/ideas-and-advice/dc-motors-guide (accessed May 10, 2025).

[15] “Hobby Motor - Mini DC Motor M260 1-3V - sparkfun Rob-11696,” BOTLAND, https://botland.store/dc-motors-without-a-gearbox/20804-hobby-motor-mini-dc-motor-m260-1-3v-sparkfun-rob-11696-5904422379063.html (accessed May 10, 2025).

[16] S. Monk, Programming Arduino: Getting Started with Sketches, 2nd ed. Sebastopol, CA: O'Reilly Media, 2016.

[17] “Potentiometer – 20K COMM23,” Faranux Electronics, https://www.faranux.com/product/potentiometer-20k/ (accessed May 10, 2025).

[18] “Microcontrolador USB Arduino Uno R3,” RobotShop Europe, https://eu.robotshop.com/es/products/microcontrolador-usb-arduino-uno-r3 (accessed May 10, 2025).

[19] “What is ultrasonic sensor: Working Principle & Applications,” Robocraze, https://robocraze.com/blogs/post/what-is-ultrasonic-sensor?srsltid=AfmBOoqFd8Ot9ZiLFYF9WaR55IInSA59MlLp5WU65BZ9D5915537uyoQ (accessed May 10, 2025).

[20] “Ultrasonic sensor (HC-SR04),” Voltaat, https://www.voltaat.com/products/ultrasonic-sensor-hc-sr04 (accessed May 10, 2025).

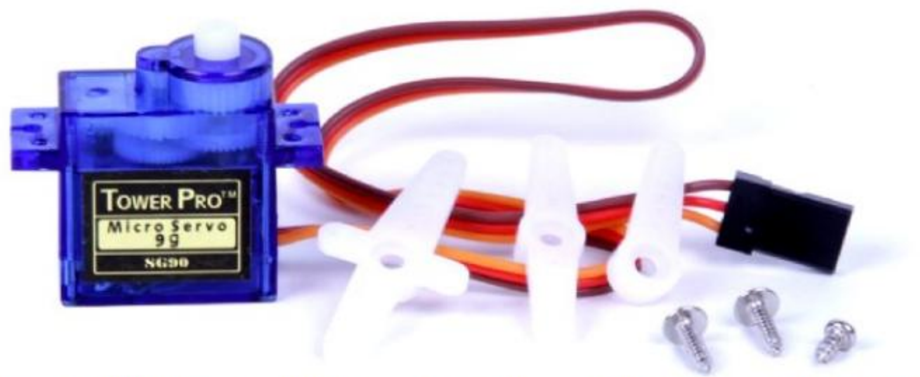
[21] Macfos, “IR sensor working principle and applications,” Robu.in | Indian Online Store | RC Hobby | Robotics, https://robu.in/ir-sensor-working/ (accessed May 10, 2025).

[22] “IR sensors: Schematic and priciple explained,” blikai, https://www.blikai.com/blog/components-parts/ir-sensors-schematic-and-priciple-explained (accessed May 10, 2025).

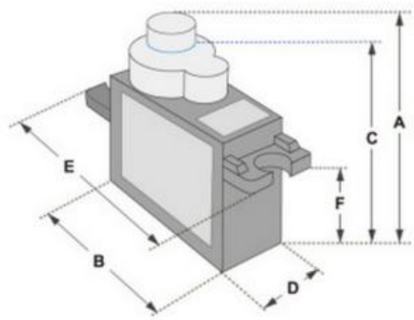
# APPENDIX

## SG90 SERVO  Motor Datasheet

### SERVO MOTOR SG90 DATA SHEET

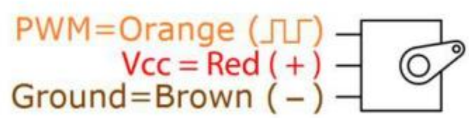


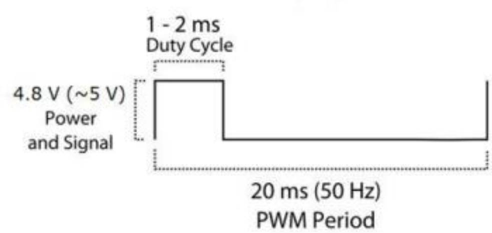

Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.




| Dimensions & Specifications |
| --- |
| A (mm) : 32 |
| B (mm) : 23 |
| C (mm) : 28.5 |
| D (mm) : 12 |
| E (mm) : 32 |
| F (mm) : 19.5 |
| Speed (sec) : 0.1 |
| Torque (kg-cm) : 2.5 |
| Weight (g) : 14.7 |
| Voltage : 4.8 - 6 |

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.




## ULTRASONIC SENSOR DATASHEET

# Ultrasonic Ranging Module HC - SR04

## Product features:

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:
(1) Using IO trigger for at least 10us high level signal,
(2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
(3) IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.
Test distance = (high level time×velocity of sound (340M/S) / 2,

## Wire connecting direct as following:

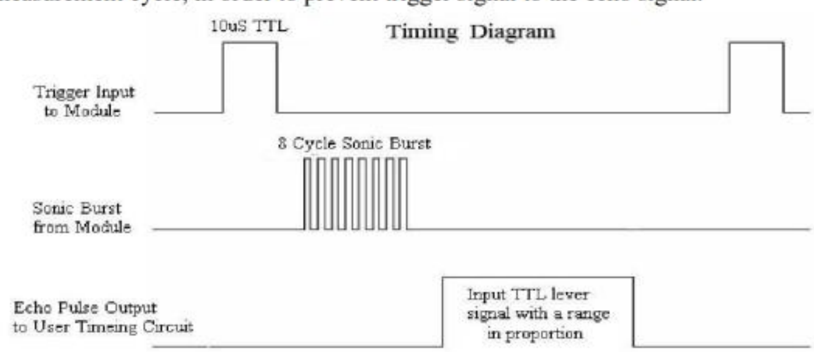
- 5V Supply
- Trigger Pulse Input
- Echo Pulse Output
- 0V Ground

## Electric Parameter

| | |
|---|---|
| **Working Voltage** | **DC 5 V** |
| **Working Current** | **15mA** |
| **Working Frequency** | **40Hz** |
| **Max Range** | **4m** |
| **Min Range** | **2cm** |
| **MeasuringAngle** | **15 degree** |
| **Trigger Input Signal** | **10uS TTL pulse** |
| **Echo Output Signal** | **Input TTL lever signal and the range in proportion** |
| **Dimension** | **45*20*15mm** |


## Timing diagram

The Timing diagram is shown below. You only need to supply a short 10uS pulse to the trigger input to start the ranging, and then the module will send out an 8 cycle burst of ultrasound at 40 kHz and raise its echo. The Echo is a distance object that is pulse width and the range in proportion .You can calculate the range through the time interval between sending trigger signal and receiving echo signal. Formula: uS / 58 = centimeters or uS / 148 =inch; or: the range = high level time * velocity (340M/S) / 2; we suggest to use over 60ms measurement cycle, in order to prevent trigger signal to the echo signal.



## Attention:

- The module is not suggested to connect directly to electric, if connected electric, the GND terminal should be connected the module first, otherwise, it will affect the normal work of the module.
- When tested objects, the range of area is not less than 0.5 square meters and the plane requests as smooth as possible, otherwise ,it will affect the results of measuring.

**www.Elecfreaks.com**

# TIP 120 TRANSISTOR DATASHEET

onsemi

DATA SHEET
www.onsemi.com

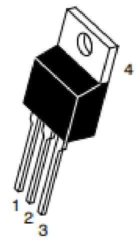
# Plastic Medium-Power Complementary Silicon Transistors

# TIP120, TIP121, TIP122 (NPN); TIP125, TIP126, TIP127 (PNP)

Designed for general−purpose amplifier and low−speed switching applications.

**Features**

- High DC Current Gain −
  $h_{FE}$ = 2500 (Typ) @ $I_C$
  = 4.0 Adc
- Collector−Emitter Sustaining Voltage − @ 100 mAdc
  $V_{CEO(sus)}$ = 60 Vdc (Min) − TIP120, TIP125
  = 80 Vdc (Min) − TIP121, TIP126
  = 100 Vdc (Min) − TIP122, TIP127
- Low Collector−Emitter Saturation Voltage −
  $V_{CE(sat)}$ = 2.0 Vdc (Max) @ $I_C$ = 3.0 Adc
  = 4.0 Vdc (Max) @ $I_C$ = 5.0 Adc
- Monolithic Construction with Built−In Base−Emitter Shunt Resistors
- Pb−Free Packages are Available*

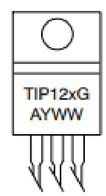



1. Base
2. Collector
3. Emitter
4. Collector

TO−220AB
CASE 221A
STYLE 1

**DARLINGTON**
**5 AMPERE**
**COMPLEMENTARY SILICON**
**POWER TRANSISTORS**
**60−80−100 VOLTS, 65 WATTS**

**MARKING DIAGRAM**




| | |
|---|---|
| TIP12x | = Device Code |
| x | = 0, 1, 2, 5, 6, or 7 |
| A | = Assembly Location |
| Y | = Year |
| WW | = Work Week |
| G | = Pb−Free Package |

**ORDERING INFORMATION**

See detailed ordering and shipping information on page 7 of this data sheet.

*For additional information on our Pb−Free strategy and soldering details, please download the **onsemi** Soldering and Mounting Techniques Reference Manual, SOLDERRM/D.



June, 2024 − Rev. 10
Publication Order Number:
TIP120/D

### MAXIMUM RATINGS

| Symbol | Rating | TIP120, TIP125 | TIP121, TIP126 | TIP122, TIP127 | Unit |
|---|---|---|---|---|---|
| $V_{CEO}$ | Collector–Emitter Voltage | 60 | 80 | 100 | Vdc |
| $V_{CB}$ | Collector–Base Voltage | 60 | 80 | 100 | Vdc |
| $V_{EB}$ | Emitter–Base Voltage | 5.0 | | | Vdc |
| $I_C$ | Collector Current – Continuous<br>– Peak | 5.0<br>8.0 | | | Adc |
| $I_B$ | Base Current | 120 | | | mAdc |
| $P_D$ | Total Power Dissipation @ $T_C$ = 25°C<br>Derate above 25°C | 65<br>0.52 | | | W<br>W/°C |
| $P_D$ | Total Power Dissipation @ $T_A$ = 25°C<br>Derate above 25°C | 2.0<br>0.016 | | | W<br>W/°C |
| E | Unclamped Inductive Load Energy (Note 1) | 50 | | | mJ |
| $T_J$, $T_{stg}$ | Operating and Storage Junction, Temperature Range | –65 to +150 | | | °C |

Stresses exceeding those listed in the Maximum Ratings table may damage the device. If any of these limits are exceeded, device functionality should not be assumed, damage may occur and reliability may be affected.

### THERMAL CHARACTERISTICS

| Symbol | Characteristic | Max | Unit |
|---|---|---|---|
| $R_{\theta JC}$ | Thermal Resistance, Junction–to–Case | 1.92 | °C/W |
| $R_{\theta JA}$ | Thermal Resistance, Junction–to–Ambient | 62.5 | °C/W |

1. $I_C$ = 1 A, L = 100 mH, P.R.F. = 10 Hz, $V_{CC}$ = 20 V, $R_{BE}$ = 100 Ω

### ELECTRICAL CHARACTERISTICS ($T_C$ = 25°C unless otherwise noted)

| Symbol | Characteristic | | Min | Max | Unit |
|---|---|---|---|---|---|
| **OFF CHARACTERISTICS** | | | | | |
| $V_{CEO(sus)}$ | Collector–Emitter Sustaining Voltage (Note 2)<br>($I_C$ = 100 mAdc, $I_B$ = 0) | TIP120, TIP125<br>TIP121, TIP126<br>TIP122, TIP127 | 60<br>80<br>100 | –<br>–<br>– | Vdc |
| $I_{CEO}$ | Collector Cutoff Current<br>($V_{CE}$ = 30 Vdc, $I_B$ = 0)<br>($V_{CE}$ = 40 Vdc, $I_B$ = 0)<br>($V_{CE}$ = 50 Vdc, $I_B$ = 0) | TIP120, TIP125<br>TIP121, TIP126<br>TIP122, TIP127 | –<br>–<br>– | 0.5<br>0.5<br>0.5 | mAdc |
| $I_{CBO}$ | Collector Cutoff Current<br>($V_{CB}$ = 60 Vdc, $I_E$ = 0)<br>($V_{CB}$ = 80 Vdc, $I_E$ = 0)<br>($V_{CB}$ = 100 Vdc, $I_E$ = 0) | TIP120, TIP125<br>TIP121, TIP126<br>TIP122, TIP127 | –<br>–<br>– | 0.2<br>0.2<br>0.2 | mAdc |
| $I_{EBO}$ | Emitter Cutoff Current ($V_{BE}$ = 5.0 Vdc, $I_C$ = 0) | | – | 2.0 | mAdc |
| **ON CHARACTERISTICS** (Note 2) | | | | | |
| $h_{FE}$ | DC Current Gain ($I_C$ = 0.5 Adc, $V_{CE}$ = 3.0 Vdc)<br>($I_C$ = 3.0 Adc, $V_{CE}$ = 3.0 Vdc) | | 1000<br>1000 | –<br>– | – |
| $V_{CE(sat)}$ | Collector–Emitter Saturation Voltage<br>($I_C$ = 3.0 Adc, $I_B$ = 12 mAdc)<br>($I_C$ = 5.0 Adc, $I_B$ = 20 mAdc) | | –<br>– | 2.0<br>4.0 | Vdc |
| $V_{BE(on)}$ | Base–Emitter On Voltage ($I_C$ = 3.0 Adc, $V_{CE}$ = 3.0 Vdc) | | – | 2.5 | Vdc |
| **DYNAMIC CHARACTERISTICS** | | | | | |
| $h_{fe}$ | Small–Signal Current Gain ($I_C$ = 3.0 Adc, $V_{CE}$ = 4.0 Vdc, f = 1.0 MHz) | | 4.0 | – | – |
| $C_{ob}$ | Output Capacitance ($V_{CB}$ = 10 Vdc, $I_E$ = 0, f = 0.1 MHz | TIP125, TIP126, TIP127<br>TIP120, TIP121, TIP122 | –<br>– | 300<br>200 | pF |

Product parametric performance is indicated in the Electrical Characteristics for the listed test conditions, unless otherwise noted. Product performance may not be indicated by the Electrical Characteristics if operated under different conditions.

2. Pulse Test: Pulse Width ≤ 300 μs, Duty Cycle ≤ 2%

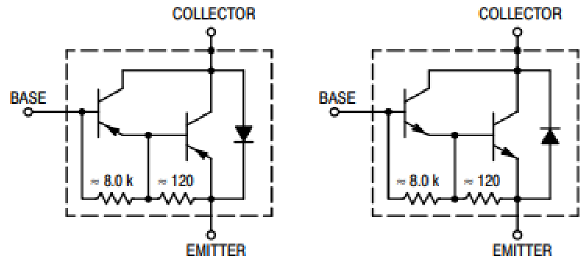



**Figure 1. Darlington Circuit Schematic**

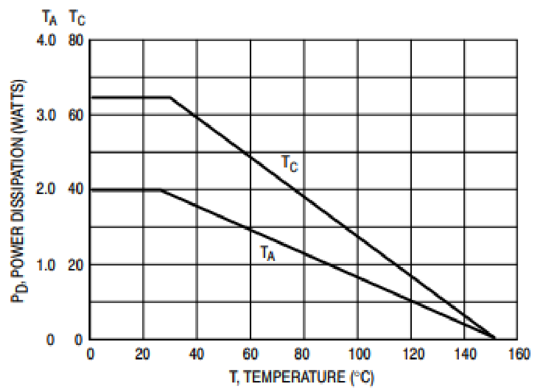



**Figure 2. Power Derating**

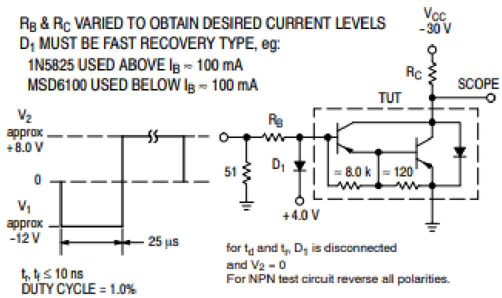



Figure 3. Switching Times Test Circuit

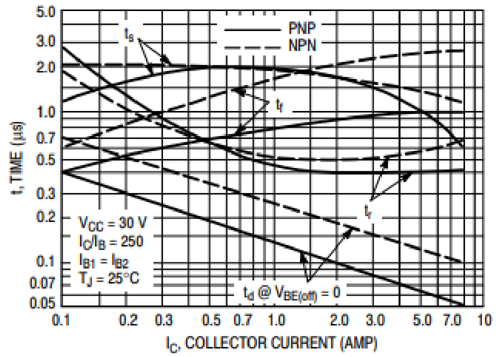



Figure 4. Switching Times

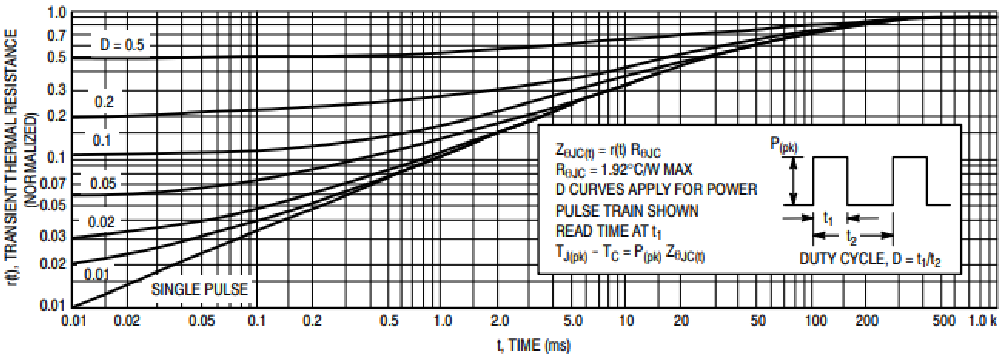



Figure 5. Thermal Response

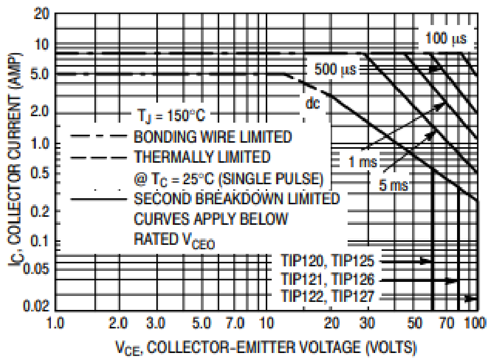



**Figure 6. Active–Region Safe Operating Area**

There are two limitations on the power handling ability of a transistor: average junction temperature and second breakdown. Safe operating area curves indicate $I_C - V_{CE}$ limits of the transistor that must be observed for reliable operation, i.e., the transistor must not be subjected to greater dissipation than the curves indicate.

The data of Figure 6 is based on $T_{J(pk)} = 150°C$; $T_C$ is variable depending on conditions. Second breakdown pulse limits are valid for duty cycles to 10% provided $T_{J(pk)} < 150°C$. $T_{J(pk)}$ may be calculated from the data in Figure 5. At high case temperatures, thermal limitations will reduce the power that can be handled to values less than the limitations imposed by second breakdown

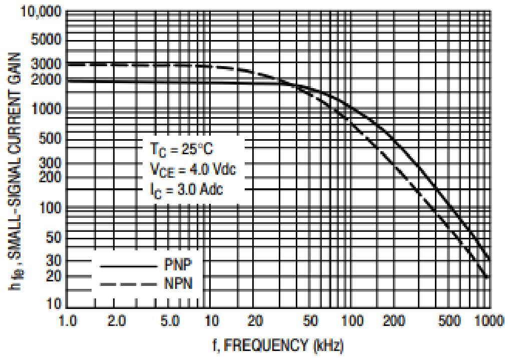



**Figure 7. Small–Signal Current Gain**

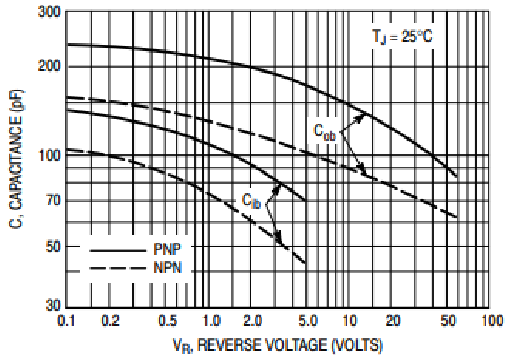



**Figure 8. Capacitance**

**NPN**
**TIP120, TIP121, TIP122**

**PNP**
**TIP125, TIP126, TIP127**

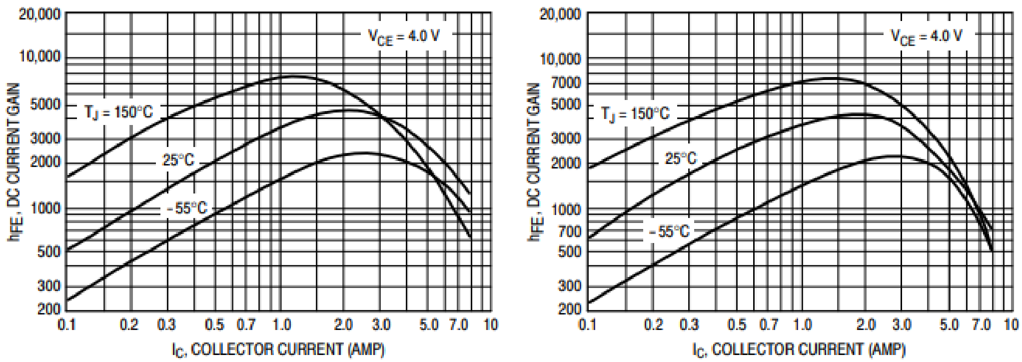



**Figure 9. DC Current Gain**

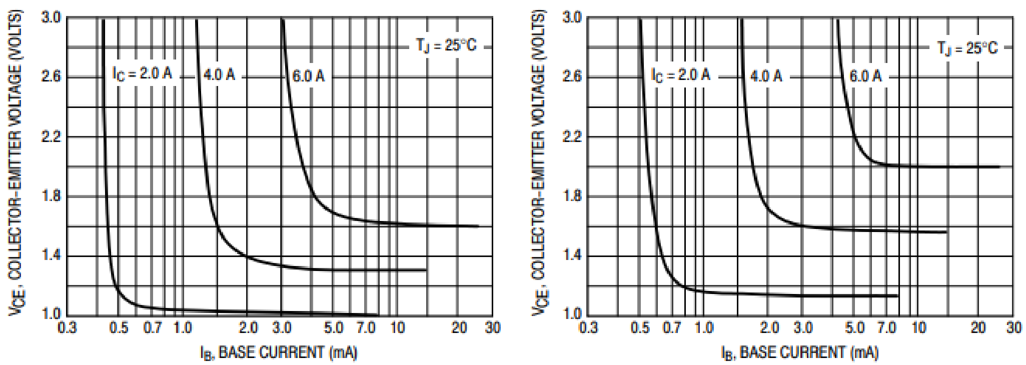



**Figure 10. Collector Saturation Region**

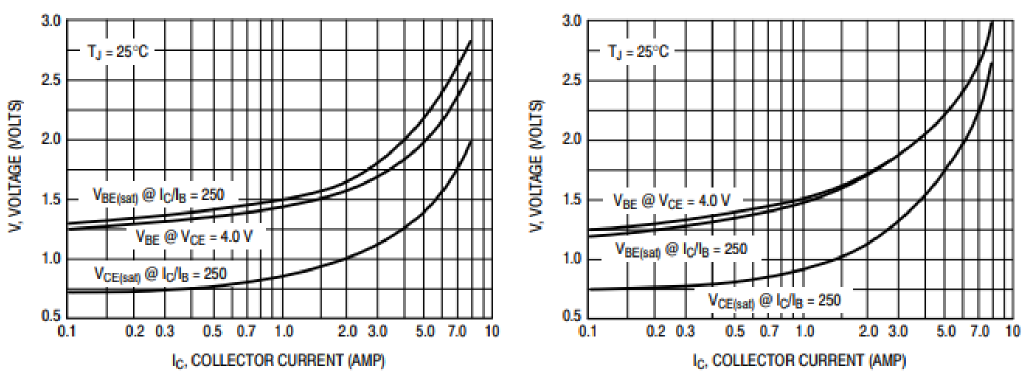
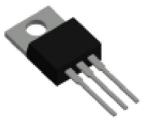



**Figure 11. “On” Voltages**

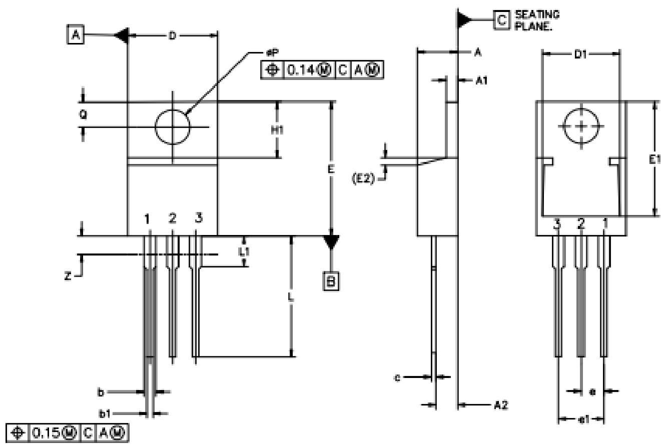
## ORDERING INFORMATION

| Device | Package | Shipping |
|---|---|---|
| TIP120 | TO−220 | 50 Units / Rail |
| TIP120G | TO−220<br>(Pb−Free) | 50 Units / Rail |
| TIP121 | TO−220 | 50 Units / Rail |
| TIP121G | TO−220<br>(Pb−Free) | 50 Units / Rail |
| TIP122 | TO−220 | 50 Units / Rail |
| TIP122G | TO−220<br>(Pb−Free) | 50 Units / Rail |
| TIP125 | TO−220 | 50 Units / Rail |
| TIP125G | TO−220<br>(Pb−Free) | 50 Units / Rail |
| TIP126 | TO−220 | 50 Units / Rail |
| TIP126G | TO−220<br>(Pb−Free) | 50 Units / Rail |
| TIP127 | TO−220 | 50 Units / Rail |
| TIP127G | TO−220<br>(Pb−Free) | 50 Units / Rail |



**TO−220−3 10.10x15.12x4.45, 2.54P**
CASE 221A
ISSUE AL

DATE 05 FEB 2025




| MILLIMETERS | | | |
|---|---|---|---|
| DIM | MIN | NOM | MAX |
| A | 4.07 | 4.45 | 4.83 |
| A1 | 1.15 | 1.28 | 1.41 |
| A2 | 2.04 | 2.42 | 2.79 |
| b | 1.15 | 1.34 | 1.52 |
| b1 | 0.64 | 0.80 | 0.96 |
| c | 0.36 | 0.49 | 0.61 |
| D | 9.66 | 10.10 | 10.53 |
| D1 | 8.43 | 8.63 | 8.83 |
| E | 14.48 | 15.12 | 15.75 |
| E1 | 12.58 | 12.78 | 12.98 |
| E2 | 1.27 REF | | |

| MILLIMETERS | | | |
|---|---|---|---|
| DIM | MIN | NOM | MAX |
| e | 2.42 | 2.54 | 2.66 |
| e1 | 4.83 | 5.08 | 5.33 |
| H1 | 5.97 | 6.22 | 6.47 |
| L | 12.70 | 13.49 | 14.27 |
| L1 | 2.80 | 3.45 | 4.10 |
| Q | 2.54 | 2.79 | 3.04 |
| øP | 3.60 | 3.85 | 4.09 |
| Z | −.−− | −.−− | 3.48 |

NOTES:
1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M, 2018.
2. CONTROLLING DIMENSION: MILLIMETERS.
3. DIMENSION Z DEFINES A ZONE WHERE ALL BODY AND LEAD IRREGULARITIES ARE ALLOWED.

STYLE 1:
PIN 1. BASE
2. COLLECTOR
3. EMITTER
4. COLLECTOR

STYLE 2:
PIN 1. BASE
2. EMITTER
3. COLLECTOR
4. EMITTER

STYLE 3:
PIN 1. CATHODE
2. ANODE
3. GATE
4. ANODE

STYLE 4:
PIN 1. MAIN TERMINAL 1
2. MAIN TERMINAL 2
3. GATE
4. MAIN TERMINAL 2

STYLE 5:
PIN 1. GATE
2. DRAIN
3. SOURCE
4. DRAIN

STYLE 6:
PIN 1. ANODE
2. CATHODE
3. ANODE
4. CATHODE

STYLE 7:
PIN 1. CATHODE
2. ANODE
3. CATHODE
4. ANODE

STYLE 8:
PIN 1. CATHODE
2. ANODE
3. EXTERNAL TRIP/DELAY
4. ANODE

STYLE 9:
PIN 1. GATE
2. COLLECTOR
3. EMITTER
4. COLLECTOR

STYLE 10:
PIN 1. GATE
2. SOURCE
3. DRAIN
4. SOURCE

STYLE 11:
PIN 1. DRAIN
2. SOURCE
3. GATE
4. SOURCE

STYLE 12:
PIN 1. MAIN TERMINAL 1
2. MAIN TERMINAL 2
3. GATE
4. NOT CONNECTED

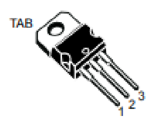
| DOCUMENT NUMBER: | 98ASB42148B | Electronic versions are uncontrolled except when accessed directly from the Document Repository. Printed versions are uncontrolled except when stamped “CONTROLLED COPY” in red. |
|---|---|---|
| DESCRIPTION: | TO−220−3 10.10x15.12x4.45, 2.54P | PAGE 1 OF 1 |

**TIP 125 TRANSISTOR DATASHEET**

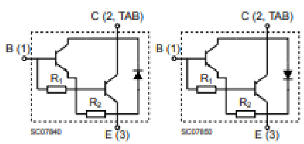
# TIP120, TIP121, TIP122
# TIP125, TIP127

Datasheet

## Low voltage complementary power Darlington transistors




TO-220




NPN: $R_1$ = 7 kΩ, $R_2$ = 70 Ω PNP: $R_1$ = 16 kΩ, $R_2$ = 60 Ω

| Product status links |
|---|
| TIP120 |
| TIP121 |
| TIP122 |
| TIP125 |
| TIP127 |

## Features

- Low collector-emitter saturation voltage
- Complementary NPN - PNP transistors

## Application

- General purpose linear and switching

## Description

The devices are manufactured in planar technology with “base island” layout and monolithic Darlington configuration. The resulting transistors show exceptional high gain performance coupled with very low saturation voltage.

# 1 Electrical ratings

Table 1. **Absolute maximum ratings**

| Symbol | Parameter | | Value | | | Unit |
|---|---|---|---|---|---|---|
| | | NPN | TIP120 | TIP121 | TIP122 | |
| | | PNP | TIP125 | | TIP127 | |
| $V_{CBO}$ | Collector-base voltage ($I_E$ = 0 A) | | 60 | 80 | 100 | V |
| $V_{CEO}$ | Collector-emitter voltage ($I_B$ = 0 A) | | 60 | 80 | 100 | V |
| $V_{EBO}$ | Collector-base voltage ($I_C$ = 0 A) | | 5 | | | V |
| $I_C$ | Collector current | | 5 | | | A |
| $I_{CM}$ | Collector peak current | | 8 | | | A |
| $I_B$ | Base current | | 0.12 | | | A |
| $P_{TOT}$ | Total power dissipation at $T_C \leq 25°C$ | | 65 | | | W |
| | Total power dissipation at $T_A \leq 25°C$ | | 2 | | | |
| $T_{stg}$ | Storage temperature range | | -65 to 150 | | | °C |
| $T_J$ | Maximum operating junction temperature | | 150 | | | °C |

*Note: For PNP types voltage and current values are negative.*

Table 2. **Thermal data**

| Symbol | Parameter | Value | Unit |
|---|---|---|---|
| $R_{thJC}$ | Thermal resistance, junction-to-case | 1.92 | °C/W |
| $R_{thJA}$ | Thermal resistance, junction-to-ambient | 62.5 | °C/W |

## 2 Electrical characteristics

$T_C$ = 25°C unless otherwise specified.

Table 3. **Electrical characteristics**

| Symbol | Parameter | Test conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $I_{CEO}$ | Collector cut-off current | $I_B$ = 0 A, $V_{CE}$ = 30 V for TIP120/125 | | - | 0.5 | mA |
| | | $I_B$ = 0 A, $V_{CE}$ = 40 V for TIP121 | | - | 0.5 | |
| | | $I_B$ = 0 A, $V_{CE}$ = 50 V for TIP122/127 | | - | 0.5 | |
| $I_{CBO}$ | Collector cut-off current | $I_B$ = 0 A, $V_{CE}$ = 60 V for TIP120/125 | | - | 0.2 | mA |
| | | $I_B$ = 0 A, $V_{CE}$ = 80 V for TIP121 | | - | 0.2 | |
| | | $I_B$ = 0 A, $V_{CE}$ = 100 V for TIP122/127 | | - | 0.2 | |
| $I_{EBO}$ | Emitter cut-off current | $I_C$ = 0 A, $V_{EB}$ = 5 V | | - | 2 | mA |
| $V_{CEO(sus)}$ (1) | Collector-emitter sustaining voltage | $I_B$ = 0 A, $I_C$ = 30 mA for TIP120/125 | 60 | - | | V |
| | | $I_B$ = 0 A, $I_C$ = 30 mA for TIP121 | 80 | - | | |
| | | $I_B$ = 0 A, $I_C$ = 30 mA for TIP122/127 | 100 | - | | |
| $V_{CE(sat)}$ (1) | Collector-emitter saturation voltage | $I_C$ = 3 A, $I_B$ = 12 mA | | - | 2 | V |
| | | $I_C$ = 5 A $I_B$ = 20 mA | | - | 4 | |
| $V_{BE(on)}$ (1) | Base-emitter on voltage | $I_C$ = 3 A, $V_{CE}$ = 3 V | | - | 2.5 | V |
| $h_{FE}$ (1) | DC current gain | $I_C$ = 0.5 A, $V_{CE}$ = 3 V | 1000 | - | | |
| | | $I_C$ = 3 A, $V_{CE}$ = 3 V | 1000 | - | | |

1. *Pulsed: Pulse duration ≤ 300 µs, duty cycle ≤ 2%.*

*Note:* *For PNP types voltage and current values are negative.*

## 2.1 Electrical characteristics (curves)

Figure 1. Safe operating area

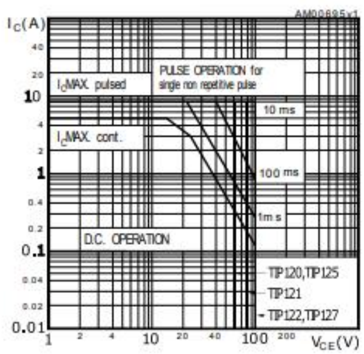



Figure 2. Derating curve

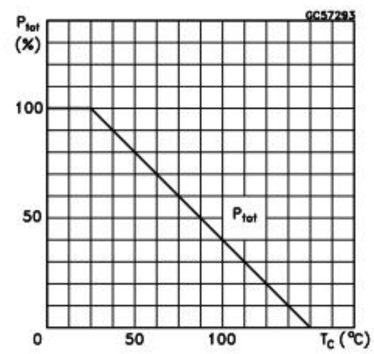



Figure 3. DC current gain (NPN)

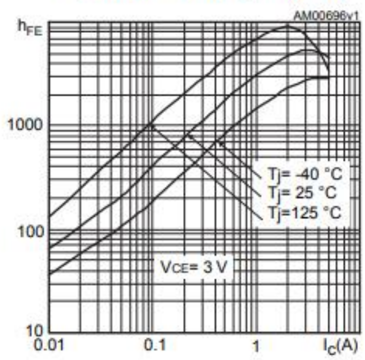



Figure 4. DC current gain (PNP)

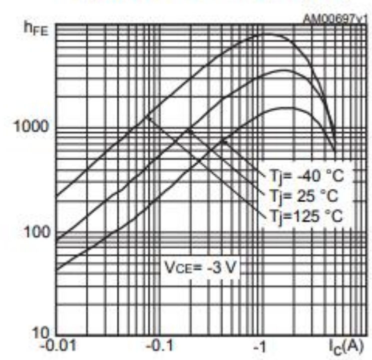



Figure 5. Collector-emitter saturation voltage (NPN)

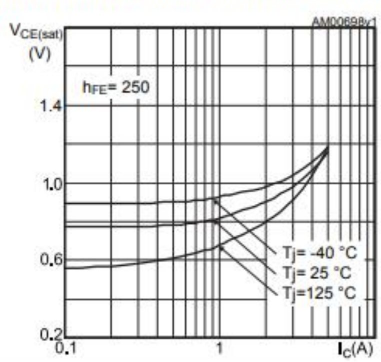



Figure 6. Collector-emitter saturation voltage (PNP)

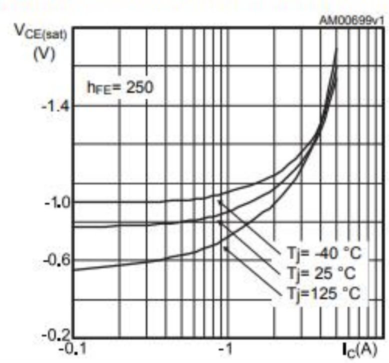


Figure 7. Base-emitter saturation voltage (NPN)

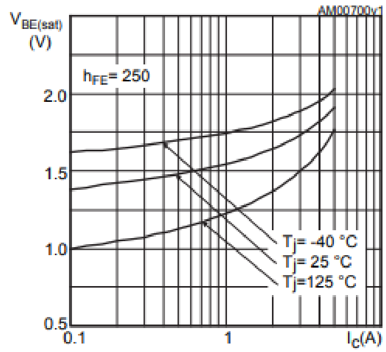



Figure 8. Base-emitter saturation voltage (PNP)

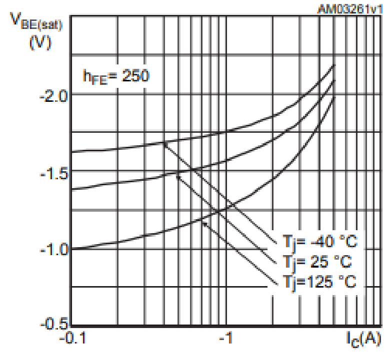



Figure 9. Base-emitter on voltage (NPN)

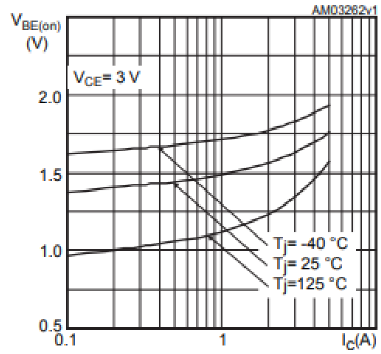



Figure 10. Base-emitter on voltage (PNP)

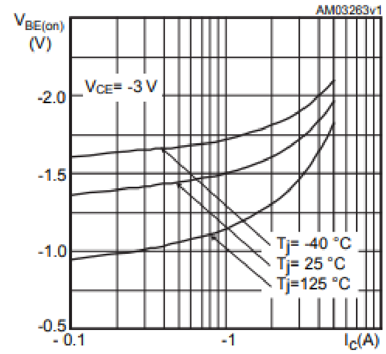



Figure 11. Resistive load switching time (NPN, on)

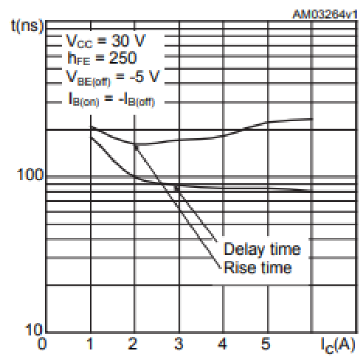



Figure 12. Resistive load switching time (PNP, on)

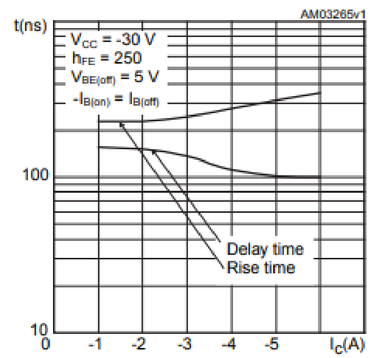


Figure 13. Resistive load switching time (NPN, off)

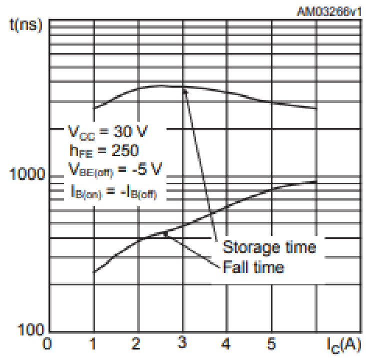



Figure 14. Resistive load switching time (PNP, off)

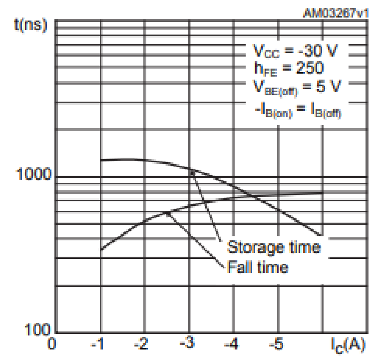



Figure 15. Capacitance characteristics (NPN)

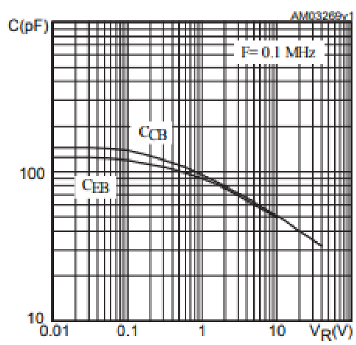
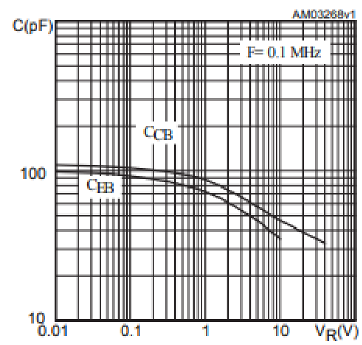
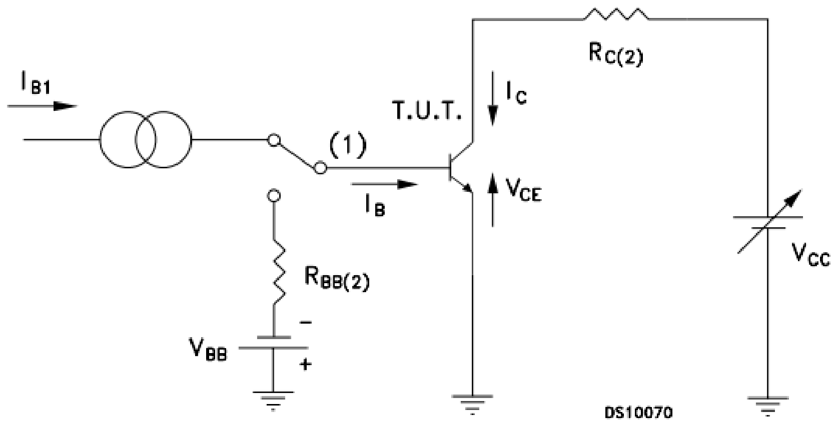



Figure 16. Capacitance characteristics (PNP)



## 3 Test circuits

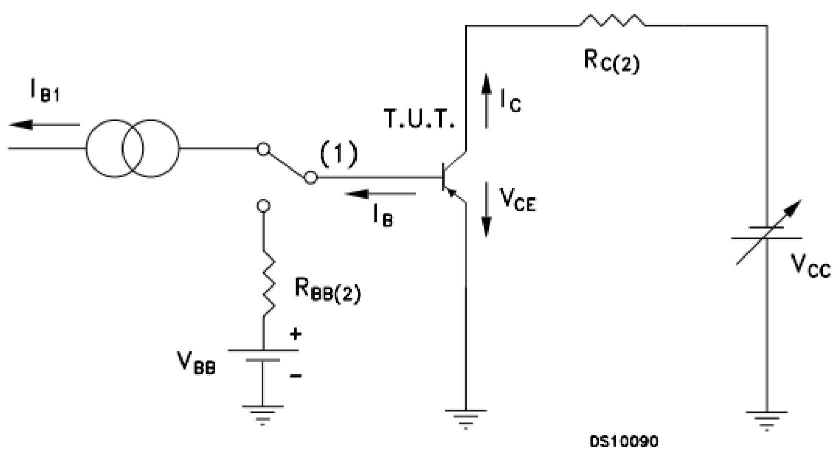
Figure 17. **Resistive load switching for NPN type**




*Note: (1) Fast electronic switch.*

*Note: (2) Non-inductive resistor.*

Figure 18. **Resistive load switching for PNP type**




*Note: (1) Fast electronic switch.*

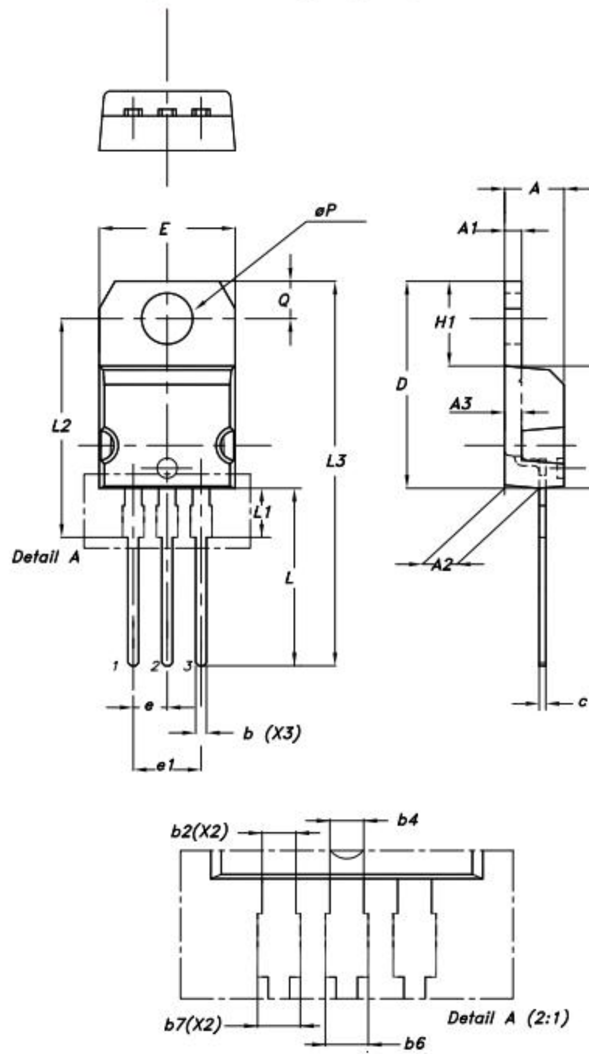
*Note: (2) Non-inductive resistor.*

# 4 Package information

In order to meet environmental requirements, ST offers these devices in different grades of ECOPACK packages, depending on their level of environmental compliance. ECOPACK specifications, grade definitions and product status are available at: www.st.com. ECOPACK is an ST trademark.

## 4.1 TO-220 type H package information

Figure 19. TO-220 type H package outline




0015988_H_23

Table 4. TO-220 type H package mechanical data

| Dim. | mm | | |
|---|---|---|---|
| | Min. | Typ. | Max. |
| A | 4.40 | 4.45 | 4.50 |
| A1 | 1.22 | | 1.32 |
| A2 | 2.49 | 2.59 | 2.69 |
| A3 | 1.17 | 1.27 | 1.37 |
| b | 0.78 | | 0.87 |
| b2 | 1.25 | | 1.34 |
| b4 | 1.20 | | 1.29 |
| b6 | | | 1.50 |
| b7 | | | 1.45 |
| c | 0.49 | | 0.56 |
| D | 15.40 | 15.50 | 15.60 |
| D1 | 9.05 | 9.15 | 9.25 |
| E | 10.08 | 10.18 | 10.28 |
| e | 2.44 | 2.54 | 2.64 |
| e1 | 4.98 | 5.08 | 5.18 |
| H1 | 6.25 | 6.35 | 6.45 |
| L | 13.20 | 13.40 | 13.60 |
| L1 | 3.50 | 3.70 | 3.90 |
| L2 | 16.30 | 16.40 | 16.50 |
| L3 | 28.70 | 28.90 | 29.10 |
| ∅P | 3.75 | 3.80 | 3.85 |
| Q | 2.70 | 2.80 | 2.90 |
| Slug flatness | | 0.03 | 0.10 |

## 5 Ordering information

Table 5. Order codes

| Order codes | Marking | Polarity | Package | Packing |
|---|---|---|---|---|
| TIP120 | TIP120 | NPN | TO-220 | Tube |
| TIP121 | TIP121 | | | |
| TIP122 | TIP122 | | | |
| TIP125 | TIP125 | PNP | | |
| TIP127 | TIP127 | | | |

## Revision history

Table 6. **Document revision history**

| Date | Version | Changes |
|---|---|---|
| 21-Jun-2004 | 3 | Document migration, no content change. |
| 25-Nov-2008 | 4 | Inserted new *Section 2.1: Electrical characteristics (curves)*. |
| 11-May-2021 | 5 | The part number TIP126 have been removed and the document has been updated accordingly.<br>Updated title and added STPOWER LOGO in cover page.<br>Updated Section 4 Package information.<br>Minor text changes. |

# ARDUINO UNO R3 DATASHEET



Product Reference Manual
SKU: A000066

## Description

The Arduino® UNO R3 is the perfect board to get familiar with electronics and coding. This versatile development board is equipped with the well-known ATmega328P and the ATMega 16U2 Processor.

This board will give you a great first experience within the world of Arduino.

## Target areas:

Maker, introduction, industries

## Features

- **ATMega328P** Processor
  - **Memory**
    - AVR CPU at up to 16 MHz
    - 32 kB Flash
    - 2 kB SRAM
    - 1 kB EEPROM
  - **Security**
    - Power On Reset (POR)
    - Brown Out Detection (BOD)
  - **Peripherals**
    - 2x 8-bit Timer/Counter with a dedicated period register and compare channels
    - 1x 16-bit Timer/Counter with a dedicated period register, input capture and compare channels
    - 1x USART with fractional baud rate generator and start-of-frame detection
    - 1x controller/peripheral Serial Peripheral Interface (SPI)
    - 1x Dual mode controller/peripheral I2C
    - 1x Analog Comparator (AC) with a scalable reference input
    - Watchdog Timer with separate on-chip oscillator
    - Six PWM channels
    - Interrupt and wake-up on pin change
  - **ATMega16U2 Processor**
    - 8-bit AVR® RISC-based microcontroller
  - **Memory**
    - 16 kB ISP Flash
    - 512B EEPROM
    - 512B SRAM
    - debugWIRE interface for on-chip debugging and programming
  - **Power**
    - 2.7-5.5 volts

# CONTENTS

# 1 The Board

## 1.1 Application Examples

The UNO board is the flagship product of Arduino. Regardless if you are new to the world of electronics or will use the UNO R3 as a tool for education purposes or industry-related tasks, the UNO R3 is likely to meet your needs.

**First entry to electronics:** If this is your first project within coding and electronics, get started with our most used and documented board; UNO. It is equipped with the well-known ATmega328P processor, 14 digital input/output pins, 6 analog inputs, USB connections, ICSP header and reset button. This board includes everything you will need for a great first experience with Arduino.

**Industry-standard development board:** Using the UNO R3 board in industries, there are a range of companies using the UNO R3 board as the brain for their PLC's.

**Education purposes:** Although the UNO R3 board has been with us for about ten years, it is still widely used for various education purposes and scientific projects. The board's high standard and top quality performance makes it a great resource to capture real time from sensors and to trigger complex laboratory equipment to mention a few examples.

## 1.2 Related Products

- Arduino Starter Kit
- Arduino UNO R4 Minima
- Arduino UNO R4 WiFi
- Tinkerkit Braccio Robot

# 2 Ratings

## 2.1 Recommended Operating Conditions

| Symbol | Description | Min | Max |
|---|---|---|---|
| | Conservative thermal limits for the whole board: | -40 °C (-40 °F) | 85 °C ( 185 °F) |

**NOTE:** In extreme temperatures, EEPROM, voltage regulator, and the crystal oscillator, might not work as expected.

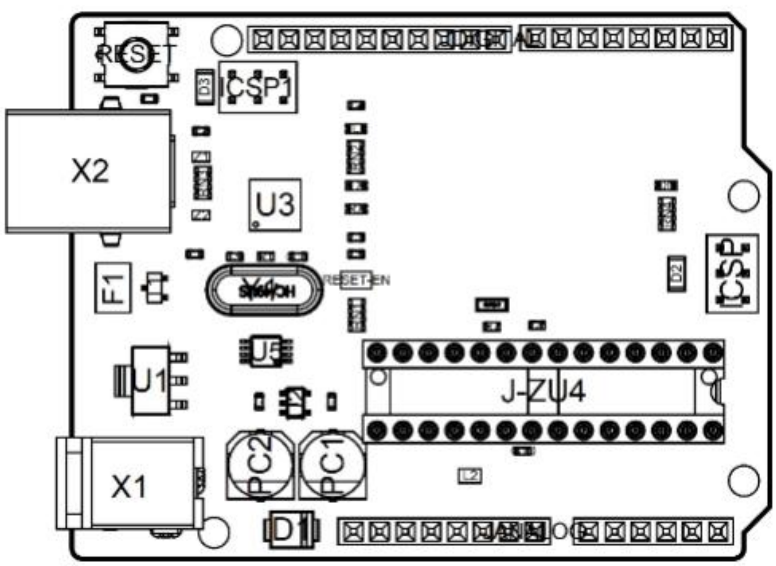
## 2.2 Power Consumption

| Symbol | Description | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| VINMax | Maximum input voltage from VIN pad | 6 | - | 20 | V |
| VUSBMax | Maximum input voltage from USB connector | | - | 5.5 | V |
| PMax | Maximum Power Consumption | - | - | xx | mA |

# 3 Functional Overview

## 3.1 Board Topology

Top view



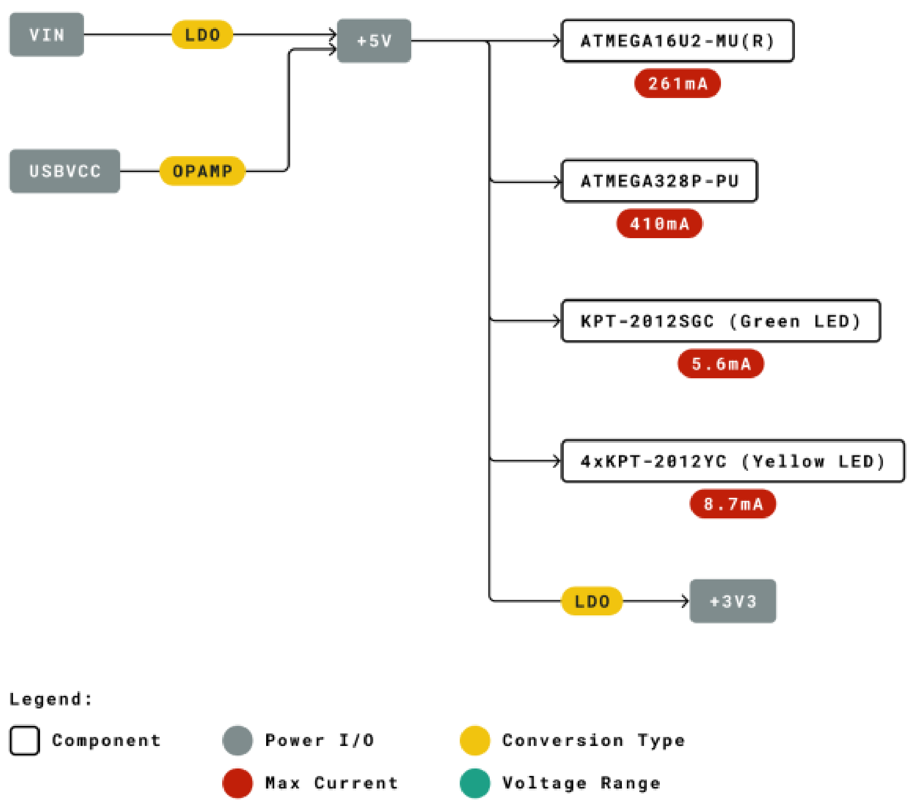

*Board topology*

| Ref. | Description | Ref. | Description |
|---|---|---|---|
| X1 | Power jack 2.1x5.5mm | U1 | SPX1117M3-L-5 Regulator |
| X2 | USB B Connector | U3 | ATMEGA16U2 Module |
| PC1 | EEE-1EA470WP 25V SMD Capacitor | U5 | LMV358LIST-A.9 IC |
| PC2 | EEE-1EA470WP 25V SMD Capacitor | F1 | Chip Capacitor, High Density |
| D1 | CGRA4007-G Rectifier | ICSP | Pin header connector (through hole 6) |
| J-ZU4 | ATMEGA328P Module | ICSP1 | Pin header connector (through hole 6) |
| Y1 | ECS-160-20-4X-DU Oscillator | | |

### 3.2 Processor

The Main Processor is a ATmega328P running at up to 20 MHz. Most of its pins are connected to the external headers, however some are reserved for internal communication with the USB Bridge coprocessor.

### 3.3 Power Tree




*Power tree*

# 4 Board Operation

## 4.1 Getting Started - IDE

If you want to program your UNO R3 while offline you need to install the Arduino Desktop IDE [1] To connect the UNO R3 to your computer, you'll need a USB-B cable. This also provides power to the board, as indicated by the LED.

## 4.2 Getting Started - Arduino Cloud Editor

All Arduino boards, including this one, work out-of-the-box on the Arduino Cloud Editor [2], by just installing a simple plugin.

The Arduino Cloud Editor is hosted online, therefore it will always be up-to-date with the latest features and support for all boards. Follow **[3]** to start coding on the browser and upload your sketches onto your board.

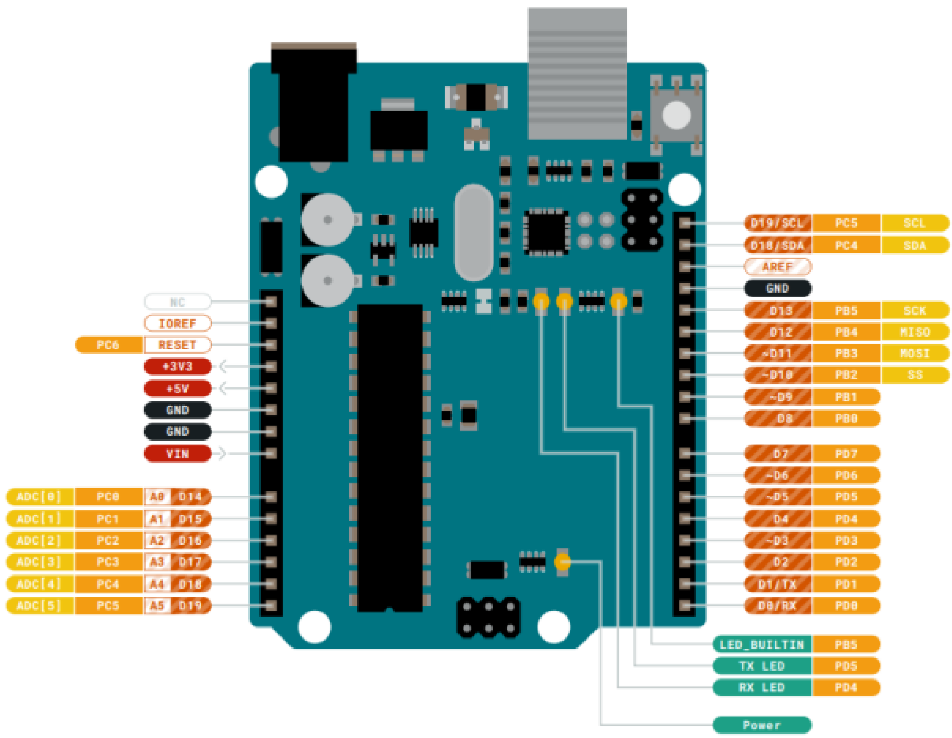
## 4.3 Sample Sketches

Sample sketches for the UNO R3 can be found either in the "Examples" menu in the Arduino IDE or in the "Documentation" section of the Arduino website [4].

## 4.4 Online Resources

Now that you have gone through the basics of what you can do with the board you can explore the endless possibilities it provides by checking exciting projects on Arduino Project Hub [5], the Arduino Library Reference [6] and the online Arduino store [7] where you will be able to complement your board with sensors, actuators and more.

## 5 Connector Pinouts




*Pinout*

## 5.1 JANALOG

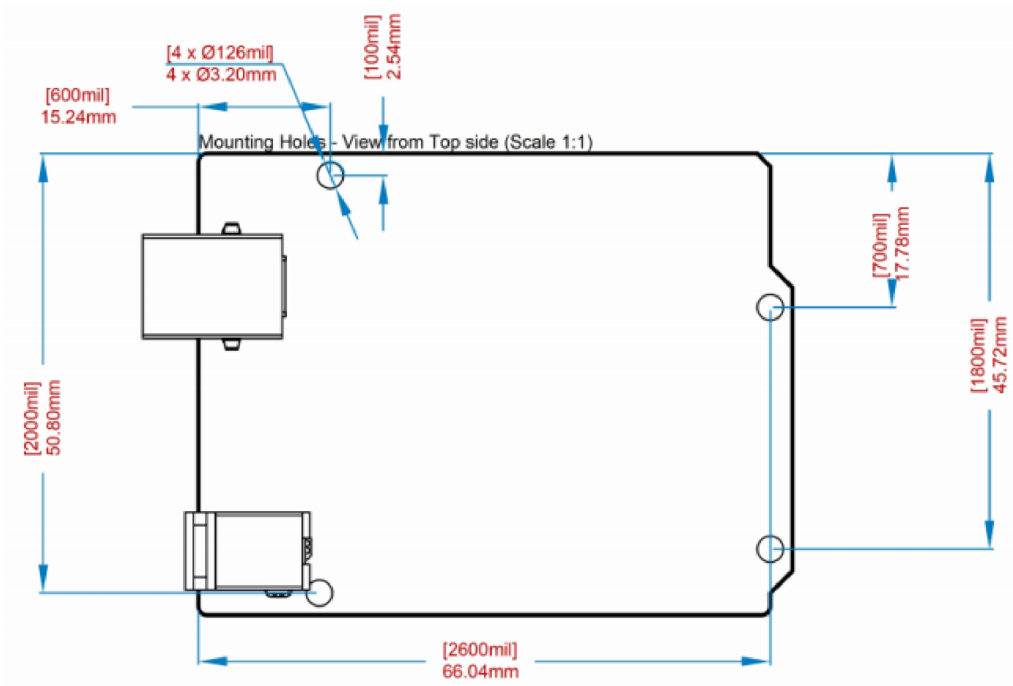
| Pin | Function | Type | Description |
|---|---|---|---|
| 1 | NC | NC | Not connected |
| 2 | IOREF | IOREF | Reference for digital logic V - connected to 5V |
| 3 | Reset | Reset | Reset |
| 4 | +3V3 | Power | +3V3 Power Rail |
| 5 | +5V | Power | +5V Power Rail |
| 6 | GND | Power | Ground |
| 7 | GND | Power | Ground |
| 8 | VIN | Power | Voltage Input |
| 9 | A0 | Analog/GPIO | Analog input 0 /GPIO |
| 10 | A1 | Analog/GPIO | Analog input 1 /GPIO |
| 11 | A2 | Analog/GPIO | Analog input 2 /GPIO |
| 12 | A3 | Analog/GPIO | Analog input 3 /GPIO |
| 13 | A4/SDA | Analog input/I2C | Analog input 4/I2C Data line |
| 14 | A5/SCL | Analog input/I2C | Analog input 5/I2C Clock line |

## 5.2 JDIGITAL

| Pin | Function | Type | Description |
|---|---|---|---|
| 1 | D0 | Digital/GPIO | Digital pin 0/GPIO |
| 2 | D1 | Digital/GPIO | Digital pin 1/GPIO |
| 3 | D2 | Digital/GPIO | Digital pin 2/GPIO |
| 4 | D3 | Digital/GPIO | Digital pin 3/GPIO |
| 5 | D4 | Digital/GPIO | Digital pin 4/GPIO |
| 6 | D5 | Digital/GPIO | Digital pin 5/GPIO |
| 7 | D6 | Digital/GPIO | Digital pin 6/GPIO |
| 8 | D7 | Digital/GPIO | Digital pin 7/GPIO |
| 9 | D8 | Digital/GPIO | Digital pin 8/GPIO |
| 10 | D9 | Digital/GPIO | Digital pin 9/GPIO |
| 11 | SS | Digital | SPI Chip Select |
| 12 | MOSI | Digital | SPI1 Main Out Secondary In |
| 13 | MISO | Digital | SPI Main In Secondary Out |
| 14 | SCK | Digital | SPI serial clock output |
| 15 | GND | Power | Ground |
| 16 | AREF | Digital | Analog reference voltage |
| 17 | A4/SD4 | Digital | Analog input 4/I2C Data line (duplicated) |
| 18 | A5/SD5 | Digital | Analog input 5/I2C Clock line (duplicated) |

## 5.3 Mechanical Information

## 5.4 Board Outline & Mounting Holes




*Board outline*

# 6 Certifications

## 6.1 Declaration of Conformity CE DoC (EU)

We declare under our sole responsibility that the products above are in conformity with the essential requirements of the following EU Directives and therefore qualify for free movement within markets comprising the European Union (EU) and European Economic Area (EEA).

| | |
|---|---|
| **ROHS 2 Directive 2011/65/EU** | |
| Conforms to: | EN50581:2012 |
| **Directive 2014/35/EU. (LVD)** | |
| Conforms to: | EN 60950-1:2006/A11:2009/A1:2010/A12:2011/AC:2011 |
| **Directive 2004/40/EC & 2008/46/EC & 2013/35/EU, EMF** | |
| Conforms to: | EN 62311:2008 |

## 6.2 Declaration of Conformity to EU RoHS & REACH 211 01/19/2021

Arduino boards are in compliance with RoHS 2 Directive 2011/65/EU of the European Parliament and RoHS 3 Directive 2015/863/EU of the Council of 4 June 2015 on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

| Substance | Maximum limit (ppm) |
|---|---|
| Lead (Pb) | 1000 |
| Cadmium (Cd) | 100 |
| Mercury (Hg) | 1000 |
| Hexavalent Chromium (Cr6+) | 1000 |
| Poly Brominated Biphenyls (PBB) | 1000 |
| Poly Brominated Diphenyl ethers (PBDE) | 1000 |
| Bis(2-Ethylhexyl} phthalate (DEHP) | 1000 |
| Benzyl butyl phthalate (BBP) | 1000 |
| Dibutyl phthalate (DBP) | 1000 |
| Diisobutyl phthalate (DIBP) | 1000 |

Exemptions: No exemptions are claimed.

Arduino Boards are fully compliant with the related requirements of European Union Regulation (EC) 1907 /2006 concerning the Registration, Evaluation, Authorization and Restriction of Chemicals (REACH). We declare none of the SVHCs (https://echa.europa.eu/web/guest/candidate-list-table), the Candidate List of Substances of Very High Concern for authorization currently released by ECHA, is present in all products (and also package) in quantities totaling in a concentration equal or above 0.1%. To the best of our knowledge, we also declare that our products do not contain any of the substances listed on the "Authorization List" (Annex XIV of the REACH regulations) and Substances of Very High Concern (SVHC) in any significant amounts as specified by the Annex XVII of Candidate list published by ECHA (European Chemical Agency) 1907 /2006/EC.

### 6.3 Conflict Minerals Declaration

As a global supplier of electronic and electrical components, Arduino is aware of our obligations with regards to laws and regulations regarding Conflict Minerals, specifically the Dodd-Frank Wall Street Reform and Consumer Protection Act, Section 1502. Arduino does not directly source or process conflict minerals such as Tin, Tantalum, Tungsten, or Gold. Conflict minerals are contained in our products in the form of solder, or as a component in metal alloys. As part of our reasonable due diligence Arduino has contacted component suppliers within our supply chain to verify their continued compliance with the regulations. Based on the information received thus far we declare that our products contain Conflict Minerals sourced from conflict-free areas.

## 7 FCC Caution

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference

(2) this device must accept any interference received, including interference that may cause undesired operation.

**FCC RF Radiation Exposure Statement:**

1. This Transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.
2. This equipment complies with RF radiation exposure limits set forth for an uncontrolled environment.
3. This equipment should be installed and operated with minimum distance 20cm between the radiator & your body.

English: User manuals for license-exempt radio apparatus shall contain the following or equivalent notice in a conspicuous location in the user manual or alternatively on the device or both. This device complies with Industry Canada license-exempt RSS standard(s). Operation is subject to the following two conditions:

(1) this device may not cause interference

(2) this device must accept any interference, including interference that may cause undesired operation of the device.

French: Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes :

(1) l' appareil nedoit pas produire de brouillage

(2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

**IC SAR Warning:**

English This equipment should be installed and operated with minimum distance 20 cm between the radiator and your body.

French: Lors de l' installation et de l' exploitation de ce dispositif, la distance entre le radiateur et le corps est d 'au moins 20 cm.

**Important:** The operating temperature of the EUT can't exceed 85℃ and shouldn't be lower than -40℃.

Hereby, Arduino S.r.l. declares that this product is in compliance with essential requirements and other relevant provisions of Directive 2014/53/EU. This product is allowed to be used in all EU member states.

## 8 Company Information

| Company name | Arduino S.r.l |
|---|---|
| Company Address | Via Andrea Appiani 25 20900 MONZA Italy |

## 9 Reference Documentation

| Reference | Link |
|---|---|
| Arduino IDE (Desktop) | https://www.arduino.cc/en/Main/Software |
| Arduino Cloud Editor | https://create.arduino.cc/editor |
| Arduino Cloud Editor - Getting Started | https://docs.arduino.cc/arduino-cloud/guides/editor/ |
| Arduino Website | https://www.arduino.cc/ |
| Arduino Project Hub | https://create.arduino.cc/projecthub?by=part&part_id=11332&sort=trending |
| Library Reference | https://www.arduino.cc/reference/en/ |
| Arduino Store | https://store.arduino.cc/ |

## 10 Revision History

| Date | Revision | Changes |
|---|---|---|
| 25/04/2024 | 3 | Updated link to new Cloud Editor |
| 26/07/2023 | 2 | General Update |
| 06/2021 | 1 | Datasheet release |

## RAW DATA VALUES TABLE

| Left | Center | Right | IR |
|---|---|---|---|
| 311 | 282 | 999 | 4 |
| 125 | 279 | 999 | 4 |
| 90 | 64 | 289 | 4 |
| 122 | 61 | 999 | 4 |
| 999 | 275 | 275 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 294 | 999 | 4 |
| 91 | 287 | 289 | 4 |
| 999 | 999 | 278 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 291 | 999 | 4 |
| 999 | 293 | 999 | 4 |
| 999 | 291 | 999 | 4 |
| 999 | 278 | 262 | 4 |
| 999 | 999 | 999 | 4 |
| 308 | 277 | 999 | 4 |
| 318 | 999 | 999 | 4 |
| 999 | 288 | 999 | 4 |
| 305 | 256 | 999 | 4 |
| 302 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 286 | 999 | 999 | 4 |
| 85 | 55 | 192 | 4 |
| 83 | 55 | 999 | 4 |
| 84 | 55 | 191 | 4 |
| 286 | 58 | 999 | 4 |
| 322 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 81 | 55 | 240 | 4 |
| 81 | 52 | 55 | 4 |
| 82 | 52 | 83 | 4 |
| 84 | 52 | 53 | 4 |
| 111 | 52 | 53 | 4 |

| | | | |
|---|---|---|---|
| 86 | 53 | 54 | 4 |
| 113 | 54 | 55 | 4 |
| 88 | 55 | 55 | 0 |
| 87 | 55 | 83 | 0 |
| 88 | 54 | 55 | 0 |
| 86 | 53 | 55 | 0 |
| 85 | 53 | 80 | 4 |
| 85 | 52 | 80 | 4 |
| 84 | 53 | 81 | 4 |
| 85 | 53 | 55 | 4 |
| 84 | 53 | 55 | 4 |
| 86 | 55 | 84 | 4 |
| 93 | 58 | 59 | 4 |
| 124 | 62 | 64 | 4 |
| 49 | 88 | 88 | 0 |
| 121 | 88 | 89 | 4 |
| 120 | 89 | 89 | 4 |
| 20 | 16 | 17 | 4 |
| 246 | 59 | 62 | 4 |
| 57 | 88 | 62 | 4 |
| 79 | 45 | 62 | 4 |
| 121 | 89 | 90 | 4 |
| 121 | 42 | 90 | 4 |
| 125 | 90 | 90 | 4 |
| 123 | 61 | 62 | 4 |
| 121 | 88 | 89 | 4 |
| 121 | 88 | 88 | 4 |
| 45 | 88 | 62 | 4 |
| 121 | 96 | 17 | 4 |
| 122 | 60 | 61 | 4 |
| 114 | 55 | 55 | 4 |
| 111 | 52 | 53 | 4 |
| 111 | 50 | 52 | 4 |
| 111 | 52 | 54 | 4 |
| 117 | 55 | 56 | 4 |
| 112 | 55 | 82 | 4 |
| 115 | 83 | 57 | 4 |
| 247 | 82 | 83 | 4 |
| 110 | 76 | 77 | 4 |

| | | | |
|---|---|---|---|
| 83 | 52 | 81 | 4 |
| 120 | 61 | 82 | 4 |
| 98 | 65 | 70 | 4 |
| 110 | 74 | 76 | 4 |
| 144 | 85 | 85 | 4 |
| 281 | 120 | 117 | 4 |
| 192 | 296 | 126 | 0 |
| 202 | 171 | 137 | 0 |
| 215 | 184 | 190 | 4 |
| 221 | 192 | 191 | 4 |
| 233 | 200 | 199 | 4 |
| 244 | 210 | 22 | 4 |
| 255 | 27 | 26 | 4 |
| 293 | 29 | 24 | 4 |
| 999 | 999 | 32 | 4 |
| 66 | 38 | 999 | 4 |
| 66 | 36 | 38 | 4 |
| 68 | 36 | 38 | 4 |
| 69 | 40 | 38 | 4 |
| 69 | 35 | 37 | 4 |
| 68 | 38 | 41 | 4 |
| 67 | 38 | 40 | 4 |
| 67 | 37 | 40 | 4 |
| 64 | 35 | 40 | 4 |
| 68 | 16 | 17 | 4 |
| 68 | 18 | 20 | 4 |
| 43 | 40 | 16 | 4 |
| 67 | 38 | 40 | 4 |
| 67 | 38 | 41 | 4 |
| 43 | 38 | 40 | 4 |
| 42 | 16 | 41 | 4 |
| 44 | 15 | 12 | 4 |
| 67 | 16 | 15 | 4 |
| 67 | 38 | 40 | 4 |
| 67 | 39 | 41 | 4 |
| 68 | 39 | 42 | 4 |
| 68 | 40 | 69 | 4 |
| 68 | 39 | 70 | 4 |
| 67 | 38 | 69 | 4 |

| | | | |
|---|---|---|---|
| 66 | 37 | 68 | 4 |
| 67 | 38 | 70 | 4 |
| 67 | 38 | 999 | 4 |
| 69 | 39 | 70 | 4 |
| 70 | 40 | 44 | 4 |
| 76 | 45 | 48 | 4 |
| 84 | 53 | 53 | 4 |
| 92 | 60 | 60 | 4 |
| 130 | 68 | 67 | 4 |
| 133 | 100 | 72 | 4 |
| 138 | 103 | 102 | 4 |
| 999 | 109 | 106 | 4 |
| 999 | 108 | 106 | 4 |
| 139 | 99 | 82 | 4 |
| 142 | 101 | 99 | 4 |
| 134 | 99 | 99 | 4 |
| 134 | 101 | 83 | 4 |
| 137 | 105 | 84 | 4 |
| 150 | 110 | 109 | 4 |
| 157 | 119 | 115 | 4 |
| 163 | 137 | 124 | 4 |
| 172 | 162 | 134 | 4 |
| 999 | 999 | 166 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 238 | 249 | 999 | 4 |
| 247 | 215 | 214 | 4 |
| 90 | 999 | 999 | 4 |
| 121 | 63 | 999 | 4 |
| 71 | 43 | 100 | 4 |
| 72 | 44 | 999 | 4 |
| 74 | 46 | 999 | 4 |
| 73 | 46 | 999 | 0 |
| 72 | 45 | 107 | 0 |
| 72 | 44 | 107 | 0 |
| 71 | 43 | 999 | 0 |
| 71 | 42 | 104 | 4 |
| 75 | 49 | 47 | 4 |
| 108 | 49 | 50 | 4 |

| | | | |
|---|---|---|---|
| 86 | 53 | 53 | 4 |
| 123 | 61 | 59 | 4 |
| 143 | 104 | 74 | 4 |
| 14111 | 111 | 4 | |
| 254 | 101 | 104 | 4 |
| 164 | 220 | 247 | 4 |
| 242 | 212 | 89 | 4 |
| 126 | 92 | 91 | 4 |
| 134 | 98 | 71 | 4 |
| 182 | 240 | 107 | 4 |
| 192 | 162 | 285 | 4 |
| 205 | 173 | 241 | 4 |
| 217 | 184 | 185 | 4 |
| 251 | 198 | 198 | 4 |
| 261 | 214 | 228 | 4 |
| 296 | 243 | 241 | 4 |
| 999 | 281 | 253 | 4 |
| 999 | 285 | 283 | 4 |
| 493 | 999 | 269 | 4 |
| 80 | 275 | 263 | 4 |
| 77 | 48 | 261 | 4 |
| 77 | 999 | 261 | 4 |
| 333 | 297 | 274 | 4 |
| 80 | 51 | 293 | 4 |
| 81 | 52 | 322 | 4 |
| 76 | 55 | 56 | 4 |
| 78 | 48 | 60 | 4 |
| 85 | 53 | 55 | 4 |
| 91 | 59 | 58 | 4 |
| 102 | 68 | 68 | 4 |
| 142 | 82 | 81 | 4 |
| 185 | 124 | 120 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 143 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 999 | 999 | 999 | 0 |
| 999 | 999 | 999 | 0 |

| | | | |
|---|---|---|---|
| 999 | 999 | 999 | 0 |
| 999 | 999 | 999 | 0 |
| 130 | 71 | 999 | 4 |
| 228 | 72 | 999 | 4 |
| 233 | 999 | 999 | 4 |
| 999 | 999 | 999 | 4 |
| 271 | 999 | 999 | 4 |
| 121 | 91 | 999 | 4 |
| 112 | 128 | 127 | 4 |
| 118 | 58 | 140 | 4 |
| 237 | 241 | 96 | 4 |
| 150 | 154 | 250 | 4 |
| 159 | 130 | 253 | 4 |
| 155 | 144 | 178 | 4 |
| 190 | 155 | 155 | 4 |
| 207 | 174 | 172 | 4 |
| 249 | 195 | 189 | 4 |
| 303 | 237 | 232 | 4 |
| 115 | 280 | 253 | 4 |
| 119 | 90 | 268 | 4 |
| 124 | 93 | 95 | 4 |
| 121 | 92 | 95 | 4 |
| 113 | 84 | 87 | 4 |
| 136 | 80 | 81 | 4 |
| 132 | 73 | 74 | 4 |
| 120 | 64 | 65 | 4 |
| 87 | 57 | 58 | 4 |
| 83 | 53 | 55 | 4 |
| 84 | 54 | 55 | 4 |
| 85 | 55 | 57 | 4 |
| 93 | 62 | 267 | 4 |
| 102 | 71 | 239 | 4 |
| 270 | 240 | 236 | 4 |
| 270 | 237 | 236 | 4 |
| 25 | 37 | 16 | 4 |
| 25 | 16 | 35 | 4 |
| 25 | 16 | 35 | 4 |
| 24 | 37 | 16 | 4 |
| 33 | 37 | 35 | 4 |

| | | | |
|---|---|---|---|
| 23 | 15 | 16 | 4 |
| 74 | 42 | 30 | 4 |
| 44 | 42 | 55 | 4 |
| 45 | 42 | 60 | 4 |
| 54 | 53 | 21 | 4 |
| 46 | 15 | 36 | 4 |
| 35 | 49 | 53 | 4 |
| 37 | 41 | 55 | 4 |
| 47 | 19 | 48 | 4 |
| 29 | 16 | 15 | 4 |
| 29 | 16 | 16 | 4 |
| 29 | 16 | 16 | 4 |
| 29 | 16 | 16 | 0 |
| 29 | 16 | 16 | 0 |
| 28 | 16 | 16 | 0 |
| 28 | 16 | 16 | 0 |
| 38 | 16 | 16 | 0 |
| 44 | 16 | 16 | 0 |
| 44 | 16 | 16 | 4 |
| 28 | 16 | 16 | 4 |
| 29 | 16 | 16 | 4 |
| 28 | 16 | 16 | 4 |
| 29 | 16 | 16 | 4 |
| 39 | 16 | 16 | 4 |