

## Ass.1 [Hands on Arduino](#)

In case of any doubts in this tutorial, contact Ayush Sinha (9850892135),  
Ram==kumar (8778096606) and HimaRa (9340607951)

### Given Components:-

A Motor Driver, 1 DC motor, jumpers, breadboard and a battery.

- **Step 1: The Circuit**

Make the circuit for connecting motor driver to motor and battery.  
Then for motor driver to arduino.

- **Step 2: The Code**

Start with writing the header files and void setup.  
Declare arduino connections and variables.  
Start writing the code for increasing/decreasing the speed of motor in a  
for loop.  
You can set the direction of motor within the loop.

**Try to play with it by giving random speed to  
motors.**

### References

- [Jeremy | Tutorials](#) for learning arduino.

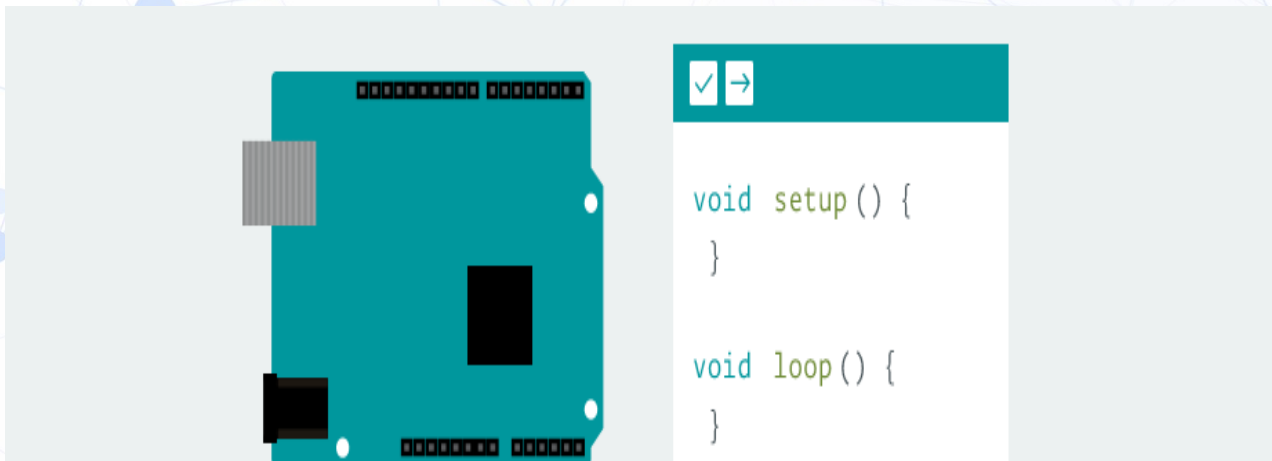
---

## Ass.1 [Hands on Arduino](#)

In case of any doubts in this tutorial, contact Ayush Sinha (9850892135),  
Ram==kumar (8778096606)

# Foundations

The Foundations section gives you some specific knowledge about the principles and techniques that are behind the Arduino platform, its whole family of boards and the software you use to tell your boards what to do.

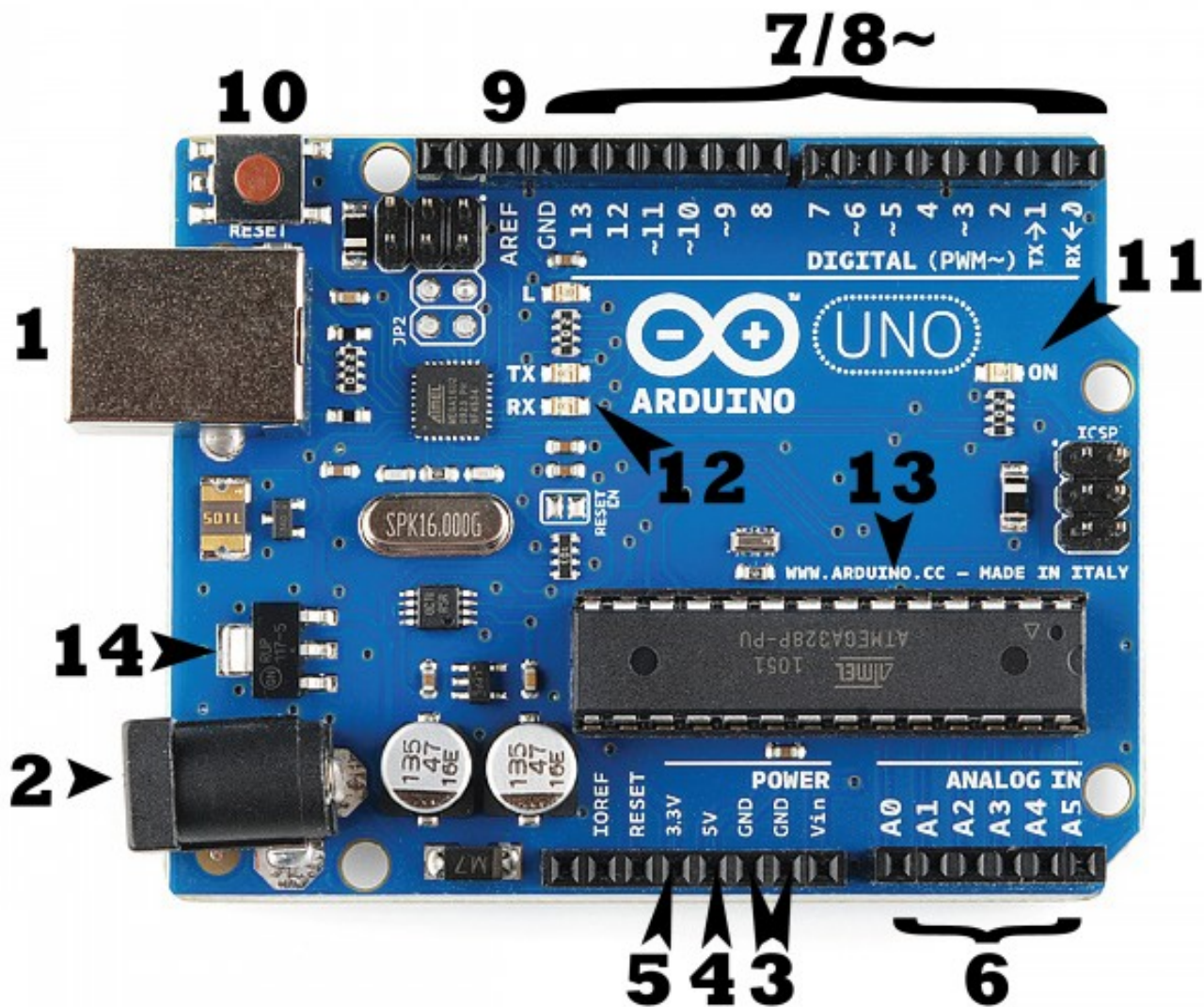


If you want to get some knowledge before you start using things, this is the right place. If you want to jump on tinkering with your board visit the [Getting Started page](#).



## Ass.1 [Hands on Arduino](#)

In case of any doubts in this tutorial, contact Ayush Sinha (9850892135), HimaRa (9340007051)



---

## Ass.1 [Hands on Arduino](#)

In case of any doubts in this tutorial, contact Ayush Sinha (9850892135), HimaRa (9340607951)

### 1 & 2 :- Power (USB / Barrel Jack)

Pins (5V, 3.3V, GND, Analog, Digital, PWM, AREF)

- **GND (3):** Short for 'Ground'. There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- **5V (4) & 3.3V (5):** As you might guess, the 5V pin supplies 5 volts of power, and the 3.3V pin supplies 3.3 volts of power. Most of the simple components used with the Arduino run happily off of 5 or 3.3 volts.
- **Analog (6):** The area of pins under the 'Analog In' label (A0 through A5 on the UNO) are Analog In pins. These pins can read the signal from an analog sensor (like a [temperature sensor](#)) and convert it into a digital value that we can read.
- **Digital (7):** Across from the analog pins are the digital pins (0 through 13 on the UNO). These pins can be used for both digital input (like telling if a button is pushed) and digital output (like powering an LED).
- **PWM (8):** You may have noticed the tilde (~) next to some of the digital pins (3, 5, 6, 9, 10, and 11 on the UNO). These pins act as normal digital pins, but can also be used for something called Pulse-Width Modulation (PWM). We have [a tutorial on PWM](#), but for now, think of these pins as being able to simulate analog output (like fading an LED in and out).



---

## Ass.1 [Hands on Arduino](#)

In case of any doubts in this tutorial, contact Ayush Sinha (9850892135), HimaRa (9340607951)

1 & 2 :- Power (USB / Barrel Jack)

Pins (5V, 3.3V, GND, Analog, Digital, PWM, AREF)

- **AREF (9):** Stands for Analog Reference. Most of the time you can leave this pin alone. It is sometimes used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pin

11. Power LED Indicator

## Programming.

[Variables](#): Understand how to define and use variables in a Sketch

Char, String, int, long, float, etc..

[Functions](#): Learn how to define and use functions in a Sketch.

digitalwrite(), digitalRead(), pinMode(), analogRead(), delay(), etc.

[Library](#): Using and installing Arduino Libraries.

## Structure

loop



## Other Guides

[Parallel to Serial conversion](#): Use a parallel to serial shift register if needing more digital input than there are on your Arduino.

[Serial to Parallel conversion](#): Use a serial to parallel shift register if needing more digital output than there are on your Arduino.

[Capacitance Meter](#): Demonstrates use of RC time constants to measure the value of a capacitor.

[Bit Mask](#): Learn how access specific bits in a byte of data.

Visit the [ITP Physical Computing site](#) for more resources on learning the basics of [Electronics](#), [Sensors](#), [Serial Communication](#), [Motors](#) and more.

## References

[Arduino](#)

## Ass.-3 [Automatic Door](#)

In case of any doubts in this tutorial, contact Ayush Sinha (9850892135), Ramkumar (8778096606) and HimaRa (9340607951)

### Given Components:-

A Motor Driver, 2 ultrasonic sensors, 1 DC motor, jumpers, breadboard and a battery.

#### ● Step 1: The Circuit

- Make the circuit for connecting ultrasonic sensors to Arduino.
- Similarly make the connections for motor driver and DC motor to connect with arduino.
- You can use an online tool for making circuits [TinkerCAD](#). Groups of 4 will be formed and components will be given and you can make your circuit on breadboard.

#### ● Step 2: The Code Steps involved in writing the code:-

- Write the basic header files and void setup function.
- Calculate the distance using basic echo principles as done in assignment no. 2.
- Use the distance calculated to write a simple if statement for actuating the motor in clockwise direction.
- Do it for counterclockwise direction using an else statement.

### References

- [Jeremy | Tutorials](#) for learning arduino.
- [HC-SR04 working , circuit and code](#)
- [Motor Driver](#)