

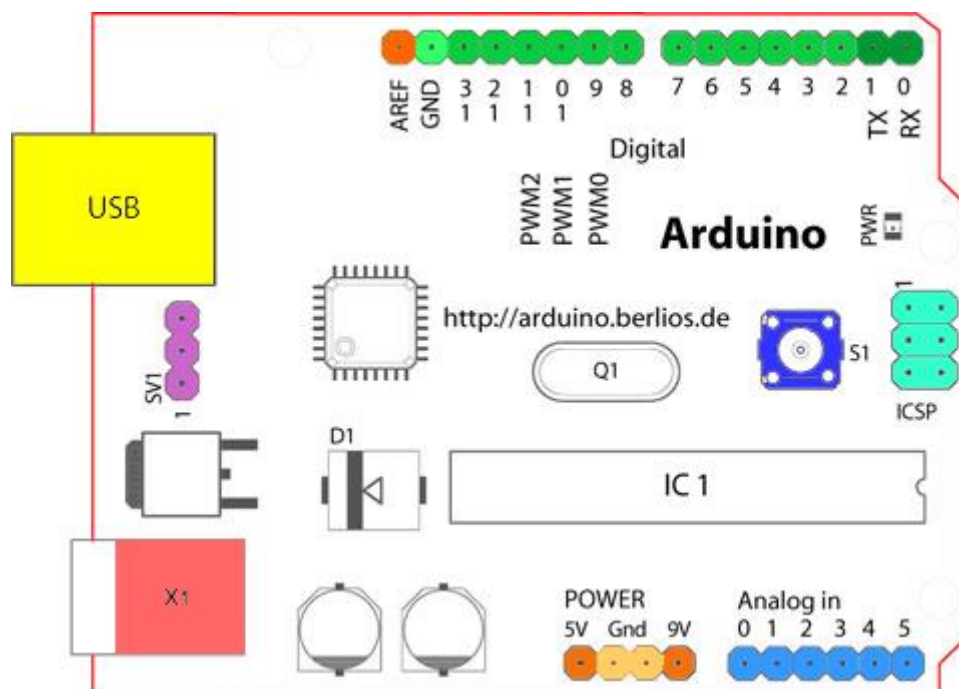


Lecture 2: Introduction to the Arduino Board

Arduino

Arduino is an open-source electronic prototyping platform enabling users to create interactive electronic objects.

Arduino makes it as easy as possible to program tiny computers called **Microcontrollers**.



- Analog Reference pin (orange)
- Digital Ground (light green)
- Digital Pins 2-13 (green)

- Digital Pins 0-1/Serial In/Out - TX/RX (dark green) - *These pins cannot be used for digital i/o (digitalRead and digitalWrite)*
- Reset Button - S1 (dark blue)
- In-circuit Serial Programmer (blue-green)
- Analog In Pins 0-5 (light blue)
- Power and Ground Pins (power: orange, grounds: light orange)
- External Power Supply In (9-12VDC) - X1 (pink)
- Toggles External Power and USB Power (place jumper on two pins closest to desired supply) - SV1 (purple)
- USB (used for uploading sketches to the board and for serial communication between the board and the computer; can be used to power the board) (yellow)

Types of Arduino pins

There are three main types of Arduino pins: **digital pins**, **analog pins** and **power pins**.

Digital pins can be used as both input and output pins on most Arduino microcontrollers. This means that a single pin can both send and receive digital data. Digital Arduino pins can be only "high" or "low." When a digital pin is high, a constant voltage is present. When the pin is low, this voltage is gone.

Switching a light emitting diode (LED) on and off is one common instance of a digital output, and checking the status of a button is an example of a digital input.

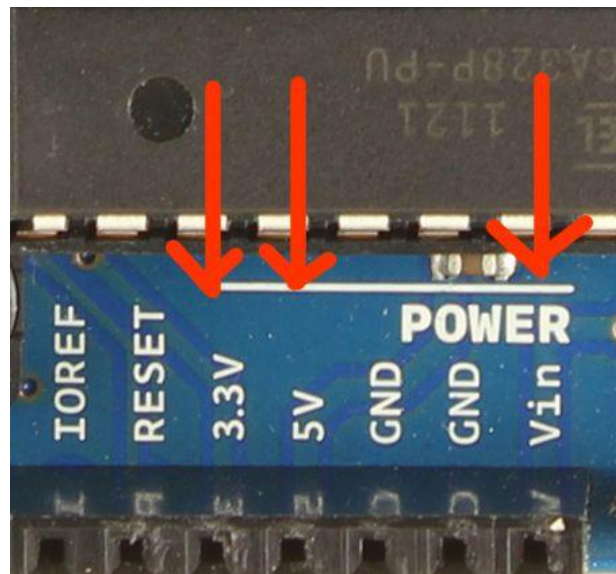
Analog input pins are used to read the state of a connected device, such as a sensor. Although digital pins can only tell if a connection is high or low, analog input pins are able to read the exact voltage value. This is useful for interfacing with analog sensors that have a wide range of possible values. For example, analog thermometer sensors often send a voltage reading that corresponds to the temperature. Using an analog input pin, an Arduino can read the value directly.

NOTE

The analog input pins can be used as digital pins, referred to as A0, A1, etc. The exception is the Arduino Nano, Pro Mini, and Mini's A6 and A7 pins, which can only be used as analog inputs.

Power Pins

Most Arduino microcontrollers have several power pins that receive and output a variety of voltages.



The Vin, 5V, 3.3V, and GND pins are Arduino power pins. You can use the **Vin pin** to Power your Arduino with an unregulated 7 to 12-volt power source. Like a 9V battery or a wall adapter that is in the range of 7 to 12 volts.

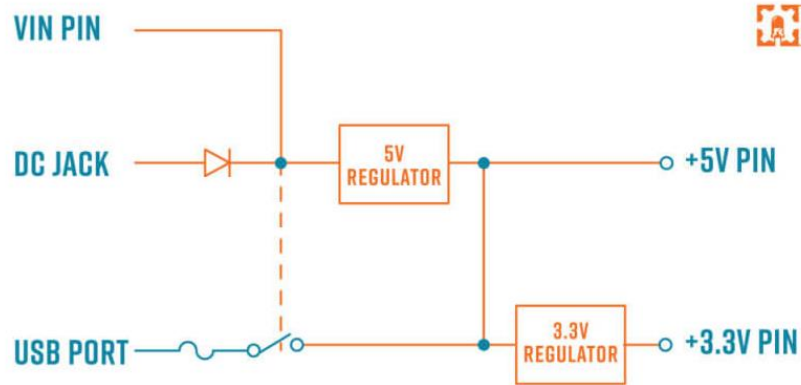
Alternatively, you can power your Arduino through the **5V pin** with an external regulated 5V power supply.

A **3.3V** pin to power sensors and modules that need 3.3V power.

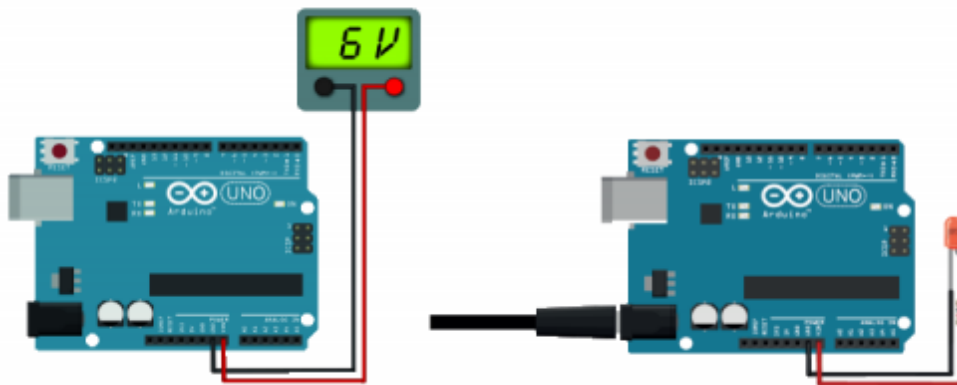
There are several **GND pins** on the Arduino, any of which can be used to ground your circuit. The **GND pins** are used to close the electrical circuit and provide a common logic reference level throughout your circuit.

How to supply the power to an Arduino board?

The board can be supplied with power either from the **VIN pin of the board (7-12V)**, the **DC power jack (7 - 12V)**, or the **USB connector (5V)**.

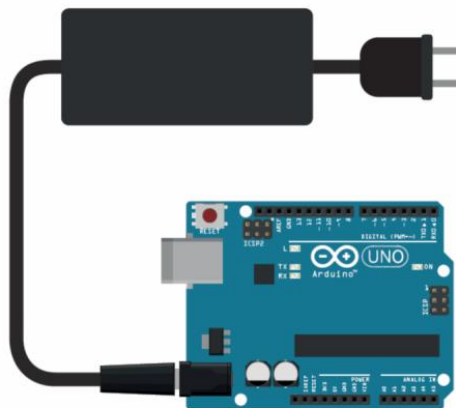


Vin pin



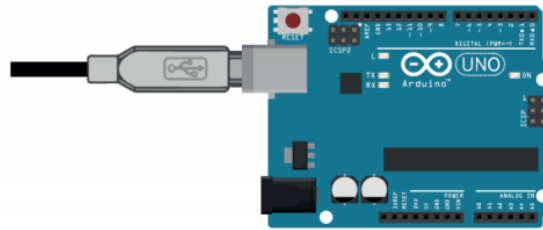
The Vin socket is used as input or as output

DC Jack



Powering Arduino by means of a JAPAN JACK socket

USB port



Powering Arduino by means of USB Port

External Power Sources

USB: that's probably the most common way people try their projects. By simply plugging your Arduino board into your computer through a USB cable it will power up your Arduino. The 5V USB power is used to power a 5V Arduino directly, or it's regulated down if it's a 3.3V Arduino.



Batteries: You can use two different types of batteries:

Disposable batteries: One of the most common ways is to power up an Arduino with a 9V battery, just like the image below shows. Keep in mind that those batteries are a bit limited, so your Arduino won't be able to supply many sensors or motors.



Re-charged batteries: Those are simply battery cells that you can recharge. Are a great solution for robotics.



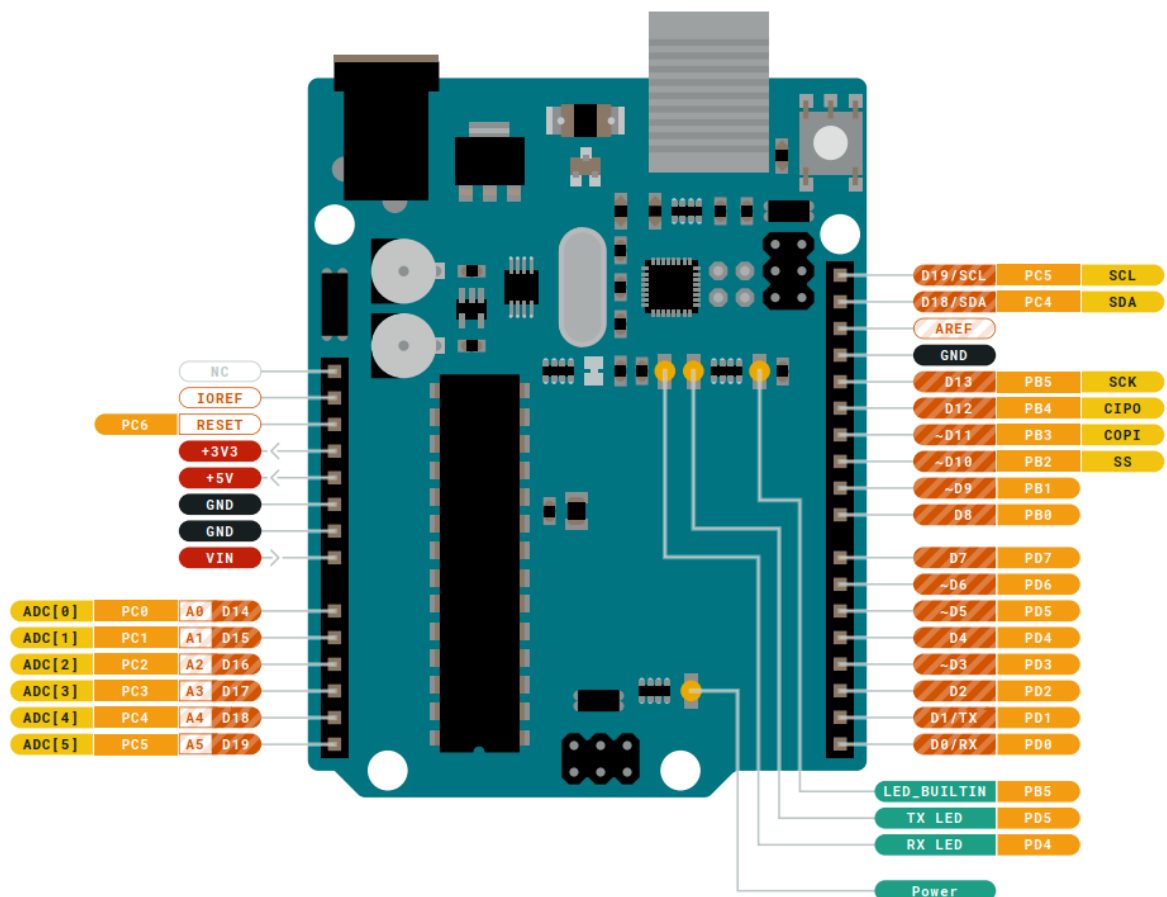
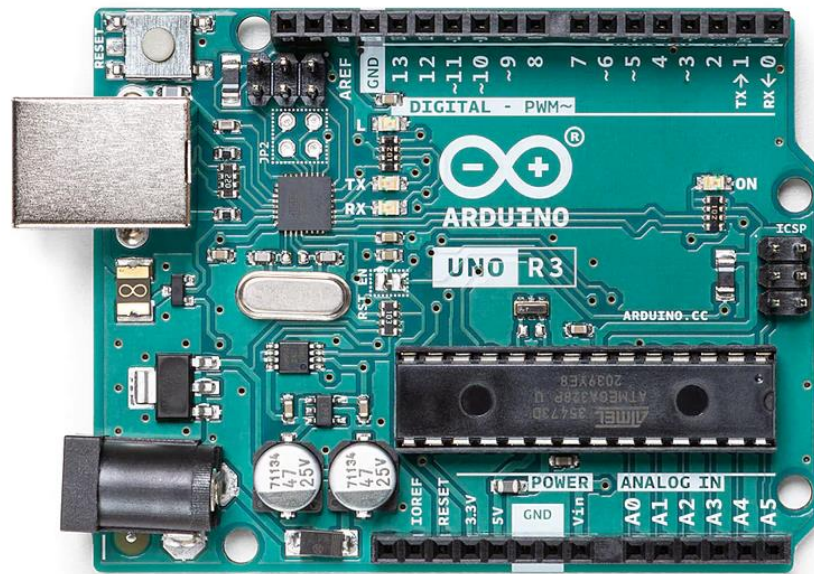
Power Adapter You must be careful when you're using a power adapter because sometimes they produce much more voltage than the actually power supply says in the label. And the voltage regulator on your Arduino won't be able to handle so much voltage.



Types of the Arduino

ARDUINO UNO

MICROCONTROLLER	ATmega328P
DIGITAL I/O PINS	14 (of which 6 provide PWM (Pulse Width Modulation) output)
PWM DIGITAL I/O PINS	6
ANALOG INPUT PINS	6

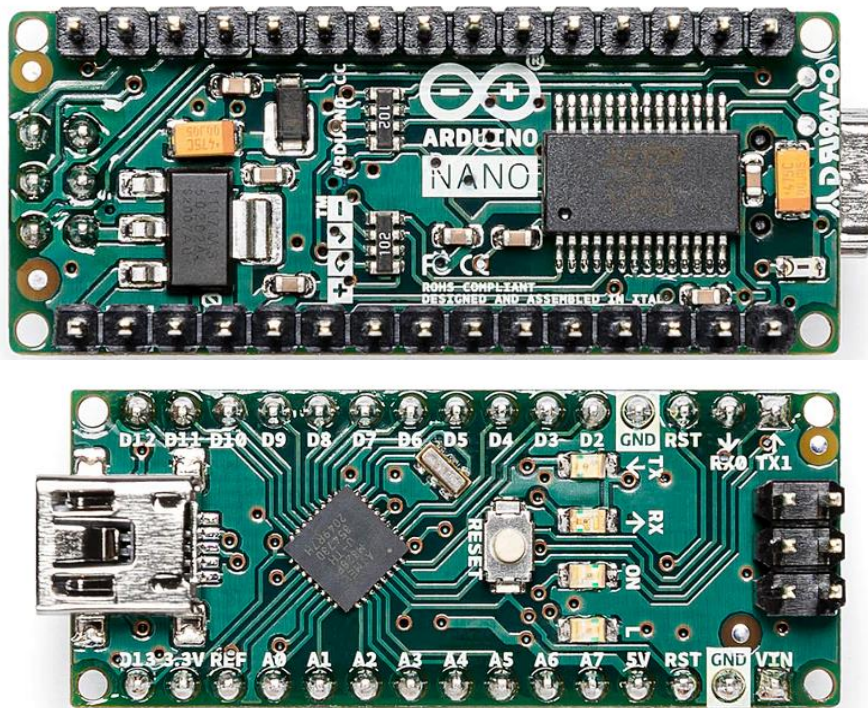


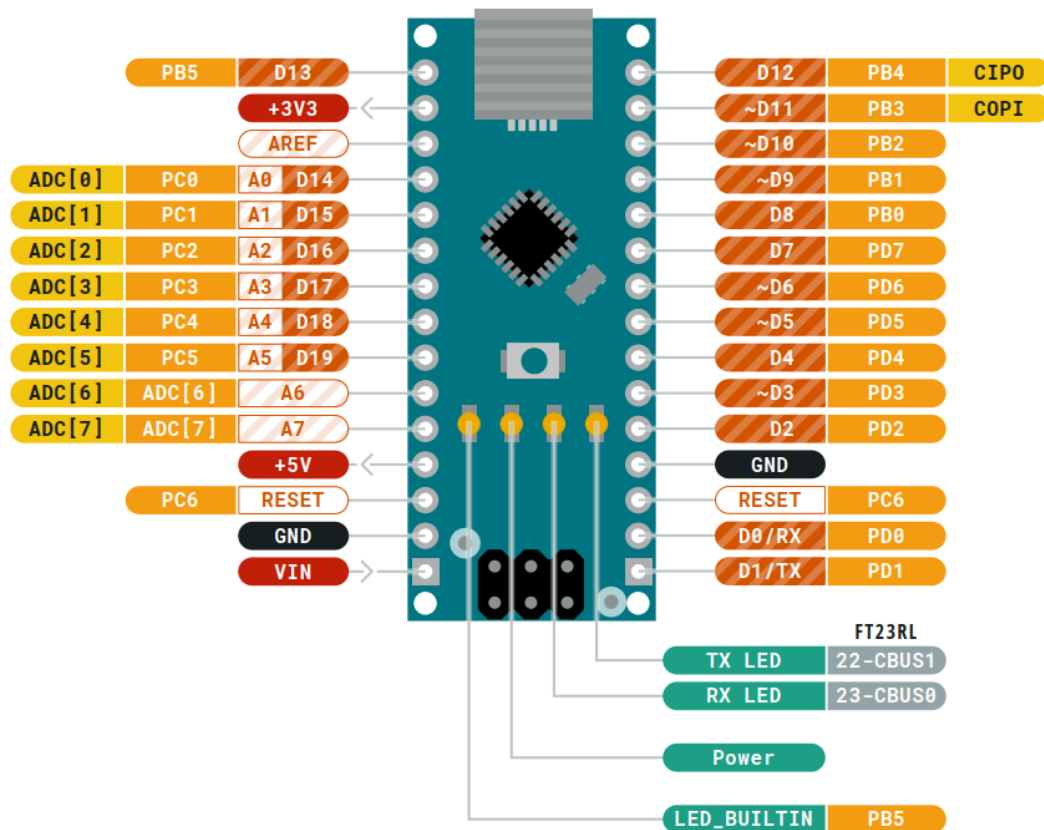
Note

A pulse width modulation signal is a type of analog modulation signal. We use it to generate analog signals using the digital signals as input.

ARDUINO NANO

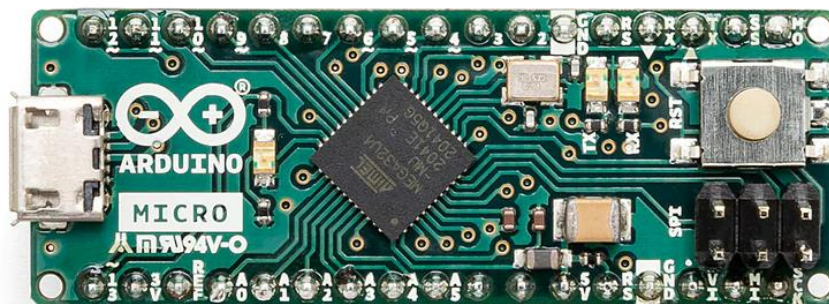
MICROCONTROLLER	ATmega328
ANALOG IN PINS	8
DIGITAL I/O PINS	22

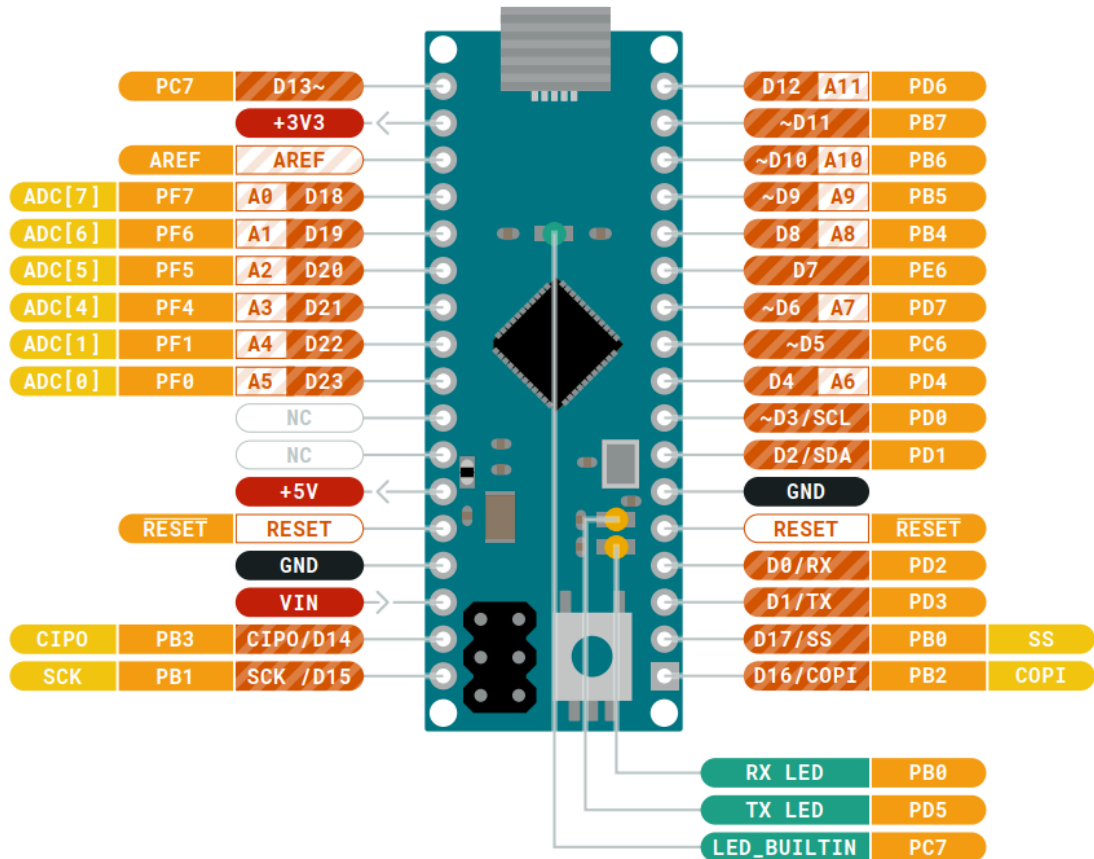




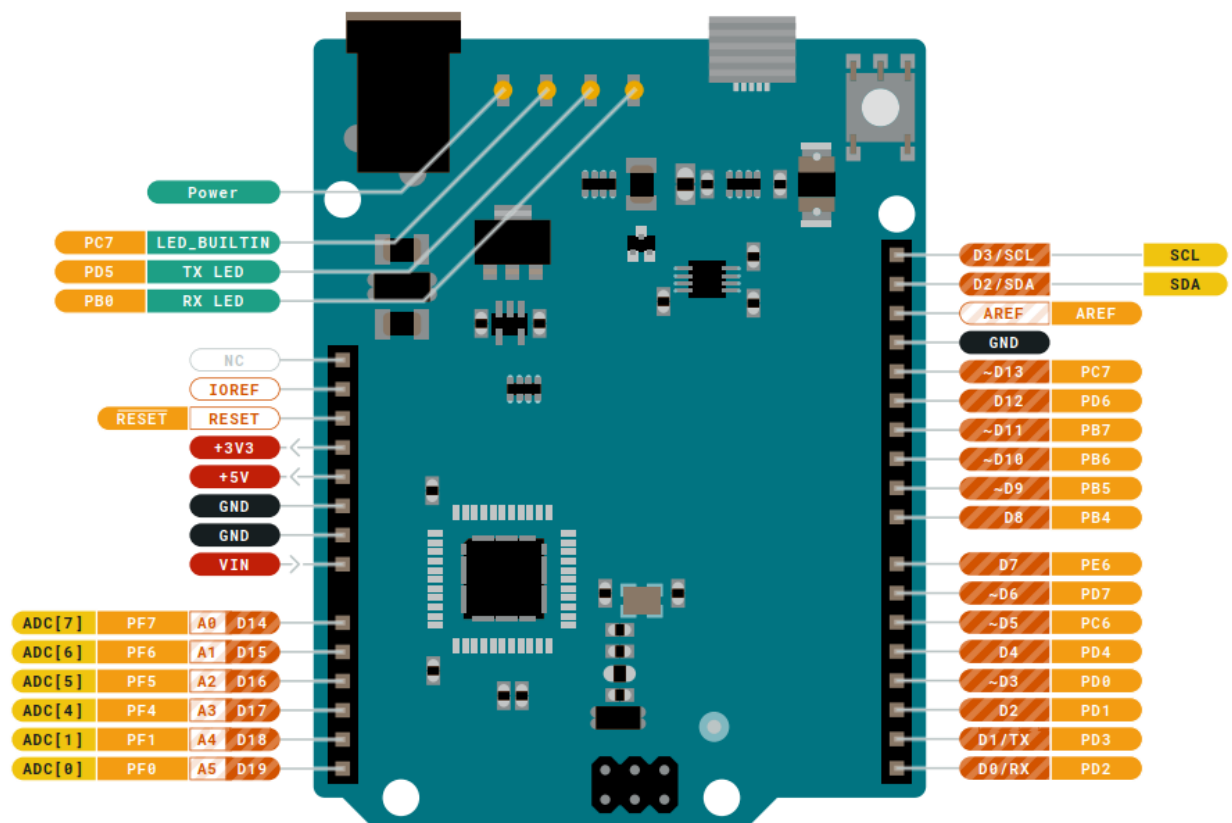
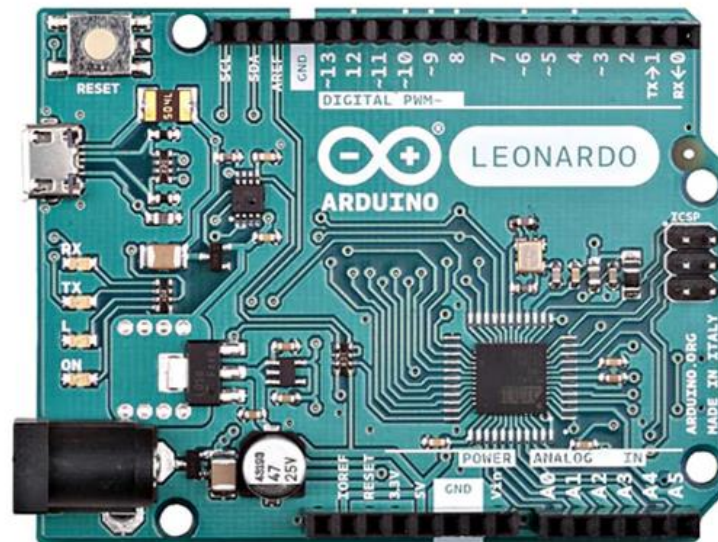
ARDUINO MICRO

MICROCONTROLLER	ATmega32U4
DIGITAL I/O PINS	20
PWM CHANNELS	7
ANALOG INPUT CHANNELS	12





MICROCONTROLLER	ATmega32u4
DIGITAL I/O PINS	20
PWM CHANNELS	7
ANALOG INPUT CHANNELS	12



ARDUINO MEGA

MICROCONTROLLER	ATmega2560
DIGITAL I/O PINS	54 (of which 15 provide PWM output)
ANALOG INPUT PINS	16

