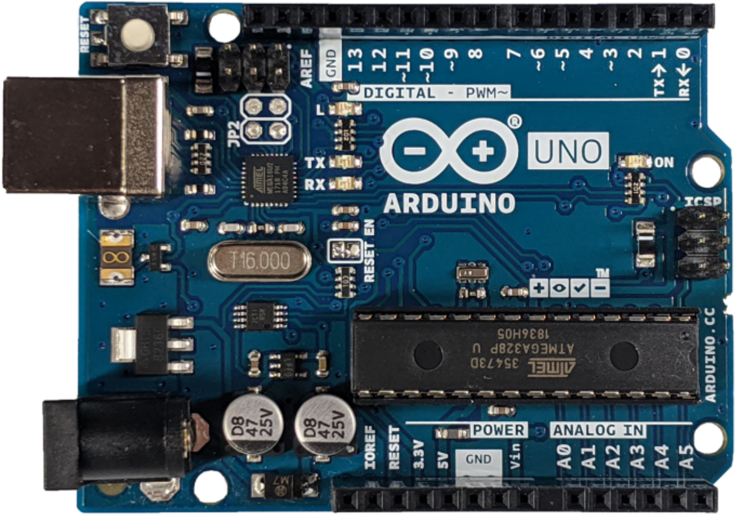
Arduino

Internet of Things, Lecture-10

Rahul Shandilya

Introduction

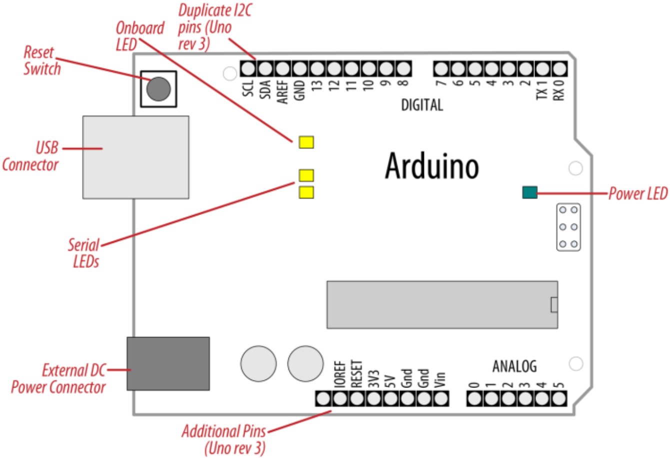
Arduino is one of the most popular development ecosystem for microcontroller which is supported by an easy-to-use programming environment that allows us to quickly start developing software.



Ecosystem

There are three main components for your arduino project.

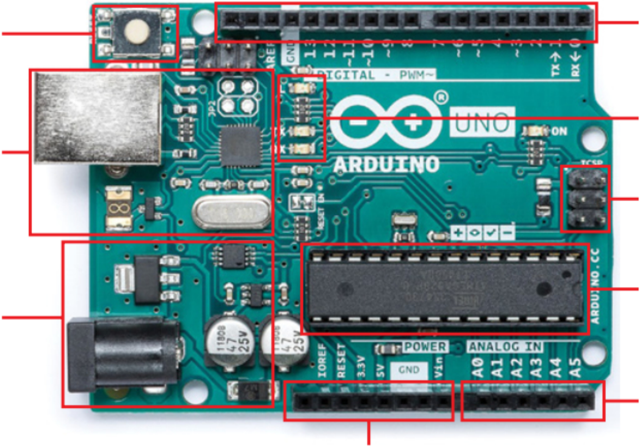
- First-party or third-party Arduino boards
- External hardware (including both *shields* and manually created circuits)
- The Arduino integrated development environment, or Arduino IDE



Board Functionality

All Arduino boards have a few key capabilities and functions.These are some functional groups that you'll be concerning yourself with:

- **Microcontroller:** At the heart of every Arduino is a microcontroller. This is the brain of Arduino
- **Programming:** Programming interfaces enable you to load software onto your Arduino.
- **I/O:** Input/Output (I/O) circuitry is what enables your Arduino interface with sensors, actuators, etc.
- **Power:** There are a variety of ways to supply power to an Arduino. Most Arduino boards can automatically switch between power from multiple sources (such as USB and a battery.



Programming & I/O: Reset Button

Programming & Power: USB-to-Serial Interface

Power: 7-12VDC Input and Voltage Regulator

I/O: General Purpose, PWM, and Communication Buses

I/O: Debug LED and USB LEDs

Programming: In-Circuit Serial Programming (ICSP) Header

Microcontroller: ATmega 328P

I/O: Analog-to-Digital (ADC) Inputs

Power: Voltage and GND pins

Microcontroller

The Arduino's microcontroller is responsible for holding all your compiled code and executing the commands you specify. The Arduino programming language gives you access to microcontroller peripherals, including analog-to-digital converters (ADCs), general-purpose input/output (GPIO or just I/O) pins, communication buses (including I2C, SPI, UART, and others), and serial/USB interfaces.

- The most popular Arduino UNO uses ATmega328p microcontroller from Atmel (now MicrochipTM) based on the AVR architecture, which is the large chip with 28 legs.
- It is a controller with 8-bit-wide registers, and operates at a clock frequency of 16 MHz.
- It has 32 kB RAM memory and 1 kB non-volatile EEPROM memory, which can be used to store persistent data that need to survive tuning off and on the supply voltage.
- Most Arduino boards come with a debug LED already connected to pin 13, which enables you to run your first program (blinking an LED) without connecting any additional circuitry.

Programming Interfaces

Ordinarily, microcontroller programs are written in C or assembly, and programmed via the In-Circuit Serial Programming (ICSP) interface using a dedicated programmer. The most important characteristic of an Arduino is that you can program it directly using only an ordinary USB cable.

- It is the Arduino bootloader which made possible it to program using USB directly.
- The bootloader is loaded onto the microcontroller at the factory (using the ICSP header), which allows a serial USART (Universal Synchronous/Asynchronous Receiver/Transmitter) to load your program on the Arduino without using a separate programmer.
- In the case of the Arduino Uno and Mega 2560, a secondary microcontroller (an ATmega16U2 or ATmega8U2, depending on your revision) serves as an interface between a USB cable and the serial USART pins on the main microcontroller.
- The Arduino Leonardo, which uses an ATmega32U4 as the main microcontroller, has USB incorporated, so a secondary microcontroller is not needed.

Input/Output: GPIO, ADCs, and Communication Busses

The part of the Arduino that we care most about during projects is the general-purpose Input/Output (GPIO) and ADC pins. All of these pins can be individually addressed via the programs we write.

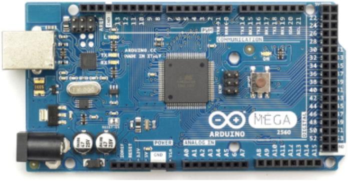
- The UNO interacts with its environment through 13 digital input-output (IO) pins, of which most can be configured to be either input or output, and have software-configurable pull-up resistors.
- There are six analog input pins. They measure voltages of up to the supply voltage of 5 V. An alternative internal reference voltage source provides a 1.1 V reference.
- Several of the pins are configurable to support I2C, SPI, and RS-232 communication.
- Furthermore, the built-in hardware RS-232 port is connected to an RS-232-to-USB converter that allows communication and programming from a host computer.
- There is no WiFi, Bluetooth, or Ethernet support on the UNO board, but extension boards, so-called *shields*, are available

Power

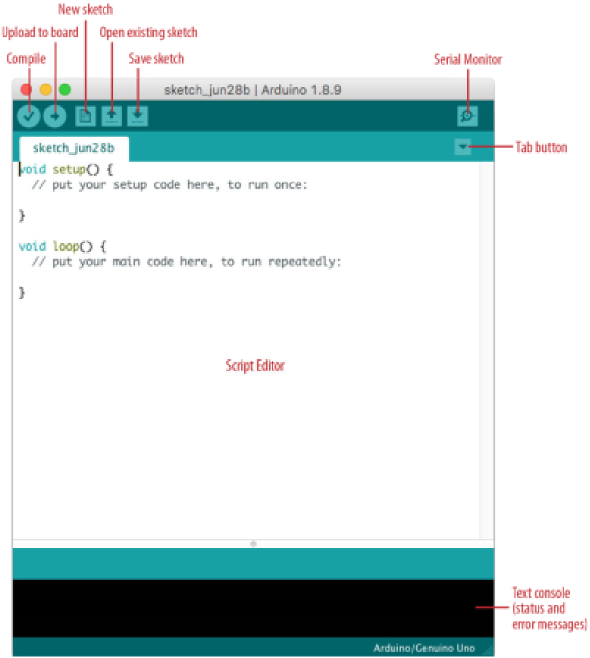
For most of your projects, you will simply use the 5V power that is provided over your USB cable. However, when you're ready to untether your project from a computer, you have other power options.

- Most Arduinos can accept between 6V and 20V (7V to 12V is the recommended voltage supply range) via the direct current (DC) barrel jack connector.
- Some Arduinos operate at 5V logic levels, and others operate at 3.3V logic levels.
- 5V is used for all the logic on the Uno board. In other words, when you toggle a digital I/O pin, you are toggling it between 5V and 0V
- 3.3V is broken out to a pin to accommodate 3.3V shields and external circuitry.

Variants of Arduino



Arduino IDE



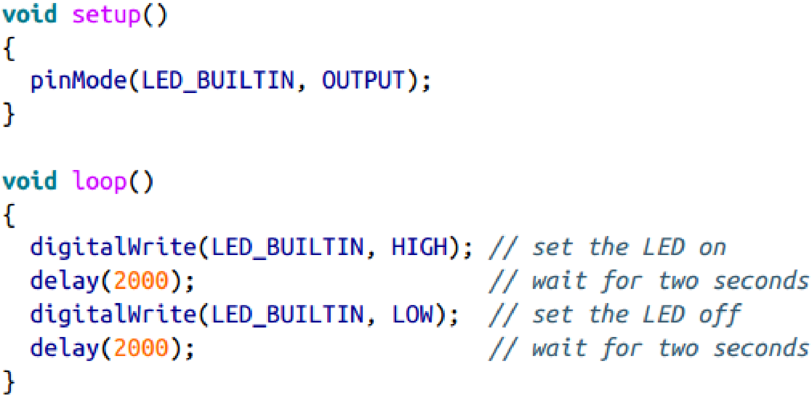
```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Arduino Program



```
void setup()
{
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop()
{
  digitalWrite(LED_BUILTIN, HIGH); // set the LED on
  delay(2000);                     // wait for two seconds
  digitalWrite(LED_BUILTIN, LOW);  // set the LED off
  delay(2000);                     // wait for two seconds
}
```