

Introduction to Arduino

ROCHESTER MAKERSPACE

2021



Class Objectives

1. Become familiar with Arduino hardware and software
2. Be aware of the range of Arduino-supported boards and how to choose one for your project
3. Understand how to connect and operate Arduino hardware from a PC or Mac
4. Understand how to create and run a program on an Arduino
5. Understand how to control a simple circuit from an Arduino
6. Get a starter list of resources for learning more
7. Be excited by the possibilities!

Class notes and code

All the class notes and code can be downloaded from GitHub:

- <https://github.com/RochesterMakerSpace/ArduinoClass>

Pull down the green “Code” button and select “Download ZIP”

Computers, Microcontrollers, Arduino

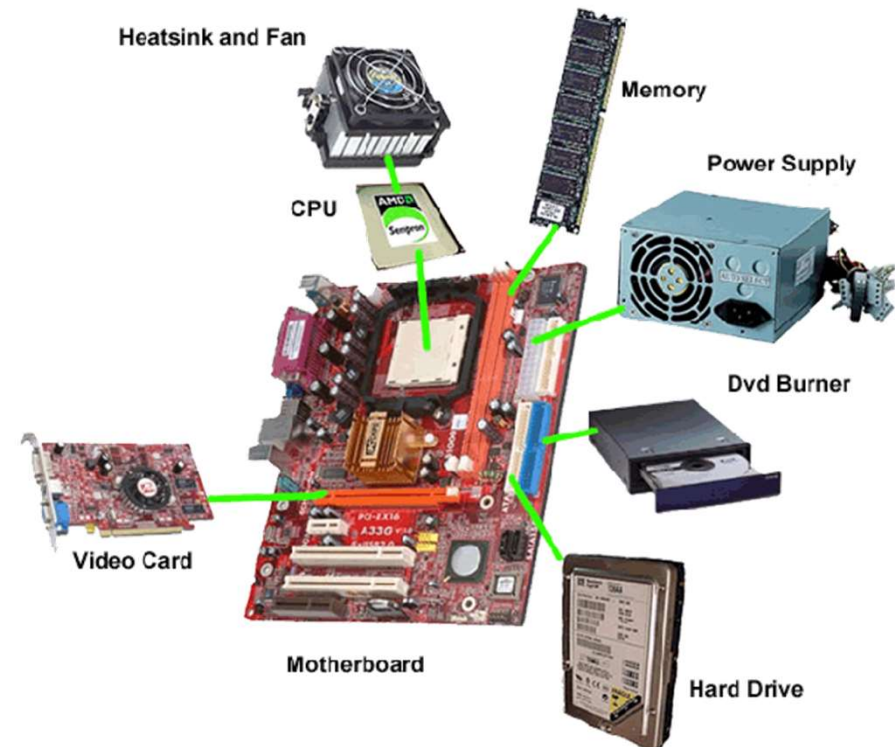
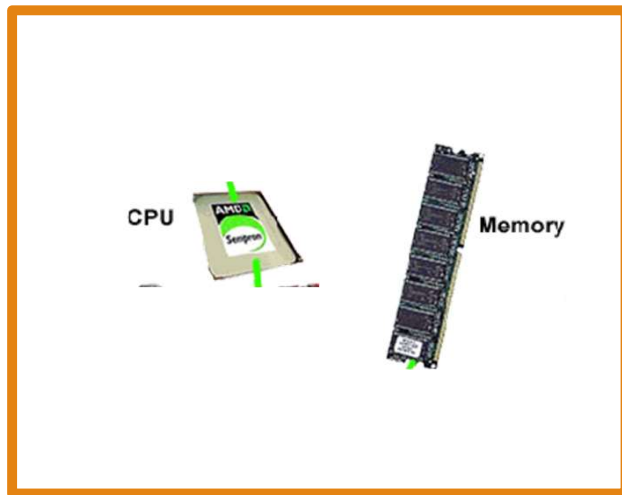
Conventional computers can be described by 5 main components:

- CPU – the Central Processing Unit executes instructions
- Program memory – the instructions
- Data memory – the data
- I/O interfaces and devices – connecting disks, screens, keyboards, mice, etc.
- Software - Operating system, utility programs, applications

Microcontrollers are a computers on a chip typically including a CPU, and program and data memory with connectors for General Purpose Input and Output (GPIO).

Arduino is an open-source board design, originally designed in 2006, that is combined with a free, basic development environment

Microcontrollers → Computer systems



Arduino Uno R3

The canonical Arduino design

Focus is on experimentation and learning

A simple, low-cost, small computer

- Genuine: \$22, Clone: \$11
- Modest processing power (16 Mhz)
- Small space for code (32KB)
- Small space for data (2KB)
- Wide range of GPIO connectivity options for devices or circuits
- Easy USB connection and good, free software development environment

Huge community of 'makers' providing videos, tutorials, examples, projects, devices, advice



Many Arduino variants

Faster processor

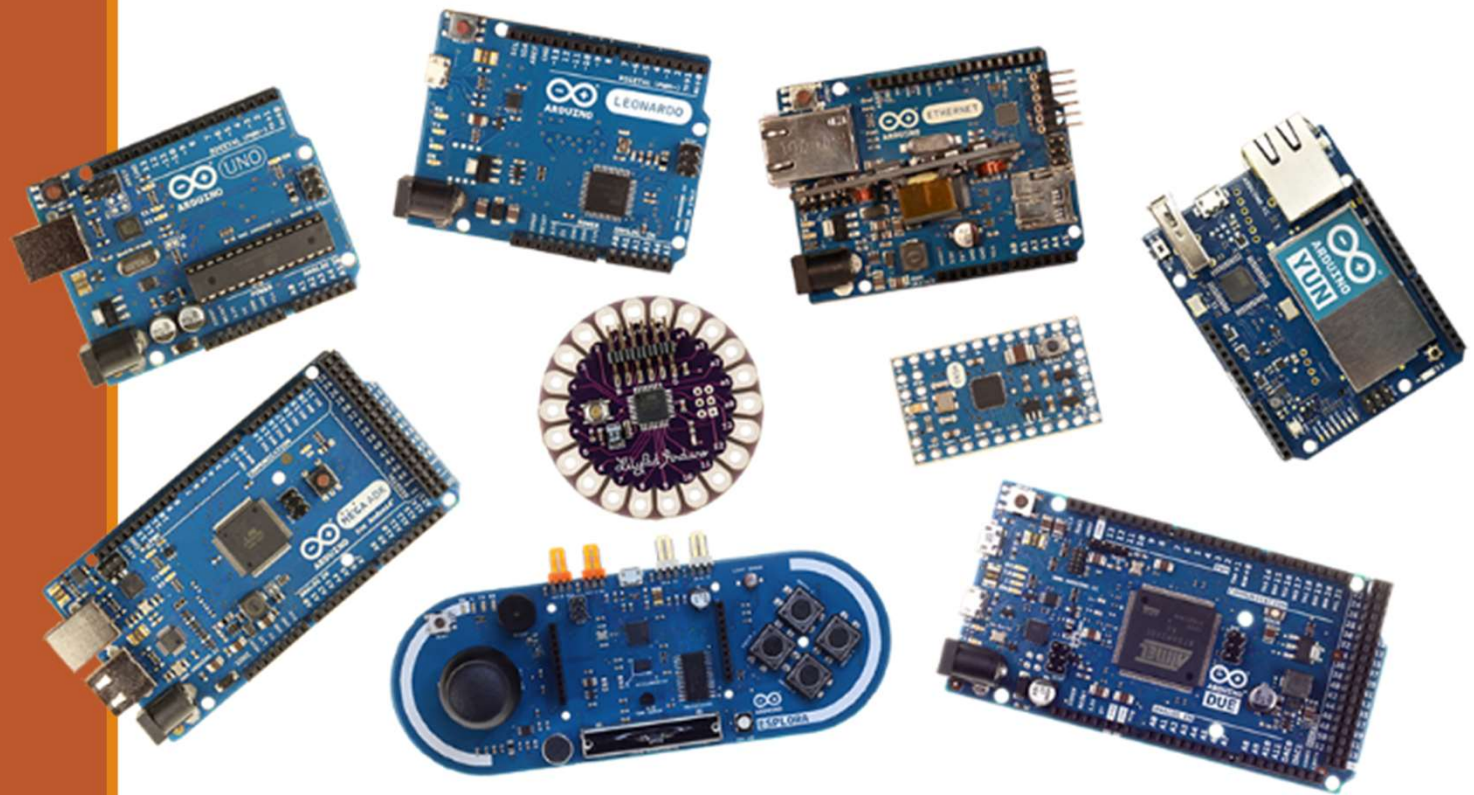
Bigger programs

More data

More pins to connect devices

More portable

Different form factor



New Arduino Nanos

Arduino Nano Every \$11.90

- ATmega4809 20 Mhz
- 48 KB Flash, 6 KB RAM

Arduino Nano 33 IoT \$18.40

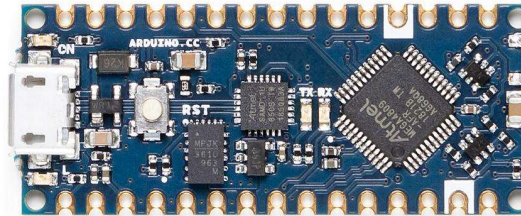
- SAMD21, 32-bit, 48 Mhz
- 256 KB Flash, 32 KB RAM
- WiFi, Bluetooth, BLE, 6 Axis IMU

Arduino Nano 33 BLE \$20.20

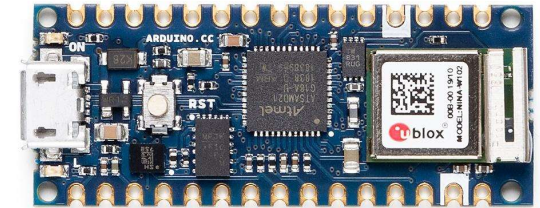
- nRF52840, 32-bit, 64 Mhz
- 1 MB Flash, 256 KB RAM
- Bluetooth, BLE, 9 Axis IMU

Arduino Nano 33 Sense \$31.10

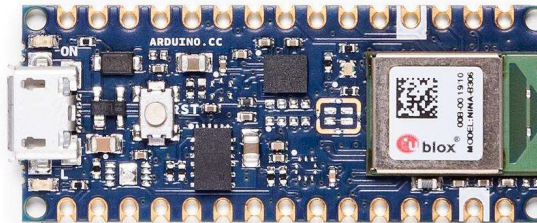
- nRF52840, 32-bit, 64 Mhz
- 1 MB Flash, 256 KB RAM
- Bluetooth, BLE, 9 Axis IMU
- Humidity, Temp, Pressure, Mic, gesture ,proximity, light sensors



Arduino Nano Every



Arduino Nano 33 IoT



Arduino Nano 33 BLE



Arduino Nano 33 Sense

Arduino GPIO

Simple direct connection for digital input and output

Simple direct connection for analog input

Onboard pulse width modulation (PWM)

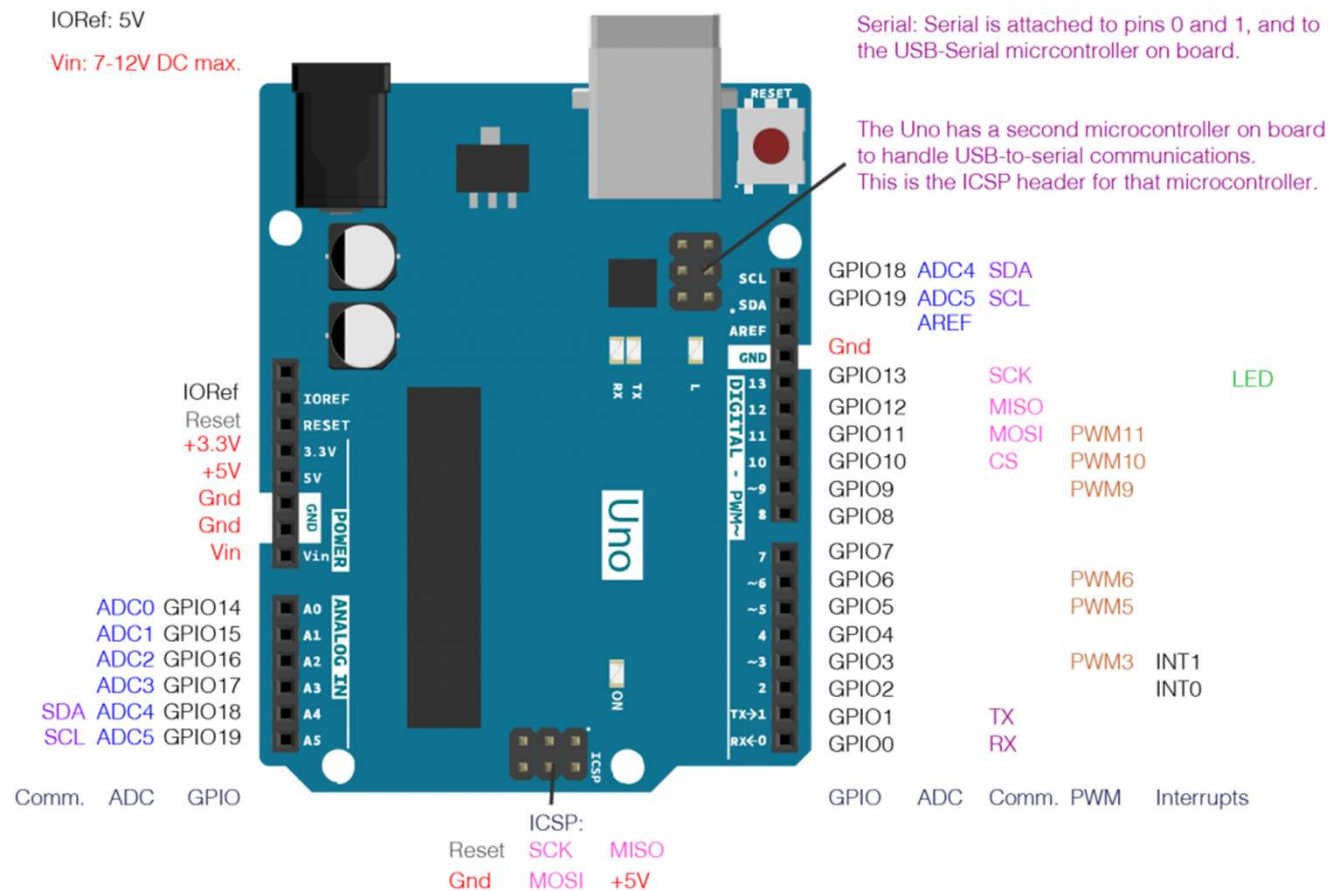
4 ways to connect to other chips:

GPIO – Digital I/O or Analog Input

I2C – Inter-Integrated-Circuit

SPI – Serial Peripheral Interface

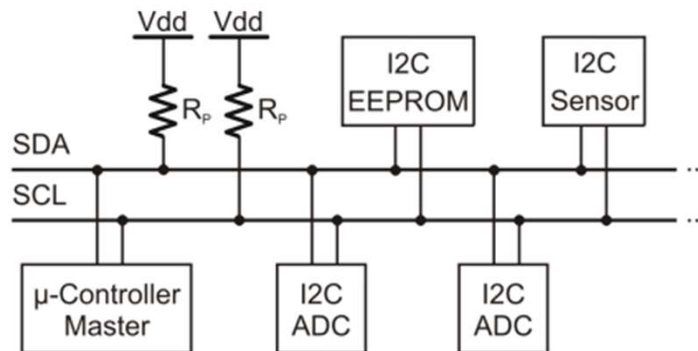
Serial – asynchronous serial



I2C

I²C (Inter-Integrated Circuit), pronounced I-squared-C , is a synchronous , multi-master, multi-slave , packet switched , single-ended , serial computer bus invented in 1982 by Philips Semiconductor (now NXP Semiconductors). It is widely used for attaching lower-speed peripheral ICs to processors and microcontrollers in short-distance, intra-board communication. Alternatively I²C is spelled I2C (pronounced I-two-C) or IIC (pronounced I-I-C).

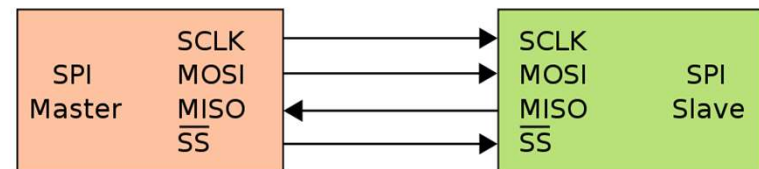
Wikipedia



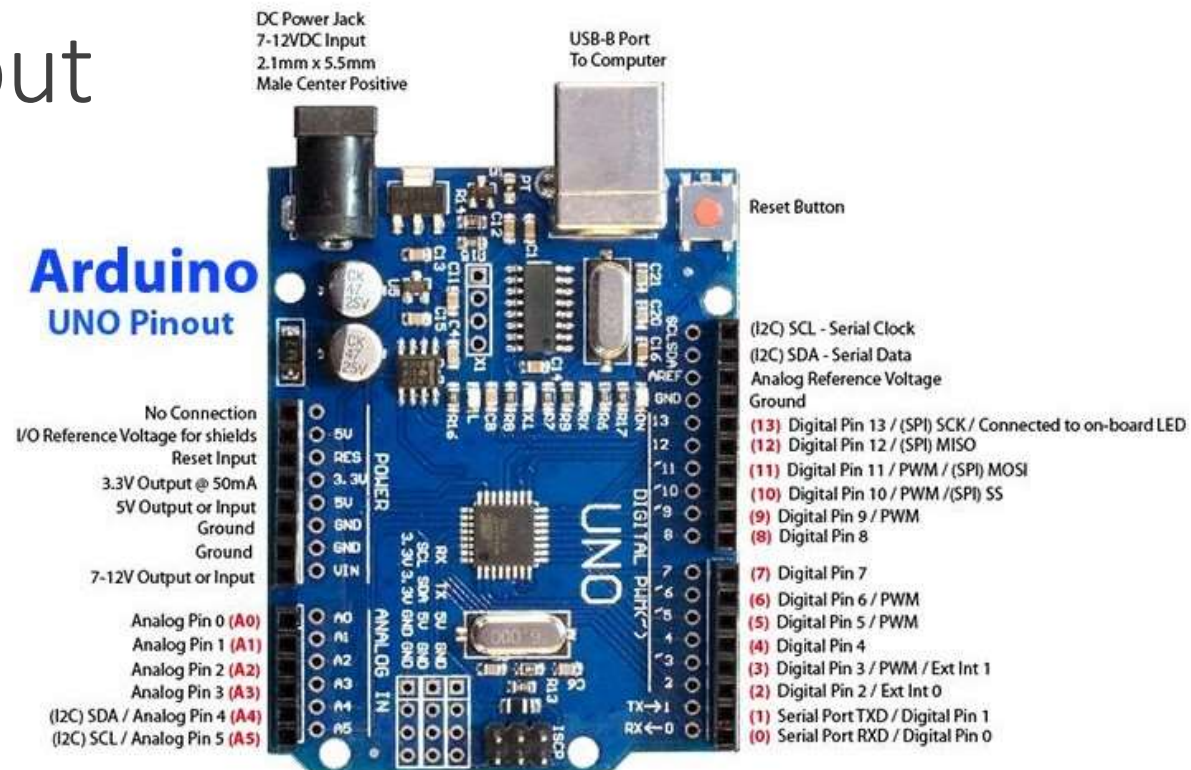
SPI

The **Serial Peripheral Interface (SPI)** is a [synchronous serial communication](#) interface specification used for short distance communication, primarily in [embedded systems](#). The interface was developed by [Motorola](#) in the mid 1980s and has become a [de facto standard](#). Typical applications include [Secure Digital](#) cards and [liquid crystal displays](#).

Wikipedia



Uno R3 Pinout



Red numbers in paranthesis are the name to use when referencing that pin.
Analog pins are references as A0 thru A5 even when using as digital I/O

Arduino Integrated Development Environment (IDE)

Free download from
<https://www.arduino.cc/en/Main/Software>

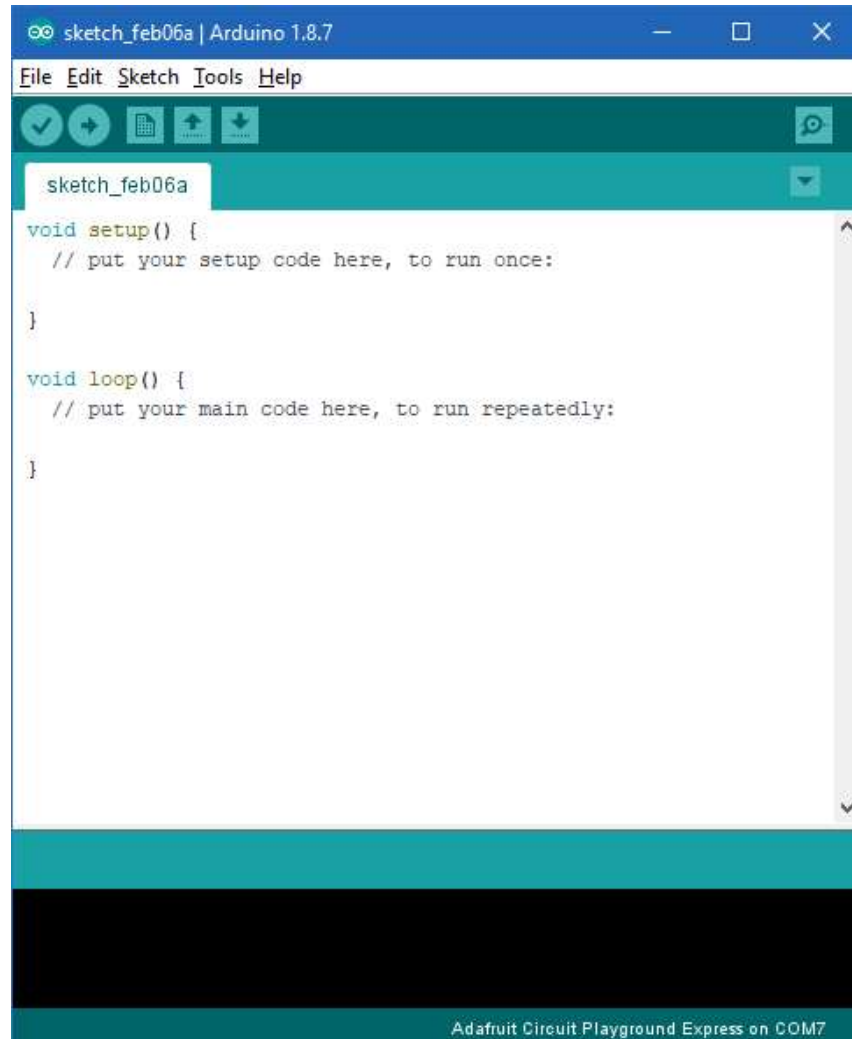
Simple, fixed program structure

Uses a programming language
that is a simplified variant of c++

Support for other processors,
ESP8266, STM32, Pi Pico, etc.

Alternatives:

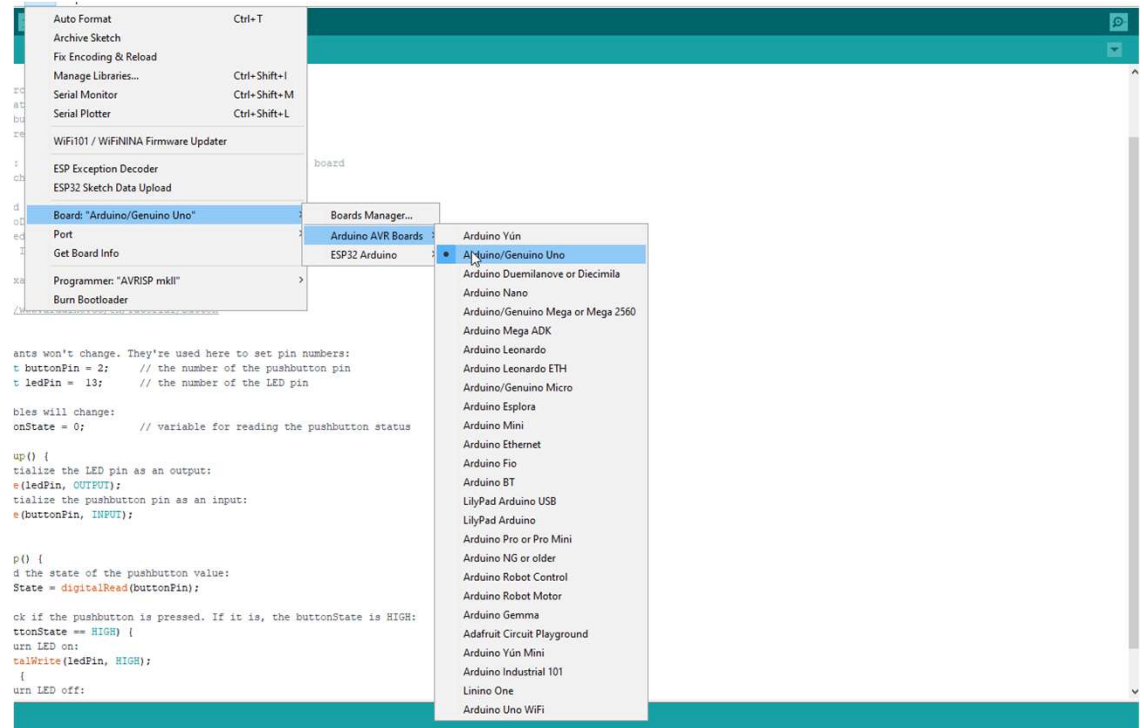
- Arduino IDE 2.0
- Platform IO (VS Code/Atom)



Arduino Integrated Development Environment (IDE)

Change board type

- Arduino UNO or Arduino UNO clone
 - Tools | Board | Arduino AVR Boards | Arduino\Genuine UNO
- Arduino Leonardo or Arduino Leonardo clone (Velleman – white board)
 - Tools | Board | Arduino AVR Boards | Arduino Leonardo

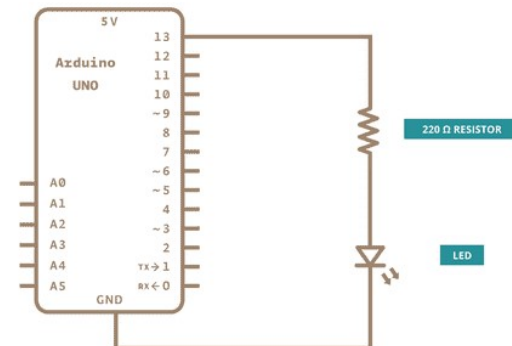
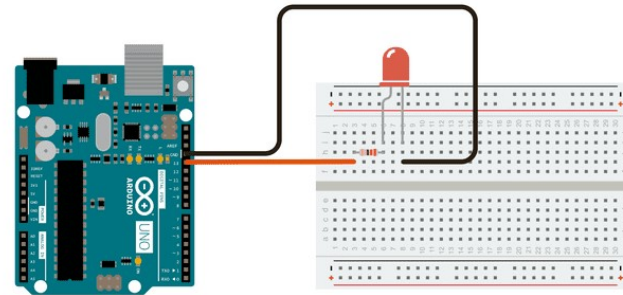


Sketches and Circuits

Blink (File | Examples | 01.Basics | Blink)

- Focus: basic code of a sketch
- Experiment: change the rate of blinking

Can use Built In LED connected to pin 13 instead of external LED

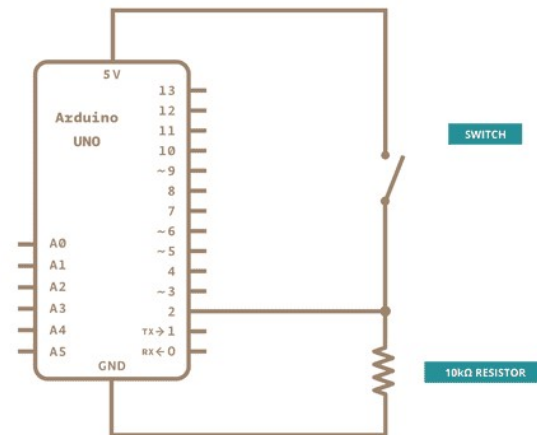
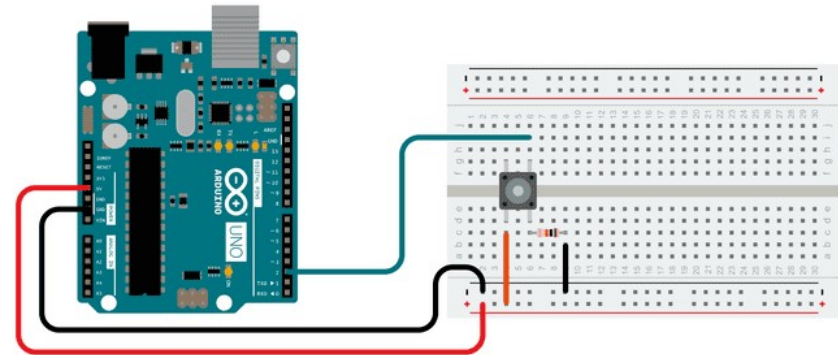


Sketches and Circuits

Button (File | Examples | 01.Digital | Button)

- Focus: breadboarding, variables
- Experiment: reverse the effect of a button press

Can use Built In LED connected to pin 13

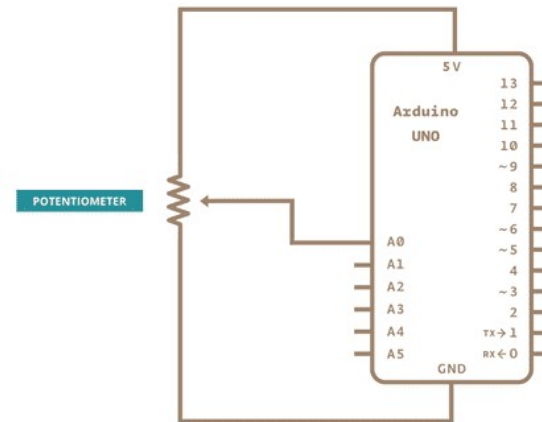
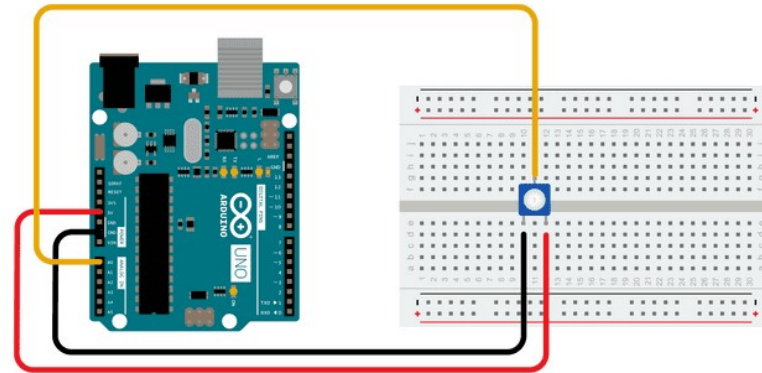


Sketches and Circuits

ReadAnalogVoltage

(File | Examples | 01.Basics | ReadAnalogVoltage)

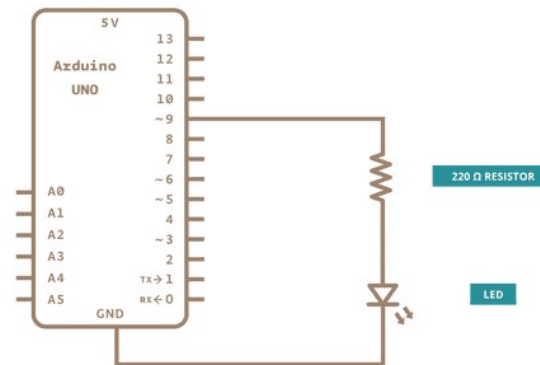
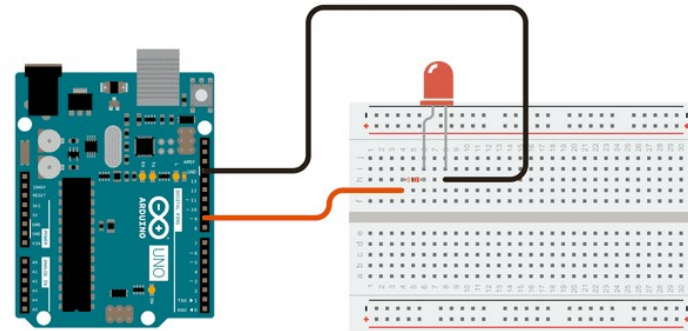
- Focus: analog input, monitor/plotter
- Experiment: convert to use the 3.3v pin



Sketches and Circuits

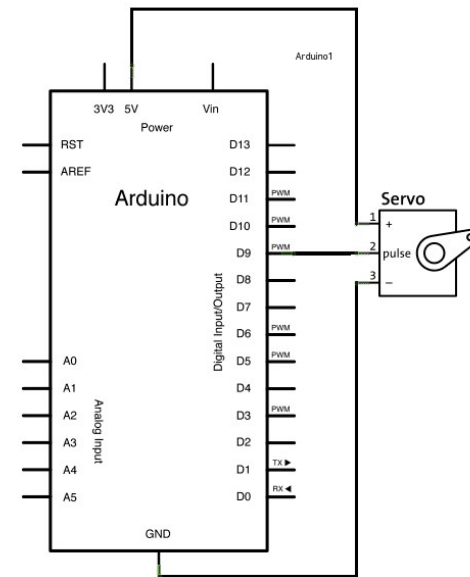
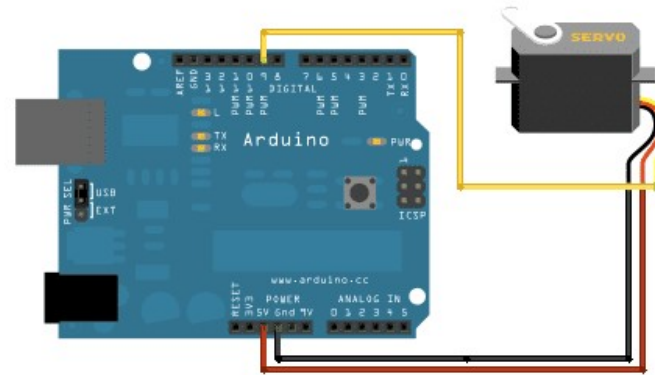
Fade (File | Examples | 01.Basics | Fade)

- Focus: PWM
- Experiment: change the variables



Sweep (File | Examples | Servo | Swee[)

- Focus: #includes, motor control
- Experiment: restrict the servo range of motion

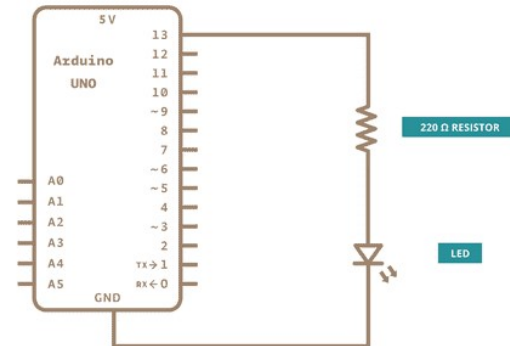
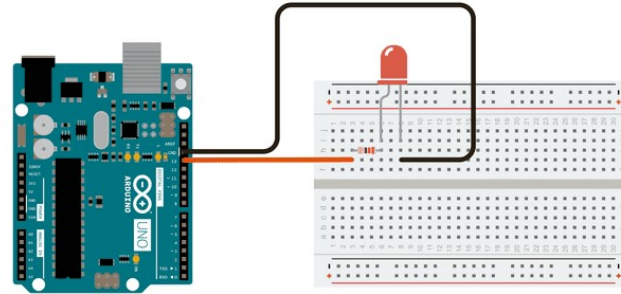


Sketches and Circuits

BlinkWithoutDelay
(File | Examples | 02.Digital | BlinkWithoutDelay)

- Focus: code execution timing
- Experiment: print to the serial monitor at the same time

Can use Built In LED connected to pin 13 instead of external LED

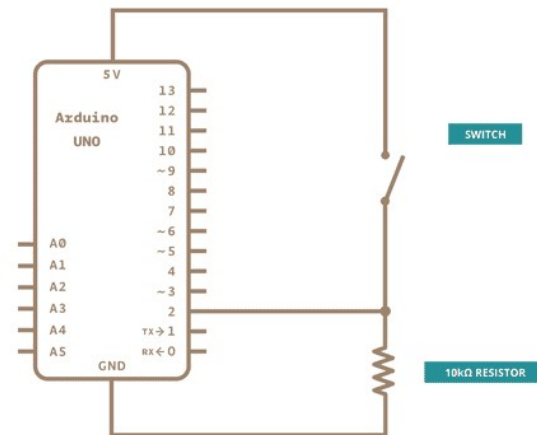
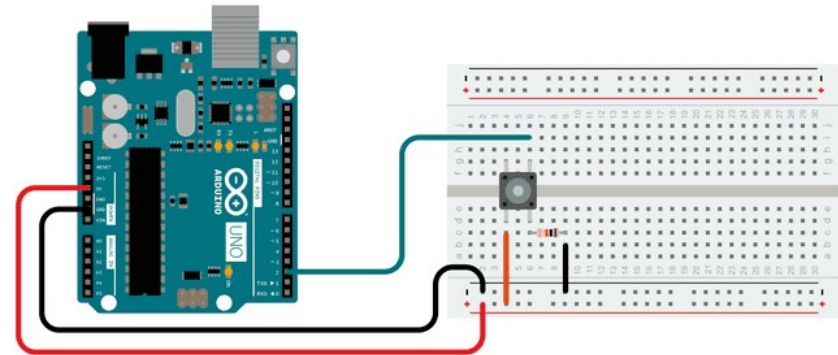


Sketches and Circuits

Debounce
(File | Examples | 02.Digital | Debounce)

- Focus: the mechanical world
- Experiment: how short can your delay be without flickering the led?

Can use Built In LED connected to pin 13 instead of external LED



Resources

<https://www.instructables.com/id/Arduino-Projects/>

A great source of inspiration

Shows many cool projects you can accomplish with an Arduino

Introduction to Arduino: A piece of cake!

Alan G. Smith (alan@introtoarduino.com)

Hardcopy available at <http://www.amazon.com>

The most recent PDF is free at <http://www.introtoarduino.com>

<https://www.arduino.cc>

The official web site for Arduino

Tutorials, documentation, example projects, shop

<https://www.adafruit.com>

A DIY site loaded with Arduino and Raspberry Pi products

Tutorials, step-by-step instructions, example projects, shop

<https://www.sparkfun.com/>

An electronics retailer with lots of Arduino and Raspberry Pi products

<https://www.pololu.com/>

An online retailer with lots of robotics components

<https://www.seeedstudio.com/>

An online retailer of project components

<https://blog.tinkercad.com/official-guide-to-tinkercad-circuits/>

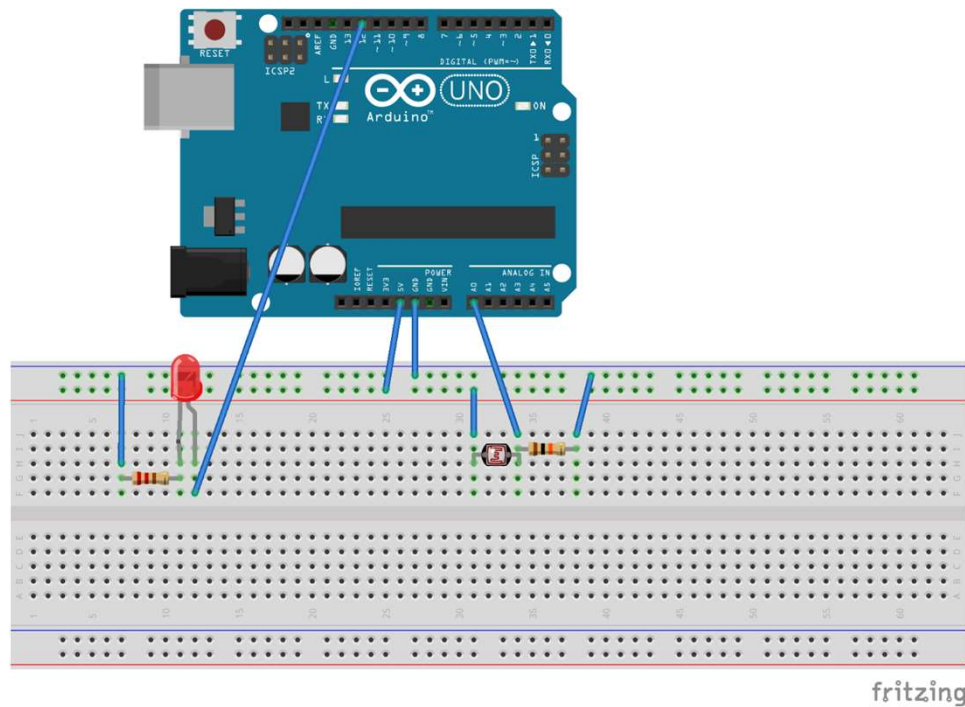
Cloud-based 3D CAD and Circuit design tool

Nightlight example

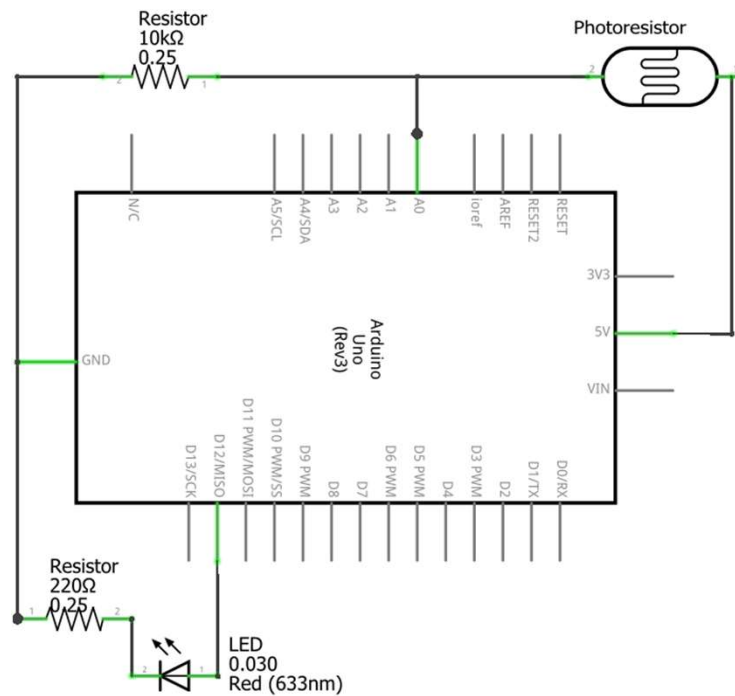
Night Light – a simple circuit to switch on an LED when it gets dark

- Open | Maker/Documents/ArduinoClass/nightlight/nightlight.ino
- Demonstrates use of analog input and digital output
- Demo: <https://www.youtube.com/watch?v=2GqKbUyhUww>

Nightlight example - Breadboard



Nightlight example - Schematic



fritzing