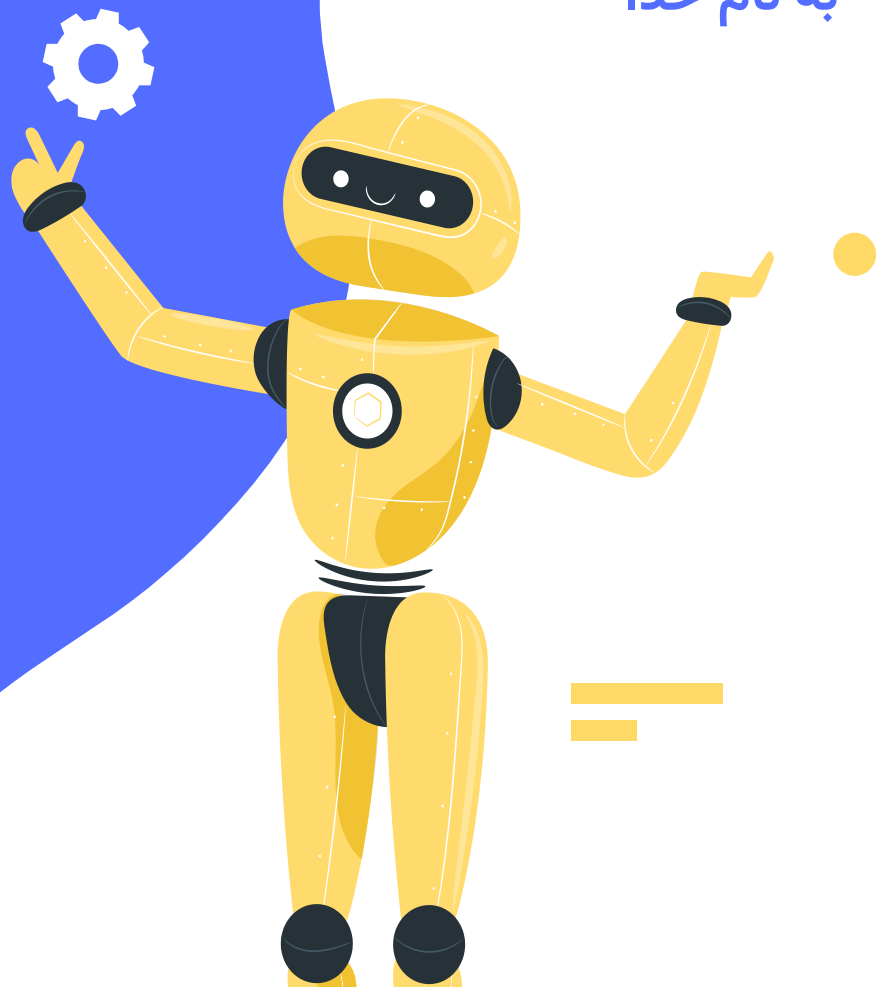


به نام خدا

Robotics Course

With MohammadReza Gholami

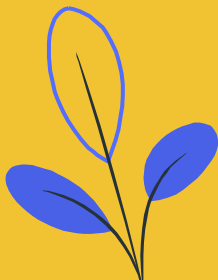




2nd Session

جلسه دوم

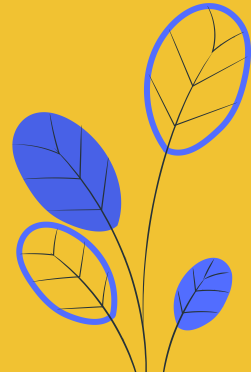
کنترل در رباتیک، شروع برنامه نویسی در آردوینو، یاد آوری زبان `cpp`،
کار با `GPIO` ها در آردوینو، کنترل `LED`، پروژه `blink`



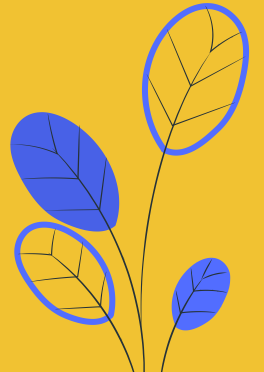
Three yellow gears of different sizes are positioned on the left side of the slide. One large gear is partially visible on the far left, and two smaller gears are positioned above and below it.

02

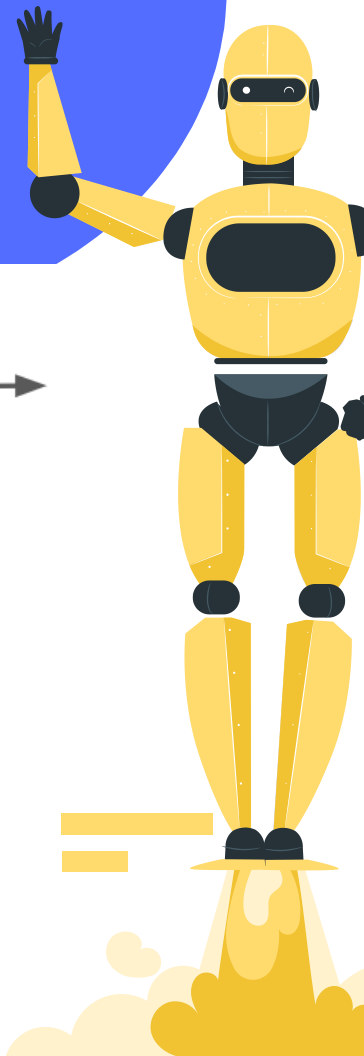
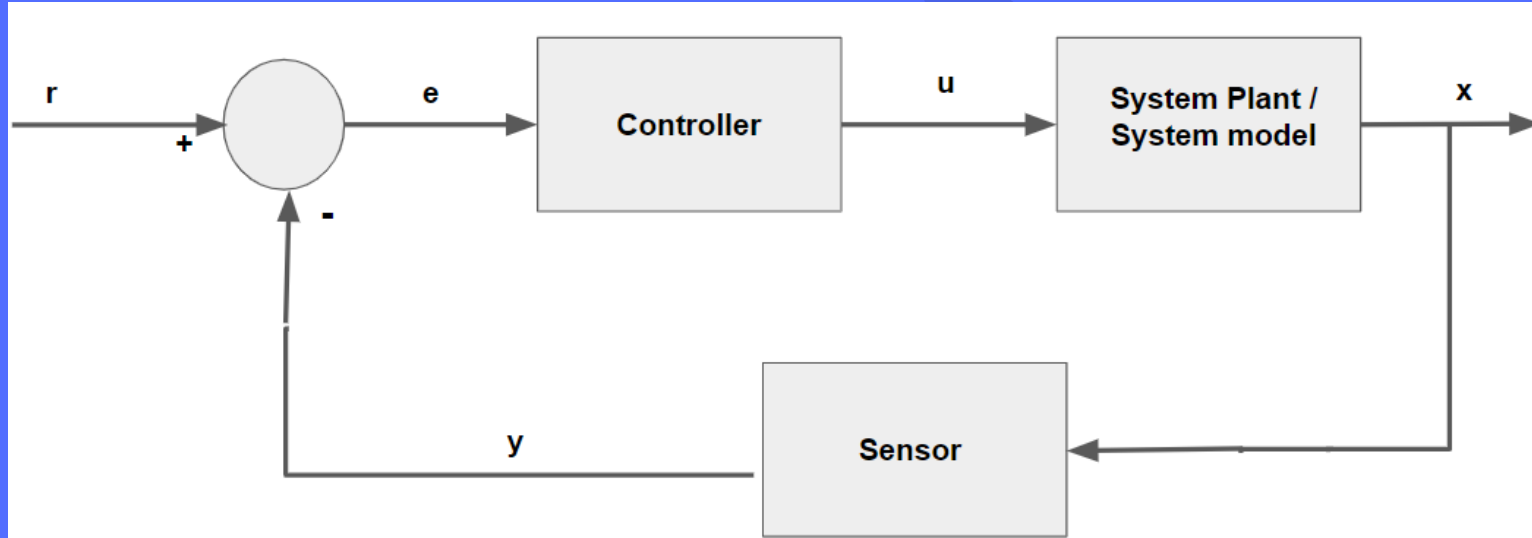
Control

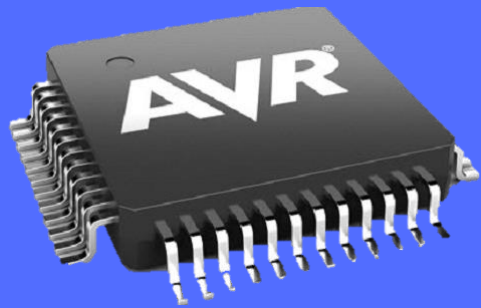


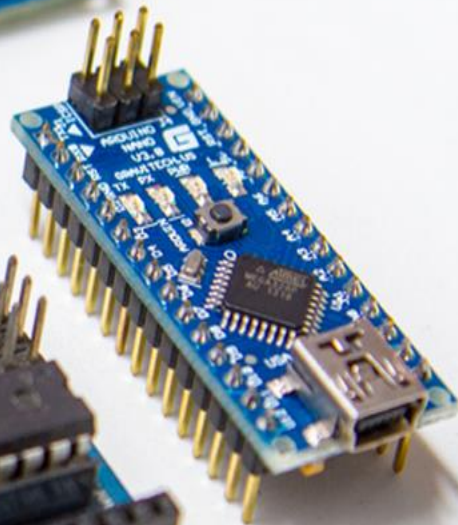
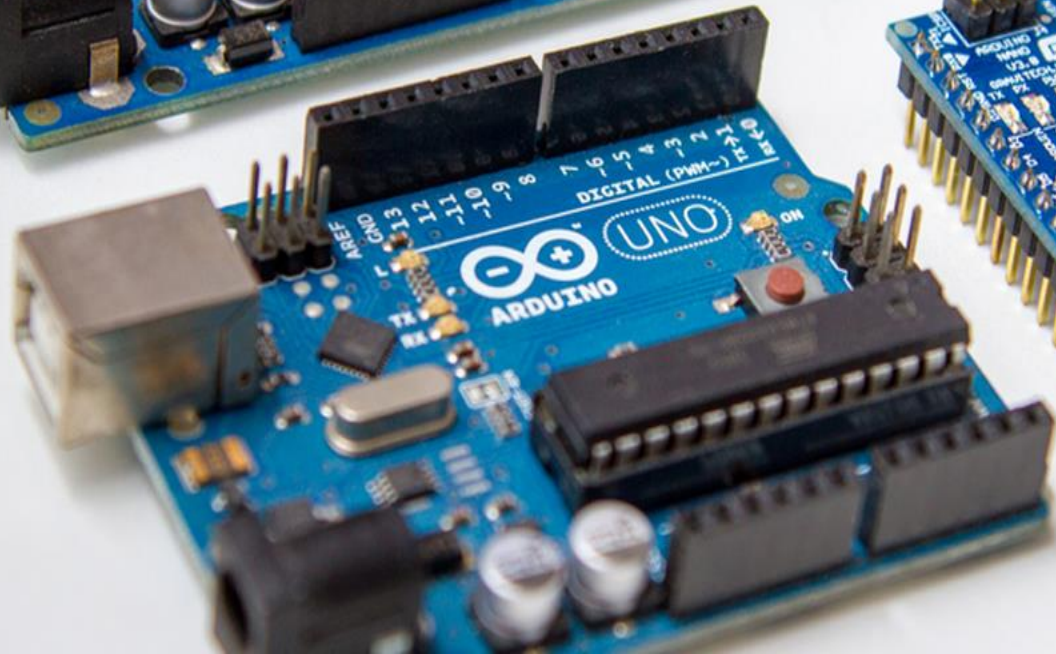
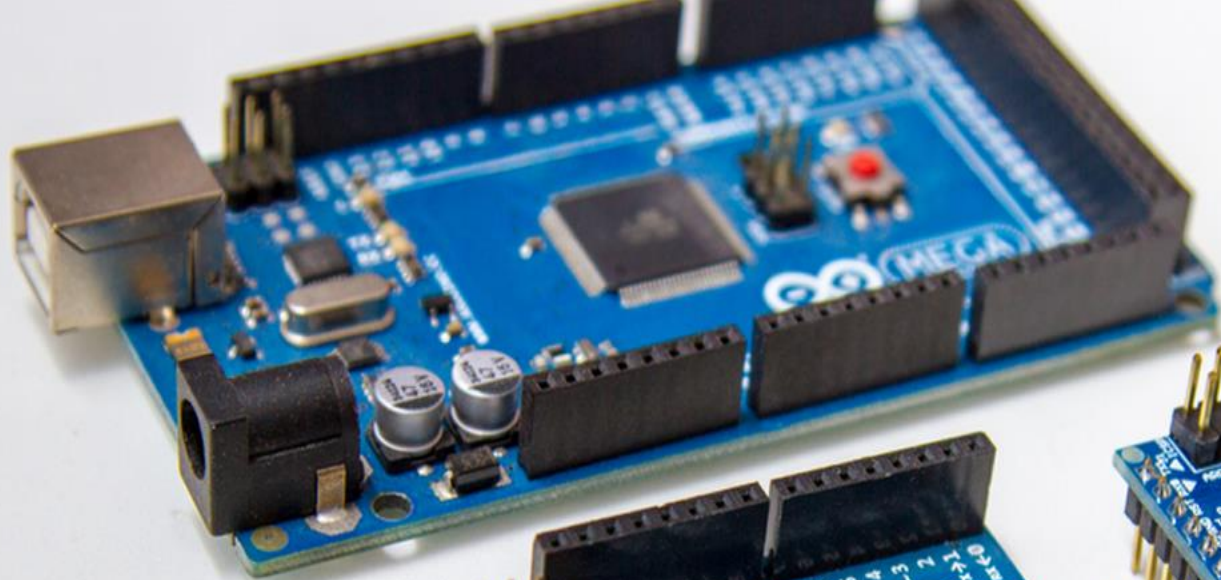
شروع کار با آردوینو



Robotic Control Systems

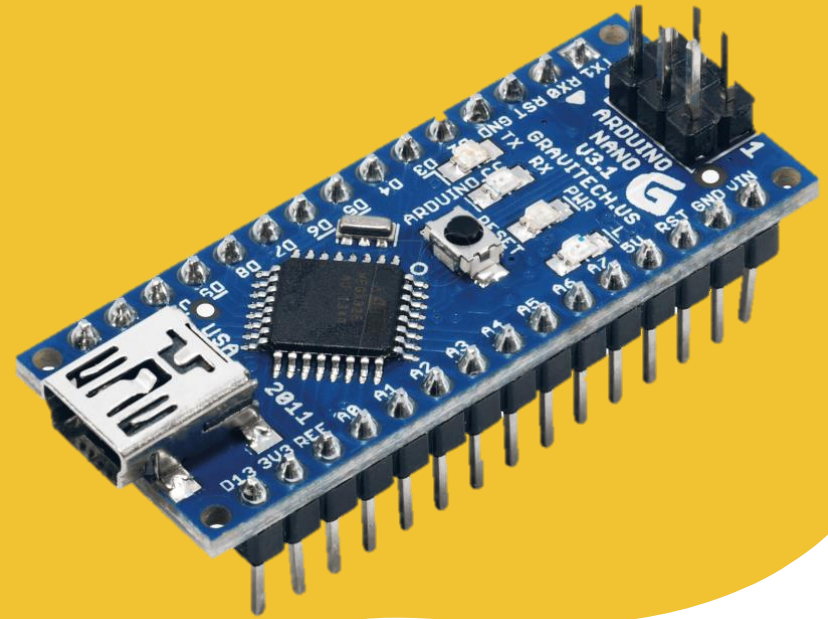
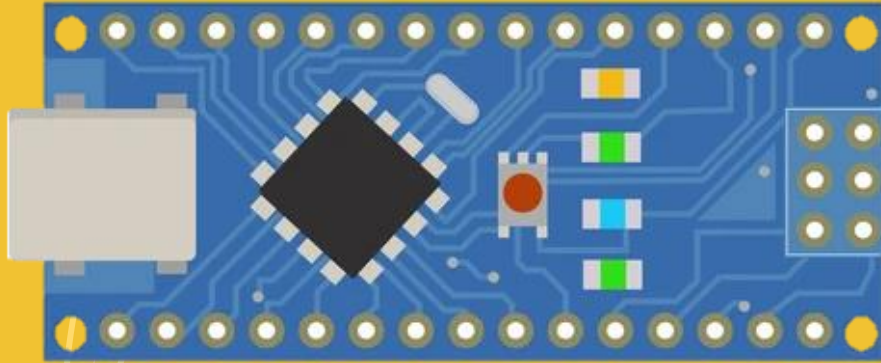








Arduino (Nano)



Introduction

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.x). It lacks only a DC power jack, and works with a Mini-B USB cable instead of a standard one. In the design of this series of Arduino Nano, IC CH340 is used instead of FT232, and in order to communicate with the board, the CH340 driver must be installed on the system.

Specification

Microcontroller: ATmega168 or ATmega328

Using IC CH340 instead of FT232

Operating Voltage (logic level): 5 V

Input Voltage (recommended): 5-9 V

Digital I/O Pins: 14 (of which 6 provide PWM output)

Analog Input Pins: 8

DC Current per I/O Pin: 40 mA

Flash Memory: 16 KB (ATmega168) or 32 KB (ATmega328) of which 2 KB used by bootloader

SRAM: 1 KB (ATmega168) or 2 KB (ATmega328)

EEPROM: 512 bytes (ATmega168) or 1 KB (ATmega328)

Clock Speed: 16 MHz

Dimensions: 45 x 18 mm

Weigth: 5 g

آردوینو نانو یک برد قابل برنامه ریزی اوپن سورس کوچک و کامل است که بر اساس میکرو کنترلر ATmega328 سری AVR ساخته شده است. در طراحی این سری از آردوینو نانو از آی سی CH340 به جای FT232 استفاده شده که جهت ارتباط با برد باید درایور CH340 بر روی سیستم نصب شود. این نوع از برد های آردوینو فاقد جک آداپتور می باشد و با یک کابل USB-Mini کار می کند. همچنین طراحی آردوینو نانو به صورتی است که دارای پین هدر بوده که امکان اتصال آسان به بردبرد را فراهم می کند.

مشخصات

- میکروکنترلر: ATmega168 یا ATmega328
- استفاده از آی سی CH340 به جای FT232
- ولتاژ کاری (سطح منطقی): 5 ولت DC
- ولتاژ ورودی (توصیه شده): 9-12 ولت DC
- پین ورودی/خروجی دیجیتال: 14 عدد
- پین PWM ورودی /خروجی : 6 پین
- پین ورودی آنالوگ: 8 پین
- جریان DC در هر پین ورودی/خروجی: 40 میلی آمپر
- حافظه فلش: 16 کیلوبایت (ATmega168) یا 32 کیلوبایت (ATmega328) که 2 کیلوبایت آن توسط بوت لودر استفاده می شود
- SRAM: در (ATmega168) مقدار 1 کیلوبایت و در (ATmega328) 2 کیلوبایت
- EEPROM: در (ATmega168) مقدار 512 بایت و در (ATmega328) 1 کیلوبایت
- سرعت ساعت: 16 مگاهرتز
- ابعاد: 45x18 میلی متر
- وزن: 5 گرم

Arduino IDE v2.0

loop()

setup()

if()

else()

Code structure: setup , loop

The screenshot shows the Arduino IDE interface with a sketch named `sketch_nov24a.ino`. The code is as follows:

```
1 void setup() {  
2   // put your setup code here, to run once:  
3  
4 }  
5  
6 void loop() {  
7   // put your main code here, to run repeatedly:  
8  
9 }  
10
```

setup()

The `setup()` function is called when a sketch starts. Use it to initialize variables, pin modes, start using libraries, etc. The `setup()` function will only run once, after each powerup or reset of the Arduino board.

loop()

After creating a `setup()` function, which initializes and sets the initial values, the `loop()` function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond. Use it to actively control the Arduino board.

pinMode()

[Digital I/O]

Description

Configures the specified pin to behave either as an input or an output. See the Digital Pins page for details on the functionality of the pins.

As of Arduino 1.0.1, it is possible to enable the internal pullup resistors with the mode `INPUT_PULLUP`. Additionally, the `INPUT` mode explicitly disables the internal pullups.

Syntax

```
pinMode(pin, mode)
```

Parameters

pin: the Arduino pin number to set the mode of.

mode: `INPUT`, `OUTPUT`, or `INPUT_PULLUP`. See the Digital Pins page for a more complete description of the functionality.

Returns

Nothing



digitalWrite()

[Digital I/O]

Description

Write a HIGH or a LOW value to a digital pin.

If the pin has been configured as an OUTPUT with pinMode(), its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW. If the pin is configured as an INPUT, digitalWrite() will enable (HIGH) or disable (LOW) the internal pullup on the input pin. It is recommended to set the pinMode() to INPUT_PULLUP to enable the internal pull-up resistor. See the Digital Pins tutorial for more information.

If you do not set the pinMode() to OUTPUT, and connect an LED to a pin, when calling digitalWrite(HIGH), the LED may appear dim. Without explicitly setting pinMode(), digitalWrite() will have enabled the internal pull-up resistor, which acts like a large current-limiting resistor.

Syntax

```
digitalWrite(pin, value)
```

Parameters

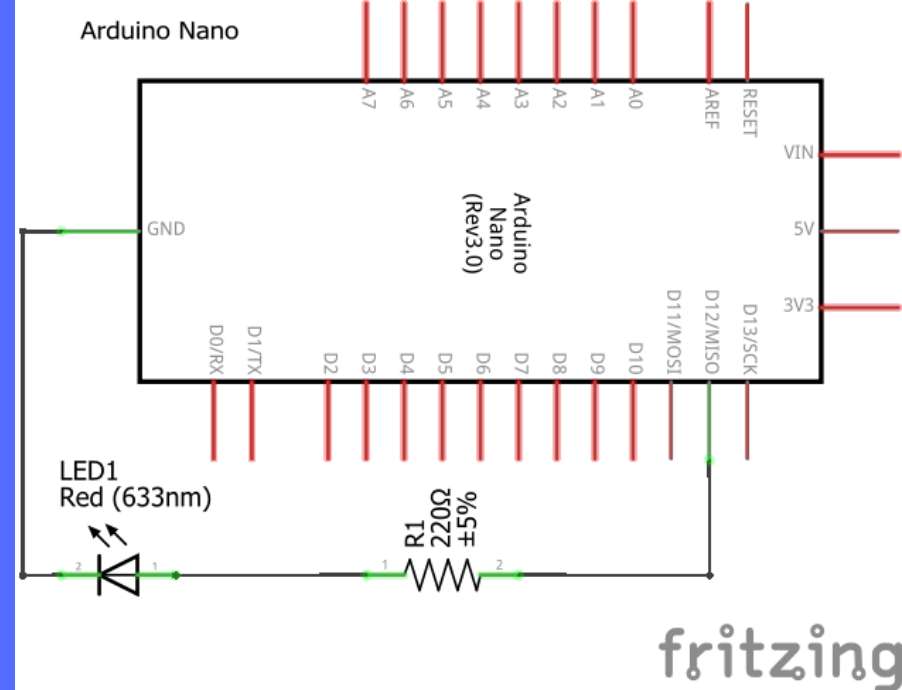
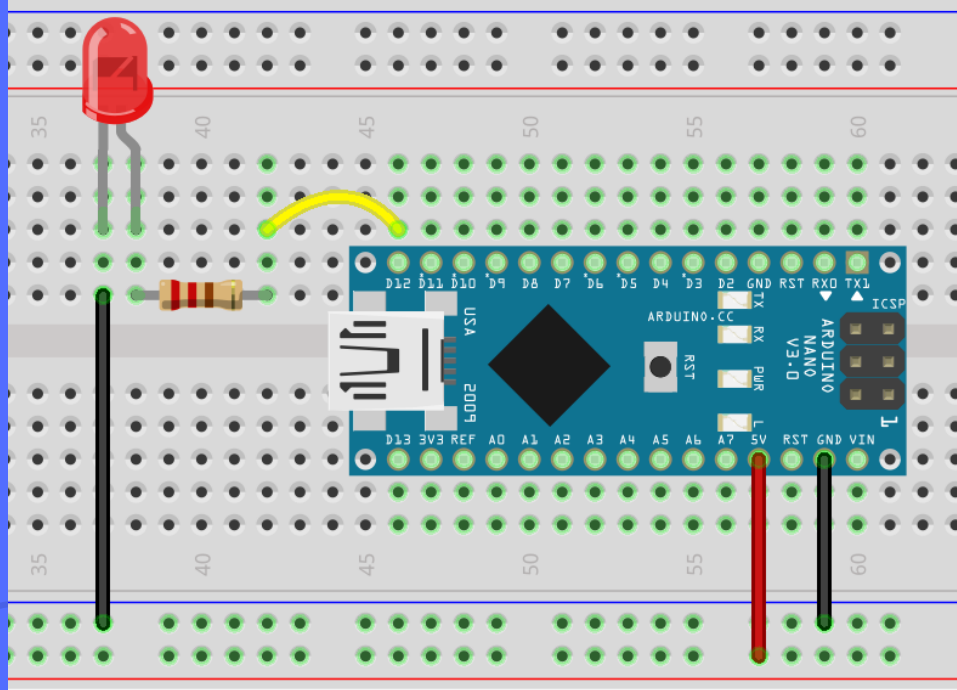
pin: the Arduino pin number.

value: HIGH or LOW.

Returns

Nothing



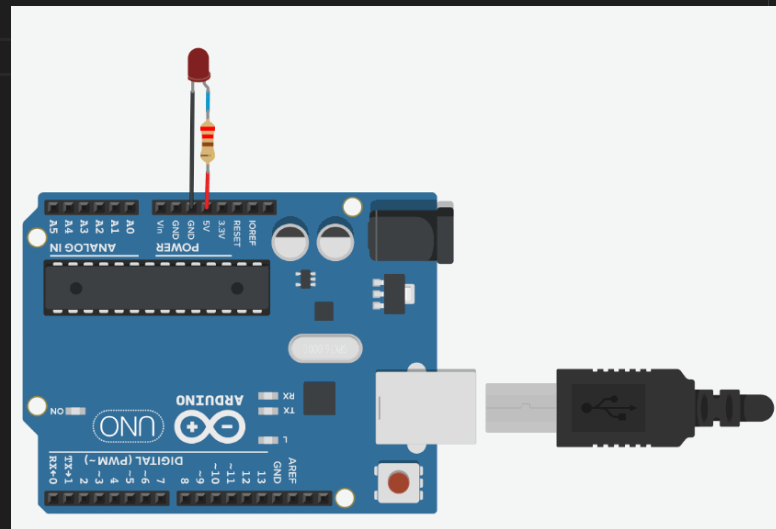




Arduino Uno

sketch_dec1a.ino

```
1 // the setup function runs once when you press reset or power the board
2 void setup() {
3   // initialize digital pin LED_BUILTIN as an output.
4   pinMode(LED_BUILTIN, OUTPUT);
5 }
6
7 // the loop function runs over and over again forever
8 void loop() {
9   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
10 }
11
```



digitalRead()

[Digital I/O]

Description

Reads the value from a specified digital pin, either HIGH or LOW.

Syntax

```
digitalRead(pin)
```

Parameters

pin: the Arduino pin number you want to read

Returns

HIGH or LOW

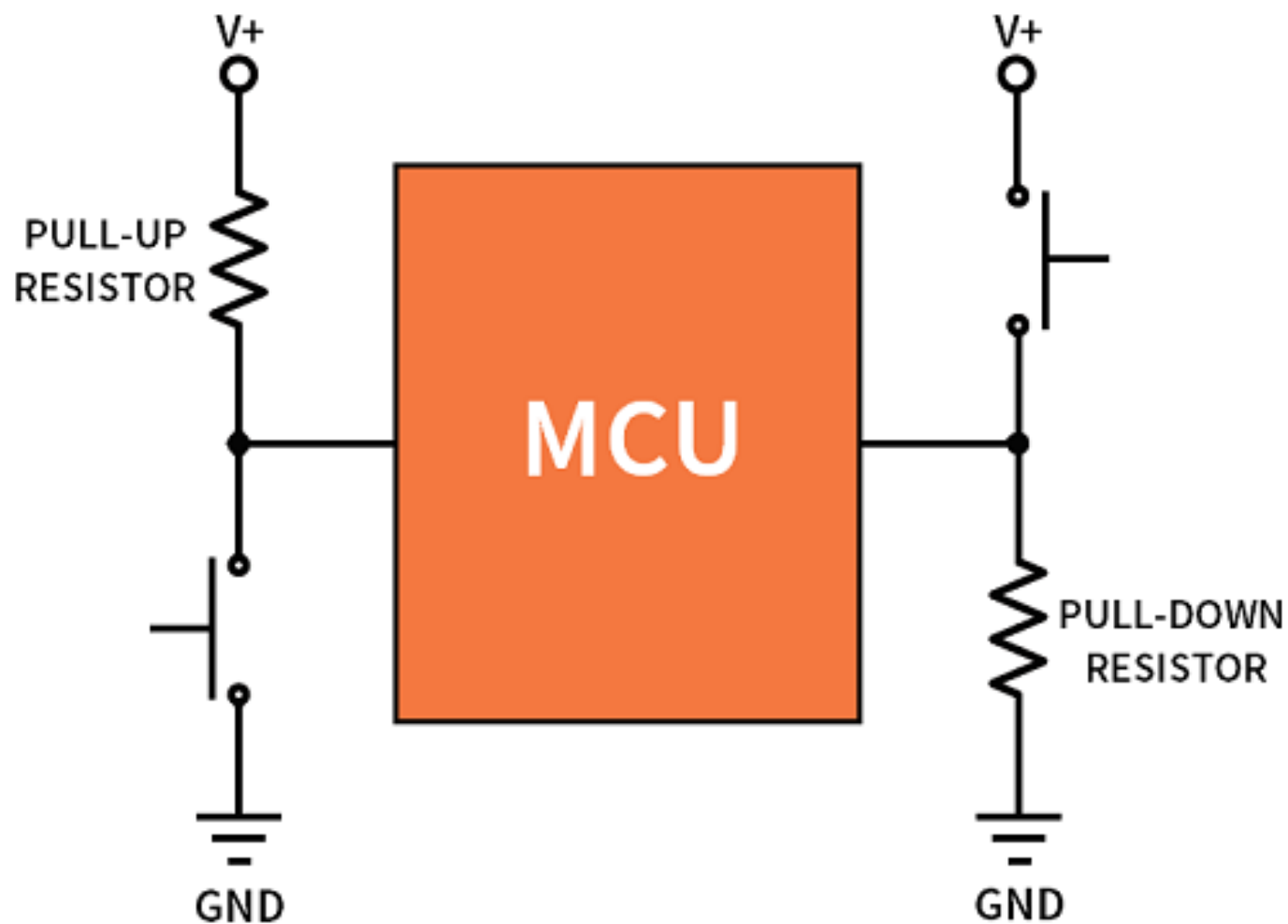


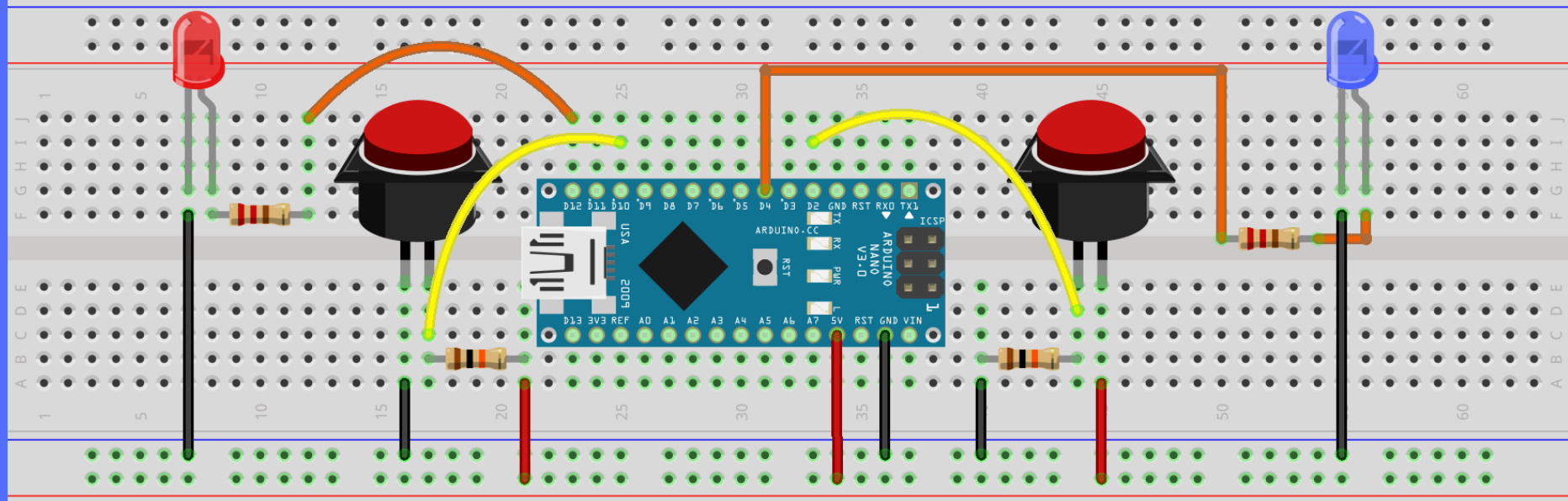
Notes and Warnings

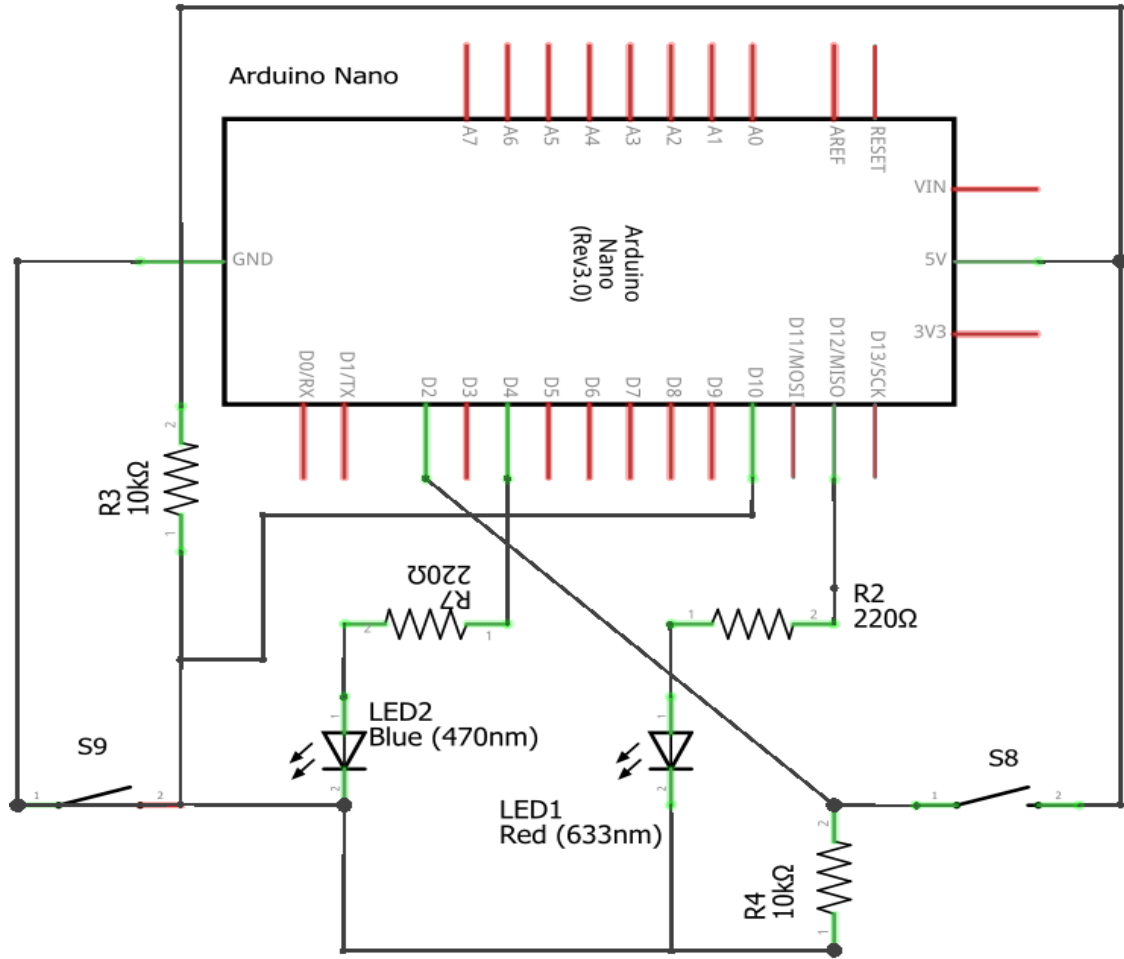
If the pin isn't connected to anything, `digitalRead()` can return either HIGH or LOW (and this can change randomly).

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.











References

- [Arduino Language Reference](#)
- [Digital Pins](#)
- [pinMode\(\)](#)
- [digitalWrite\(\)](#)
- [digitalRead\(\)](#)



THANKS

Do you have any questions?

E-Mail me

09939996370



<https://github.com/mmd00Z>

