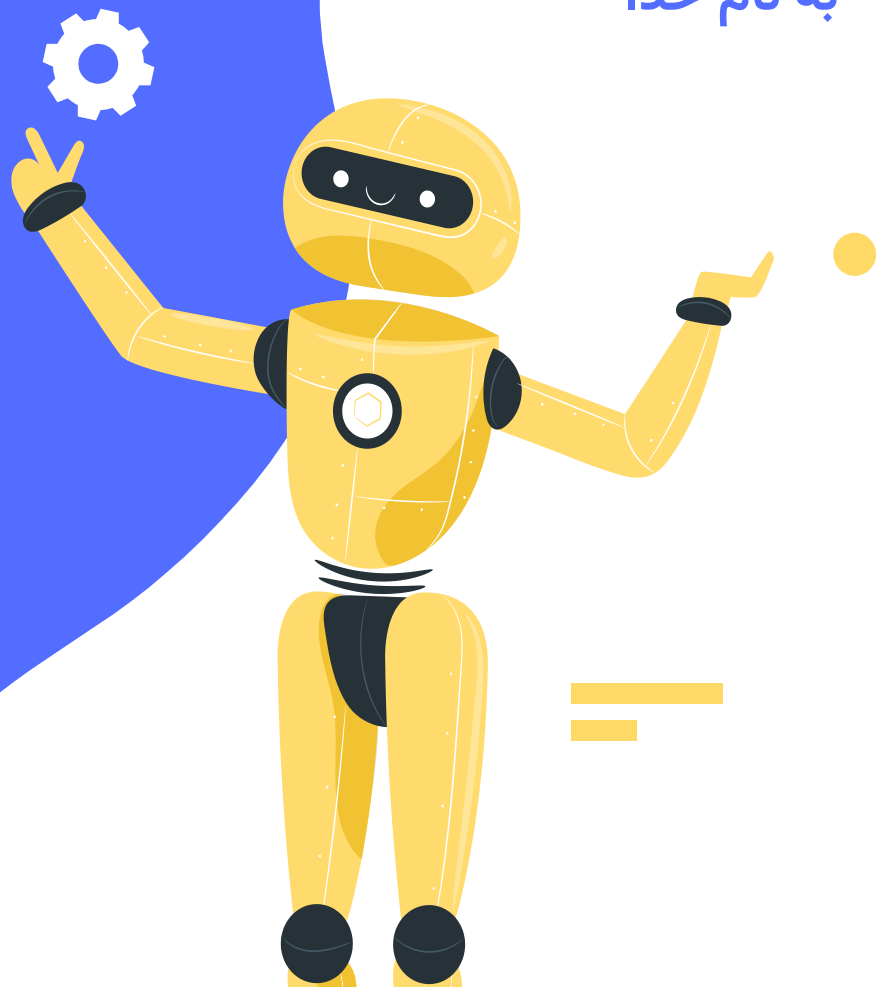


به نام خدا

Robotics Course

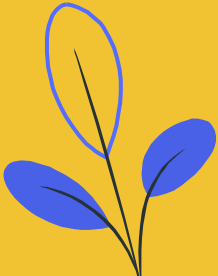
With MohammadReza Gholami





6th Session

جلسه ششم
راه اندازی موتور DC، درایور موتور L298، واحد PWM، ورودی و خروجی آنالوگ



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.

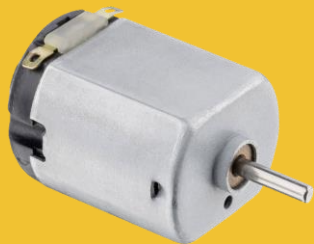
06

Motors

A stylized blue plant with four leaves and a thin stem is located in the bottom right corner of the slide.

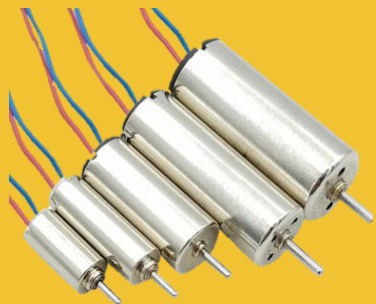
موتور ها و حرکت ربات





DC

موتور DC صرفاً یک موتور الکتریکی ساده با قابلیت کار با جریان مستقیم DC بوده و دارای سیم پیچ داخلی و شفت متحرک می‌باشد.



Coreless

موتور بدون هسته یک موتور DC بدون هسته متحرک می‌باشد که دارای وزن کم، اندازه کوچک و شتاب بالایی می‌باشد. این نوع موتورها دارای دور بسیار بالایی است که در صنعت رباتیک، پروژه‌های الکترونیک و به ویژه ساخت مینی کوادکوپترها مورد استفاده قرار می‌گیرد.

به این نوع موتورها، موتورهای ثابت بدون صدا نیز گفته می‌شود. موتورهای Coreless فاقد هسته ثابت و روتور آن یک سیم پیچ تو خالی می‌باشد که معمولاً به شافت موتور وصل می‌شود.



Gearbox

موتورگیربکس از ترکیب یک موتور الکتریکی و گیربکس ساخته می‌شود که در آن وظیفه گیربکس تغییر دور و گشتاور موتور برای رسیدن به دور و قدرت مورد نظر است. گیربکس قادر است قدرت چرخش را افزایش و سرعت آن را کاهش دهد. هرچه نرخ کاهش سرعت در گیربکس بیشتر شود، قدرت آن بالاتر خواهد رفت.



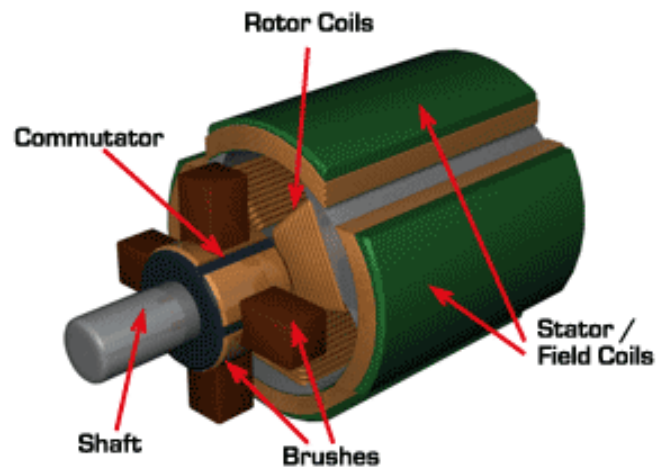
Servo

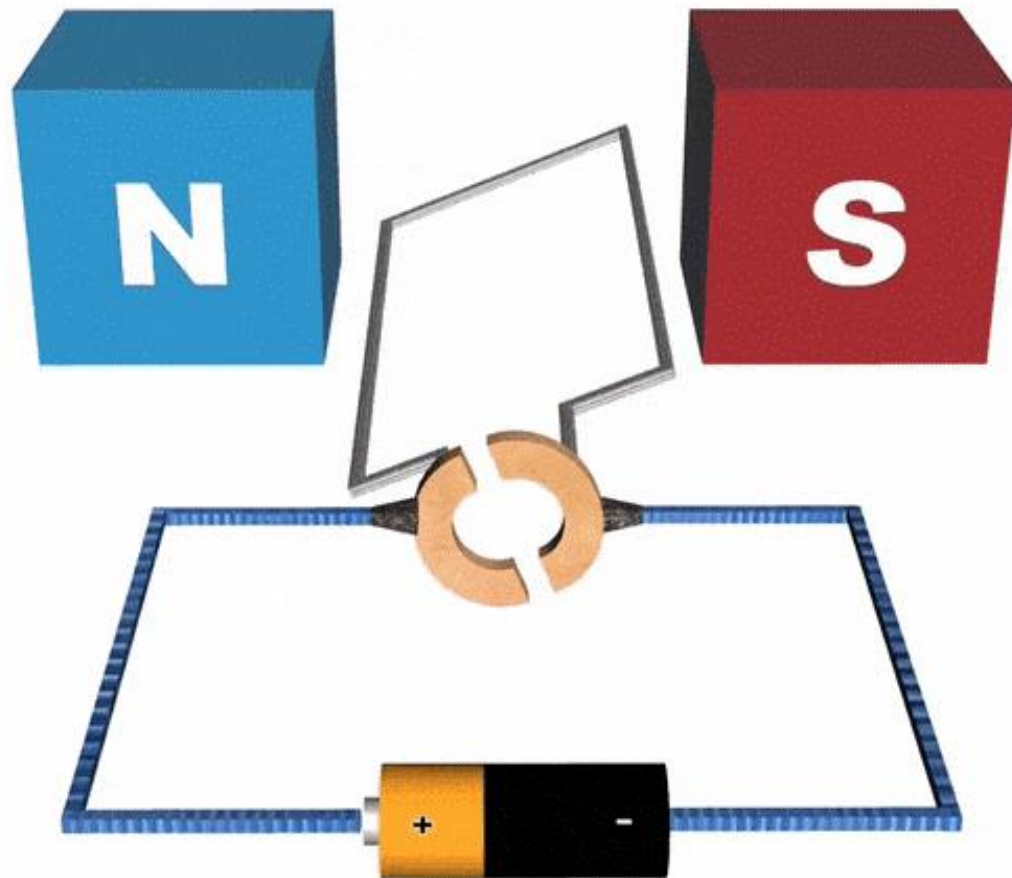
سرو موتور نوعی موتور DC است که شافت آن توانایی حرکت در زاویه مشخص را دارد. توسط سیگنال PWM و کنترل پهنای باند می‌توانید زاویه این موتورها را کنترل کنید. سروو موتورها مناسب برای ساخت انواع رباتهای پیشرفته و دقیق و بازوهای صنعتی می‌باشند.

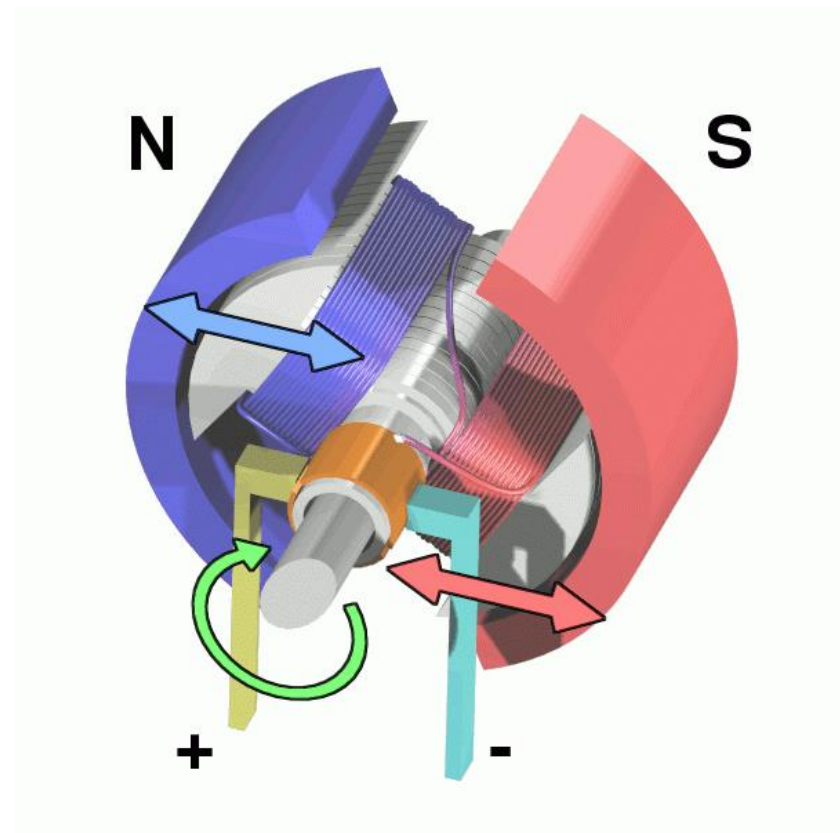
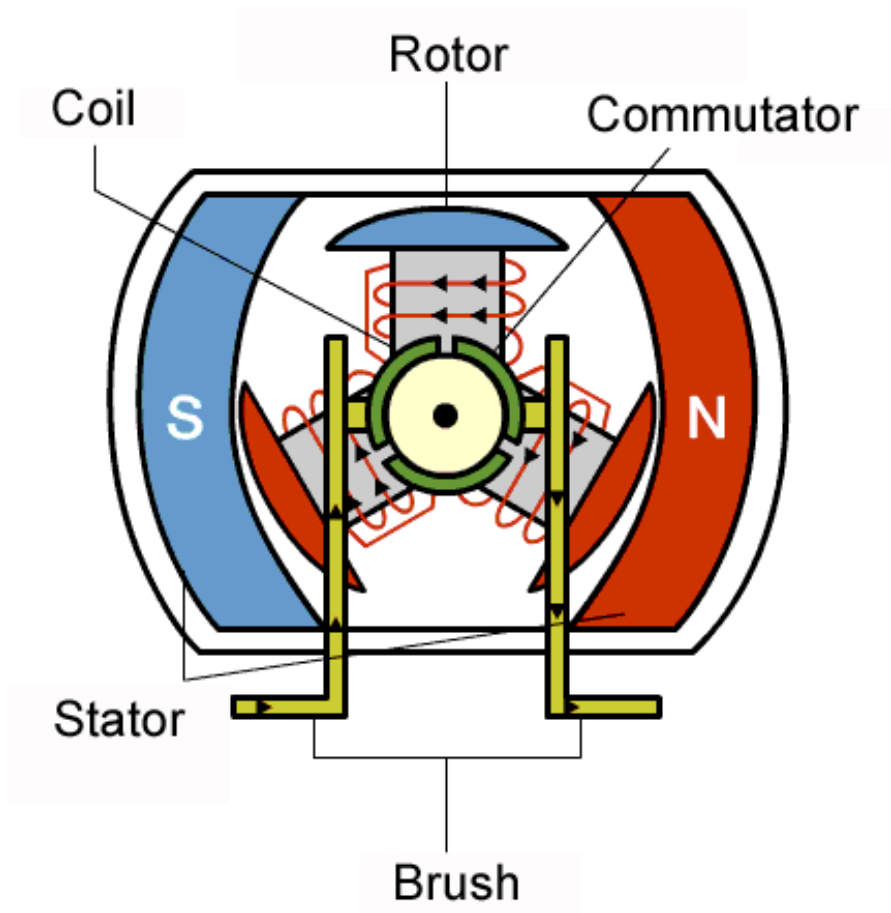


Stepper

موتور پله‌ای نوعی از موتورهای الکتریکی است که شیوهی عملکرد آن‌ها دقیق و همچنین مانند موتورهای AC به صورت پالس‌های الکتریکی فرمان می‌گیرند. با استفاده از استپر موتور می‌توانید در یک زاویه کاملاً دقیق با تلورانس فیکس شافت موتور را مدیریت کنید.

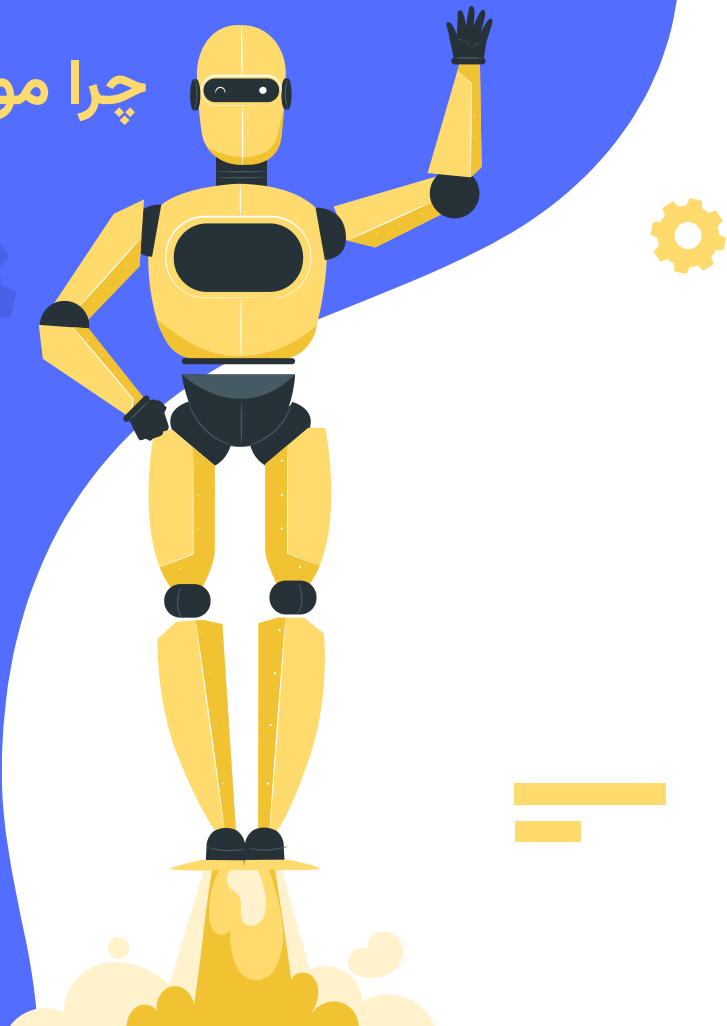






چرا موتور را مستقیم به آردوینو وصل نمی کنیم!؟

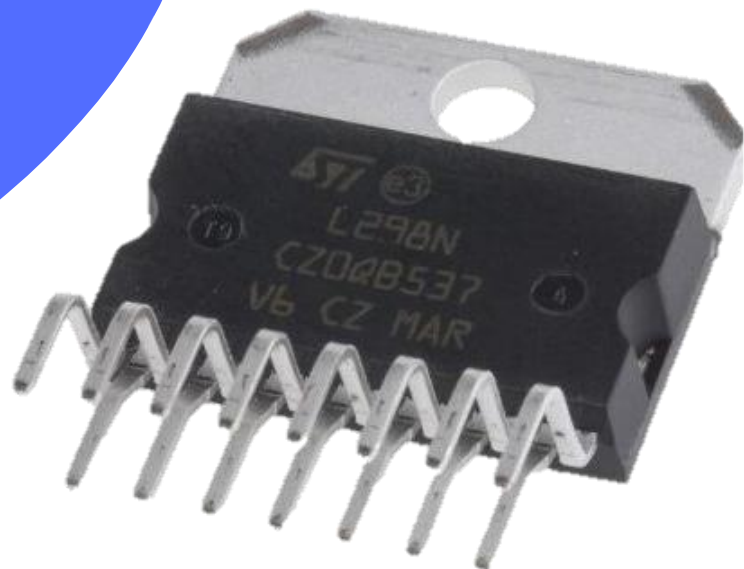
از هرکدام از GPIO های برد آردوینو می توان
تنها 50mA جریان کشید.
اما یک موتور DC بسته به نوع آن ممکن است
تا چند آمپر جریان مصرف کند.

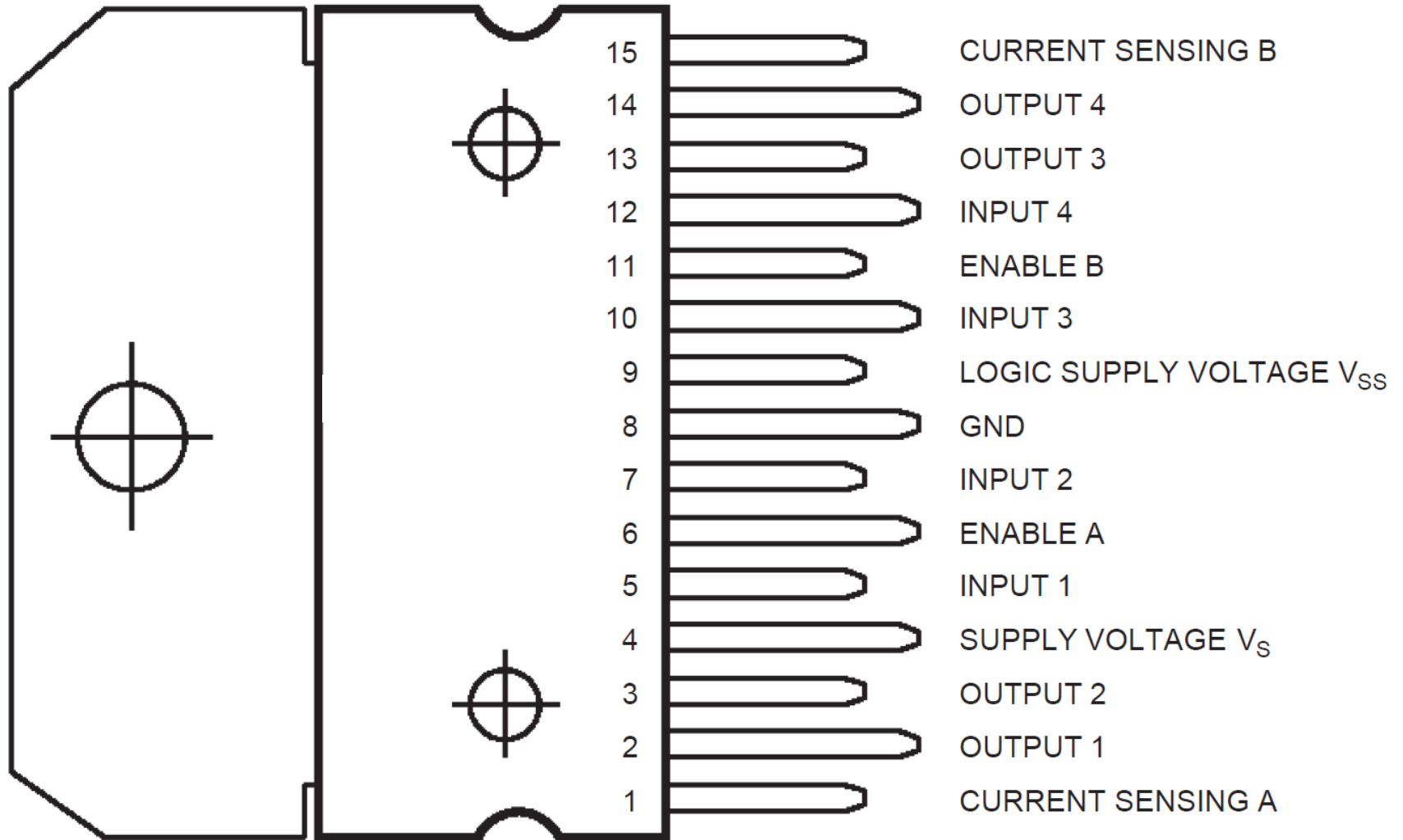


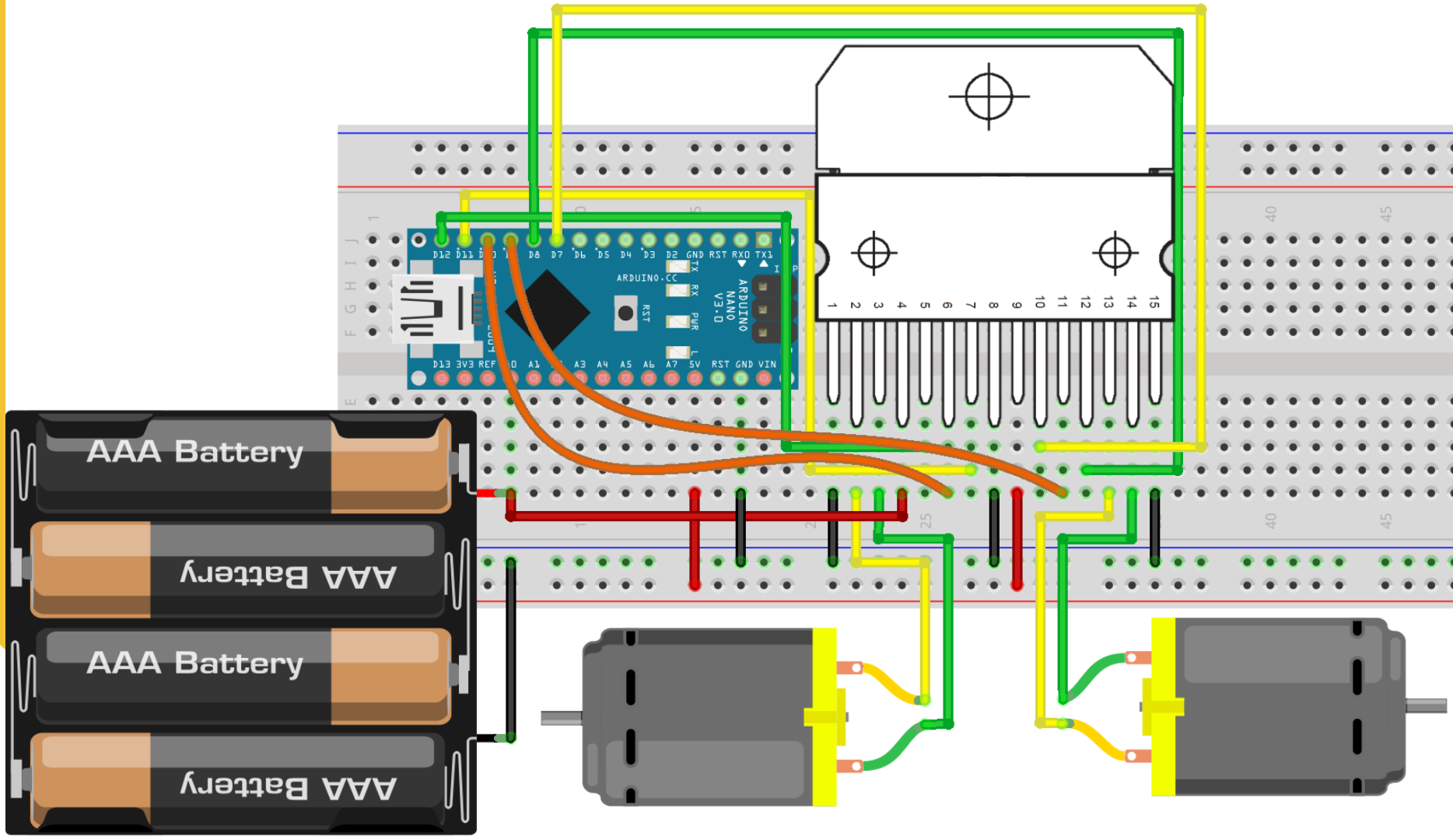
L298 driver motor IC

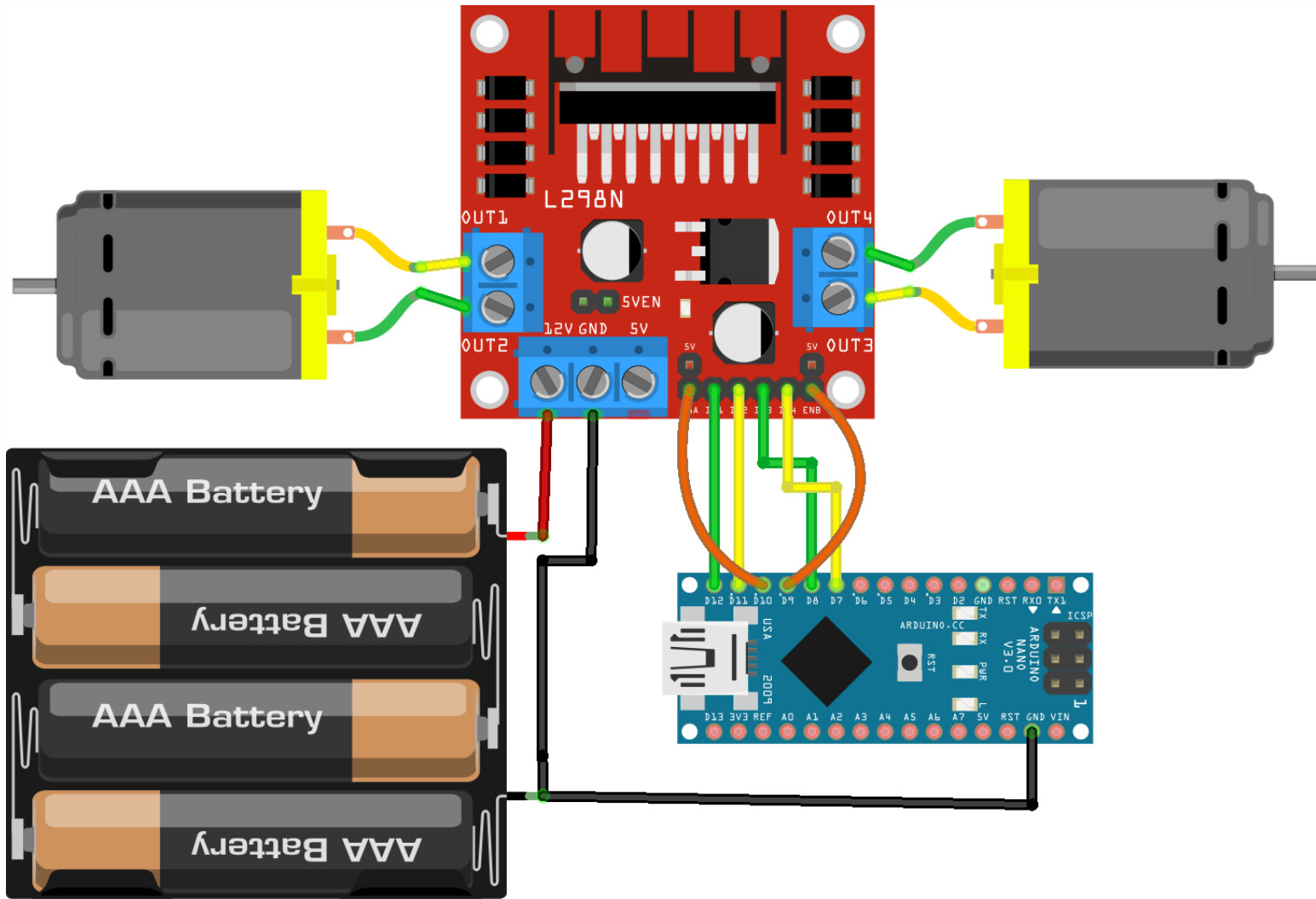
L298 یک درایور موتور است که شامل ۲ پل H با قابلیت کنترل جریان عبوری است. توسط این آی سی می توان سرعت چرخش، گشتاور و همچنین جهت چرخش موتور را کنترل کرد. ضمن اینکه این آی سی خود عمل تقویت جریان را نیز انجام می دهد.

برای راه اندازی موتور های DC با جریان مصرفی کمتر از 2 آمپر، استفاده از آی سی درایور L298N گزینه مناسبی می باشد.

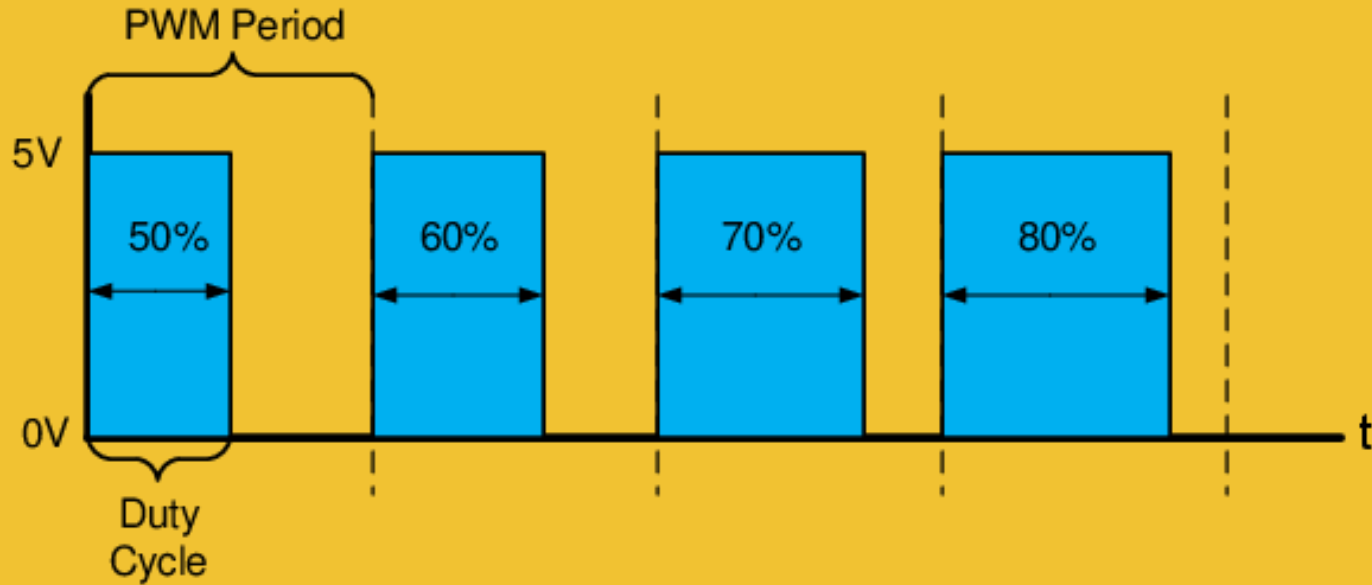




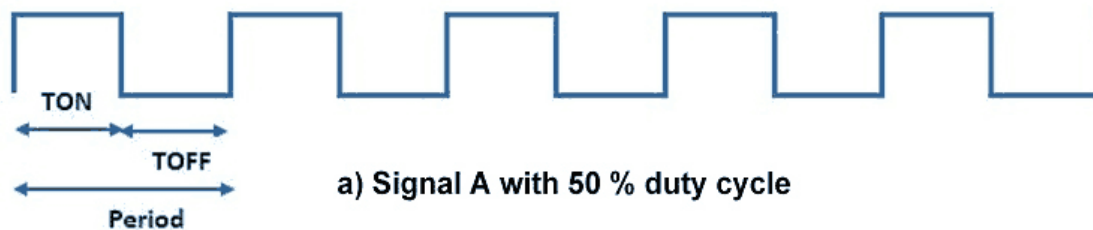




PWM(Pulse Width Modulation)



50%



a) Signal A with 50 % duty cycle

10%



b) Signal B with 10% duty cycle

30%

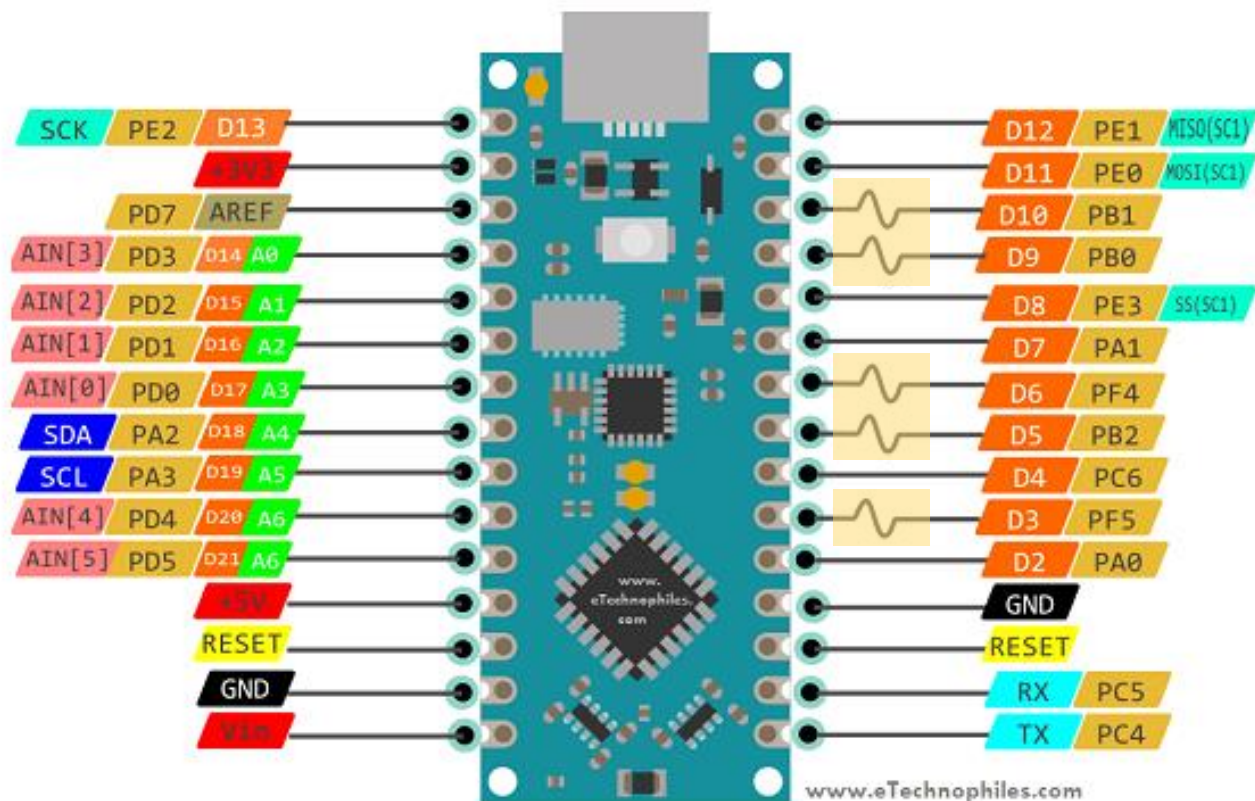


c) Signal C with 30% duty cycle

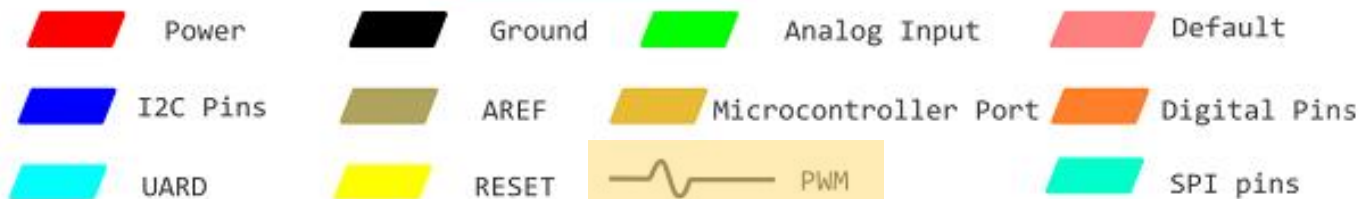
70%



d) Signal D with 70% duty cycle



Arduino Nano Every Pinout



analogWrite()

[Analog I/O]

Description

Writes an analog value (PWM wave) to a pin. Can be used to light a LED at varying brightnesses or drive a motor at various speeds. After a call to `analogWrite()`, the pin will generate a steady rectangular wave of the specified duty cycle until the next call to `analogWrite()` (or a call to `digitalRead()` or `digitalWrite()`) on the same pin.

Syntax

```
analogWrite(pin, value)
```

Parameters

pin: the Arduino pin to write to. Allowed data types: int.

value: the duty cycle: between 0 (always off) and 255 (always on). Allowed data types: int.

Returns

Nothing



Notes

You do not need to call `pinMode()` to set the pin as an output before calling `analogWrite()`.

The `analogWrite` function has nothing to do with the analog pins or the `analogRead` function.

Warnings

The PWM outputs generated on pins 5 and 6 will have higher-than-expected duty cycles. This is because of interactions with the `millis()` and `delay()` functions, which share the same internal timer used to generate those PWM outputs. This will be noticed mostly on low duty-cycle settings (e.g. 0 - 10) and may result in a value of 0 not fully turning off the output on pins 5 and 6.



BOARD	PWM PINS	PWM FREQUENCY
Uno, Nano, Mini	3, 5, 6, 9, 10, 11	490 Hz (pins 5 and 6: 980 Hz)
Mega	2 - 13, 44 - 46	490 Hz (pins 4 and 13: 980 Hz)
Leonardo, Micro, Yún	3, 5, 6, 9, 10, 11, 13	490 Hz (pins 3 and 11: 980 Hz)
Uno WiFi Rev2, Nano Every	3, 5, 6, 9, 10	976 Hz
MKR boards *	0 - 8, 10, A3, A4	732 Hz
MKR1000 WiFi *	0 - 8, 10, 11, A3, A4	732 Hz
Zero *	3 - 13, A0, A1	732 Hz
Nano 33 IoT *	2, 3, 5, 6, 9 - 12, A2, A3, A5	732 Hz
Nano 33 BLE/BLE Sense	1 - 13, A0 - A7	500 Hz
Due **	2-13	1000 Hz
101	3, 5, 6, 9	pins 3 and 9: 490 Hz, pins 5 and 6: 980 Hz

analogRead()

[Analog I/O]

Description

Reads the value from the specified analog pin. Arduino boards contain a multichannel, 10-bit analog to digital converter. This means that it will map input voltages between 0 and the operating voltage(5V or 3.3V) into integer values between 0 and 1023. On an Arduino UNO, for example, this yields a resolution between readings of: 5 volts / 1024 units or, 0.0049 volts (4.9 mV) per unit. See the table below for the usable pins, operating voltage and maximum resolution for some Arduino boards.

Syntax

```
analogRead(pin)
```

Parameters

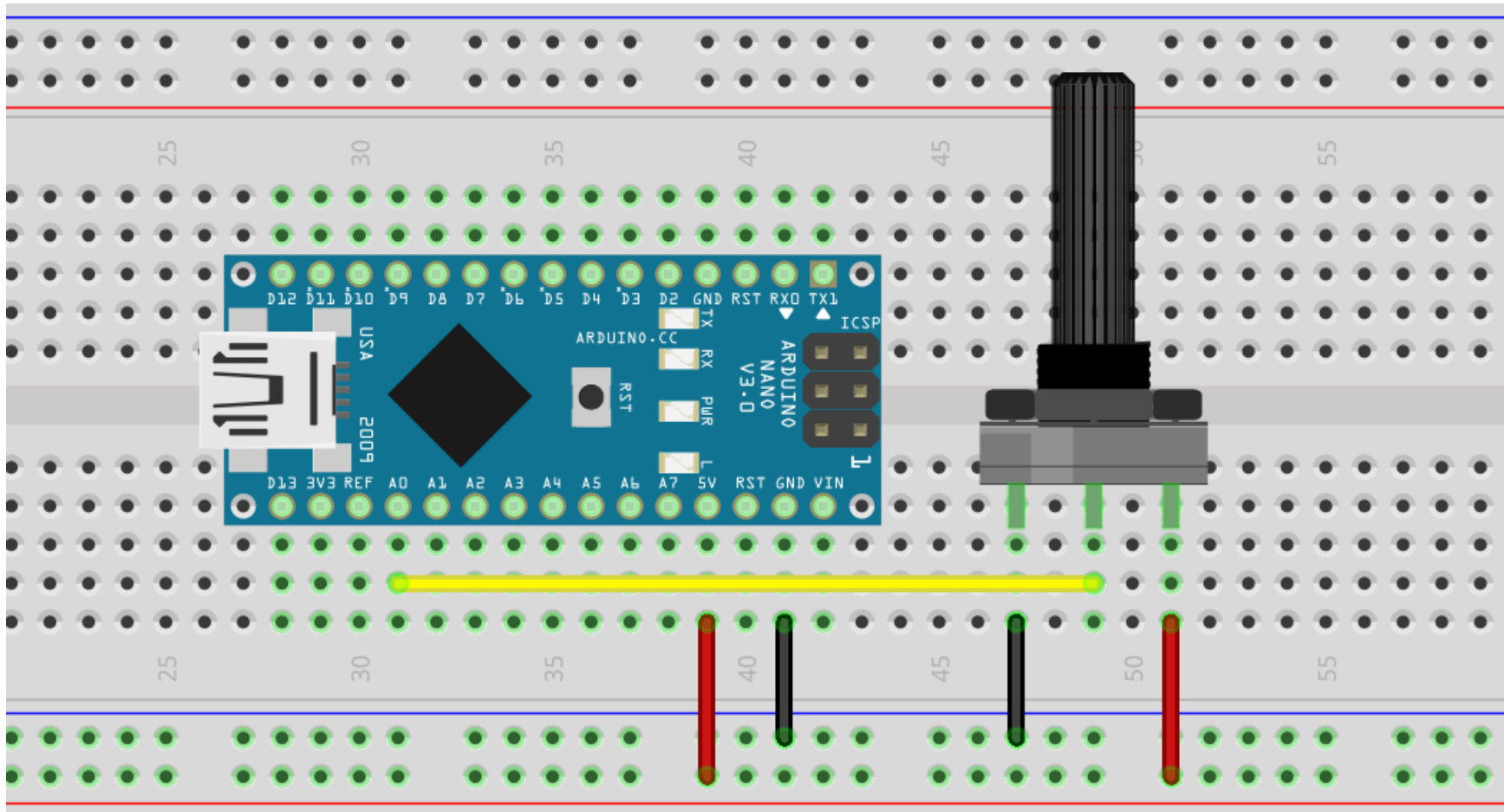
pin: the name of the analog input pin to read from (A0 to A5 on most boards, A0 to A6 on MKR boards, A0 to A7 on the Mini and Nano, A0 to A15 on the Mega).

Returns

The analog reading on the pin. Although it is limited to the resolution of the analog to digital converter (0-1023 for 10 bits or 0-4095 for 12 bits). Data type: int.



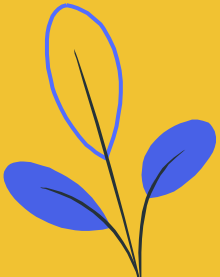
BOARD	OPERATING VOLTAGE	USABLE PINS	MAX RESOLUTION
Uno	5 Volts	A0 to A5	10 bits
Mini, Nano	5 Volts	A0 to A7	10 bits
Mega, Mega2560, MegaADK	5 Volts	A0 to A14	10 bits
Micro	5 Volts	A0 to A11*	10 bits
Leonardo	5 Volts	A0 to A11*	10 bits
Zero	3.3 Volts	A0 to A5	12 bits**
Due	3.3 Volts	A0 to A11	12 bits**
MKR Family boards	3.3 Volts	A0 to A6	12 bits**





References

- [Arduino Language Reference](#)
- [analogWrite\(\)](#)
- [analogRead\(\)](#)
- [SerialMonitor](#)



THANKS

Do you have any questions?

E-Mail me

09939996370



<https://github.com/mmd00Z>

