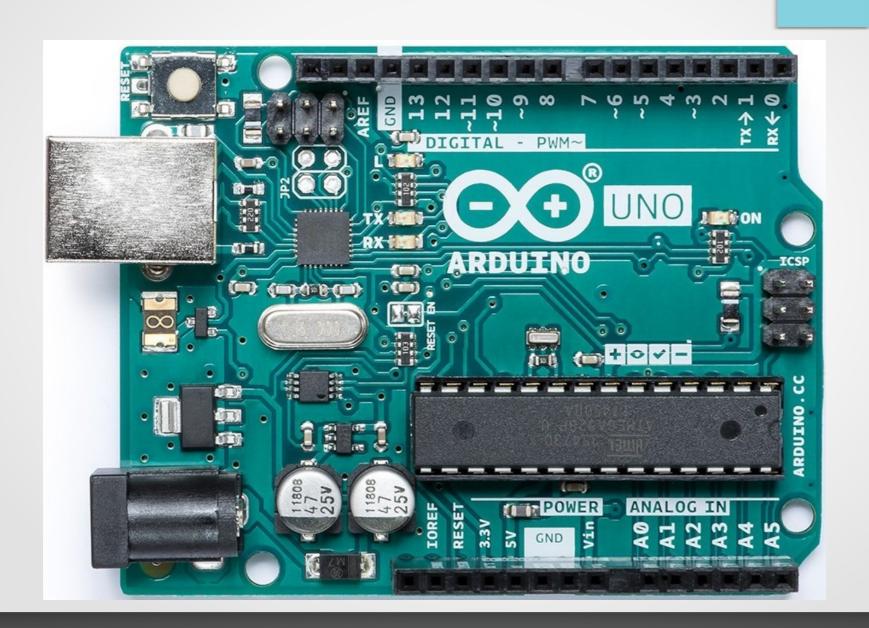
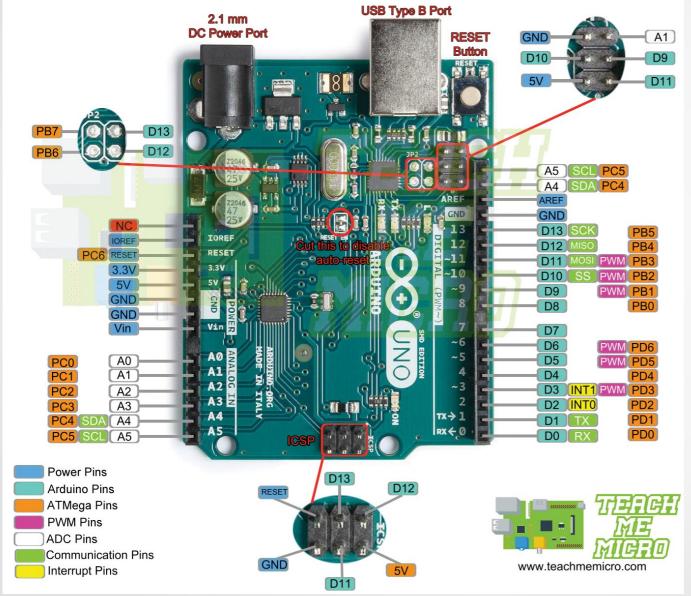
Servomotores



Servomotores – pinout Arduino



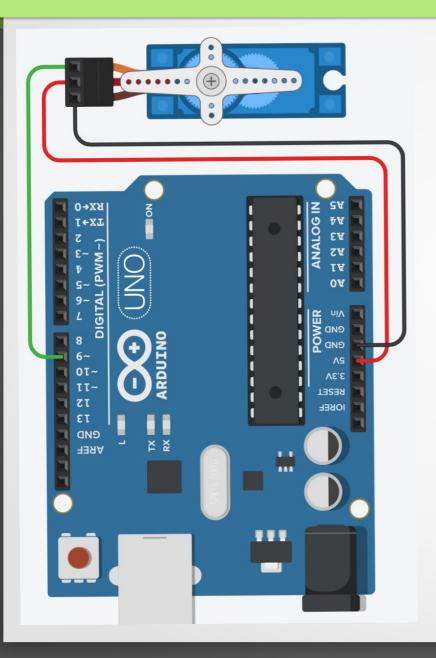
E/S Dixitais:

Pins 0 a 13
 0 a 5V, 20 mA
 Low: 0 a 2V
 High: 3 a 5V

Entradas Analóxicas

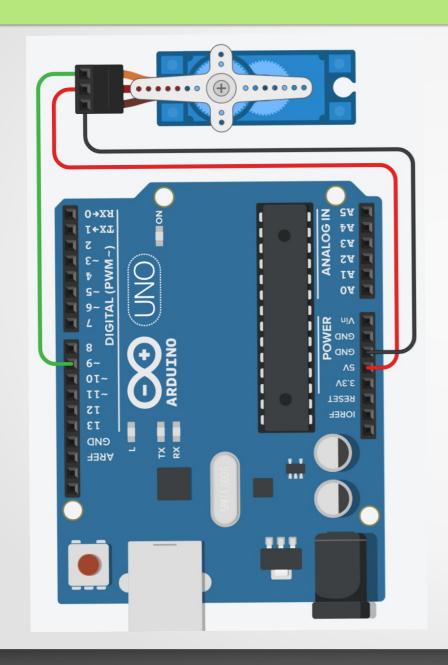
- Pins A0 A5
 0 a 5V, 20 mA prec 1024
 0 a 3V, 50 mA prec 1024
 Saídas PWM
- ~ Pins 3, 5, 6, 9, 10, 11
 0 a 5 V, 20 mA prec 256
 Comunic. Serie TX/RX
- Pins 0 e 1
- 6 pins para comunicarse directamente co proc. Atmega328
- 6 pins para programar o USB

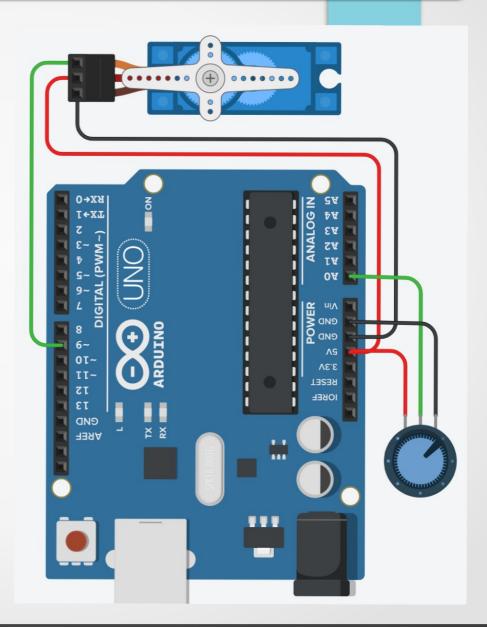
- Un servomotor ou servo é un dispositivo electromecánico que pode facer xirar o seu eixo un determinado número de graos, e manter a posición indicada.
- O sinal de control dun servo require dunha temporización complexa, sen embargo as librerías dispoñibles en Arduino facilitan moito o seu uso.
- Os servos teñen o aspecto exterior dunha caixiña cun perno que pode xirar un máximo de 180º ou 270º dependendo do modelo. Dentro do servo hai:
 - un motor DC,
 - un potenciómetro, usado para detectar a posición do motor,
 - un conxunto de engrenaxes que conecta o potenciómetro e o motor e aumentan a potencia mecánica do dispositivo,
 - un circuíto, que recibe o sinal de control, acciona o motor e detecta a posición ao ler o potenciometro.
- O cableado dun servo é simple: dous cables para alimentación (vermello e negro) e un adicional para o sinal de control.
- Dependendo da tensión de alimentación do servo, pode alimentarse ou non directamente co Arduino. Se non se alimenta directamente, ten a precaución de conectar o negativo da alimentación ao GND do Arduino.

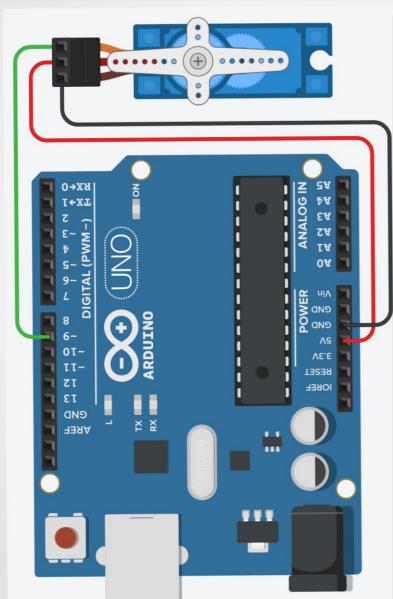


- Imos probar a librería
 Servo.h> mediante dous scripts:
 - avanzar o perno entre 0º e 180º, e a continuación retroceder,
 - outro script que ligue a rotacion dun potenciómetro coa do servo.
- A documentación da librería pódela encontrar en:

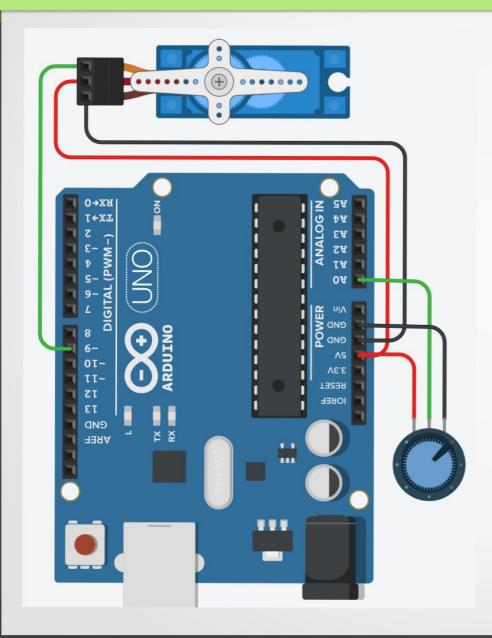
librería <Servo.h>





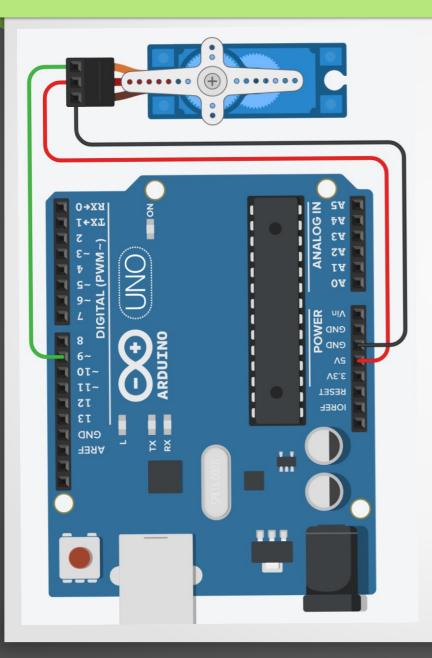


```
servo
 * Proba de funcionamento dun servomotor
* empregando a librería <Servo.h>
#include <Servo.h>
//Pin de control do servo
#define CTRL 9
//Declaramos o obxecto motor
//da clase Servo
Servo motor;
int veloc = 100;
void setup() {
 motor.attach(CTRL);
void loop() {
 //Move o servo entre 0° e 180° en
  //incrementos de 5°
  for(int i = 0; i \le 180; i++) {
   motor.write(i);
    delay(veloc);
  delay(15*veloc);
 //Move o servo entre 180° e 0° en
 //decrementos de 5°
  for(int i = 180; i >= 0; i--) {
   motor.write(i);
    delay(veloc);
}
```

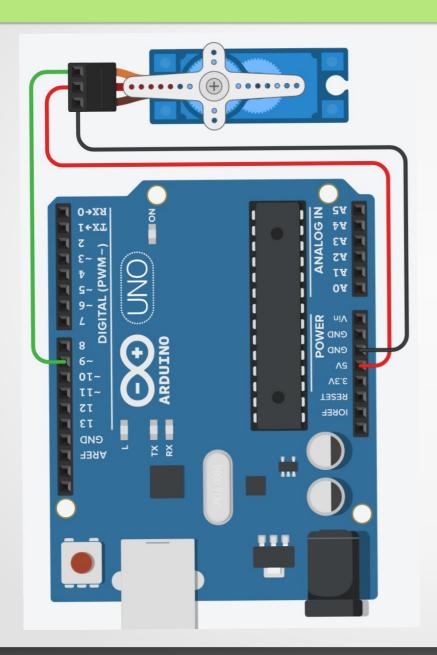


servo.con.potenciometro

```
* Proba de funcionamento dun servomotor
 * empregando a librería <Servo.h> e
 * ligado a un potenciómetro.
#include <Servo.h>
//Pin de control do servo
#define CTRL 9
//Pin de lectura do potenc
#define POT A0
//Declaramos o obxecto motor
//da clase Servo
Servo motor;
int angulo = 0;
int veloc = 100;
void setup() {
 motor.attach(CTRL);
void loop() {
 //Lectura do potenciometro
 angulo = analogRead(POT);
 angulo = map(angulo, 0, 1023, 0, 180);
 //Actualiza servo
 motor.write(angulo);
 delay(veloc);
```



- Imos reaproveitar a primeira montaxe para controlar o servo mediante a comunicación serie.
- Empregaremos as función proporcionadas polas clases 'Serial' e 'String', a fin de facilitar a programación.
- Para posicionar o servo empregaremos tres ordes de tipo texto ou 'String':
 - "centro",
 - "dereita" e
 - "esquerda".
- Ademais queremos posicionar o servo de xeito máis preciso introducindo o ángulo exacto como un número enteiro.



servo.con.posicionamento.serie * Proba de funcionamento dun servomotor * empregando a librería <Servo.h> * e a entrada serie para o posicionamento #include <Servo.h> //Pin de control do servo #define CTRL 9 //Declaramos o obxecto motor //da clase Servo Servo motor; int veloc = 20; String orde =""; int posicion = 0; void setup() { motor.attach(CTRL); Serial.begin(9600); void loop() { //Comprobamos se hai orde no teclado if(Serial.available()) { orde = Serial.readStringUntil('\n'); orde.toLowerCase(); if(orde.equals("esquerda")) posicion = 180; else if(orde.equals("dereita")) posicion = 0; else if(orde.equals("centro")) posicion = 90; else { int tmp = orde.toInt(); if(tmp >= 0 && tmp <= 180) posicion = tmp;else posicion = 0;

}

motor.write(posicion);

delay(veloc);

- Nesta unidade aprendemos a:
 - usar a librería <Servo.h>, para controlar un servomotor,
 - posicionar un servo en unidades angulares,
 - por software,
 - empregando un sensor externo (potenciómetro),
 - enviando ordes numéricas e de texto, mediante comunicación serie.
 - repasamos algunhas estructuras de programación, como iteracións (bucle for), condicionais (estructura ifelse), así como a elaboración de condicions.