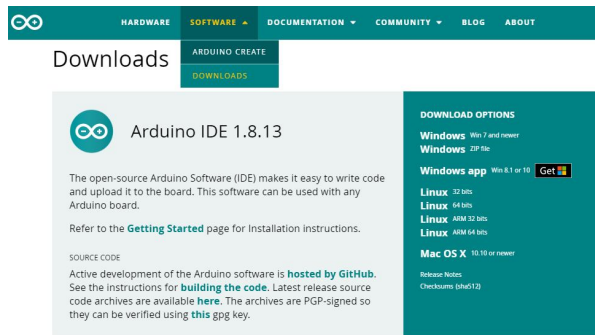


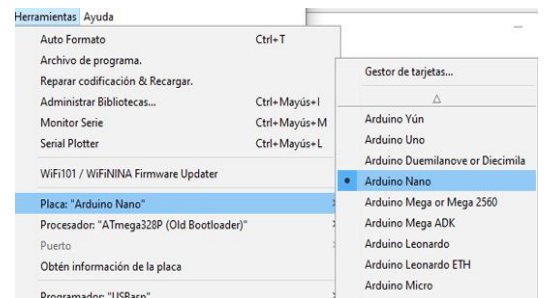
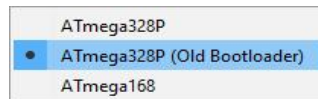
Anexo Programación

- Ha llegado el momento de programar el Escornabot (Arduino Nano).
 - Instala en tu ordenador el Arduino IDE (Software libre).
 - Nota: I.D.E. *Integrated Development Environment*
 - <https://www.arduino.cc/en/Main/Software>

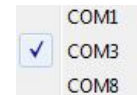


- La instalación dependerá de cada sistema operativo, es un proceso sencillo.

- Ejecuta el programa (Arduino)
- Conecta el Escornabot al ordenador.
 - En el menú "Herramientas", elije en "Placa" Arduino nano.
 - En "Procesador" elije

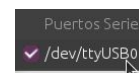


- Comprueba que el ordenador reconoce el puerto del Escornabot.
 - Cuando conectas el Escornabot al ordenador, aparece un puerto nuevo que antes no tenía (ojo el COM1 en Windows de sobremesa suele estar siempre, y no se corresponde con el puerto del USB).
 - Ejemplo, en Win antes de conectar el Escornabot tengo



- Nota: Para actualizar la lista de puertos en el menú "Herramientas" sal de ese menú y vuelve a entrar.

- EN GNU Linux aparece algo como: `/dev/ttyUSB0`, o `/dev/ttyACM0`



- En macOS tendrás algo así



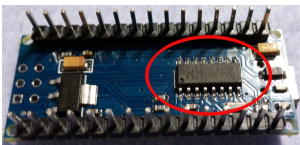
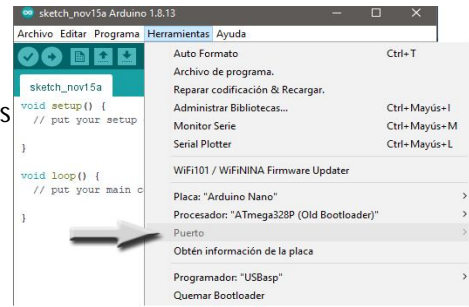
Anexo Programación

- Si no reconoce tendrás algo como esto: (el submenú "Puerto" aparece en gris)

- En ese caso instala el controlador (driver) para el chip de comunicaciones de ese Arduino Nano

- (en GNU Linux y macOs generalmente no es necesario).

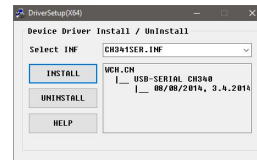
- En la parte trasera del Arduino Nano podemos ver el chip de comunicaciones, este dependerá del Arduino Nano tengas, lo más habitual es el chip CH340.



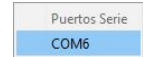
- Baja el paquete comprimido del [CH341SER](https://github.com/CH341SER), guardalo y descomprímelo en una carpeta.

- Ejecuta el instalador SETUP

- Pincha en "INSTALL" en unos segundos el led del Escornabot parpadeará y ya tienes instalado del controlador del puerto USB



- Regresa al Arduino IDE, ya reconoce el nuevo puerto, en este caso el COM6.



- Baja el firmware para el Escornabot creado por Rafa Couto. (software libre) <https://github.com/escornabot/arduino/releases/tag/v1.6.2>

- Descomprímelo, carga el fichero escornabot.ino en el Arduino IDE

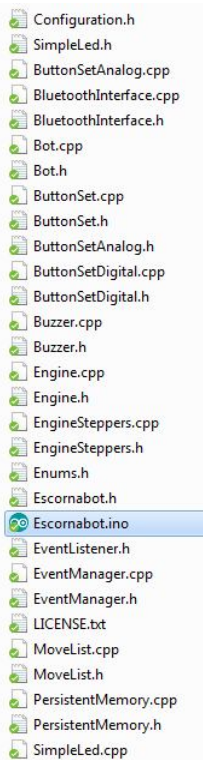
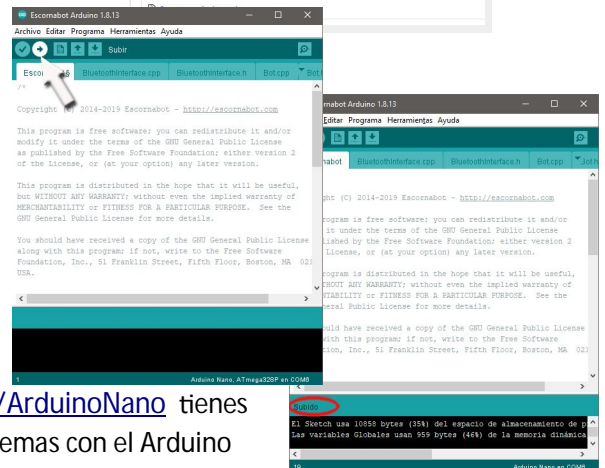
- Súbelo al robot.

- Si todo ha ido bien podemos ver el mensaje en la barra de notificaciones.

- Una vez programado el Escornabot pitará y encenderá uno de los LEDES del Arduino nano.

- En

<https://www.arduino.cc/en/Guide/ArduinoNano> tienes más información para resolver problemas con el Arduino Nano.



- Algunos ajustes que podemos hacer:
 - Ajustar la distancia recorrida por el robot, en la pestaña "Configuration.h" busca "Steppers engine setup" y en "// step calibration" puedes ajustar los valores de:

```
////////////////////////////////////  
///// Steppers engine setup  
////////////////////////////////////  
  
#ifndef ENGINE_TYPE_STEPPERS  
  
// stepper pin setup (digital outputs)  
#define STEPPERS_MOTOR_RIGHT_IN1 5  
#define STEPPERS_MOTOR_RIGHT_IN2 4  
#define STEPPERS_MOTOR_RIGHT_IN3 3  
#define STEPPERS_MOTOR_RIGHT_IN4 2  
#define STEPPERS_MOTOR_LEFT_IN1 9  
#define STEPPERS_MOTOR_LEFT_IN2 8  
#define STEPPERS_MOTOR_LEFT_IN3 7  
#define STEPPERS_MOTOR_LEFT_IN4 6  
  
// step calibration  
#define STEPPERS_STEPS_PER_SECOND 1000  
#define STEPPERS_LINE_STEPS 1738  
#define STEPPERS_TURN_STEPS 1024
```

- Velocidad "STEPPERS_STEPS_PER_SECOND", de 2200 en adelante suele dar problemas.
- Distancia de avance "STEPPERS_LINE_STEPS", esos 1738 pasos se corresponden con 10 cm, si quieres otra distancia, o has hecho unas ruedas más grandes, puedes modificarlo.
- Ángulo de giro "STEPPERS_TURN_STEPS" en este caso 1024 se corresponde con los giros 90°.
- En "PROGRAM_RESET_ALWAYS" puedes cambiar el valor "true" a "false" para que se guarden los movimientos, y seguir desde la última posición programada. Para comenzar de cero pulsa el botón de RESET del Arduino