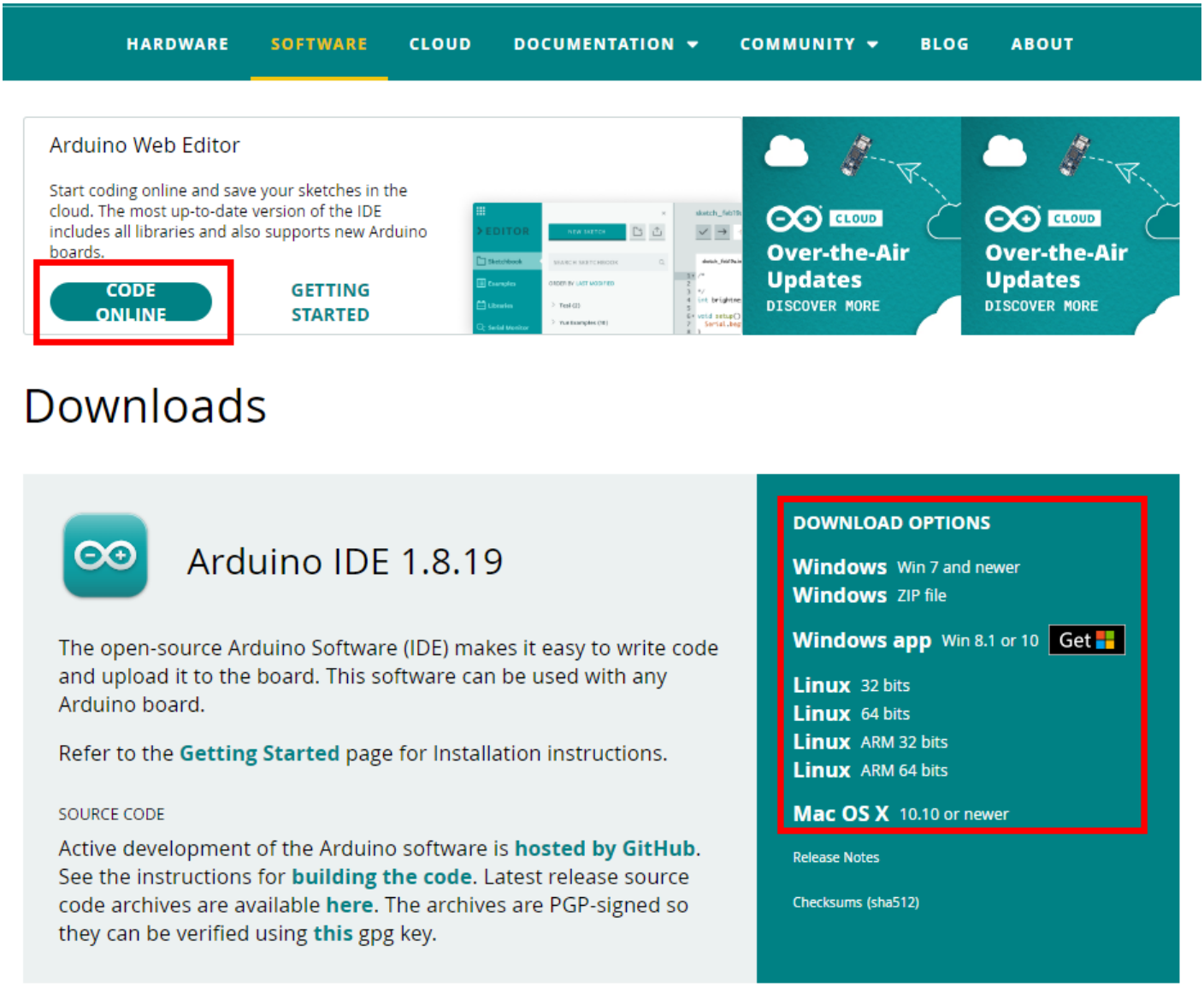
Software del Arduino

El microcontrolador en la placa Arduino se programa mediante dos tipos de programación:

OPCIÓN LENGUAJE POR CÓDIGO

Recomendable a partir de secundaria. Es un lenguaje basado en Wiring y permite la programación del Arduino en un entorno de desarrollo (basado en Processing). El programa se llama **ARDUINO IDE** y se puede descargar desde la página oficial de Arduino: https://www.arduino.cc/en/software.

> Hay otra posibilidad que es utilizarlo online, con la ventaja de tener tus proyectos "en la nube" y no depender del equipo. OJO, TIENES QUE TENER INSTALADO EL SOFTWARE **CREATE AGENT**
>
> https://create.arduino.cc/getting-started/plugin/welcome



En los cursos de CATEDU se ha utilizado el lenguaje por código empezando desde cero en:

- **CURSO ARDUINO CON CÓDIGO** donde se trabaja con el Arduino y con diferentes sensores y actuadores, con o sin placa Shield Edubásica.
- **CURSO DE DOMOTICA CON ARDUINO** donde se realiza una maqueta de una casa controlada con domótica. También el curso ofrece la versión de hacer la misma maqueta utilizando lenguaje gráfico por bloques.

Recomendamos estas hojas resumen si vas a trabajar con código:

- En Español: enlaceDrive, enlaceGithub
- En Inglés: enlaceDrive, enlaceGithub, enlaceSpakrfun

OPCIÓN LENGUAJE GRÁFICO POR BLOQUES

Recomendado para primaria. Tenemos muchas posibilidades de lenguajes gráficos. Destacamos dos:

- **ARDUINOBLOCKS** se trabaja online, muy visual y muy amigable. Está adaptado tanto para trabajar tanto Arduino como muchas placas controladoras y en el aula. Podemos verlo en los siguientes cursos:
  - **CURSO ROVER CON ARDUINO** aunque no se utiliza un Arduino, sino un NodeMCU pero la programación es exactamente igual
  - **CURSO DE ARDUINO CON ARDUINOBLOCKS** donde se utiliza el Arduino con una placa protoboard
  - **CURSO ARDUINOBLOCKS EN EL AULA** donde se utiliza la Shield TDR-STEAM
  - **CURSO ESP32 EN EL AULA** donde también utiliza la Shiedl TDR Steam pero la placa no es un Arduino, sino ESP32, la programación es exactamente igual.
- **MBLOCK** Basado en Scratch. Aunque es un programa especializado en el robot comercial mBot, (basado en Arduino), el mismo programa está adaptado para programar Arduino.
  - **CURSO ARDUINO CON MBLOCK** se utiliza Arduino y placa Protoboard
  - **CURSO DE ECHIDNA CON MBLOCK** se utiliza la Shield Echidna
  - **CURSO DE MBOT** se utiliza el robot mBot

Otros softwares para programar con bloques

- **Snap4Arduino** https://snap4arduino.rocks/run/ Online, libre... ver compartiva vs mBlock
- **S4A** https://s4a.cat/

> **EN VIVO ¿Qué es eso?**
> Existe una posibilidad de utilizar la placa "en vivo" (lo normal es "cargar" el programa en la placa).
> Es decir, interactuando con el ordenador. El programa está en el PC, En la placa hay un firmware que le dice que este a las órdenes del PC. De esta manera podemos por ejemplo:
> - Ordenador ➡ Placa: pulsando la tecla espacio que se encienda el led 13
> - Placa ➡Ordenador: Que por pantalla nos muestre la cantidad de luz, o que si es de noche que suene una canción..
>
> Que nosotros sepamos, sólo hay dos programas de hacer esto;
> - mBlock
> -Snap4Arduino

VENTAJAS E INCONVENIENTES LENGUAJE GRÁFICO POR BLOQUES vs CÓDIGO

El lenguaje gráfico por bloques es un lenguaje **sencillo de utilizar**, nos evita tener en cuenta muchas **librerías** y **cálculos.**

> Otra ventaja, es que el lenguaje por bloques es el único que permite programación "en vivo"

Por ejemplo, la instrucción leer valor distancia el sensor ultrasonidos, mediante programación por bloques es

mientras que en código es

```
double distancia;


double fnc_ultrasonic_distance(int _t, int _e){
  unsigned long dur=0;
  digitalWrite(_t, LOW);
  delayMicroseconds(5);
  digitalWrite(_t, HIGH);
  delayMicroseconds(10);
  digitalWrite(_t, LOW);
  dur = pulseIn(_e, HIGH, 18000);
     // devuelve cuanto tarda el pulso alto en microseg; 18000 es el tiempo a esperar limite
  if(dur==0)return 999.0;
  return (dur/57);
     // la velocidad del sonido es 344m/s = 34400 cm/seg = 0,0344 cm/microseg
    // como v=e/t luego e = v*t y como cuenta la ida y la vuelta distancia = v*t/2
    // luego distancia = 0,0344/2 * dur = dur/57
}

void setup()
{
   pinMode(6, OUTPUT);
  pinMode(5, INPUT);

}



void loop()
{
```

```
  distancia = fnc_ultrasonic_distance(6,5);


}
```

Como se puede ver en **código**, tiene que calcular la distancia haciendo cálculos del tiempo de rebote del eco, mientras que el **gráfico** es sumamente sencillo y se centra en el objetivo del algoritmo a crear, no en lo accesorio. Esto hace que un lenguaje gráfico por bloques se puede aplicar desde los 8 años.

Por otra parte, el lenguaje **código tiene todo el potencial,** es decir, no todo está en los lenguajes gráficos. Si se quiere cosas más avanzadas, hay que recurrir al código.

Un lenguaje **gráfico** se convierte en lenguaje **código**, pero **al revés no se puede.**

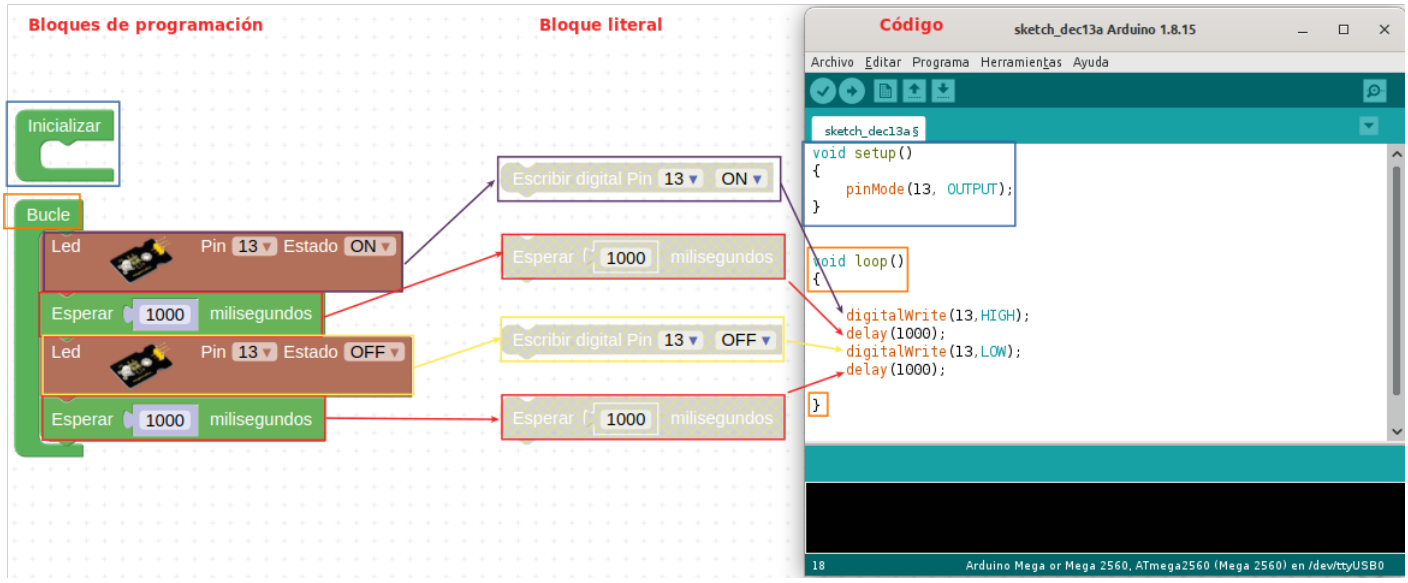


Imagen Federico Coca Notas sobre ESP32 STEAMakers CC-BY-SA

El lenguaje **código** se traduce en lenguaje **máquina** (ceros y unos) entendible para el microprocesador, **pero al revés no se puede**.

En este vídeo, en mi opinión se olvida de mBlock, Snap4Arduino, S4A pero puedes ver un vistazo de los diferentes editores

> Vale, me decanto por EDITOR **GRÁFICO** (bloques) y permita la opción **EN VIVO** ¿Cual es mejor mBlock o Snap4Arduino ?
>
> Ver MBLOCK VS SNAP4ARDUINO

OPCIÓN SIMULACIÓN

Incluimos dentro del apartado de Software los diferentes programas que hay para simular placas electrónicas como Arduino, ESP32, etc...

Tinkercad

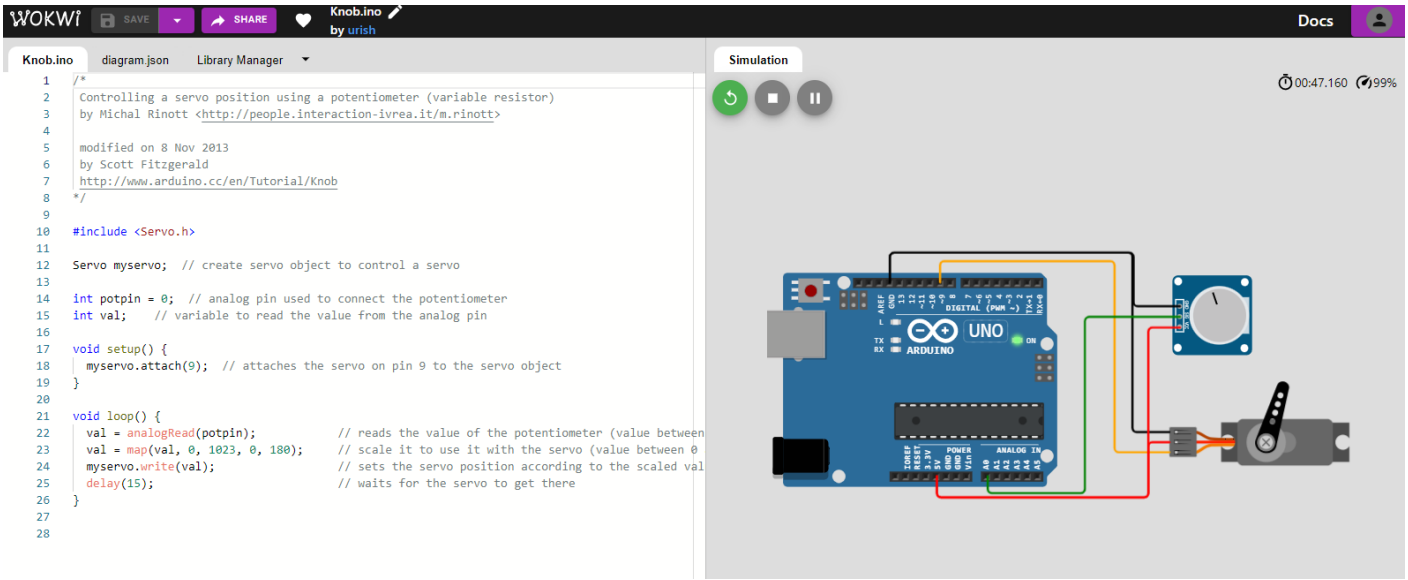
Esta herramienta https://www.tinkercad.com aparece en el **Curso Arduino con código** en la práctica Comunicación entre dos Arduinos, pero también es una plataforma que sirve para hacer los diseños de elementos 3D, ver curso **Impresión 3D con Tinkercad**

Tiene la ventaja que es aplicación **online**, muy **visual** y buscan un reflejo de la práctica **real,** además de estar la herramienta **adaptada al aula** (gestión de alumnos y proyectos). Como desventajas podemos decir que no tiene mucha variedad de componentes electrónicos y la simulación es algo lenta.

https://www.youtube.com/embed/pXEv0wxW9Jo?rel=0

Wolwi

Si Tinkercad se queda corto, puedes probar esta plataforma https://wokwi.com/ con muchas posibilidades. Es **online** y puede trabajar con **multitud de placas**: ArduinoUno, ESP32, Raspberry,,,,
Como única desventaja que encontramos, es que echamos de menos la realidad de Tinkercad, por ejemplo no puedes poner una placa protoboard para realizar las conexiones, pero a cambio se gana simplicidad de cableado.



UnoArduSim

Es una **aplicación local.** **UnoArduSim** además es una aplicación portable fácil de instalar y con los elementos de leds, motores servos ya preparados, ideal para ejemplos sencillos y para examinar señales y no depender de Internet, pero no es tan versátil.

https://www.youtube.com/embed/WLJ_l4uGjXg?rel=0

Revision #1
Created 28 October 2022 19:23:04 by Javier Quintana
Updated 28 October 2022 19:23:56 by Javier Quintana