MONTAJE 1 SEMÁFORO mediante código (avanzado y optativo)

MONTAJE 1 SEMÁFORO MEDIANTE CÓDIGO

PRIMERO QUÉ ES "MEDIANTE CODIGO"

Es la forma de trabajar de forma profesional el Arduino: con su lenguaje código, en realidad, los otros lenguajes traducen el programa gráfico en lenguaje código Arduino, es decir son meros intermediarios.



El programa se puede descargar de su web oficial, aunque también hay una versión online

https://www.arduino.cc/

Copyright 2026

SEGUNDO LA CONFIGURACIÓN

Aquí no hay que instalar ningún Firmware, pues el código, o sea tú programa, es el mismo “firmware”. Digamos que no necesitas intermediarios si tratas con el agricultor ;)

TERCERO EL EJEMPLO SEMÁFORO CON CÓDIGO

Aquí es donde vemos que la programación no es apropiada para Primaria, y la explicación de cada línea necesitaría un curso entero, como éste, pero si eres un poco *pito* seguro que eres capaz de leerlo intuitivamente sin problemas.

```
/* Semáforo Arduino
  Leds conectados a pines 11,12,13 = EchidnaShield */
int verde = 11;
int amarillo = 12;
int rojo = 13;

void setup()
{
 pinMode(verde, OUTPUT);
 pinMode(amarillo, OUTPUT);
 pinMode(rojo, OUTPUT);
 Serial.begin(9600); //inicializa la comunicación Serial
}

void loop()
{
 Serial.println("Semaforo - Inicio"); //Escribe el texto
 digitalWrite(verde, HIGH);
 Serial.println("Semaforo - Verde"); //Escribe el texto
 delay(2000);
 digitalWrite(verde, LOW);
 digitalWrite(amarillo, HIGH);
```

Copyright 2026

```
  Serial.println("Semaforo - Amarillo"); //Escribe texto
  delay(1000);
  digitalWrite(amarillo, LOW);
  digitalWrite(rojo, HIGH);
  Serial.println("Semaforo - Rojo"); //Escribe el texto
  delay(2000);
  digitalWrite(rojo, LOW);
 }
```

Este programa se escribe (o copia y pega ¡es un texto !!) en el software del Arduino y se carga en la placa, en el vídeo se ve muy bien:

https://www.youtube.com/embed/S8dQdCqOtto

Desventaja

En este programa queremos que te fijes en una **desventaja**: La interacción con el ordenador se pierde frente a mBlock y Snap4Arduino:

- En los lenguajes gráficos, tenemos a **interacción con el ordenador** igual que en Scratch: si te fijas en la pantalla el dibujo del semáforo va cambiando de disfraz para representar los colores de la luz que se enciende a la vez que en el Arduino.
- En un programa grabado en el Arduino **perdemos esa interacción**, lo máximo que podemos visualizar es una ventana donde se representa en formato texto qué es lo que le está pasando al Arduino (en el ejemplo del semáforo sale "Semáforo - Verde, Semáforo-Rojo ….").

Esto lo vemos como una **desventaja** desde el punto de vista de la enseñanza de la programación pues perdemos el potencial de interactuar con los elementos del ordenador crear personajes, disfraces, sonidos, teclado…, es decir **el sprite** (el oso panda que aparece por defecto en mBlock, la flecha que aparece por defecto en Snap4Arduino o el que nosotros creamos, como este ejemplo el dibujo del semáforo).

Ventaja

Pero... (siempre hay un “pero” para estropear la fiesta) la programación en código tiene una **ventaja**: Se graba en el Arduino, no hay ningún intermediario entre nuestro programa y el Arduino (en mBlock y Snap está el ordenador), quien manda en el Arduino es nuestro programa, no nuestro ordenador, esto se traduce en: **rapidez !**, en mBlock veremos que podemos grabar nuestro programa en Arduino, pero perdemos la interacción con el sprite, como lo veremos más adelante.

Conclusión

Profesionalmente es mejor utilizar lenguaje con código pero en la enseñanza es mejor el lenguaje gráfico. Lo hemos visto en el apartado ¿Cómo se programa Echidna?. Es una Shield Educativa, y lo lógico es utilizar un lenguaje adaptado al nivel educativo: Gráfico. Pero ... ¿cual?

Revision #4
Created 2022-03-31 08:48:41 CEST by Equipo CATEDU
Updated 2024-12-20 14:58:58 CET by Juanjo Mejías

Copyright 2026