

La memoria EEPROM permite guardar información en la memoria de Arduino de forma no volátil, es decir, que permanecerá la información guardada aunque desconectemos la alimentación eléctrica.

Es una memoria perfecta para guardar valores de configuración o para almacenar el estado de una variable y recuperarla después de un corte de alimentación.

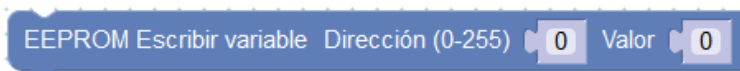
La memoria EEPROM tiene una capacidad de 1024 bytes (1kByte), pero en ArduinoBlocks las variables usadas internamente son de tipo decimal y ocupan 4 bytes cada una por lo que el número máximo de variables que podemos guardar en la memoria EEPROM será 256.

$1024 \text{ bytes} / 4 \text{ bytes por variable} = 256$

BLOQUES



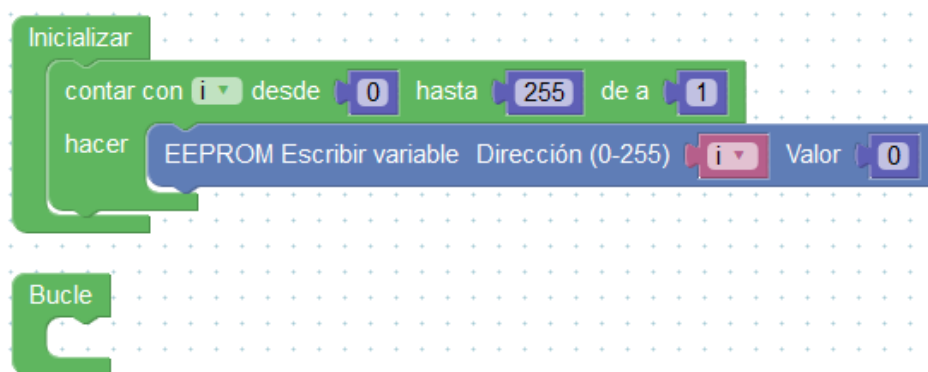
Lee el valor de una posición de memoria de la EEPROM.



Guarda un valor en una posición de memoria de la EEPROM.

La memoria EEPROM en la placa Arduino normalmente viene pregrabada de fábrica al valor 0xFF (255) en cada uno de sus bytes, por lo que en algunos casos es importante inicializarla a los valores que nos interese según la aplicación.

Ejemplo: Iniciar toda la memoria EEPROM a 0

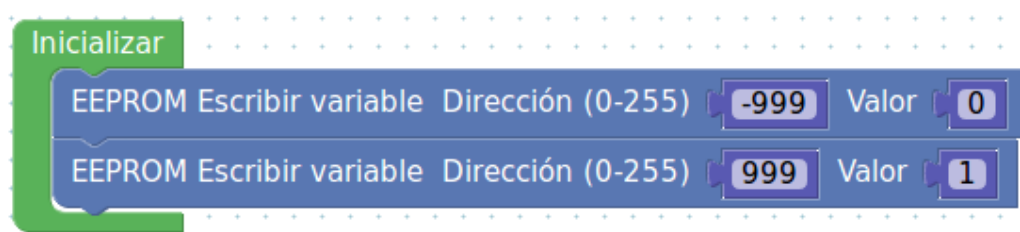


PRÁCTICA 17.1 Guardar temperatura máxima y mínima

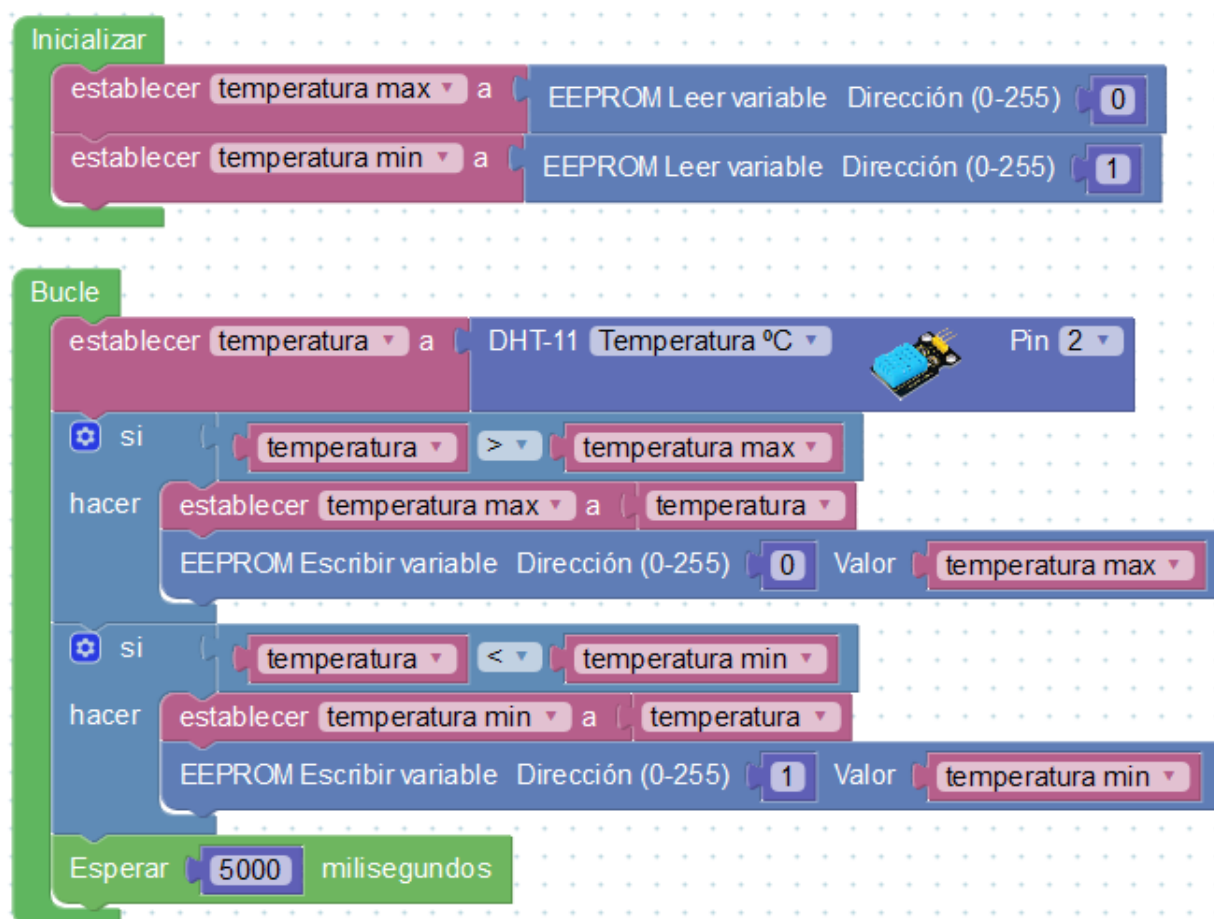
CÓDIGO DE PROYECTO:

Guardar el valor máximo y mínimo de temperaturas registrado con un sensor DHT-11 y que esta información se mantenga aunque desconectemos la alimentación eléctrica.

Inicialización de la memoria EEPROM:



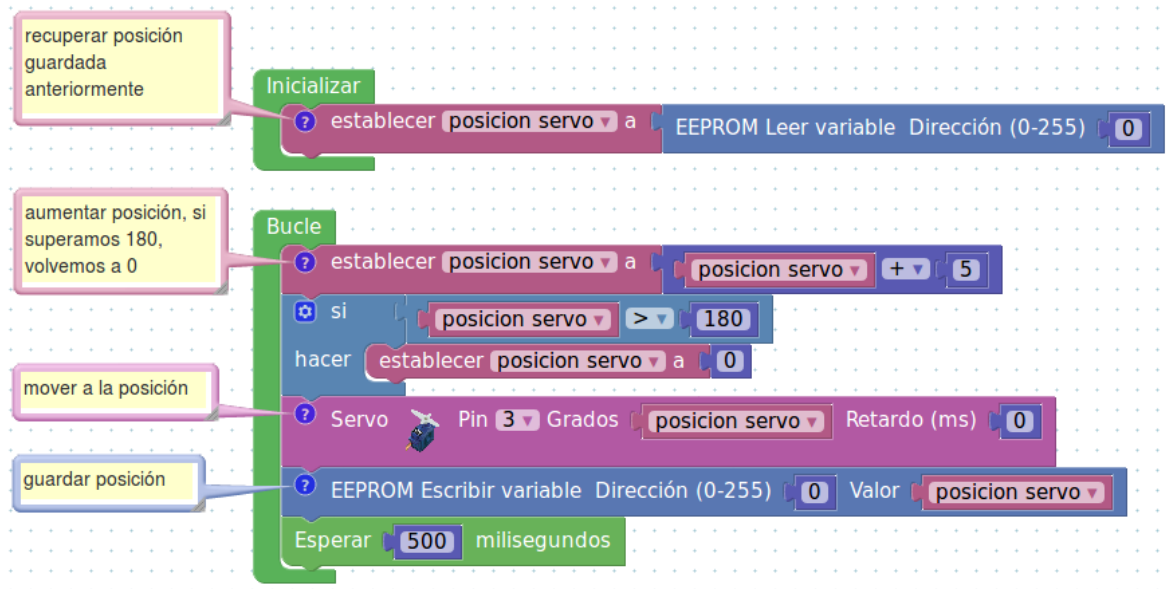
Programa para el funcionamiento normal:



PRÁCTICA 17.2 Guardar posición del servo

CÓDIGO DE PROYECTO:

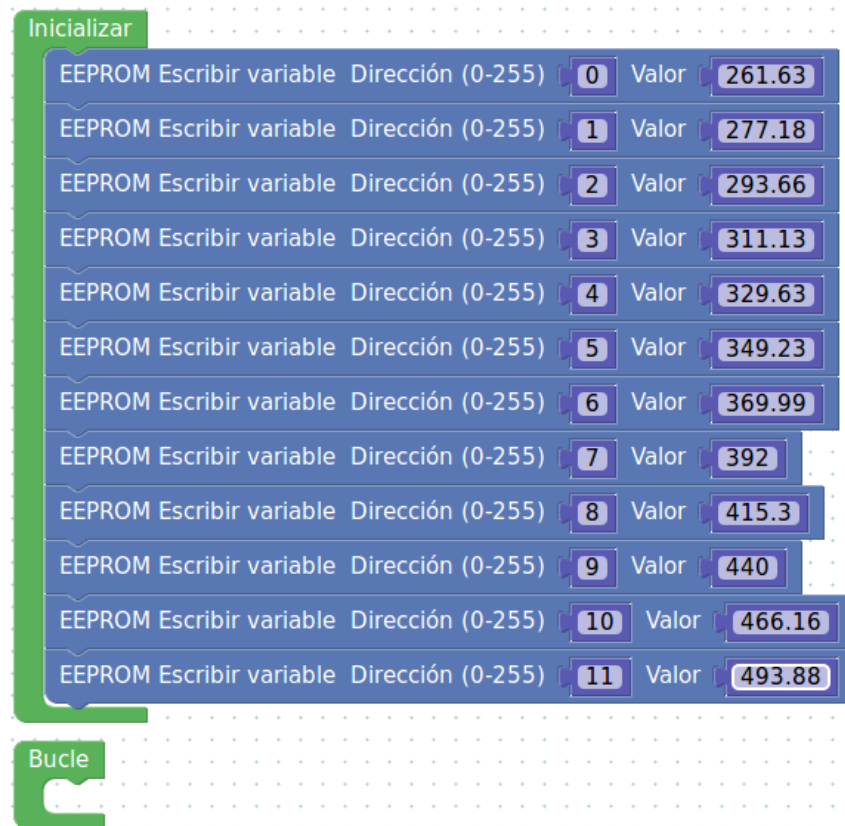
Realizar un movimiento progresivo del servo, en caso de corte de alimentación al reiniciar seguirá por la misma posición donde estaba. Cada vez que el servo se mueve se guarda su posición en la memoria EEPROM y en caso de reinicio se recupera para seguir por la misma posición.



PRÁCTICA 17.3 Melodía desde EEPROM

CÓDIGO DE PROYECTO:

Iniciación



Programa en funcionamiento normal:

