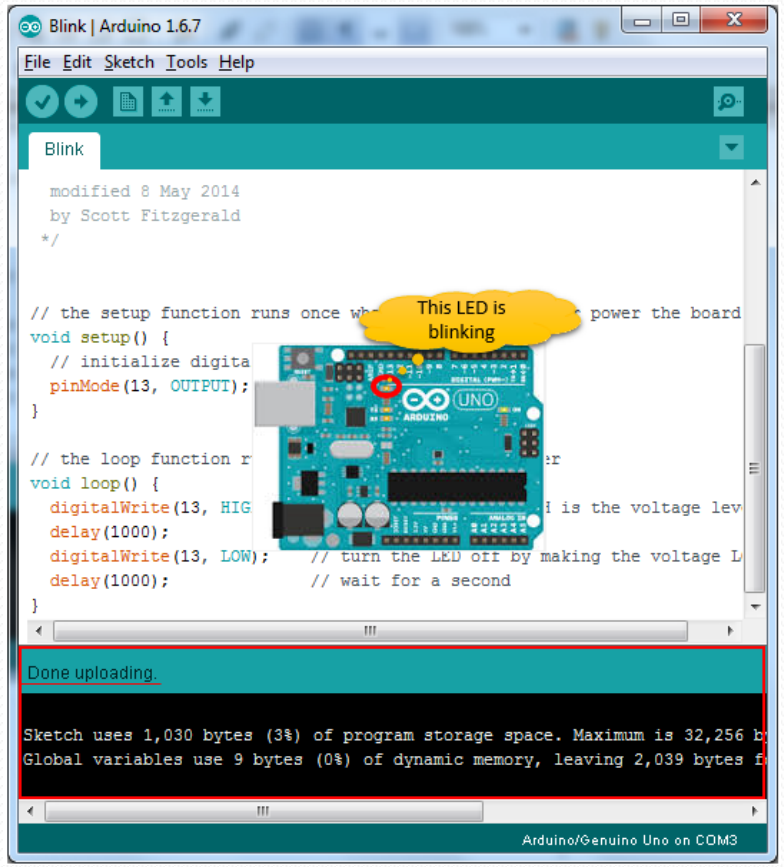
ARDUINO Y DCC

QUE SE PUEDE HACER CON ARDUINO PARA LA MAQUETA DIGITAL DCC

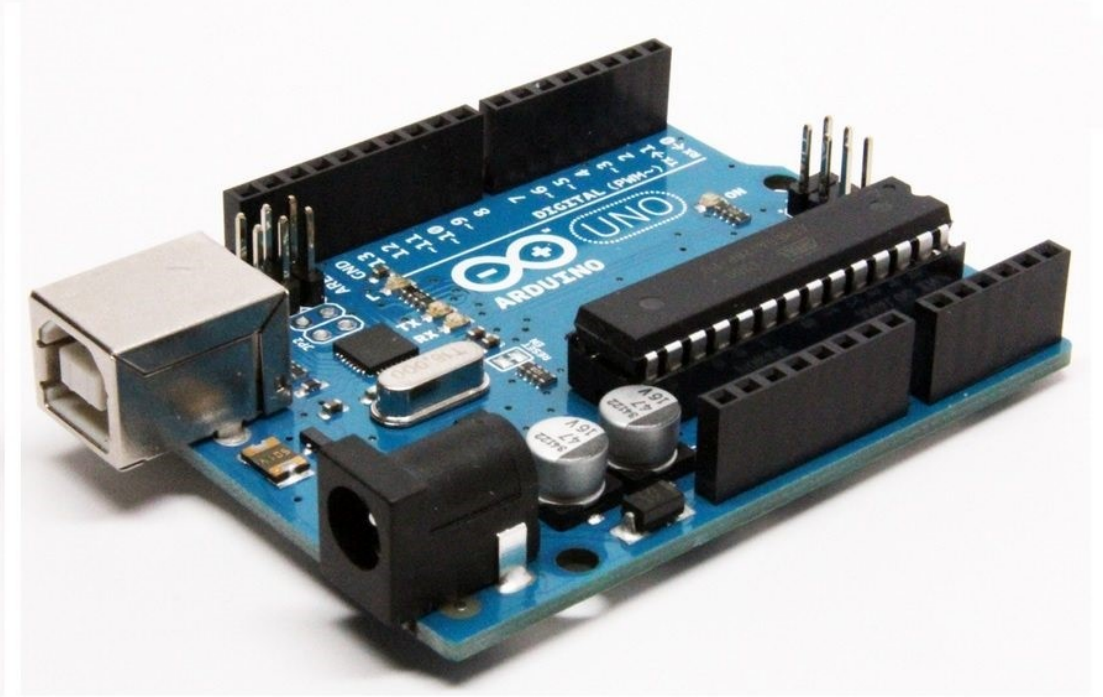
¿QUÉ ES ARDUINO?

- Arduino es una plataforma electrónica de código abierto basada en hardware y software de uso fácil. Está pensado para cualquier persona que haga proyectos interactivos.
- El software (Arduino IDE) hace que sea fácil escribir código (lenguaje C) y subirlo a la placa Arduino.
- Hay diferentes tipos de placas, básicamente son un microprocesador con diferentes tipos de entradas y salidas con conexión USB y de alimentación.



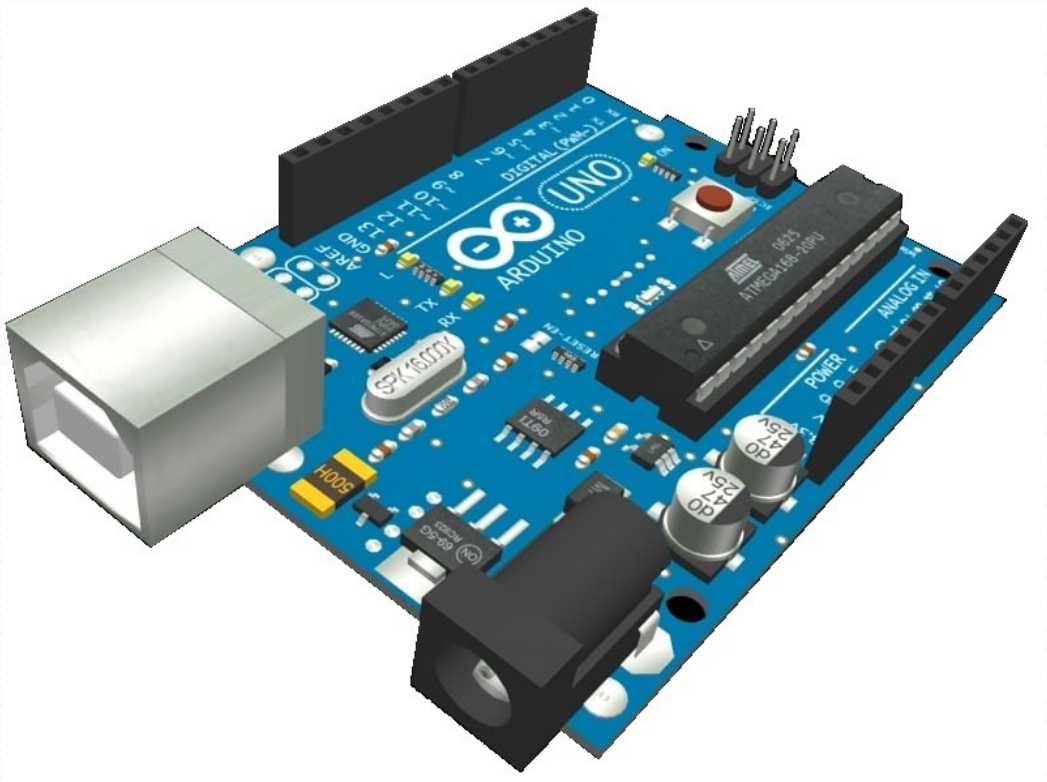
LA PLACA ARDUINO UNO

- La placa Arduino UNO es la más popular y utilizada de toda la familia Arduino



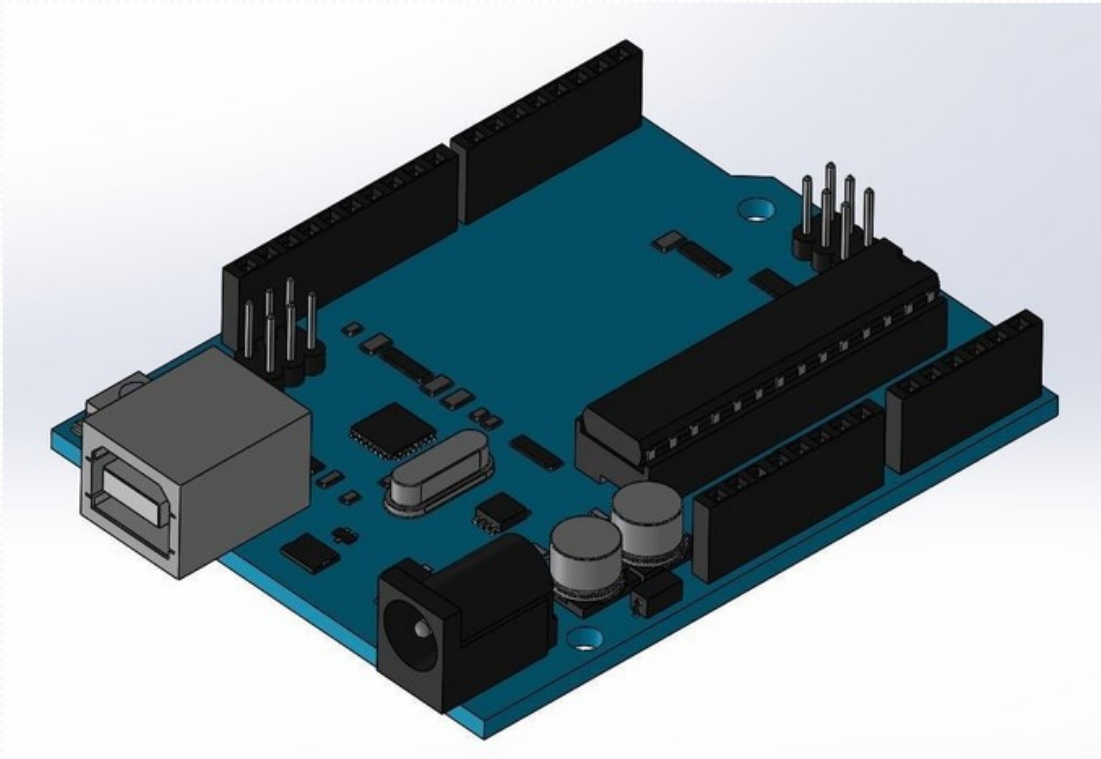
LA PLACA ARDUINO UNO

- Hay diferentes versiones de Arduino UNO. Con diferente tamaño pero básicamente igual está la Arduino Nano.



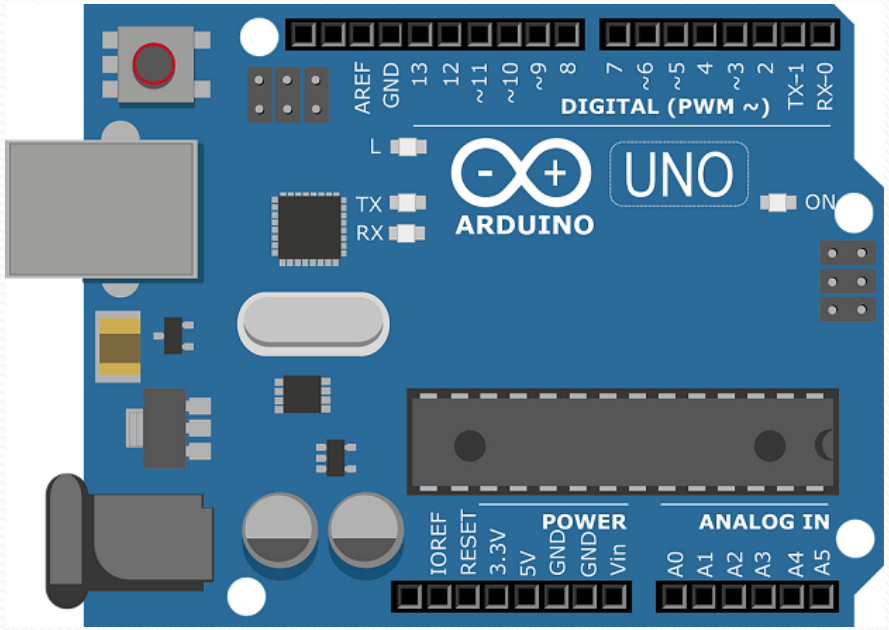
LA PLACA ARDUINO UNO

- Con diferentes microprocesadores y número de entradas y salidas hay otras placas Arduino (MEGA, DUE, Leonardo,…) pero todas se programan igual desde el software Arduino IDE.



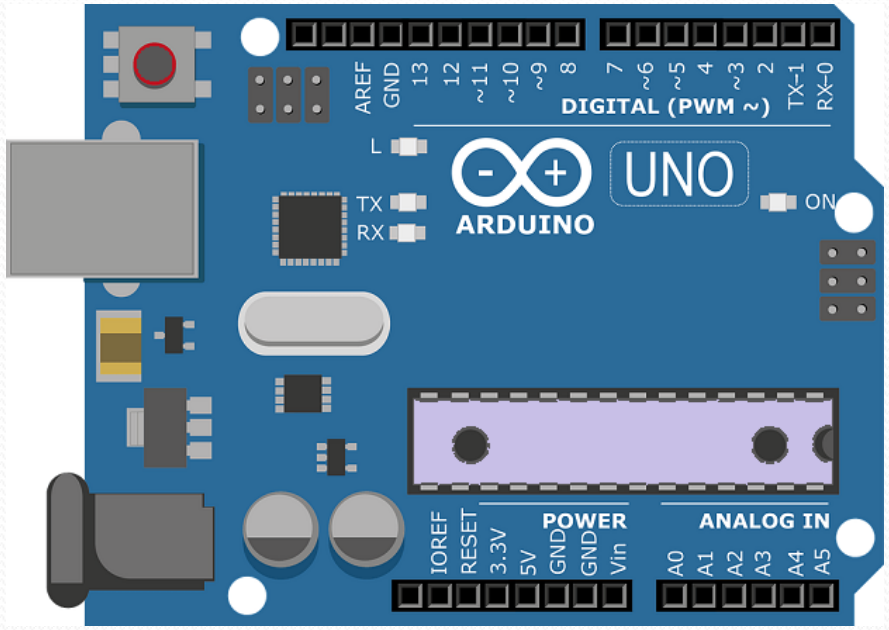
LA PLACA ARDUINO UNO

- Podéis decir a vuestro Arduino lo que tiene que hacer escribiendo código en el lenguaje de programación Arduino o copiando el código ya hecho por otros y descargarlo a la placa desde el Arduino IDE



CONOCIENDO LA PLACA ARDUINO UNO

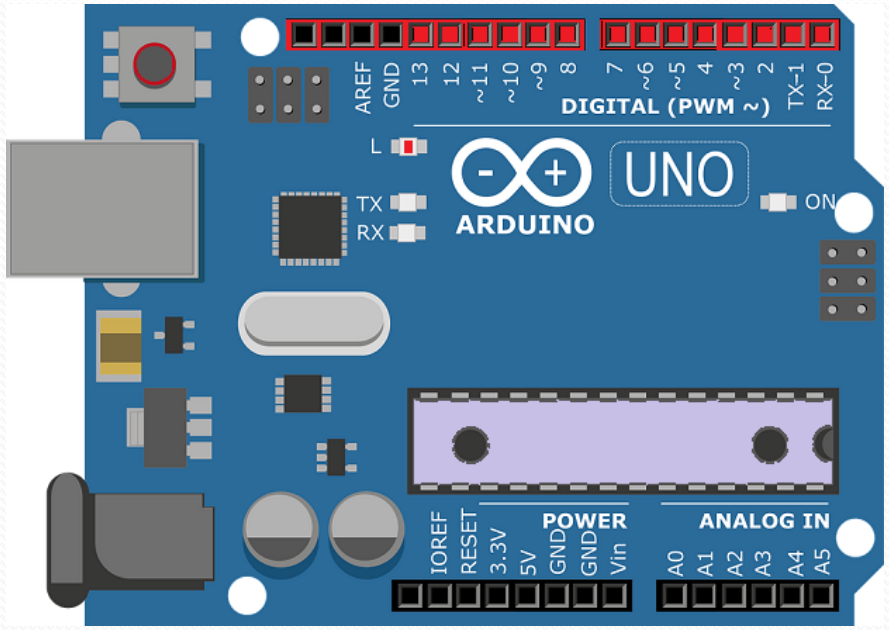
- Arduino Uno es una placa con un micro controlador basado en ATmega328P



Microprocesador

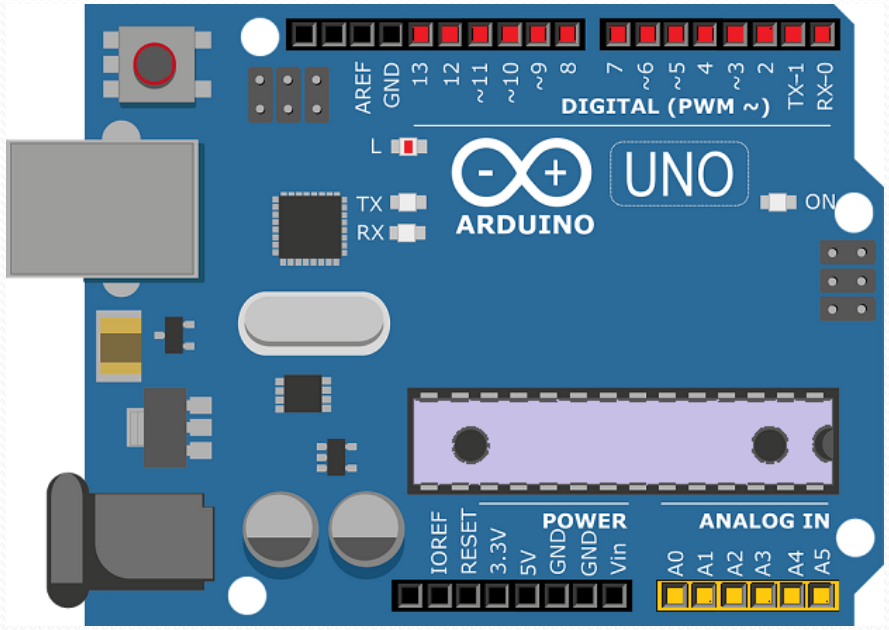
CONOCIENDO LA PLACA ARDUINO UNO

- Tiene 14 pines digitales de entrada / salida (de los cuales 6 se pueden utilizar como salidas PWM)



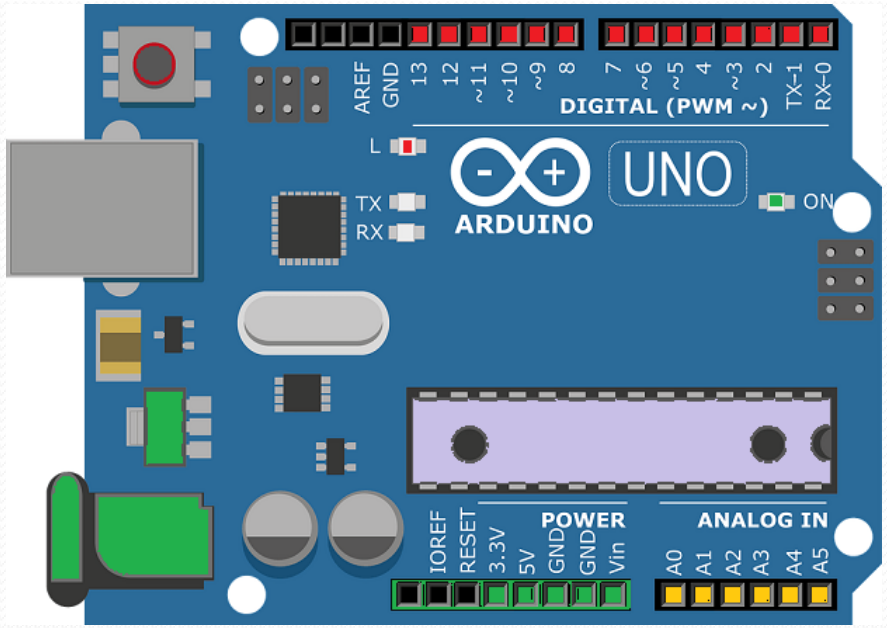
CONOCIENDO LA PLACA ARDUINO UNO

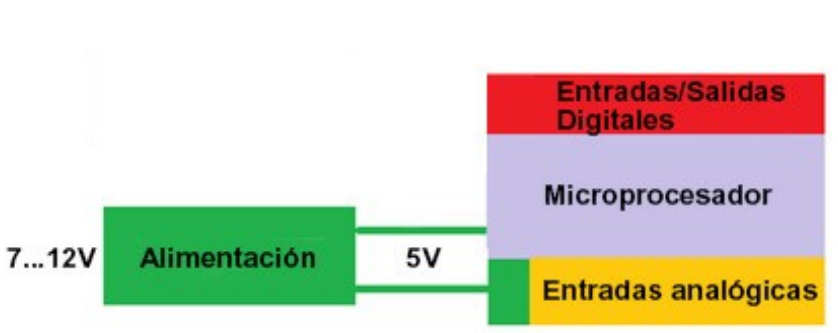
- Hasta 6 entradas analógicas. Pueden ser usados como pines digitales extra.



CONOCIENDO LA PLACA ARDUINO UNO

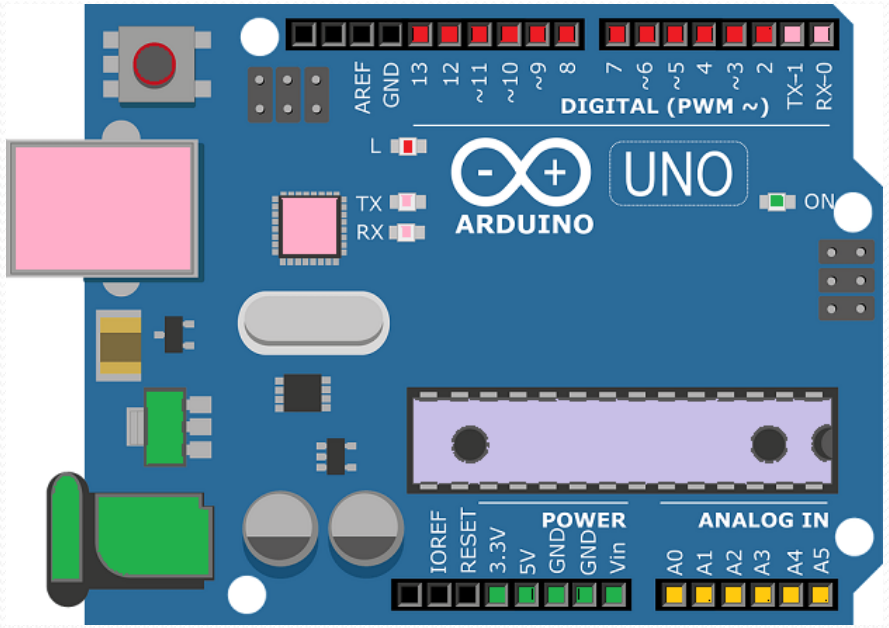
- Se puede alimentar externamente con tensión continua entre 7V y 12V. El microprocesador trabaja a 5V y en el conector hay disponibles las tensiones de alimentación, los 5V y 3.3V, ésta última solo da 50mA

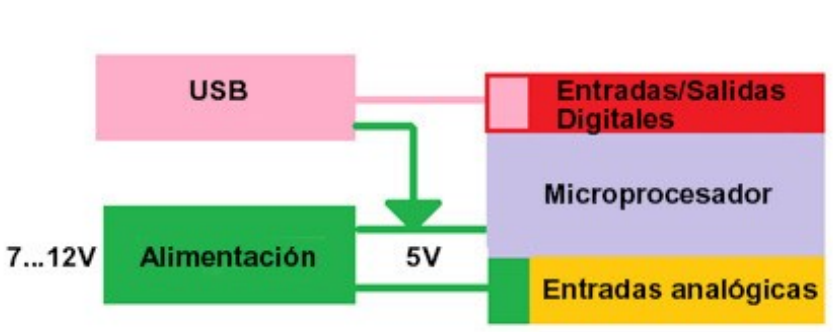




CONOCIENDO LA PLACA ARDUINO UNO

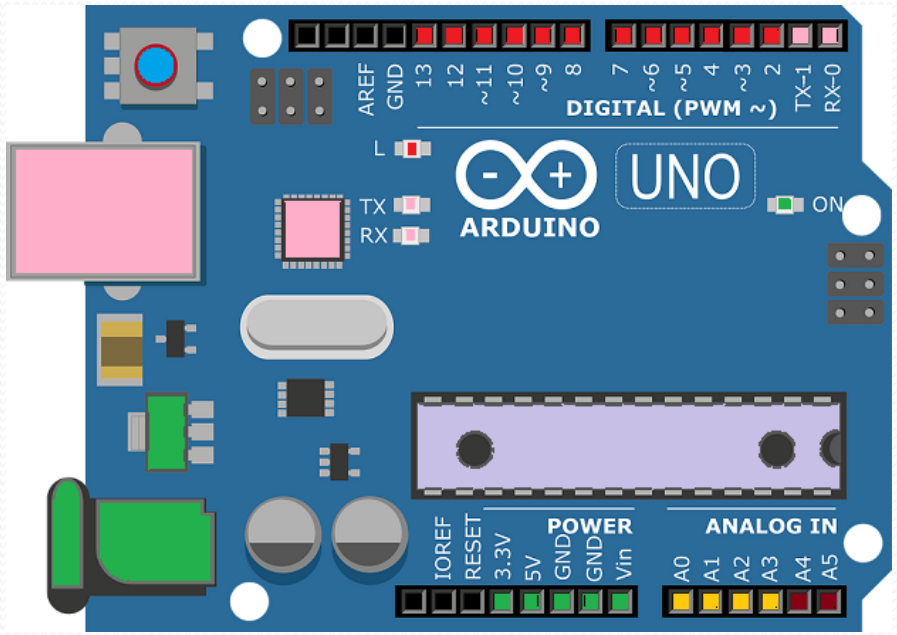
- Tiene una conexión USB para poder cargar la programación desde el Arduino IDE y comunicarse con él con un puerto serie, utiliza dos pines digitales. También puede ser alimentado solo desde USB si el consumo es pequeño.

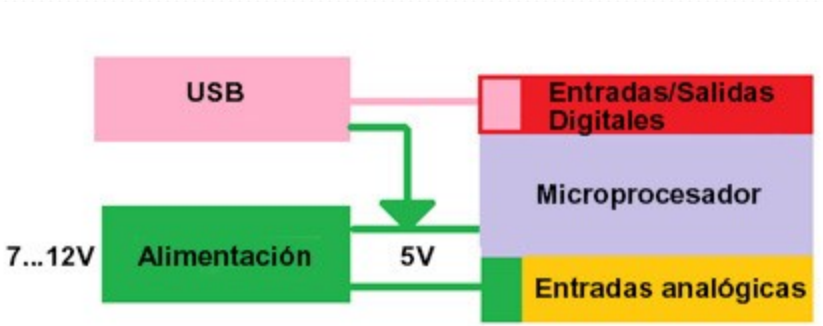




CONOCIENDO LA PLACA ARDUINO UNO

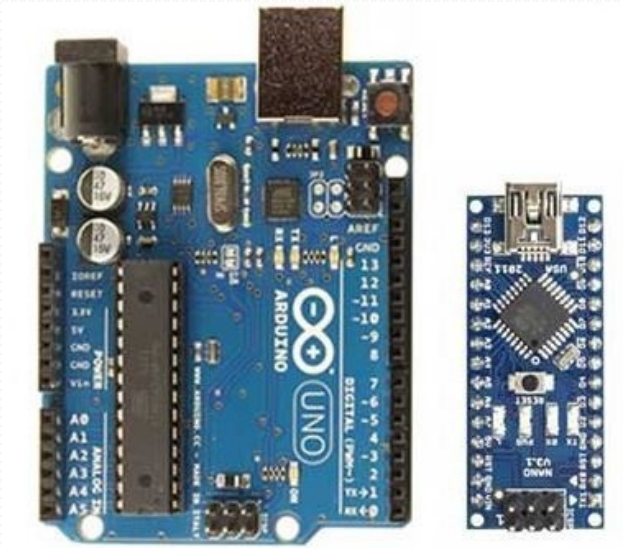
- Tiene una sèrie de LEDs indicadores: Alimentación, Puerto serie y pin13.
- Botón de Reset
- Dos de los pines (A4, A5) pueden funcionar como un bus I2C

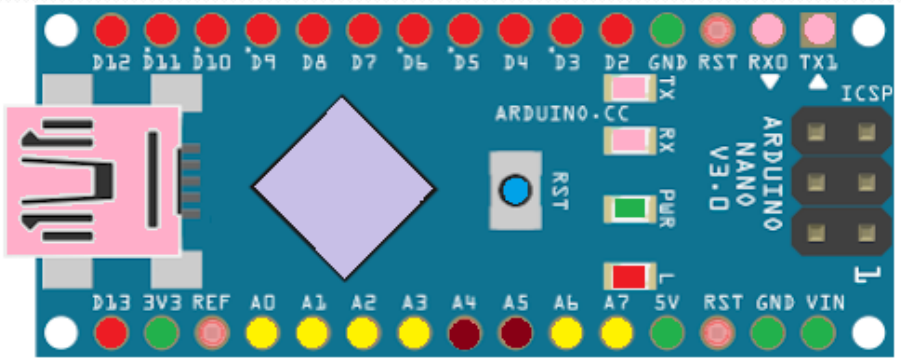


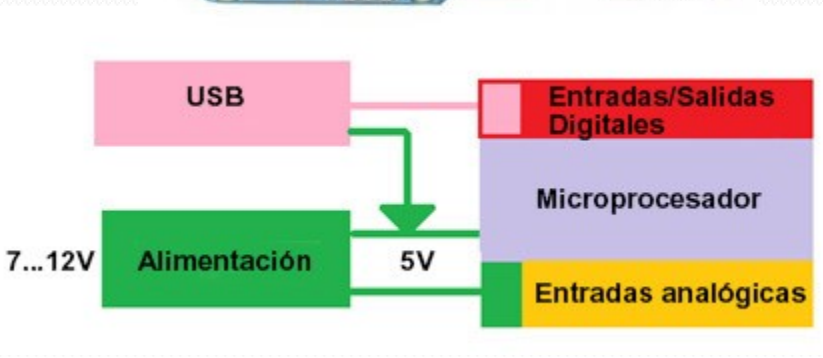


CONOCIENDO LA PLACA ARDUINO NANO

- El Arduino Nano, de medida mucho más pequeña que el Arduino UNO, tiene las mismas prestaciones ya que hace servir el mismo procesador. Han reducido conectores y medidas de la placa.

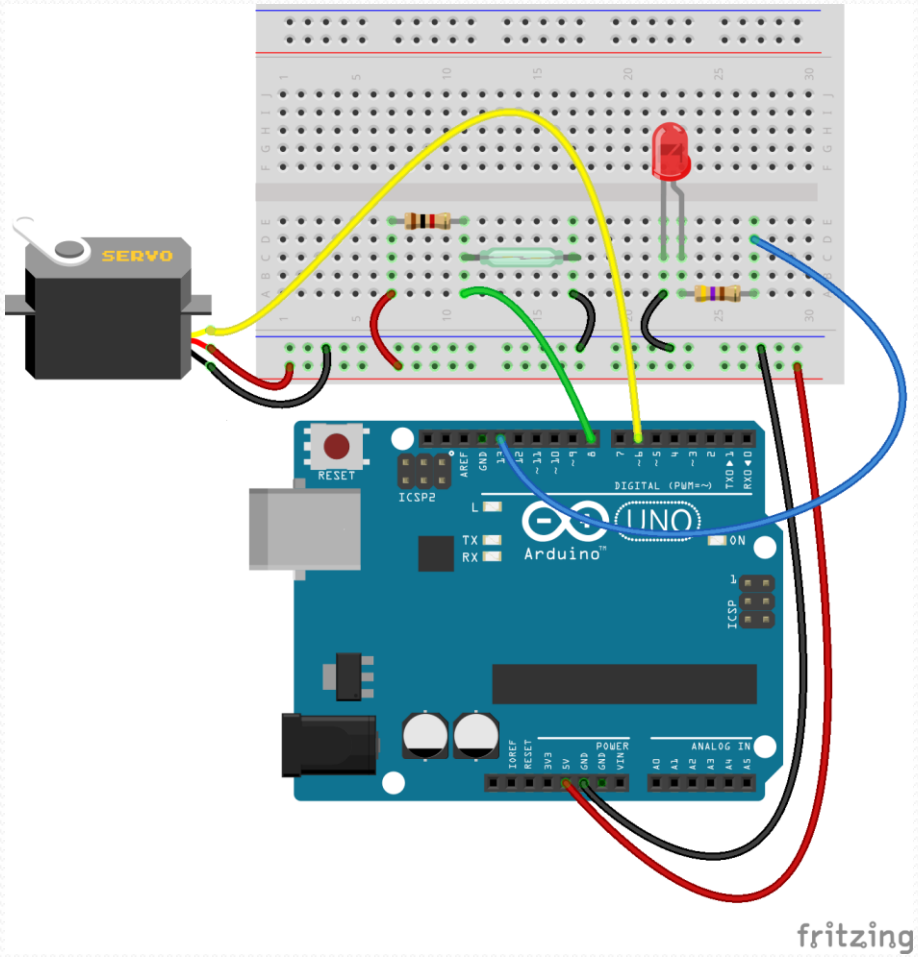






COMO CONECTAR LOS ELEMENTOS

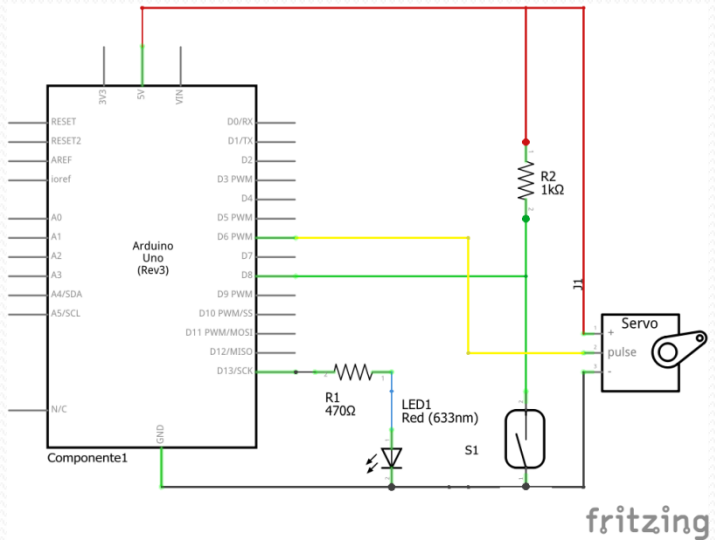
Para pruebas se puede hacer servir una placa dónde poner los elementos que no necesita soldadura para hacer las conexiones, sólo cables.



Los LED necesitan su resistencia, el pin da 5V y 20mA máximo

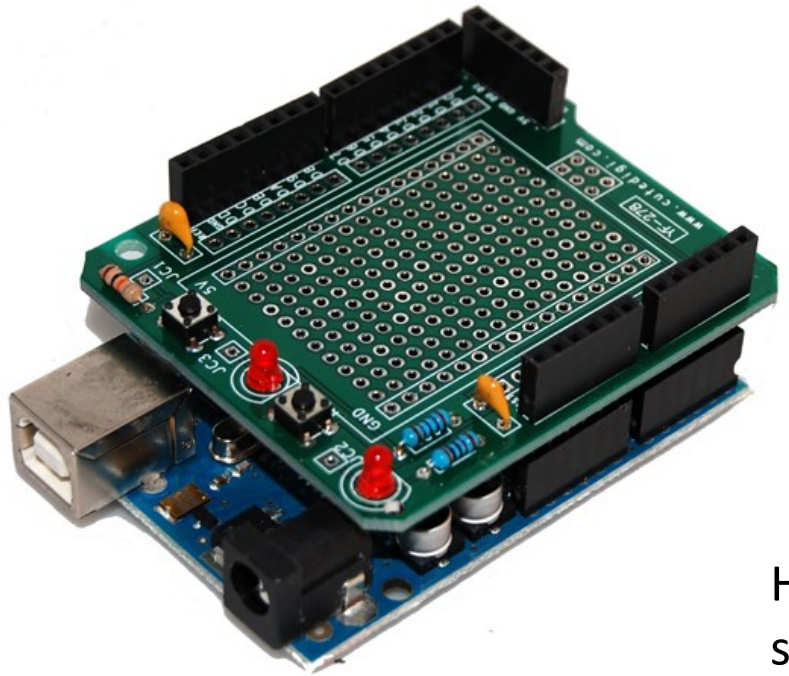
Los REED necesitan una resistencia de pull-up

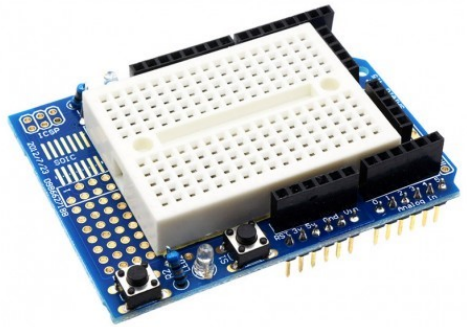
Los SERVOS normalmente se conectar a los pines PWM

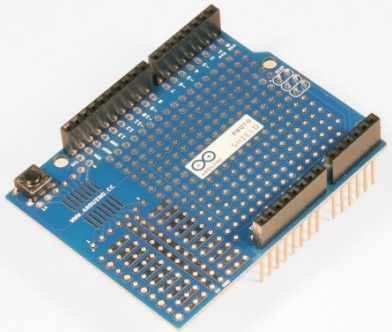


LAS PLACAS DE EXTENSIÓN O ‘SHIELD’

Un ‘shield’ en Arduino es una placa que se apila sobre el Arduino o sobre otro ‘shield’, de manera que nos permite ampliar el hardware o para dar funcionalidad extra a un Arduino.



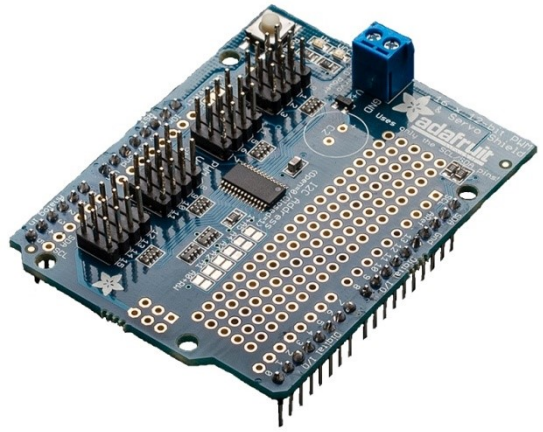


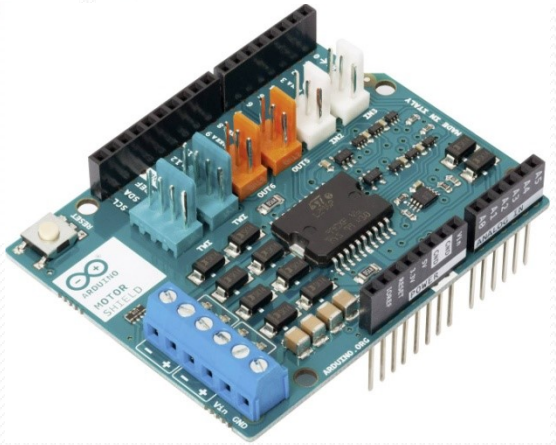


Hay placas ‘shield’ para prototipos con soldadura o con tablero

LAS PLACAS DE EXTENSIÓN O ‘SHIELD’

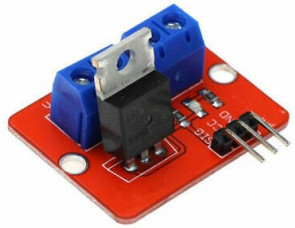
Otros ‘shield’ interesantes para las maquetes incluyen control de motores de corriente continua, motores paso a paso, relés, sensores, servos,...

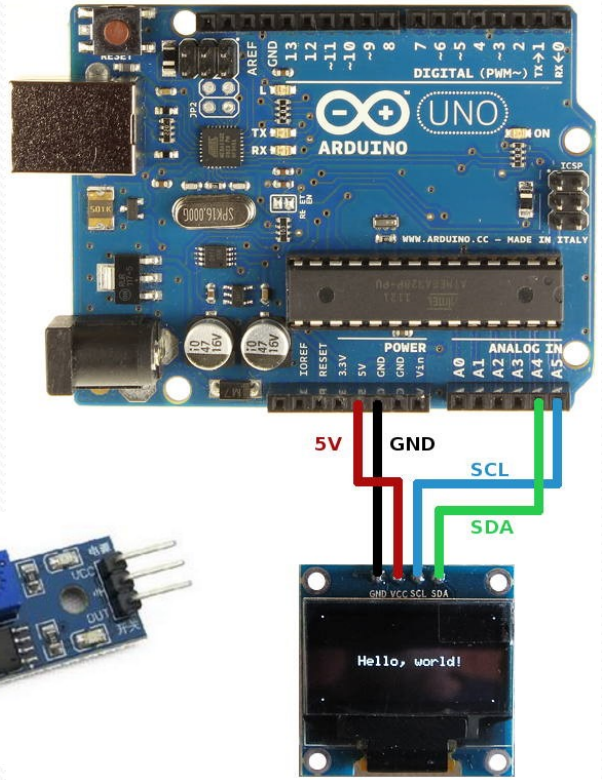


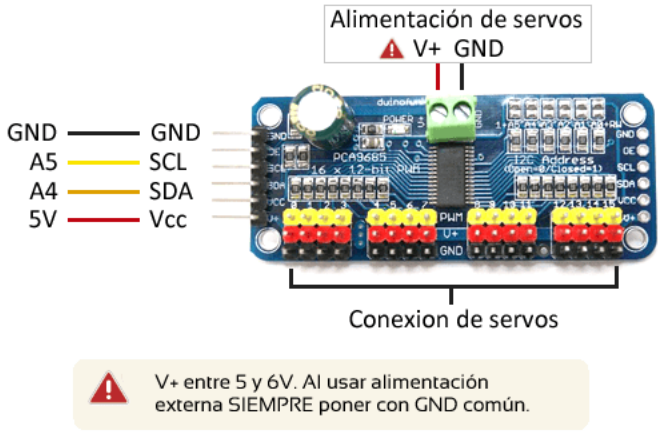


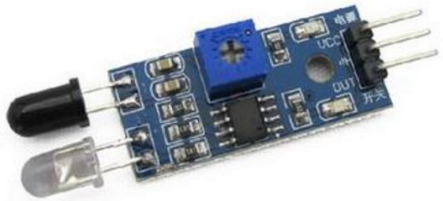
OTROS TIPOS DE PLACAS DE EXTENSIÓN

Otras placas de extensión no se pinchan en el Arduino, pero también pueden ser controladas desde los pines analógicos y digitales o por el bus I2C.
Hay de relés, servos, barreras de infrarrojos, pantallas OLED, transistores de potencia, ...



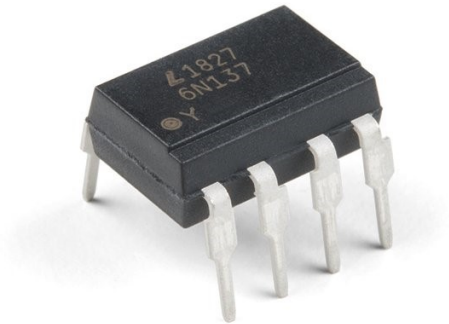




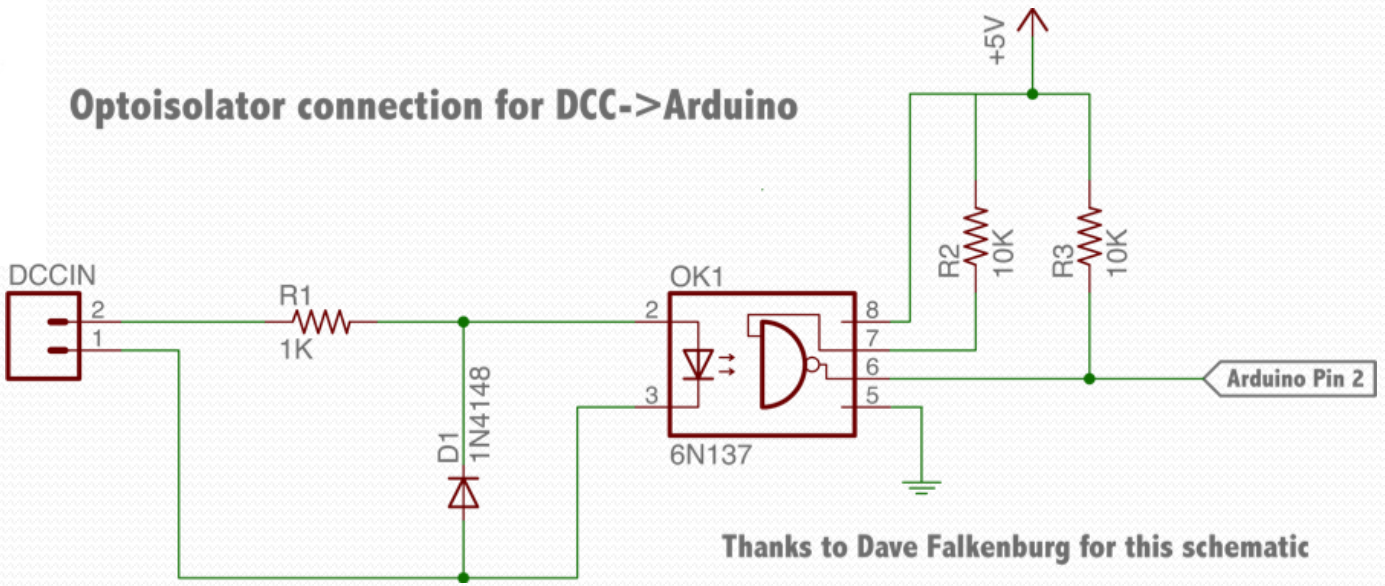


CONEXIÓN DE ARDUINO A DCC

Para hacer un descodificador de accesorios con Arduino necesitamos un circuito para leer la señal DCC y que a la vez nos aísle de las tensiones altas que tenemos en la vía perjudiciales para el Arduino, recordemos que trabaja a 5V.

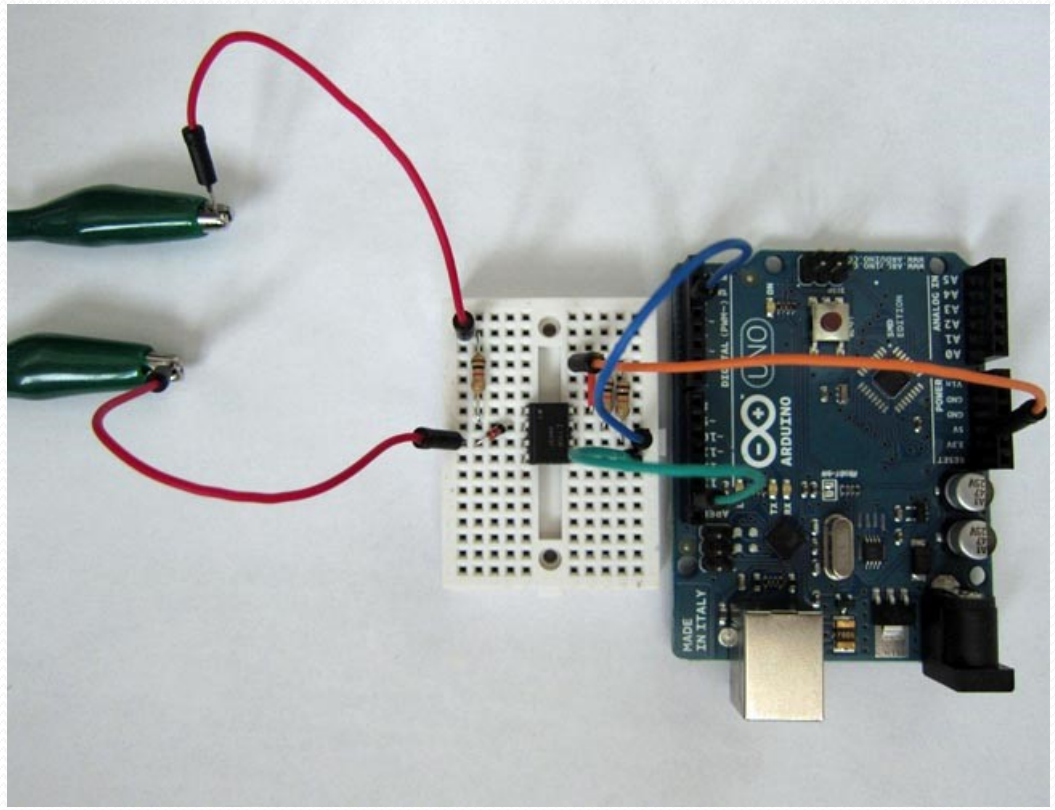


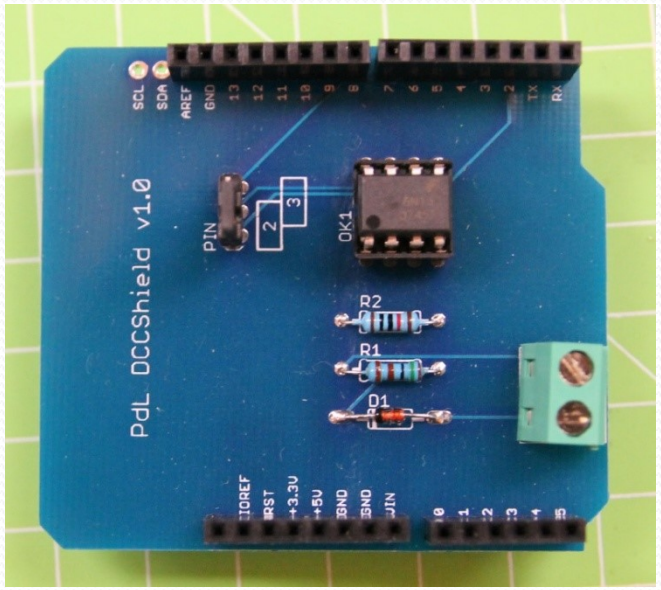
Esto se consigue mediante un optoacoplador, el 6N137 que es lo suficientemente rápido para leer la señal DCC y entregársela al Arduino por el pin 2



CONEXIÓN DE ARDUINO A DCC

Desgraciadamente no hay una ‘shield’ comercial así que la tendremos que montar nosotros con una placa prototipo o hacerla desde un diseño de una web

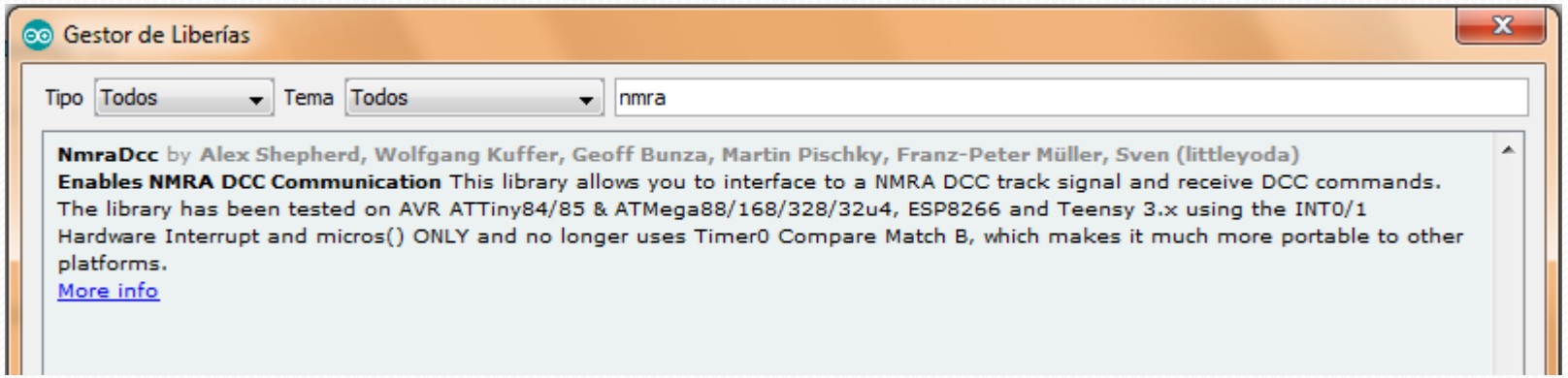




https://github.com/lucadentella/arduino-dccshield

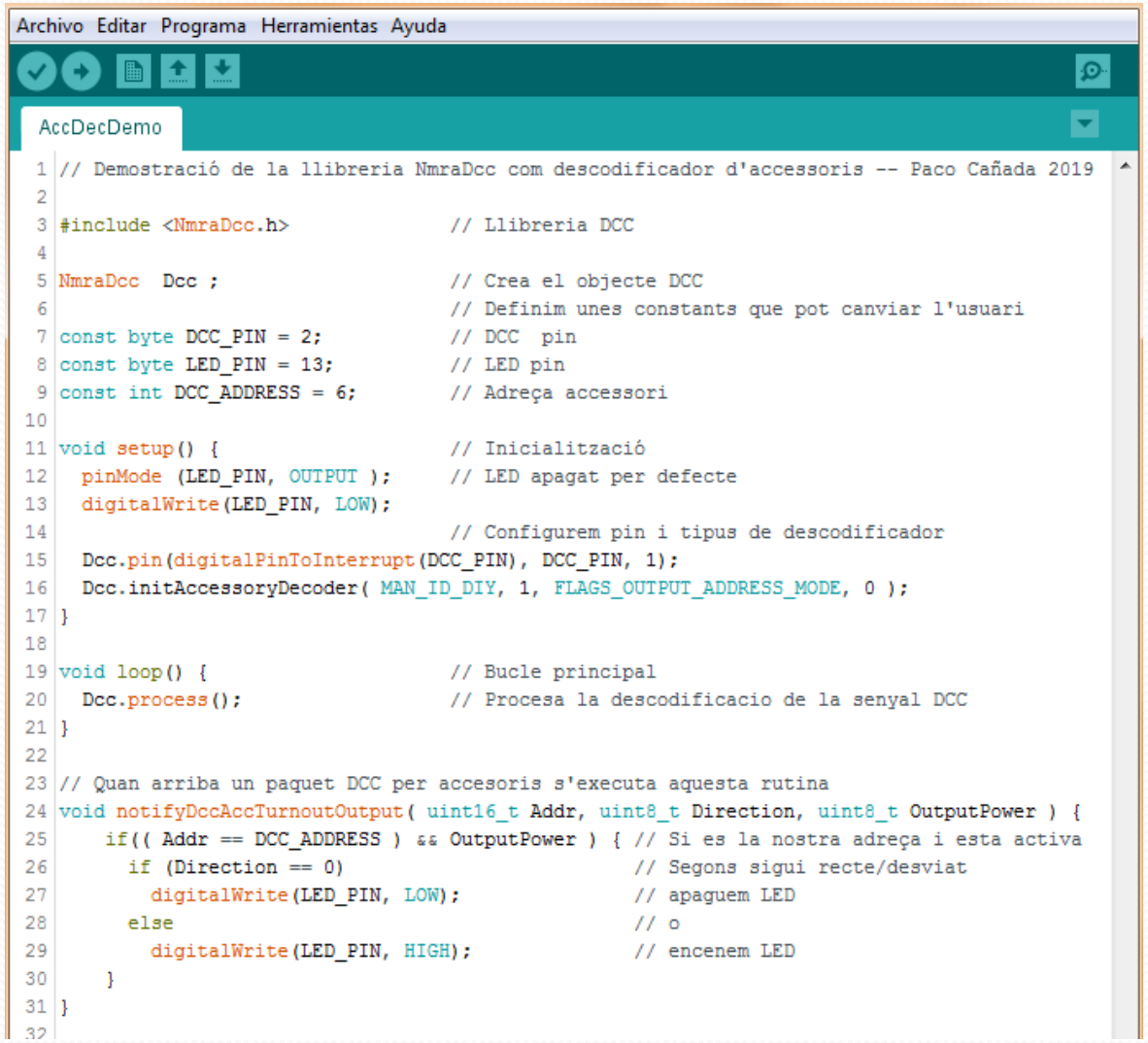
CONEXIÓN DE ARDUINO A DCC

La señal DCC llegará al pin 2 del Arduino pero para hacerla servir con nuestro programa hará falta descodificarla. Por suerte la comunidad Arduino ha puesto a disposición una librería para el Arduino IDE para poder recibir y entender la señal DCC .



Desde el mismo Arduino IDE se puede instalar. Con la librería, el trabajo de hacer, por ejemplo, un descodificador de accesorios DCC es bastante simple.

CONEXIÓN DE ARDUINO A DCC

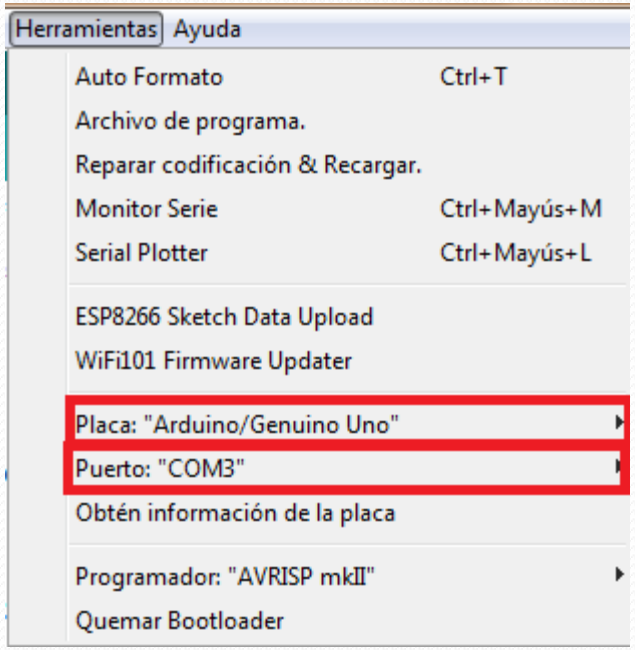



```
1  // Demostració de la llibreria NmraDcc com descodificador d'accessoris -- Paco Cañada 2019
2
3  #include <NmraDcc.h>              // Llibreria DCC
4
5  NmraDcc  Dcc ;                   // Crea el objecte DCC
6                                   // Definim unes constants que pot canviar l'usuari
7  const byte DCC_PIN = 2;          // DCC  pin
8  const byte LED_PIN = 13;         // LED pin
9  const int DCC_ADDRESS = 6;       // Adreça accessori
10
11 void setup() {                   // Inicialització
12   pinMode (LED_PIN, OUTPUT );    // LED apagat per defecte
13   digitalWrite(LED_PIN, LOW);
14                                  // Configurem pin i tipus de descodificador
15   Dcc.pin(digitalPinToInterrupt(DCC_PIN), DCC_PIN, 1);
16   Dcc.initAccessoryDecoder( MAN_ID_DIY, 1, FLAGS_OUTPUT_ADDRESS_MODE, 0 );
17 }
18
19 void loop() {                    // Bucle principal
20   Dcc.process();                 // Procesa la descodificacio de la senyal DCC
21 }
22
23 // Quan arriba un paquet DCC per accesoris s'executa aquesta rutina
24 void notifyDccAccTurnoutOutput( uint16_t Addr, uint8_t Direction, uint8_t OutputPower ) {
25     if(( Addr == DCC_ADDRESS ) && OutputPower ) { // Si es la nostra adreça i esta activa
26       if (Direction == 0)                         // Segons sigui recte/desviat
27         digitalWrite(LED_PIN, LOW);               // apaguem LED
28       else                                        // o
29         digitalWrite(LED_PIN, HIGH);              // encenem LED
30     }
31 }
32
```

Este pequeño programa enciende y apaga el LED de la placa Arduino (conectado al pin 13) cuando recibe la orden correspondiente para la activación del accesorio 6 en posición recto / desviado.

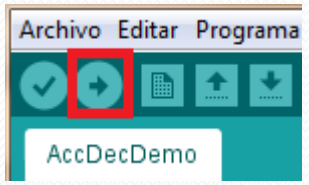
Si queremos que lo haga por otro accesorio sólo hay que cambiar en la línea 9 la dirección 6 por el accesorio que escojamos y volver a subir el programa a Arduino.

CONEXIÓN DE ARDUINO A DCC

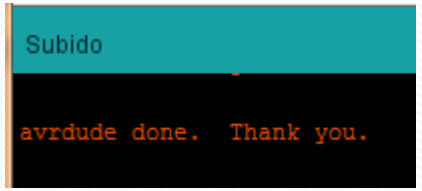


En el Arduino IDE comprobamos que hemos seleccionado la placa Arduino que tenemos y el Puerto al que está conectada.

Para cargar el programa presionaremos en el botón 'Subir' programa.



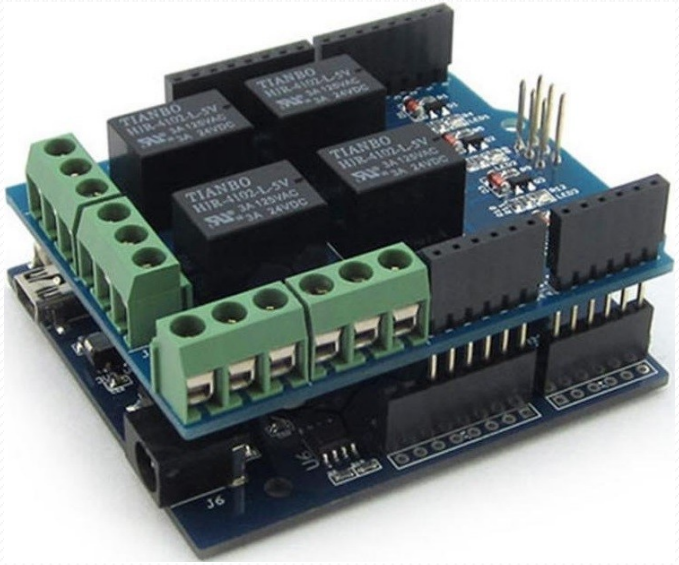
Si todo va bien nos informará que se ha subido con éxito



La librería permite hacer tanto descodificadores de accesorios como descodificadores embarcados. También tiene funciones para poder gestionar las CV.

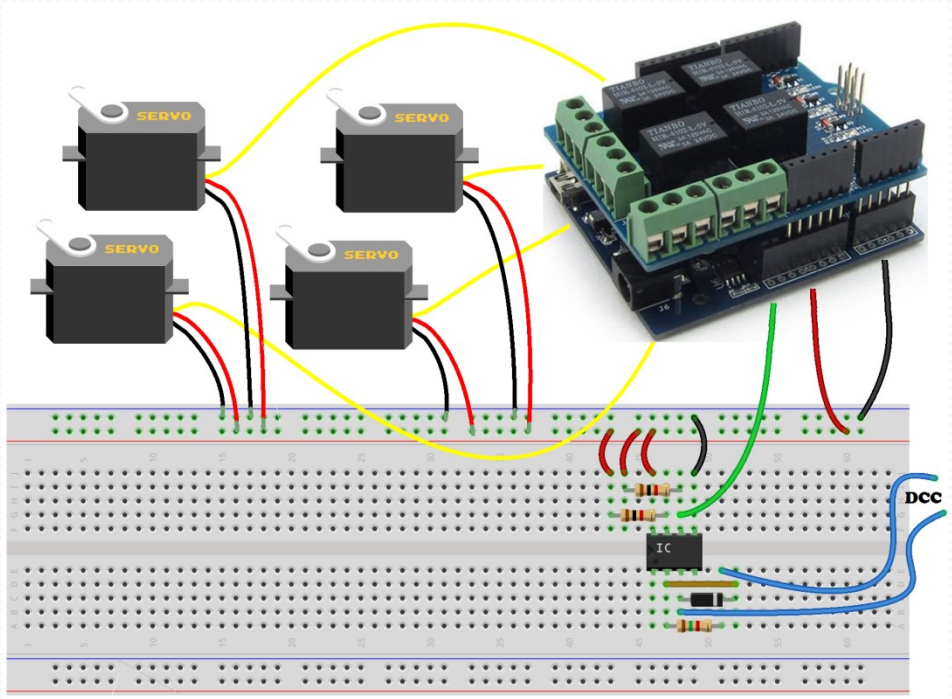
Hay otras librerías para descodificar la señal DCC:
MynaBay: https://github.com/MynaBay/DCC_Decoder

CONEXIÓN DE ARDUINO A DCC



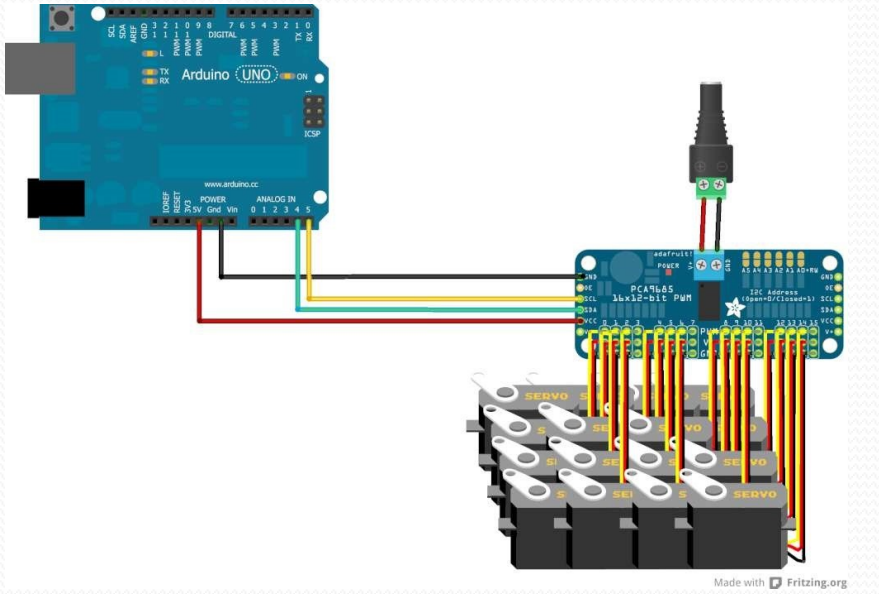
Pinchando una ‘shield’ de relés y ampliando un poco el programa podemos tener un descodificador de accesorios con 4 relés similar a los comerciales y aún podríamos aprovechar alguna de las salidas para, p.ejem. encender unos LEDs.

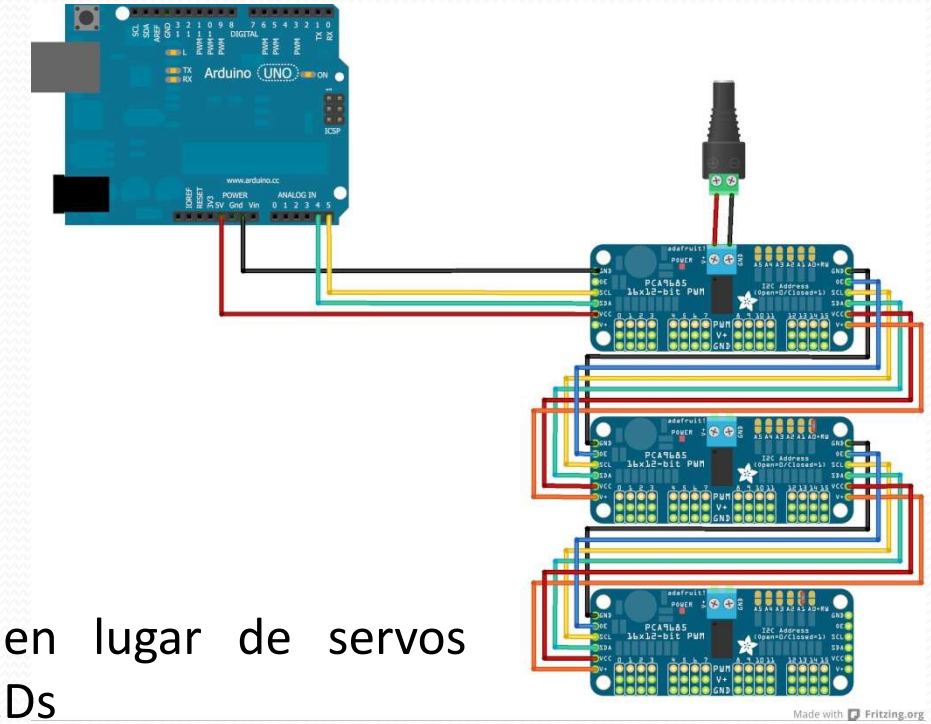
O bien, añadiendo otra de las librerías disponibles que nos permite controlar un servo, y modificando convenientemente el programa podemos tener un descodificador para desvíos accionados por servo con polarización de los corazones



CONEXIÓN DE ARDUINO A DCC

Con la librería de servos solo podríamos controlar hasta 12 servos en el Arduino Uno, pero con placas adicionales y haciendo uso del bus I2C podríamos controlar de 16 a 1000. Necesitaremos instalar las librerías adecuadas para trabajar con el bus I2C y el chip controlador de la placa de servos.

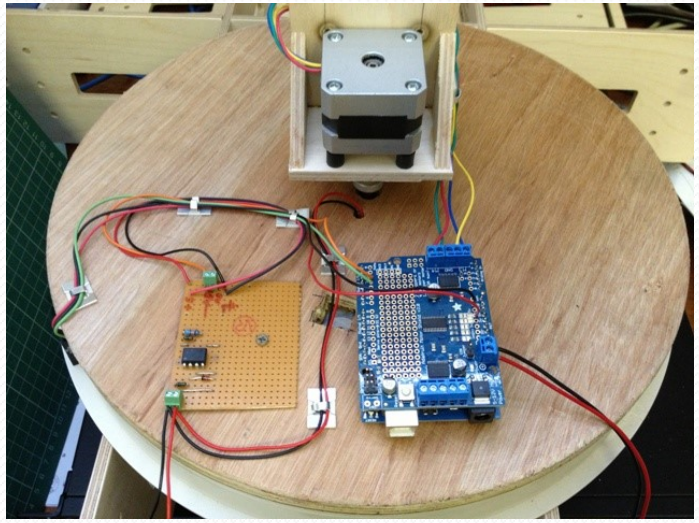




Las salidas son PWM por lo que en lugar de servos podríamos controlar la intensidad de LEDs

CONEXIÓN DE ARDUINO A DCC

Con librería I2C y otra para pantallas OLED, y un poco de trabajo de programación podemos tener carteles anunciadores en las estaciones.

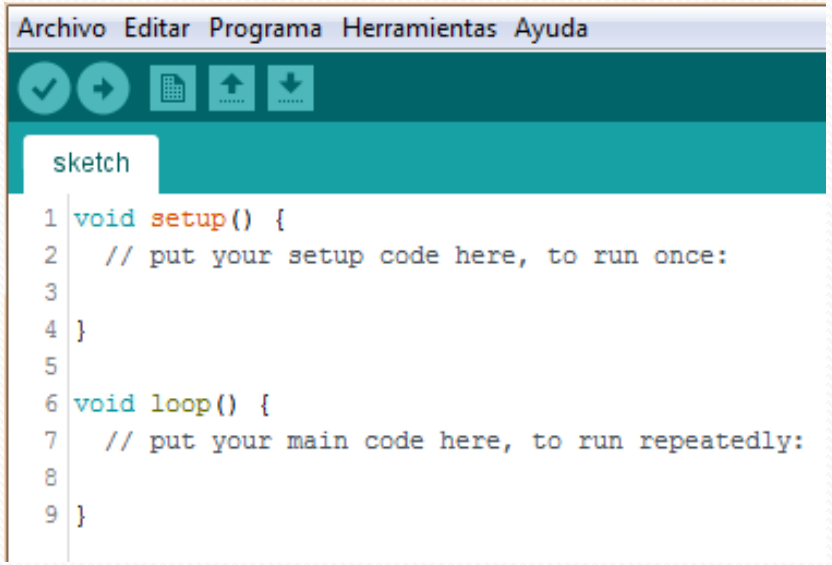


Con una ‘shield’ de control de motores podremos controlar un motor paso a paso para mover una plataforma giratoria.

Hay una librería para este tipo de motores.

APRENDER A PROGRAMAR EL ARDUINO

Se llama ‘sketch’ a un programa hecho para la plataforma Arduino.



Tiene dos rutinas principales:

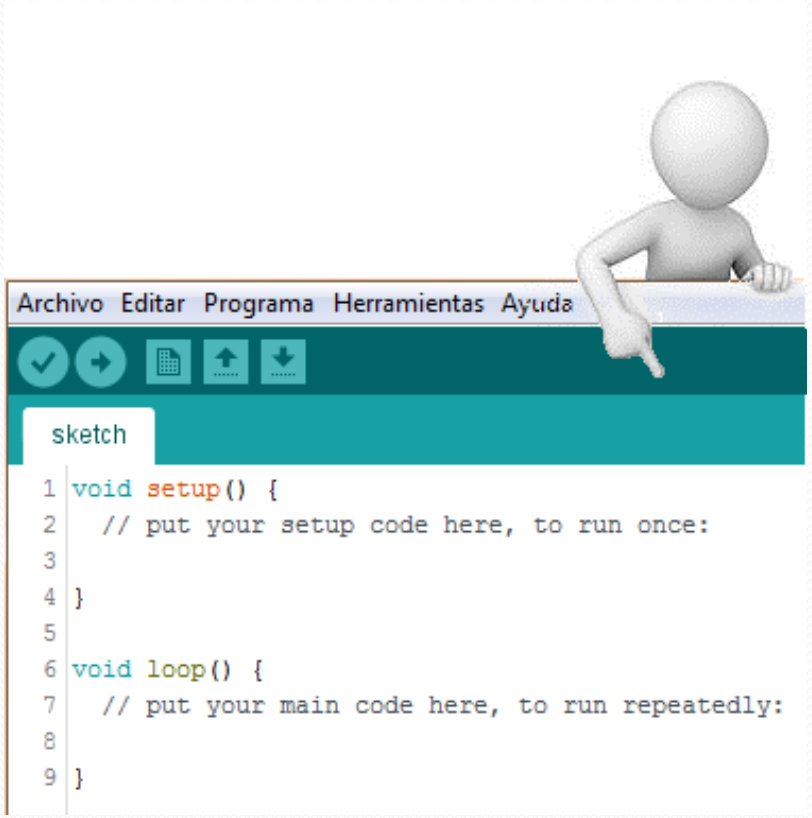
setup() – Contiene el código de configuración, se ejecuta solo una vez al dar tensión o presionar el botón Reset.

loop () – Es el programa principal, se ejecuta una vez y otra sin parar.

Hay una gran cantidad de librerías que podemos hacer servir y que hacen muy fácil interactuar con el hardware, para utilizarlas las tendremos que instalar y se han de incluir en el código haciendo servir la directiva #include <libreria.h>

Tendremos que estudiar la librería o mirar los ejemplos que normalmente llevan para saber qué funciones añade y como las tenemos que utilizar.

APRENDER A PROGRAMAR EL ARDUINO



¡Eh!

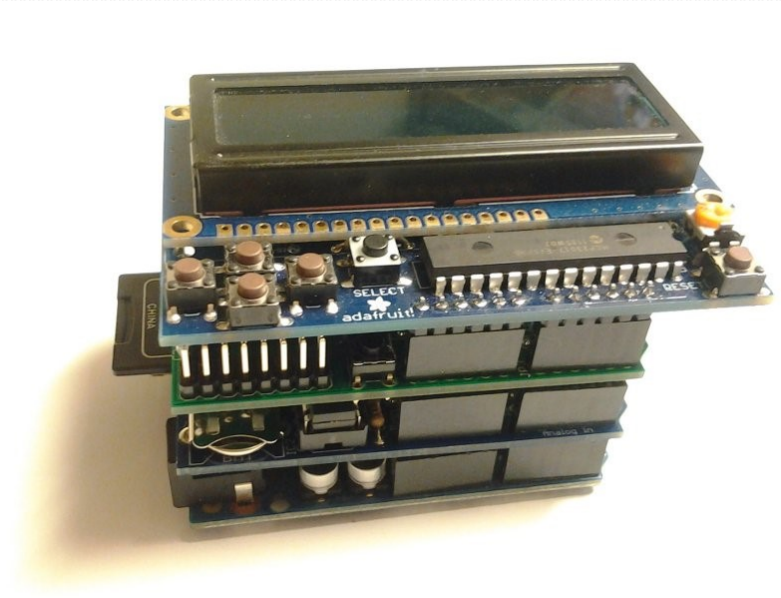
¡Un momento!

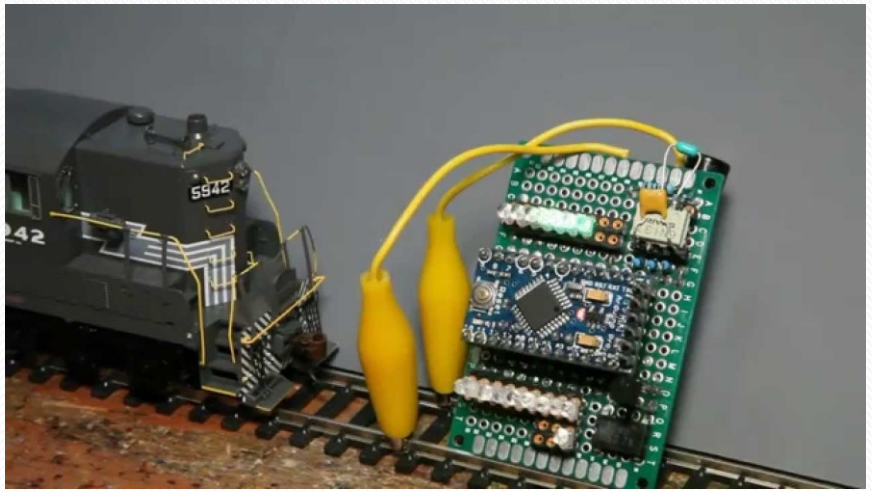
¿He de aprender todo esto para hacer servir el Arduino en mi maqueta?

APRENDER A PROGRAMAR EL ARDUINO

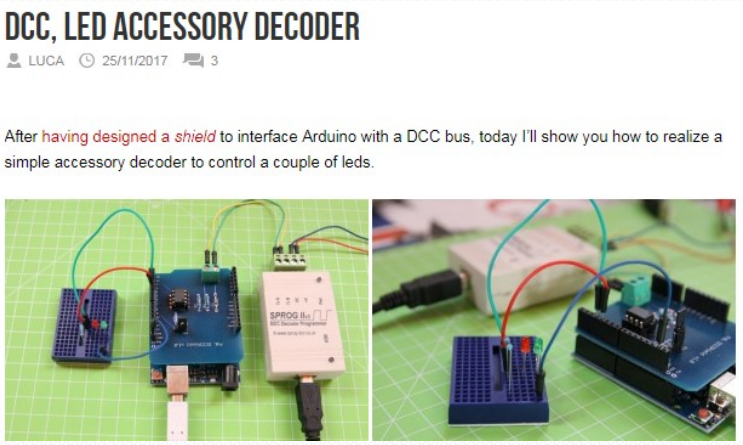
Quiero un diseño especial ya que no he encontrado nada que haga lo que necesito.

Me vale con el diseño que he visto en Internet en que el Trabajo ya está hecho.

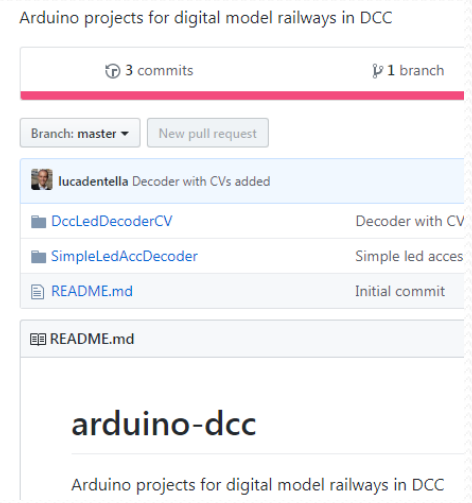




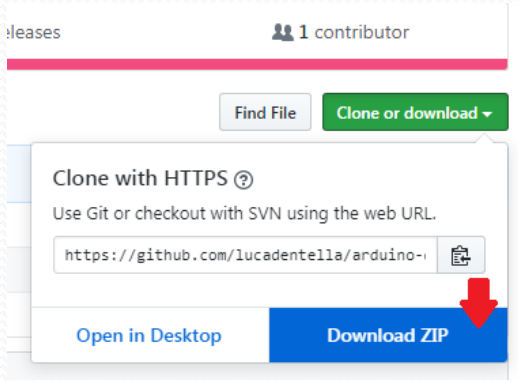
NO APRENDER A PROGRAMAR EL ARDUINO



1 He visto en Internet ☜ un descodificador que me sirve y ya tiene el software listo para descargar ☞

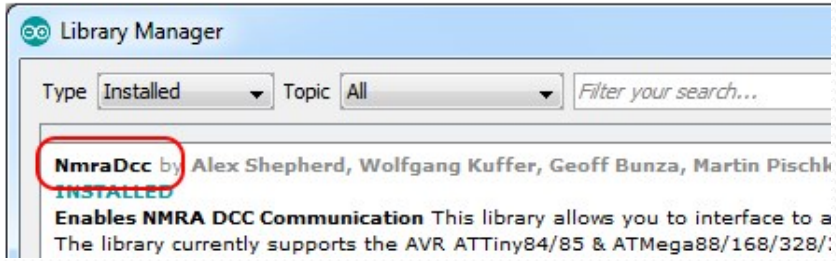


2 Descargamos el software y lo descomprimimos si hace falta. Lo abriremos con el Arduino IDE.



3 Instalamos las librerías que nos indiquen o miramos en el código si nos hace falta alguna librería

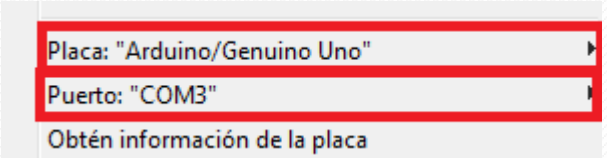
```
#include "NmraDcc.h"
NmraDcc Dcc;
```



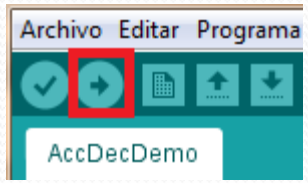
NO APRENDER A PROGRAMAR EL ARDUINO

```
#define BOARD_ADDRESS    5
#define PORT_ADDRESS     1
```

4 Comprobamos si hemos de cambiar algo en el código para que se adapte a mis necesidades

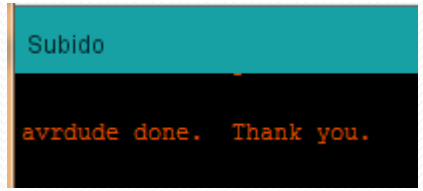


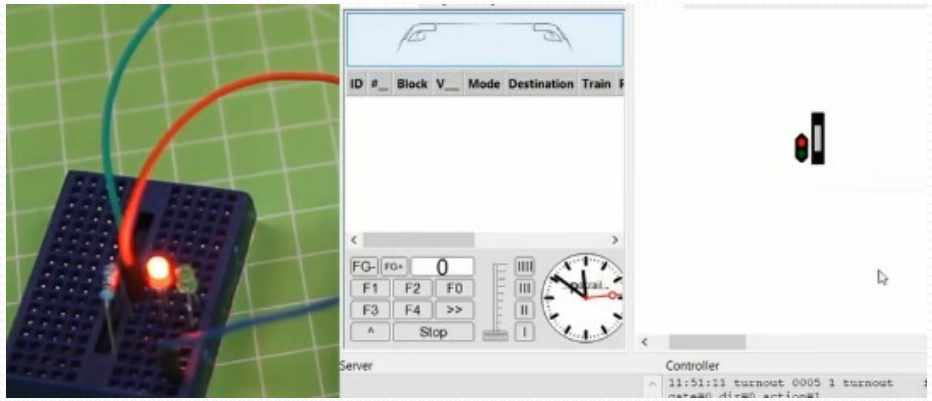
5 Escogemos nuestra placa en el Arduino IDE y el Puerto dónde está conectada



6 Pulsamos sobre el botón de 'Subir' el programa a nuestro Arduino

7 Comprobamos que haya ido bien, sin errores





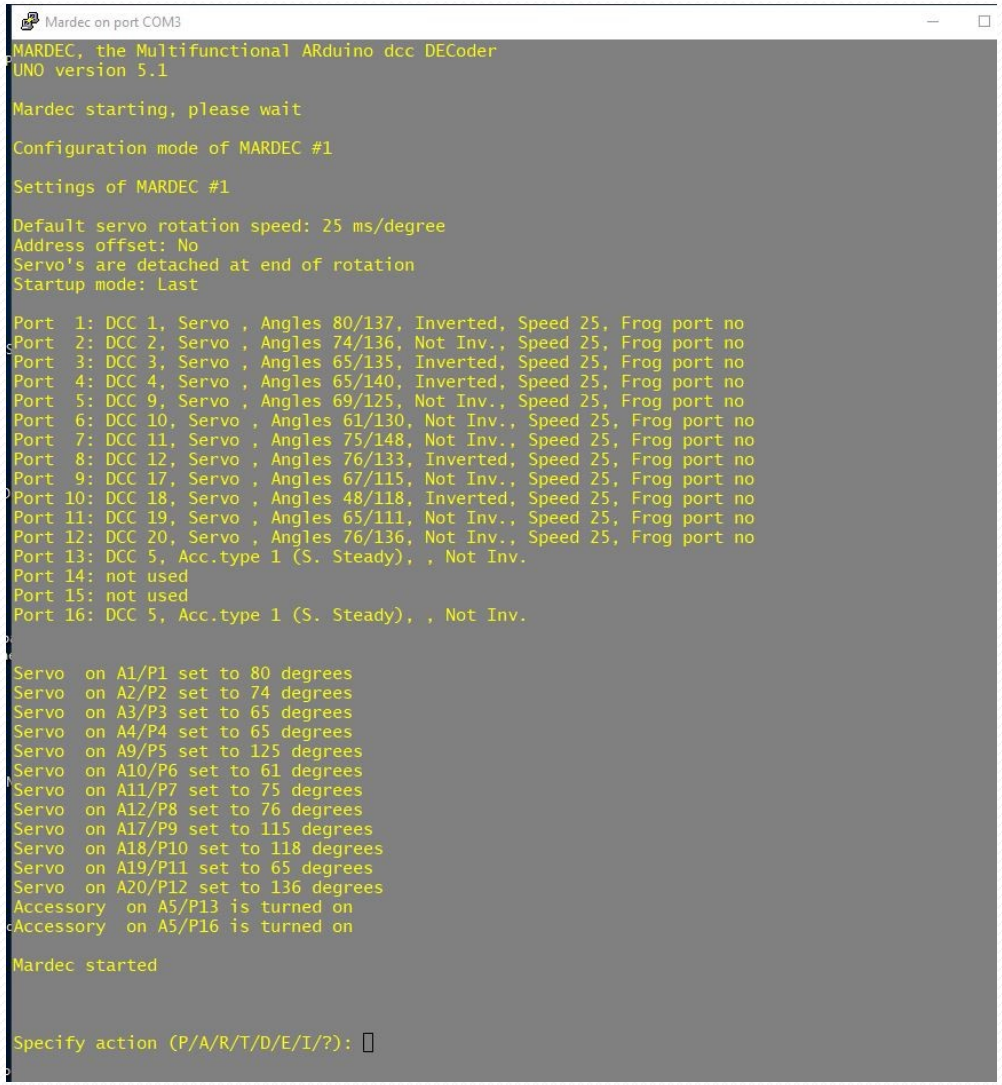
8 Comprobamos que funciona

ARDUINO SIN COMPLICACIONES

En Internet hay Montajes de descodificadores DCC en que el trabajo ya está hecho. Sólo hay que bajarse el código, instalar las librerías que use y subir el programa al Arduino.

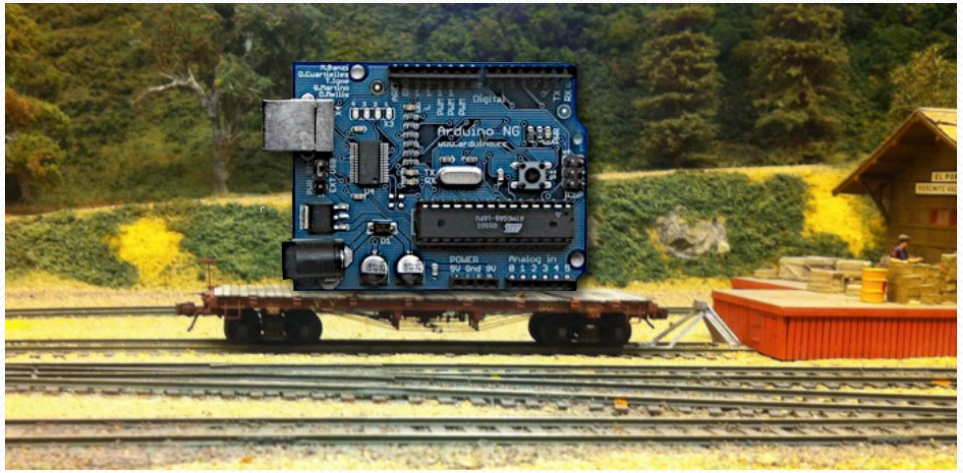
Muchas veces lo único que se ha de modificar del código es la dirección a la que queremos que responda o algún valor que dependa de lo que hemos montado si hay diferentes opciones.

Si hacen servir CV es como un descodificador comercial normal pero hay que no hace falta, se le puede decir desde el monitor serie:

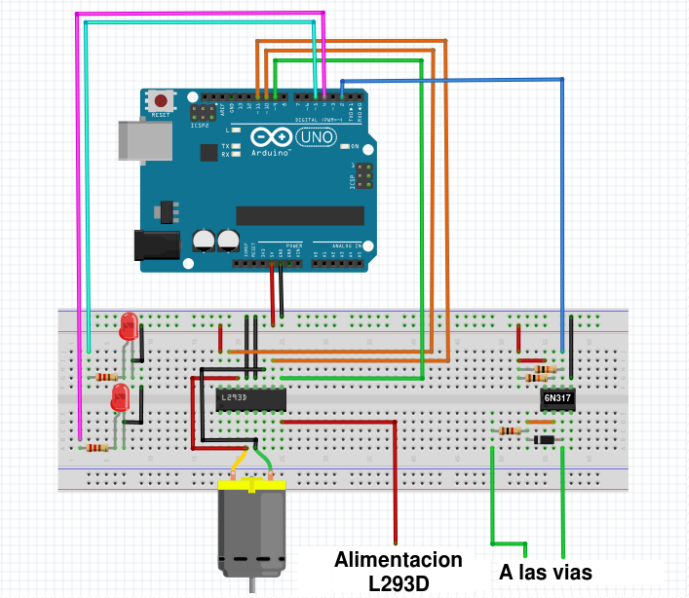


DESCODIFICADORES PARA LOCOMOTORAS

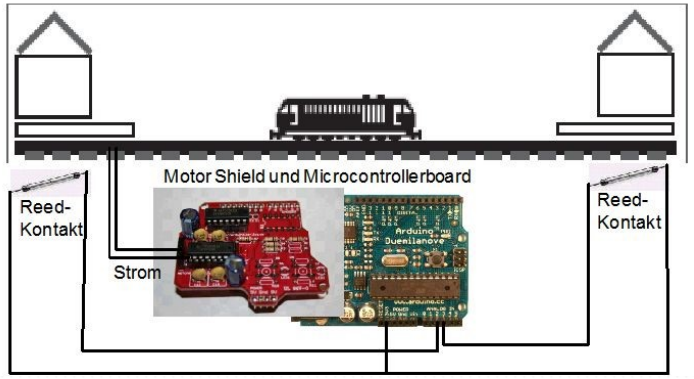
También hay algún diseño de descodificadores embarcados para locomotoras, pero el tamaño es grande para las escalas pequeñas.



Están basados en la utilización del circuito de control de motores.



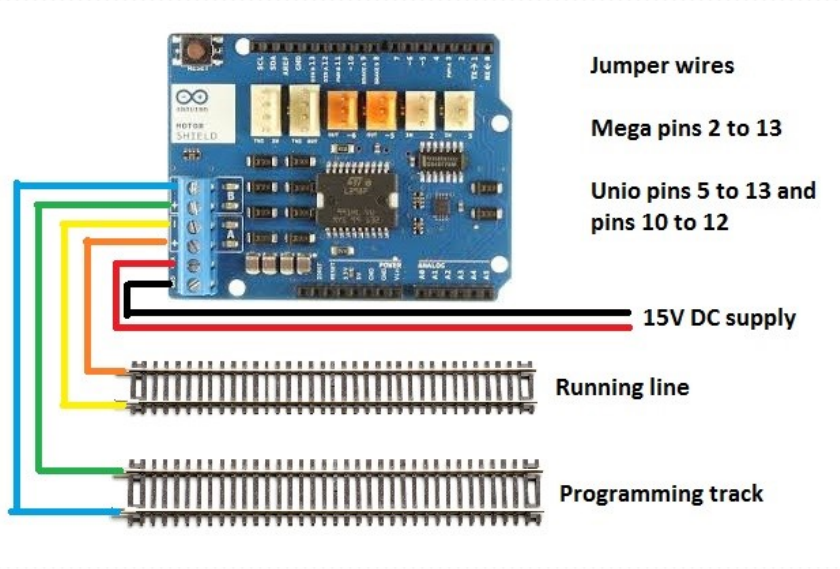
Lo más habitual es encontrar circuitos de control para ida y vuelta de locomotoras.

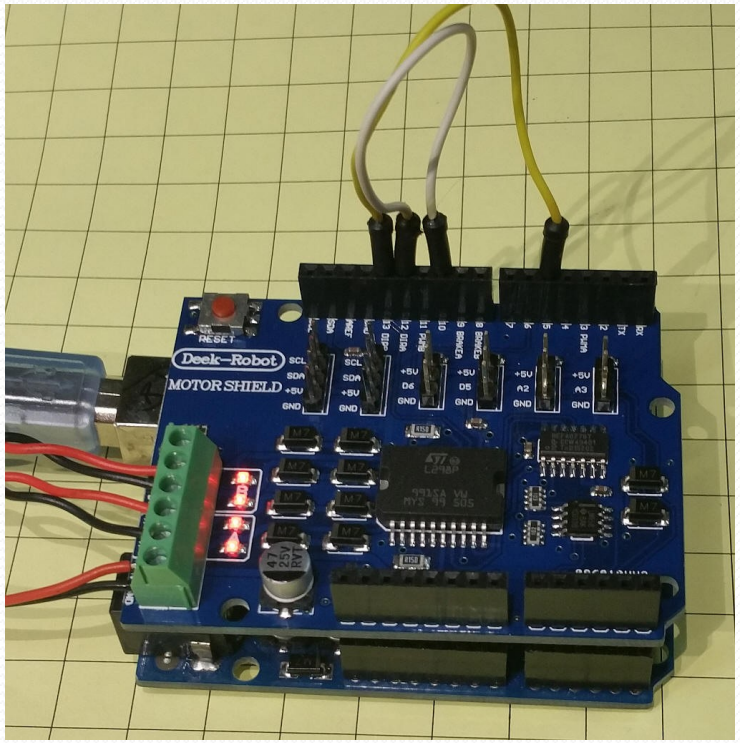


CENTRAL DCC CON ARDUINO

DCC++ es el diseño de una central DCC hecha con Arduino y una ‘shield’ de control de motores.

Se controla desde el USB con el monitor serie mediante comandos de texto.



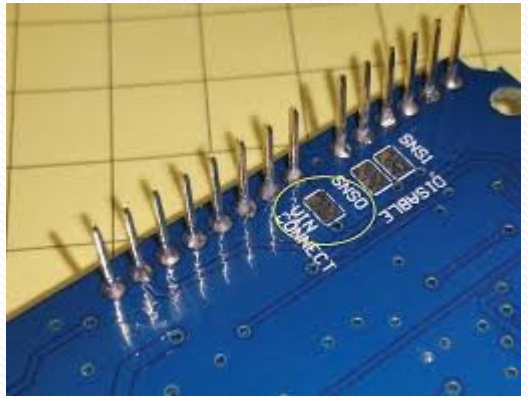


Tiene salidas para vía principal y vía de programación para leer y escribir CV.

También se puede controlar desde JMRI y Rocrail.

CENTRAL DCC CON ARDUINO

Recordamos que el Arduino se alimenta entre 7 y 12V, así que hemos de aislar la conexión VIN de la 'shield' para que la tensión a la que alimentamos la DCC++ que es superior no estropee el Arduino, éste se alimentará desde el USB.



En Internet, hay montajes de la DCC++ con 'shield' Ethernet o WiFi y con una pantalla LCD para hacerla autónoma y no depender del ordenador.

También se han diseñados mandos por cable o inalámbricos para la conexión serie.

INTERNET

- Librerías Arduino
  https://www.arduino.cc/en/Reference/Libraries
- Adafruit PCA9685 16-Channel Servo Driver
  https://learn.adafruit.com/16-channel-pwm-servo-driver?view=all
- Arcomora
  https://www.arcomora.com/
- DCC++
  https://sites.google.com/site/dccppsite/home
- La maqueta de Infotronikblog
  http://lamaquetade.infotronikblog.com/
- Club N Caldes
  http://www.clubncaldes.com/search/label/Arduino
- Locoduino
  http://www.locoduino.org/

GRACIAS POR SU ATENCIÓN

Ponente: Paco Cañada

The POWS

www.fut.es/~fmco