CURSOS INTERSEMESTRALES

PROTECO

ARDUINO

INTERMEDIO

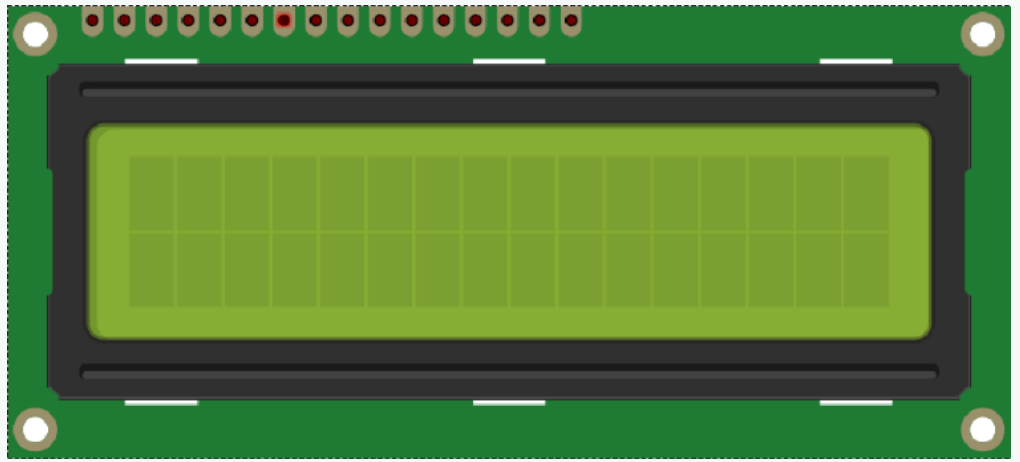
TEMA:

Pantalla de cristal liquido (LCD)

PROTECO

Introducción

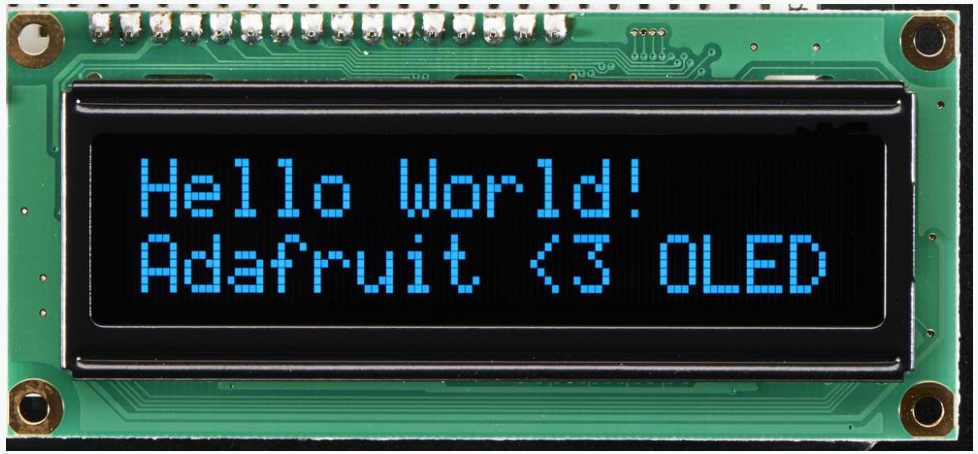
Las pantallas LCD son dispositivos diseñados para mostrar información en forma gráfica. LCD significa Liquid Crystal Display (Display de cristal líquido). La mayoría de las pantallas LCD vienen unidas a una placa de circuito y poseen pines de entrada/salida de datos.



PROTECO

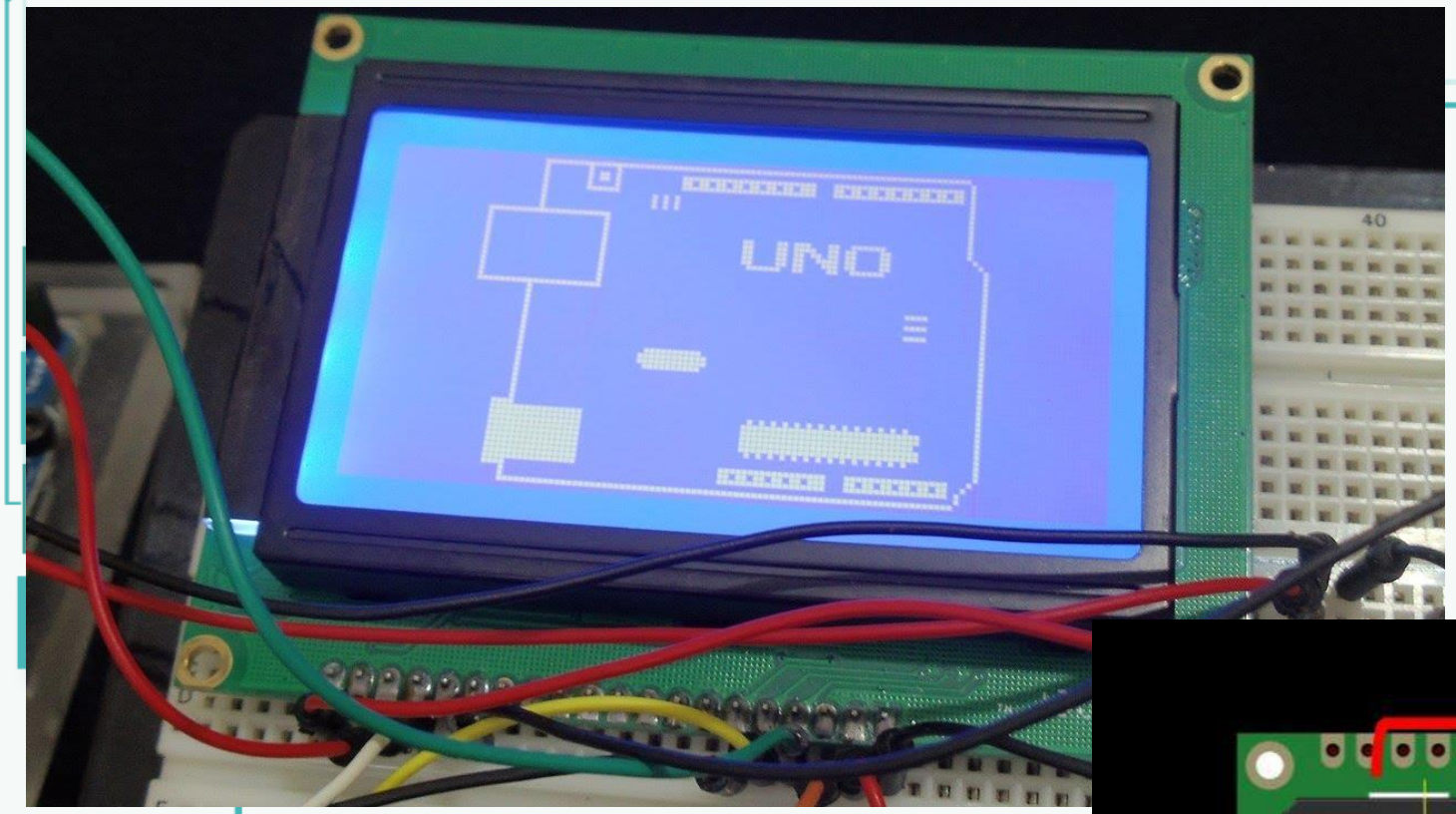
Características

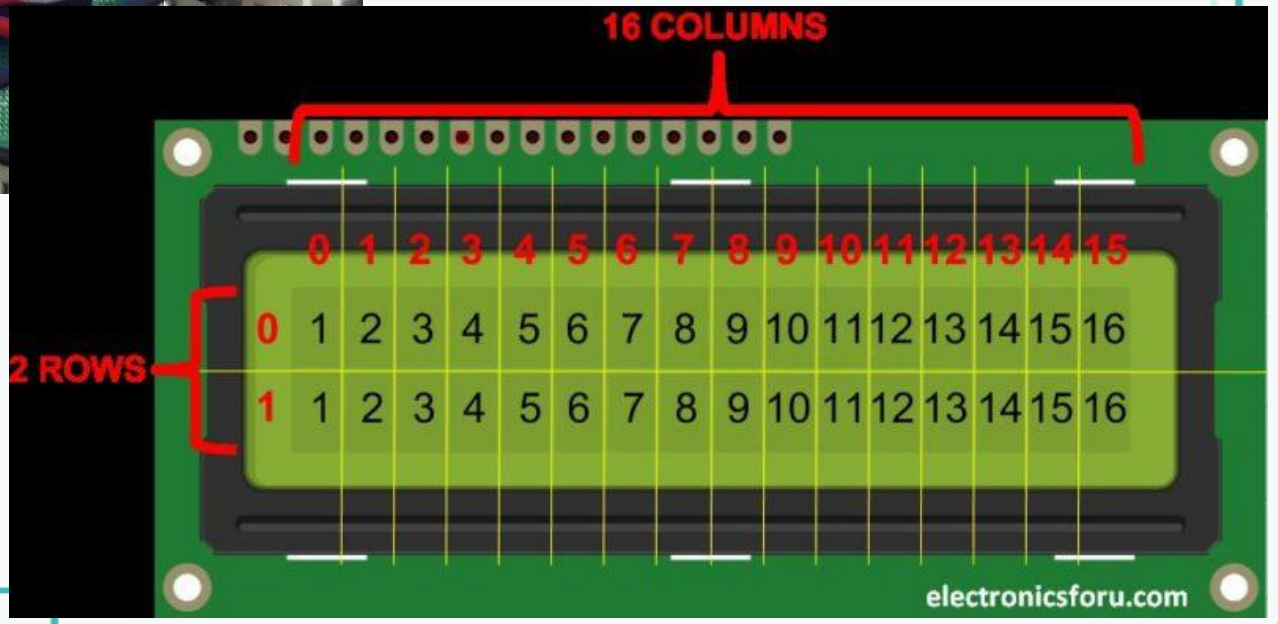
- Tamaño: en pulgadas o por numero de líneas
- Resolución: cantidad de pixeles
- Brillo: cantidad de iluminación
- Contraste: diferencia entre obscuro y brillante
- Angulo de visión: desde donde puede verse
- Número de caracteres: cantidad desplegable



PROTECO

¿Cómo identifico mi lcd?





PROTECO

LiquidCrystal

Para poder enviar datos a nuestro display es necesario ocupar la biblioteca LiquidCrystal, para importarla desde el menú se elige:

sketch → import library → liquidCrystal.

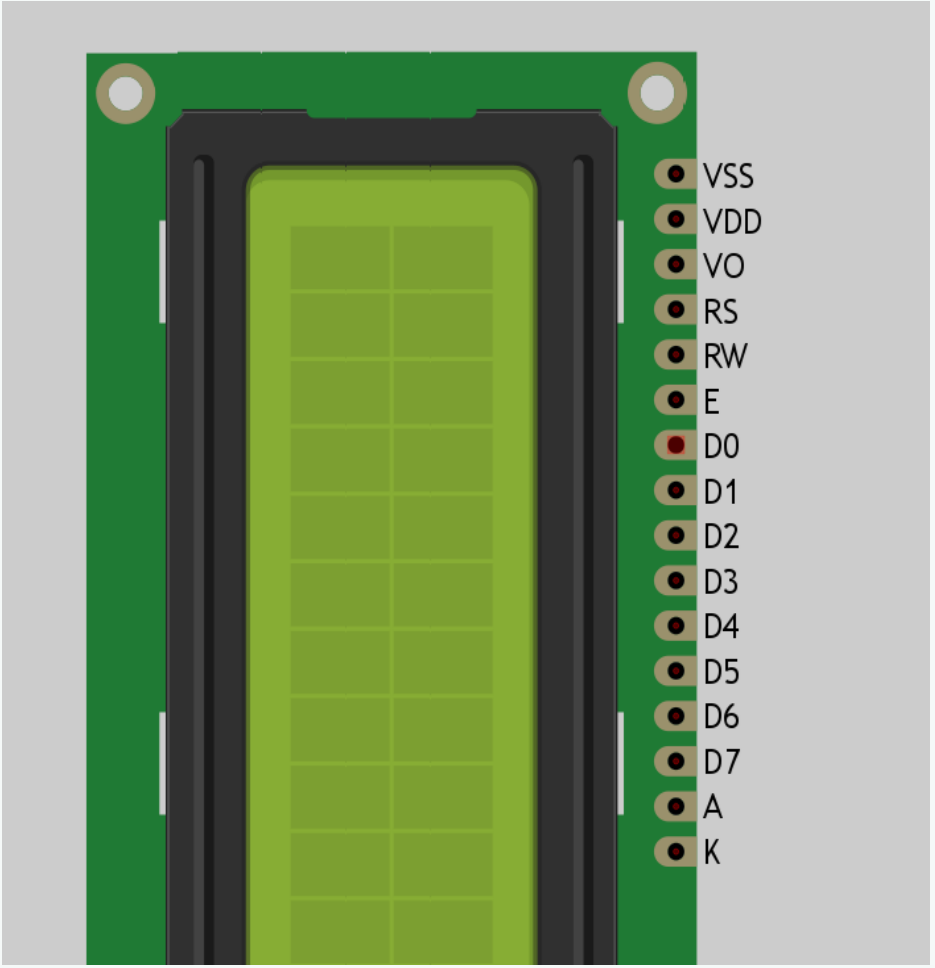
#include <LiquidCrystal.h>

Luego debemos declarar, como si de variables fuera, nuestra pantalla LCD y debemos asignarle un nombre. Además debemos especificar los pines que vamos a usar para comunicar Arduino con la LCD.

PROTECO

Diagrama de conexiones

Pin LCD		
1	VSS	0 V
2	VDD	5 V
3	V0	Potenciómetro (de 0 V a 5 V) para controlar el contraste de la pantalla, o una resistencia de 2,2K a 0 V.
4	RS	Register select
5	R/W	0 V (solo vamos a escribir, no a leer datos)
6	E	enable
7	DB0	(sólo para transmitir datos a 8 bits, cosa que no será necesaria nunca)
8	DB1	
9	DB2	
10	DB3	
11	DB4	BUS de transmisión de datos a 4 bits
12	DB5	
13	DB6	
14	DB7	
15	LED+	5 V con una resistencia de 220 Ω (para LCD con retroiluminación)
16	LED-	0 V (para LCD con retroiluminación)

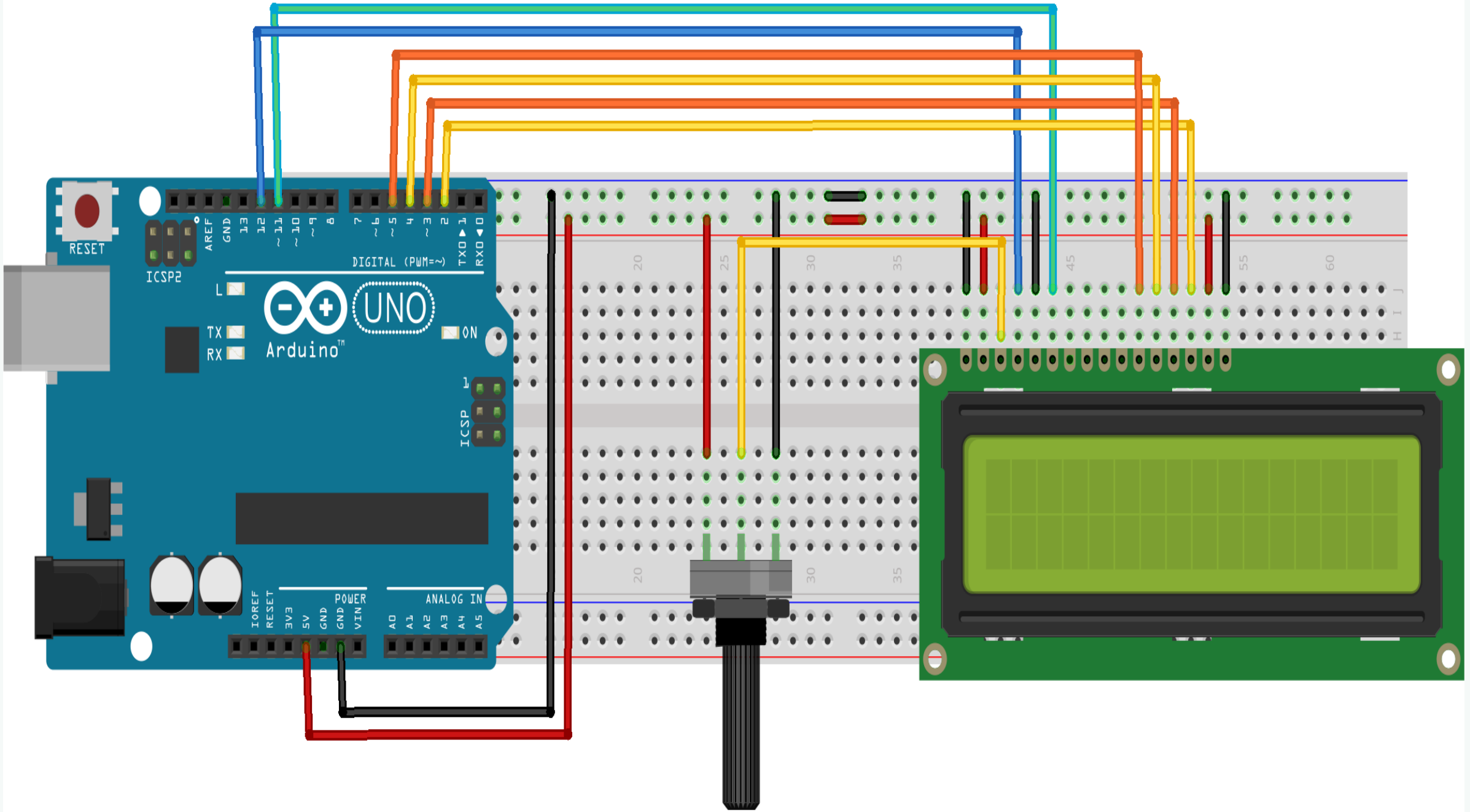


PROTECO

Descripción detallada de cada pin

- VSS que es el pin de negativo, tierra, 0 volts o GND.
- VDD es la alimentación principal de la pantalla y el chip, lleva 5 volts (recomendable ponerle en serie una resistencia para evitar daños, con una de 220 ohm es suficiente).
- VO es el contraste de la pantalla, debe conectarse con un potenciómetro de unos 10k ohm o una resistencia fija una vez que encontremos el valor deseado de contraste. Tengan en cuenta que si no conectan esto, no verán nada.
- RS es el selector de registro (el microcontrolador le comunica a la LCD si quiere mostrar caracteres o si lo que quiere es enviar comandos de control, como cambiar posición del cursor o borrar la pantalla, por ejemplo).
- RW es el pin que comanda la lectura/escritura. En nuestro caso siempre estará en 0 (conectado a GND) para que escriba en todo momento.
- E es enable, habilita la pantalla para recibir información.
- D0~D3 no los vamos a utilizar. Como pueden ver la pantalla tiene un bus de datos de 8 bits, de D0 a D7. Nosotros solamente utilizaremos 4 bits, de D4 a D7, que nos servirán para establecer las líneas de comunicación por donde se transfieren los datos.
- A y K son los pines del led de la luz de fondo de la pantalla. A se conectará a 4 o 5 volts y K a gnd.

PROTECO


RESET
ICSP2
AREF
GND
13
12
11
10
9
8
7
6
5
4
3
2
1
0
DIGITAL (PWM=~)
TXD
RXD
L
TX
RX
UNO
Arduino™
ON
ICSP
POWER
ANALOG IN
IOREF
RESET
3V3
5V
GND
GND
VIN
A0
A1
A2
A3
A4
A5

¿Qué podemos usar de LiquidCrystal?

Declaración de nuestra LCD

```
LiquidCrystal nombreLcd(rs,e,db4,db5,db6,db7);
```

Dentro del void setup() debemos iniciar la librería para nuestra LCD, utilizando la
función begin() y especificando el tamaño, que en nuestro caso es de 16x2
caracteres:

```
nombreLcd.begin(16,2);
```

PROTECO

* Para limpiar la pantalla se utiliza el comando clear():

```
nombrelcd.clear();
```

* Para posicionar el cursos del LCD:

```
nombrelcd.setCursor(columna,fila);
```

* Para escribir un texto:

```
nombrelcd.print('texto');
```

* Mostrar el cursor:

```
lcdnombre.cursor();
```

* Apagar y prender la pantalla mostrando el texto que ya estaba:

```
nombrelcd.nodisplay();
nombrelcd.display();
```

PROTECO

Probando el LCD

```
#include ;
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);


void setup()
{
        lcd.begin(16, 2);
        lcd.print("Hola mundo!");
}
void loop() {
}
```

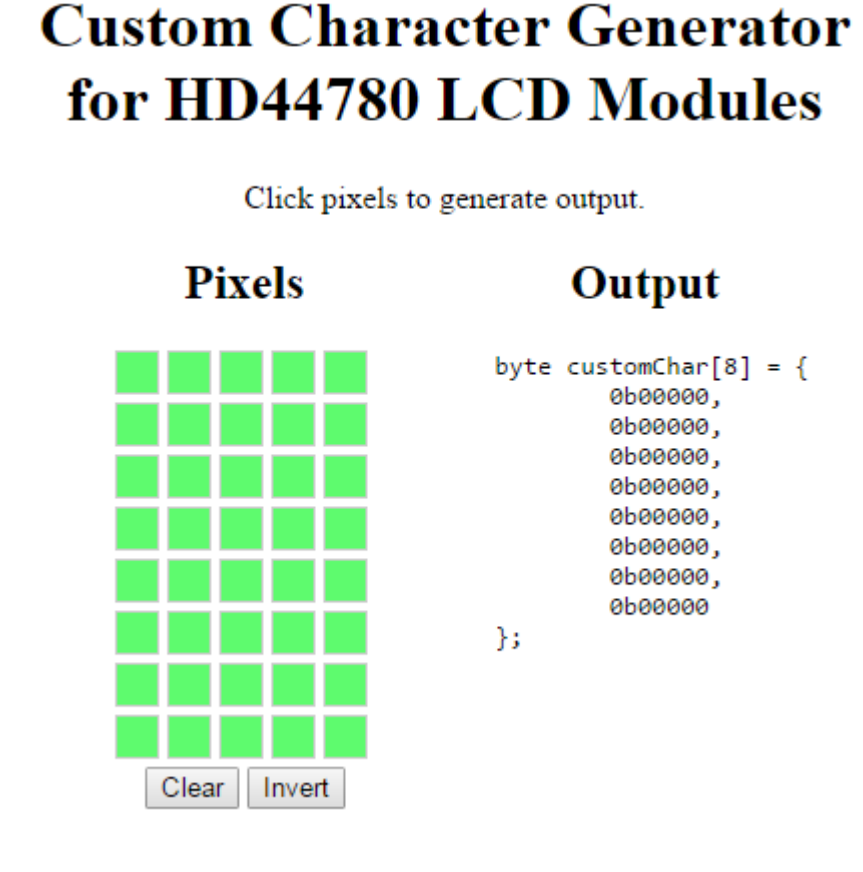
PROTECO

Caracteres preestablecidos

PROTECO

Caracteres propios

https://omerk.github.io/lcdchargen/



PROTECO