

# Curso avanzado sobre Arduino: LCDs

[ElCacharreo.com](http://ElCacharreo.com)



[ElCacharreo.com](http://ElCacharreo.com)



# Arduino avanzado: Presente



# Arduino avanzado: Presente

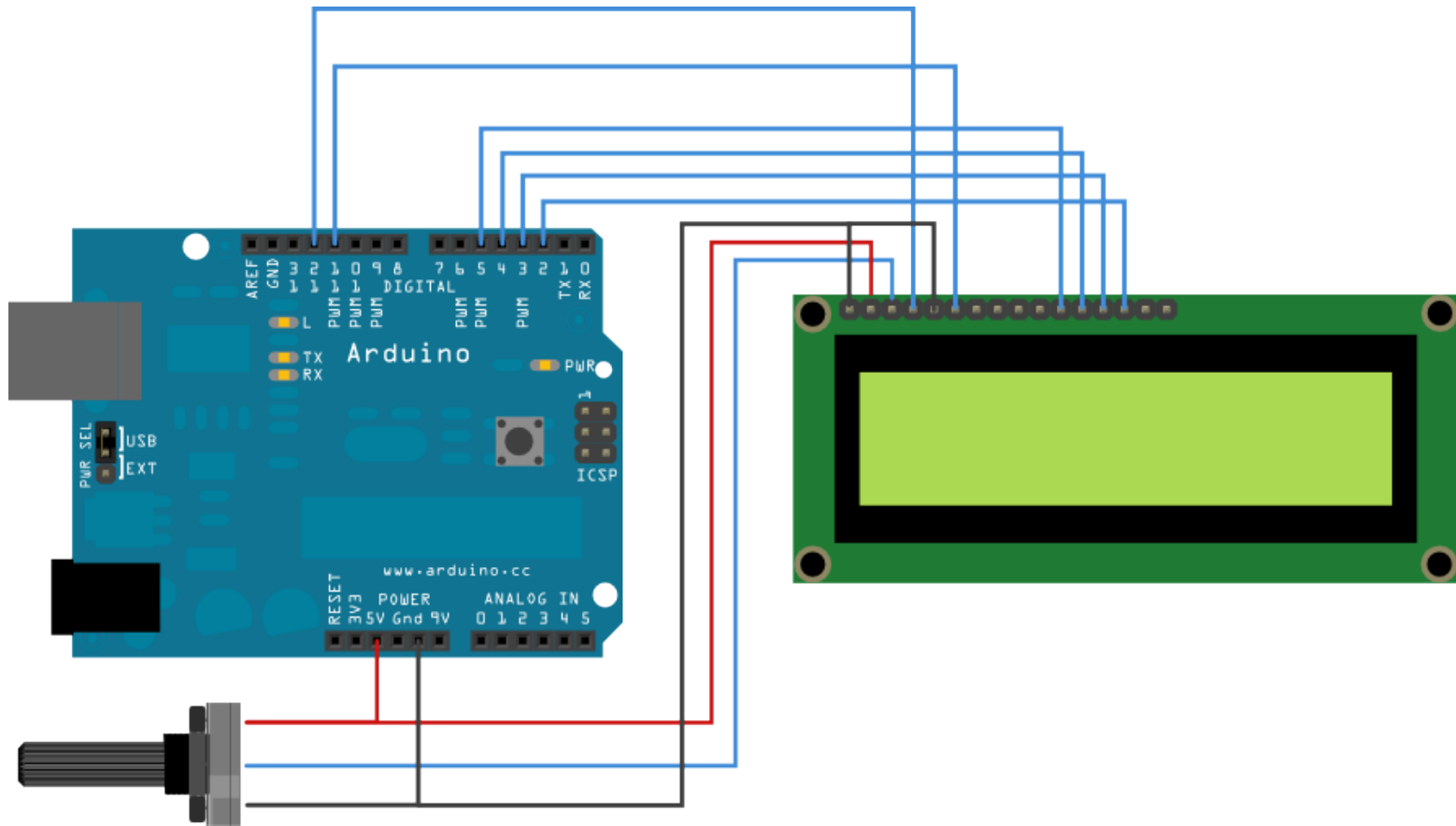


José Antonio Vacas Martínez

blog  
javacasm@elcacharreo.com  
twitter  
linkedin



# Arduino avanzado: LCD



<http://arduino.cc/en/Tutorial/LiquidCrystal>



# Arduino avanzado: LCD

A **register select (RS) pin** that controls where in the LCD's memory you're writing data to. You can select either the data register, which holds what goes on the screen, or an instruction register, which is where the LCD's controller looks for instructions on what to do next.

A **Read/Write (R/W) pin** that selects reading mode or writing mode

An **Enable pin** that enables writing to the registers

**8 data pins (D0 -D7)**. The states of these pins (high or low) are the bits that you're writing to a register when you write, or the values you're reading when you read.

There's also a **display constrast pin (Vo)**, **power supply pins (+5V and Gnd)** and **LED Backlight (Bklt+ andBKlt-)**



# Arduino avanzado: LCD

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

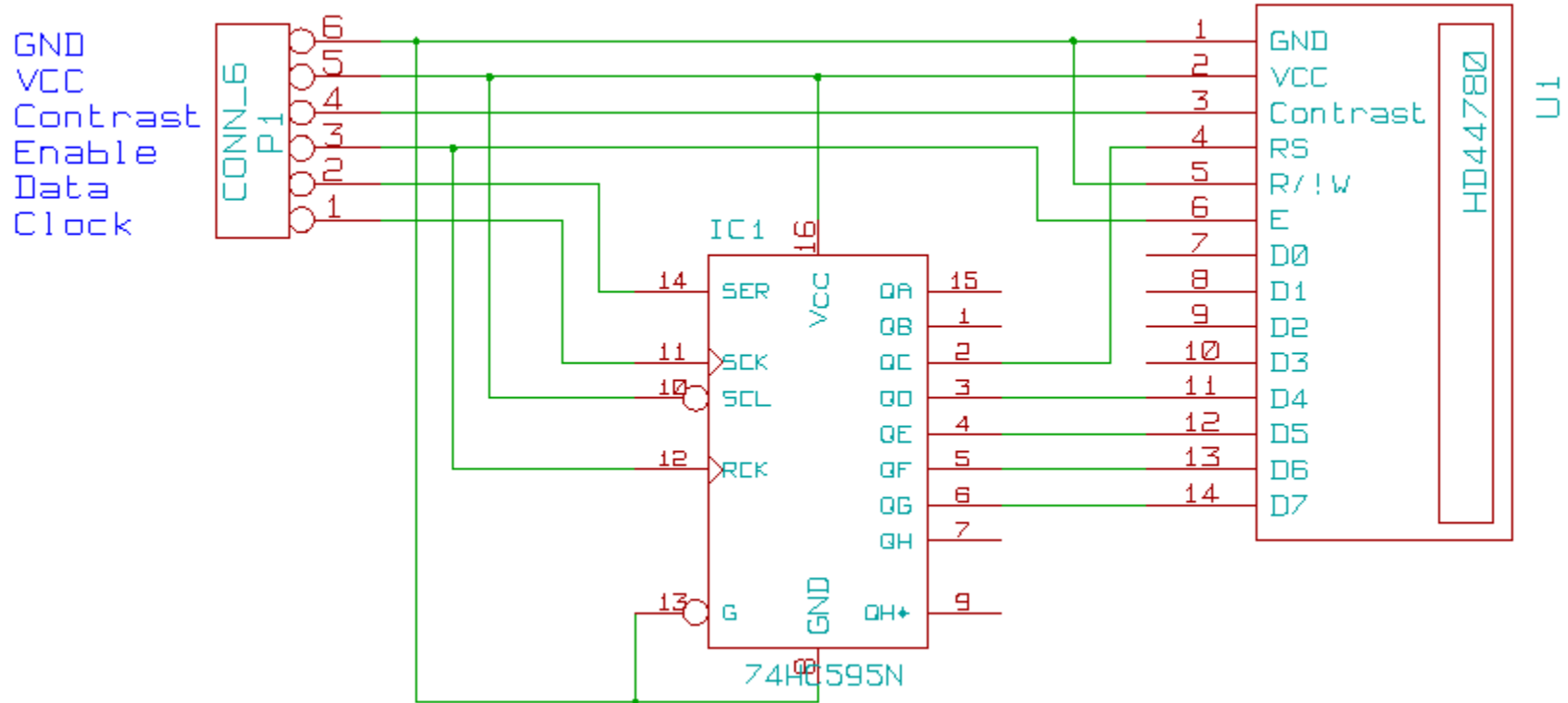
void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("hello, world!");
}

void loop() {
  // set the cursor to column 0, line 1
  // (note: line 1 is the second row, since counting begins with 0):
  lcd.setCursor(0, 1);
  // print the number of seconds since reset:
  lcd.print(millis()/1000);
}
```

<http://arduino.cc/en/Tutorial/LiquidCrystal>



# Arduino avanzado: LCD



<http://arduino.cc/en/Tutorial/LiquidCrystal>



# Arduino avanzado: LCD595

```
#include <LiquidCrystal595.h>      // include the library
LiquidCrystal595 lcd(7,8,9); // datapin, latchpin, clockpin

void setup() {
    lcd.begin(16,2);      // 16 characters, 2 rows

    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Wow. 3 pins!");
    lcd.setCursor(0,1);
    lcd.print("Fabulous");
}

void loop() {
    // not used.
}
```

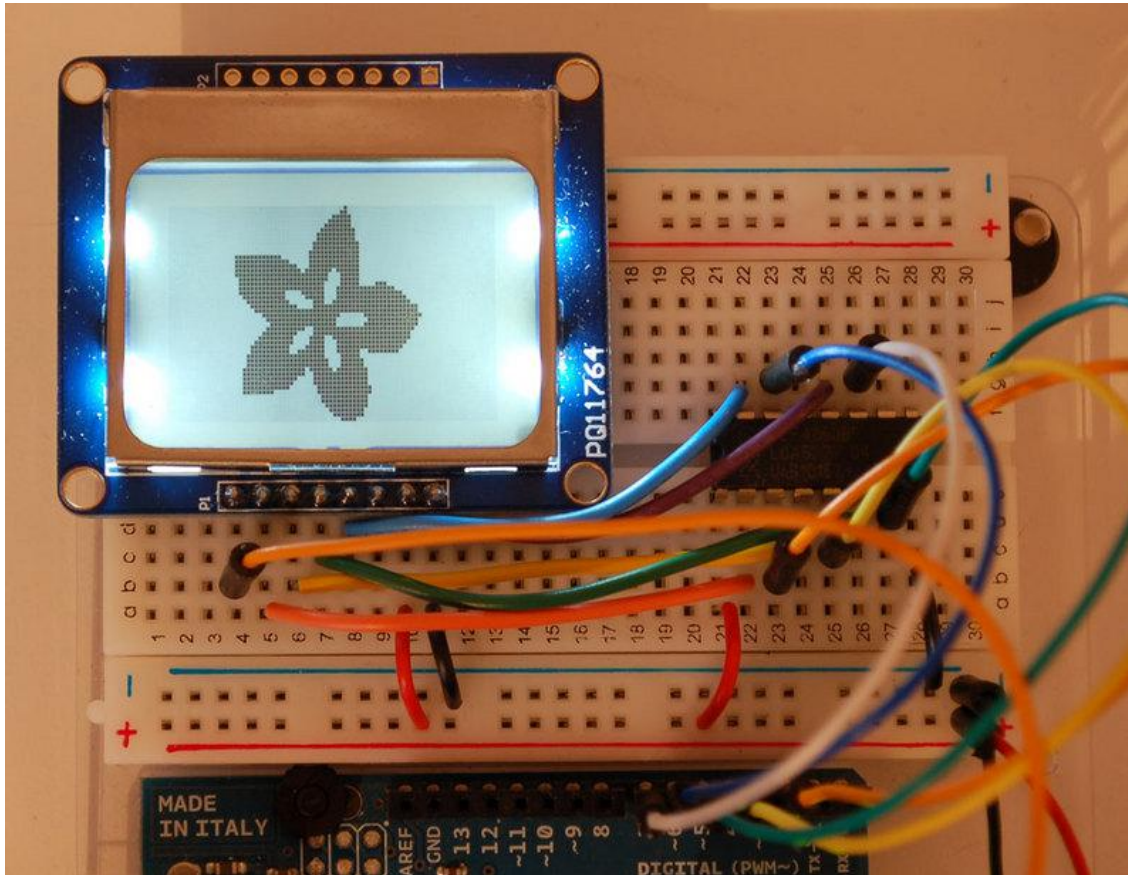


<http://rowansimms.com/article.php/lcd-hookup-in-seconds>

<https://code.google.com/p/arduino-lcd-3pin/downloads/list>



# Arduino avanzado: GLCD 5110



<http://learn.adafruit.com/nokia-5110-3310-monochrome-lcd>



ElCacharreo.com

Arduino  
Avanzado

# Termómetro con GLCD 5110

```
#include <Adafruit_GFX.h>
#include <Adafruit_PCD8544.h>

// pin 3 - Serial clock out (SCLK)
// pin 4 - Serial data out (DIN)
// pin 5 - Data/Command select (D/C)
// pin 7 - LCD chip select (CS)
// pin 6 - LCD reset (RST)
Adafruit_PCD8544 display = Adafruit_PCD8544(3, 4, 5, 7, 6);
//Control temperatura
int analogPin = 0; // Pin analogico 0 de Arduino
float temperature = 0.0;
int grados, decimas, gradR, deciR;
int lm35read, tmedia;
int val,cnt;
void setup() {
  // Iniciamos comunicacion serie
  Serial.begin(9600);
  // Iniciamos comunicacion con la pantalla
  display.begin();
  // Establecemos el contraste
  display.setContrast(45);
  // Mostramos la pantalla de inicio durante 2 segundos
  display.display();
  delay(2000);
  // borramos la pantalla
  display.clearDisplay(); }
```

<http://www.ardumania.es/termometro-con-lcd-grafico-y-lm35d/>



# Termómetro con GLCD 5110

```
void loop()
{
    tmedia=0;
    for ( cnt=0; cnt<10; cnt++) {
        lm35read = analogRead(analogPin);
        tmedia=tmedia+lm35read;    }

    tmedia=tmedia/cnt;
    temperature = (5.0 * float(tmedia) * 100.0)/1024.0;
    val = map(tmedia,0,1023,0,4990);

    grados=val/10;
    decimas=val%10;
    if (decimas< 3) {          gradR=grados;          deciR=0;          }
    if ((decimas > 2)&(decimas<7)) {          gradR=grados;          deciR=5;          }
    else {          gradR=grados+1;          deciR=0;          }
    // Sacamos los datos por el puerto serie
    Serial.print("Lectura en bruto ");    Serial.print(lm35read);
    Serial.print(" temperatura sin redondear ");    Serial.print(temperature);    Serial.print("C ");
    Serial.print("temperatura redondeada ");    Serial.print(grados);    Serial.print(".");    Serial.println(deciR);
    // Y por ultimo lo mandamos a la pantalla LCD
    display.setTextSize(1);
    display.println(" Temperatura");    display.print(" ");
    display.setTextSize(2);
    display.print(gradR);    display.print(",");    display.print(deciR);    display.print(".C");    display.print(" ");
    display.display();
    delay(500);
    display.clearDisplay();
}
```

<http://www.ardumania.es/termometro-con-lcd-grafico-y-lm35d/>

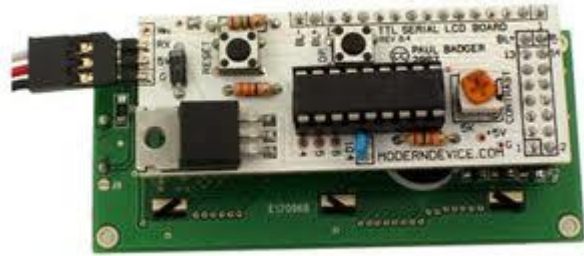


# LCD: adaptador serie LCD117

Incluye un pic 16f84 que procesa comandos enviados por serie y que permite un control avanzado del lcd:

- Gráficos de barras
- Letras grandes,
- Facilidad de uso de diferentes geometrías

[http://cdn.shopify.com/s/files/1/0038/9582/files/LCD117\\_Board\\_Command\\_Summary.pdf?1260768875](http://cdn.shopify.com/s/files/1/0038/9582/files/LCD117_Board_Command_Summary.pdf?1260768875)



# LCD: Proyectos

arduino 5110 lcd shield 3dCube rotating

<http://knolleary.net/2009/01/10/nokia-lcd-shield-for-arduino/>

Modelo para termómetro

<http://www.ardumania.es/termometro-con-lcd-grafico-y-lm35d/>

Gráficos

<http://electronics.stackexchange.com/questions/48634/how-to-plot-a-graph-on-nokia-5110-lcd>

Conexionado

<http://helpfactory.wordpress.com/2012/01/01/arduino-connecting-a-nokia-5110-display-to-arduino-part1-wiring/>

Usando 5V

<http://www.avdweb.nl/arduino/hardware-interfacing/nokia-5110-lcd.html#h0-1-1-1-operation-at-5v>

Osciloscopio

[http://filear.com/?p=211&utm\\_source=rss&utm\\_medium=rss&utm\\_campaign=diy-arduino-oscilloscope-with-the-nokia-3310-glcd-screen](http://filear.com/?p=211&utm_source=rss&utm_medium=rss&utm_campaign=diy-arduino-oscilloscope-with-the-nokia-3310-glcd-screen)



# Conclusiones

Gracias por vuestra atención

