

Cisco Calls This Opportunity the Internet of Everything (IoE)

Networked Connection of People, Process, Data, Things



ARISE | CISO CHANGING SKILLS

© 2013 Cisco and/or its affiliates. All rights reserved. Cisco Confidential

Práctica #15

Conexión de un LCD

Introducción al Internet del Todo

Aplicación de las Telecomunicaciones

Arduino & Cisco



Mostrar en un LCD el mensaje "Hola Mundo". Enviado desde Arduino y conocer algunas funciones de la librería display liquid crystal

Práctica #14 : Conexión de un LCD

Descripción:

Mostrar en un LCD el mensaje “Hola Mundo”. Enviado desde Arduino y conocer algunas funciones de la librería display liquid crystal.

Objetivos:

- Conocer comandos como:
 - `lcd.begin()`
 - `lcd.print()`
 - `lcd.setCursor()`
- Auxiliarse de un elemento de salida exterior para comunicarse.



Procedimiento

1. Entender los objetivos y la descripción de la práctica.
2. Elaborar el diagrama de cableado.
3. Elaborar el diagrama esquemático.
4. Escribir el código.
5. Cargar el código al arduino.
6. Verificar el funcionamiento y obtener evidencias



Diagrama de Cableado

- * LCD RS pin to digital pin 12
- * LCD Enable pin to digital pin 11
- * LCD D4 pin to digital pin 5
- * LCD D5 pin to digital pin 4
- * LCD D6 pin to digital pin 3
- * LCD D7 pin to digital pin 2
- * LCD R/W pin to ground
- * LCD A to VCC
- * LCD K to GND
- * LCD VSS pin to GND
- * LCD VDD pin to VCC
- * LCD V0 pin al potenciometro

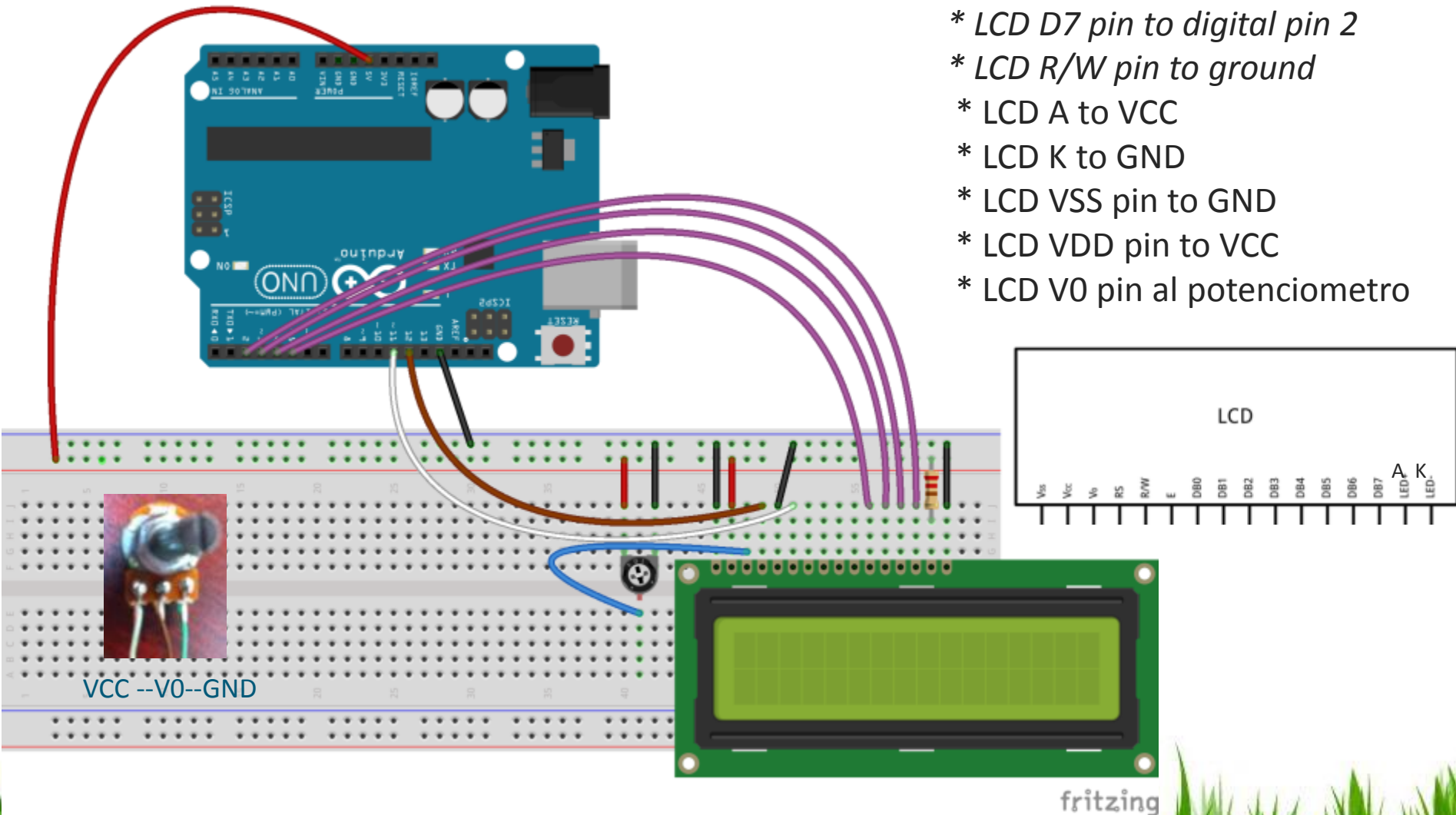
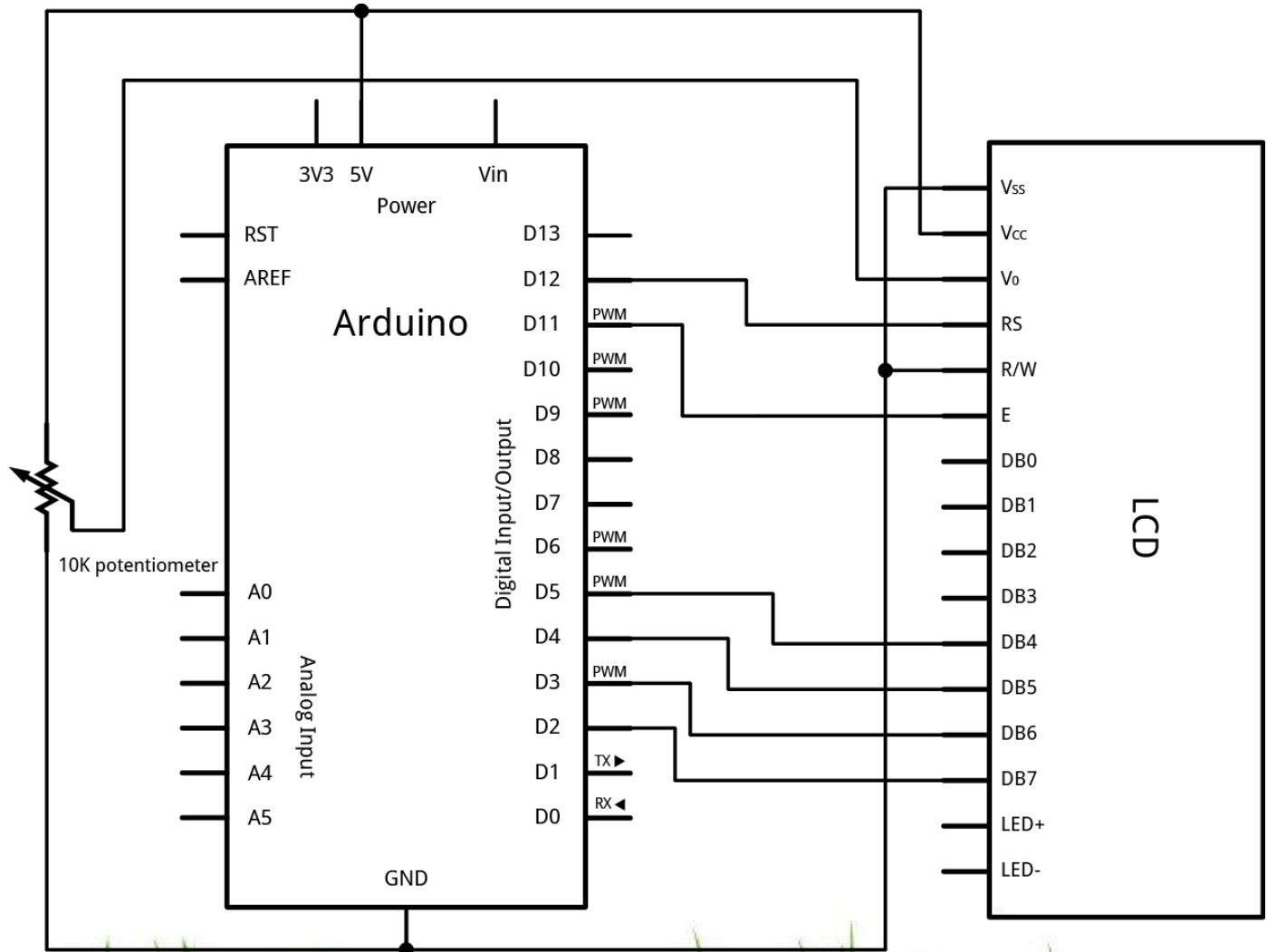


Diagrama Esquemático




Código #1

```
// include the library code:
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
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("Bienvenidos ITIC");
}

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



Código #2

```
// include the library code:
#include <LiquidCrystal.h>
// initialize the library with the numbers of the interface pins
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("..Aviso..");
}

void loop() {
  // Turn off the display:
  lcd.noDisplay();
  delay(500);
  // Turn on the display:
  lcd.display();
  delay(500);
}
```

Código #3

```
// include the library code:
#include <LiquidCrystal.h>
// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
}
void loop() {
  // set the cursor to (0,0):
  lcd.setCursor(0, 0);
  // print from 0 to 9:
  for (int thisChar = 0; thisChar < 10; thisChar++) {
    lcd.print(thisChar);
    delay(500);
  }
  // set the cursor to (16,1):
  lcd.setCursor(16, 1);
  // set the display to automatically scroll:
  lcd.autoscroll();
  // print from 0 to 9:
  for (int thisChar = 0; thisChar < 10; thisChar++) {
    lcd.print(thisChar);
    delay(500);
  }
  // turn off automatic scrolling
  lcd.noAutoscroll();
  // clear screen for the next loop:
  lcd.clear();
}
```


Código #4

The screenshot shows the Arduino IDE interface with the following components:

- Menu:** Archivo, Editar, Programa, Herramientas, Ayuda.
- Code Editor:**

```
// start again at 0
thisChar = 'a';
}
// print the character
lcd.write(thisChar);
// wait a second:
delay(1000);
// increment the letter:
thisChar++;
}
```
- Serial Monitor:** Subido. Global variables use 43 bytes (2,005 bytes for local variables).
- Library Manager:** A list of libraries is shown, with 'LiquidCrystal' selected. A sub-menu is open for 'LiquidCrystal', and 'CustomCharacter' is circled in orange.

Código #5

