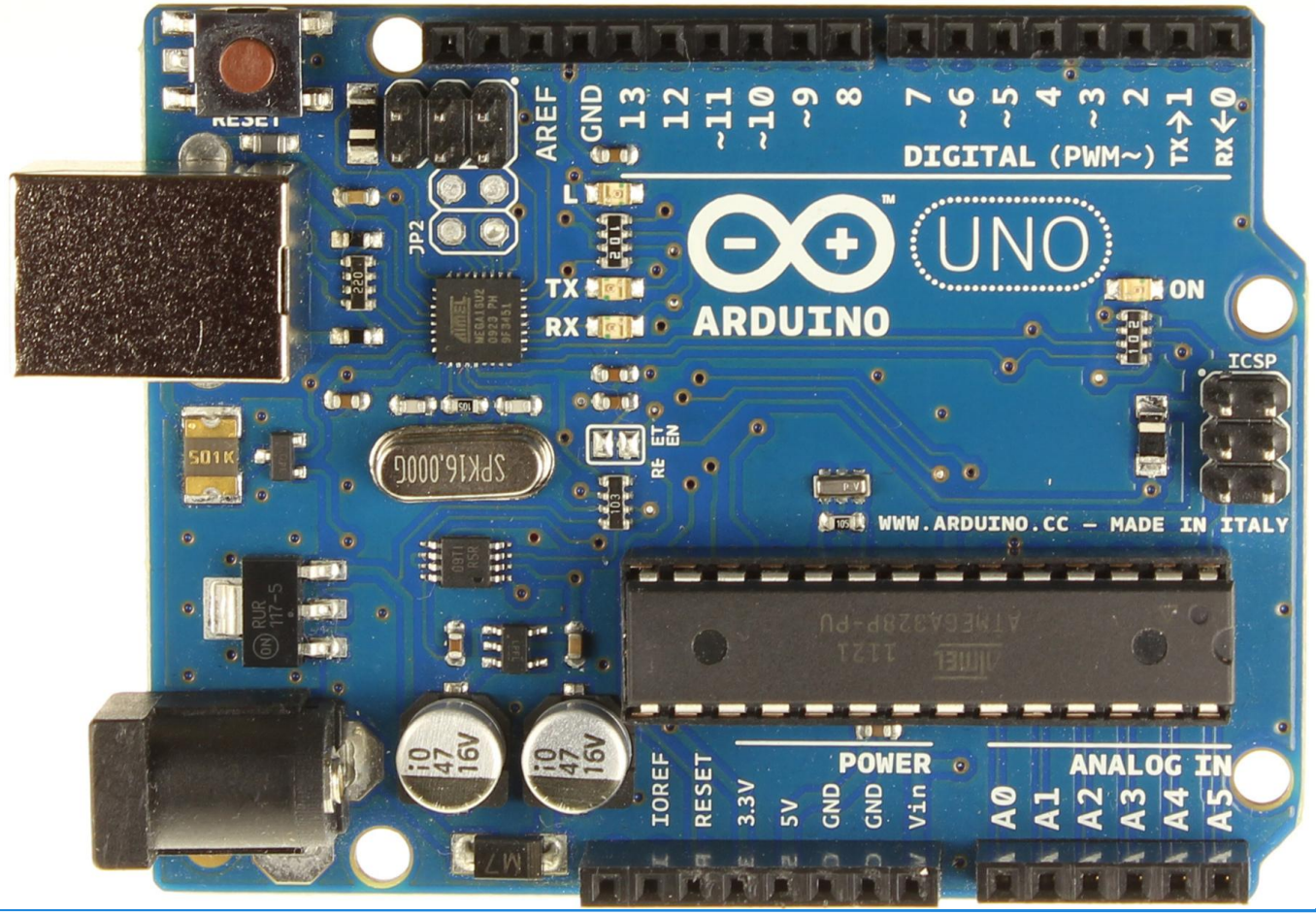
Curso avanzado sobre Arduino

Arduino Avanzado

ElCacharreo.com

Introducción a Arduino: Presente



Introducción a Arduino: Presente

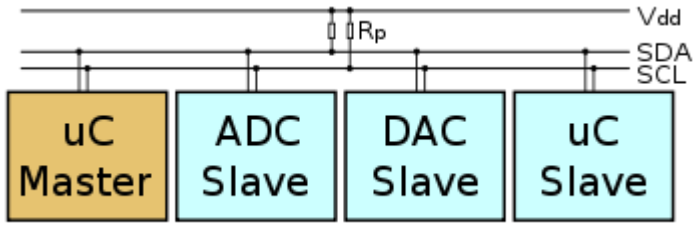
José Antonio Vacas Martínez

blog

javacasm@elcacharreo.com

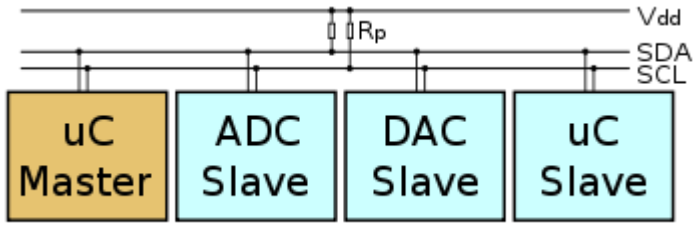
twitter
linkedin

Comunicaciones: I2C



i²C es un bus de comunicaciones en serie. Su nombre viene de *Inter-Integrated Circuit* (Circuitos Inter-Integrados). La versión 1.0 data del año 1992 y la versión 2.1 del año 2000, su diseñador es Philips. La velocidad es de 100Kbits por segundo en el modo estándar, aunque también permite velocidades de 3.4 Mbit/s. Es un bus muy usado en la industria, principalmente para comunicar microntroladores y sus periféricos en sistemas integrados (*Embedded Systems*) y generalizando más para comunicar circuitos integrados entre si que normalmente residen en un mismo circuito impreso.

Comunicaciones: I2C



| start | A7 A6 A5 A4 A3 A2 A1 | R/W | ACK | ... DATA ... | ACK | stop | idle |

Ejemplo I2C

Comunicaciones: I2C

Dispositivos I2C

- Memorias externas
- Sensores
- GPIO
- Potenciómetros
- ADC
- DAC
- .....

Comunicaciones: I2C

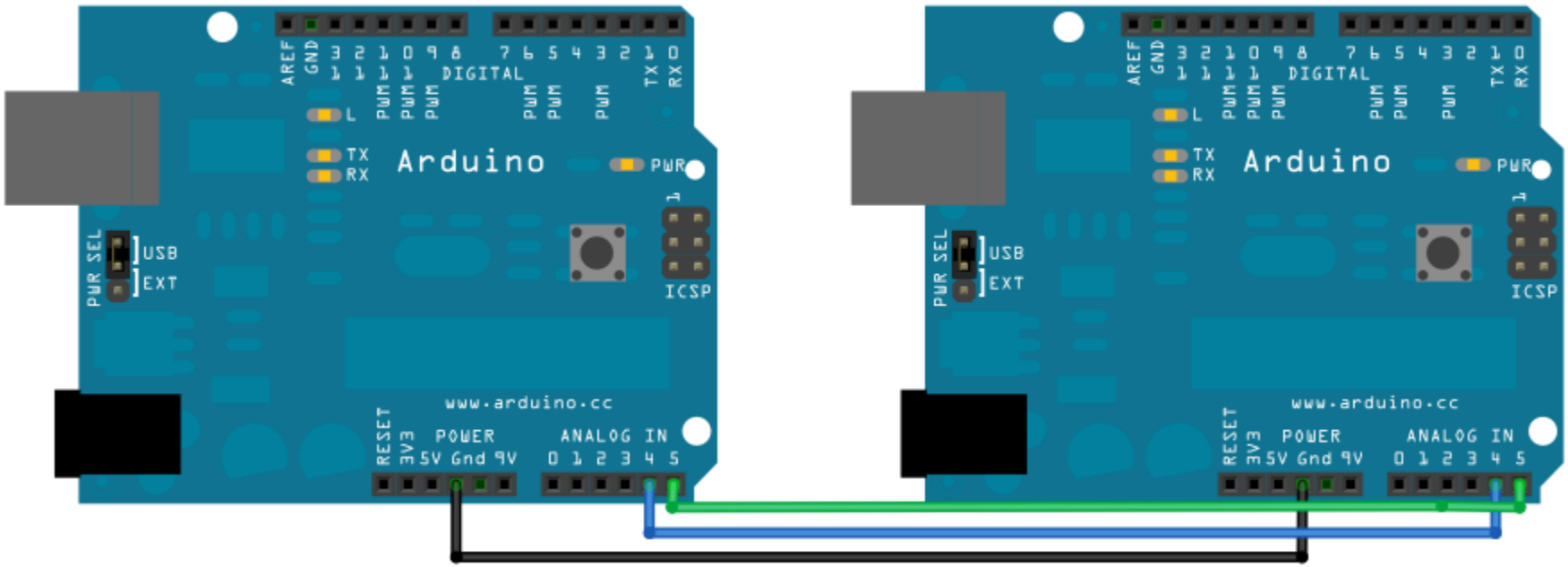
Librería Wire

Esta librería te permite comunicar con dispositivos I2C / TWI. En la mayoría de las placas Arduino, SDA (línea de datos) está en el pin analógico 4, y SCL (línea de reloj) está en el pin analógico 5. En Arduino Mega, SDA esta en el pin digital 20 y SCL en el 21.

Funciones

- begin()
- begin(address)
- requestFrom(address, count)
- beginTransmission(address)
- endTransmission()
- send()
- byte available()
- byte receive()
- onReceive(handler)
- onRequest(handler)

Comunicando 2 arduinos con I2C



Comunicando 2 arduinos con I2C

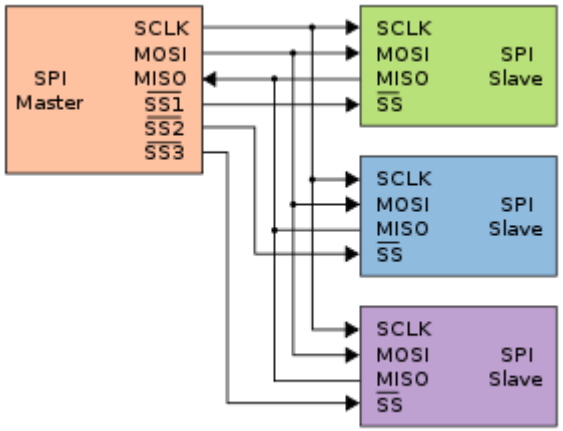
```
#include <Wire.h>

void setup()
{   Wire.begin();
// join i2c bus (address optional for master)
}

byte x = 0;

void loop()
{
  Wire.beginTransmission(4); // transmit to device #4
  Wire.write("x is ");       // sends five bytes
  Wire.write(x);             // sends one byte
  Wire.endTransmission();    // stop transmitting

  x++;
  delay(500);
}
```

http://arduino.cc/en/Tutorial/MasterWriter

```
#include <Wire.h>

void setup()
{
  Wire.begin(4);                // join i2c bus with address #4
  Wire.onReceive(receiveEvent); // register event
  Serial.begin(9600);           // start serial for output
}

void loop()
{
  delay(100);
}

// function that executes whenever data is received from master
// this function is registered as an event, see setup()
void receiveEvent(int howMany)
{
  while(1 < Wire.available()) // loop through all but the last
  {
    char c = Wire.read(); // receive byte as a character
    Serial.print(c);         // print the character
  }
  int x = Wire.read();    // receive byte as an integer
  Serial.println(x);         // print the integer
}
```

Comunicaciones: SPI

SPI



Functions

- begin()
- end()
- setBitOrder()
- setClockDivider()
- setDataMode()
- transfer()

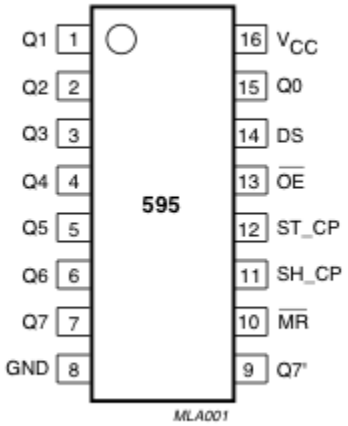
Comunicaciones: SPI

Ejemplos:

- Tarjetas SD
- Módulos Ethernet
- RTC
- ...

Arduino ethernet usa el pin 4 para seleccionar la SD y el 11 para la ethernet. Enlace

Comunicaciones: HC595

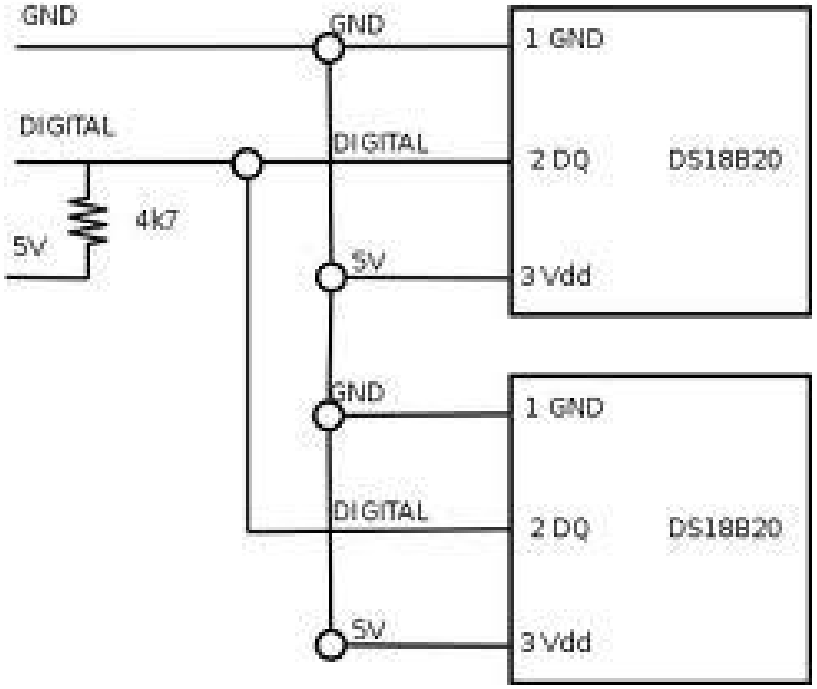


PINS 1-7, 15	Q0 " Q7	Output Pins
PIN 8	GND	Ground, Vss
PIN 9	Q7"	Serial Out
PIN 10	MR	Master Reclear, active low
PIN 11	SH_CP	Shift register clock pin
PIN 12	ST_CP	Storage register clock pin (latch pin)
PIN 13	OE	Output enable, active low
PIN 14	DS	Serial data input
PIN 16	Vcc	Positive supply voltage

Ejemplo

Comunicaciones: OneWire

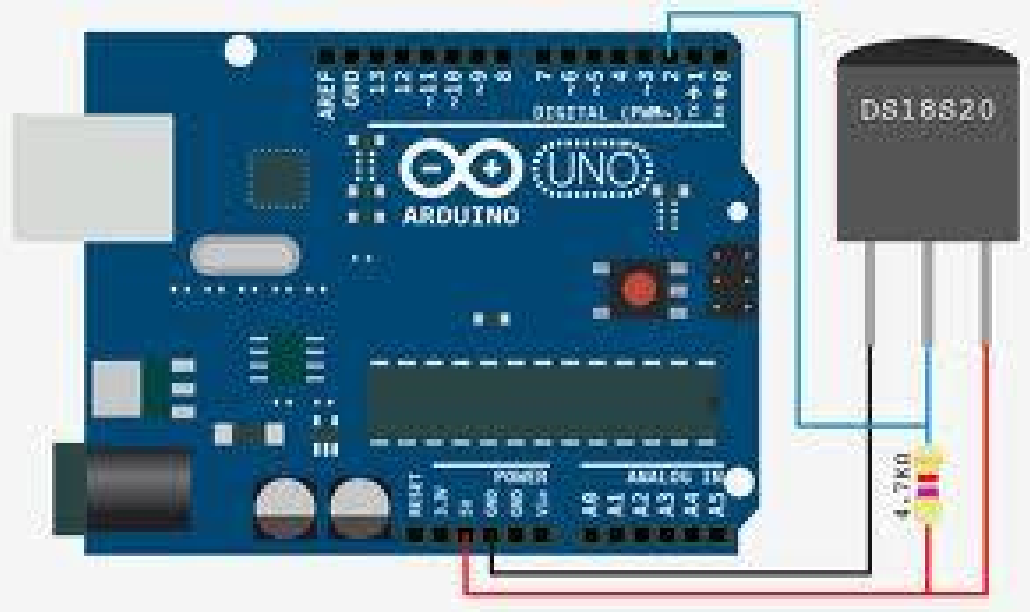
Protocolo propietario de Dallas (Maxim-IC)



Único hilo
ID único
Capacidad parásita

Comunicaciones: OneWire

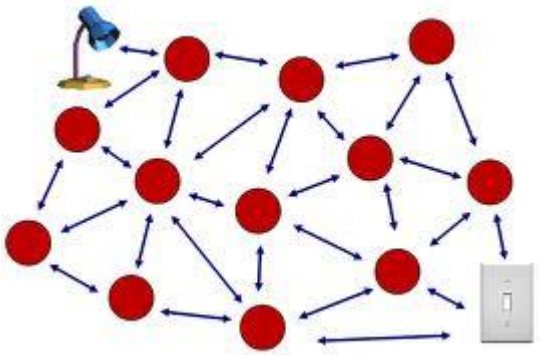
Implementación

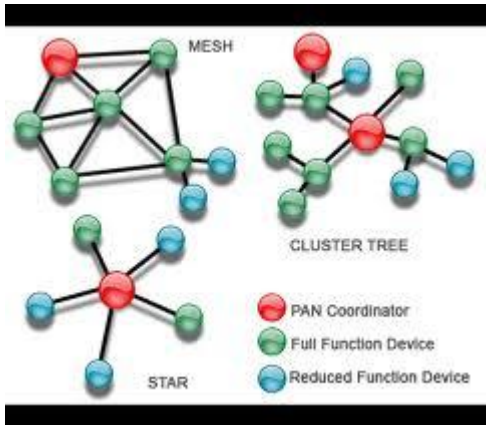


Programando Arduino: Lenguaje

Zigbee

- Coordinador
- Router
- Nodo





Programando Arduino: Lenguaje

Jeenode

Fuentes

arduino

arduino programing notebook

freeduino

Conclusiones

Gracias por vuestra atención