Funcionamiento I2C

Enlace wikipedia funcionamiento I2C

<http://es.wikipedia.org/wiki/I%C2%B2C>

La libreria de arduino encargado de la comunicacion entre i2c es la wire;

<http://arduino.cc/en/pmwiki.php?n=Reference/Wire>

Funciones libreria wire arduino

#### **Functions**

* [begin](http://arduino.cc/en/Reference/WireBegin)()
* [requestFrom](http://arduino.cc/en/Reference/WireRequestFrom)()
* [beginTransmission](http://arduino.cc/en/Reference/WireBeginTransmission)()
* [endTransmission](http://arduino.cc/en/Reference/WireEndTransmission)()
* [write](http://arduino.cc/en/Reference/WireWrite)()
* [available](http://arduino.cc/en/Reference/WireAvailable)()
* [read](http://arduino.cc/en/Reference/WireRead)()
* [onReceive](http://arduino.cc/en/Reference/WireOnReceive)()
* [onRequest](http://arduino.cc/en/Reference/WireOnRequest)()

Documentacion aprendizaje i2c

<http://playground.arduino.cc/Main/InterfacingWithHardware#i2c>

## I2C / TWI

* [The Wire Reference](http://www.arduino.cc/en/Reference/Wire) The Arduino Wire library is used for I2C.
* [i2c\_scanner](http://playground.arduino.cc/Main/I2cScanner) A simple sketch to scan the i2c-bus for devices.
* [Arduino I2C Expansion I/O](http://www.neufeld.newton.ks.us/electronics/?p=241): detailed tutorial on getting I2C working.
* [I2C for X9241 Digital Potentiometer](http://combustory.com/wiki/index.php/X9241A_-_Digital_Potentiometer) - Interfacing with the X9241 Digital Potentiometer
* [I2C for RTC DS1307 Real Time Clock](http://combustory.com/wiki/index.php/RTC1307%20-%20Real%20Time%20Clock) - Interfacing with the DS1307 date/time keeping clock - Great for logging events.
* [Using i2c Based LCD displays, including library and suggested API](http://playground.arduino.cc/Code/LCDi2c)
* [I2C on the ATtiny85 - Libraries for using I2C (both master and slave) on the ATtiny processors.](http://playground.arduino.cc/Code/USIi2c)
* [I2C and Arduino tutorials](http://tronixstuff.wordpress.com/2010/10/20/tutorial-arduino-and-the-i2c-bus/) - Beginners' guides to using I2C bus with Arduino, including worked examples to follow; part of the Arduino tutorials at [tronixstuff.com](http://tronixstuff.com/tutorials)
* Library for the I2C [DS1307 RTC](http://code.google.com/p/ds1307new) with NV-RAM support. With v1.20 you can calculate with the time and a program has been added to modify the RTC over the serial port.
* [Atmel Two Wire Interface](http://playground.arduino.cc/Code/ATMELTWI): Detailed description of how the Wire library accomplishes I2C communication via a deep dive into the source code.
* [Si4707 NOAA Weather Receiver - SAME Decoder Interface](http://www.raydees.com/Weather_Radio.html): I2C based receiver interface for NOAA Weather Radio.
* [I2C communication between two Arduino Boards, with Bus Extender and Optical Isolation](http://www.smacula.blogspot.co.uk/2011/12/communicating-between-arduino-boards.html): A case study that discusses communicating between two Arduino boards, using a P82B715 bus extender, and optically isolating the two Arduino supplies using an ADuM1250.
* [Wire Library Detailed Reference](http://playground.arduino.cc/Main/WireLibraryDetailedReference). An indepth reference manual on the Wire Library. Expands upon the official library documentation.
* [I2C bi-directional level shifter](http://playground.arduino.cc/Main/I2CBi-directionalLevelShifter) Combining 3.3V and 5V I2C components.
* [I2C communication via digital pins](https://sites.google.com/site/marthalprojects/home/arduino/i2c-via-digital-pins) - code which enables any two digital pins to take over I2C communication, in this example with a 24lc256 EEPROM
* [Light-weight and very fast software I2C library *SoftI2CMaster*](http://playground.arduino.cc/Main/SoftwareI2CLibrary) supporting all pins on all ATmegas and ATtinys. It implements only the master part of the protocol but supports clock stretching by slave devices and permits to set a timeout value for clock stretching (up to 10 seconds).
* [PCF8574 Class](http://playground.arduino.cc/Main/PCF8574Class) Class for I2C PCF8574 IO expander.

**Página de guia total desde cero en i2c**

[www.electroensaimada.com/i2c.html](http://www.electroensaimada.com/i2c.html)

Unir mediante programacion i2c arduino raspberry

<http://blog.retep.org/2014/02/15/connecting-an-arduino-to-a-raspberry-pi-using-i2c/>

There’s [RS232](http://es.wikipedia.org/wiki/RS-232) which both support, **however the PI runs on 3v3 whilst the Arduino (UNO) is 5v so I need to add a level converter between the two.** It also limits me to just one arduino and I might need to use more than one so another solution is needed. No supone problema para nosotros porque no vamos a implementar este protocolo de comunicación.

## Enter I2C

Both the PI and Arduino support two additional types of communication for talking to peripheral devices. There’s SPI which is a high speed serial protocol and I2C. **Like RS232, SPI needs level shifters, but not exactly so for I2C.**

**Aqui tenemos la razón para comunicarnos por I2C Y NO POR SPI O TX, RX**

**I2C is a 2 wire protocol allowing for 127 devices to be connected to a single bus. One device is the master (The PI in our case) and then the peripherals.**

**So, we have a solution as long as the Raspberry PI is the I2C Master which is what we want. Also, of the available GPIO pins, only SDA and SCL have pull up resistors, so we are set.**

**El raspberry Pi va a ser el i2c master, y los demás elementos deben ser slave.**