

# Curso avanzado sobre Arduino

Arduino avanzado



ElCacharreo.com



# Arduino avanzado: Presente



# Arduino avanzado: Presente



José Antonio Vacas Martínez

blog  
javacasm@elcacharreo.com  
twitter  
linkedin



# Arduino avanzado: Python

Después de instalar **pySerial** (`python setup.py install`), usamos lo siguiente para leer datos: (<http://pyserial.sourceforge.net/>)

```
>>> import serial
>>> ser = serial.Serial('/dev/tty.usbserial', 9600)
>>> while 1:
...     ser.readline()
'1 Hello world!\r\n'
'2 Hello world!\r\n'
'3 Hello world!\r\n'
```

Enviar datos a Arduino es muy sencillo:

```
>>> import serial # if you have not already done so
>>> ser = serial.Serial('/dev/tty.usbserial', 9600)
>>> ser.write('5')
```

Conectaremos al mismo device que conectamos el entorno

Se suele crear un link entre el dispositivo que se usa y /dev/tty.usbserial para facilitar

<http://playground.arduino.cc/Interfacing/Python>



# Arduino avanzado: Python

```
# Ejemplo tomado de https://gist.github.com/Juanlu001/8256958
# Librerías empleadas: Pyserial y Numpy
import time
import serial
import numpy
Lecturas=20
data=numpy.zeros((Lecturas, 2))
arduino = serial.Serial ('COM3', baudrate=9600,timeout=1.0)    # Abrimos la conexión con Arduino
arduino.setDTR(False)      # Reseteo de Arduino
time.sleep(1)
arduino.flushInput()
arduino.setDTR(True)
with arduino:
    contador = 0
    while contador < Lecturas:
        try:
            line = arduino.readline()
            print (line)
            if not line:      # Descartamos líneas vacías
                continue
            data[contador] = numpy.fromstring(line.decode('ascii', errors='replace'),sep=' ')
            contador += 1
        except KeyboardInterrupt:
            print("Exiting")
            break
    print(data)    # Reproducir muestras
```



# Conclusiones

Gracias por vuestra atención

