

# Osciloscopio con Arduino

Federico Soto Badilla, Diego Osvaldo Ochoa de la Cruz

30 de mayo del 2018



# Objetivos

- Aprender a utilizar la tarjeta Arduino.
- Aprender a graficar en tiempo real en Python.
- Utilizar lo aprendido en clase para poder elaborar un osciloscopio.

# Funcionamiento

## Lista de materiales

### Electrónicos

- Arduino UNO
- Computadora
- 8 Resistencias de 1kOhm

### Mecánico

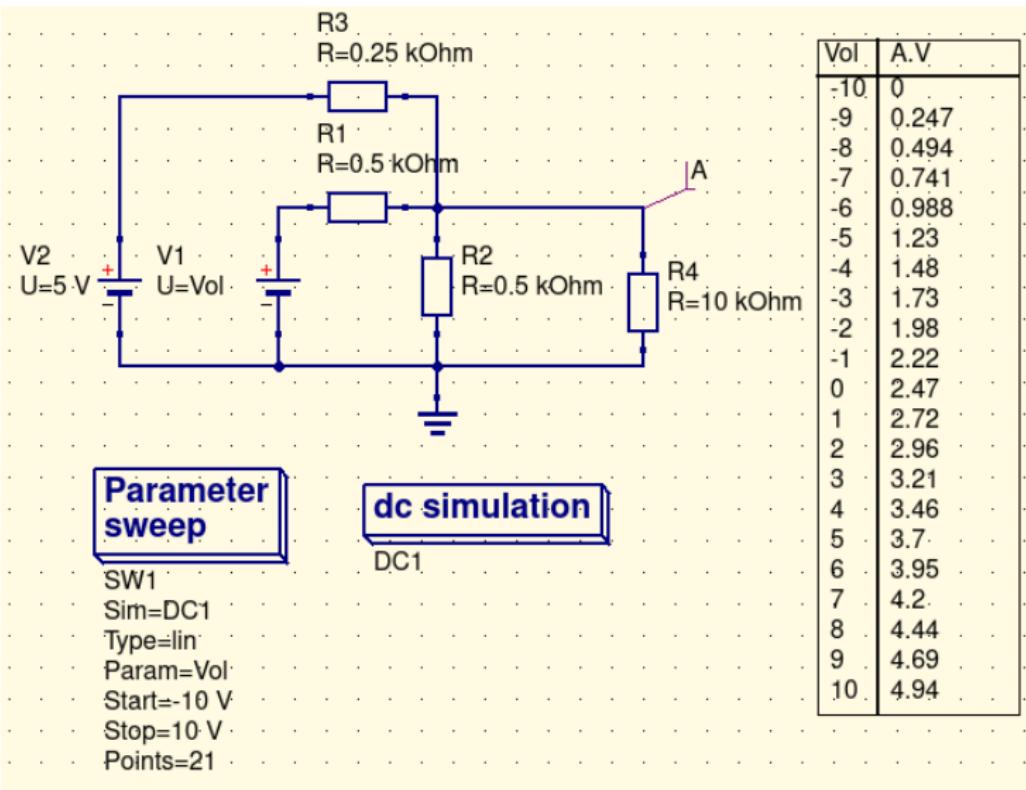
- Protoboard
- Jumpers

### Software

- Editor de Arduino
- Python

# Funcionamiento

## Circuito

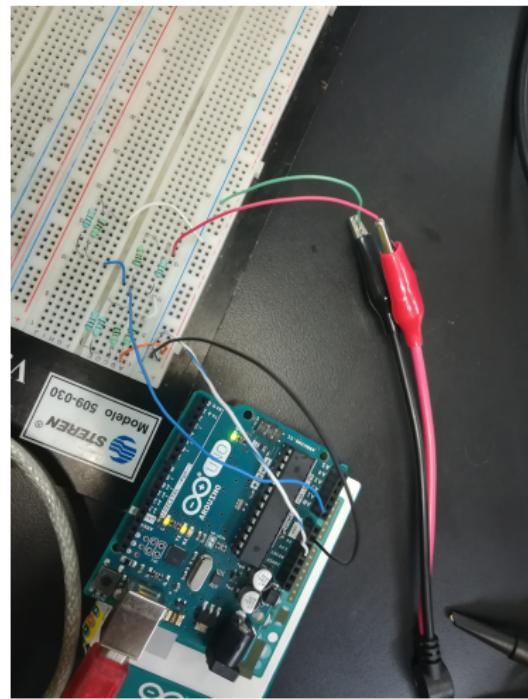
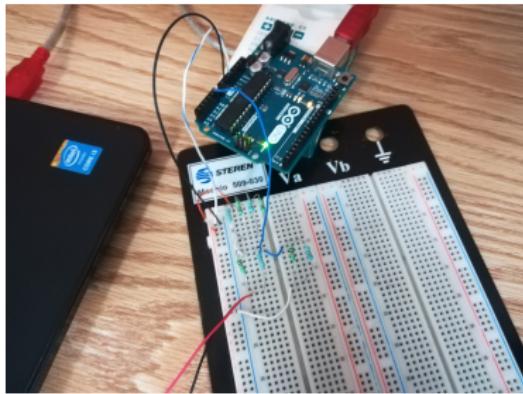


# Funcionamiento

## Programa

```
52 #Modulo para leer de un puerto serial
53 import serial
54 #Modulo para trabajar con arreglos
55 import numpy as np
56 #Modulo para hacer el plot
57 import matplotlib.pyplot as plt
58 #Modulo para graficar en tiempo real
59 from drawnow import *
60
61 #Función para hacer la gráfica
62 def MakeFig():
63     #Definir límites para Y
64     plt.ylim(-10,10)
65     plt.xlim(0,50)
66     #Título de la gráfica
67     plt.title('Osciloscopio')
68     #Colocar la rejilla
69     plt.grid()
70     #Título para Y
71     plt.ylabel('Volts')
72     #Plot del arreglo
73     plt.plot(VOLT, 'ro-')
74
75 #Crear un objeto serial para los datos
76 ArduinoData = serial.Serial("/dev/ttyACM0", 115200)
77 VOLT = []
78 X = []
79 #Le dice a matplotlib que inicie en modo interactivo
80 plt.ion()
81 i = 0
82
83 #Ciclo que continua para siempre
84 while True:
85     #Ler una linea de texto del puerto serial
86     ArduinoString = ArduinoData.readline()
87     try:
88         #Convertir el texto a float y mandarlo a la función de voltaje con los datos del circuito
89         #Volt = V/float(ArduinoString), 5,0.5,0.5,0.25)
90         Volt = 4*float(ArduinoString) - 2.78
91         #Volt = float(ArduinoString) - 2.78
92         #Agrega datos a los arreglos para el plot
93         VOLT.append(Volt)
94         X.append(len(X))
95         #Llama a drawnow para actualizar el plot
96         drawnow(MakeFig)
97         #Avanzar el contador
98         i += 1
99         #Borrar el primer elemento de cada arreglo
100        if i > 50:
101            VOLT.pop(0)
102            X.pop(0)
103    except:
104        pass
```

# Resultados



# Resultados

