Monitorización (y control) de una carga mediante el uso de Arduino

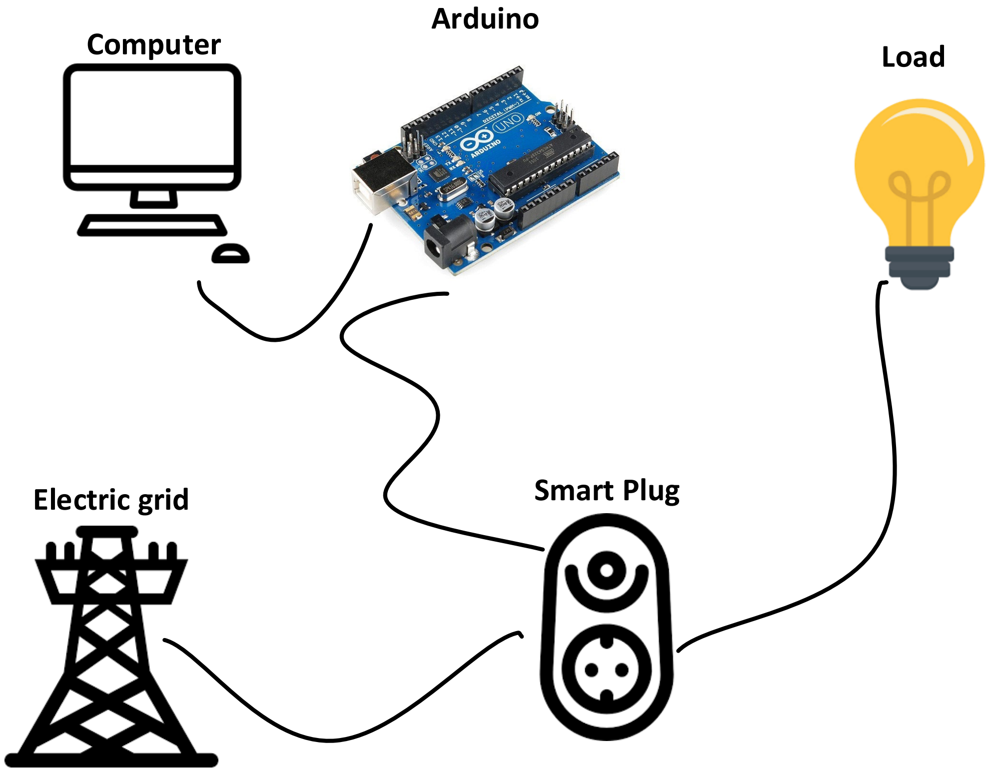
Montaje de los elementos básicos del Smart Plug para medir y controlar una carga domestica para ayudar a los estudiantes a desarrollar el Proyecto Arduino

Sesión 1: Monitorización de la carga

Table of contents

Introduction

Today's laboratory session has the objective of assembling the main elements to measure our project domestic load, putting into practice the preliminary concepts developed in the documentation uploaded to the Learnify platform. The main goal of the session is to build up a Smart Plug device to measure the load with our Arduino device and print the current results on our computer screen. Furthermore, there will be some instructions to help the student to control the load with a relay.



Theory resume

The Arduino analogic pins (from A0 to A5) have a 10-bit resolution, so when there is a call to read an analogic pin, it returns integers from 0 to 1023. These integers are the so-called ADC value. The ADC takes as a reference voltage the Arduino V_{cc} supply to convert the signal from analogic to digital.

The signal is scaled from 0 to V_{cc} and it is proportional to the current that flows through the sensor. To show the current consumed by the load, the analogic lecture (ADC Value) must be converted to amperes. The mathematical conversion needed depends on the sensor used.

Root Mean Square value (RMS) and sampling

The current from the power outlet is a cyclically alternating wave (sinusoidal), so the instantaneous value given by the sensor is not useful to calculate the mean consumption of a load. The root mean square (RMS hereinafter) value will be used, which is the value of the current that would produce the same average power dissipation in a resistive load.

The RMS value is calculated:

$$I_{RMS} = \lim_{T \to \infty} \sqrt{\frac{1}{T} \cdot \int_0^T i^2(t)dt} \quad \textbf{(1)}$$

T is the time of RMS calculation, in seconds.
i(t) is the value of the current wave in the instant t, in amperes.

The current value is read every time the loop is executed, so it is not a continuous value in time, but a discrete one, depending on the time where each measure is done. Hence, the formula to calculate the RMS is:

$$I_{RMS} = \sqrt{\frac{1}{T} \cdot \sum_0^N i^2(n) \cdot \Delta t} \quad \textbf{(2)}$$

T is the period of RMS calculation. $1/T$ *is the frequency of the cycle, 50 Hz*

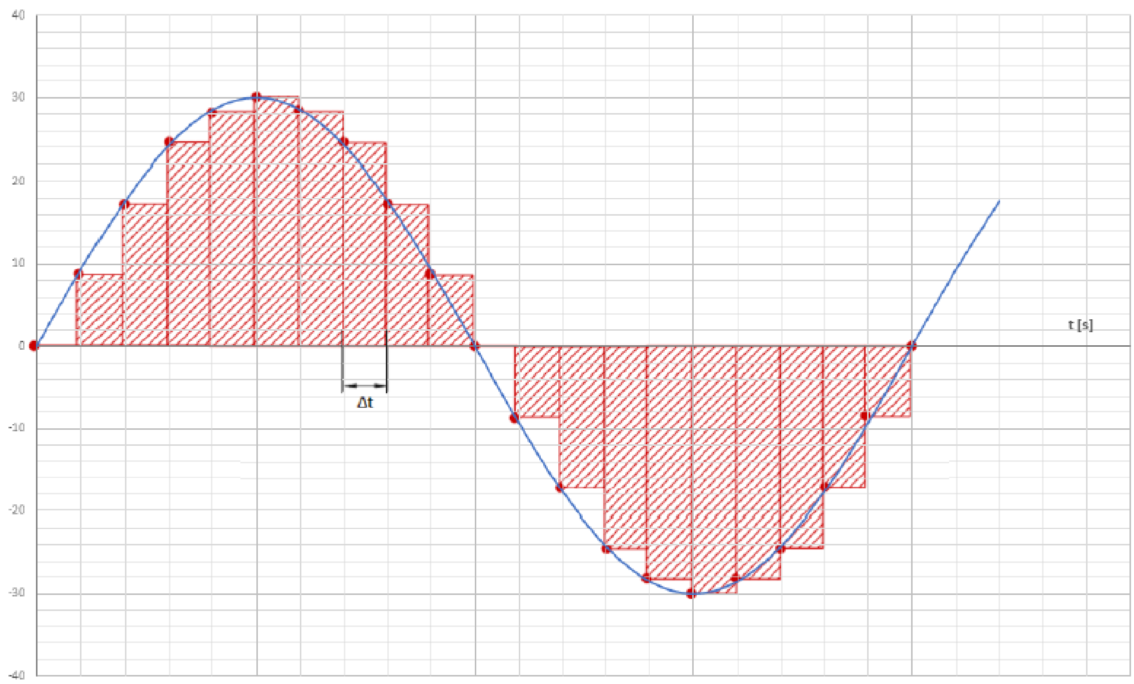
N is the number of samples taken per period.

n is one of the N samples.

i(n) is the recorded value of the current in sample number n, in amperes.

Δt is the lapse of time between samples (in seconds).

The figure below is the graphic representation of the transformation from continuous to discrete form.



Since the frequency of the current wave is 50 Hz, 20 samples (1 sample each millisecond) should be enough to obtain an accurate result.

Replacing these values in the calculation of RMS:

$$I_{RMS} = \sqrt{50 \cdot \sum_{0}^{20} i^2(n) \cdot 0,001} \quad (3)$$

Smart plug current measurement assembling

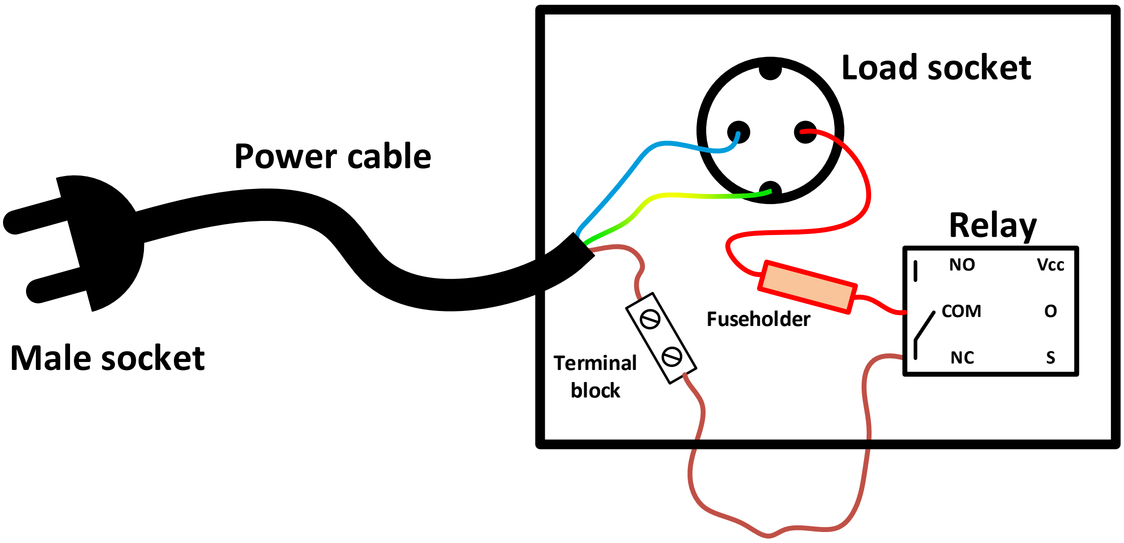
Hardware assembling

The first step before measuring current is to assemble the smart plug elements to have an available power outlet ready to be measured.

The elements for this first assembling will be the next ones:

- Connection box with a female power socket.
- Mains power cable with a male power socket.
- 10 A fuse.
- Inline fuse holder.
- Electrical wire and a terminal block to do the connections.
- 10 A relay.

The basic assembling is shown in the figure below.



The power will flow from the grid to the load through the power cable and the load socket. However, it will be necessary to separate the power cable wires to measure the current through one of them. Furthermore, we will add two extra elements. A fuse holder to protect the system and a power relay to be able to connect and disconnect the load in the future.

In this laboratory session, it is better to have connected the consumption all the time, for this reason, we will do the connections using the NC (normally closed) port of the relay.

Current measurement assembling

The SCT013 is a current transformer (CT) device which provides a measurement proportional to the current flowing through the main circuit. The measurement is made by electromagnetic induction.

The current transformers are made up of a primary coil, a ferromagnetic core, and a secondary coil. The current that flows through the primary coil, I_p, and induces a current in the secondary coil, I_s, proportional to the first.

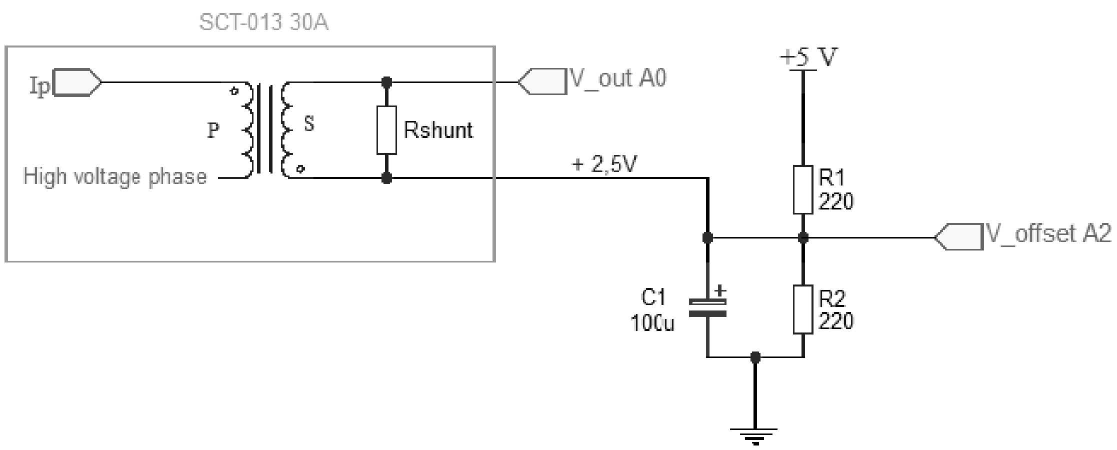
The transformer has a built-in burden resistor to provide a measurement of voltage proportional to that current flowing in the secondary coil.

The current consumed by the electrical appliances is AC, and therefore is the current flowing in the secondary coil of the sensor. After it goes through the burden resistor, the output voltage is also alternating, so it may turn to negative values. Analog input pins of the Arduino can only read positive voltages, so the output voltage of the sensor must be modified to avoid damages on the hardware.

One way to do so is to sum a voltage offset in DC to the signal by adding two resistors to provide a middle voltage point between GND and Vcc in which the ground terminal of the sensor is connected.

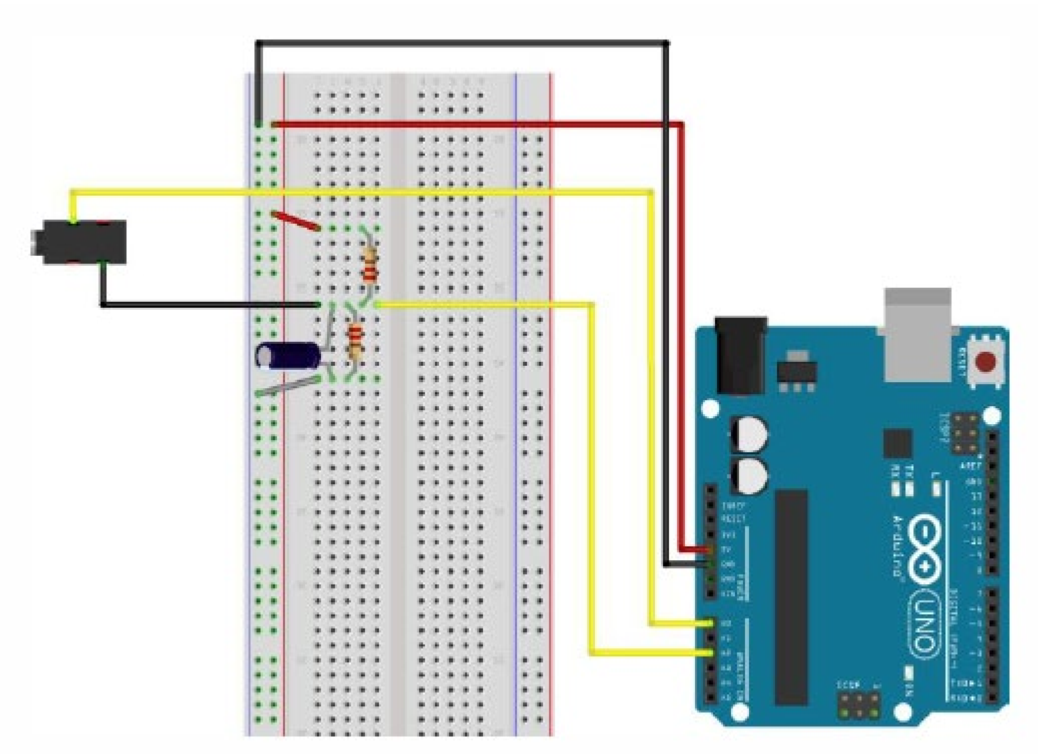
For example, if a sensor has a range output of ± 1 V, the voltage peak is of ± 1,414 V and the voltage from minimum to maximum is 2,828 V. With a DC offset of 2,5 V, the voltage range goes from 1,086 V to 3,914 V.

The electrical circuit is represented in the figure below. The resistors can be of any value, but they must be equal (in this case, two resistors of 220 Ω have been used). The capacitor can be added to act as a filter of the noise in case of voltage peaks.

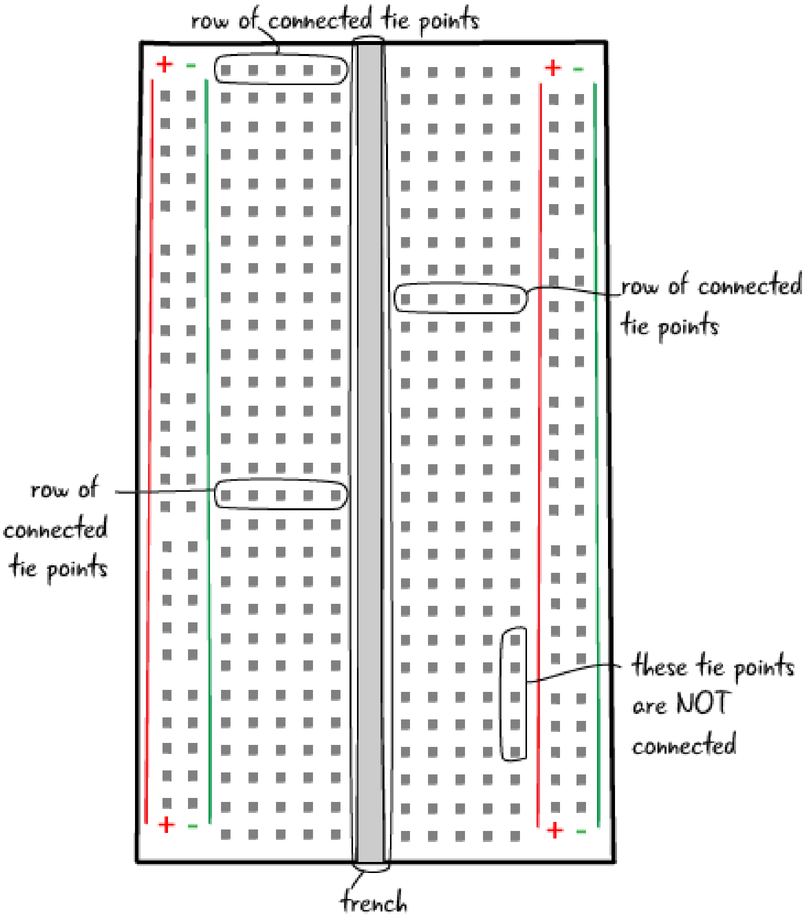


Two analogic inputs of the Arduino are used in this project. The first is to read the output voltage of the sensor, and the second is to read the exact DC voltage offset to which the sensor is referenced. By measuring the two voltages, instead of taking for granted that the offset is exactly 2,5 V, the actual difference of voltages can be found and, with this, the exact current flowing in the primary coil of the CT.

The physical connections with the breadboard can be done in the following way:



Remember the connections of a breadboard:

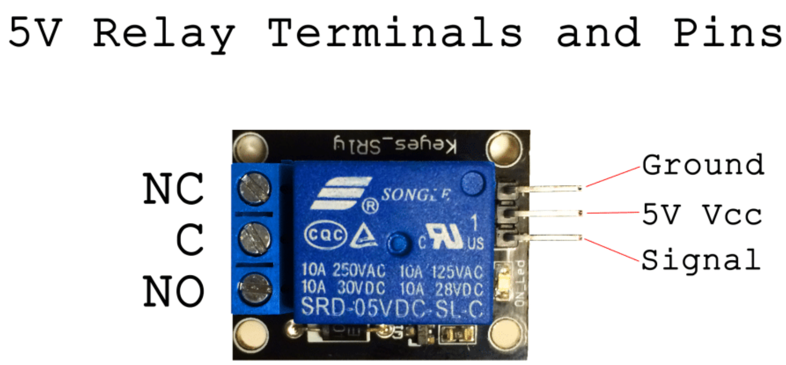


Smart plug relay behavior

(Info extracted from http://www.circuitbasics.com/setting-up-a-5v-relay-on-the-arduino).

One of the most useful things you can do with an Arduino is to control high voltage devices like fans, lights, heaters, and other household appliances. Since the Arduino operates at 5V it can't control these devices directly, but you can use a 5V relay to switch the 230V current and use the Arduino to control the relay.

A relay has three high voltage terminals (NC, C, and NO) that connect to the device you want to control. The other side has three low voltage pins (Ground, Vcc, and Signal) which connect to the Arduino.



- NC: Normally closed 230V terminal
- NO: Normally open 230V terminal
- C: Common terminal
- Ground: Connects to the ground pin on the Arduino
- 5V Vcc: Connects the Arduino's 5V pin
- Signal: Carries the trigger signal from the Arduino that activates the relay

Inside the relay is a switch that is connected to an electromagnet. When the relay receives a HIGH signal at the signal pin, the electromagnet becomes charged and moves the contacts of the switch open or closed.

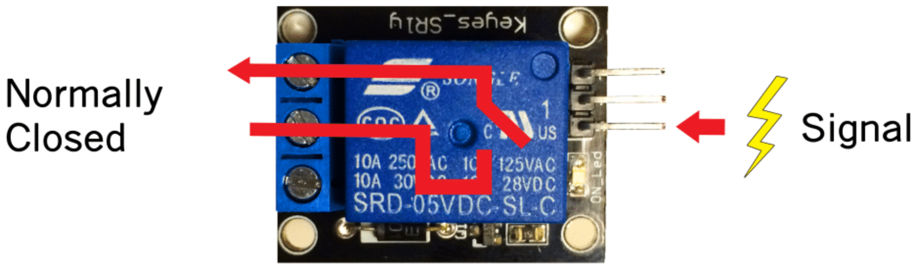
The relay has two different types of electrical contacts inside – normally open (NO) and normally closed (NC). **The one you use will depend on whether you want the 5V signal to turn the switch on or turn the switch off.** The 230V supply current enters the relay at the common (C) terminal in both configurations.

Normally Open

In the normally open configuration, when the relay receives a HIGH signal the 230V switch closes and allows current to flow from the C terminal to the NO terminal. A LOW signal deactivates the relay and stops the current. So if you want the HIGH signal to turn ON the relay, use the normally open terminal:

Normally Closed

In the normally closed configuration, a HIGH signal opens the switch and interrupts the 230V current. A LOW signal closes the switch and allows current to flow from the C terminal to the NC terminal. Therefore, if you want the HIGH signal to turn OFF the 230V current, use the normally closed terminal:



For the development of your course project, you will use the configuration more suited for your appliance. For example, if your device will work in a short period of time each day (e.g. a washing machine), it will be recommended to use the NO configuration and only connect the device in these periods of time.

On the contrary, if your device will be usually running except for short periods of time (e.g. a fridge) it will be more adequate to use an NC configuration. In this way, if the Arduino signal fails for some reason, the appliance will continue running.