SCIENCE
ENGINEERING
TECHNOLOGY

25th
Anniversary

Excellence in education for 25 years

Matrix Technology Solutions Limited
The Factory, 33 Gibbet Street
Halifax, HX1 5BA, UK

t: +44 (0)1422 252380
f: +44 (0)1422 341830
e: sales@matrixtsl.com

www.matrixtsl.com

0 - 24V DC Power Monitor for Arduino UNO

Hardware:

ACS723 Current Sensor

1 x 10K Resistor 1% or better

1 x 2.7K Resistor 1% or better

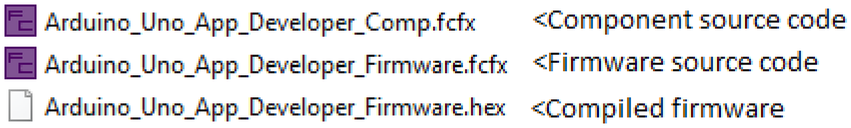
Solderless Breadboard

Before any of the free Apps for Arduino can be run for the first time, you will need to have Arduino driver installed via the Arduino ide software page.

You will also need to load the Arduino Uno App Developer Firmware from this Wiki page to your Arduino Uno.

You can manually find the firmware for different target devices by going to the Wiki and searching for scada slaves.

The zip file will contain three files:



Arduino_Uno_App_Developer_Comp.fcfx	<Component source code
Arduino_Uno_App_Developer_Firmware.fcfx	<Firmware source code
Arduino_Uno_App_Developer_Firmware.hex	<Compiled firmware

Once loaded on to Ardunio, any of the Arduino apps can be run without reloading the firmware.

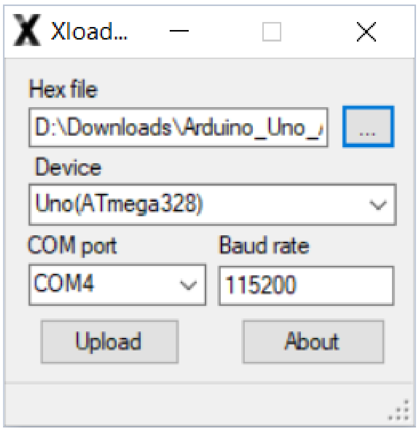
Download the firmware from the first link.

Extract the zip file into a folder of your choice.

Of course, you can use Flowcode to load the firmware then send it to the Arduino.

If you do not have Flowcode, then download XLoader from here.

XLoader enables you to upload hex files very easily:



Hex file

D:\Downloads\Arduino_Uno_

Device

Uno(ATmega328)

COM port

COM4

Baud rate

115200

Use the three dots on XLoader to browse to where you extracted the Arduino_Uno_App_Developer_Firmware.hex zip file.

Select **Arduino_Uno_App_Developer_Firmware.hex**

Make sure Device is showing **Uno(ATmega328)**

Run Device **Manager** by right-clicking on the windows key and left-click on Device manager.

Expand

Look for either Arduino Uno or USB Serial Device (depends on which drivers are used) and remember the Com port number.

Select correct com port in XLoader, then click on Upload.

You should see the RX and TX LEDs on the Arduino flash while data is being transferred.

If it is not, then make sure no other software has taken the port over.

After the transfer is complete, you are now able to run any of the App developer's software.

Load the App by double-clicking on Power Monitor Arduino UNO bat file.

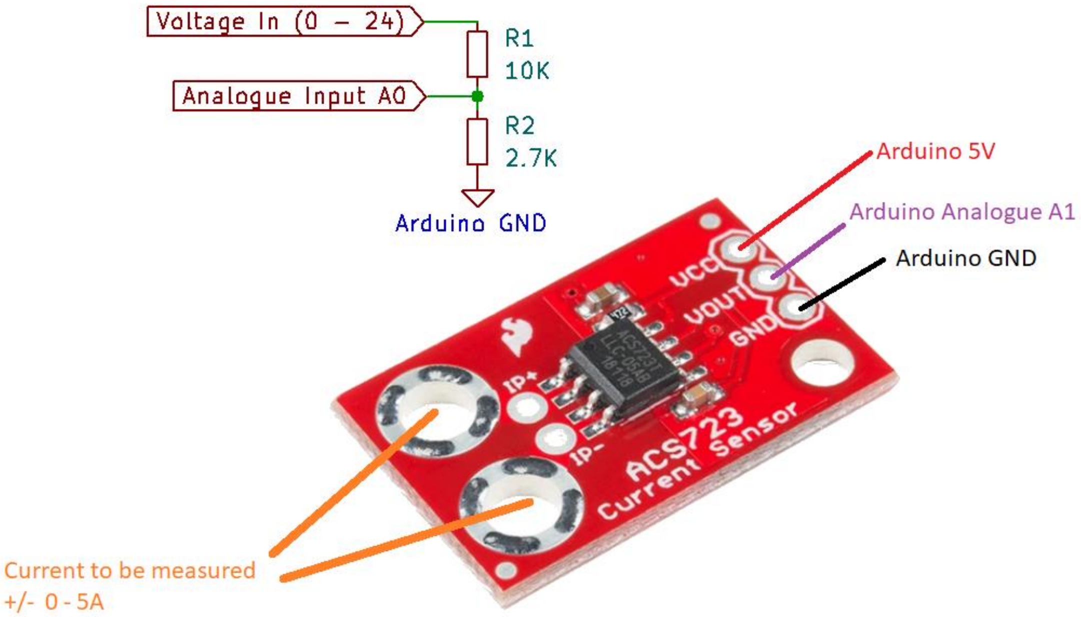
You can also right-click on the bat file and select **Send to,** then Desktop **(create a short-cut)**

The mouse scroll wheel can zoom in & out.

When happy with the size, right-click, select **Look at the origin**, Which will centralise the App

Right-click in the panel and select **Properties** & Select the correct com port for Arduino.

Before running the DC power monitor, make sure all the connections are correct:



Ensure there is no current flowing through the current sensor board, it does not matter if the input to R1 is connected to a DC voltage source or not at this stage.

To run the App, Left-click on **Go**.

If the App Developer is running successfully, then both RX and TX LEDs on the Arduino will constantly flash approximately once a second.

If they do not, then see Troubleshooting.

With no current flowing, the current should read 0.00A.

If it is slightly off, then click on the Zero, which is Next to the power gauge.

If it is more than 0.5A off, then see Troubleshooting.

If the Zero switch is used, the required offset value will be stored in a **Settings.csv** file, which is saved in the Root of Power Monitor folder.

If the value shown is 0.00A, the voltage from the 24V max D.C voltage source (if not already present on R1) can be connected as in the Connections diagram above.

The load can be connected in series with large plated through holes (PTH).

The maximum current measured is +/- 5.00A therefore does not matter which way the DC current will flow.

It will not cause any harm but will give a negative reading instead.

There are two sets of radio buttons, which can only be selected when the App is running.

The **Charts:**

All three charts will only appear when **Logging on** is enabled.

Make sure the panel is zoomed in enough for the charts to be displayed.

The chart can be temporarily paused, or reset, so the seconds start from 0.

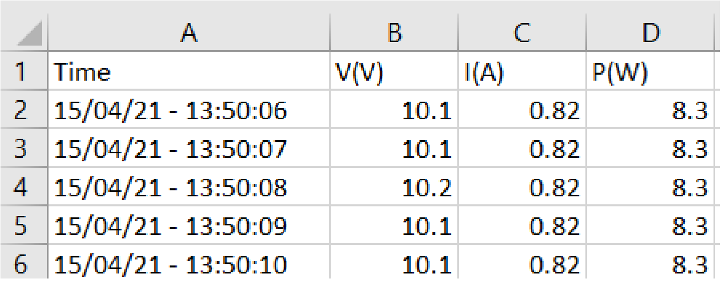
CSV File:

Each time Logging On is selected, a new **DataX.csv** containing Time, Voltage, Current & Power will be saved in the Root of Power Monitor folder.

X will increment each time a new Data file is generated.

When current logging session has finished, it is important to always select **Logging Off** before selecting Logging **On** for the new file to be created.

A typical data file will look like this:



	A	B	C	D
1	Time	V(V)	I(A)	P(W)
2	15/04/21 - 13:50:06	10.1	0.82	8.3
3	15/04/21 - 13:50:07	10.1	0.82	8.3
4	15/04/21 - 13:50:08	10.2	0.82	8.3
5	15/04/21 - 13:50:09	10.1	0.82	8.3
6	15/04/21 - 13:50:10	10.1	0.82	8.3

Troubleshooting.

When App is logging, and you get unexpected results.

Unexpected results could be down to a communication, faulty component or component /wire connection issue.

Other causes could be a faulty Arduino or a software bug.

Has the firmware been loaded and the correct com port selected (if more than one)?

There are two ways of determining if it is a communication issue.

If both TX and RX LEDs are not flashing while App Developer is logging.

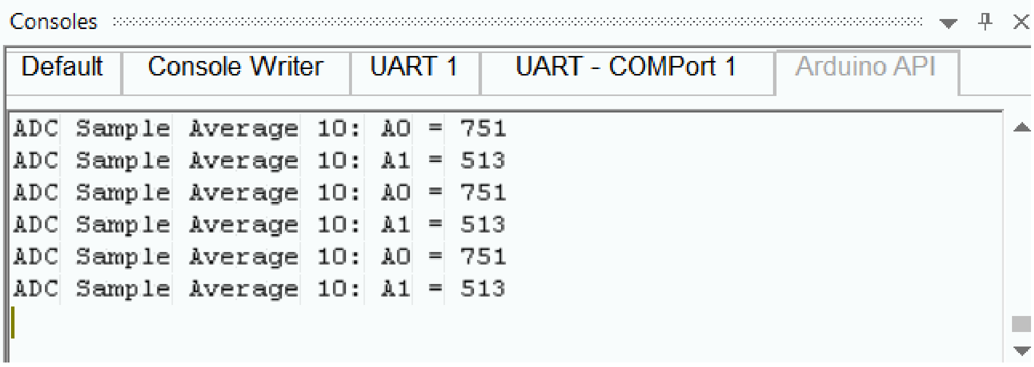
Or open the console (View menu, select Consoles).

Undock the console so the App Developer panel can be made larger.

In the **View** tab, select **Consoles** tick box.

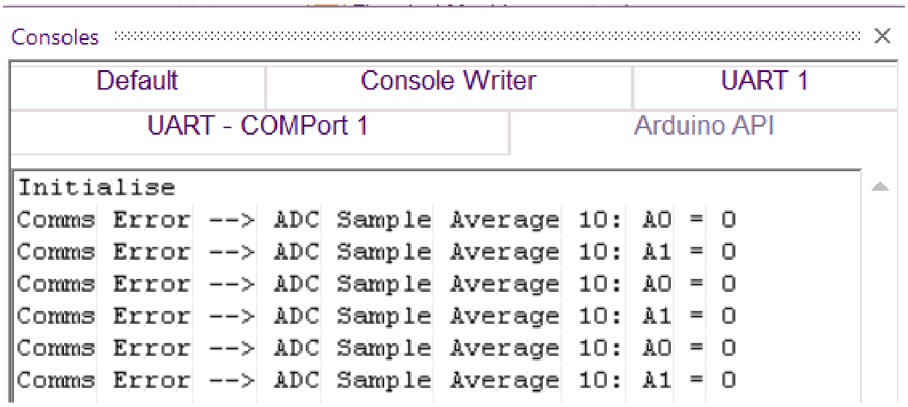
Select **Arduino API** tab.

If communication OK, you will see something like:



```
ADC Sample Average 10: A0 = 751
ADC Sample Average 10: A1 = 513
ADC Sample Average 10: A0 = 751
ADC Sample Average 10: A1 = 513
ADC Sample Average 10: A0 = 751
ADC Sample Average 10: A1 = 513
```

If there is an issue with communication or firmware, then you will see:



```
Initialise
Comms Error --> ADC Sample Average 10: A0 = 0
Comms Error --> ADC Sample Average 10: A1 = 0
Comms Error --> ADC Sample Average 10: A0 = 0
Comms Error --> ADC Sample Average 10: A1 = 0
Comms Error --> ADC Sample Average 10: A0 = 0
Comms Error --> ADC Sample Average 10: A1 = 0
```

If it is a communication issue:

Is the correct com port selected within properties?

Has some other software taken over the port?

Is only one instance of App developer run at a time?

With the hardware issue, so long as communication is confirmed working, measure the pin's voltage that is giving unexpected results.

It should be steady and not floating.

Current reading is wrong:

The current sensor board works by outputting ½ VCC to Analogue pin A1 when now current is flowing.

Measure the voltage on the output of the current sensor board (with no current flowing), confirm that it is ½ VCC.

If the voltage is not ½ VCC to Analogue pin A1, then remove the connection from the output.

If the voltage on the current sensor PCB output goes to ½ VCC, then there is an issue with Arduino hardware or loaded firmware.

Otherwise, the sensor board is faulty or the wrong type.

Make sure you measure VCC across the sensor power supply connections,

If the voltage is correct, then make sure the Arduino has the Arduino App developer firmware programmed.

Voltage reading is wrong:

The voltage reading works by the Arduino analogue pin A0 measuring the voltage across R2 (2.7K) of the potential divider R1/R2.

The voltage across R2 should always be 2.7/10 = 0.27th of the input.

For example, if the input to R1 is 10V DC the voltage going into A0 should be 0.27 * 10 = 2.7V

Confirm that is the case.

If it is then the issue could be with the Arduino firmware or hardware.

For further advice, there will be help available via the forums.