Oscilloscope using Arduino and Processing

The ADC module of Arduino is used to sample an analogue signal and send the acquired measurements to a programme running on a PC. The PC programme processes the values acquired and offers tools to:

- display them in their corresponding plots
- stabilise the display based on a trigger voltage level
- measure signal frequency and voltage
- plot the values in XY mode
- save the values in a text file

The timer module of Arduino is used to generate a square wave. The PC programme is used to set the signal's:

- frequency
- duty cycle

The system was first written by Rogerio Bego. This version was adapted from his original work, translating all legends to English and fixing a small number of behavioural and graphics bugs. It is now distributed for the use of this lab in a single zip file.

This user guide is oriented towards the use of the system in a Windows PC, but using the environment in an Apple Mac PC, should be similar.

The arduino programme

Download the distribution zip file carrying the files of the oscilloscope system.

Extract it to a convenient location.

Run the **Arduino IDE** and use it to open the oscilloscope sketch found in the folder:

...\Arduino as an oscilloscope\arduino

The sketch is called:

oscilloscope_arduino.ino

Compile and upload the sketch to the Arduino UNO.

Within the **Arduino IDE**, go to the **Tools** tab and take a note of the serial port to which the Arduino is connected to.

The Processing environment

The BegOscopio, the PC programme, is a sketch written in the language **Processing version 3**, whose environment must be running in order to execute the sketch. **Processing** can be found at the location:

...\Arduino as an oscilloscope\processing-3.5.4-windows64\processing-3.5.4

or

...\Arduino as an oscilloscope\processing-3.5.4-windows32\processing-3.5.4

and is called:

processing.exe

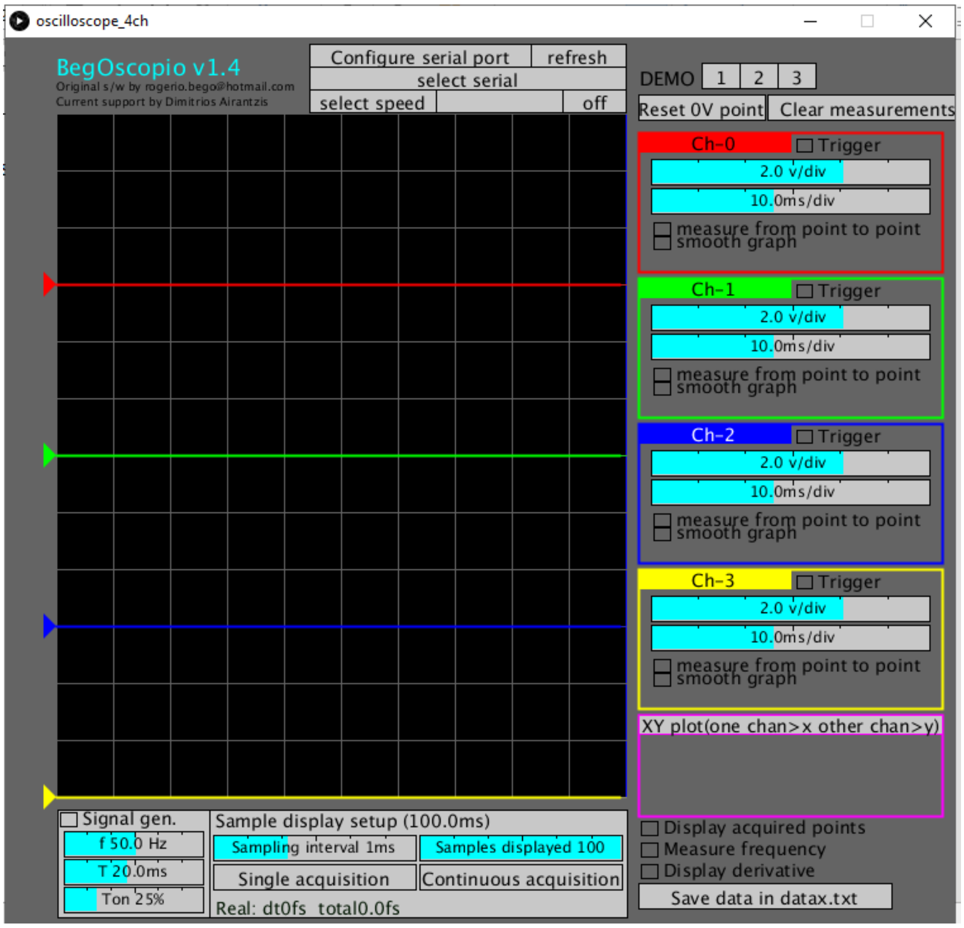
Once the Processing environment is running, go to File → Open and navigate to the subfolder:

...\Arduino as an oscilloscope\processing\oscilloscope_4ch

and select the file:

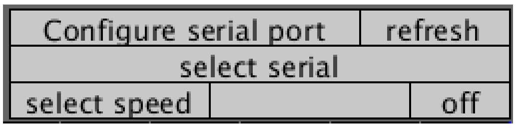
oscilloscope_4ch.pde

Once the file is loaded onto the environment, press the  button to execute the sketch. You should now see the oscilloscope screen:

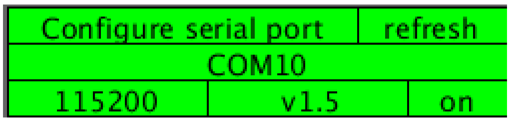


Connecting BegOscopio to the Arduino UNO

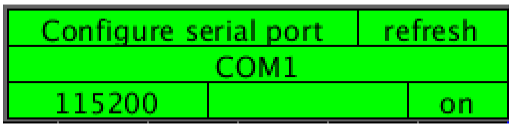
BegOscopio is connected to the Arduino via a serial port that is setup through this frame:



1. Click on the **refresh** box for BegOscopio to retrieve the available serial ports.
2. Click on the **select serial** box as many times as necessary until you see the Arduino serial port being displayed within the box.
3. Click on the **select speed** box as many times as necessary until you see 115200.
4. Click on the **off** box for the serial link to be activated.
5. The **Configure serial port** frame will change its colour to green and the bottom middle box of the frame will show the software version running on the Arduino:



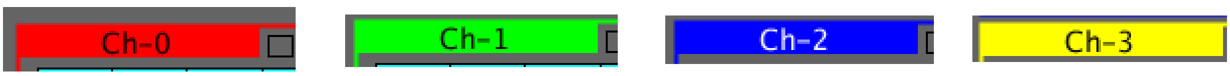
6. If the wrong serial port was selected, the frame will look similar to this:



where the bottom middle box of the frame will be empty. In this case, click on the on **box** of the frame. BegOscopio will release the serial port, the frame will turn grey and you will return to step 1, above.

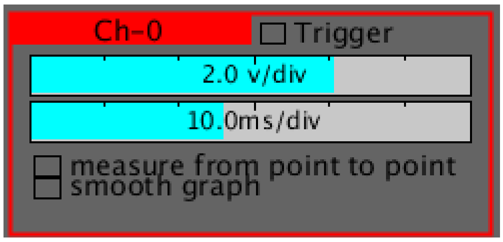
Measurement channels and tools

There are up to four measurement channels available, marked 0 to 3 with colours red, green, blue and yellow. The choice of colours cannot be changed. Channels are activated /de-activated by clicking **within** the coloured area showing their number:



Note: De-activating a channel releases the corresponding screen area, but does not release the corresponding measurement bandwidth. The Arduino keeps collecting and relaying the measurements, but they are not displayed.

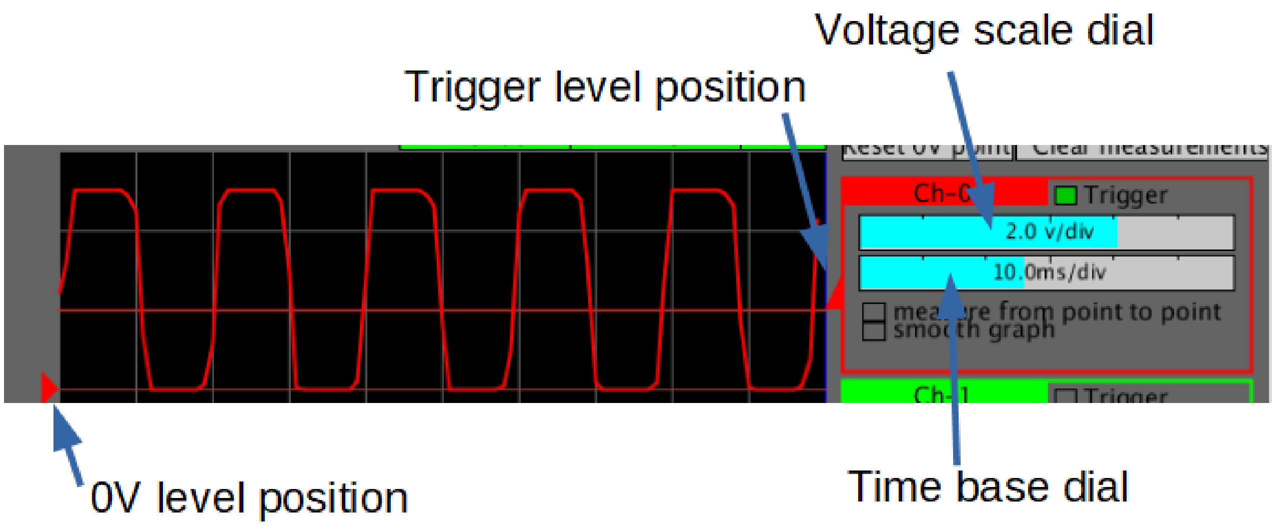
Each measurement channel is controlled by the dials and checkboxes in the corresponding control frame:



Trigger checkbox: when checked, the specific channel is monitored by the software to determine when the trigger point was crossed and thus synchronise the display to the crossing event. A trigger level line and triangle are automatically displayed. **Only one channel can act as the trigger source. Because triggering takes place at the PC level, it needs time to lock onto the incoming measurements stream.**

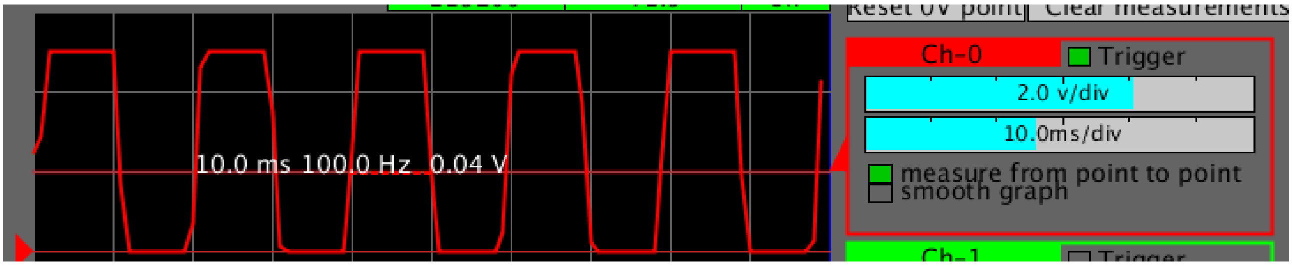
Trigger level is set by adjusting the position of the trigger level line. Click and hold the left mouse button, on the trigger level triangle that is shown at the right hand side of the channel display frame. Release the left mouse button when the trigger line is set at the required level.

0V level position: similar to the trigger level line, adjust the 0V level line by setting the position of the triangle seen at the left hand side of the channel display frame.



Measurement tool checkbox. When checked, the user may position the cursor at one point, left click, position the cursor at another point, left click and release the mouse button. The programme reports the time voltage difference between the two points. In the following example, the cursor used to measure the voltage and time difference between the two trigger crossing points.

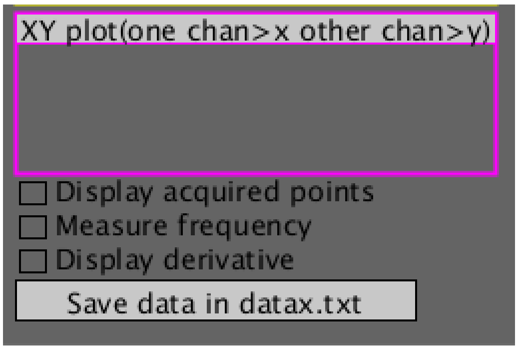
Press the **Clear measurements** button to erase the result of the measurement.



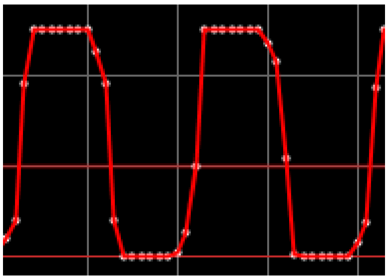
Press the **Reset 0V point** button to set the 0V level position at the bottom of the display area of the specific channel.

Smooth graph tool checkbox. When checked, the programme attempts to smooth the display of the incoming measurements. The algorithm used may result in slight display distortion.

At the bottom of the **Measurement channel** frame, under the XY plot frame, there are four tools common to all four channels:

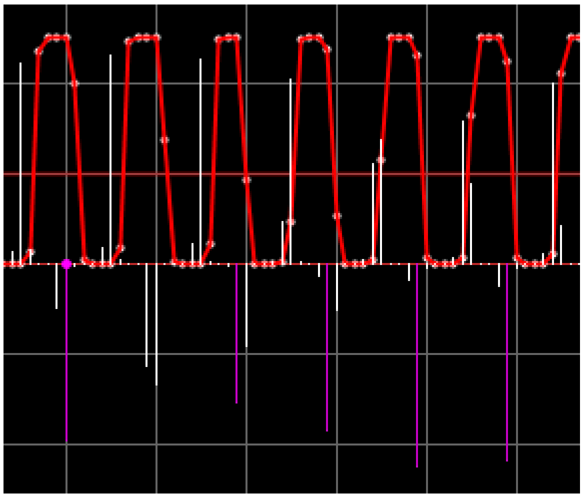


Display acquired points checkbox: when checked, a series of small dots is added to each channel graph. Each small dot shows the position of the actual measurement taken, corresponding to that point in time.



Measure frequency checkbox: when checked, the programme determines the maxima of each channel graph, measures their time difference and uses the result to determine the frequency of the displayed signal.

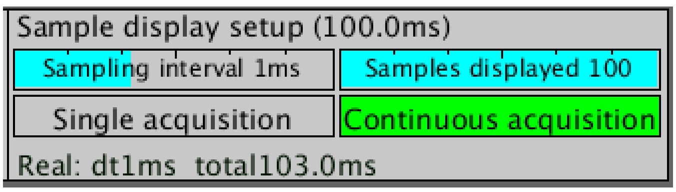
Display derivative checkbox: when checked, the programme calculates the voltage difference between successive measurements and displays the result as a series of small bars displayed at the bottom of the display area of each measurement channel.



Save data button: left click to activate the logging of the incoming measurement stream onto a text file at the same directory where the Processing sketch is located. The file is given the name: dataf<time stamp>.txt where timestamp is made of the year, month, date, hour, minute and second information of the time when the recording started. Each measurement set is saved in a separate line containing the time since the beginning of the log, followed by the voltage measured in each active channel, in mV.

Sampling tools

The **Sample display setup** frame is found below the measurement display, at the centre position of the oscilloscope screen.



Acquisition mode: if a **single acquisition** is required, left click on the **Single acquisition** button. If **Trigger is active** a single set of measurements is taken as soon as the trigger point is crossed. If **Trigger is inactive**, a single set of measurements is taken immediately.

If continuous acquisition is required, left click on the **Continuous acquisition** button. The microcontroller measures the voltages seen at the four analogue inputs and relays the data to the PC continuously.

Note: there is no automatic selection of the acquisition mode. The user needs to select either modes manually. **The display is unstable for as long as no selection is made.**

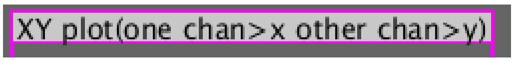
Sampling interval: adjust the Sampling interval dial to set the time between each measurement set is taken. All four channels are measured irrespective of whether their display is active, or not. Sampling interval may be set between 10 µsec and 2 sec.

Samples displayed: sets how many of the incoming measurements are displayed within the graph area.

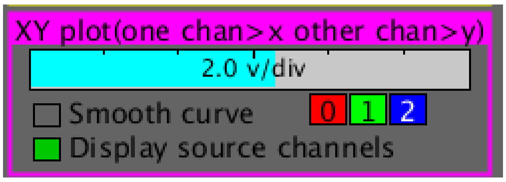
The programme recalculates and displays the time period actually displayed within the graph area.

XY plot

Clicking on the frame titled XY plot, activates the XY plot, alongside the plots of the signal channels.



In XY plot, a fifth graph is generated in which one channel (eg channel 0) controls the values of the x-axis of the graph, while another channel (eg channel 1) controls the values of the y-axis.



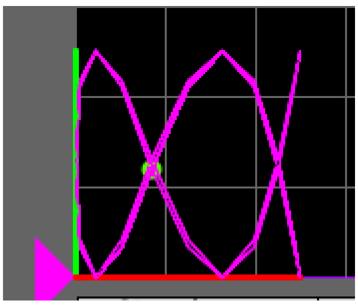
Three checkboxes are activated, each being responsible for the selection of the corresponding axis: x, y and z. The z axis is a modulator of the length of the vector defined by the values of the channels controlling the x and y axes. Click repeatedly on each of the coloured checkboxes to scan through the available signal channels.

Smooth graph tool checkbox. When checked, the programme attempts to smooth the display of the incoming measurements. The algorithm used may result in slight display distortion.

Display source channels checkbox. When checked, the channels controlling the x, y and z axes are also displayed together with the XY plot.

The scale of the XY graph can be adjusted by using the dial found at the top of the XY plot frame.

XY plots are useful when investigating the relationship between two signals. The most common type is the one that results in Lissajous curves, used to investigate the relationship between two sine waves. An example is shown below.



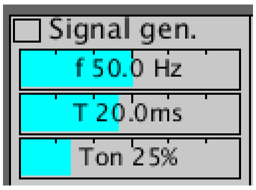
For more information on Lissajous curves, you may visit:

https://en.wikipedia.org/wiki/Lissajous_curve

Signal generator

The **Signal generator** frame is found below the measurement display, at the left position of the oscilloscope screen.

Signal gen. checkbox: when checked, the microcontroller generates a square wave at output port 9. Either of the **frequency (f)**, or the **period (T)** dials can be used to set it. Dial **Ton** is used to set the **duty cycle** of the square wave. The output frequency ranges from 125 mHz to 10 kHz.



When the signal generator is active, a second square wave is generated at output port 10. Its **frequency** is **half** of that of the one generated at output port 9 and its **duty cycle** is set to **50%**. If a low pass filter is connected to output port 10, the resulting signal can be made to approximate that of a sine wave.

Signal connections

Analogue ports 0 to 3 correspond to the oscilloscope channels 0 to 3. Square wave outputs are provided at output ports 9 (F with variable duty cycle) and 10 (F/2 with 50% duty cycle).