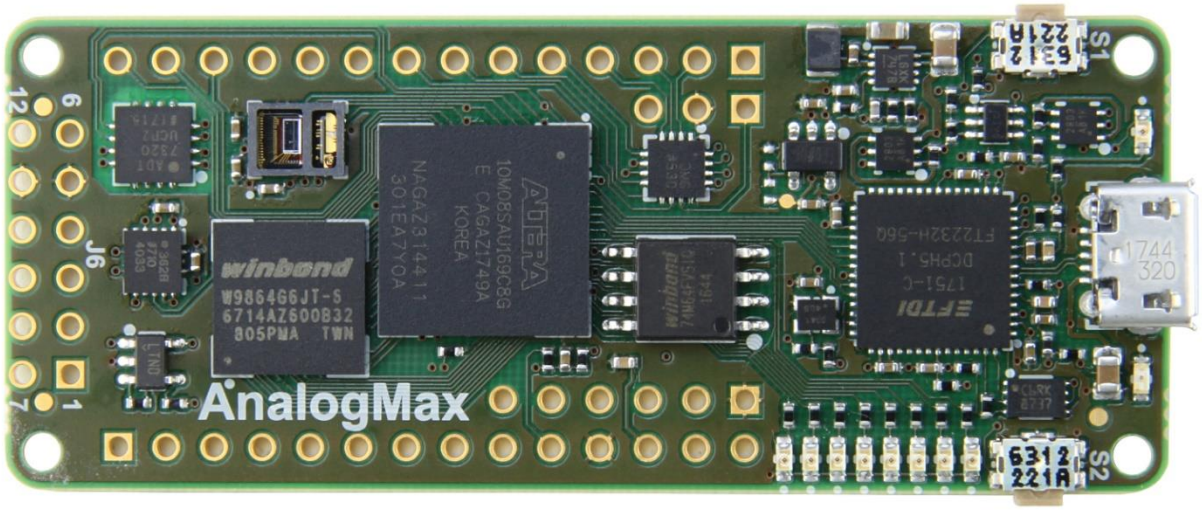
AnalogMax with Jupyter

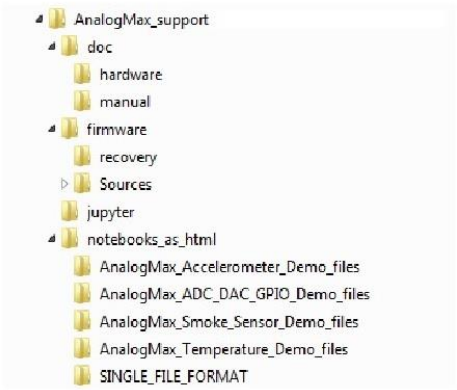
Revision 1.0.2
23 October 2018

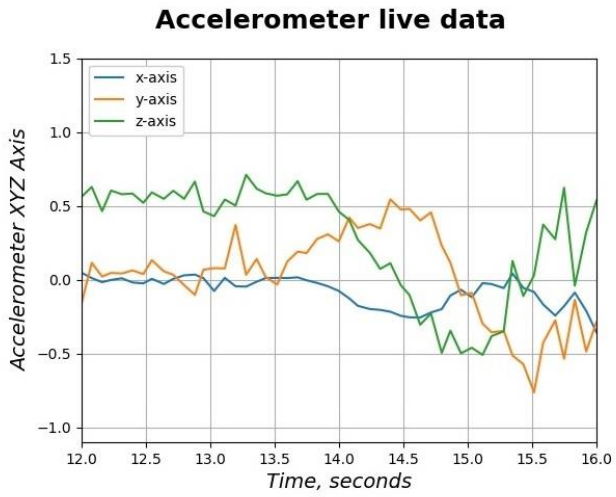


Installation and setup

Quick and easy installation guide given, advanced users can use different methods.

Installation Files





Distribution archive, included are some Jupyter notebooks and POF file for the AnalogMax FPGA.

Jupyter via Anaconda

The simplest way to get the demos running in *Jupyter* is accomplished throw the installed of *Anaconda*, this will install more than is needed but *Jupyter* is also installed without problems or much manual setup. So running and editing the Demos requires the following steps:

- download and installation of *Anaconda*
- installation of *pyserial* throw Anacondas command prompt
- driver installation for the *AnalogMax* board
- adding the demos to *Jupyter*

The following description of steps applies in its details to computers running windows, for other operation systems they are in general similar.

Installation of Anaconda

The Anaconda website provides detailed instructions on how to install the application, just follow the link http://docs.anaconda.com/anaconda/install/windows/ .
The instructions and the installer of Anaconda offer you the installation of optional application, they are not needed to run the demos or edit them.

Installation of Pyserial

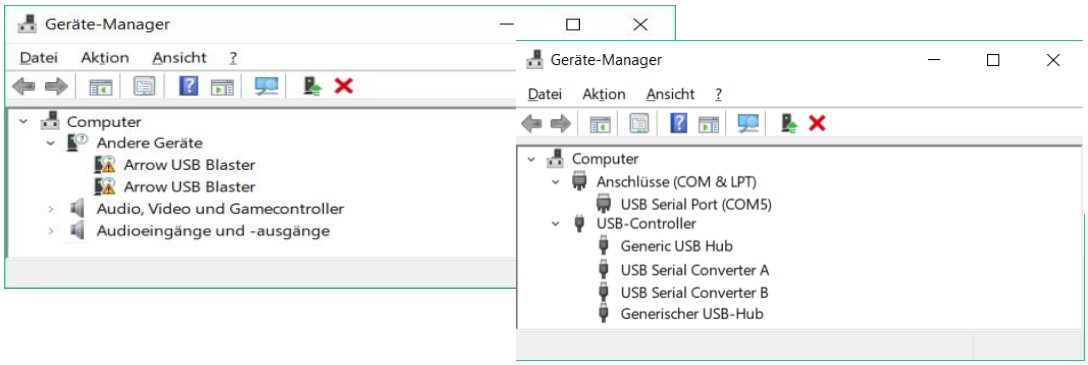
At least with *Anaconda* installation on Windows, *pyserial* (python serial port support library) is not included and hast to be installed. This can be done from Anaconda command prompt.
To open the Anaconda command prompt, just press the *windows key* and type *Anaconda prompt.* Enter

```
conda install -c anaconda pyserial
```

into the command prompt. This starts the search for software dependencies of pyserial, you are asked to proceed with the installation off these. In the end *pyserial* is installed and the command prompt is no longer needed.

Driver installation

When you connect the *AnalogMax* board to your computer, windows starts the automatic driver installation. If this is not the case, the manual approach is to open the device manager, *windows key* and type device manager. It lists two unknown devices *Arrow USB Blaster*.



Through a right mouse click on each device, the search for drivers can be started. Just follow the steps and the installation of the drivers changes the devices name into USB Serial Converter A & B. In addition, also a comport is installed.

Check its number (“COMX”) and memorize it for the later use within the AnalogMax demos.

Adding of the demos to jupyter

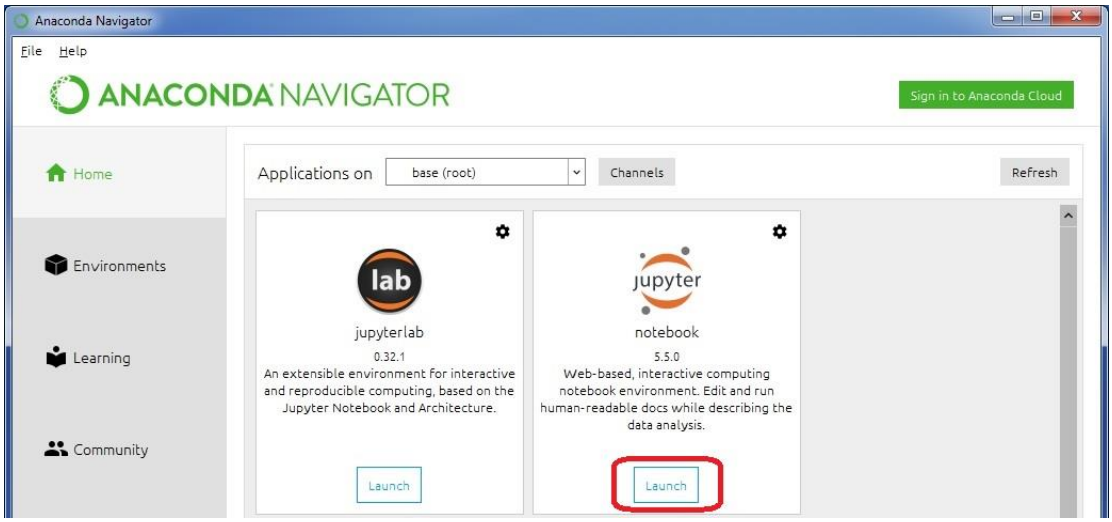
A convenient way to add the demos into *Jupyter* is to copy the folder from the *AnalogMax distribution archive* to your user folder, for example

```
C:\Users\Username
```

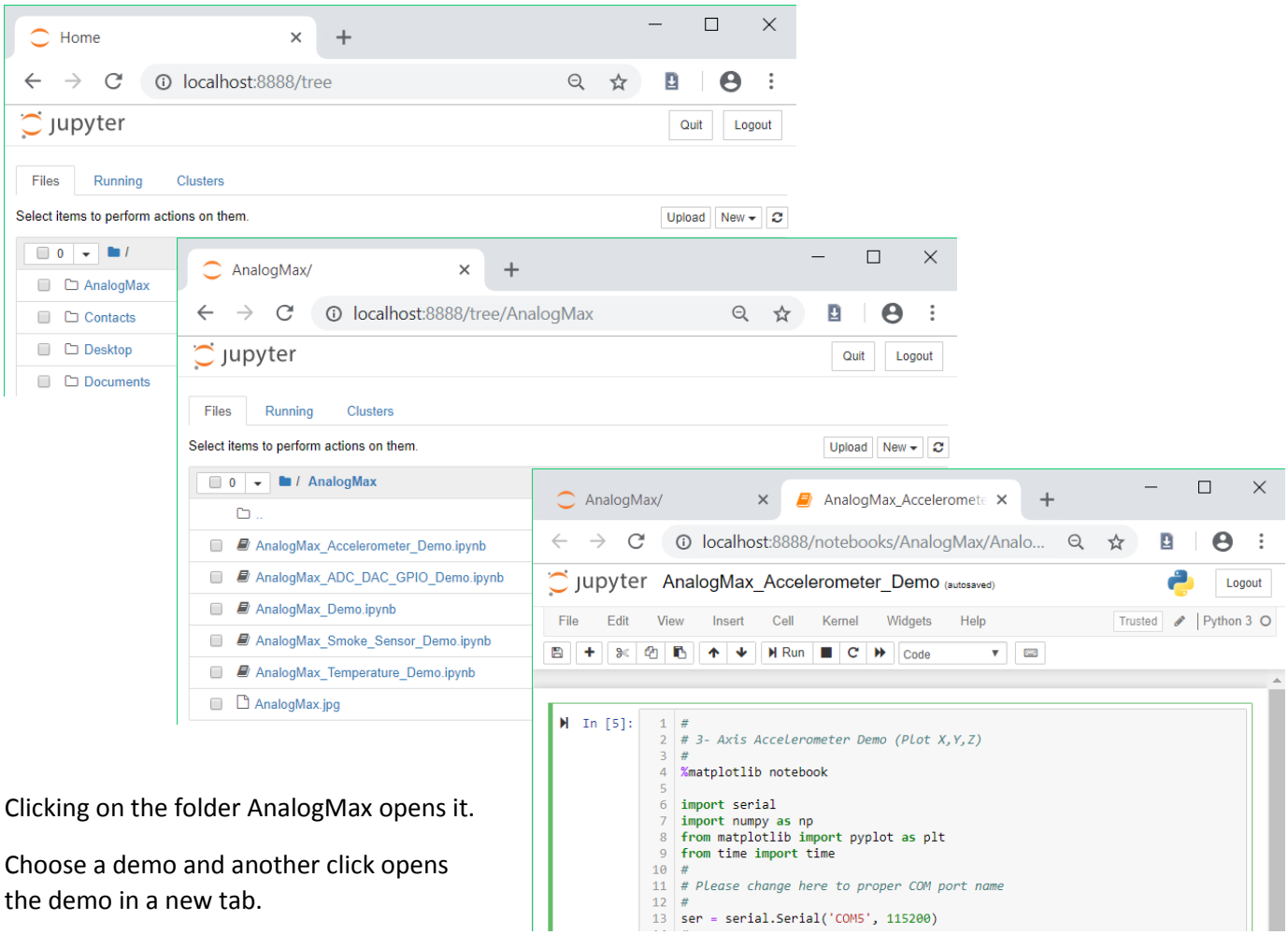
before you start *jupyter*.

Opening the demos in jupyter

Jupyter is accessible through the *Anaconda Navigator*, press the *windows key and type Anaconda Navigator.* Anaconda opens and to open jupyter, click launch, in the screenshot below, marked within the red rectangle.



The application opens a tab inside your browser which displays your user respectively home folder.



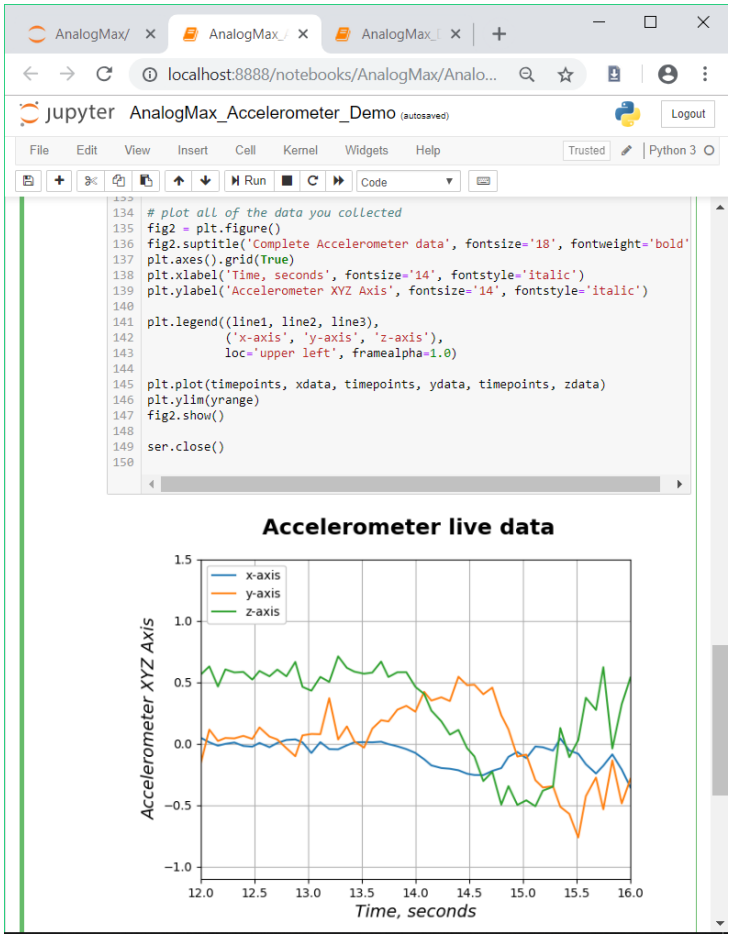
Clicking on the folder AnalogMax opens it.

Choose a demo and another click opens the demo in a new tab.

Demos

Currently there are four demos included:

- AnalogMax_Accelerometer_Demo
- AnalogMax_Smoke_Sensor_Demo
- AnalogMax_Temperature_Demo
- AnalogMax_ADC_DAC_GPIO_Demo



Each demo consists of 2 section.

- The code section, which contains the functionality of the demo and is simultaneously a python code editor so that you easily can implement your own ideas
- The graphs section, which displays the raw data during script execution and a second graph, which displays all captured data after the script finishes.

In jupyter, files are called notebooks. Common to all demo notebooks is that, at the very beginning, there is a line that selects the COM port in use.

```
#
# Please change here to proper COM port name
#
ser = serial.Serial('COM5', 115200)
```

The comport, here 'Com5', has to be change to the one, you got during the driver installation of the AnalogMax board.

To run the demo, make sure, that the mouse curser is placed inside the python code editor. An indicator for this, can be seen in the screenshot above, the green line on the left.
Clicking the Run" button on the toolbar starts the demo. During the running time, you have time to interact with the sensor.

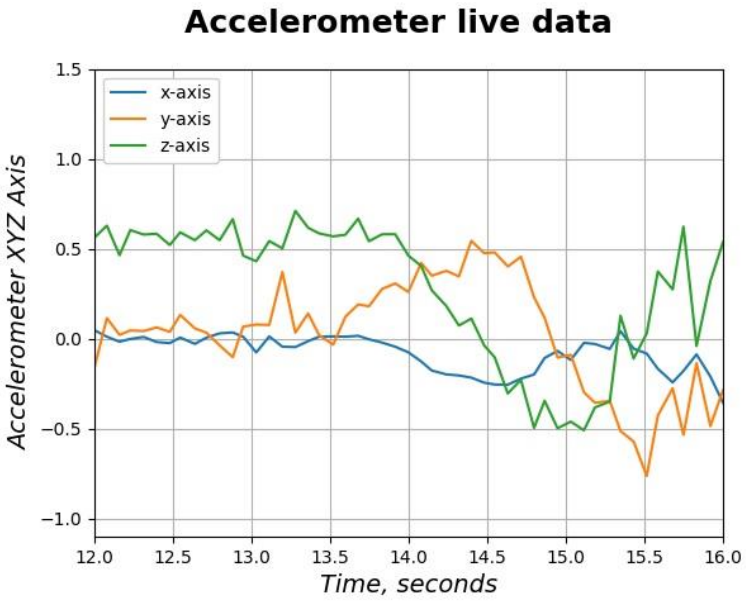
The accelerometer demo

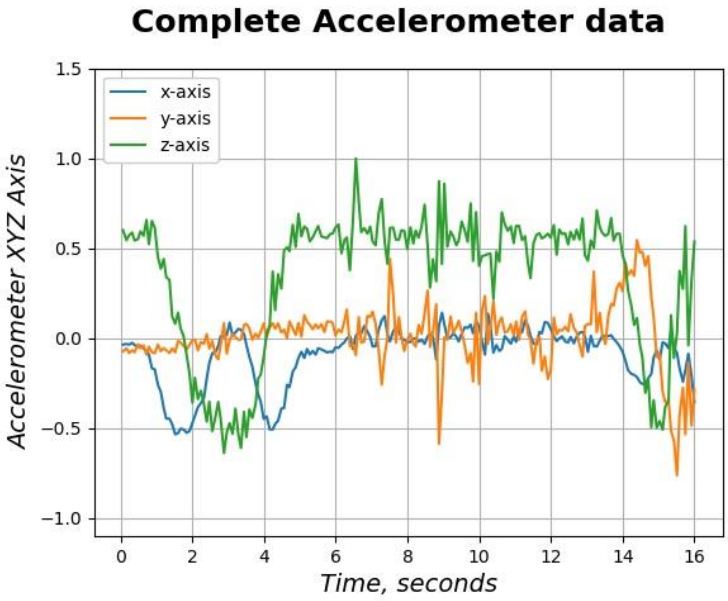
The accelerometer demo shows the gravitational acceleration.

If all was setup properly you should see collected Accelerometer data. Make sure you move the AnalogMax board to make the chart more interesting.

A long USB cable is helpful when interacting with it.

As a suggestion, you can normalized it to show g = 9.8 m/s².





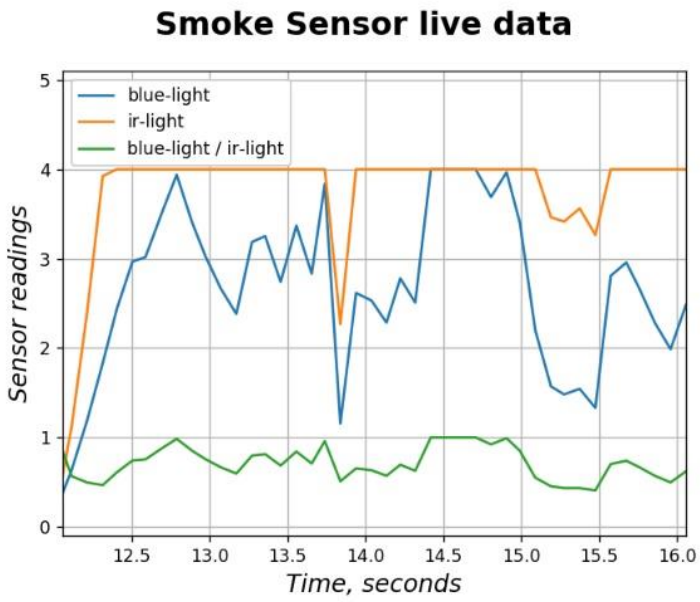
The smoke sensor demo

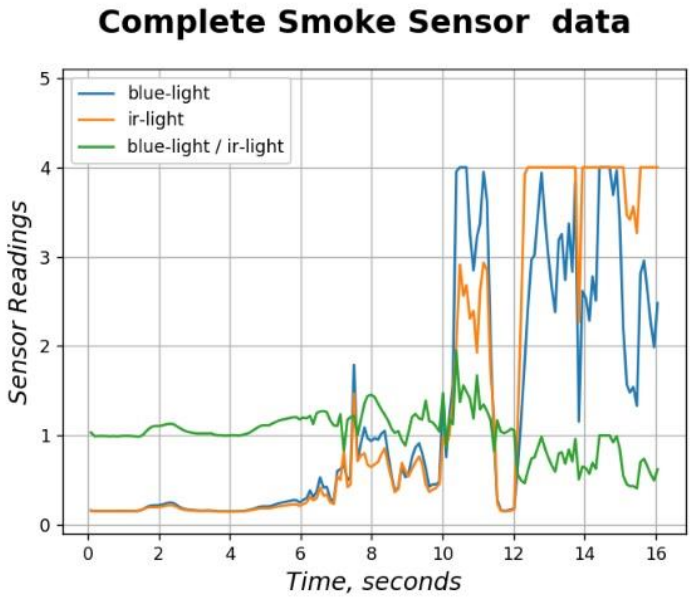
This sensor emits light of two different light spectra, the infrared and visible spectrum (blue light).

As a means to interact with the sensor, use your breath, the gas of a not lighted lighter or any convenient and safe source of gas or smoke.

Also, different material can be placed above the sensor.

It should be noted, that the sensor maxes out on reading greater than 4.

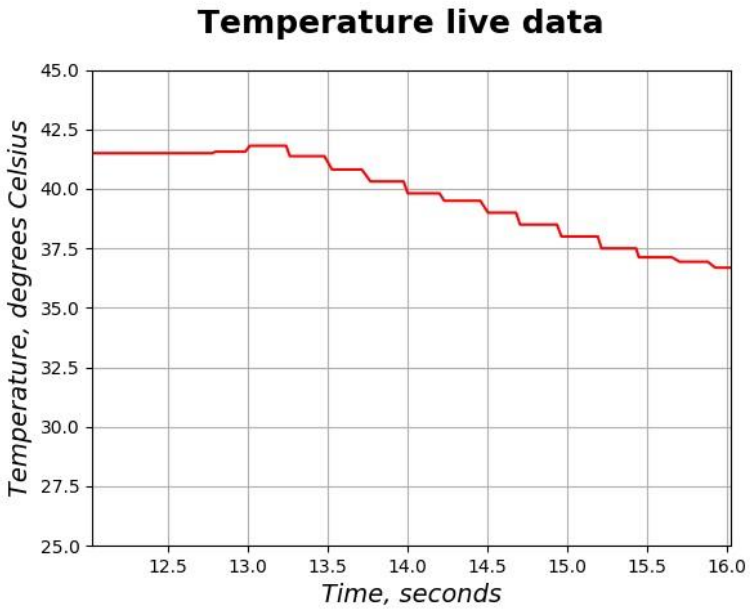


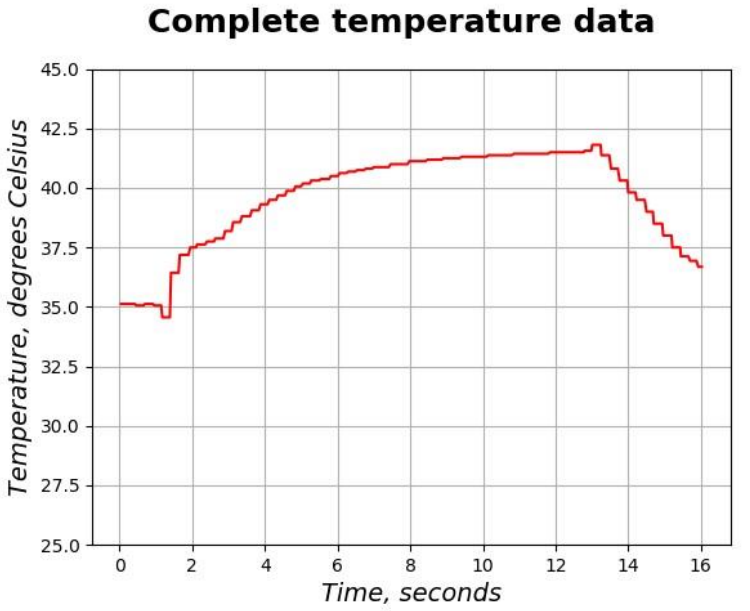


The temperature demo

Temperature is very slow changing but if you touch the ADT7320 sensor with the fingers you will get clearly visible temperature changes.

The graphs have been obtained by placing a lightly warmed soldering iron on top the sensor.

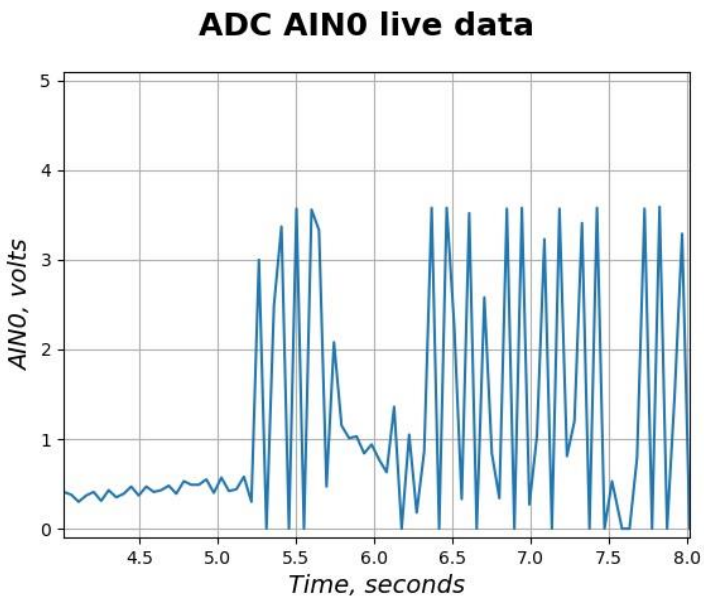


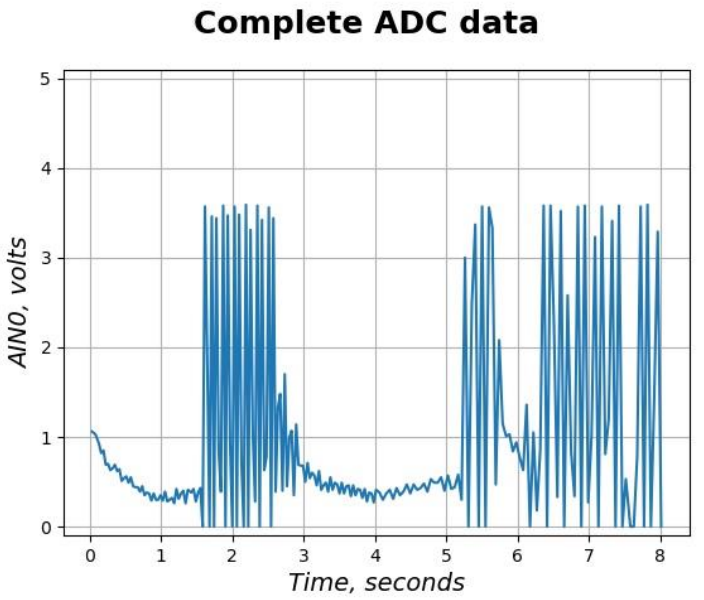


The adc/dac/gpio demo

This demo for AD559R does configure IO7 as DAC with midscale output, IO0 as Analog Input, IO1 as digital Input and IO2...IO6 as digital outputs with high value.

Then the AIN0 input pin sampled and Analog voltage displayed on screen, the signal visible is from the finger touching the PCB at the AIN0 pin via a 20kΩ resistor.





Econo Oscillator Demo

Not included in this firmware release.

PMOD Support

Not included in this firmware release

UHSA support

UHSA protocol is itself in stage of being defined, the version implemented currently has really minimal features that are sufficient to demonstrate the sensor evaluation on AnalogMax.

POF image for the FPGA includes a simple WIP UHSA protocol handler implemented purely in NIOS C code and using the standard UART at 115200 baud. There is no support for real-time capture in this version.

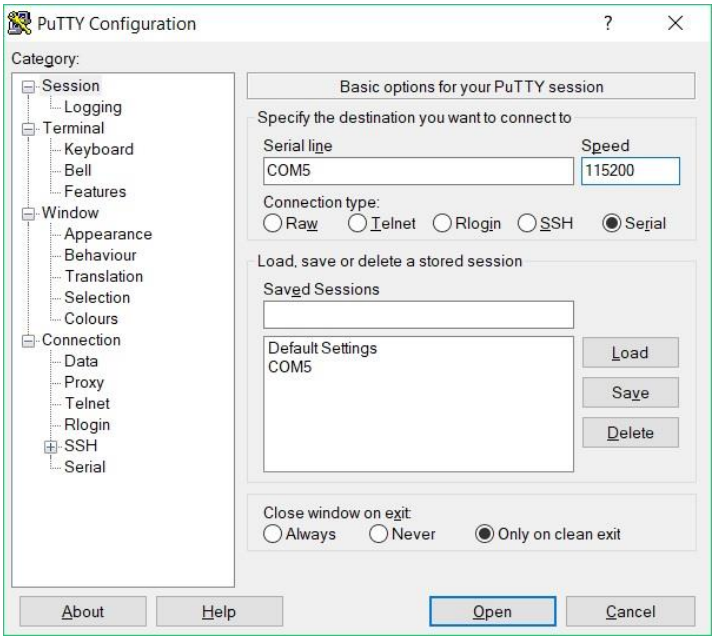
Documentation

All the notebooks have been saved as HTML those files can be viewed without the need to install jupyter (please enable scripts to see the charts in the HTML files).

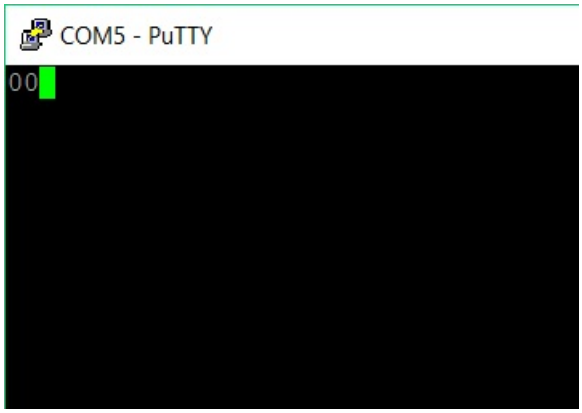
Troubleshooting

If there is an error in python code or you unplug the USB cable while the notebook is running, the serial port will not be properly closed. To “free” the serial port, the python “kernel” has to be restarted from the “notebooks” menu.

UART Connection to AnalogMax can be tested with any UART terminal (as example putty) Open a serial connection to the com port, the AnalogMax board is connected to.



The speed needs to be 115200. If you copy the command *<0b0f..>* and insert it via a right mouse click into the terminal, you should get to see the answer *00*.



History

Version 1.0.2

Updated all demos.

Version 1.0.1

Added Smoke Sensor support.

Version 1.0

Initial release, support for sensors connected to the SPI bus, I^2C bus support not included. Python code is much WIP in every sense, major code cleanup is required.