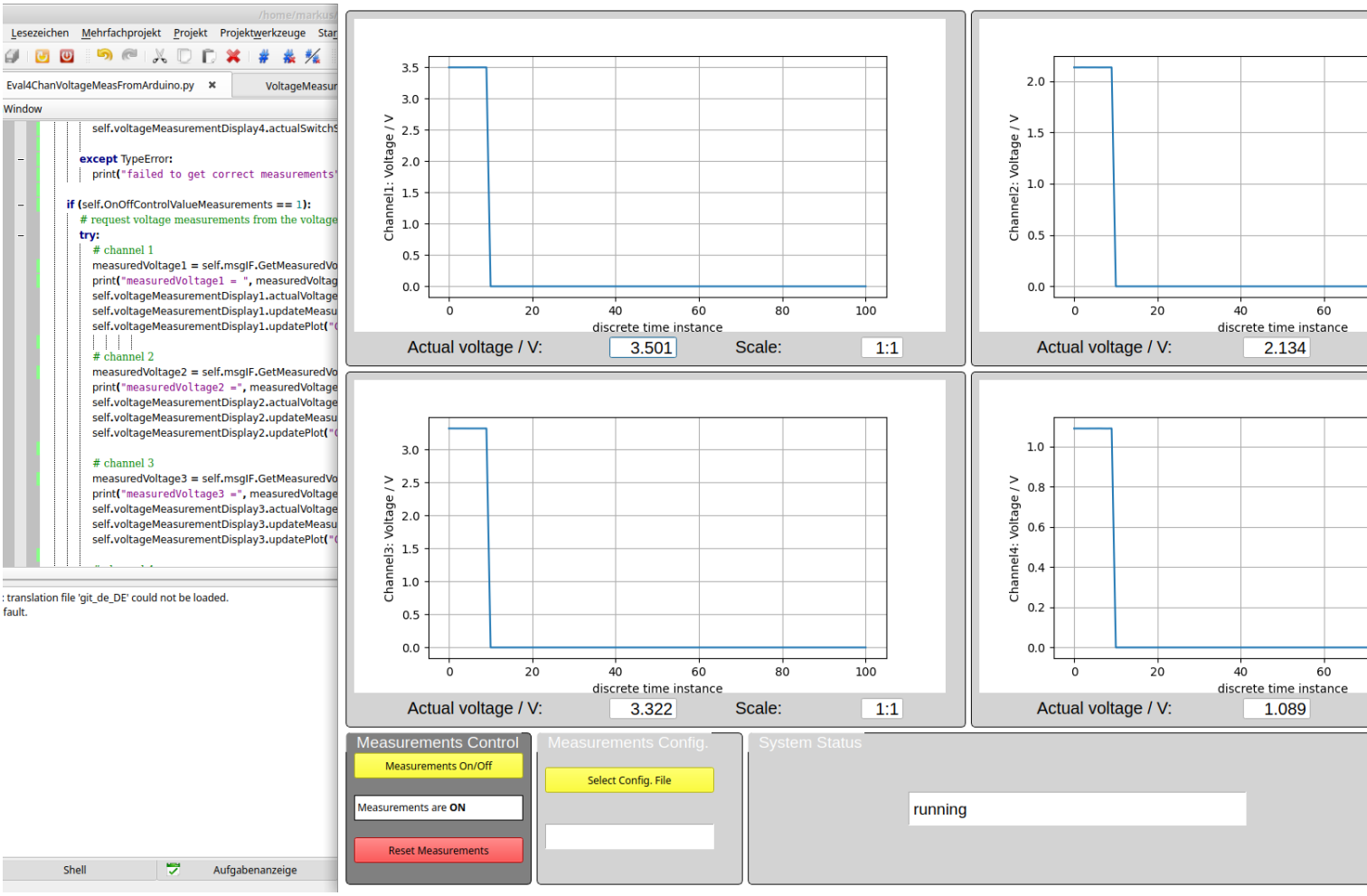
My Simple 4-Channel DAS

Description and Evaluation

by Dr. Markus Reinhardt

July 16, 2021



Contents

List of Figures

List of Tables

1 Concept description

A project to build-up a simple 4-channel low sample rate, high accuracy (16-bit ADC) Arduino based Data Acquisition System (DAS) to support general PC controlled universal measurement campaigns.

The signals to be measured are fed to a 1:10 divider or via a 1:1 connection to the buffer amplifier of each channel. The signals are then ADC converted by the 4 channel ADS1115 ADC on the ADC module. The Arduino Uno is controlling the ADS1115 and receiving the digitized signals via the I2C interface. The data is then sent to the PC via the SoftwareSerial port, a USB/Serial module using the CmdMessenger library to the PC. On the PC a Python (PyQt) or C++ (Qt) program using also the (Py)CmdMessenger library is receiving, evaluating and displaying the data.
HW components:

- Arduino Uno
- USB/Serial module to connect the PC with the Uno via SoftwareSerial interface and USB connector.
- 4-Channel ADS1115 16-bit ADC module.
- 4-Channel 1x buffer amplifier board, LM324 based.
- 10x / 1x switchable inputs and resistor divider
- Simple plastic case

SW components:

- Arduino sketch to control the ADC module and send the data via the serial interface to the PC.
- C++ / Qt based software library to sample, store and evaluate data from up to 4 channels.
- PyQt based Python program running on the PC to receive and store data from up to 4 channels.
- First SW example: Tracing the charging characteristic (voltage and current) of a LiPo cell.

2 Schematics

The schematics are shown in figure 1. The input section for each channel consists of the

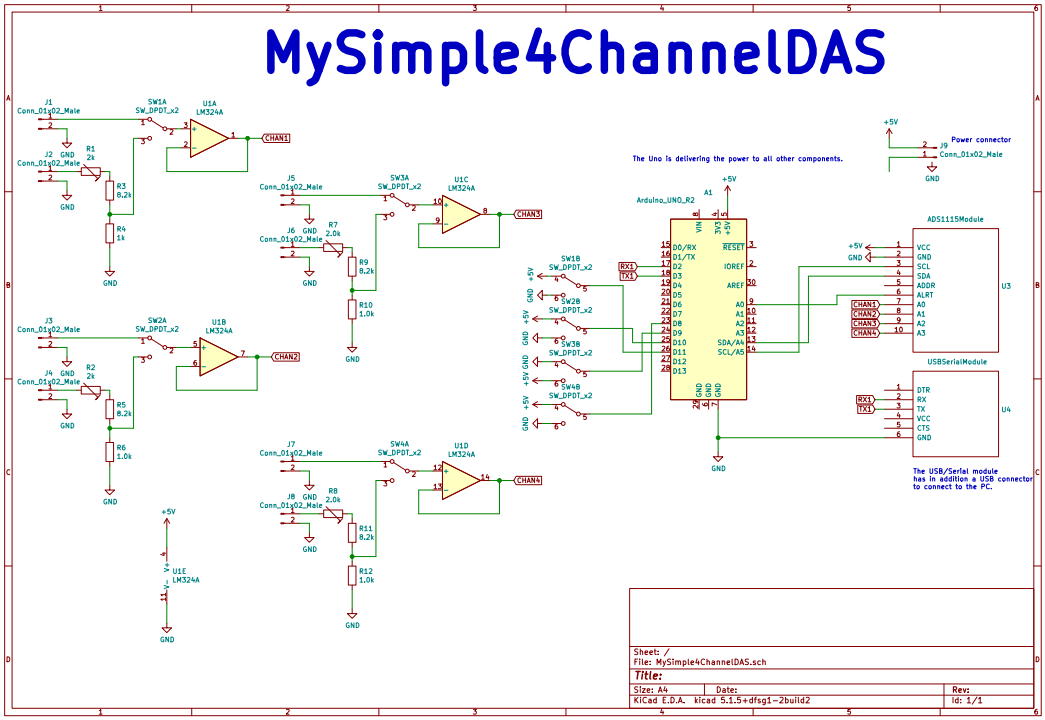


Figure 1: Schematics

connectors for the 1:1 and the 1:10 path. In the 1:10 path a 1:10 resistor divider divides the input signal by 10 and feeds the attenuated signal to the switch that selects the input path. After the switch the signal is fed to the buffer amplifier with a gain factor of 1.0 and from there to one of the four channels of the ADC module. The ADC module is controlled from an Arduino Uno that provides also the interface to the PC via a USB/Serial module.

3 Hardware setup

The four channel DAS front view of the case with the sockets and the switches four each of the four channels is shown in figure 2.
The internal HW is shown in figure 3. It shows the Arduino Uno board, the ADC board, the USB/Serial converter module, the buffer amplifier (LM324) all mounted on a bread board. There are twelve 4mm connectors assembled on the front panel, four for ground

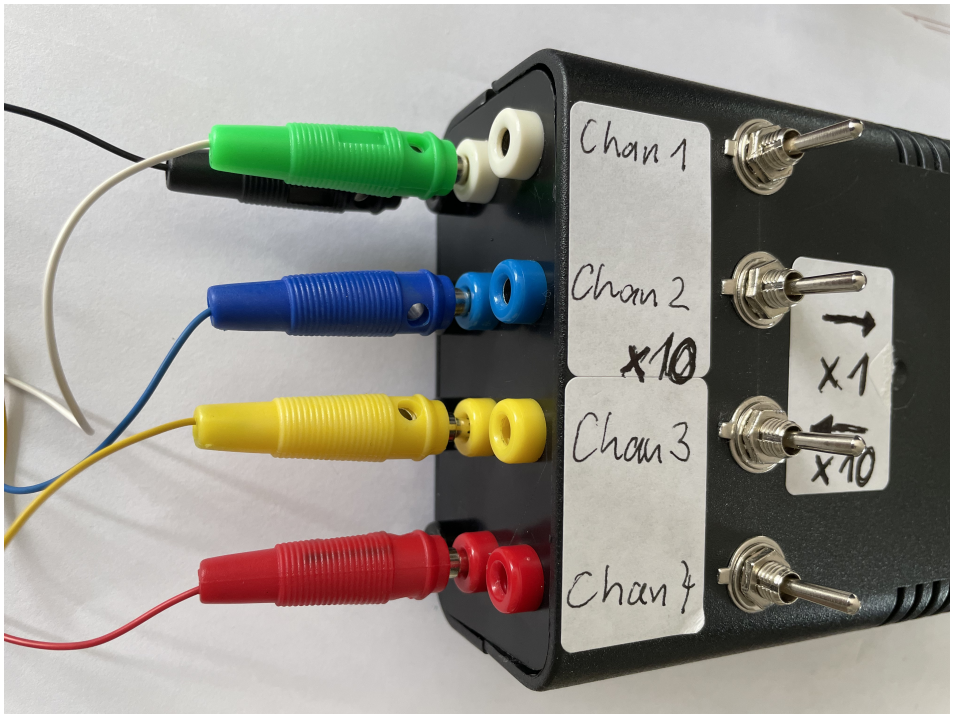


Figure 2: Four channel DAS front view of case)

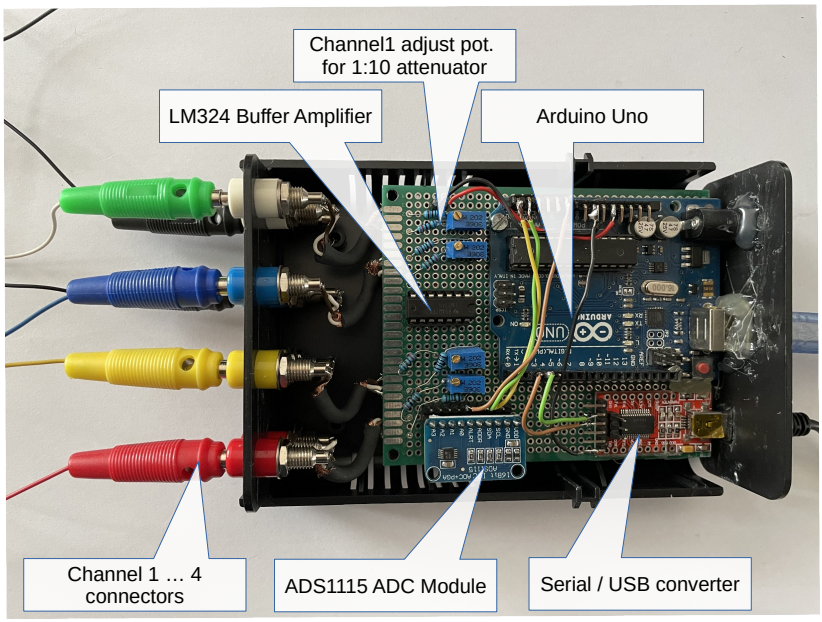


Figure 3: HW inside the case

connections, four for buffered signals and four for(by a factor of 1:10) attenuated and also buffered signals.
The used ADC module is shown in figure 4.

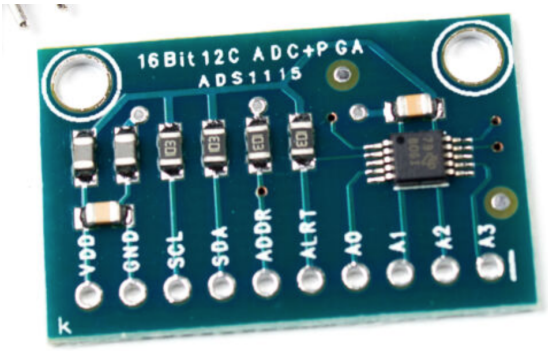


Figure 4: ADC module with ADS1115

4 Software

4.1 Arduino Sketch

The Arduino sketch is running on an Arduino Uno. It consists of 3 tasks.
The first task is the blinker task that blinks the on-board LED, i. e. it is working as an alive-signal.
The second task is the communication task that receives request/commands from the PC via the CmdMessenger Library and sends on request the voltage measurements back to the PC program. The CmdMessenger interface uses the SoftwareSerial library for the communication via a USB/Serial module to the PC. Arduino digital pins 2 (RX) and 3 (TX) are used by the SoftwareSerial library.
The third task is the measurement task that handles the I2C interface to the ADC module to get the measurement values. The standard I2C pins of the Arduino Uno are used here, i. e. Pins A4/SDA and A5/SCL). Pin A0 is used for the ADC module's ALERT pin interfacing.
The measured voltages are also printed to the standard Arduino Serial port (/dev/tty-ACM0) and can be watched with the Arduino Serial Monitor of the IDE. The trace of the measured voltages in the Arduino IDE Serial Monitor after the program setup is done is shown in figure 5.

```
01:24:37.178 -> Setup in MeasurementsTask task started
01:24:37.212 -> ADS1115 setup done
01:24:37.245 ->
01:24:37.245 -> Setup in MeasurementsTask task done
01:24:37.278 -> Setup in communications task started
01:24:37.311 -> Setup in communications task done
01:24:37.344 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:38.141 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:39.137 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:40.166 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:41.162 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:42.159 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:43.155 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:44.151 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:45.145 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
01:24:46.141 -> 0: 0.54,   1: 0.54,   2: 0.54,   3: 1.27
```

Figure 5: Arduino Serial Monitor: Trace of the four measured voltages

4.2 PC Application

The GUI of the Python PC application together with the Eric6 IDE is shown in figure 6 after around 10 voltage samples have been processed for all four channels. The program

is called by the shell script Eval4ChannelVoltageMeasurements which calls the python interpreter to execute the python program Eval4ChanVoltageMeasFromArduino.py.

The program reads via the PyCmdMessenger library the four measured voltages and the status of the scaling switches from the Arduino board via the USB/Serial module on port /dev/ttyUSB0 and displays the voltages vs. the discrete sample time index in four separate plots. In addition the actual measured voltage values and the voltage scaling values (1:1 or 1:10) chosen by the switches for all channels are displayed below the plots. The program also allows to store the measured data and to control the measurements via start/stop/reset commands in the GUI.

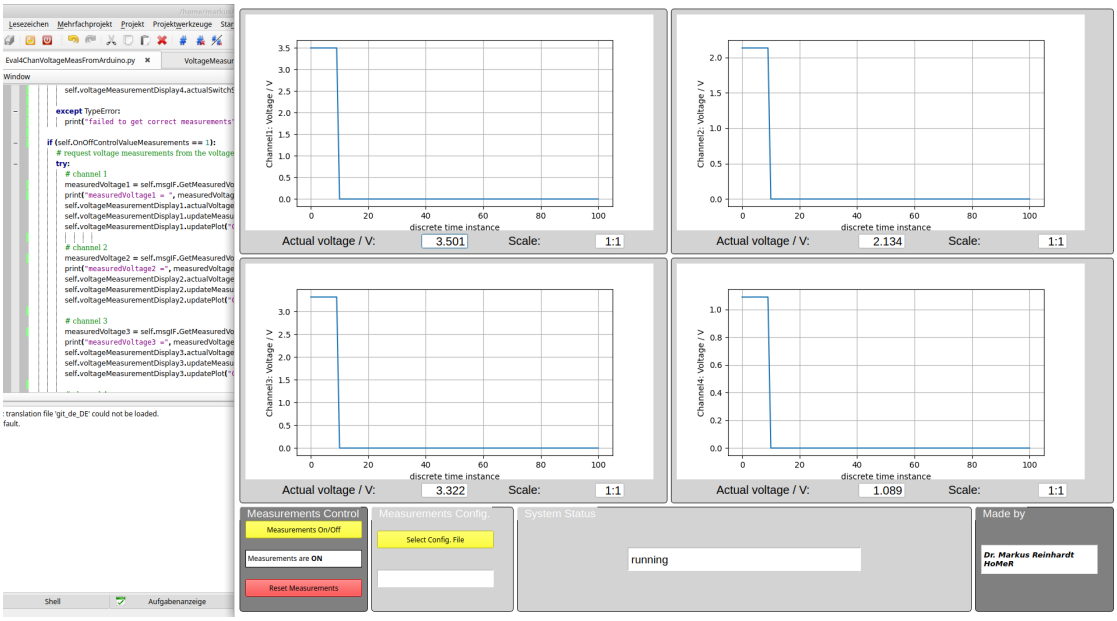


Figure 6: Python PC Application

The program when the scaling of 1:10 is selected for the channels 1 and 3 and the scaling of 1:1 is selected for channels 2 and 4 is shown in figure 7.

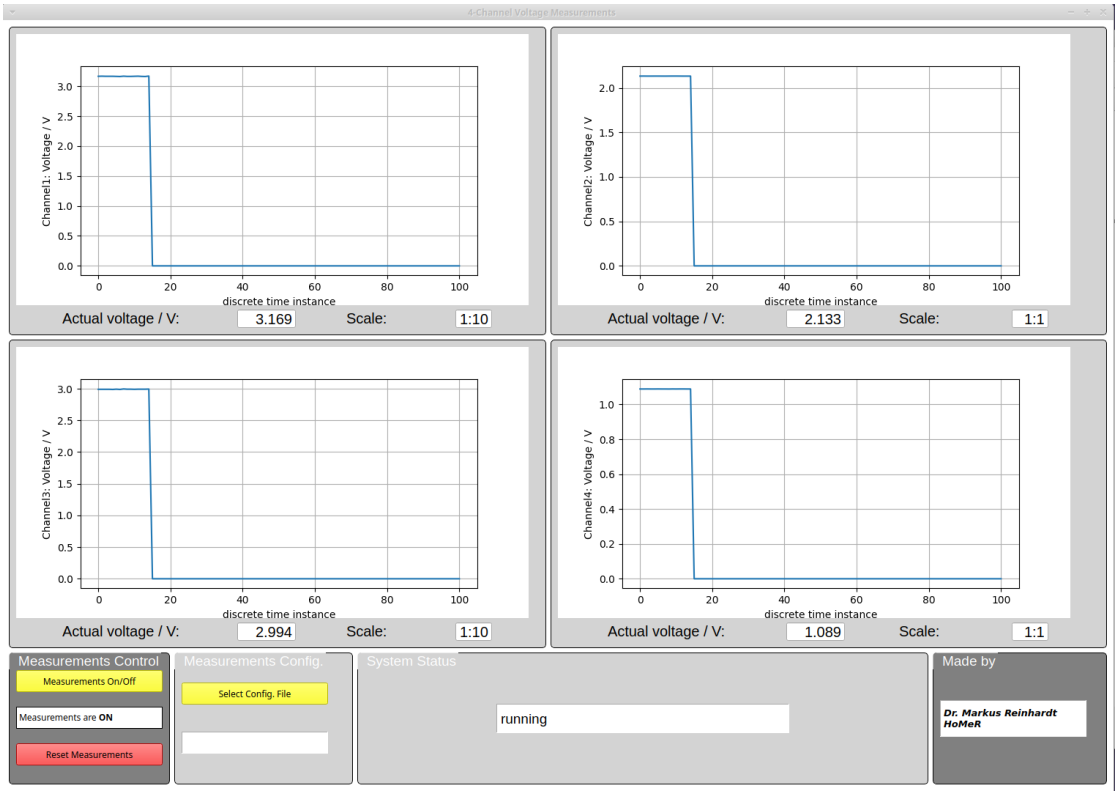


Figure 7: Python PC Application