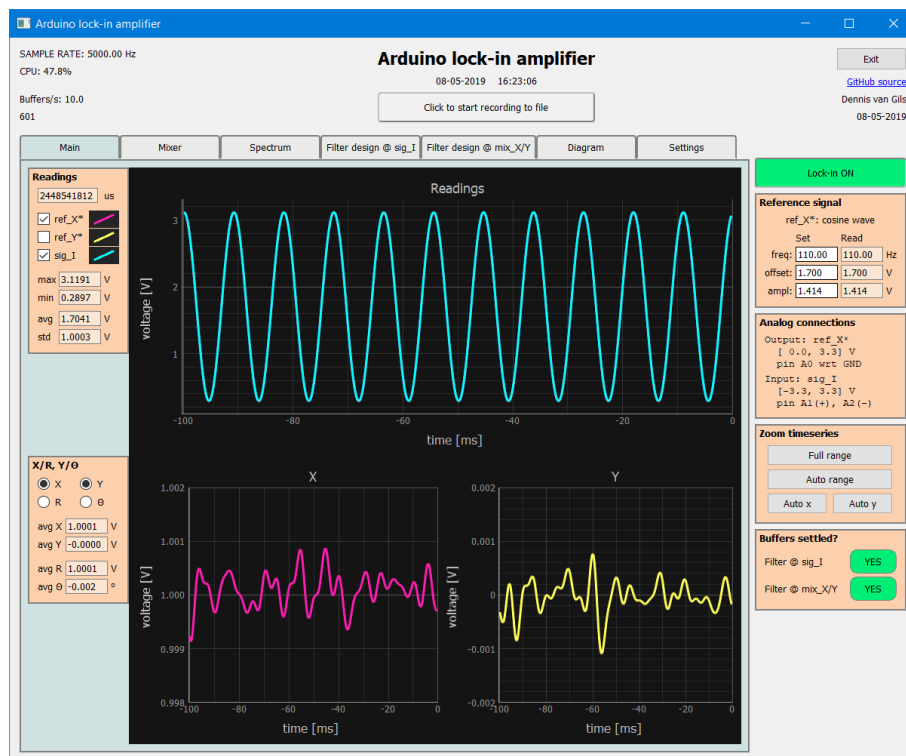
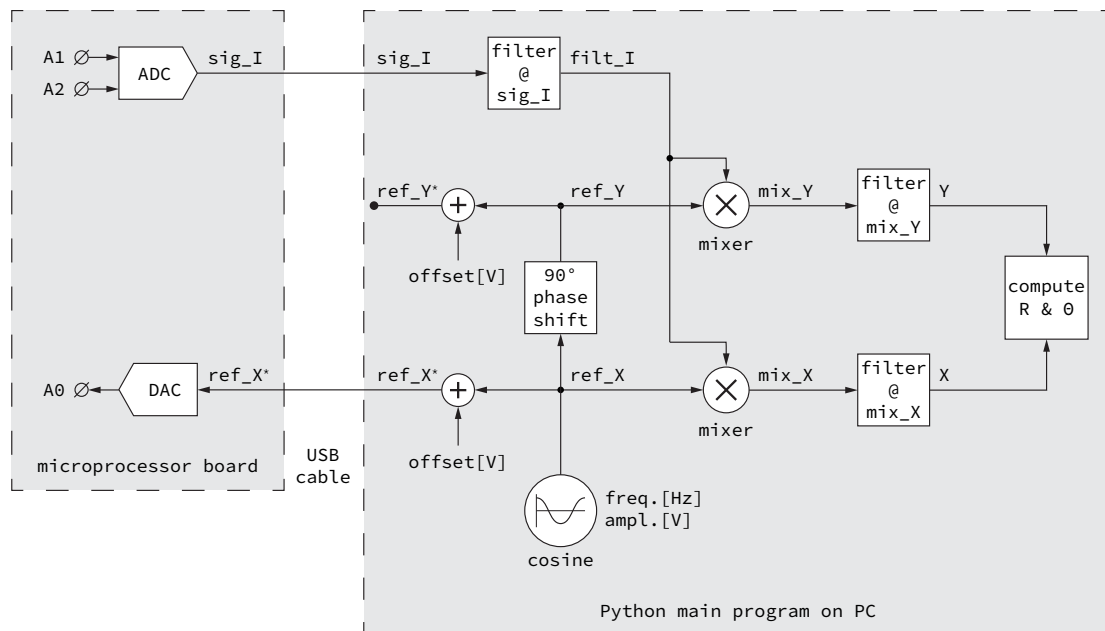


STUDENT USER MANUAL

DvG_Arduino_lock-in_amp

Dr. ir. Dennis P.M. van Gils

May 8, 2019



Contents

1	Introduction	1
2	Software installation	2
2.1	Python distribution	2
2.2	GitHub source	2
2.3	M4 Feather Express drivers and firmware	2
2.4	Running the main program	2
3	Hardware information	3
4	Signal processing	3
5	Python GUI	4

1 Introduction

This document describes a lock-in amplifier running on an Adafruit M4 Feather Express microcontroller board in combination with a PC running Python. It is part of the lab assignments of the course '*Small Signals & Detection*' of the University of Twente, Enschede, The Netherlands.

The microcontroller (MCU) board generates the reference signal **ref_X*** and subsequently acquires the input signal **sig_I**. This data is sent over USB to a PC running the main graphical user interface in Python. The Python program shows the waveform graphs of the signals in real-time, performs the heterodyne mixing and filtering of the signals similar to a lock-in amplifier, and provides logging to disk.

Current specifications MCU:

- Support for Atmel SAMD21 or SAMD51 chipsets
- True analog-out waveform generator (**ref_X*** between 0 to 3.3 V)
- Differential analog-in data acquisition (**sig_I** between -3.3 to 3.3 V)
- The DAC (digital-to-analog converter) and ADC (analog-to-digital converter) operate at 12 bits resolution at a sample rate of 5 kHz
- Double-buffered binary-data transmission over USB to a PC running Python

Current specifications Python:

- Separate computing threads for real-time communication with the MCU, signal processing and graphing
- Automatic detection of the MCU board by scanning over all COM ports
- High-quality zero-phase distortion linear FIR filters
- Optional OpenGL hardware-accelerated graphing
- Data logging to file

2 Software installation

2.1 Python distribution

The preferred Python distribution is Anaconda Python 3.7 which can be found at:
<https://www.anaconda.com/distribution/>

Install Anaconda Python on your laptop. When the installation is finished, start up Anaconda Navigator and install the packages `pyserial`, `pyqtgraph` and `pyopengl` in the environment. Alternatively, the packages can also be installed using the Anaconda Prompt window and entering the following commands:

```
conda install pyserial
conda install pyqtgraph
conda install pyopengl
```

2.2 GitHub source

Download the source files from the GitHub site of Dennis van Gils at:
https://github.com/Dennis-van-Gils/DvG_Arduino_lock-in_amp/

Click on the green button labeled '*Clone or download*' and chose '*Download ZIP*'. Extract the zip-file. This will create the folder:

```
<Download folder>\DvG_Arduino_lockin_amp-master
```

2.3 M4 Feather Express drivers and firmware

When using Windows install the Adafruit device drivers at:
https://github.com/adafruit/Adafruit_Windows_Drivers/releases/

Before the M4 Feather Express can be used as a lock-in amplifier it needs to be flashed with the correct firmware, once. This is already done for you. In case you need to do it again follows here the procedure:

Connect the M4 Feather Express via USB to your laptop. Quickly double-click on the tiny reset button of the M4 board. The LED on the board should turn green as indication that we have entered Bootloader Mode. Once the bootloader is running, check your computer. You should see a USB Disk drive called '`FEATHERBOOT`'. This folder contains the file `CURRENT.UF2` which is the current contents of the microcontroller flash. You can upload new firmware to the M4 board by copying over this file. The lock-in amplifier firmware can be found in the downloaded GitHub source files from §2.2 in folder:

```
\source_MCU_boards\pre-compiled_M4_feather
```

The M4 board should automatically reboot when the file is copied over and will now be running the lock-in amplifier firmware.

2.4 Running the main program

Connect the M4 Feather Express via USB to your laptop. Start the Anaconda Prompt window and change the folder to:

```
<Download folder>\DvG_Arduino_lockin_amp-master
```

Now run the following command in the prompt:

```
ipython DvG_Arduino_lockin_amp.py
```

This will start a user interface in which you can configure and control your measurements.

3 Hardware information

The Adafruit Feather M4 Express microcontroller board features a 120 MHz Cortex M4 with floating point support and 512 KB of Flash and 192 KB of RAM memory.

In the current lock-in amplifier firmware the DAC (digital-to-analog converter) and ADC (analog-to-digital converter) are set to operate at 12 bits resolution at a sample rate of 5 kHz. The frequency of the output reference signal ref_X^* is limited to $5 \text{ kHz} / 4 = 1250 \text{ Hz}$.

GND : ground
A0 : analog out ,single ended , 0.0 to 3.3 V: ref_X^*
A1 : analog in , differential , -3.3 to 3.3 V: $\text{sig_I}(+)$
A2 : analog in , differential , -3.3 to 3.3 V: $\text{sig_I}(-)$
3V : regulated 3.3 V out, max. 500 mA
USB : 5.0 V out as powered by the USB

For testing connect A0 to A1 and connect GND to A2,
i.e. directly feed ref_X^* into sig_I .

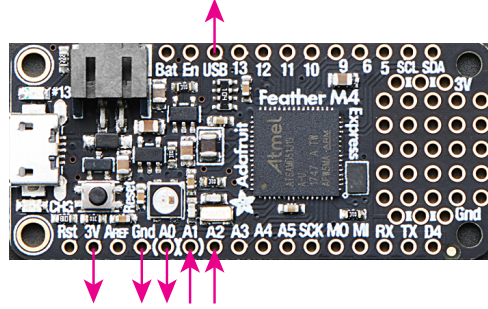


Figure 1: Pinout of the Adafruit Feather M4 Express microprocessor board.

4 Signal processing

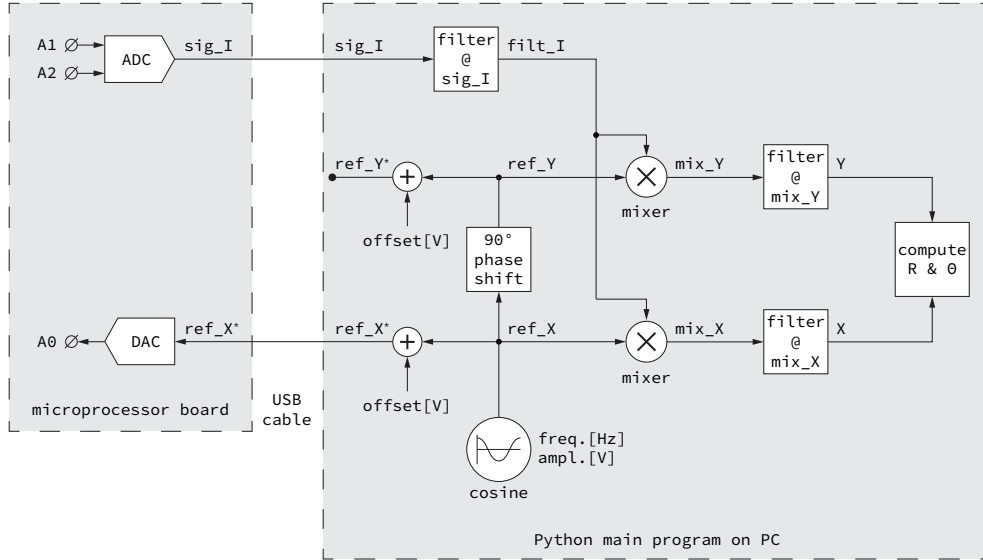


Figure 2: Diagram of the signal processing steps making up a fully functioning lock-in amplifier.

5 Python GUI

This section describes the graphical user interface (GUI) of the lock-in amplifier running in Python.

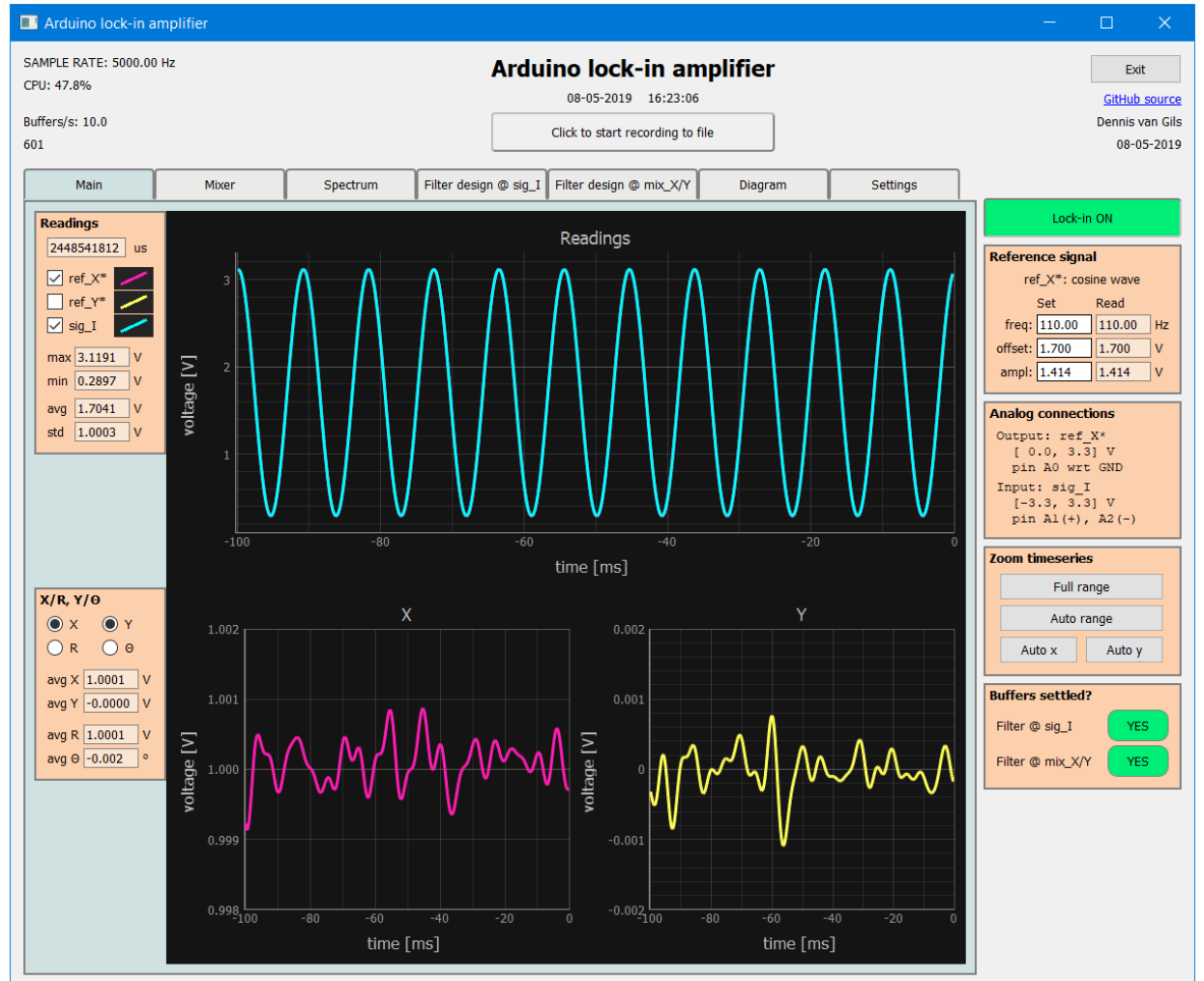


Figure 3: The *Main* tab page of the lock-in amplifier GUI.

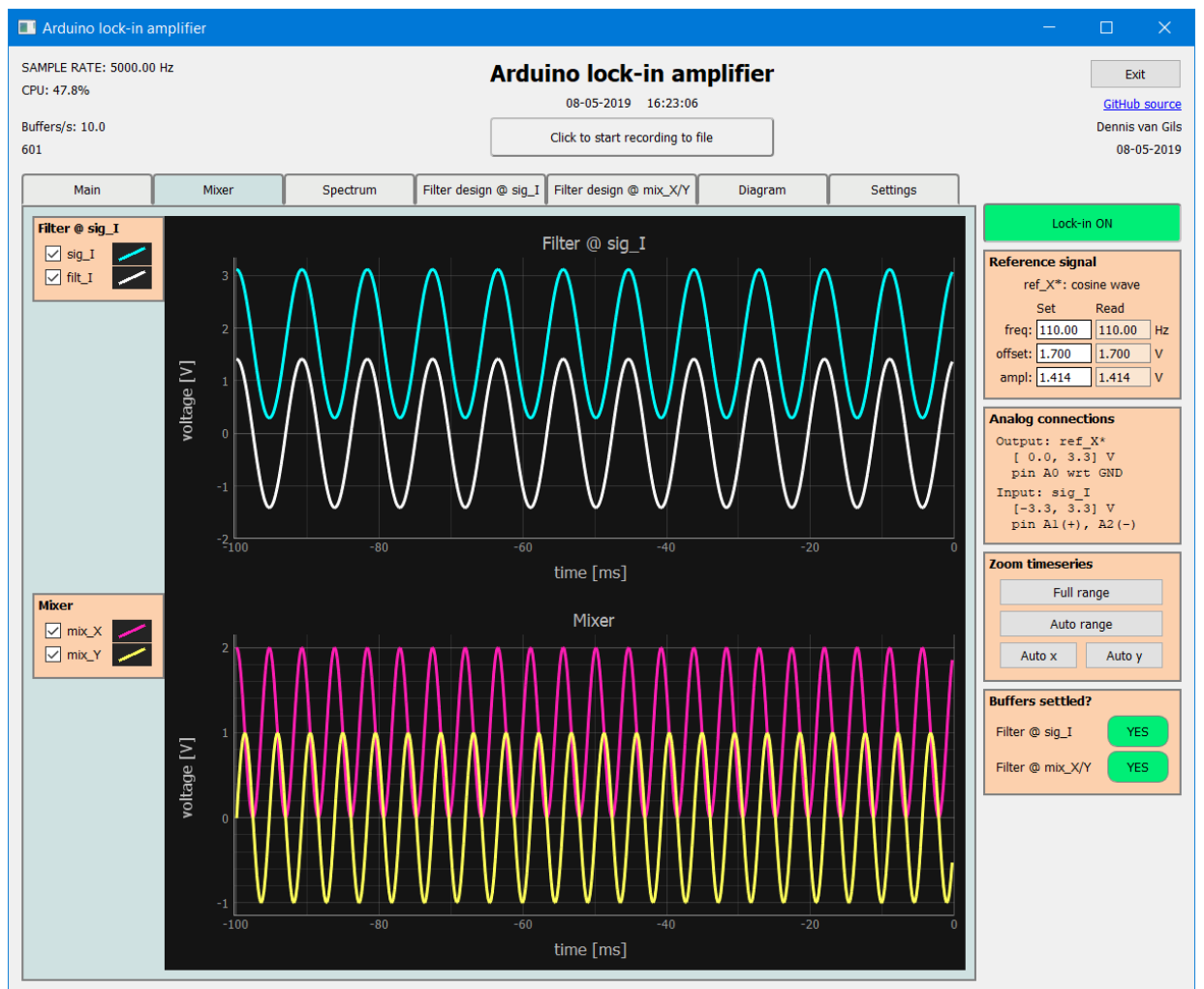


Figure 4: The *Mixer* tab page of the lock-in amplifier GUI.

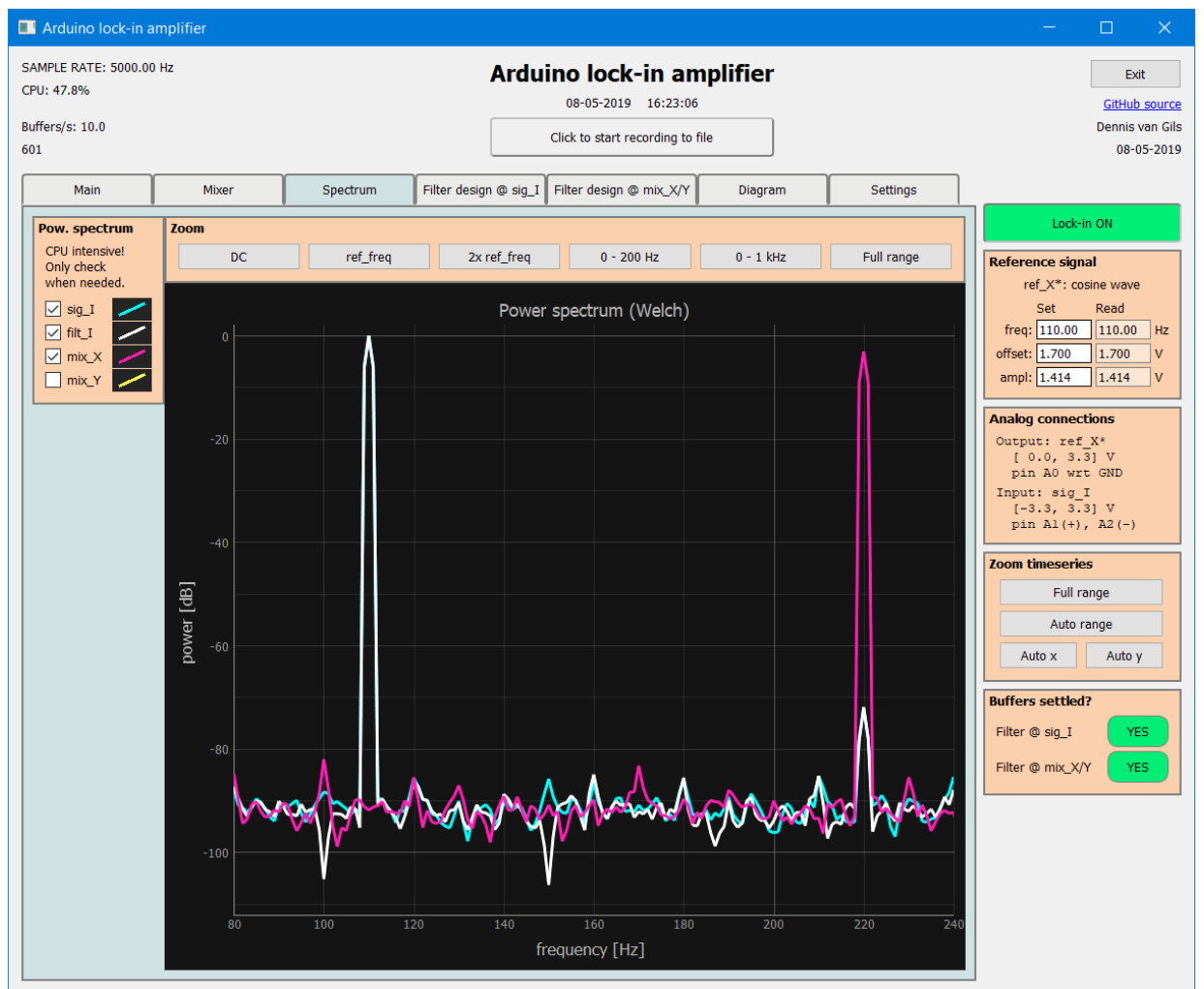


Figure 5: The *Spectrum* tab page of the lock-in amplifier GUI.

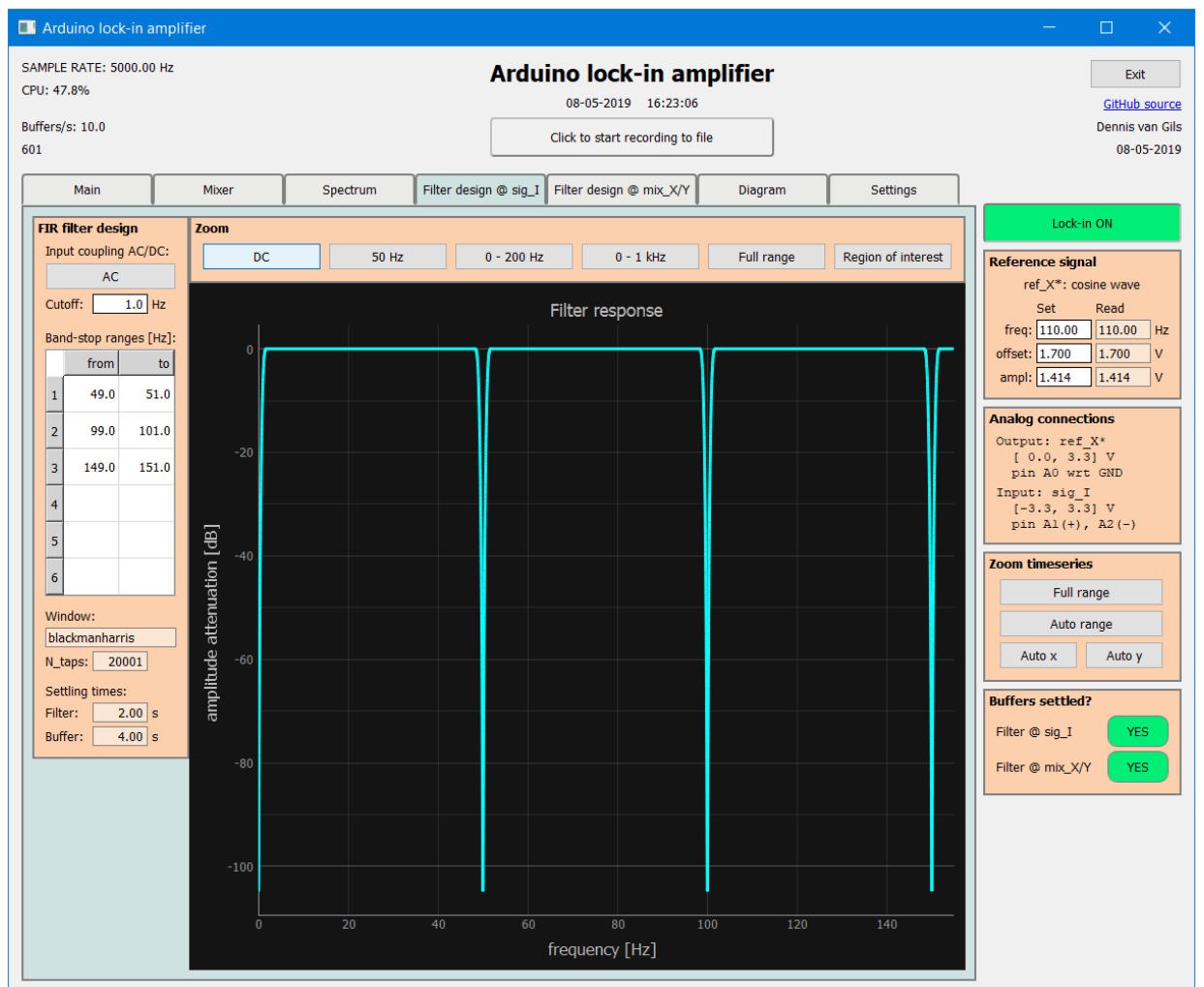


Figure 6: The *Filter design @ sig_I* tab page of the lock-in amplifier GUI.

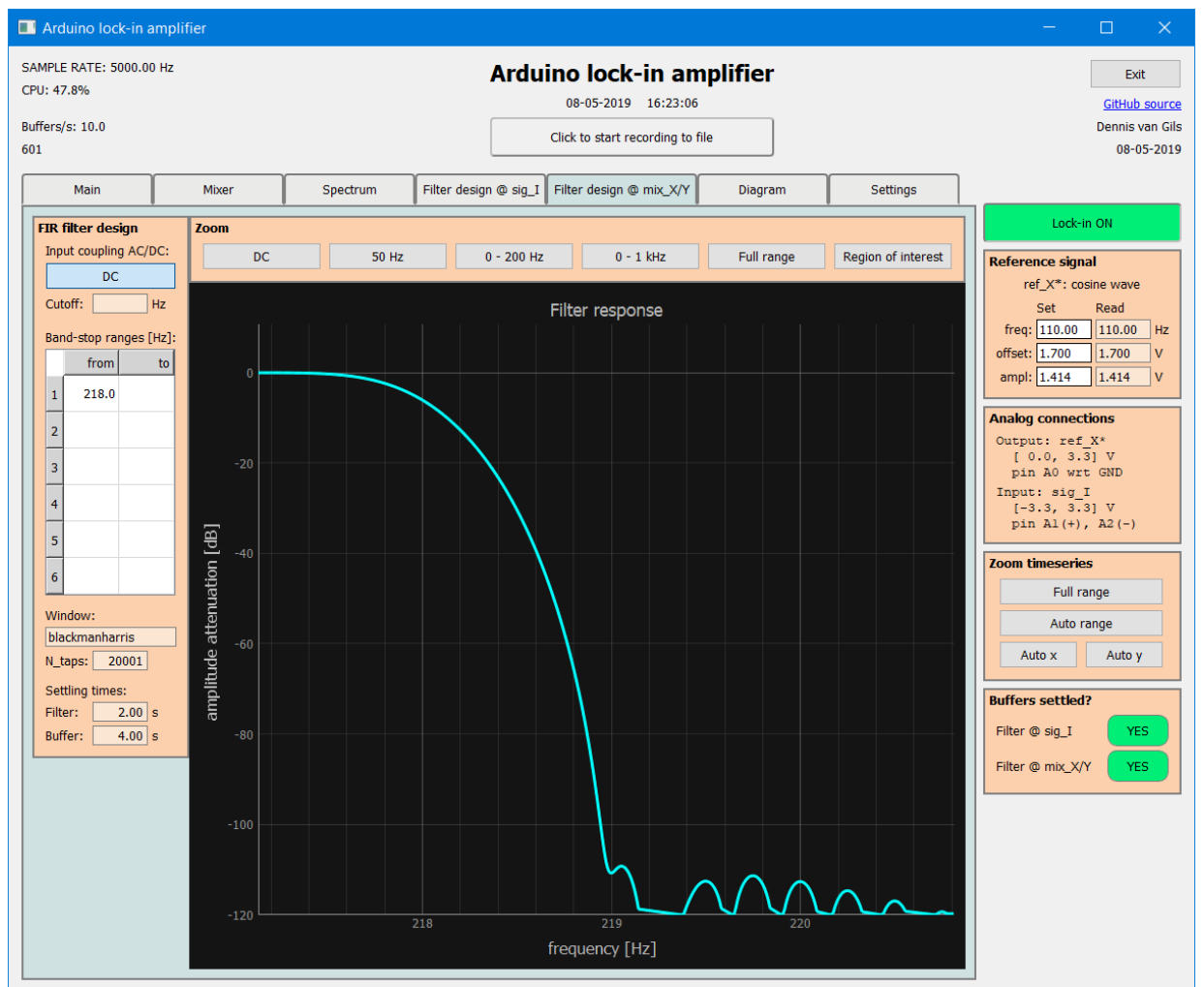


Figure 7: The *Filter design @ mix_X/Y* tab page of the lock-in amplifier GUI.