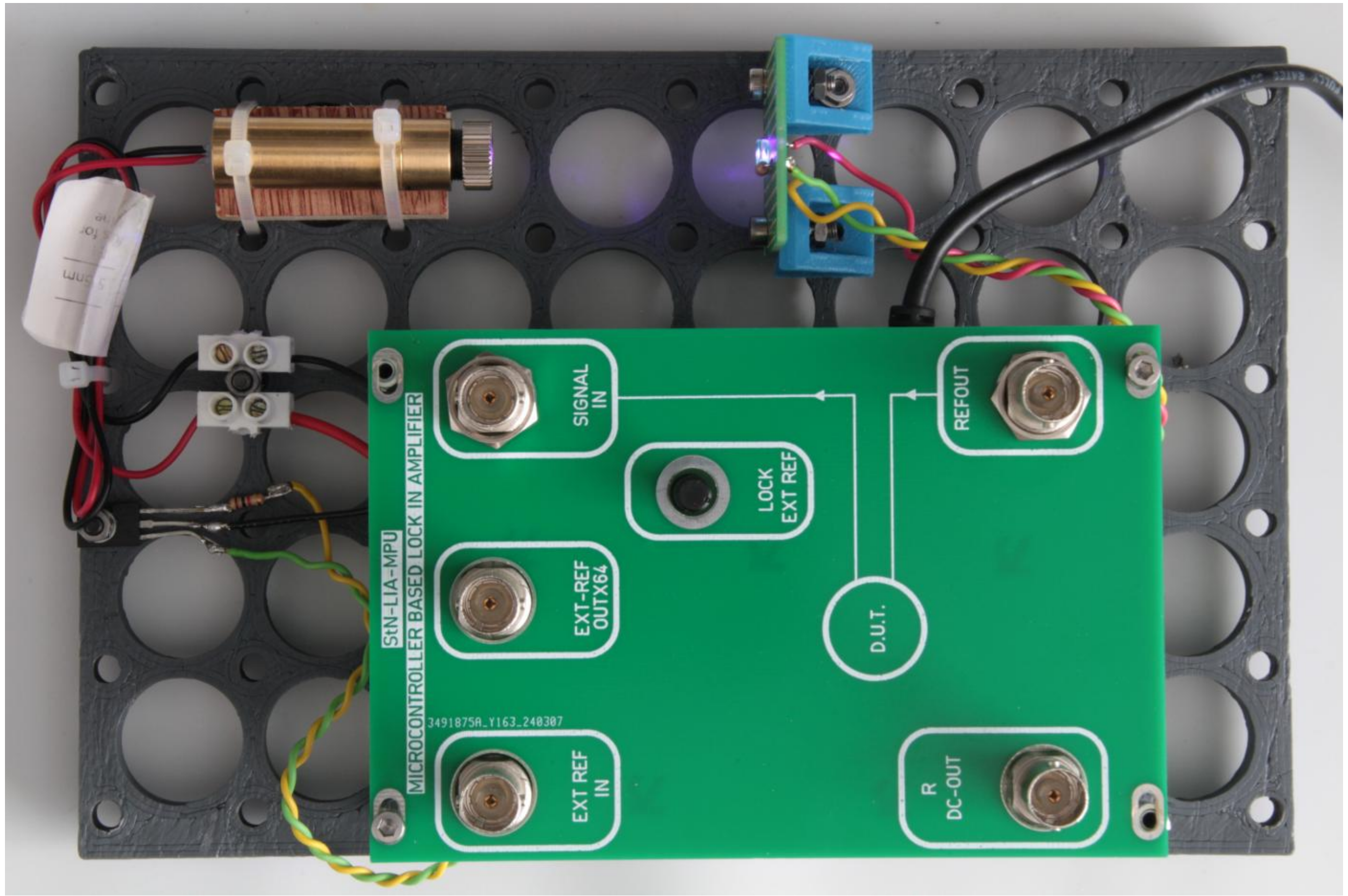
STN-LIA-MPU

An OLIA spinout development

Development/Discussion LOG

Nikos Chalikias

405 nm setup with 1000mW (electric) power



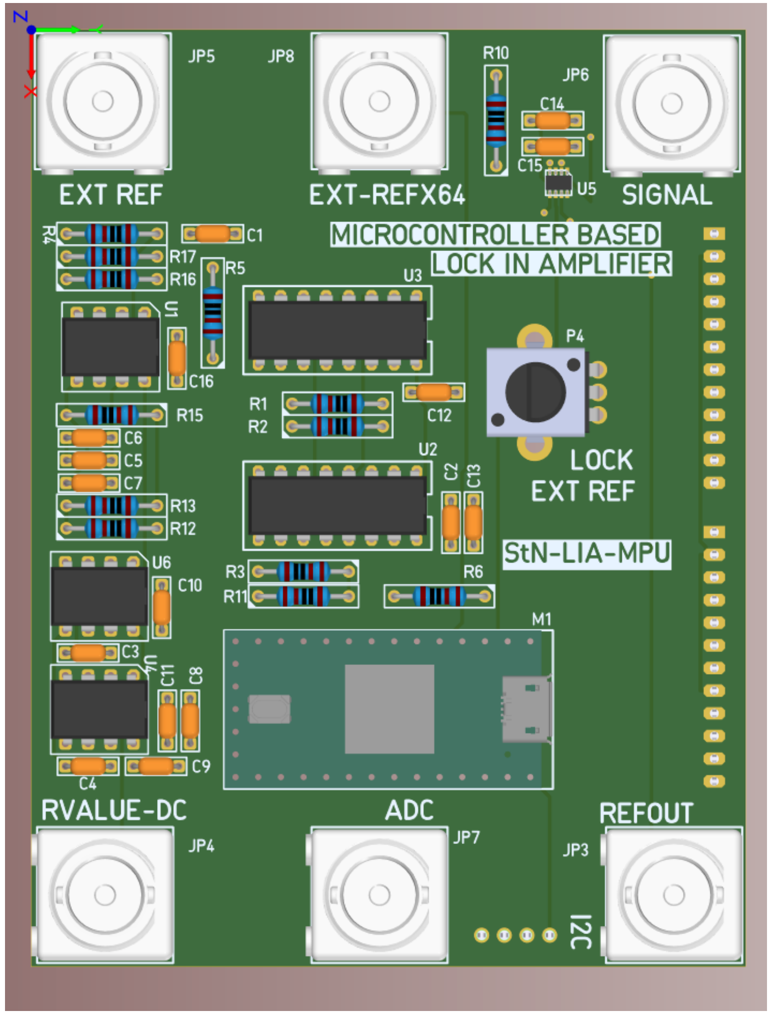
- StN-LIA-MCU with a NPN transistor to drive the 1W laser
- BPW34 general purpose Photodiode
- We used a 405nm Laser instead an infrared
- Output is high: 2000mV (direct laser beam to Photodiode
- Can see consistently a 5mV variation when exhale
- Will work with similar size/power/lens Infrared Laser
- Need for a 2nd Photodiode for liquid tests
- It is easier to multiplex and use a 2nd laser, [...if this works with the liquid project]

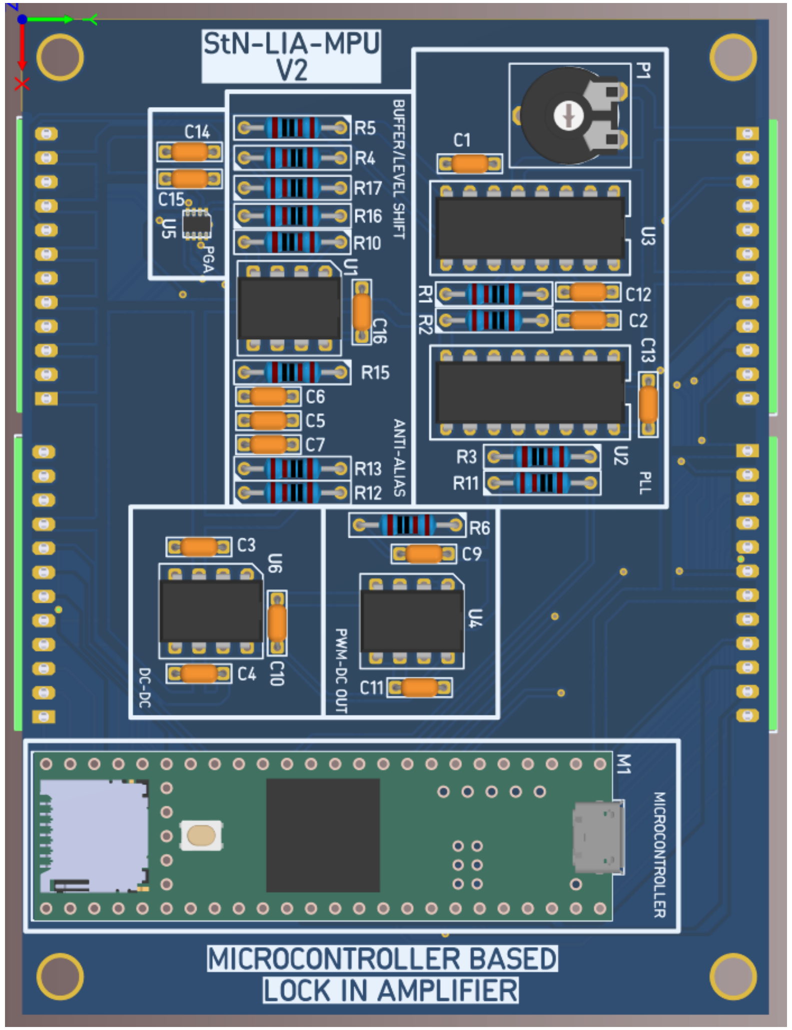
2000nm setup

- Proposed to setup a photodiode experiment only with PCBs
  - This is our goal, but many little details can block the operation for …some time

- Chinna proposal: **to use existing instrumentation and only PCB LIA (StN-LIA-MCU),** seems much better, as a 1st step
  - Then we can replace:
    - TEC with PCB (2nd step)
    - LDD with PCB (3d step)
    - Generator with PCB (4d step)
    - Add a microcontroller with LCD/buttons as experiment administrator (5th step)
    - **Add A PC** or **R-Pi as experiment administrator** (6th step)
    - Evaluate Preamplifier performance
    - Develop and evaluate, ultra low noise multi-output PSU
- This is scalable and sustainable, can start now
- 2000nm LED and photodiode to be delivered in a couple of days

StN-LIA-MCU





- V1 ongoing testing, works as expected
- A 2nd PCB to be assembled
- Most probably the 3d PCB will be V2

V2: Fixed PCB bugs, removed expensive connectors,
Easier to use screw terminals, local SD-Card, lower cost
2 layers PCB faster to be delivered from manufacturer, clean

Plan to assess LIA performance

- Started with evaluating Standard Deviation [σ] of the R component DC output of commercial LIA
  - Already have results for 3 COTS LIA

- **Propose to expand our evaluation scope with a basic setup for a QTF characterization**
  - Real problem
  - Experience on the setup
  - No need for expensive laser or optics
  - Can work with 0.5 euro 32.768 RTC crystals
  - No need for Preamplifiers
  - Can compare with Bari electronics

KNOWN ISSUES

HW and FW is based on OLIA

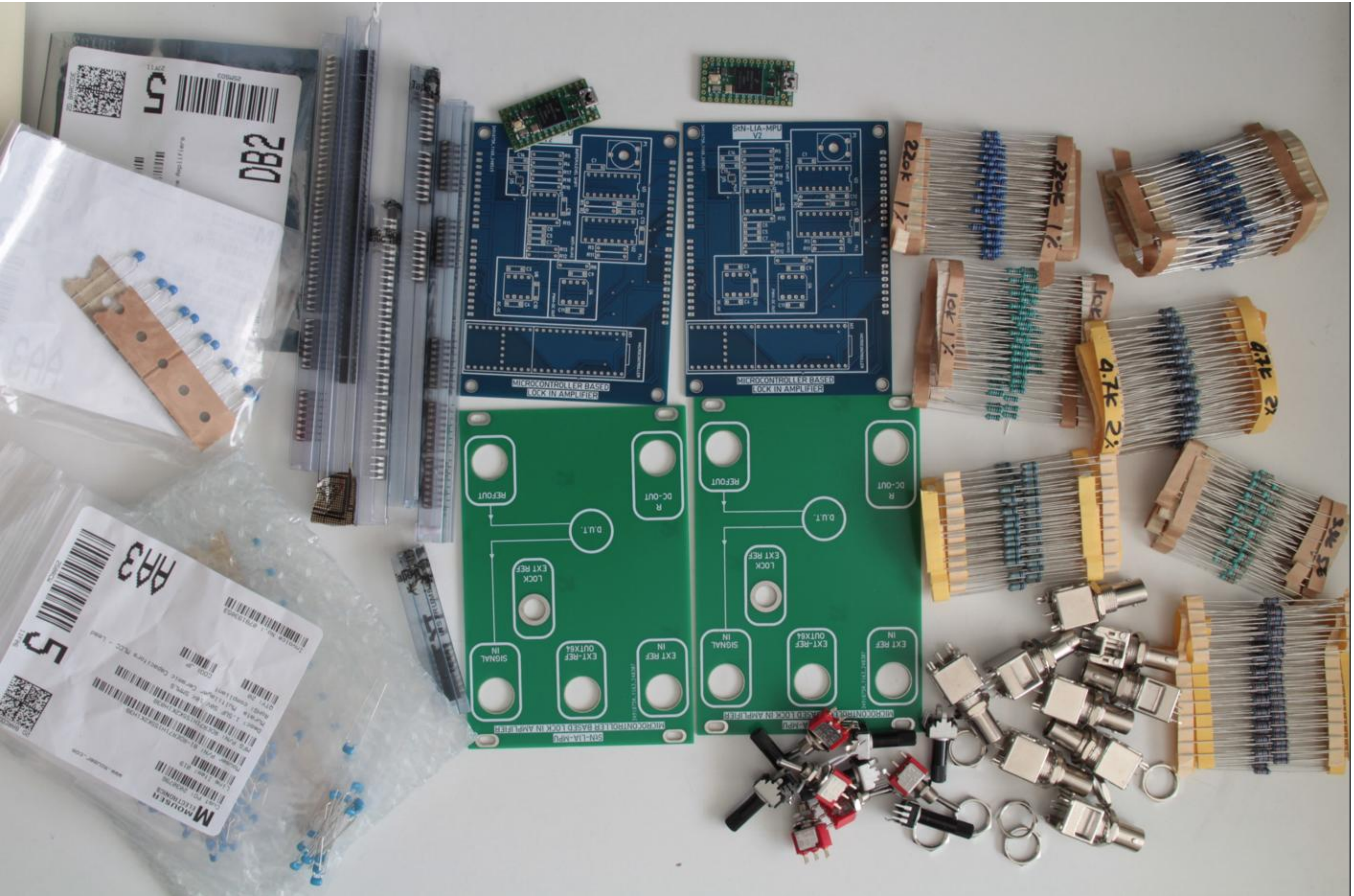
- Works good with internal REFERENCE signal generator but not for all Frequencies
- The external REFERENCE frequency is 64X the wanted Frequency, this is confusing for the user
- Works well with External REFERENCE at the order of 1KHz, but can’t follow for order of 10KHz
- In our FW version we have updated the multiplier ratio to 32X: so it works for 12KHz range
- Not any frequency for REFERENCE is working: It has to be an integer fraction of the sampling frequency
- Sampling Frequency is 200KHz
- We should not have the Integer fraction issue with external REFERENCE, as the sampling rate is (..should) be controlled by the external REFERENCE!!! ....But this not resolved yet...

N2O App4Farm GAS CELL

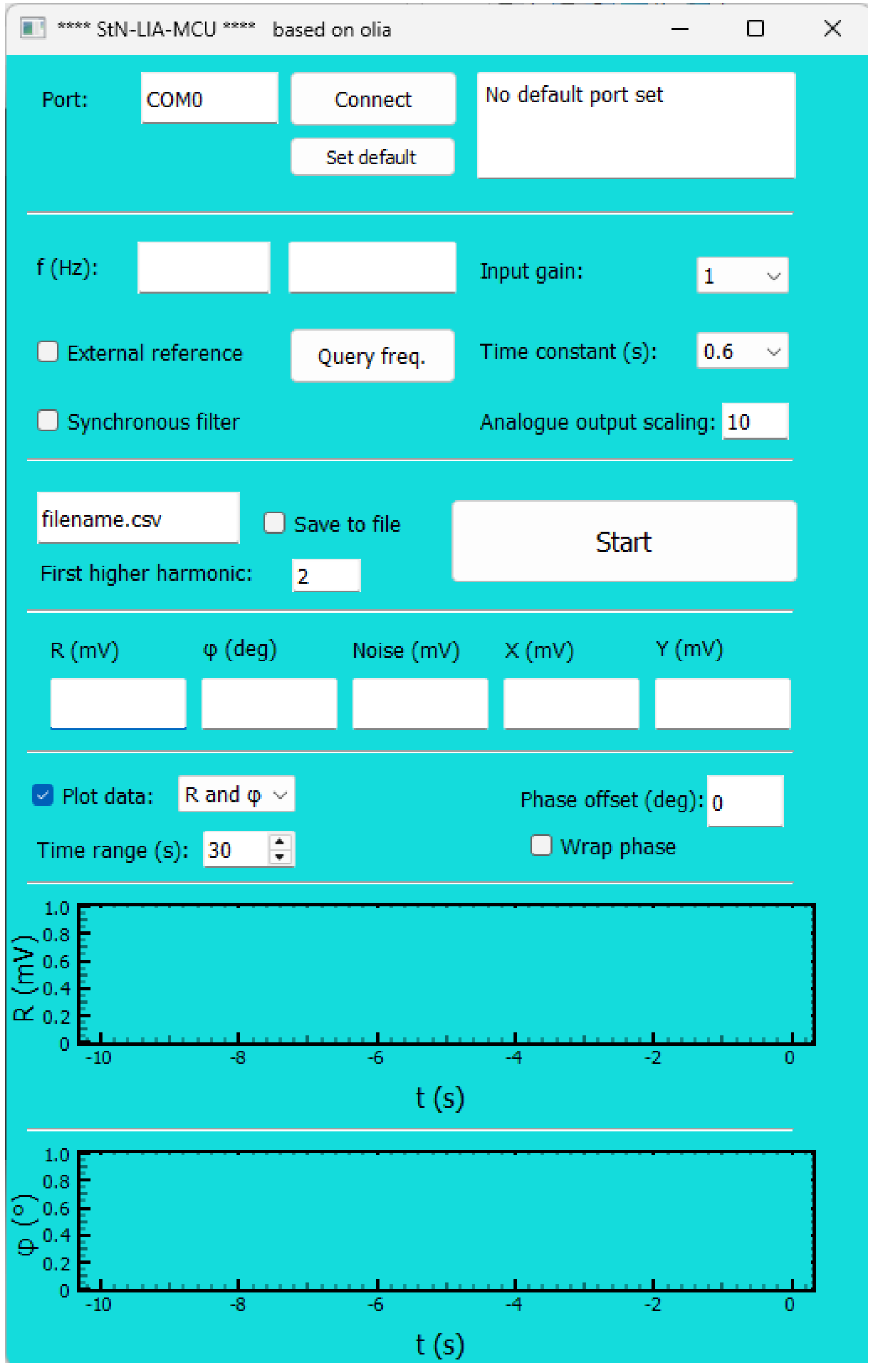
- RUN LIA 2 UNITS AT THE SAME TIME AND SAME REFERENCE
- TEST IF COMMON EXTERNAL REFERENCE CREATES 2 REF OUTPUTS AT CONSTANT PHASE DIFFERENCE
- BNC ARE NEEDED
- **IN 2 WEEKS?? YES IT IS POSSIBLE**
- DRIVE 5V 500mA LED, IS IMPORTANT
- USE EXTERNAL PSU FOR THE LED
- 3+ Boards needed

- STATUS
  - Tested Running 2 GUIs at the same time
  - 3D printing a plastic optics breadboard
  - To mount/integrate inside a Passepartout enclosure, bottle ouitside

PCBS TO BE SOLDERED

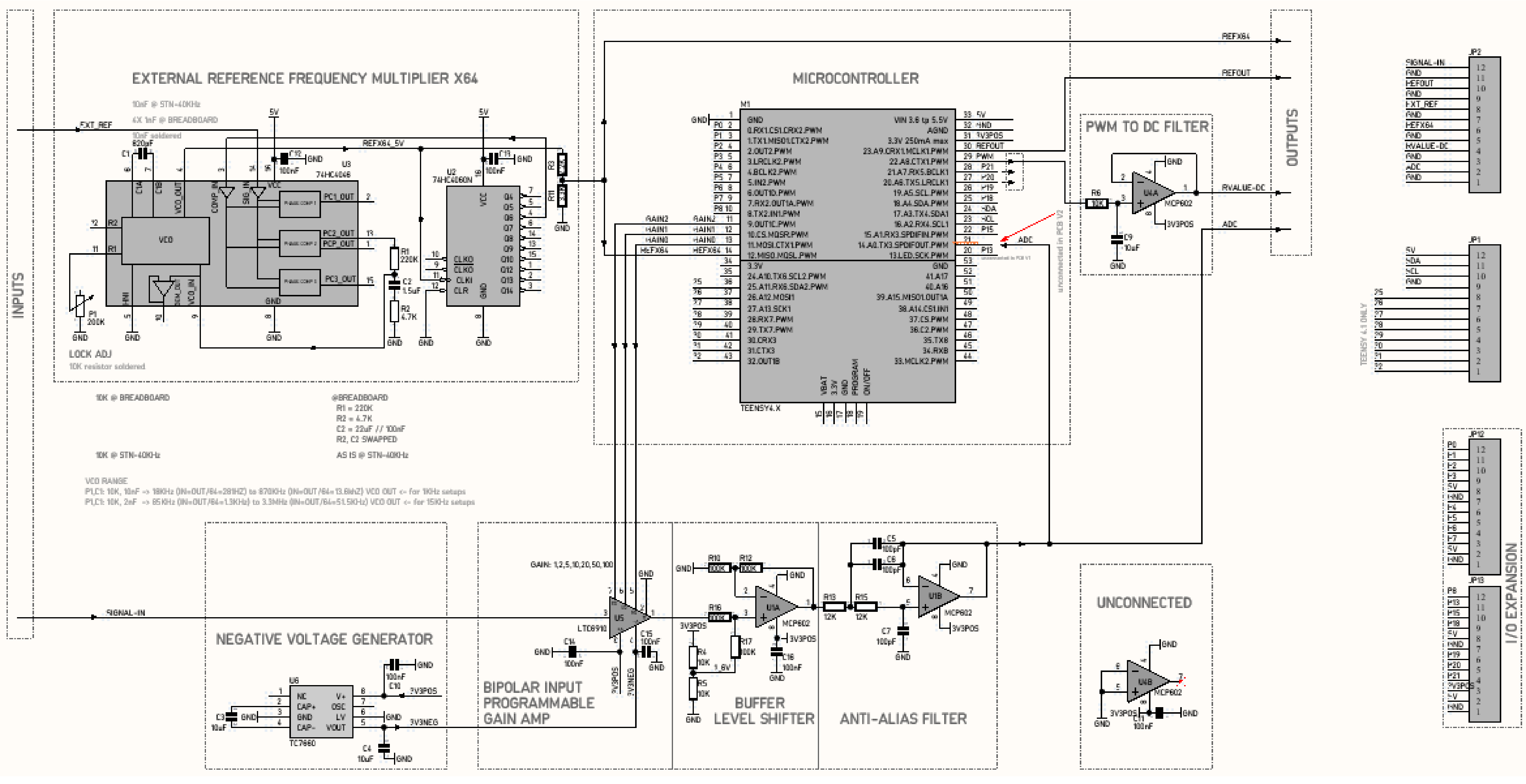


GUI



For our version of GUI: Only minor updates

Schematic


INPUTS
EXTERNAL REFERENCE FREQUENCY MULTIPLIER X64
MICROCONTROLLER
PWM TO DC FILTER
OUTPUTS
NEGATIVE VOLTAGE GENERATOR
BIPOLAR INPUT PROGRAMMABLE GAIN AMP
BUFFER LEVEL SHIFTER
ANTI-ALIAS FILTER
UNCONNECTED
I/O EXPANSION
LOCK ADJ
10K resistor soldered
10K @ BREADBOARD
10K @ STN-40KHz
@BREADBOARD
R1 = 220K
R2 = 4.7K
C2 = 22uF // 100nF
R2, C2 SWAPPED
AS IS @ STN-40KHz
VCO RANGE
GAIN: 1,2,5,10,20,50,100
TEENSY4.X
74HC4046
74HC4060N
LTC6910
MCP602
TC7660
REFX64
REFOUT
RVALUE-DC
ADC
SIGNAL-IN
EXT_REF

Function Calculate() runs every 5μs

Input is sampled 200 times for the 1ms REF period (1KHz)

```
xSig = (double)adc->analogReadContinuous(A0); //sample current adc value
```

Sin (0 to 2π) in 200 steps

0 to 199

For 1KHz: 200 samples per REF period

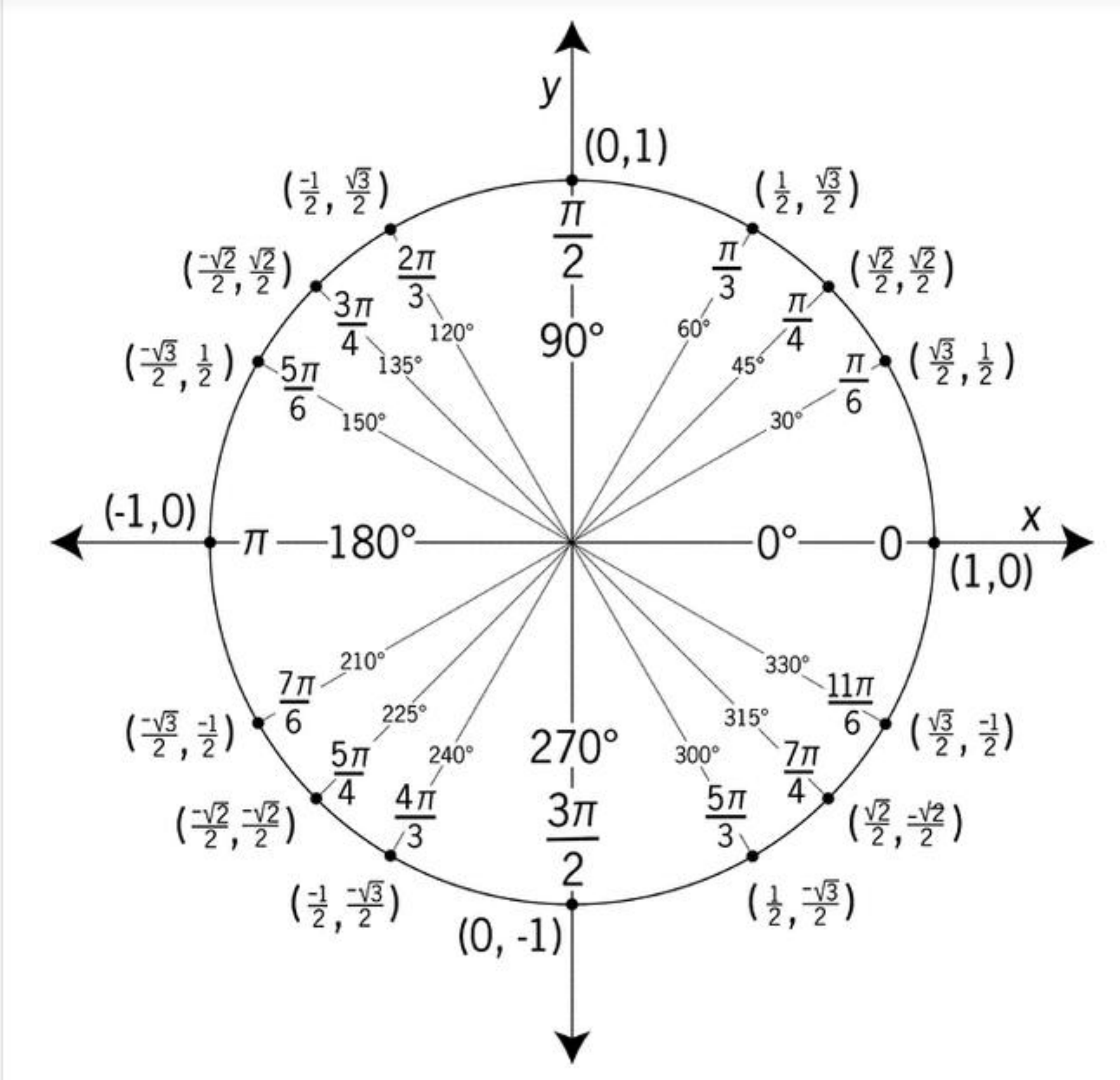
```
// Nikos: Below 2 lines are the core of the Lock In Amplifier
//calculate reference value and multiply (-1* because preamp is inverting)  // Nikos: Sinusoidal Reference signal is created here, Is like sampling a 1000Hz sinus
phasSig = -1*sin(((2.0 * (j + 1) * pi * sampleIndex) / samplesPerPeriod) - ((j + 1) * phaseDiff));
xi0 = (xSig*phasSig); //-1 because preamp is inverting                 // Nikos: Input sample is multiplied with Reference 'sample' here

```

Review of code for 1^{st} harmonic lock-in

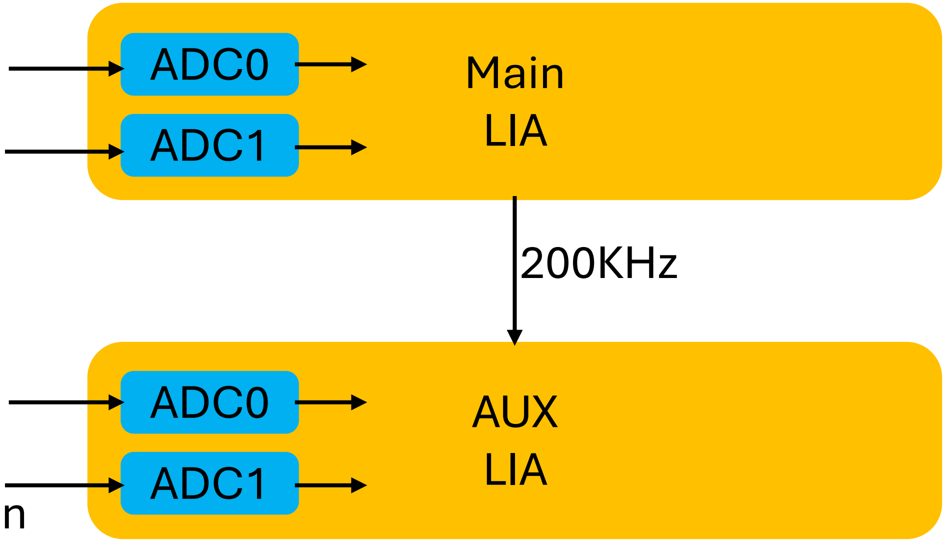
OLIA firmware Function **calculate()** is called periodically every 5μs (200KHz), to

- Sample input (ADC)
- Calculate the Reference **sinusoidal** ‘virtual-sample’ (200 samples)
    - The Reference frequency is: Fsample/A, where A is integer and Fsample is 200KHz
- Multiply the input sample with the Reference ‘sample’ and filter
- **Generate the Rectangular Reference frequency** for the D.U.T.
    - This is in phase with the Internally used sinusoidal Reference

Sinus calculation



Possible upgrades



Sync the sampling

- Use main/aux SNT-LIA-MPU boards with updated firmware
- Aux LIA board sampling interrupt is driven from the main
    - Main outputs the sampling frequency on a new output pin
    - Aux receives the sampling frequency in a new input pin
- Aux sampling interrupt(200KHz) is generated from the input pin, not from internal timer

Sample an external sinusoidal reference

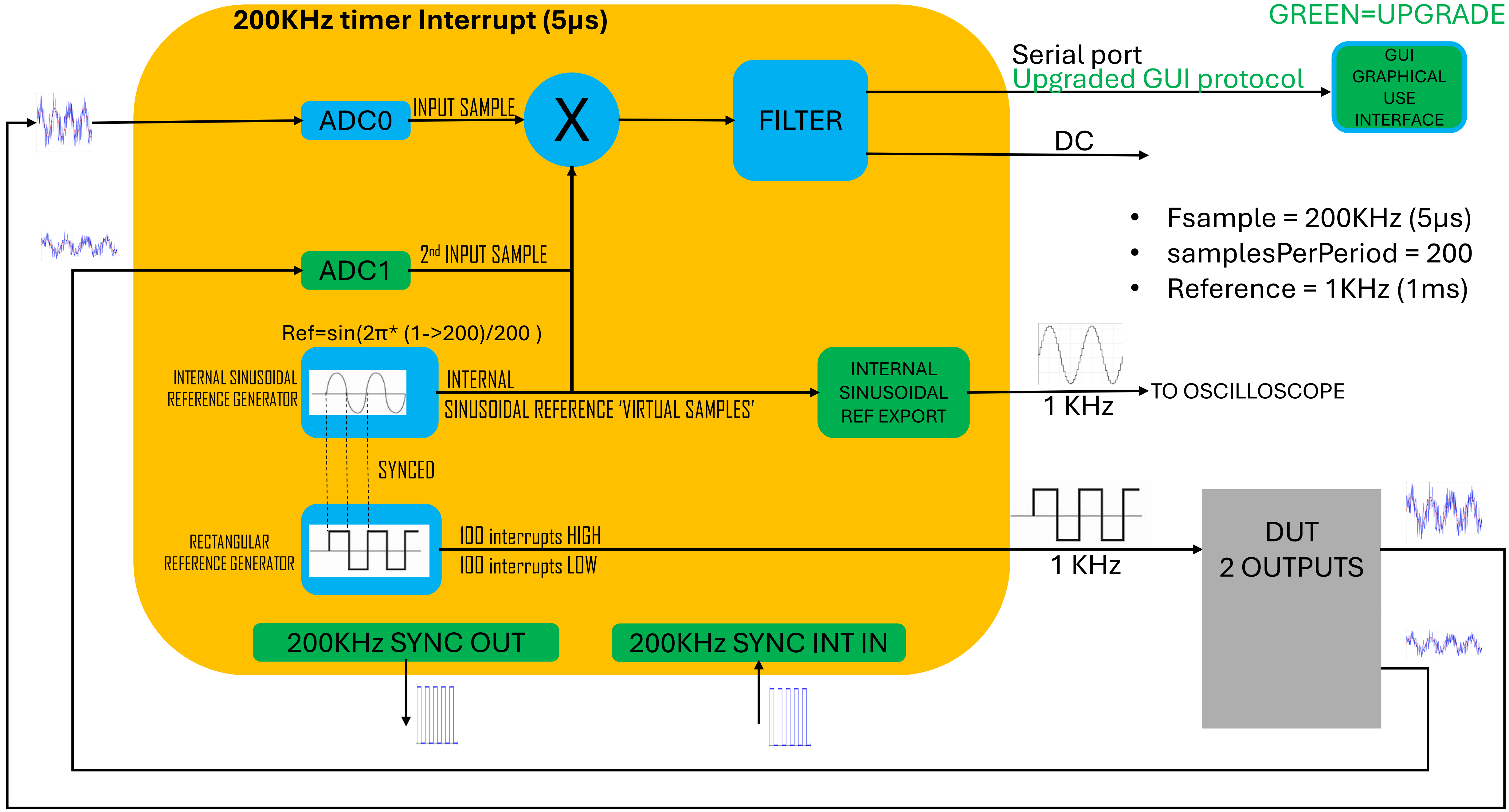
- Use a 2nd ADC analogue input pin
- Apply external sinusoidal external Reference to the new ADC analogue input
- Update the Firmware
    - Use the external ADC sampled sinusoidal Reference, instead the internally generated

Monitor the internally generated sinusoidal Reference

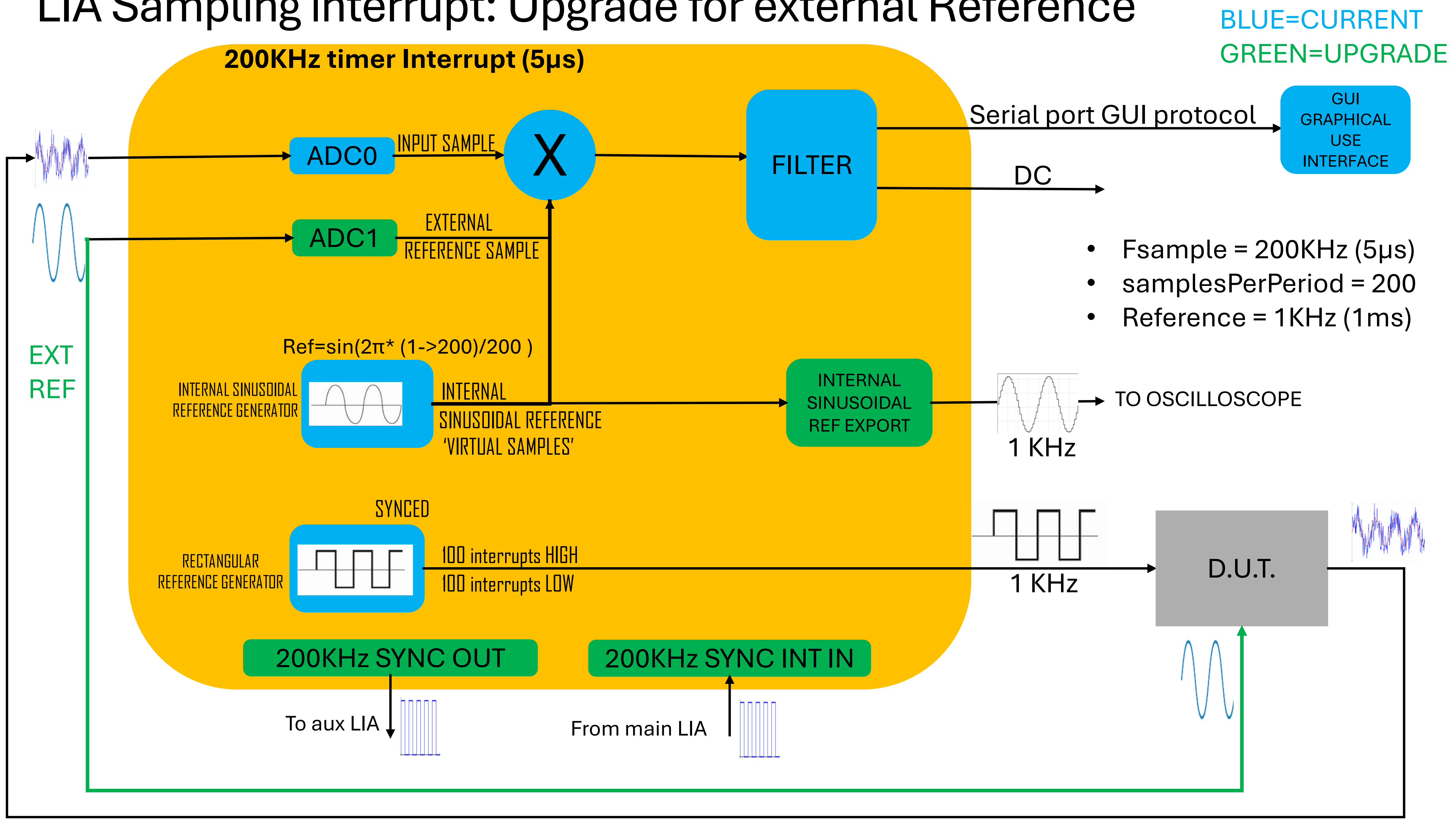
- Add a new DAC chip and connect external oscilloscope for monitoring
    - Alternatively, export the sinusoidal Reference to a PWM output, without using a new DAC
- Update the firmware to export the sinusoidal Reference to DAC, or the PWM output

LIA Sampling interrupt: Upgrade for 2nd Input

BLUE=CURRENT
GREEN=UPGRADE



LIA Sampling interrupt: Upgrade for external Reference



Higher Level Diagram for Orthogonal LIA

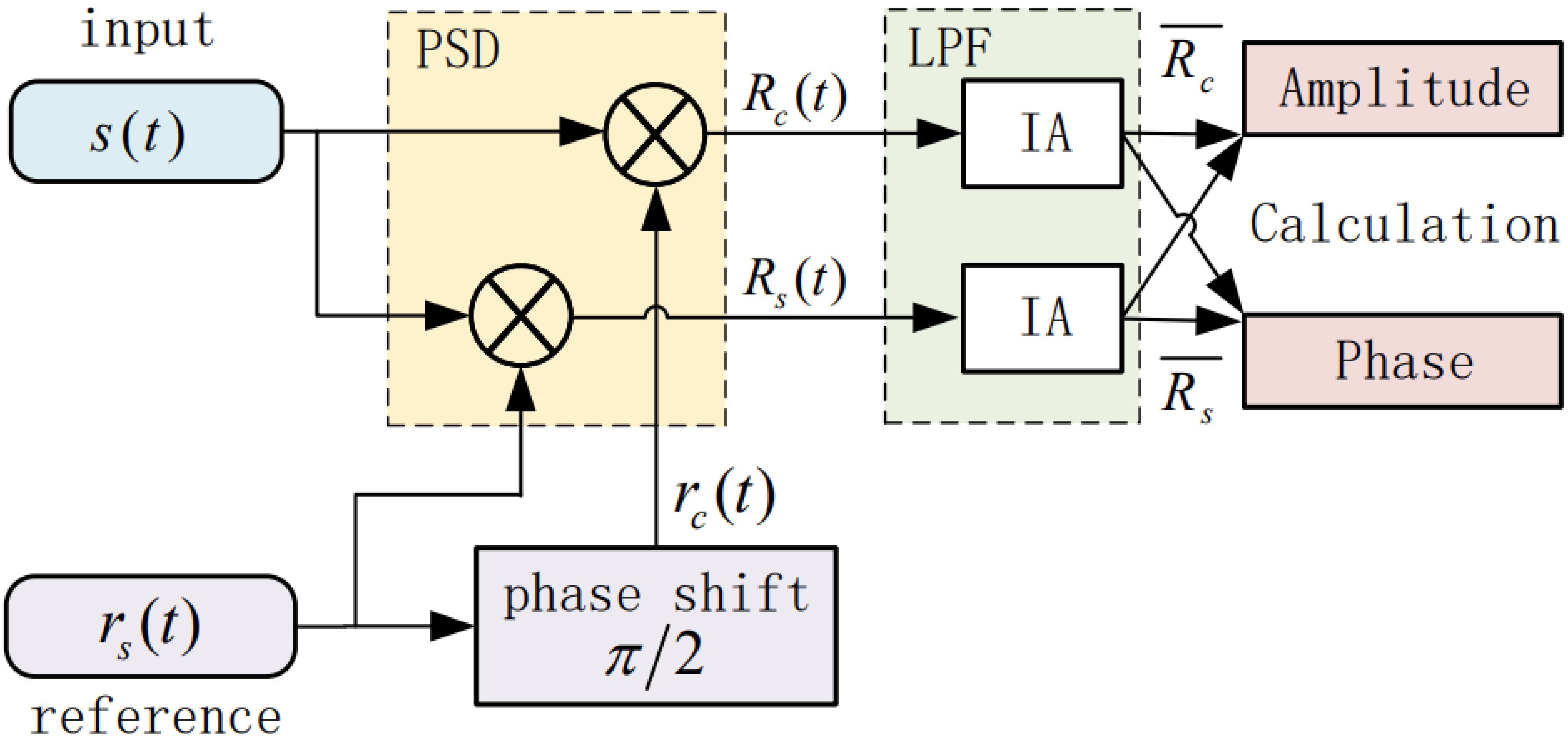
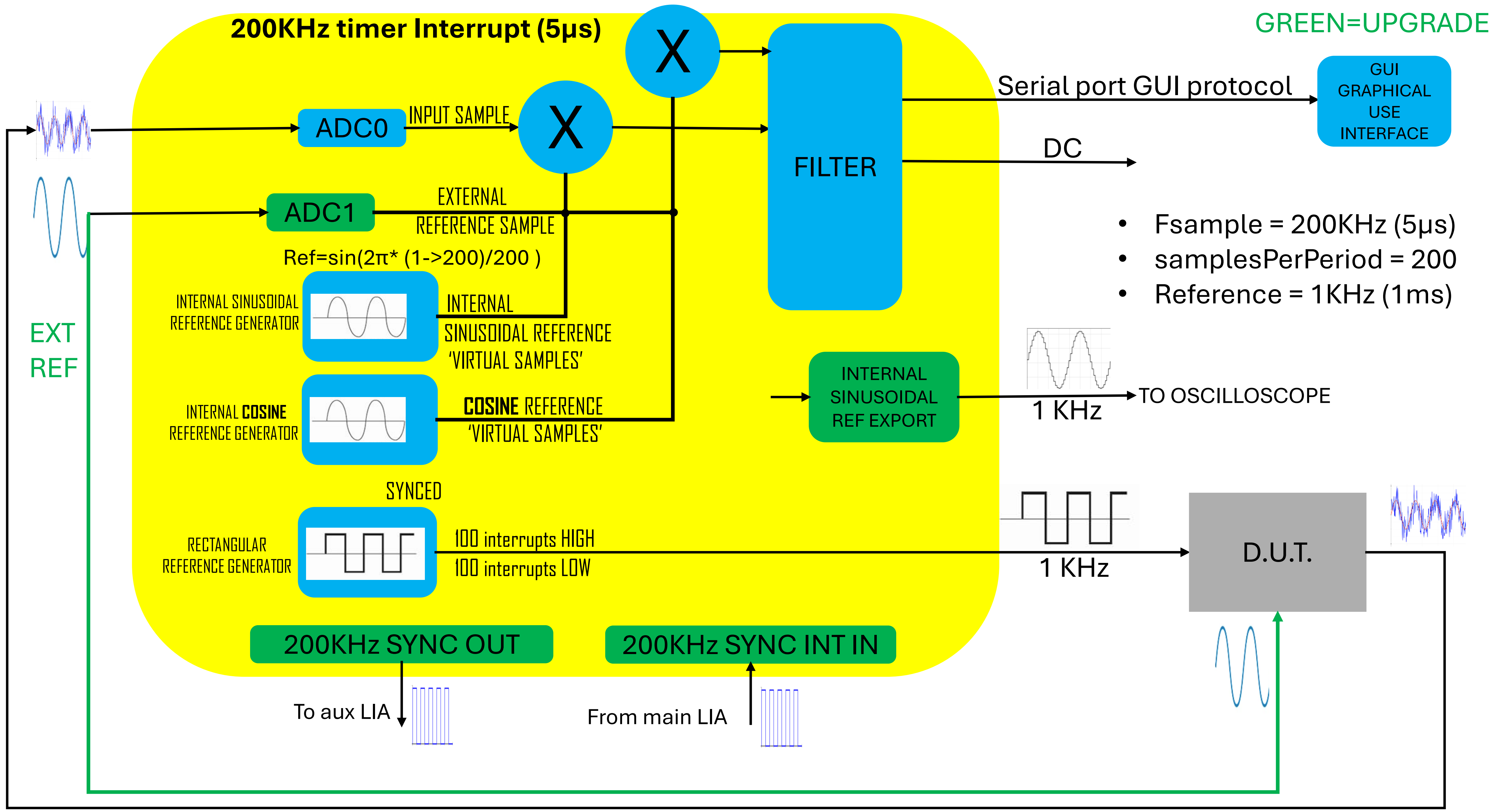


Figure 1. Structure of the orthogonal lock-in amplifier.

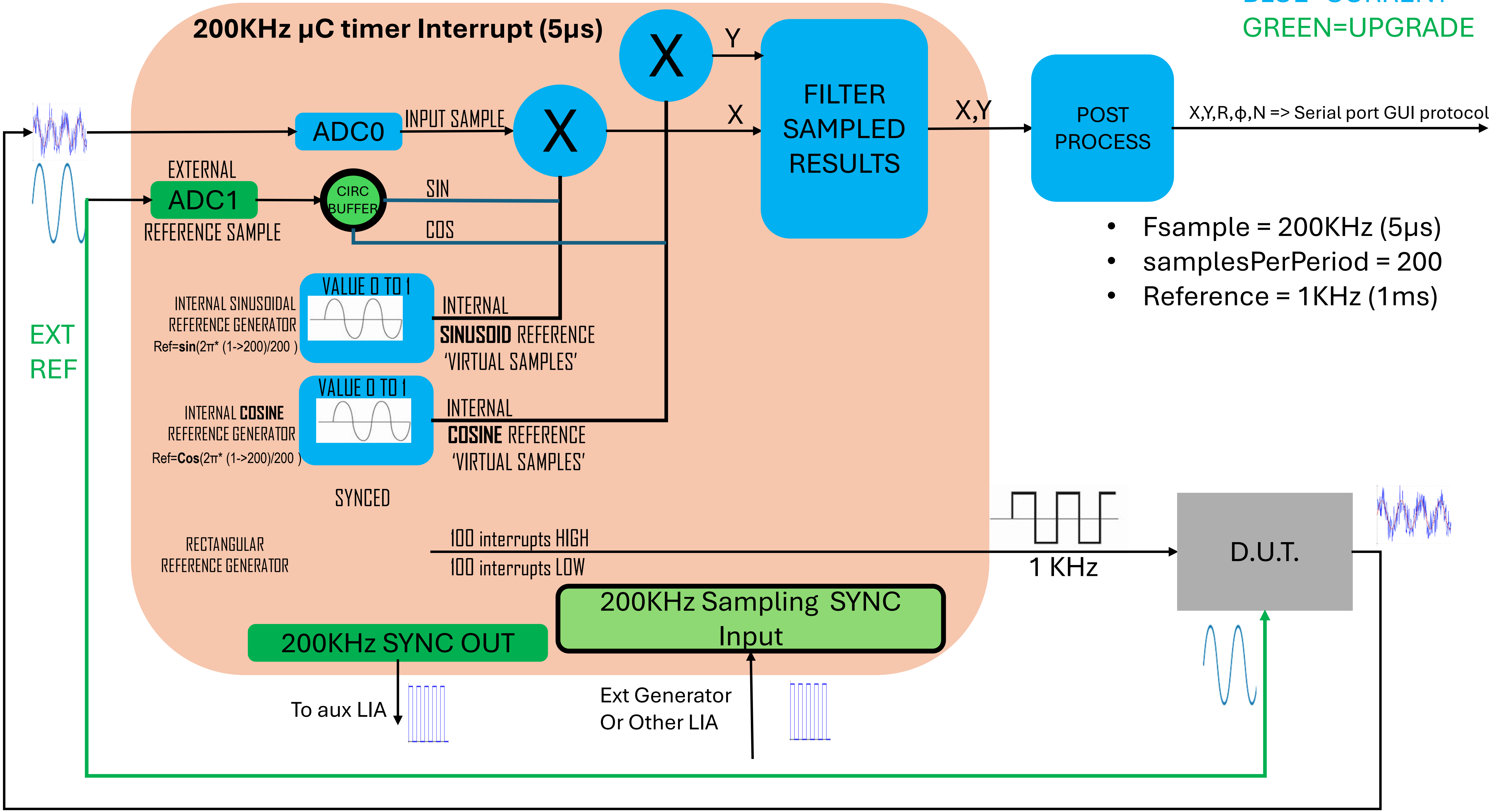
Appl. Sci. 2022, 12, 6431

LIA Sampling interrupt, Includes COS : Upgrade for external Reference

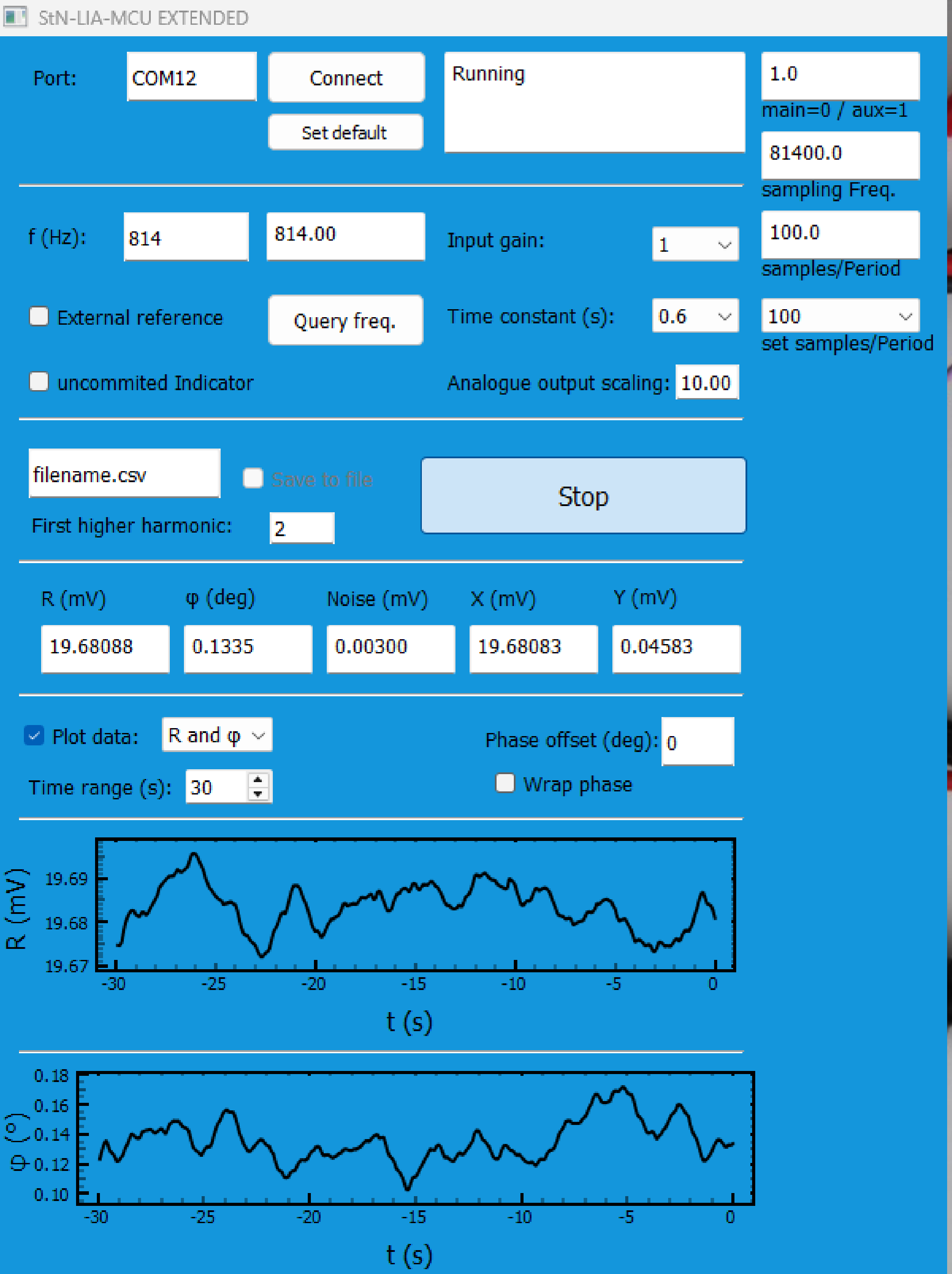
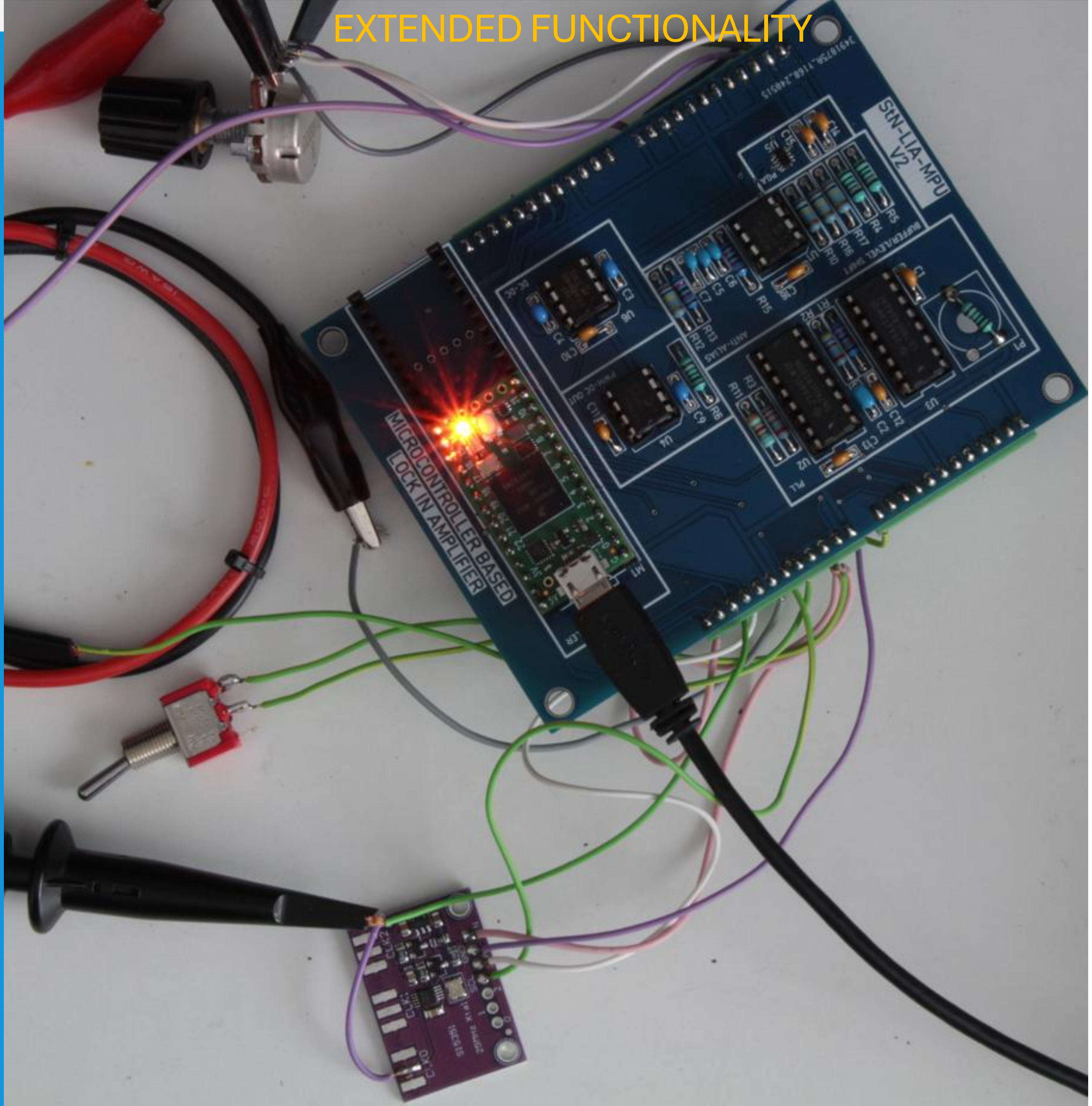


- Fsample = 200KHz (5μs)
- samplesPerPeriod = 200
- Reference = 1KHz (1ms)

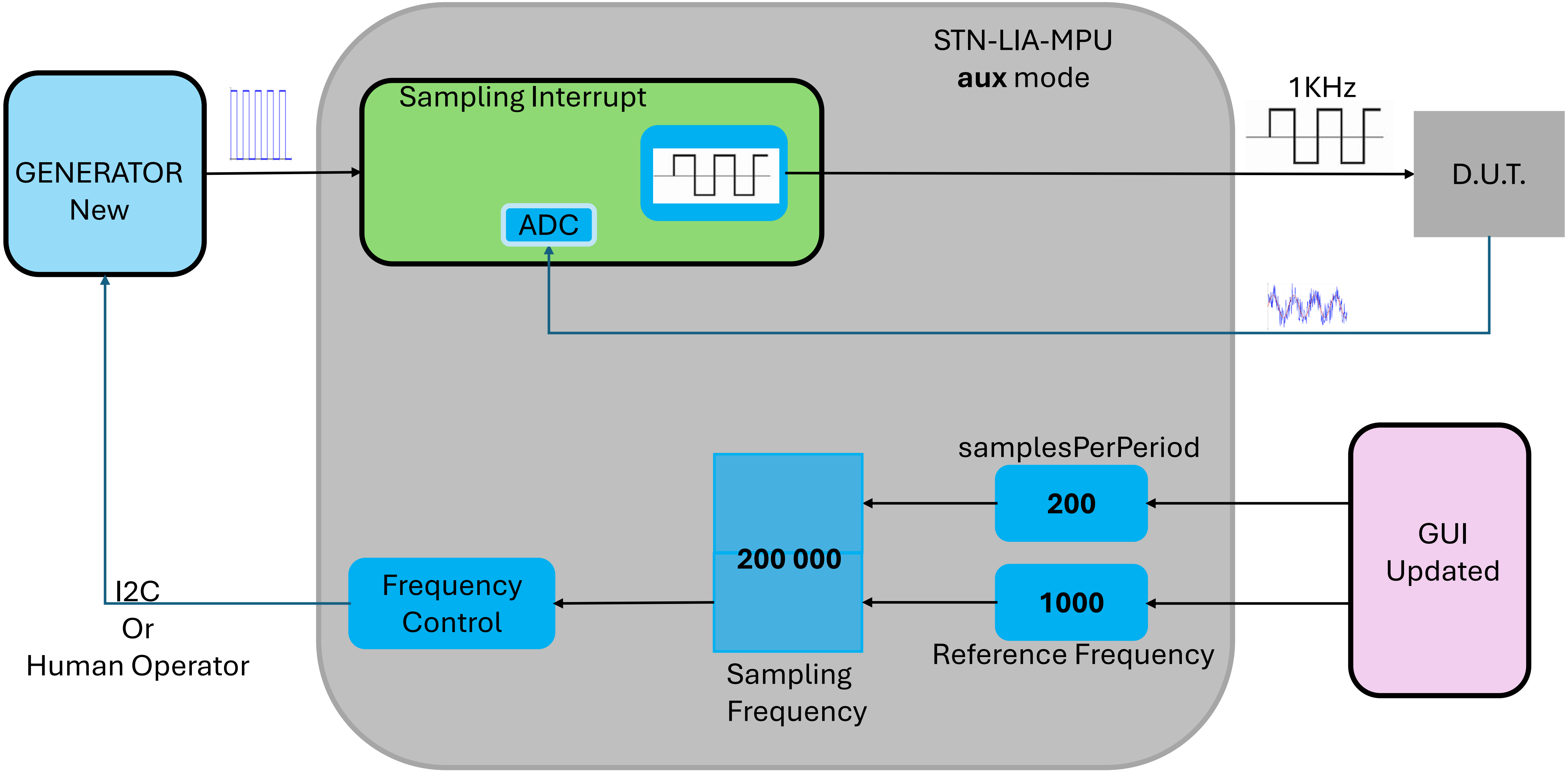
LIA Sampling interrupt, with cos and circular buffer: Upgrade for external Reference



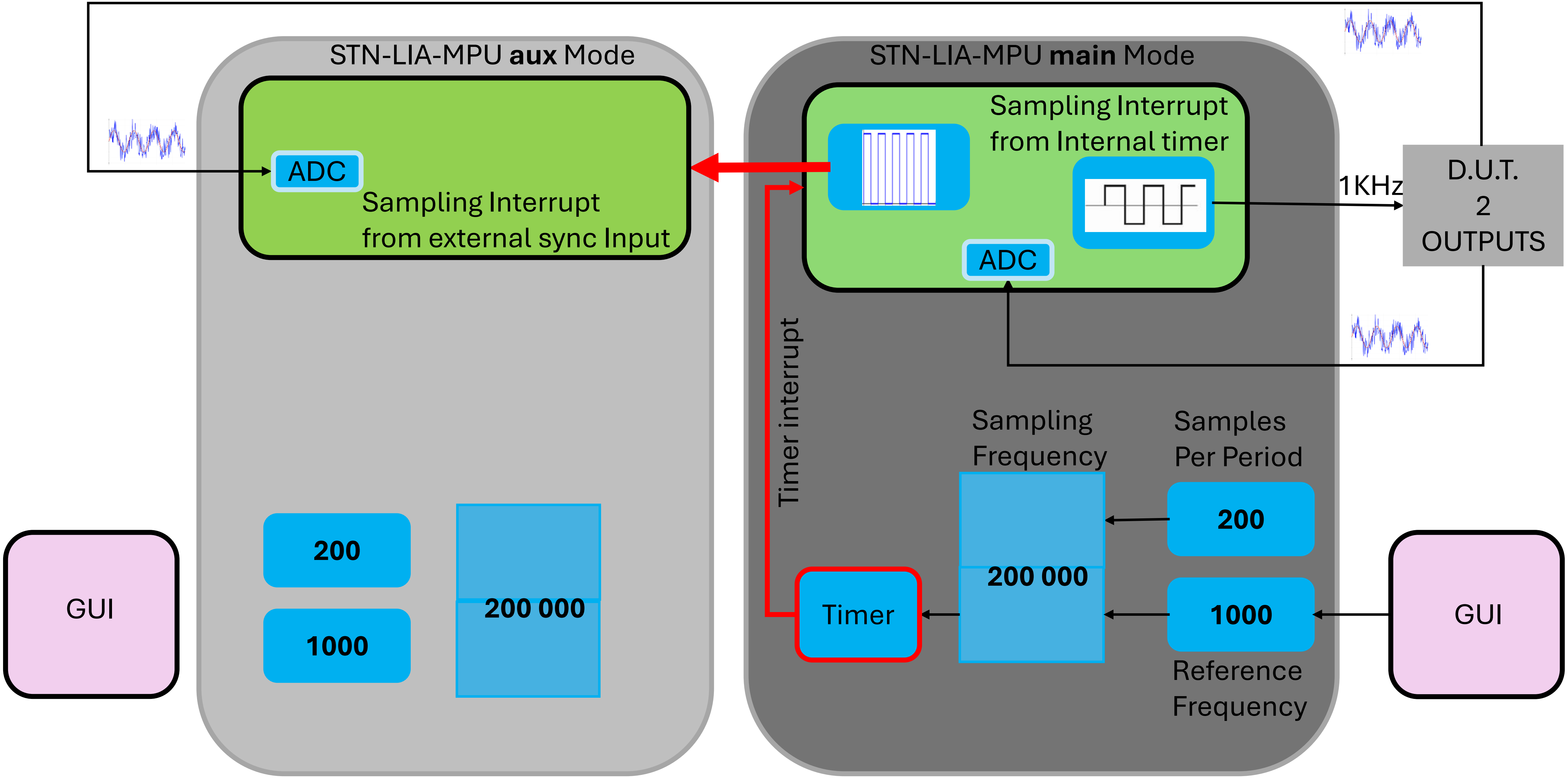
- Fsample = 200KHz (5μs)
- samplesPerPeriod = 200
- Reference = 1KHz (1ms)


StN-LIA-MCU EXTENDED
Port:
COM12
Connect
Set default
Running
1.0
main=0 / aux=1
81400.0
sampling Freq.
100.0
samples/Period
f (Hz):
814
814.00
Input gain:
1
External reference
Query freq.
Time constant (s):
0.6
100
set samples/Period
uncommited Indicator
Analogue output scaling:
10.00
filename.csv
Save to file
Stop
First higher harmonic:
2
R (mV)
φ (deg)
Noise (mV)
X (mV)
Y (mV)
19.68088
0.1335
0.00300
19.68083
0.04583
Plot data:
R and φ
Phase offset (deg):
0
Time range (s):
30
Wrap phase
R (mV)
19.69
19.68
19.67
-30
-25
-20
-15
-10
-5
0
t (s)
φ (°)
0.18
0.16
0.14
0.12
0.10
-30
-25
-20
-15
-10
-5
0
t (s)

EXTENDED FUNCTIONALITY
StN-LIA-MPU V2
MICROCONTROLLER BASED LOCK IN AMPLIFIER

StN-LIA-E: Extended Functionality



StN-LIA-E: Two Synchronized **L**ock **I**n **A**mplifiers



LIA Sampling interrupt, Control with external Precise generator

- Any Frequency, including QEPAS resolution
- Precise [0.0001Hz] Frequency Resolution (0.01Hz / 100)
- Can be used as External Generator for a Bench Lock In Amplifier
- Synchronized operation of multiple STN-LIAs [solves UCC setup]
- With a small addition the PCB setup can produce sinusoid output in the audio Freq Range

Synced Input ADC channels and Circular Buffer

Synchronized reading of 2 ADC channels and circular buffer for the Reference

```
result = adc->readSynchronizedContinuous();
result.result_adc0 = (double)result.result_adc0;
result.result_adc1 = (double)result.result_adc1;
xSig = result.result_adc0;
rSig = result.result_adc1;
adcBuffer[sampleIndex] = rSig;      // Nikos: Experimental Circulal buffer, collects sampleIndex samples of the sinusoid Reference Input,

```

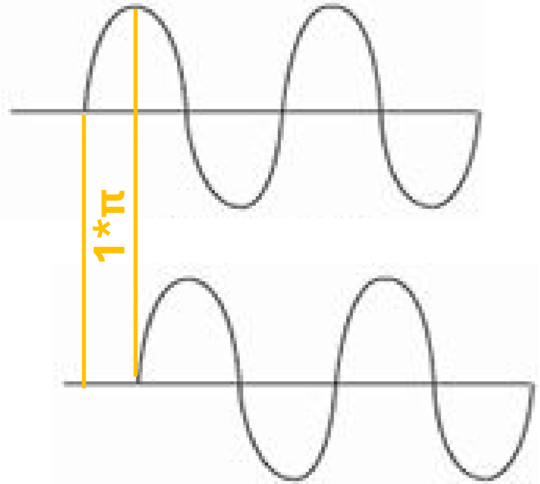
[0 to 199] for 1KHz and Sampling rate 200 KHz

The circular buffer for the sinusoid Reference signal Input, is used to create the **Cosine** of the Reference

Sin and Cos implementation in C++

```
    // Nikos: Below 2 lines are the core of the Lock In Amplifier
    //calculate reference value and multiply (-1* because preamp is inverting)  // Nikos: Sinusoidal Reference signal is created here, Is like sampling a 1000Hz sin
    phasSig = -1*sin(((2.0 * (j + 1) * pi * sampleIndex) / samplesPerPeriod) - ((j + 1) * phaseDiff));
    //xi0 = (xSig*phasSig); //-1 because preamp is inverting                    // Nikos: Input sample is multiplied with Reference 'sample' here
    xi0 = ( xSig * adcBuffer[sampleIndex]);    // Nikos: Experimentall Multilpy the 2 ADC inputs: Signal Input / Sinusoidal Signal Input
                                               // Nikos: Experimental Circular Buffer  for the sinusoid  external Reference, get the sin

    //first smoothing
    delta = xi0 - xiFilt1[j];
    incr = alpha*delta;
    xiFilt1[j] = xiFilt1[j] + incr;

    //second smoothing
    delta = xiFilt1[j] - xiFilt2[j];
    incr = alpha*delta;
    xiFilt2[j] = xiFilt2[j] + incr;

    if (j == 0){ //skip to first higher harmonic
      j = firstHigherHarmonic - 2;
    }
  }
  //same for quadrature harmonics
  for (j = 0; j < lastHigherHarmonic; ++j){

    //calculate reference value and multiply
    quadSig = -1*cos(((2.0 * (j + 1) * pi * sampleIndex) / samplesPerPeriod) - ((j + 1) * phaseDiff));
    //xq0 = (xSig*quadSig);
    xq0 = ( xSig * adcBuffer[sampleIndex+49]);  // Nikos: Experimental Circular Buffer  for the sinusoid  external Reference, get the cosine after samplesPerIndex/4

```

SIN (in phase harmonic)



COS (quadrature harmonic)

Cosine is 1π (200/4 for 1KHz) later in the Circular Buffer

Requirements discussion

We're looking for two signal channels driven by a single modulation source, ideally produced by the LIA, to modulate a single LED.

From the LED, we get two signals by splitting the light into two separate channels that are individually detected after the light passes through the sample. These need to be separately lockin amplified.

We would ideally want to get a sensitive measurement of the channels individually, or maybe just the ratio of the channels if that was more sensitive and could be determined directly.
I think this means:
one modulation source (in the LIA)
two input channels
lockin signal processing for two channels . It might be possible to do the two channels from the same op amp etc., but maybe not.

After 31/12/2024 call

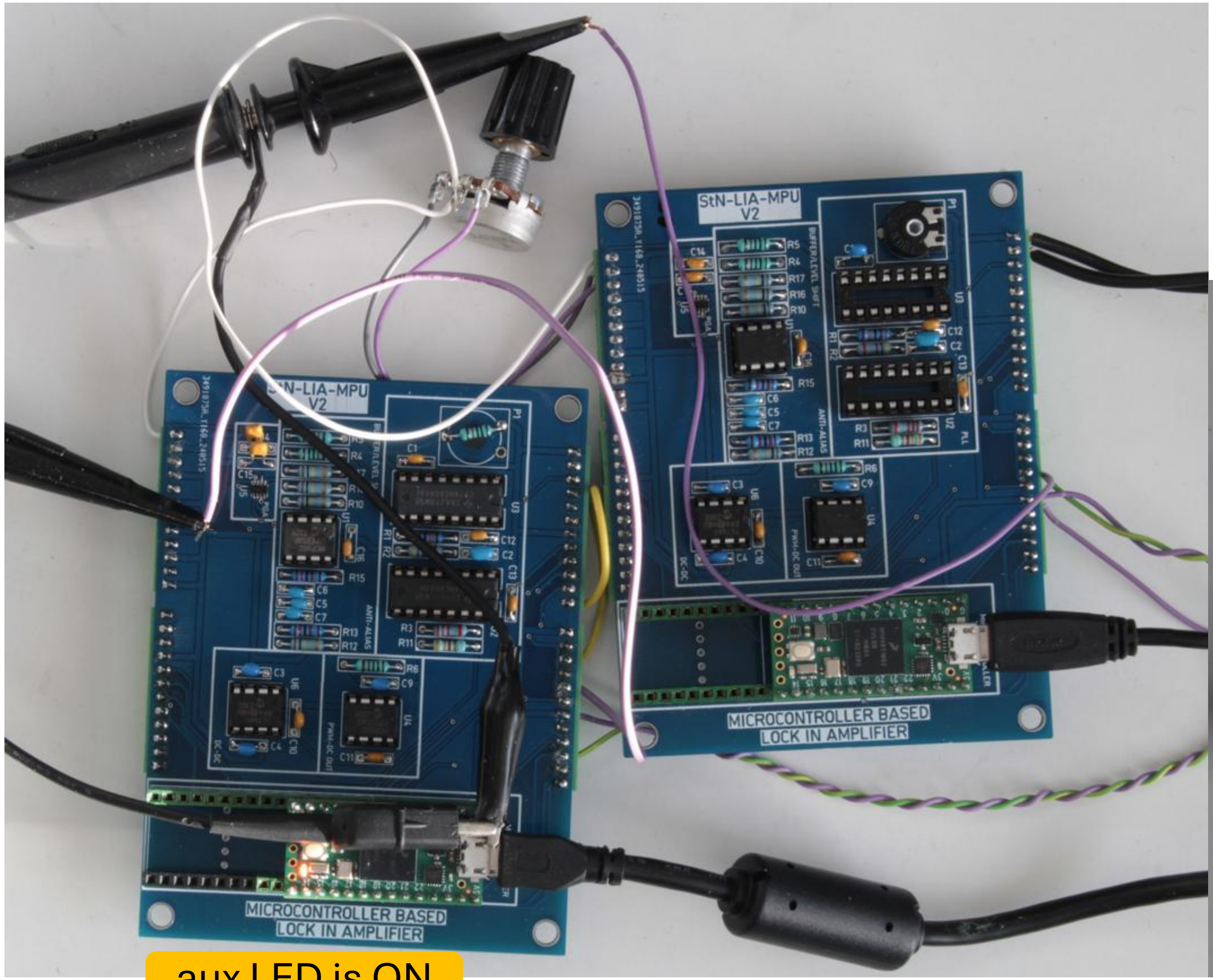
- Must measure 2 outputs from the D.U.T. in parallel
  - Implementation with 2 LIA boards that use external reference
  - Implementation with 2 signal inputs on the same LIA board, with internal reference
- Lower priority: Calibrate by measuring the PHASE Φ, at different frequencies1KHz to 10KHz

UCC Requirements discussion: 31/12/2024 call

After 31/12/2024 call

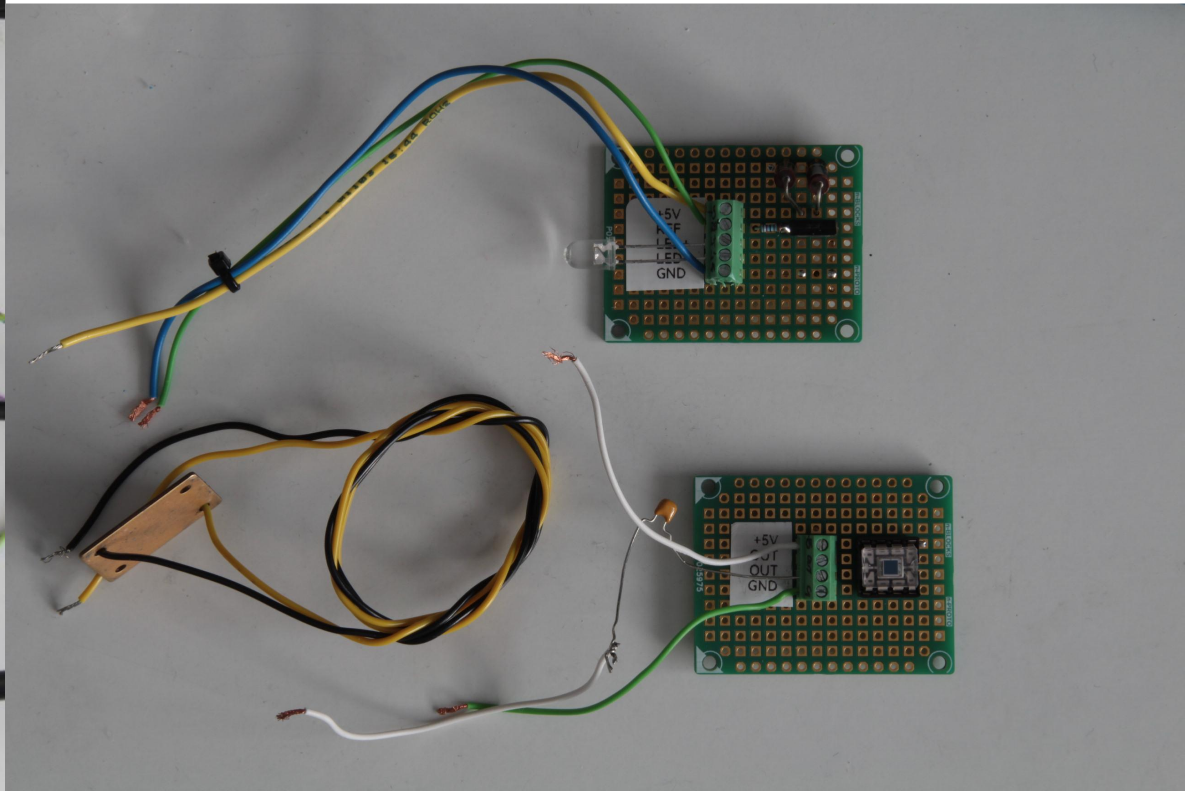
- Must measure in parallel, 2 outputs from the D.U.T.
  - Implementation with 2 LIA boards that use external reference
    - This option does not need extensive GUI updates
  - Implementation with 2 signal inputs on the same LIA board, uses only internal reference
    - Need major rework for the GUI
- Lower priority: Calibrate by measuring the PHASE ϕ, at different frequencies1KHz to 10KHz

Two PCBs tested in synchronization

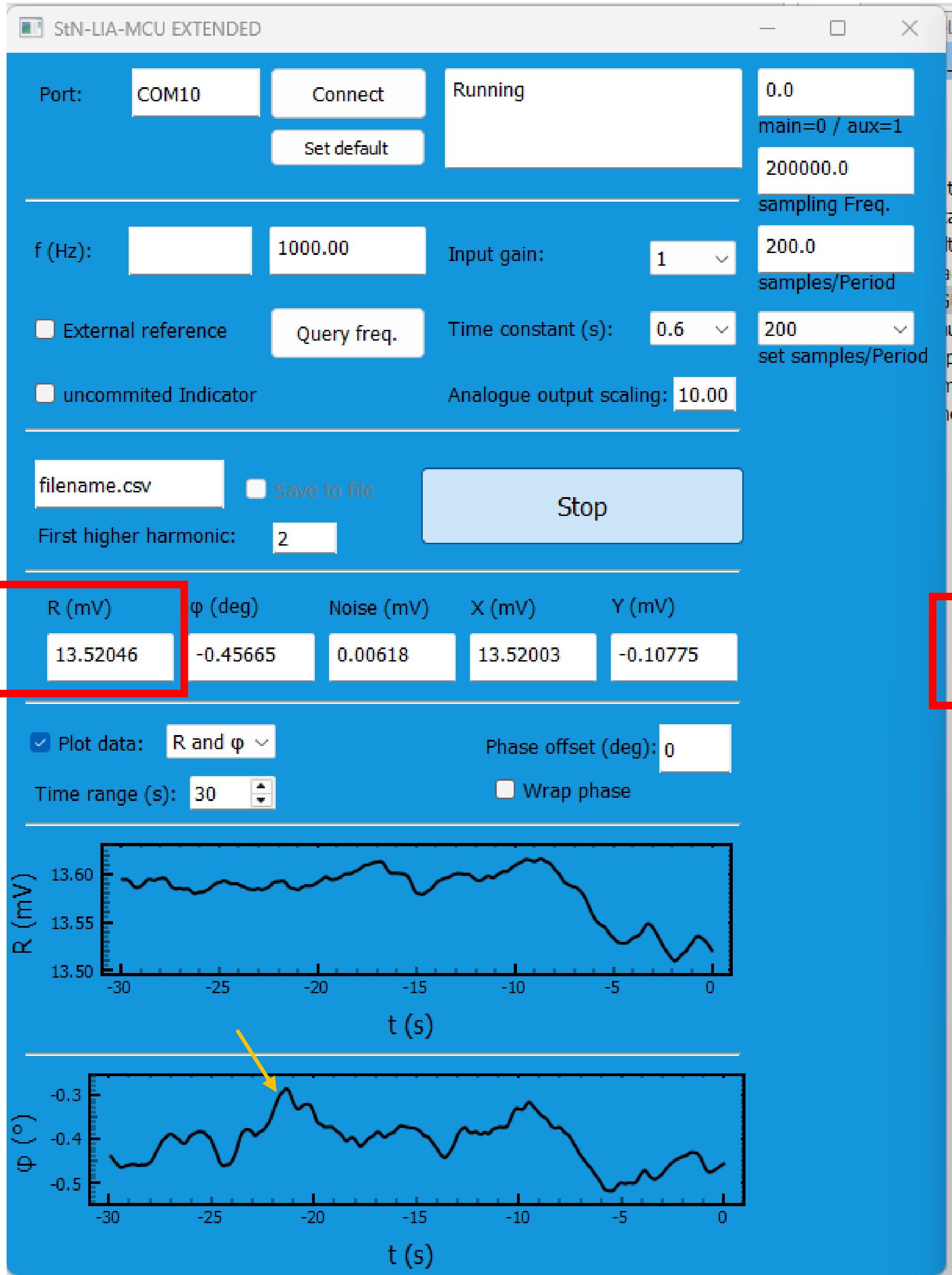


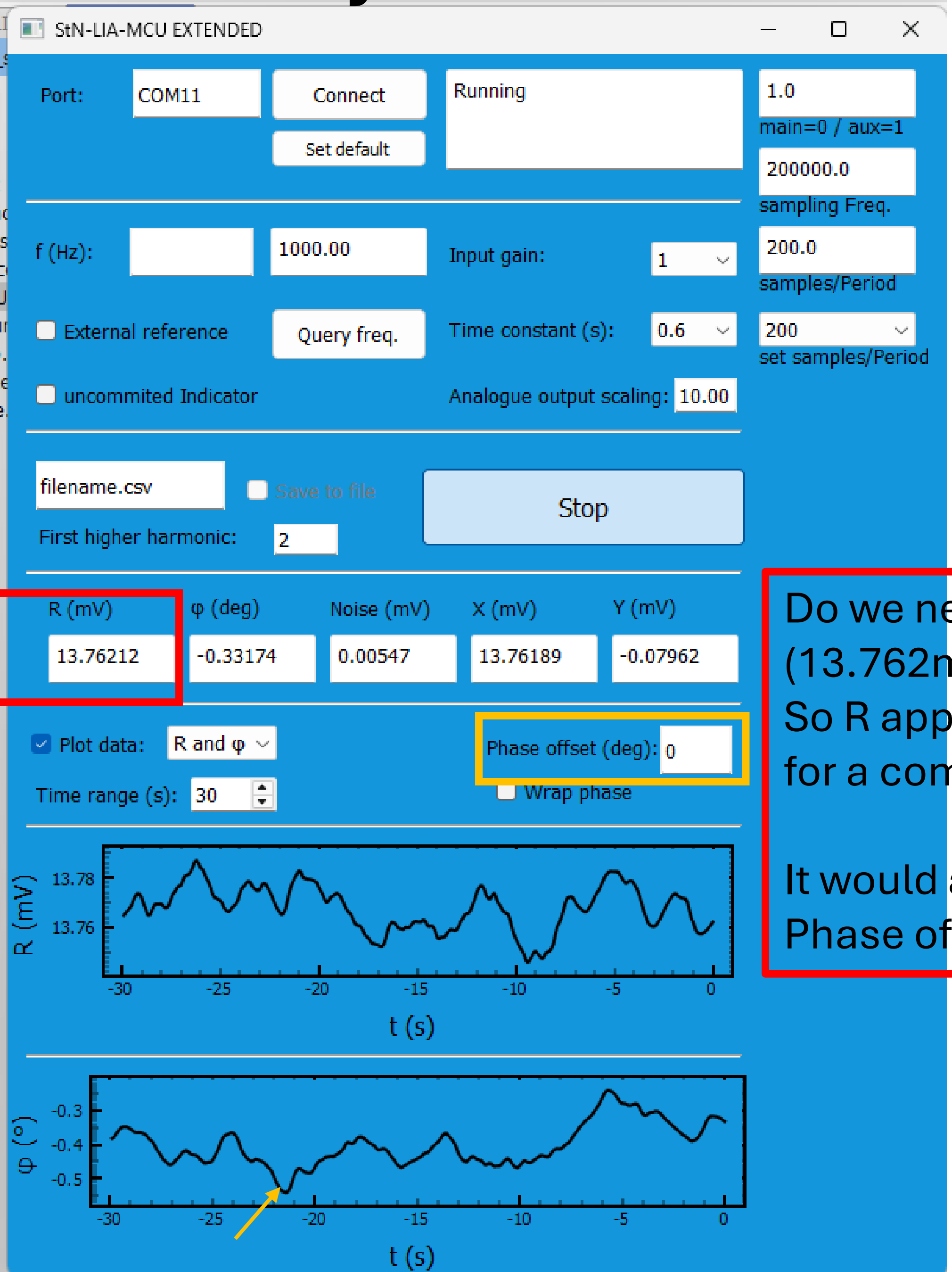
aux LED is ON

- Both PCBs have the same signal input (Potentiometer adjusted to 16mV)
- The Reference signal is created from the main PCB and drives to Potentiometer



Two PCBs tested in synchronization

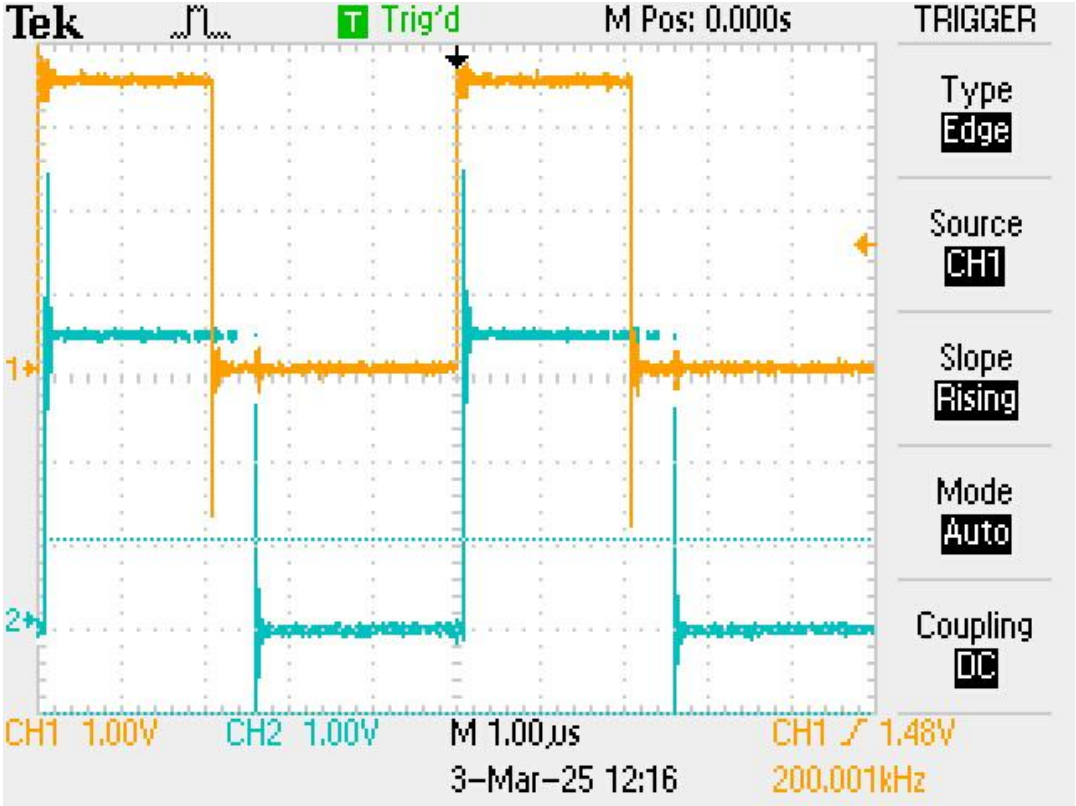


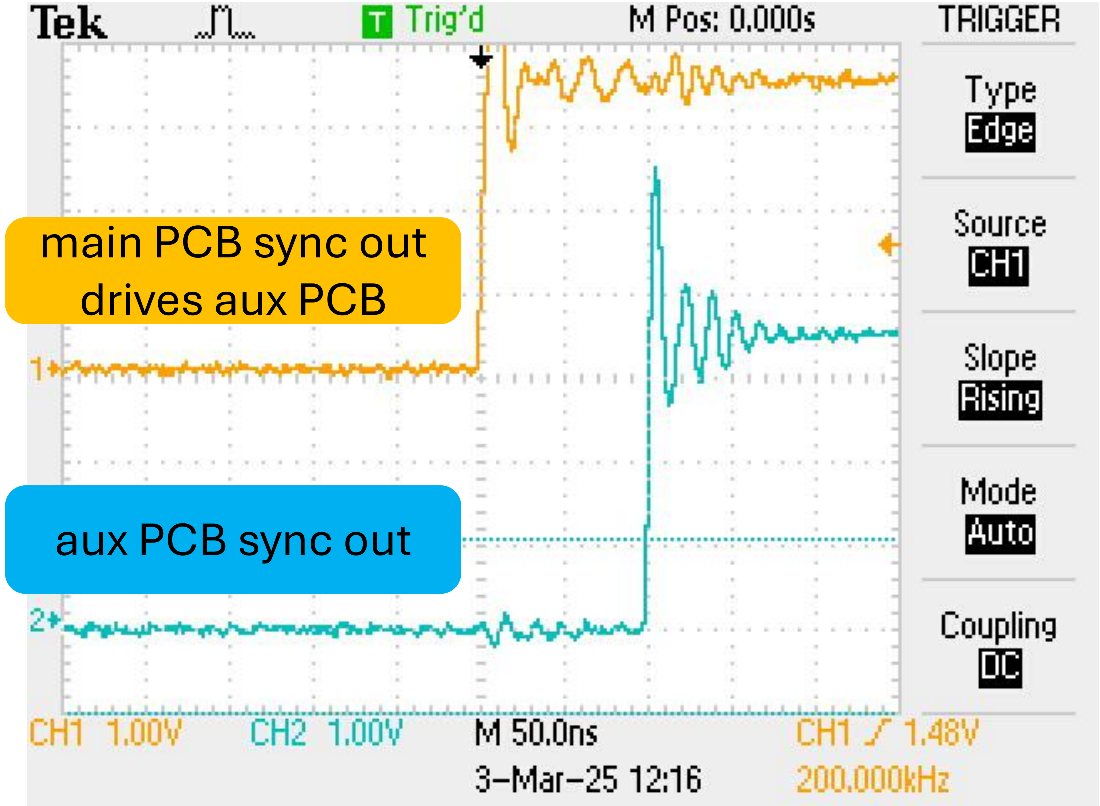


Do we need R calibration Offset (13.762mV -13.52046mV here),
So R appears exactly the same for a common input?

It would appear similar to the Phase offset **BOX**

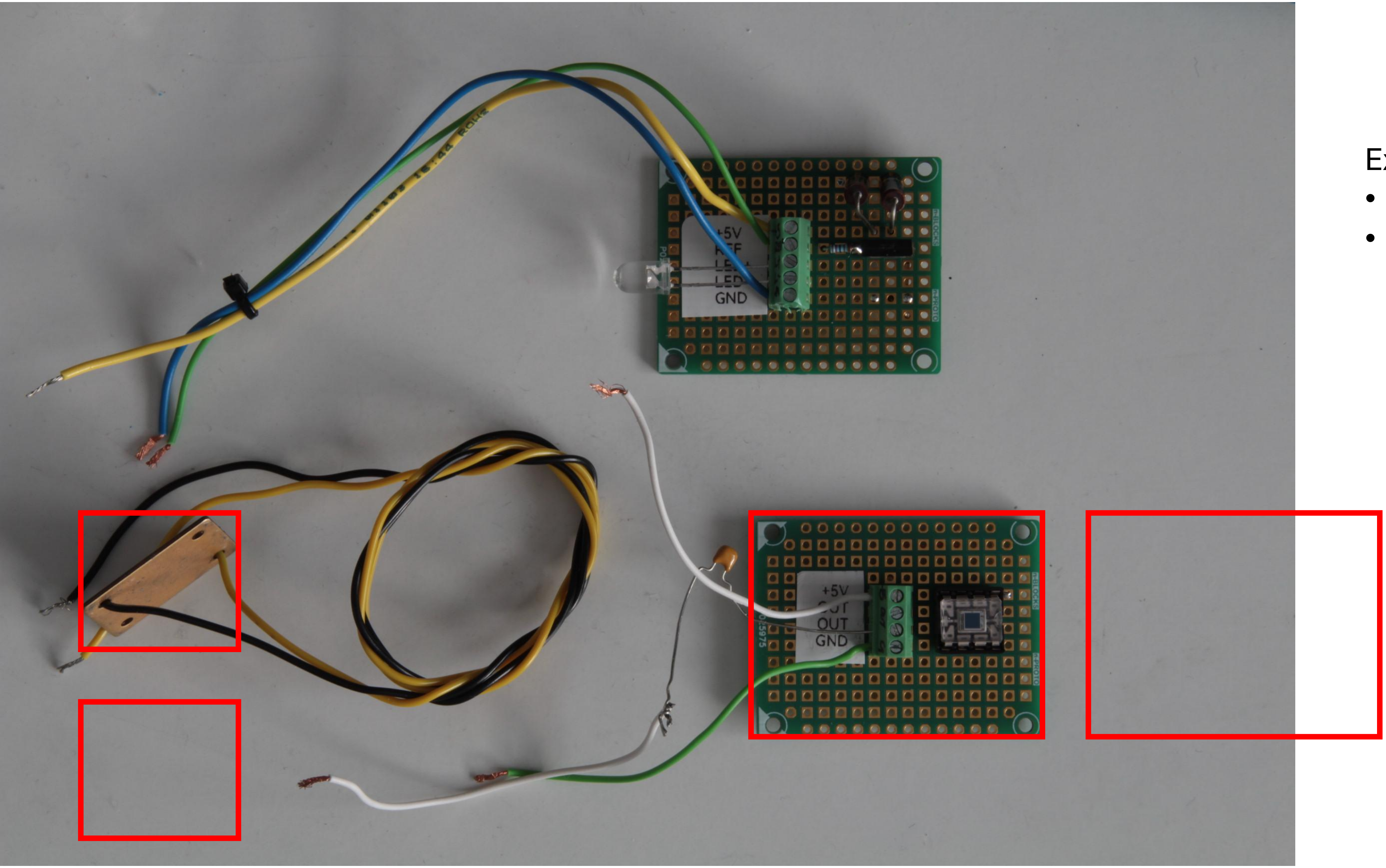
Two PCBs tested in synchronization





- The 2 PCBs are synchronized
- The aux PCB interrupt (sampling) has 100ns delay
- **The phase difference between the main and aux sampling is stable (100ns)**

OPT101 SETUP



Expand with
- 2nd OPT101 PCB
- 2nd LED **???**

QUESTIONS