Internship Report Computer Science

Making a Lock in Amplifier

Can Olmezoglu

University Supervisor: S. Wang
Company Supervisor: P. De Goeje

January, 2022

Company: Convergence B.V.
Internship Period: 6 September 2021 - 12 November 2021
Version 1.0

UNIVERSITY OF TWENTE.

Preface

I want to thank Pieter De Goeje for putting his trust on me and Valentijn Hol to work on a project that directly impacted Convergence. I also want to thank my parents for supporting me.

Contents

Making a Lock in Amplifier

Can Olmezoglu*

January, 2022

*Email: c.olmezoglu@student.utwente.nl

1 Introduction

This is an internship report done for Bachelor Internship module in University of Twente. It is about an internship conducted for the company Convergence B.V, and will cover the company, the assignment of the internship, how the assignment was done, how the academical knowledge obtained at University of Twente was utilized for the internship assignment and will end with a personal reflection.

In this report, the main objective is to convey how the internship has occurred. This will be mainly done through explaining the method and then contrasting that with academical knowledge and a reflection.

2 Description of the Company

Convergence B.V. is a company that offers designing and and manufacturing of customized fluid handling systems [2]. It was founded as a spin-off company from University of Twente in 2010. During that time, the company worked mostly on membrane characterization, but they have expanded the scale of their work following their beginning. Currently, they also work on the PLD project by delivering software and hardware assistance.

3 Project Description

3.1 Summary

The project done in this internship is a software addition for a larger European Union consortium called NextGen PLD done to make an Pulsed Laser Deposition device that can obtain information about amorphous non-crystalline materials. In order to obtain information, an *in-situ* Atomic Force Microscopy has to conducted. This requires a very precise processing software, which is the part of Convergence B.V. and was the assignment of this internship. All the technical details are explained below, and are followed by the personal tasks and goals.

3.2 Pulsed Laser Deposition

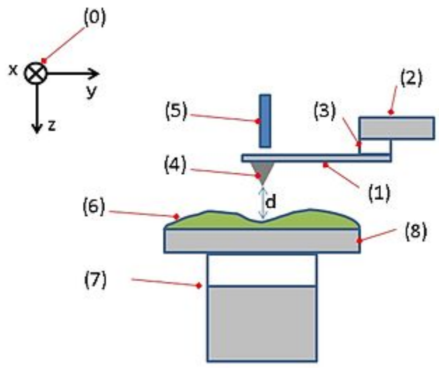
Pulsed Laser Deposition is a method where a high-powered energetic laser light hits a surface in a vacuum. This leads to the creation of deposition vapor, which can be condensed on any possible substrate [4]. The surfaces of condensed vapor can be inspected by Atomic Force Microscopy [1].

3.3 Atomic Force Microscopy

Atomic Force Microscopy is a type of Scanning Probe Microscopy, with a resolution more than a 1000 times better than microscopy conducted using optical feedback. Therefore AFM (Atomic Force Microscopy) can be used in order to see something as small as condensed deposition vapour.

The atomic force microscope, in relevance for the software part, consists of a cantilever, a piezoelectiric element for oscillation, an xyz drive and a sample. The xyz drive, shown in Figure 1 as (7), moves the sample, which is labelled as (6). As the sample is moving, the piezoelectric element, shown as (3), is oscillating the cantilever at a constant frequency. Consequently, the cantilever labelled as (1) is doing constant vertical motions with differing time spent for each motion depending on the vertical depth it has traveled. This vertical depth is visible in Figure 1, shown as 'd'.

FIGURE 1: A model of an atomic force microscope [6]



To detect changes in the sample, AFM measures how much the cantilever is deflected. This deflection is measurable from the aforementioned time the cantilever took in order to complete one vertical motion. To describe the motion mathematically, let us name the set of completed vertical motions M. For an arbitrary motion $m \in M$, the time it took to complete the motion is represented with T_m and the distance it covered is represented with D_m.The distance between the cantilever and the ground is named as D_g, and is named as 'd' in Figure 1. The height of the sample below the cantilever in motion m is represented with H_m. The height H_m can also be written as:

$$D_g - D_m \tag{1}$$

The speed of the cantilever is named as S . Therefore, the D_m of the measured point can be observed by:

$$T_m * S \tag{2}$$

Combining equations 1 and 2, it is clear that the height H_m equals to:

$$D_g - (T_m * S) \tag{3}$$

As a result, one can see that H_m of an arbitrary point of the sample can be obtained by only sampling T_m. In order to measure T_m, a generated signal is added a phase shift by certain methods too complicated for the scope of this report. However, the phase shift fundamentally represents T_m, therefore it is crucial to measure phase as accurately as possible, creating a necessity for a precise processing software. As a result, tools and theories such as a Lock in Amplifier principle needs to be implemented.

3.4 Control and Processing Software

For the core part of the project in the context of this internship, a software part for control and processing needed to be implemented. This software is essentially divided into two parts, the first one being control and integration and the second one being correlation of the values obtained from control and integration using the Lock in Amplifier.

For the control of the Atomic Force Microscope, a signal is generated at an established frequency. This generated signal and a control signal are then processed using an ADC (Analog to Digital converter) after it goes through the cantilever, in order to determine the phase shift that has occurred. In addition, all these operations have to be conducted through a Guided User Interface where the user has the option to change the signal generating frequency, start or stop the signal generation, see the results and add a DC offset.

The second part, correlation of values is done after the signal is processed and digitized. These digital values need to compared with the generated signal used as a reference and the phase shift needs to be extracted in order to obtain the H_m for the given position at the sample as aforementioned in section 3.3. As these values need to be very precise for the microscope to work, normal subtraction is not sufficient in terms of accuracy. Therefore principles such as the Lock in Amplifier principle is implemented.

3.5 Personal Tasks and Goals

The personal assignment that I was given for this internship was implementing the correlation of values as explained in section 3.4. With the aim of implementing the correlation, I needed to understand signal processing, needed to learn the Lock in Amplifier principle and collaborate with other people doing the internship so the software was integrated. In a nutshell, my goal was to learn the theory for the internship while showing an efficient collaboration with other stakeholders.

4 Methods and Results of the Assignment

4.1 Summary

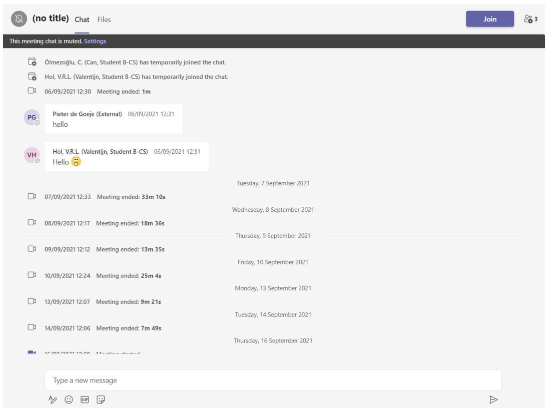
This section will explain how the assignment was done in the areas of collaboration, schedule, research and implementation. Those will be followed by the results of this method of the assignment.

4.2 Collaboration

When the project was first assigned, we agreed that some parts of the project required a practical experimentation approach that could start instantly. The other parts needed some fundamental knowledge and research before implementation. As a result, we decided to divide up the work depending on our abilities. Valentijn Hol, the other intern, had experience using Rust and QT, therefore having some directly transferable knowledge to the project. On the other hand, I had a lot of experience with Mathematics as I have taken an additional Honours in Mathematics, which we thought would make it easier to for me to understand the underlying theory for the Lock in Principle. As a conclusion, we agreed to divide up the project into the two parts mentioned in section 3.4, the control software and the correlation software. Valentijn Hol took on the control software and I took on the correlation software as our strengths matched it.

Aside from the work division, we needed to arrange the other part of collaboration, namely the communication between the other stakeholders. We decided to conduct day to day stand-up meetings with the interns and the supervisor Pieter de Goeje to handle any daily issues and align daily plans for everyone. The daily stand up meetings were conducted at a virtual room in Microsoft Teams, as seen in Figure 2. In addition, Whatsapp and Discord was used as methods of immediate communication.

Figure 2: The virtual room in Microsoft Teams



4.3 Schedule

The plan for the internship was to do the divided up work in parallel for the first 8 weeks to cover a lot of ground quickly. We deduced that to be the best option as one of the intern dropped out before the internship began, increasing the amount workload for the rest of the stakeholder. We also believed working separately would not cause issues as we had daily stand up meetings where any miscommunications could be cleared up.

My schedule was to do research for the first 2 weeks in order to understand the principle thoroughly, while Valentijn worked on to fix the old codebase given to us. In the 6 weeks following that, Valentijn was to work on the QT and the external signal generation while I implemented the lock in amplifier and we planned to conduct integration testing in the last 2 weeks.

4.4 Research

In order to do research without a predetermined source such as lectures, slides or a book, I initially tried to find material by searching the web. However, this proved to be futile as I did not have the fundamental knowledge to understand something specialized in the signal processing domain.

I decided to ask my supervisor Pieter de Goeje for book recommendations in order to understand the fundamentals of signal processing and he told me to read *The Scientist and Engineer's Guide to Digital Signal Processing* written by Steven W. Smith. As the deadline for starting the implementation was approaching, I decided to also study in the weekends to finish my job on time.

After I read the book, I found a some online resources that explained signal processing such as the application notes from Stanford Research System and I was capable enough to work on the implementation.

4.5 Implementation

We used C++ and QT to develop the software component. My job was to process the obtained signals and output the phase and the amplitude. It was done through the Lock in Principle, which will be explained in this subsection. Let's name the signal that gets phase shifted by the microscope as sig and the one that does not get phase shifted as ref . The signals can be described as

$$A_{sig} * sin(freq_{sig} * t + phase_{sig}) \tag{4}$$

$$A_{ref} * sin(freq_{ref} * t + phase_{ref}) \tag{5}$$

where A is the amplitude, $freq$ is the frequency , $phase$ is the phase and t is the time symbol for the signal in subscripts below it. The code for this internal reference signal generation can be seen in Listing 1 where $2 * Pi$ was added to work with the mathematics library of C++.

```
double SignalGenerator::getReferenceSignalSample() {
  double sineVal = 2 * Pi * frequency * time + phase;
  time += deltaTime;
  return offset + (amplitude * sin(sineVal));
}
```

Listing 1: The code for signal generation

As the signals are generated by the system, one can make $freq_{sig}$ and $freq_{ref}$ to be equal. In our system, both of them were generated to be equal. By assuming that they are equal, we can isolate the phase shift after multiplying samples of sig and ref at the same time sample.

$$(A_{sig} * sin(freq_{sig} * t + phase_{sig})) * (A_{ref} * sin(freq_{ref} * t + phase_{ref}) \tag{6}$$

$$sin\alpha * sin\beta = \frac{cos(\alpha - \beta) - cos(\alpha + \beta)}{2} \tag{7}$$

Recalling the multiplication formula for $sin\alpha * sin\beta$ as seen in Equation 7, the result for Equation 6 is below.

$$\begin{aligned} &\frac{A_{sig} * A_{ref} * cos((freq_{sig} * t + phase_{sig}) - (freq_{ref} * t + phase_{ref}))}{2} \\ &+ \frac{A_{sig} * A_{ref} * cos((freq_{sig} * t + phase_{sig}) + (freq_{ref} * t + phase_{ref}))}{2} \end{aligned} \tag{8}$$

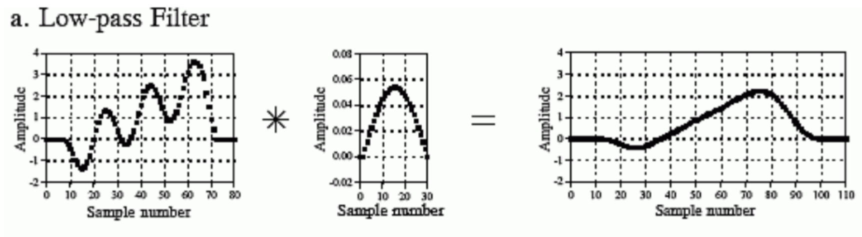
After simplification, it can be seen that the upper half of Equation 8 can be simplified onto:

$$\begin{aligned} &\frac{A_{sig} * A_{ref} * cos((freq_{sig} * t - freq_{ref} * t) + (phase_{sig} - phase_{ref}))}{2} \\ &\frac{A_{sig} * A_{ref} * cos(0 + (phase_{sig} - phase_{ref}))}{2} \text{ (since } freq_{sig} = freq_{ref})) \\ &\frac{A_{sig} * A_{ref} * cos(phase_{sig} - phase_{ref})}{2} \end{aligned} \tag{9}$$

From Equation 9, it is seen that the phase difference caused by the cantilever can be isolated after the multiplication of the two reference signals. However, in order to access that, one also needs to consider the lower half of Equation 8 and find a way to filter it out.

To filter out the lower half of Equation 8, a low-pass filter needs to be implemented. For this internship project, the implementation and understanding of a low-pass filter was obtained from the book *The Scientist and Engineer's Guide to Digital Signal Processing* mentioned in Section 4.4. The essence of a low-pass filter is to filter out frequencies higher than a certain inputted frequency while letting lower frequencies through. An example of a low-pass filter is seen in Figure 3.

FIGURE 3: A basic low-pass filter implementation [3]



There were two main low-pass filters considered for the internship, a moving average filter and a windowed-sinc filter. A moving average filter takes an average of signal samples around the given sample, and then outputs that in the place of the given sample. The output is a signal with higher frequencies partially eliminated. The implementation made as a moving average filter for the purpose of this internship can be seen in Listing 2. On the other hand, a windowed-sinc filter can perform at "incredible performance levels" but are "slow to execute" [3].

```
vector<double> MovingAverageFilter::filterWithMovingAverage(double *inputSignal, size_t inputSignalSize) {
  vector<double> filteredSignal(inputSignalSize, 0);
  uint16_t mid = floor(((float) filterSize) / 2.0);
  for (int i = mid; i < inputSignalSize; i++) {
    for (int j = -mid; j < mid + 1; j++) {
      filteredSignal[i] = filteredSignal[i] + inputSignal[i + j];
    }
    filteredSignal[i] = filteredSignal[i] / filterSize;
  }
  return filteredSignal;
    }
}
```

LISTING 2: The code for the moving average filter

In order to choose between a moving average filter and a windowed-sinc filter, we have tested the completed Lock in Amplifier with using both values. In order to see the gap between them, after the correlated phase was obtained, it was tested based on a margin for around 10000 samples each. In listing 3, the testing process can be seen. The line starting from "REQUIRE" tests whether the correlated value obtained is within the margin to the perfectly expected value. After these tests it was concluded, a windowed sinc filter was chosen for a more accurate filter.

```
for (int x = kernelSize + 1; x < result.size() - kernelSize - 1; x++) {
  double ref = expectedResult.getCosReferenceSignalSample();
  REQUIRE(result[x] == Catch::Approx(ref).margin(margin));
}
```

LISTING 3: The code for a moving average filter

In spite of the fact that the filter was able to clear the upper half of Equation 8 to some degree, the phase value could not be extracted directly from $0.5*A_{sig}*A_{ref}*cos(phase_{sig}-phase_{ref})$. Therefore it was needed to do some more calculations. As aforementioned, ref was assumed to be a sinusoidal signal. However, due to a property of trigonometric identity explained in Equation 10, it can be seen that sin value can be converted into a cos value easily.

$$sin(\alpha) = cos(\frac{\pi}{2} - \alpha) \tag{10}$$

In order to convert the resulting signal into a cos value, a duplicate of ref is created and named as $ref2$. The phase value for $ref2$ named $phase_{ref2}$, is then given the value $phase_{ref} + \frac{\pi}{2}$ with the rest of the values being equal to the values of ref. Therefore if the multiplication operation done in Equation 6 was done with sig and $ref2$ instead of sig and ref, after a low-pass filter the final resulting signal would look like [5]:

$$0.5 * A_{sig} * A_{ref} * sin(phase_{sig} - phase_{ref}) \tag{11}$$

If the signal described in Equation 12 was named Y and the signal described in Equation 9 was named X, then one could obtain the phase shift by:

$$tan^{-1}(\frac{0.5 * A_{sig} * A_{ref} * sin(phase_{sig} - phase_{ref})}{0.5 * A_{sig} * A_{ref} * cos(phase_{sig} - phase_{ref})}) = \quad tan^{-1}(\frac{X}{Y}) = phase_{sig} - phase_{ref} \tag{12}$$

using the basic trigonometric law $\frac{sin\alpha}{cos\alpha} = tan\alpha$. In addition, by calculating the magnitude of the signal vector, the amplitude of the signal can be obtained [5], as seen below:

$$(X^2 + Y^2)^{\frac{1}{2}} = A_{sig} \tag{13}$$

The code implementations made for the equations 12 and 13 can be seen in Listing 4.

```
vector<double> psdCosVector = getPSDVector(inputSignal, inputSignalSize, true);
  vector<double> psdSineVector = getPSDVector(inputSignal, inputSignalSize, false);
  int size = psdCosVector.size();
  for (int i = 0; i < size; i++) {
    currentPhase += (atan2(psdSineVector[i], psdCosVector[i]));
    currentAmplitude += (sqrt((psdCosVector[i] * psdCosVector[i]) + (psdSineVector[i] * psdSineVector[i])));
  }
  *phase = (currentPhase / size) * RadToDeg;
  *amplitude = currentAmplitude / size;
```

LISTING 4: The code for getting the phase shift and amplitude

To conclude the implementation part, here it was seen how the lock in principle was fundamentally implemented for this internship. One other element that was not discussed but had a massive part during the internship experience is the testing.

As explained in Section 4.3, most of the work for the internship was decided to be done in parallel and with a lot of collaboration. In order to explain some code concepts, writing tests are sometimes more intuitive, and therefore we decided to use a variant of Test Driven Deployment for the project. Usually before classes were written, a test was added with the expected capabilities in the case that other stakeholders did not understand the utility of the code immediately. An example code snippet of a test for the code shown in Listing 1 can be seen in Listing 5.

```
TEST_CASE("A signal sampled at 0,90,180,270,0 with phase shift", "[sinewave]") {

  SignalGenerator signalGenerator = SignalGenerator(1, 1, 90, 4);

  REQUIRE(signalGenerator.getReferenceSignalSample() == Catch::Approx(1).margin(0.5));
  REQUIRE(signalGenerator.getReferenceSignalSample() == Catch::Approx(0).margin(0.5));
  REQUIRE(signalGenerator.getReferenceSignalSample() == Catch::Approx(-1).margin(0.5));
  REQUIRE(signalGenerator.getReferenceSignalSample() == Catch::Approx(0).margin(0.5));
}
```

LISTING 5: The code for testing signal generation

These tests were very helpful when a glitch was encountered in integration testing during the last 2 weeks of the internship. To explain further, the system real life was not getting the same values as testing, and by changing the tests a bit we were able to understand and overcome the issue. From past experiences, a issue faced in system integration normally has taken more time to fix.

In conclusion, to be able to correlate phase and amplitude accurately as planned in Section 3.5, the lock in principle was followed with initiative taken in the choice of method for the low pass filter in a variant of Test Driven Deployment.

4.6 Results

After applying all these methods, I was able to create a lock in amplifier software which was able to obtain the phase and amplitude in all of the simulations. Furthermore, I was then able to use my newly obtained knowledge to solve some problems that occurred during the integration testing in the last 2 weeks.

In addition to this, I worked at Convergence at the 2 months following my internship, and I am able to confirm the software was able to extract the amplitude and phase despite some delay. Therefore it could be concluded that the methods described in this chapter were sufficient in the end of the project.

5 Applying academic knowledge and skills

5.1 Summary

In this section, how academic knowledge was applied in order to implement the developments will be discussed. The rest of the subsections will follow the categories of the given assessment criterion and will make a case on how they were fulfilled.

5.2 Applying scientific and technical competencies

As explained in the first mandatory internship meeting in October 1st, there were not a lot of directly overlapping material with the assignment of the internship and courses taken until the beginning of the internship at University of Twente. However, a few technical capabilities were overlapping, such as knowledge of the intricacies of a compiler taught in the eighth module Programming Paradigms and trigonometric functions taught in Calculus 1B.

In programming paradigms, the final project was to make a programming language from scratch using some parsing tools. This led to a more fundamental understanding of programming languages. Using this knowledge, I was able to understand C++ quicker and was able to start coding right after the research part in the schedule.

In Calculus 1B, basic trigonometric functions and complex numbers were taught. The fourth aim of the course guide for Calculus 1B states the student should be capable enough to *"work with complex numbers"*. This knowledge was directly utilized in implementing and understanding the Lock in Amplifier principle as described in Section 4.5.

On top of the directly overlapping coursework, the primary help of an academic approach was the ability to conduct research. To explain further, upon my conversations with the part-time workers from Saxion University, I saw that they were given more practical tasks compared to our internship assignments. Therefore, it was clear that due to the more academical education of University of Twente, graduates had more abilities to create a solution using academic research, and it was a primary reason why we were given the title "Research and Development Engineer" in out title for the internship contract. I applied this ability and experience of researching gained from the University of Twente to research the fundamentals of signal processing and the Lock in Amplifier principle.

5.3 Contributions to the project work

The contributions I personally made to the project work was mostly working on the theoretical part of extracting the phase shift of the cantilever as explained in Section 4.3. What made this contribution unique for the project was that there were not many people contribute using Electrical Engineering and Computer Science together for a bachelor internship. As agreed on during the first internship meeting in October 1st, our topic did not really correlate much with Computer Science. It was my job to conduct the research for the parts related to signal processing and Electrical Engineering.

I was able to understand the signal processing enough to be able to improvise solutions and debug in the last 2 weeks of the internship. In addition, I coded the amplifier at a professional level using my already existing Computer Science knowledge. In a nutshell, my contribution was to use my additional experience from taking Honours classes in Mathematics and researching a complicated mathematical principle and being able code and debug it.

5.4 Planning the work in the available time

To plan the work in the available time, I used my experience from the projects of Module 4 and 5. In addition to this, I did a previous internship I did where I gained insights on my working and coding speed. Using that we created a plan as described in Section 4.3 which was then executed successfully.

5.5 Learning to work a professional setting

In order to get used to working in a professional setting, I went to the office for the first 2 weeks to mentally get ready for a different environment. From those trips, I saw the different environment of work, where the efficiency between working hours is more important than the amount of work. To explain further, my normal daily study routine included a lot of random times where I start focusing and lose focus. However, in work I had to keep my focus synchronized with coworkers as collaboration matters more than individual output. Therefore I adjusted my rest times to match with others. On top of that, in the beginning of the internship I practiced my part before the daily stand up meetings so I was able to convey my daily progress more clearly.

6 Reflection

6.1 Summary

In this reflection, three core points that can be reflected upon has been chosen, which are academic reflection, social reflection and professional reflection. Academic reflection will be about the new scientific and technical competencies that have been obtained from this internship, social reflection will be about what I have learnt in a social context from working professionally and professional reflection will be about what I have learnt upon my role as a trainee in the company.

6.2 Academic Reflection

After doing this internship, I have learnt about the C and C++ languages and about the theory of signal processing. C++ despite being regarded as more difficult compared to Java and Python, actually proved to be very interesting with the amount of low-level control it has offered.

I did not find the theory of signal processing very interesting as it seems to be more about Physics and Electrical Engineering compared to Computer Science. I personally feel more comfortable in abstract and creative domains, which I felt was not the case for signal processing theory. This was different compared to my thoughts before the internship, as I said I wanted to do an internship in signal processing as I thought it would be similar to modules 5 and 3 from University of Twente. In conclusion, after doing the internship I had a better understanding of my academic capabilities.

6.3 Social Reflection

As I have worked as a teaching assistant for multiple modules and therefore communicated with professors in a professional environment, I thought I had a lot of experience in professional communication. Nevertheless, after communicating with my supervisors I realized that in a work environment, talking practical results is important. Whereas in an academic environment your creative tendencies and willingness to try new things matter more than being practical. For example, when I was a Teaching Assistant, I talked more about my process and approach for grading papers rather than how many papers I graded. The gap between professional and academic communication was also visible in the quick meetings that were conducted in the internship, where you needed to explain what you did and what problems you are facing very concisely. Therefore, when I am working professionally I now try to be as practical as possible in my communication and talk about what I have delivered.

Asking for help was a core point of feedback I have received from my supervisor Pieter de Goeje. When we were halfway done with the project, I asked whether the was something I could have done better and got the answer that I could ask for help directly rather than waiting for the daily meeting of the current or the following day. Following these, I now contact my group members immediately whenever I am confused with something.

6.4 Professional Reflection

My role in Convergence was more versatile than what i expected it to be. I was expecting my role to be more clear, as in my previous internship I was just developing software. However, in Convergence, my role started as a researcher, then it became a developer and following that I was offered to work as a supervisor to other interns. From this I have realized that I could partake in many roles within a company and my abilities were pretty versatile.

On the other hand, one clear weakness that I have seen was that I was not able to take initiative as fast as the other interns did. For example, Valentijn was able to start a project directly and write code, whereas I needed more guidance. As a conclusion, I am now working on making hobby projects in order to be able to think on my feet and take initiative quicker. I also learned that I would not be suitable for single person assignments without any clear guidelines as that requires being very self reliant.

7 Conclusion

In conclusion, this internship has led me to learn about areas such as C++ or Signal Processing which I was knowledgeable about previous to itself, and therefore I have learned a lot about myself, my strengths and how school has helped me when I was dealing through those challenges. In addition to this, I learned more about working in a professional setting, which I assume will help me in the rest of my life. I aim to find another internship or a company to work for in the future where I can use these newfound experiences and insights.

References

[1] Hans M Christen and Gyula Eres. Recent advances in pulsed-laser deposition of complex oxides. *Journal of Physics: Condensed Matter*, 20(26):264005, 2008.

[2] Bart Merkus. Convergence b.v. about us. Website, February. 12, 2020 [Online].

[3] Steven W. Smith. *The scientist and engineers guide to digital signal processing.* California Technical Pub., 1997.

[4] Solmates. Pulsed laser deposition explained. Website, 2021 [Online].

[5] Stanford Research Systems. About lock-in amplifiers, 2021. [Online; accessed 5-January-2022].

[6] Wikipedia contributors. Atomic force microscopy — Wikipedia, the free encyclopedia, 2021. [Online; accessed 5-January-2022].