

UNIVERSITY OF GRONINGEN
OPTICAL CONDENSED MATTER PHYSICS

PROJECT INFORMATION TECHNOLOGY

Measuring the duration of short laser pulses

Software to be used with autocorrelators in order to measure the duration of
laser pulses

A.J. Oostra

OCMP

Prof. Dr. Ir. P.H.M. van Loosdrecht

Supervisor: Ir. A.F. Kamp

October 8, 2009

Lindenlaan 33, 9713PD, Groningen
E-mail: J.Oostra@gmail.com
phone: +31616008394

student number: 1750232

Abstract

Many optical setups include objects that will cause chromatic dispersion, a process that changes the duration of ultrashort laser pulses. Because one would like to know what the duration of a laser pulse is after it went through a specific measurement setup, it is useful to be able to determine this duration. This can be done with use of interferometric autocorrelators. The output of these autocorrelators needs to be read with use of computersoftware.

This paper shows three different programs that can be used in combination with an interferometric autocorrelator to determine the duration of laser pulses. The first program is used for calibration, the second program is used for verification, and the final program is the "main-program" for the duration measurements. The programs have two requirements: They have to be fast (<5 seconds / measurement) and accurate (< 5 fs difference between the results of the verification and final program).

The final program fulfilled both of these requirements: a measurement can be done within 2 seconds, and give a result which, when verified, lies within a 5fs margin. There is however a dependency with the amplitude of the speaker used within the autocorrelator. This dependency is slightly caused by the calibration files, and may also have a cause within the data rate used to measure with. It is however unlikely this explains the full differences that are seen when the speaker amplitude is increased. Further research into this is required, but lies outside the scope of this report.

Contents

1	Introduction	1
2	Background information	2
2.1	Ultrashort pulses	2
2.2	Autocorrelator	3
2.2.1	Michelson Interferometer	3
2.2.2	Interferometric Autocorrelator	4
2.3	LED sensor	4
3	Software	6
3.1	Calibration software	6
3.2	Final program	7
3.3	Verification program	9
4	Results	11
4.1	Calibration program	11
4.2	Verification program	11
4.3	Final program	12
5	Conclusion	15
6	Future Reference	16
	Appendix A Calibration Program	17
	Appendix B Final Program	18
	Appendix C Confirmation program	19
	Appendix D Program results	20
	References	24

List of Figures

2.1	Michelson interferometer.	3
2.2	A simple model of an intoferometric autocorrelator.	4
2.3	Example of a pulse emitted by a 80MHz laser.	5
3.1	Autocorrelator output for a HeNe laser	7
3.2	Output of the autocorrelator when a 1-kHz pulsed laser is used	8
D.1	Calibration file for 0.4V	20
D.2	Calibration file for 0.5V	20
D.3	Calibration file for 0.6V	21
D.4	Comparison of the voltages in the calibration data	21
D.5	Averaged FWHM vs time	21
D.6	FWHM averaged over 10, vs time	22
D.7	Measurements as a function of time and frequency	22
D.8	Time required to average n measurements	22
D.9	Speaker amplitude vs FWHM	23
D.10	Continues measurement of the FWHM.	23
D.11	FWHM as a function of speaker frequency	23

List of Tables

4.1 FWHM as function of the calibration files	13
---	----

CHAPTER 1

Introduction

Ultrafast lasers are used in various setups. These lasers emit high-energy, short-duration pulses which then pass through a setup of choice. As the laser pulse passes through the various elements of the setup, the duration of the laser pulse will change due to chromatic dispersion (explained in section 2.1). Being able to measure the duration of a pulse, after it went through a measurement setup, is therefore useful since one will know how the original pulse was changed. The purpose of this paper is to show a program which can be used, in combination with an autocorrelator, to determine the duration of laser pulses.

There is some theoretic background information required before the program can be explained. This information can be found in chapter 2, which shows, among others, how an autocorrelator works. The following chapter then explains the workings of three different types of software: A tool for calibrating the "main-program", the "main-program" and a program that can be used to verificate the "main-program". The results of using these various programs are then shown in chapter 4, followed by the conclusion and a brief recommendation for future reference.

As with any project, certain requirements were set. The software had to fulfill the following two: the program had to be both fast as well as accurate. Since fast is a rather relative term, it had been defined as fewer than 5 seconds per measurement. In order to test the accuracy, the difference between the found pulse duration of the "main-program" and the verification program had to be less than 5 fs. If it fulfilled this requirement, it would be seen as accurate.

CHAPTER 2

Background information

Ultra-fast lasers are required to produce ultra-short laser pulses. The most important types of ultrafast lasers are:

- Titanium-sapphire lasers
- Diode-pumped lasers
- Fiber lasers
- mode-locked diode lasers

^[1]

The lasers that was used is a Titanium-sapphire laser. This type of laser has the potential of generating laser pulses down to 5 femtoseconds, with a very high peak power. The output power of these lasers are generally in the order of 0.3-1W with a typical pulse-repetition rate of 80MHz ^[2]. Our laser was specified to generate pulses of 120 femtoseconds. This chapter will explain what happens to the ultrashort pulses as they pass through various optical elements, followed by a section on how to measure these pulses with use of an autocorrelator. The chapter is then concluded with a section on the sensor of the autocorrelator.

2.1 Ultrashort pulses

When pulses emitted by lasers are in the order of tens of picoseconds, up to the femtosecond range, they are usually called ultra-short pulses. These pulses are generated by, for example, the lasers mentioned previously, and are further characterised by a high peak intensity ^[3].

The pulses have a broad optical bandwidth, which means that there is a large difference in the upper and lower frequencies of the laser pulse. Because of this broad optical bandwidth, there is a problem with chromatic dispersion in lens materials. This means that as light travels through an optical medium, the phase velocity and the group velocity depend on the optical frequency.

There are two types of dispersion: material dispersion and waveguide dispersion. We speak of material dispersion when the wavelengths of the pulse interact with the refractive index of a material. This is what happens when light passes through a prism, and the colors get seperated. The dispersion is called waveguide dispersion

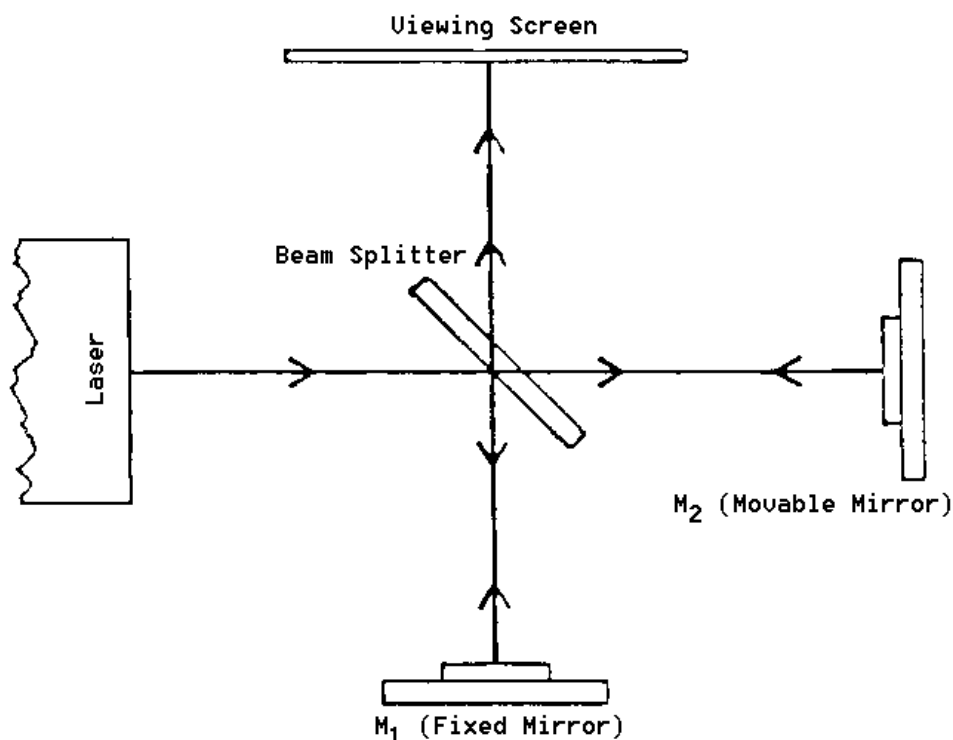


Figure 2.1: A schematic overview of a Michelson interferometer. ^[6]

when the phase and group velocities wavelengths depend on geometric properties rather than the refractive index of the material. This is for example the case in photonic crystals. ^[4,5]

Because of this dispersion, it is useful to be able to measure the duration of a pulse after it went through a measurement setup. This can be done with an autocorrelator, which will be explained in the following sector.

2.2 Autocorrelator

The autocorrelator that was used in our measurements was an interferometric autocorrelator. This means that the autocorrelator contains an michelson interferometer. The first subsection will explain how the interferometer works, and the following subsection than explains how it is used within the autocorrelator.

2.2.1 Michelson Interferometer

An interferometer is a device that utilizes the effect of interference. In the Michelson interferometer (shown in figure 2.1), a single beamsplitter is used for splitting and recombining the beams. In this case, an interference pattern is created by splitting a beam with the beamsplitter, reflecting the beams back, and recombining them. The different paths may differ in length and this will create an interference pattern.

In case the difference in path lengths is a whole number of wavelengths ($0, 1..n$), there will be constructive interference. If they differ by a whole number and half a wavelength ($0.5, 1.5, \dots$) there will be deconstructive interference. In case of constructive interference, the measured signal will be strong, while it will be weak for deconstructive interference.

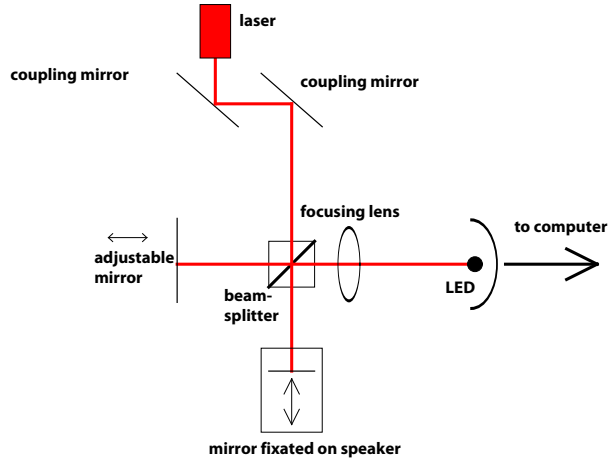


Figure 2.2: A simple model of an interferometric autocorrelator. Light passes through the beam splitter, is reflected and later combined with part of the original beam. As the reflected part is re-united with the part of the beam that went directly to the sensor, interference will occur.

2.2.2 Interferometric Autocorrelator

An interferometric autocorrelator is a Michelson interferometer with an object, such as a non-linear crystal, placed at the output in order to generate the second harmonic. This generated second harmonic can give us information about the phases of the pulse. The entire process is called interferometric autocorrelation.

The setup that was used does not have a non-linear crystal. A LED was used instead. There are a few reasons for using a LED over a non-linear crystal: a LED is less expensive, easier to use and a LED does not change the pulse duration due to chromatic dispersion. Since the amount of dispersion can only be lessened by making the non-linear crystal thinner, which in turn affects the measured second harmonic, it is easier to use a LED. With a LED one can make use of two-photon absorption, a process explained in the next section.

Figure 2.2 shows a schematic overview of an interferometric autocorrelator. The lens focuses the beam onto the LED, which functions as a sensor. The setup that was used has a few additional elements. First it has a quartz plate. This is because the beams do not travel an equal amount of times through the quartz of the beam-splitter if there is only one. By placing a quartz plate at the correct position, both beams pass through the quartz an equal amount of times. Secondly we have a few diaphragms placed within the setup, for easier alignment.

2.3 LED sensor

A LED can be used as a sensor because of two-photon absorption, a non-linear process where two photons are absorbed simultaneously and leave an ion or atom in a higher lying state. The energy increase is then equal to the sum of the photon energies and because the absorption coefficient is proportional to the optical intensity, the process only occurs when the optical intensity is high.^[7] The required nonlinear result is then only found when two-photon absorption takes place. This is because the energy of a single photon is smaller than the bandgap of the LED, while the

combined energy is larger than the bandgap.

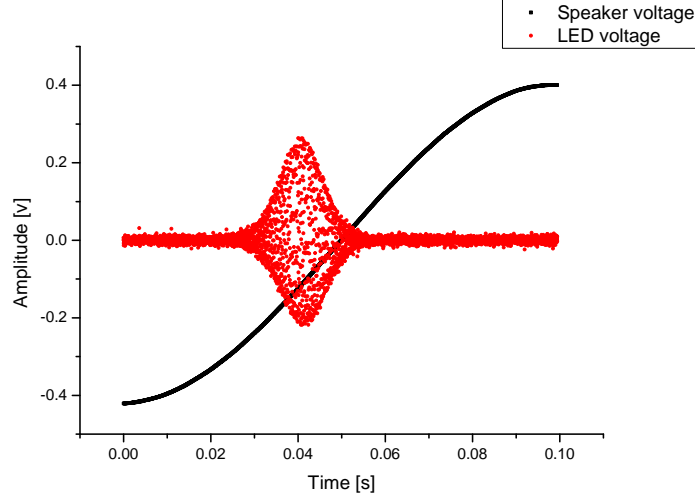


Figure 2.3: Example of a pulse emitted by a 80MHz laser.

When using a pulse from a 80-MHz laser, the LED will return an output as shown in figure 2.3.

In this figure we see both the generated laser pulse, as well as the speaker voltage. The LED-pulse can be moved along the x-axis (time-dependent) by adjusting the distance between the fixed mirror and the beamsplitter. By adjusting this distance, the pulse can be "placed" on a specific region along the slope of the speaker-voltage.

When determining the duration of a laser pulse, the term "Full width at half maximum" (FWHM) is used. This parameter gives the distance between points on a curve, at which the function reaches half its maximum value. The FWHM is often used to describe the width of a curve. In case of the laser pulse shown in figure 2.3 (looking at only positive values) the FWHM can be calculated by describing the curve as a gaussian function (equation 2.1) and using equation 2.2

$$f(x) = a * e^{-\frac{(x-b)^2}{2c^2}} \quad (2.1)$$

a is the height of the peak, b is the position of the center of the peak, and c is the width of the peak its bell. c is also related to the FWHM of the peak, by equation 2.2.

$$FWHM = c * 2 * \sqrt{2 * \ln(2)} \quad (2.2)$$

The FWHM we calculate with equation 2.2 is the FWHM of the autocorrelation width, which is $\sqrt{2}$ longer than the intensity width. To deconvolute this FWHM, it must therefore be divided by $\sqrt{2}$.

CHAPTER 3

Software

Labview has been used to build a program that can determine the duration of the laser pulse. The input for the software is the voltage of the LED-sensor, which is dependent on the displacement of the speaker. Since it is unknown how much the speaker will displace, the software will have to be calibrated. This can be done with the calibration software, which will plot the voltage as a function of the optical displacement, with use of a HeNe laser. This program will be explained in section 3.1.

The calibration program is used to write a calibration file for the final program. The final program is a program that can measure laserpulses produced by a 1-kHz laser and a 80-MHz laser. Its working is described in section 3.2. This program is then verified for a 80-MHZ laser with use of a verification program, explained in section 3.3.

3.1 Calibration software

The output voltage of the LED needs to be calibrated and plotted as a function of the displacement of the mirror. This is done with use of a continuous wave HeNe laser. The laser will be aligned with the autocorrelator, and the LED-output will be as shown in figure 3.1. The fringes each correspond with a speaker displacement of half the wavelength ($\lambda = 632\text{nm}$).

The calibration software creates a calibration file by counting these fringes, multiplying them with the wavelength, and plotting them as a function of the speaker voltage. From this we can see the optical displacement as a function of the measured voltage. The program has been attached as HeNe.exe and its source-files can be found in the folder "HeNe calibration". Its overview is shown in appendix A.

The calibration program works within a message-queue-handler. This is a loop that constantly remains in an idle state, but can receive messages and then process certain cases/commands. The start-command is given while the program remains in the idle-state. The program then goes to a case which will measure the voltage that is applied to the speaker. From this input, it determines the frequency and the amplitude of the voltage. After measuring this, the case sends the command "calibration" to the idle-case, which then proceeds executing the calibration case. The idle-case is used in case the user wants to stop the program inbetween cases. By going to the idle-case, one does not have to complete the full program but can

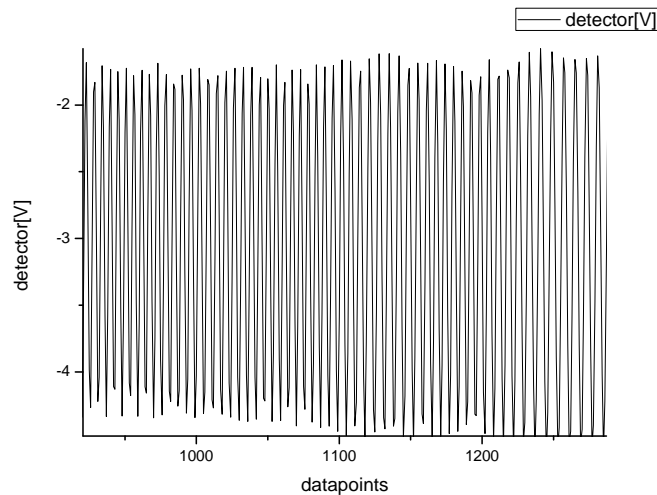


Figure 3.1: Autocorrelator output for a HeNe laser

interrupt it in the middle.

The calibration-case is the "main-case" of the calibration program. It reads the voltage of both the speaker and the LED on the upgoing slope of the speaker voltage (triggered). It determines the point where $V=0$ on the speaker voltage, and looks for peaks within the LED-signal. Then it splits both signals at the point where $V=0$. (this in order to make a reference point). It counts the number of peaks, and for each numbered peak it attaches the corresponding speaker voltage. So for example, there is an array with peak number 1 at 0.1V, peak number 2 at 0.2V etc. The number of peaks is then multiplied with the wavelength of the laser, and then plotted in a graph versus the voltage of the speaker. The user can define in the program how many times he wants the measurement to be saved. So when the program is set to saving the data, a "save" command may be sent to the idle-state so the measurement is saved. After saving the measurement x times, it is possible to plot the average of these saved signals with the "plot average" button. This button opens a case that reads the data in for n files. It combines the data of the various arrays and makes it one size. It then plots the averaged data and saves it to a file, which can be used in the other programs.

3.2 Final program

The core of the final program is a subVI that can fit a formula through a collection of data points. In this case, the formula is the gaussian function shown in equation 2.1.

The subVI will fit the function through the collected data points, and give the values for a, b and c . a is here the height of the peak, b is the position of the center of the peak, and c is the width of the peak its bell. c is also related to the full-width-half-maximum of the peak, by equation 2.2.

The fitting-subVI is the core of the program, however, before it can be used several steps need to be taken. When measuring one pulse of a 1-kHz laser, not enough data points are collected to determine the shape of the pulse. Because of this, multiple pulses need to be measured before the data can be sent to the fitting-subVI. Another problem is that when one pulse is measured, the data points are

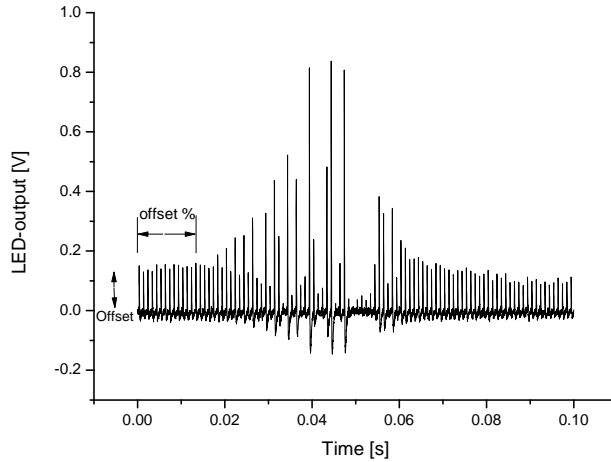


Figure 3.2: Output of the autocorrelator when a 1-kHz pulsed laser is used. Not all measured points are found on the envelope of the pulse.

not always on the envelope of the pulse. This is illustrated in figure 3.2.

So, in order to not affect the curve-fitting process, only data points that are positioned on the envelope of the pulse can be used. Finally, there is a certain offset for the pulse, which can also be seen in figure 3.2. This offset also needs to be removed.

The sequential overview of the final program is shown in appendix B. The first step of the program is to determine the amplitude and frequency of the speaker-signal, followed by opening a calibration file. A percentage has been assigned which will be used to determine what part of the signal must be used for an offset-determination. With use of the percentage assigned, the program calculates how many data points will be used. Then it gets to the point where it will either summarize n measurements, or do a continues measurement. In both cases, the program does the same steps. However, when summarizing n measurements, the program will do one measurement, but before plotting the data it will start with a new measurement and combine both results. After doing this n times, the results are plotted, curve-fitted, and the FWHM will be calculated and shown. The process will be re-started after this.

In case of the continues measurement (running average), the data will be collected, plotted and memorized. The collected data continues expanding, increasing the accuracy of the measurement.

The measurement process is the same for both cases: The LED- and speaker-signal are read in by the DAQ-card, which combines it into one datastream. This combined datastream is splitted into a seperate LED- and speaker-signal. The voltage-signal of the speaker is used to find the corresponding optical displacement, with use of the calibration file. This displacement is then converted to femtoseconds.

For the LED-signal, the vertical offset is determined, and removed from the signal. The signal is then filtered, by allowing only data points that have a greater value than the previous value. This is done for the signal as well as the inverted signal. This gives both the upgoing and downward slope of the signal, with only values that are positioned on the envelope of the signal. Both of these signals are recombined (up- and downward-slope) before being placed in an array with the converted speaker-signal.

This new data is then combined with older data from the shift-register. The data is sorted, and again only increasing values will be allowed. The data is then placed in the shift-register for following iterations. After placing it in a shift-register, the data is used to plot a graph of the pulse before being used within the fitting-subVI. With use of the fitting-subVI, the FWHM is determined and shown.

3.3 Verification program

To verify the results of the final program, a verification program has been written. While this program can not be used for the 1-kHz laser, it works for a 80-MHz laser. This because the 80 MHz-signal can be measured with such resolution that it is possible to see the interference fringes, just as with the HeNe laser.

Each measured fringe corresponds to an optical displacement of the wavelength ($\lambda = 800nm$). The pulse duration can be measured by counting all the fringes that are found within the full-width at half maximum of the signal. This measured pulse duration can be used to verify the results of the final program because the result is completely independent of the calibration file and the measured speaker-voltage. By using this program and comparing it to the results of the final program (while using a 80-MHz laser) it will be shown whether the results of the final program are correct.

The program outline is shown in appendix C.

The program is similar to the calibration software in that it also has an idle-state from which it executes several cases. The first case is executed by the "start" button. In this case the program cleans out all the arrays, and sends the command "determ. freq" to the idle-state. In the "determ. freq" case, the program does a measurement on the speaker-signal, and determines from this signal the frequency and the amplitude of the speaker-voltage. It also opens a calibration file and places its values in an array for later use. After these things have been done, the command "measure" is send to the idle-state.

The "measure" case is the case in which the FWHM is determined for each iteration. This is done by first determining the amount of samples required for a measurement, along with the amount of samples needed for the peak-detection subVI's. The LED- and speaker-signal are read in by the DAQ-card, and combined in one datastream. The program then splits this datastream into a seperate LED- and speaker-signal. The speaker-signal is converted to femtoseconds with use of the calibration file, and forms the basis of an femtosecond x-axis. The unconverted signal is also plotted in a graph. The LED-signal is also plotted in a graph, and then plotted as a function of the femtosecond x-axis.

Meanwhile, the amount of peaks and valleys are measured within the LED-signal. A peak-detection subVI is used for this, with as requirement that the peak/valleys must be larger than the maximum/minimum value of the LED-signal, divided by two. The amount of peaks/valleys is converted to femtoseconds, with use of the wavelength of the laser, and placed in an array. During this process, the speaker-voltage is found that corresponds to the first and last peak/valley found. These values are looked up in the calibration file, and the corresponding displacement is converted to femtoseconds and also placed in the array. After this is done, the command "temporary" is send to the idle-state.

In the "temporary"-case, multiple measurements are combined in order to find an averaged result. The user sets a number, N of required measurements before determining the average of the results. The program will add the new measured data to the older data in case $n \leq N$. If this is not the case, the program will empty out the arrays, reset the iteration-counter, and send the "measure" command to the idle-state.

After adding the new data to the older data, it is checked whether $n = N$. In case this is false, it means more measurements need to be made. This is done by sending the "measure" command to the idle-state. In case this is true, it means that enough measurements have been done and that the average of these measurements can be determined. In order to do this, the array needs to be reshaped into a form that can be used by the average-subVI. After reshaping the array, it is send to this subVI, and the average and standard deviation are determined. This gives two results for both the average and standard deviation. The first result is calculated with use of the calibration file and the minimum and maximum voltage at which the first peak/valley was determined that passed the $y = 1/2 * Vmax$ limit. The second average found is determined with use of the laser wavelength and the number of peaks/valleys that were found. The standard deviations each correspond to one of these averages. The data is plotted and the averages and standard deviations are shown in numeric indicators. Finally the command "measure" is send to the idle-state.

CHAPTER 4

Results

The following chapter will list the results of the various programs. The figures discussed in the various sections are listed in appendix D. This chapter will use the following terms:

- Data point: Measurement over one single pulse.
- Measurementset: Averaged collection of n data points.
- Measurement: One measurementset.

4.1 Calibration program

After using the calibration program, several calibration files were made as shown in figure D.1 to D.3. From these graphs it is clear that slightly before the 0-V position, the graphs appear to be similar for the same voltage, independent of the used frequencies. After the 0 V position, the higher frequency (15Hz) moves away from the other 2 lines, which remain similar. Therefore it is best to measure in the region below 0V for the frequency-dependent measurements. There the frequency has less impact on the measurement, so it will matter less if the frequency is not exactly 5, 10 or 15Hz.

In figure D.4 the voltages are compared for the 10Hz signal. Again the signal is most linear in the region just below 0V. The lines however do not divert from each other after 0V, while they do in the beginning of the graph. This could have as possible effect on the measurements that if the voltage is not exactly 0.4V, 0.5V or 0.6V that there may be variations in the found FWHM if the pulse is mostly positioned on the region where the speaker voltage is lower than 0V. In this case the measured FWHM would increase, as the voltage is increased.

4.2 Verification program

The verification program was run several times, during verification of the final program. The results of the final program measurements and the comparisons made with the verification software are shown in the next section. The verification program results had a difference of less than 5 fs per measurement, and were used

to highlight a few calculation errors made in a beta version of the final program. Because the verification program can either use or not use the calibration file, both results were also checked against each other. Each time the results were within a 5 fs difference.

4.3 Final program

After creating the calibration files, the final program can be tested in two different setups. The first setup is when a 1-kHz laser is used. The second setup is when a 80-Mhz laser is used. This second setup can also be confirmed with use of the verification software. Figure D.5 to D.11 list all the results of the final program. Each graph will be discussed as it comes along.

The first graph with results of the final program is figure D.5. In this graph the FWHM is measured over time. This has been done for 4 different options, related to the amount of data points that is averaged each time within a measurementset. The different options were 5, 10, 15 and 20 data points in one measurementset. The amount of data points that are averaged should affect two aspects of the measurement. The first is the accuracy of the measurement, and the second is the time required per measurement. When we look at figure D.5 we see that over time, the result does not change much. We also see that indeed the accuracy is better when using more data points to average over. This is all as one would expect.

Next we look at the time required per measurement. This is shown in figure D.8. The figure shows a clear linear function that can be fitted through the data points, and that for each increasement in data points to average over, the required time per measurementset increases with roughly 1 second. This linearity is also as one would expect.

Figure D.6 shows again the FWHM as a function of time, when averaged over 10 data points. This time it is however a bit more clear how the FWHM behaves when measured for longer times. The result is similar as the results previously discussed and requires therefore no further explanation.

The next figure is figure D.7 and shows the required time to measure n measurementsets of 10, for three different frequencies. From this figure it is clear that when the frequency is increased, the measurement-speed increases as well. This is not unexpected, since the program is triggered by the upgoing slope of the speaker-voltage. When the frequency is increased, the program measures more frequently. Since the program is triggered more often in the same time, more measurements can be made. From this graph we can also see that in 2 seconds, the program can do one measurement of a measurementset of 10 data points when the speaker is run with a frequency of 5Hz and amplitude of 0.4V. From figure D.6 we can see that with 10 averaged data points, the program is able to give an accurate result. This means that within 2 seconds the program can give an accurate result.

In figure D.9 we see the FWHM as a function of the amplitude applied to the speaker. We see here that as the applied amplitude is increased, the measured FWHM increases as well. It would be expected that the FWHM would remain the same, independent of applied speaker amplitude. The increase in FWHM is also not small, but instead rather significant. The errors of the various data points do not even overlap. For an explanation of this, we would probably need to look at the calibration file.

While the voltage applied to the speaker may not have been 0.4, 0.5 or 0.6V exactly, the small variations in the voltage would probably not give such a significant change in the measured FWHM. The calibration curve however does divert more from the "combined area" that we see slightly below 0V as we draw closer to -0.4, -0.5 and -0.6V. To see if this is the cause in deviation of the measured FWHM,

the FWHM can be measured with several different calibration curves, while the data remains the same. This has been done for a set of stored data with speaker amplitude 0.4V and frequency 5Hz. The stored data was of several pulses emitted by the 80-MHz laser. The pulse that was shown in figure 2.3 is an example of one of the pulses that was sent to the programs. Doing this, gave the results in table 4.1 for the various calibration files. The measurement was repeated with the verification program, which can work with as well as without the calibration file. These values are also shown in the table. We see that in each case that the calibration file is used, the FWHM increases with amplitude.

FWHM			
Calib. file	Final program result	verification calib. result	verification non-calib. result
0.4V	86.14 fs	89.84 +/- 3.29 fs	92.4 +/- 3.60 fs
0.5V	88.07 fs	92.52 +/- 3.30 fs	92.4 +/- 3.60 fs
0.6V	90.52 fs	95.55 +/- 3.33 fs	92.4 +/- 3.60 fs

Table 4.1: FWHM as function of the different calibration files. The submitted data was each time exactly the same, but the calibration files give different values for the FWHM. Results are compared to the verification program calibration result and the verification program result that does not make use of the calibration data.

From these results we see that even if the exact same data is submitted to the program, the values of the measured FWHM increase as a calibration file is used which was calibrated at a higher voltage. We can see a deviation for the different calibration curves below the 0V region. Positioning the laser pulse in a region where the different calibration curves deviate from each other, could result in a measured FWHM affected by the selected calibration curves. The calibration file itself is not exactly correct, and when measuring at higher voltages the pulse may be positioned differently compared to the speaker-voltage. If it's positioned in the wrong direction (closer to the -0.6V) the results will change even more. This might explain why the measured FWHM is different for different speaker amplitudes. This could be solved by positioning the measured pulse differently related to the speaker voltage (after 0.0V), so that the calibration curves are exactly the same even though the speaker amplitude is different. Or by using a different speaker that has more similar calibration curves for different voltages. So, in conclusion: At -0.1V, the speaker should have the same displacement, independent of whether the full amplitude is -0.4, -0.5 or -0.6V. When this is not the case, the measured results will vary. In order to solve this, the FWHM must be measured on a position where the calibration curves all give the same displacement for the same applied voltage. If this is not possible, a different speaker needs to be used. Since the position that was measured on during our measurements was mostly around 0V, leaning towards the -0.x V this most likely caused some difference in the measured FWHM for different speaker amplitudes. However, it is not expected that it caused the full difference in the measured results. Other differences may be caused by using the same data rate and amount of samples the DAQ-card must measure. If the amplitude increases, the distance between the data points increase, decreasing the amount of data points to measure the laser pulse, which remains the same size. Or, in other words: there are less data points to measure the laser pulse with, if the amplitude is increased. This could result in a lessening of the accuracy. It is unclear if these two factors are enough to cause the large differences in the measured results. Further research into this will be required, but falls outside the scope of this report.

Another aspect we can see from table 4.1 is that when looking at the data calculated with use of the calibration files, that the results are generally in agreement with each other, since the difference is mostly below 5fs. This shows that the results

of the final program are accurate, although dependent on using a good calibration file and proper settings. The results for the verification program is similar whether the calibration file is used or not, for the calibration file at 0.5V. Assuming this is the "best" calibration file to use, we see that all the data overlaps with a 5 fs margin, again showing that the results measured in the programs appear to be accurate.

Finally we look at the effect of the speaker frequency on the FWHM. From figure D.11 we see that the frequency has no real effect on the FWHM, which is as expected.

CHAPTER 5

Conclusion

At the beginning of the project two requirements were set: The measurement had to be accurate as well as fast. Fast was here defined as less than 5 seconds per measurement, and accurate was defined as less than 5 fs difference between different results. The written final program fulfilled these requirements. The program was "fast", with 2 seconds per averaged measurement of 10 samples, in case the speaker had a frequency of 5 Hz, and the voltage applied had an amplitude of 0.4V. Running the speaker at 5Hz was the slowest setting to run it at. Increasing the frequency would increase the speed of the measurement.

The second requirement was that the program was accurate. This was tested with use of a verification program, when the laser pulses were created by a 80Mhz laser. Table 4.1 showed the outcomes of the programs when the same data was supplied. From these results it was clear that both programs gave similar results, with less than 5 fs difference. The results were however dependant on the calibration file that was used. Positioning the laser pulse (in time) in such a way that it uses the correct area of the calibration file is therefore important for finding meaningful values for the full-width-at-half-maximum. In conclusion this means that the written programs function correctly, and that the final program gives a fast as well as accurate result, as long as the pulse is positioned in such a way that it uses the proper area of the calibration file, and that proper settings are supplied.

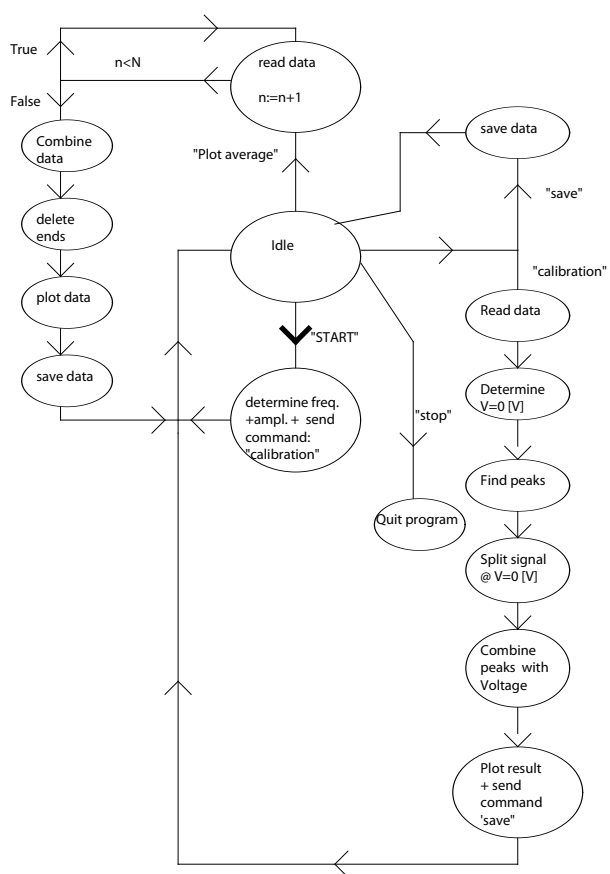
CHAPTER 6

Future Reference

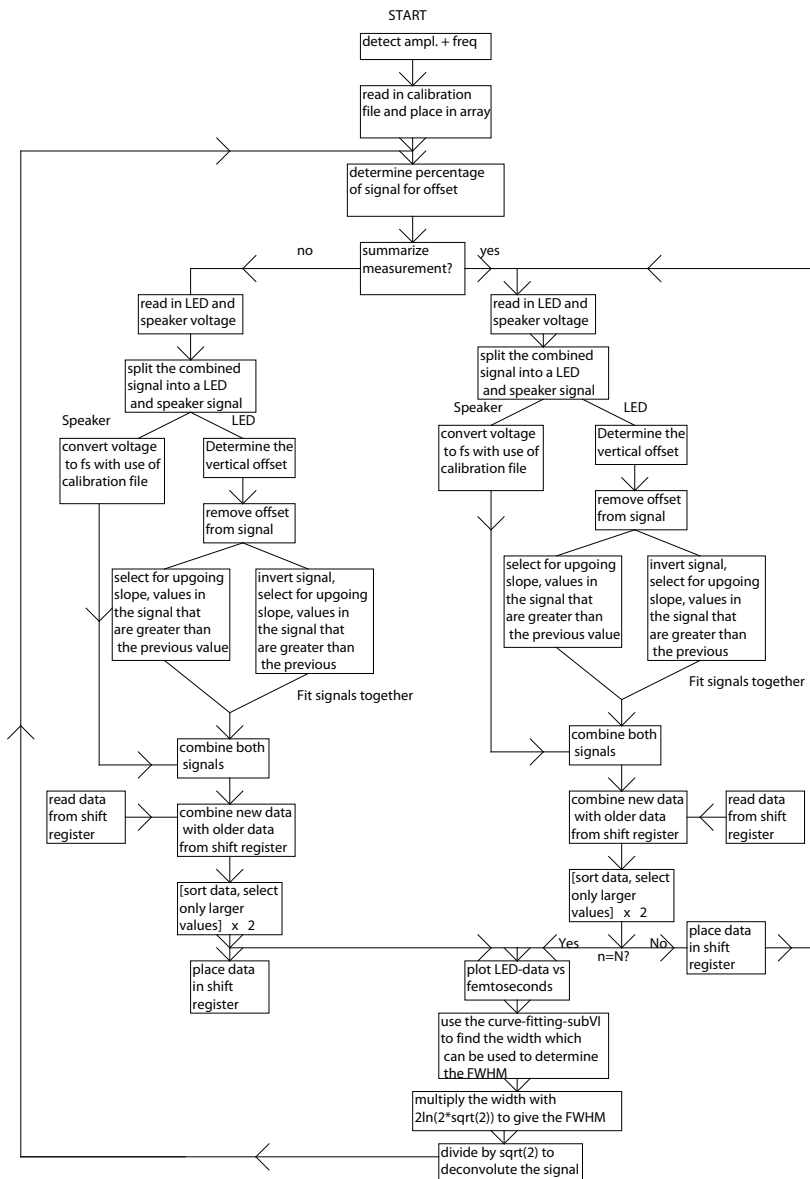
For future reference, it would be worthwhile to look into the amplitude-dependency. It is unexpected that the calibration curve and the possible difference in amount of data points to measure a laser pulse with, are enough to cause the differences in the measured results of the final program. Looking further into this amplitude-dependency might give a solution to make the program more accurate, and more precise.

Appendix A: Calibration Program

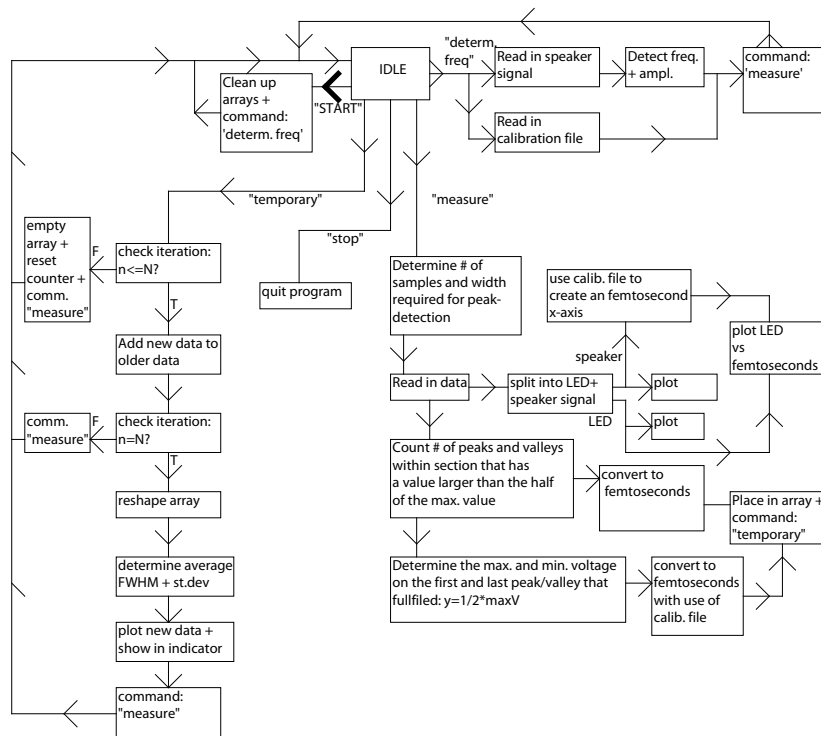
Calibration Software



Appendix B: Final Program



Appendix C: Confirmation program



Appendix D: Program results

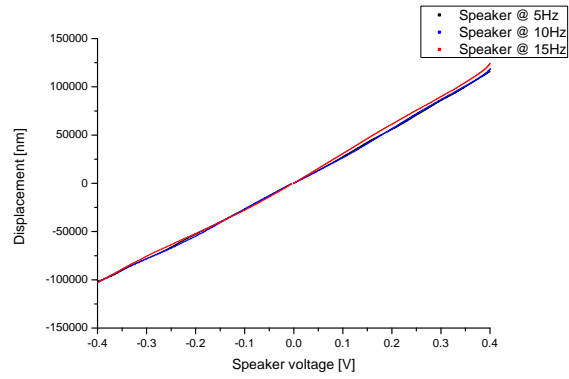


Figure D.1: Calibration file when a sinusoid voltage with amplitude 0.4V was applied to the speaker, with different frequencies.

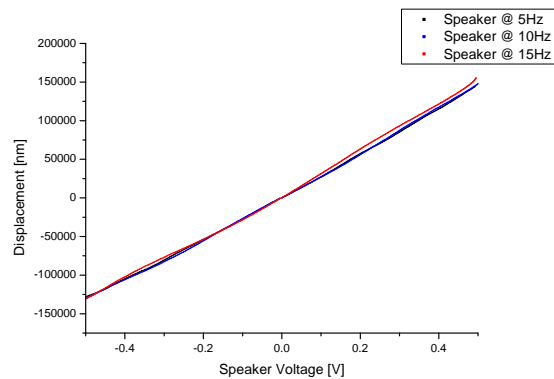


Figure D.2: Calibration file when a sinusoid voltage with amplitude 0.5V was applied to the speaker, with different frequencies.

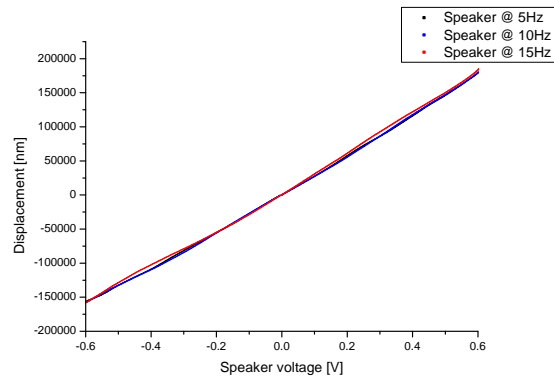


Figure D.3: Calibration file when a sinusoid voltage with amplitude 0.6V was applied to the speaker, with different frequencies.

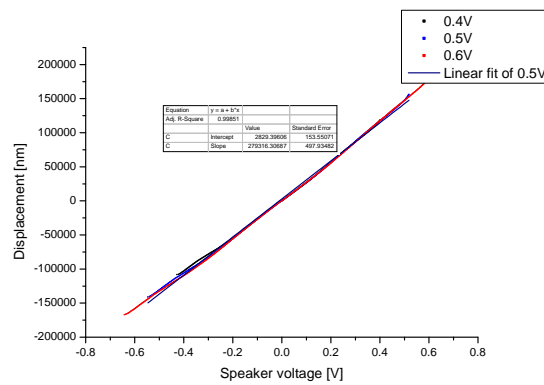


Figure D.4: A combined plot of the different voltages at 10Hz, along with a linear fit of the 0.5V signal.

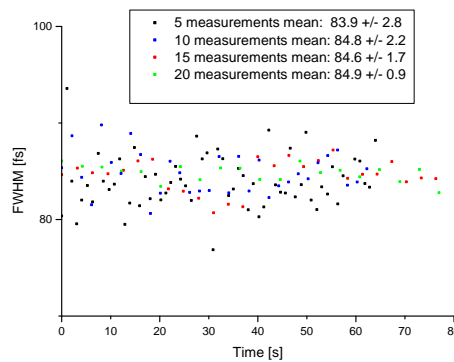


Figure D.5: The FWHM functioned as time, when averaged over either 5, 10, 15 or 20 measurements .

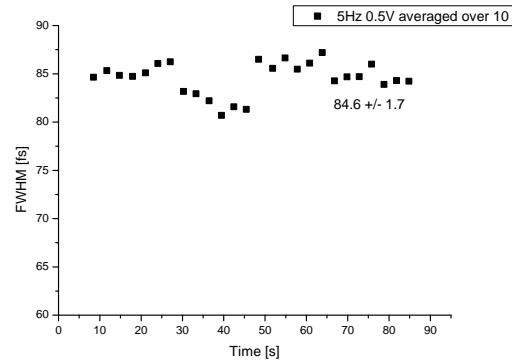


Figure D.6: FWHM averaged over 10 measurements, set as a function of time. In this measurement the frequency of the speaker was 5Hz, and the amplitude 0.5V.

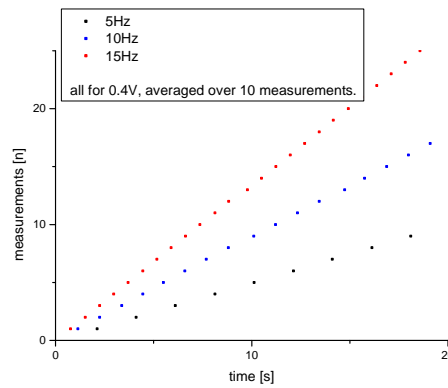


Figure D.7: Time required for n measurements, depending on frequency. All measurements were done at 0.4V, and 10 data points were averaged per measurementsets.

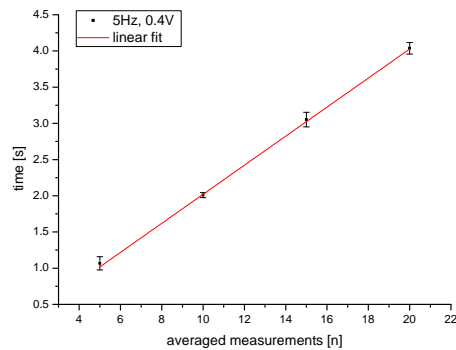


Figure D.8: Time required to measure and average n measurements, along with the error. All measurements were done with a speaker at 0.4V and 5Hz.

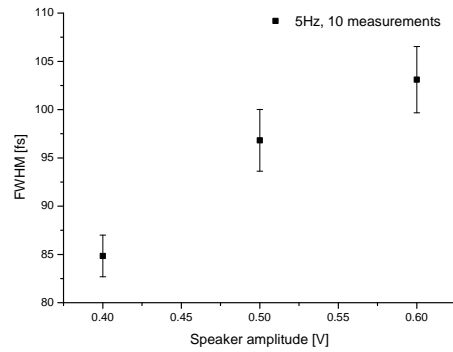


Figure D.9: The FWHM as a function of the speaker amplitude. Measurements were done with a speaker at 5Hz. Each data point consists of > 30 measurements of measurementsets of 10.

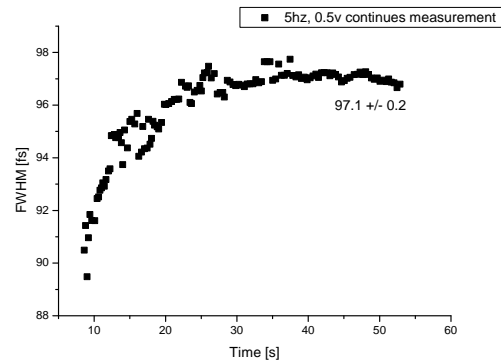


Figure D.10: The FWHM as a function of time when the program is in continuous measurement mode. The speaker was set at a amplitude of 0.5V and a frequency of 5Hz

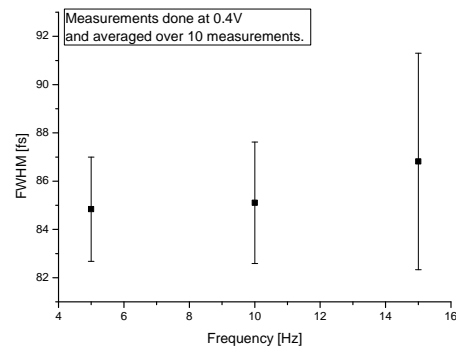


Figure D.11: FWHM as a function of speaker frequency. Each measurement was averaged over 10 samples and the speaker had an amplitude of 0.4V

References

- [1] D. R. Paschotta, Encyclopedia of laser physics and technology: Ultrafast lasers, http://www.rp-photonics.com/ultrafast_lasers.html.
- [2] D. R. Paschotta, Encyclopedia of laser physics and technology: Titanium-sapphire lasers, http://www.rp-photonics.com/titanium_sapphire_lasers.html.
- [3] Wikipedia, Ultrashort pulse, http://en.wikipedia.org/wiki/Ultrashort_pulse.
- [4] Wikipedia, Dispersion (optics), [http://en.wikipedia.org/wiki/Dispersion_\(optics\)](http://en.wikipedia.org/wiki/Dispersion_(optics)).
- [5] D. R. Paschotta, Encyclopedia of laser physics and technology: Chromatic dispersion, http://www.rp-photonics.com/chromatic_dispersion.html.
- [6] P. B. Allen, Michelson interferometer, http://felix.physics.sunysb.edu/~allen/252/PHY251_Michelson.html.
- [7] D. R. Paschotta, Encyclopedia of laser physics and technology: Two-photon absorption, http://www.rp-photonics.com/two_photon_absorption.html.