

# Laser Intensity Stabilisation

Semester Thesis

Caspar Giehr

Frieder Lindenfesler

Supervisor

Prof. Jonathan Home

Institute for Quantum Electronics

ETH Zurich

May 27, 2014

## Abstract

This semester thesis describes the digital and analogue components of an active laser intensity stabilisation device named CIS box as well as passive means of reducing the fluctuations the laser suffers while going through a fibre. The CIS box is based on an Arduino UNO microcontroller and is designed to stabilise laser fluctuations on the time-scale of seconds up to hours. The approach taken was to implement a feedback loop, that converts an input signal from a photo detector into a correction signal via a proportional integral derivative (PID) scheme. The signal was fed back to an acousto-optic modulator (AOM) controlling the laser intensity. In order to ensure compatibility with a broad range of applications, a set of electronic features allowing to control the signal range was added. The intended application of the CIS box is a laser beam, that is transferred between two labs through a 20 m polarisation maintaining fibre and hence suffers severe intensity fluctuations on time-scales varying from seconds (bending of fibre) to minutes (temperature fluctuations). In the most recent set up, the device exhibits a cut-off frequency in the few Hz region, a gain-bandwidth product of around 100 Hz, a settling time of 10 s of milliseconds and a slew rate around 200 mV/ms.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Preparation</b>	<b>4</b>
2.1	Alignment of PM Fibre . . . . .	4
2.2	Experimental Set up . . . . .	5
<b>3</b>	<b>Digital Components</b>	<b>6</b>
3.1	PID Feedback control . . . . .	6
3.2	PID Implementation . . . . .	7
3.3	Tuning Experiments . . . . .	8
<b>4</b>	<b>Analog Components</b>	<b>10</b>
4.1	Front Panel . . . . .	10
4.2	Printed Circuit Board . . . . .	11
<b>5</b>	<b>Functionality</b>	<b>12</b>
5.1	Modes of Operation . . . . .	12
5.2	Initialisation and Calibration . . . . .	13
5.3	Device Characterisation . . . . .	14
<b>6</b>	<b>Conclusion</b>	<b>16</b>
<b>A</b>	<b>The PID Algorithm</b>	<b>17</b>
<b>B</b>	<b>Analog Components</b>	<b>22</b>
<b>C</b>	<b>Electronics</b>	<b>24</b>
<b>D</b>	<b>Test Analysis</b>	<b>27</b>
<b>E</b>	<b>Additional Programmes for Setting Up</b>	<b>29</b>

# 1 Introduction

In order to achieve coherent manipulation of the internal states of ions in a trap, it is vital to be in control of the rabi-frequency which in turn depends on the intensity of the laser-beam. Hence, the intensity of the laser needs to be monitored and kept at a predetermined level over a long period of time. Since the laser is guided to the trap via optical fibres, the beam intensity fluctuates due to changes in the external conditions such as temperature or stress.

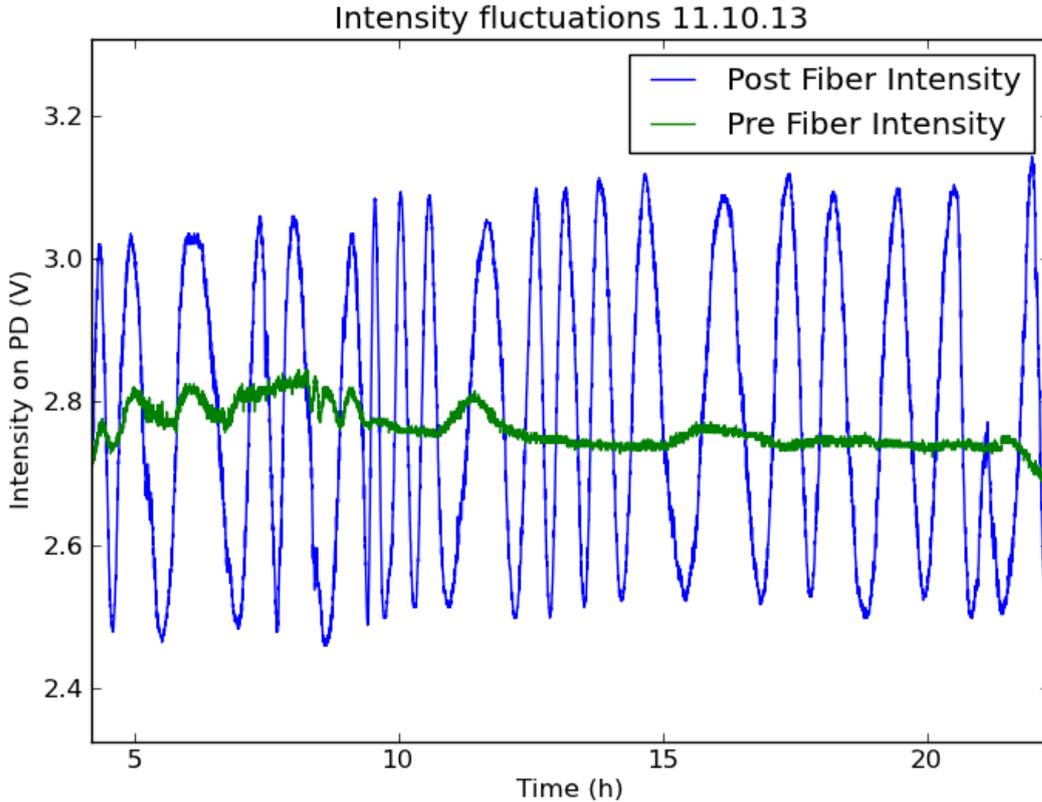


Figure 1: Fluctuations of the intensity of a 854 nm laser after a 20 m polarisation maintaining (pm) fibre (OZ Optics PM 850, Core/cladding ratio: 4/125) after the polarisation direction was aligned with the principal axis. For comparison, the pre-fibre intensity pattern is displayed as well, though not to scale due to a different photo detector (PD) gain. The traces were recorded with a photo-diode (Thorlabs PDA 36A). One can clearly see the periodicity of the fluctuations on the order of 30 minutes.

Minimising the disturbance of a laser-beam travelling through a fibre aims at two distinct criteria: First, the overall attenuation should be as small as possible. Second, and vital for the application at hand, the fluctuations of the intensity and polarisation must be minimised. Due to the small wavelengths involved in visible light, the polarisation tends to suffer from noise (high frequency) while the pure intensity fluctuates (low frequency) when going through a fibre. To reduce the overall attenuation, one shall optimise the coupling efficiency of the beam into the fibre. The approach to minimising noise and fluctuations consisted of both passive and active measures. On the passive side, a polarisation maintaining fibre and a post selection polariser can reduce the high frequency noise due to the changes in polarisation. However, this comes at the expense of large, low frequency fluctuations in the beam intensity. Here we need the device for active stabilisation of the beam intensity. In fact, this is only one example for the

application of the CIS box. Endowed with an floating set-point operation mode, the intensity can not only be stabilised but also easily and reliably manipulated, such that the intensity itself becomes a manipulable parameter in experiments. Also, scanning the laser beam across a range of frequencies typically induces an intensity ramp as well, which can be removed easily with the CIS box. The active intensity stabilisation is realised by an acousto-optic modulator (AOM) in feedback operation. The CIS box controlling the AOM is based on an Arduino UNO microcontroller and operates a PID feedback loop.

In standard single mode optical fibres, the polarisation has no preferred direction because the refractive index is uniform. Due to the short wavelengths involved, tiny changes in refractive index due to bending, temperature gradients or material defects cause significant crosstalk of the polarisation modes. A polarisation maintaining (pm) fibre exhibits strong birefringence itself, that is the refractive index varies strongly with respect to the polarisation direction. Thus, potential polarisation crosstalk during the transmission is suppressed by interference. At points separated by half a beat-length, the coupling into the second polarisation mode is out of phase. Since the beat-length is typically much smaller than regions of induced birefringence (i.e. through bending), the net intensity in the second polarisation mode suffers from destructive interference and averages to zero. The idea is, that aligning the polarisation of the beam with one of the principal axis of the fibre theoretically decouples the output polarisation from temperature or stress induced changes in refractive index or fibre length. We are then left with a variable attenuation due to the intensity lost into the second polarisation mode. Having optimised the spatial and polarisation in-coupling, the fluctuations of the intensity of the polarised output should be strongly reduced. Using a long fibre, these disturbances are inevitable and hence need to be corrected for actively. Figure 1 shows the level and time-scale of fluctuations of the 854 nm diode laser used throughout this thesis after the alignment process.

## 2 Preparation

### 2.1 Alignment of PM Fibre

An initial measurement of the fluctuations after the 20 m fibre transferring the 854 nm laser from one lab to the lab where the trap was based, indicated the necessity of optimising the alignment of the polarisation axis of the beam and the principal axis of the fibre. For preliminary alignment, there is a mark on the fibre, indicating the principal axis. To improve the alignment beyond the accuracy of reading the mark, the procedure taken from reference [1] was applied. The theoretical basis of this alignment procedure is the frequency dependence of the output polarisation in birefringent materials. The polarisation state after a certain length of birefringent material depends on the relative phase shift of the two polarisation modes:

$$\Delta\phi \approx \frac{2\pi\Delta n l}{c} f \quad (1)$$

Here,  $\Delta n$  denotes the difference in the refractive indices and  $l$  denotes the length of the fibre. The frequency of the laser is given by  $f$  and  $c$  is the speed of light in vacuum. Clearly, the fibre acts as a high order wave-plate, where the order of the wave-plate depends on the frequency. The alignment procedure was carried out using the set up presented in figure 2. If one modulates the frequency of the laser and monitors the intensity in both polarisation directions after the fibre, one can see that the intensity in the two polarisation modes is modulated out of phase. Ideally, one would now use a  $\lambda/2$ -wave-plate in front of the fibre to align the polarisation such that the modulation in either mode vanishes. Note that if the light enters the fibre polarised

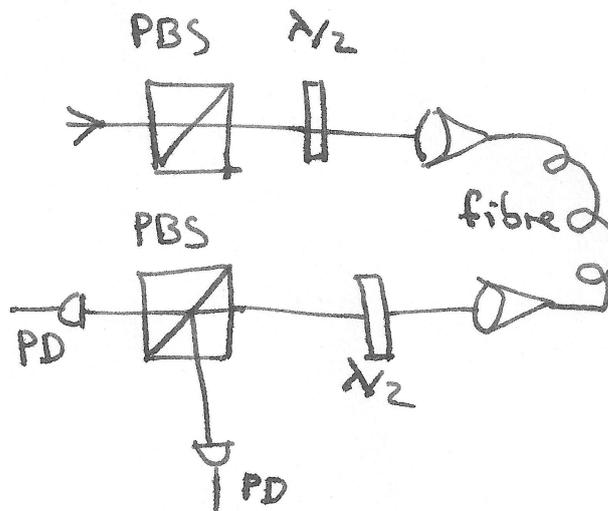


Figure 2: The set up of the polarisation maintaining fibre alignment process. The first  $\lambda/2$ -wave-plate was used to align the polarisation direction. The two photo-diodes were connected to an oscilloscope. The second  $\lambda/2$ -wave-plate was used to distribute similar intensities to both photo-diodes to make the fluctuations comparable. The result of the alignment process is displayed in figure 3.

along only one of the principal axis, there is no effective difference in refractive index. However, modulating the frequency of the laser also induced a modulation in the intensity. So instead of tuning the  $\lambda/2$ -wave-plate such that the fluctuations in both polarisation modes vanish, it is sufficient to tune the wave-plate such that the intensity modulations in both modes are in phase. A second  $\lambda/2$ -wave-plate after the fibre was used to achieve similar intensities in both polarisation modes, which simplified the tuning process. The result of the tuning process is shown in figure 3.

## 2.2 Experimental Set up

After the alignment process, the set up for active stabilisation was prepared as in figure 4. The idea is to pick out a fraction of the beam intensity using a beam-splitter and guide it onto a photo-diode. This photo-detector (PD) signal is then sent to an Arduino UNO microprocessor, which compares the instantaneous intensity with a reference intensity and calculates a correction signal which is fed back to an AOM modulating the beam intensity such as to maintain a fixed level. In order to test the device, the beam intensity was modulated using a second AOM before the stabilisation set up. The AOMs were driven at 80 and 167 MHz respectively by a direct digital synthesiser (dds) board. The feedback signal was generated by a digital to analogue converter (Adafruit MPC4725 DAC) that can be directly integrated in the microcontroller structure (see reference [4]). The modulation signal for the additional AOM was produced by a function generator. In order to measure the success of the stabilisation of the output, some light was picked out in front of the stabilisation AOM and compared with the signal used for the generation of the feedback. The photo-diode signals were processed using an analogue to digital converter (ADC) and plotting the intensity pattern on a computer. This allowed the analysis of the response times to intensity steps and large amplitude fluctuations of the incoming beam.

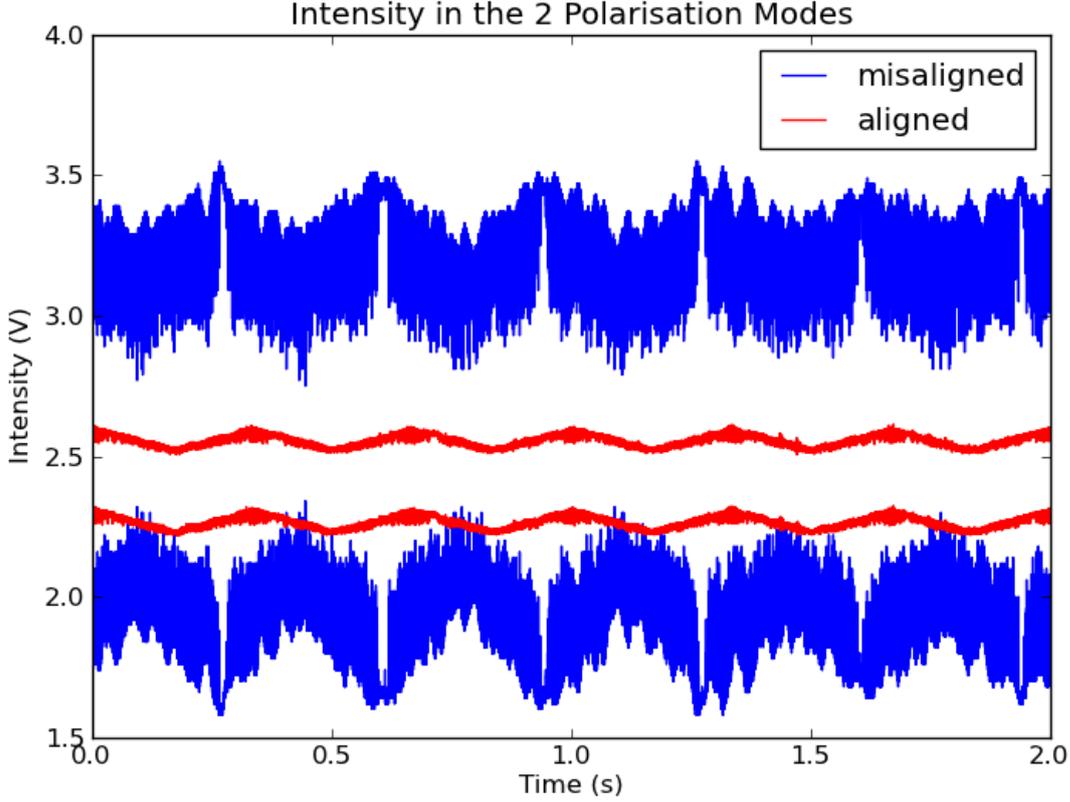


Figure 3: Fluctuation of the polarisation in the aligned and the non-aligned case. One can clearly see, that in the non-aligned case the fluctuations in the two polarisation modes are out of phase and much larger than in the aligned case. Moreover, in the aligned case, the fluctuations are in phase. In each case, both polarisation modes were monitored using the set up in figure 2.

## 3 Digital Components

### 3.1 PID Feedback control

The idea of a PID feedback loop is to compare an input signal with a set-point, and use the deviation to calculate a correction signal. This correction signal influences the physical process such that the resulting input is pushed towards the set-point. In the case of intensity stabilisation, the input is generated by a fraction of the beam being picked out and deviated to the photo-diode. The PD voltage  $V_{in}$  in the range [0-5 V] is then read by the analogue digital converter on the Arduino UNO microcontroller and mapped to the 10 bit range [0, 1023]. The absolute error in the  $i^{th}$  step,  $e_i = V_0 - V_{in,i}$ <sup>1</sup>, is then used to calculate the  $i^{th}$  correction signal  $\Delta V_i$  via the discretised PID scheme:

$$\Delta V_i = K_p e_i + K_i \delta t \sum_j^i e_j + K_d \frac{e_{i-1} - e_i}{\delta t} \quad (2)$$

In this equation,  $K_p$  is called proportional gain,  $K_i$  is the integral gain and  $K_d$  is the derivative gain.  $\delta t$  is the time constant, which depends on the sampling rate at which new correction signals are generated. While the proportional gain corrects for the bulk of the error, the integral

<sup>1</sup>i.e. the difference of the  $i^{th}$  input and the set-point  $V_0$ , also set to some number in [0,1023]

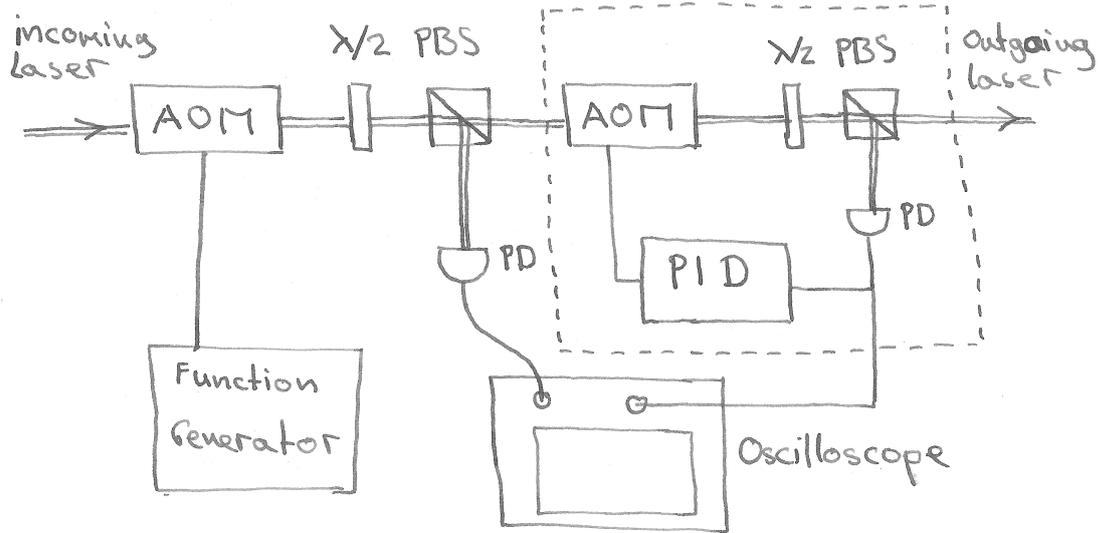


Figure 4: The figure shows the experimental set up used for determining the tuning parameters of the PID algorithm. In the dashed box, one can see the actual PID stabilisation device with the pickoff, the photo diode, the PID processing unit (CIS box) and the AOM closing the feedback loop. In the experiment an additional AOM, driven by a function generator, was installed in front of the set up. This way, different signals and their effect on the PID feedback loop were analysed and the PID gain parameters optimised. To monitor the signals, both the intensity before and after the stabilisation device were recorded on a computer. The  $\lambda/2$  plates in combination with the polarising beam splitters (PBS) were used as variable pickoffs.

gain is necessary to avoid steady state errors. Note that for the weight of the integral term, the time constant of the integration is also relevant. The derivative term is typically added for fast responding PID systems and is not necessary for this task. Due to the architecture of the system, this correction term  $\Delta V_i$  is then added to the previous output voltage  $V$ . Accordingly, if the input is smaller than the set-point, the voltage fed to the AOM is increased and hence its transmission is enhanced. Thereby, the beam intensity increases, leading to a higher input from the photo-diode and a smaller error. In order to find the optimal gain parameters, the device needs to be tuned. This was done using the experimental set up with two AOMs in figure 4.

### 3.2 PID Implementation

This paragraph will only address the core calculation. A description of all features of the algorithm is given in the appendix A, The PID Algorithm. The different modes of operation and how they work will be dealt with in the section Functionality.

After reading the signal from the photo-diode,  $V_{in}$  into the variable Input, the Compute function of the PID library is employed. The variables Input, Output and Setpoint are directly called and manipulated by the Compute function using references. If a new output voltage  $\Delta V$  is calculated, it needs to be scaled appropriately and added to the prior voltage  $V$ . Rescaling of the correction signal  $\Delta V$  is necessary since the change in voltage directly affects the AOM transmission,  $T$ . However, the correction signal shall be proportional to the change in intensity, not the change in AOM transmission. Otherwise, equally large correction signals  $\Delta V$  lead to equally large changes in transmission of the AOM instead of equally large changes in the beam intensity  $\Delta I$ . To achieve the latter, one needs to divide the correction signal by

the total beam intensity:

$$\Delta I \propto I_{tot} \Delta T \propto I_{tot} \Delta V. \quad (3)$$

The second proportionality relies on the approximately linear relationship between voltage and transmission of an AOM in the relevant central region (see figure 12). Rescaling of  $\Delta V$  by dividing by  $I_{tot}$  yields the desired proportionality.

However, the only measure of the total beam intensity at hand is a combination of the Input  $V_{in}$  and the Voltage  $V$ :

$$I_{tot} = \frac{I_{trans}}{T} = c \frac{V_{in}}{V}. \quad (4)$$

Here  $I_{tot}$  is the total beam intensity,  $I_{trans}$  is the transmitted intensity and  $T$  is the transmission of the AOM. Again, the transmission  $T$  is assumed to be roughly proportional to the voltage  $V$ . The PD signal  $V_{in}$  is proportional to the transmitted intensity. Hence, the second equality holds approximately for a proportionality factor  $c$ .

Under the assumption of approximate linearity of the AOM transmission function, the correct rescaling of the correction signal is given by:

$$\Delta \tilde{V} = \tilde{c} \frac{V}{V_{in}} \Delta V \quad (5)$$

This is then added to the prior voltage  $V_{new} = V + \Delta \tilde{V}$ . The variable `factor` plays the role of the constant of proportionality  $\tilde{c}$ , fitting together the 10 and 12 bit scales of the ADC and the DAC. With the parameter `Voltage` ranging in  $[0,4095]$ , the parameter `Input` in  $[10,1023]$  and the parameter `Output` being of the order of the `Input`, the constant of proportionality `factor` was chosen to be  $10/4095$ . This number only deals with the proportionality and is arbitrary, though related to the tuning factors `Kp` and `Ki`. Those were set by tuning experiments.

### 3.3 Tuning Experiments

In order to determine the optimal tuning parameters, the set up was extended by a second AOM as shown in figure 4. Two characteristic disturbances were applied to the beam: A step signal and a sine wave. First, the recovery time after a step signal was optimised by variation of the proportional gain. The experimental results are displayed in figure 5 and 6, and the optimal recovery time was found for a proportional gain of  $K_p = 50$ . Afterwards the response to a sinusoidal signal was tested for a range of frequencies. The performance for different gain parameters exhibited a trade-off between the successful correction of the two perturbation signals: Whereas the sine fluctuation was corrected more effectively by a larger  $K_p$ , these values lead to more oscillations in the correction of the step signal. Hence, an integral gain parameter  $K_i$  had to be introduced to improve the response to the sine fluctuation. Here, the optimal value that still showed stability over a wide range of input levels was found to be  $K_i = 5$  (see figure 7 and ??), allowing for an effective compensation of large amplitude fluctuations up to 7 Hz (see figure 8).

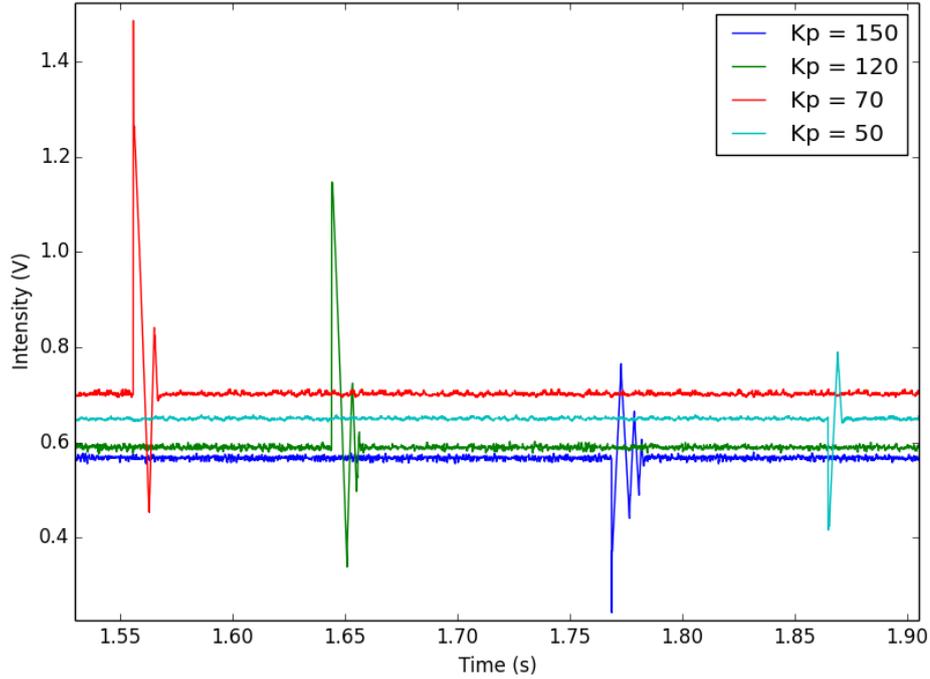


Figure 5: Recovery of the stabilised beam intensity after a 1 V step modulation of the incoming laser-beam for several different proportional gain parameters  $K_p$ . One can clearly see, that the recovery time for  $K_p = 50$  is much lower than for all higher proportional gain parameters. Absolute times and intensities are arbitrary. Figure 6 shows a close up of the recovery of the  $K_p = 50$  case, indicating a recovery time on the order of milliseconds. The measured recovery times for all proportional gain parameters were found to be around  $\tau_{50} = 7 \text{ ms}$ ,  $\tau_{70} = 10 \text{ ms}$ ,  $\tau_{120} = 12 \text{ ms}$ ,  $\tau_{150} = 27 \text{ ms}$ . Note that higher  $K_p$  parameters lead to oscillations before returning to the set-point. The

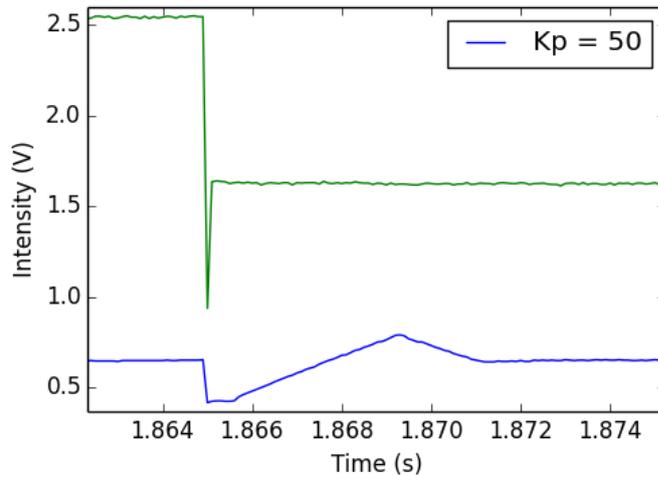


Figure 6: This figure shows the original laser beam with the step modulation and the recovery of the stabilised beam intensity with a proportional gain parameter  $K_p = 50$ . Note also the characteristic slope of the signal oscillations, seemingly corresponding the fastest change in intensity the algorithm can produce (slew rate). Interestingly, this slope is independent of the gain parameter.

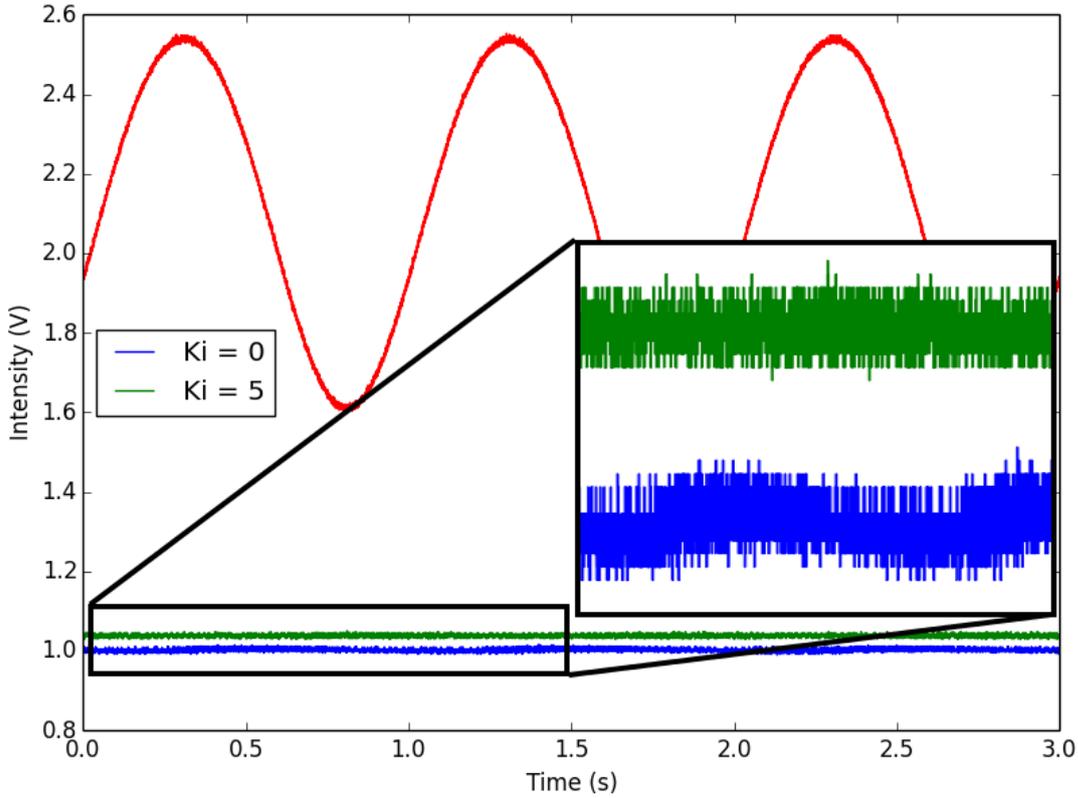


Figure 7: Stabilisation of sinusoidal beam intensity modulations. Whereas the step recovery was improved by lowering the proportional gain parameter, the trade off was observed in the sine response. Here a proportional gain of 50 on its own suffered from noticeable disturbance. As it can be seen in the close up of the stabilised intensities, introducing an integral gain parameter  $K_i$  can compensate for that. Testing several  $K_i$  parameters over a range of inputs, it was found that  $K_i = 5$  was both able to compensate a sensible ramp signal and remained stable at larger frequencies.

## 4 Analog Components

To ease the reproduction of the CIS box and ensure compatibility with a range of systems (PDs and AOMs), several electronic features were embedded into a printed circuit board (PCB). Moreover, the algorithm was endowed with four different modes of operation, to be controlled via switches on a front panel, which also includes an LCD displaying the most important system parameters.

### 4.1 Front Panel

The front panel was designed using the CAD program AutoDesk. On the passive side, the front panel features the BNC sockets for the input from the PD and the output to the AOM driver. The active features are two switches ("lock" on the left and "manual" on the right) to choose the operation mode. Moreover, there is a knob to tune the laser intensity in the variable power modes, a LCD displaying the key parameters in each mode of operation and two LEDs

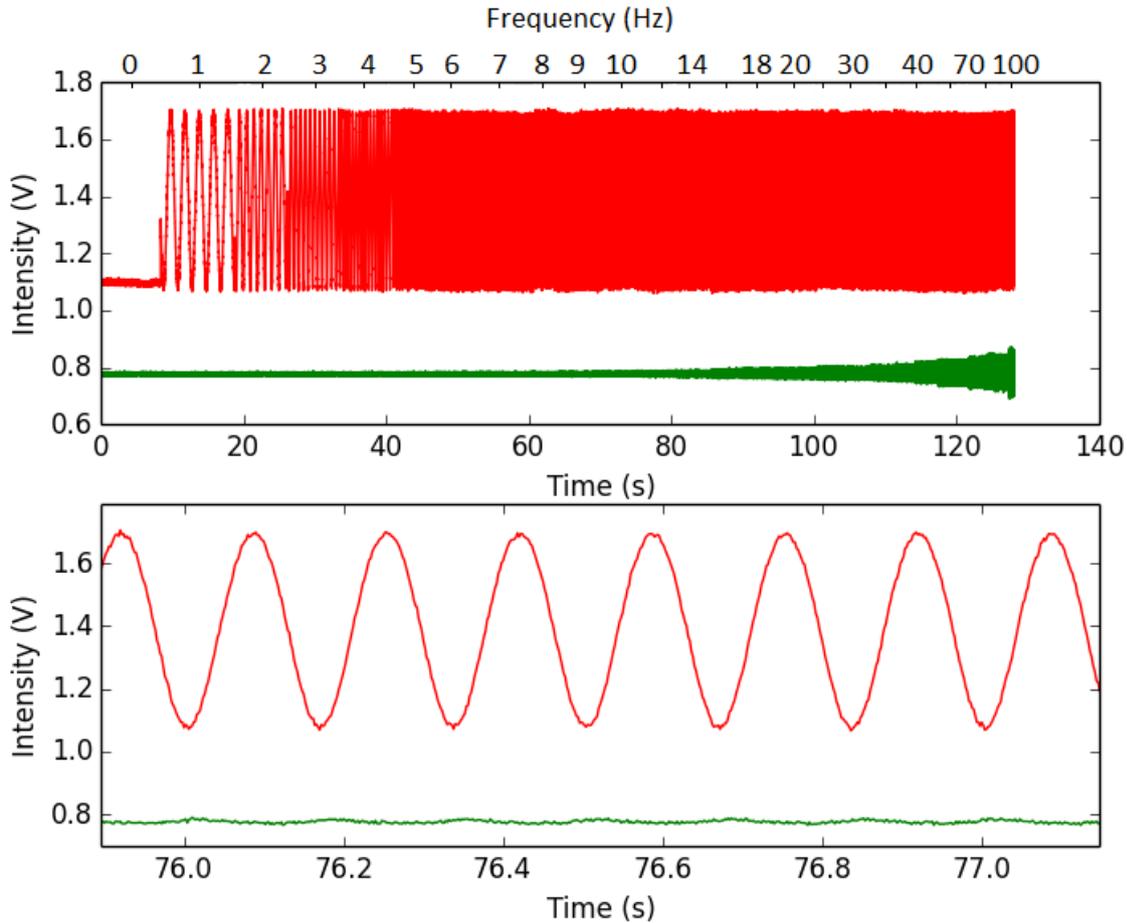


Figure 8: In order to analyse the bandwidth and stability of the PID, sine fluctuations of frequencies up to 100 Hz were applied. The red trace shows this modulated laser intensity, whereas the green trace is the stabilised intensity. In the first plot one can see that the stabilisation is successful up to around 70 seconds, where a frequency of 7 Hz was applied. Note though, that the fluctuations are very large and not to be expected at such frequencies. Still, one can see that up to 100 Hz there is no singular behaviour and a reduction of the fluctuations is still achieved. The lower plot shows a close up of the two signals at 7 Hz input modulation. Although one can clearly see the fluctuations in the stabilised signal, they are greatly reduced.

indicating power supply (green) and failing stabilisation (red). The design of the front panel can be found in figure 15 in the appendix.

## 4.2 Printed Circuit Board

To simplify the reproduction of the device, a printable circuit board including all the electronics was designed and is displayed in figure 13 in the appendix. The board requires power supply pins at 15 V and -15 V, as well as a ground connection. The Arduino UNO board can be directly attached to the board and receive its 12 V supply. Most importantly, non-inverting operational amplifiers were placed on the input and the output. On the input side, this allows the device to be used universally, with photo-diodes of different signal parameters. The circuit allows for the compensation of an offset of the PD voltage from -15 V to 15 V and can then map the signal to the Arduino UNO range [0,5 V]. In order to protect the ADC input of the Arduino UNO from

negative voltage inputs, a diode is placed before the `Analog In` pin. On the output side, the range can be varied from  $[0, 2.5 \text{ V}]$  up to  $[0, 15 \text{ V}]$ , and hence adapted to a range of differently responding AOM drivers. Since the output frequency of the Arduino UNO is given by the loop duration, all higher frequencies are a result of electronic noise and hence shall be suppressed by two low pass filters before and after the non-inverting operational amplifier. Furthermore, the board features sockets for the switches, the turning knob and the LCD. The wiring is described in figure 14 in the appendix B Analogue Components.

## 5 Functionality

### 5.1 Modes of Operation

During the design and testing phase, it proved very useful to manually control the set-point intensity. This way, one can not only keep the beam power stable for the duration of a single experiment, but one can also reproduce a certain intensity in successive experiments. Furthermore, one can use the intensity as a variable parameter in experiments.

The algorithmic implementation of this floating set-point or manual lock operation mode leads to a severe decrease in the ability to stabilise the intensity, because the `analogRead` operation, necessary for the adaptation of the set-point in each loop, increases the loop duration significantly. In order to retain a constant loop duration and hence integration constant, the sample time of the PID library must be set to  $8000 \mu\text{s}$  in the manual lock operation. However, during experiments this is not acceptable. The idea is to switch to the manual lock mode to change the set-point to the desired value and then resume the efficient automatic lock mode with the new set-point.

Another mode of operation that proved helpful, was simply the direct manipulation of the laser intensity by turning the potentiometer knob. The fourth mode of operation is then simply the inactive mode, that switches the transmission parameter of the AOM to the maximum.

**Inactive Mode:** To choose the inactive mode that simply maximises the transmission of the AOM, both the lock and manual switches need to be off, i.e. the respective pins pulled low. The display now shows `Inactive` and the current beam power. The warning LED (red) is pulled high, indicating that the laser-beam is not stabilised.

**Manual Mode:** To manually vary the intensity, the manual pin needs to be pulled high. Then, turning the knob will directly vary the output voltage and hence the AOM transmission. The LCD now displays the relative signal strength (i.e. the ratio  $\frac{\text{currentVoltage}}{\text{maximalVoltage}}$ ) which approximates the AOM transmission. Furthermore, the current intensity is displayed as a number ranging from 0 to 1023<sup>2</sup> and the set-point is continuously adapted to the current input level. Note that set-points corresponding to an ADC input (10 bit) below 20 can lead to instability and hence the set-point cannot be reduced beyond. In manual mode, the display is updated every loop.

---

<sup>2</sup>This is the standard setting. If you wish to calibrate the display so that it displays the actual beam intensity, you need to change a parameter in the code as described in the paragraph Adapting the Code

**Manual Lock Mode:** In order to start the stabilisation, the lock pin needs to be pulled high. If one starts locking in manual mode, the initial set-point is the instantaneous intensity when locking is initiated, thus allowing for a smooth transition into the lock mode. Turning the knob now does not directly change the AOM transition but the set-point and hence the intensity is varied indirectly. Note that the set-point cannot be increased further if the transmission is already at the maximum. In this manual lock mode the display shows the relative signal strength and the current set-point and is still updated constantly.

**Automatic Lock Mode:** To run in the automatic lock mode, the lock pin needs to be high while the manual pin is low. Then the set-point set before is stabilised and turning the knob has no effect. Starting the lock mode directly from the inactive mode sets the set-point to 70% of the beam intensity at the time of locking. To optimise the performance, the display is not updated any more. It shows the signal strength at the time of setting the set-point and the set-point itself.

## 5.2 Initialisation and Calibration

In order to use the device, it is necessary to adjust the input and output electronics to the specific set up. On the input side, different photo-diodes may yield voltages in different ranges and with different offsets. In the lab, the offsets were on the order of a few volts (also negative). On the output side, different techniques to control the AOM need voltages in different ranges. The variable voltage radio frequency attenuators used in the lab needed voltages up to 6 V, but some had upper endpoints below 5 V.

**Input Electronics:** In order to be able to remove voltage offsets and map the photo-diode signal reliably to [0,5 V], the set up in figure 16 was designed. For an analytic description of the circuit, see appendix C. In order to adapt the set up to the desired input voltage range, one needs to first fix the lower bound. This is done by applying the lowest voltage expectable in the set up (i.e. the photo-diode voltage with the beam blocked) and tuning the offset potentiometer RO (see schematic 16 and the board in figure 13) such that the voltage after the amplification stage of the input (but before the diode) vanishes (measure at point M on the board in figure 13). Subsequently, the upper bound shall be fixed. Set the input voltage to the maximum value that will be obtained in the experiment and then tune the gain potentiometer RA (see schematic 16 and the board figure 13) such that the voltage at the point M hits 5 V (or slightly less to avoid saturation of the Arduino ADC). Now the selected input range is mapped linearly to the range of the Arduino, thus optimising its resolution.

**Output Electronics:** The output electronics are designed to map the Arduino output range [0,5 V] to any non-offset range with upper endpoint in [2.5 V, 10 V] (in the first board design). The circuit consists of a voltage divider that halves the voltage and a non-inverting operational amplifier whose gain can be tuned to any value in [1,5]. For the set up, simply let the Arduino output a constant 5 V voltage (using `dac.setVoltage` as in Appendix E, Set Up). Then tune the potentiometer RB (see the schematic in figure 18 or the board in figure 13) such that at the output socket the voltage corresponds to the upper bound of the voltage taken by the variable voltage attenuator (i.e. the radio frequency intensity controller).

**Adapting the Code (optional):** If it is of great interest to know the actual beam intensity in the stabilised beam, the parameter `input_to_intensity` in the code has to be changed. By default it is set to 1, such that the intensities given on the display simply range from 0 to 1023 in some arbitrary units determined by the photo-diode gain and the pickoff ratio. In order to find the right conversion parameter, one needs to calibrate the photo-diode by finding the relation of output signal and intensity (i.e. the output signal  $V_{PD}$  at some known beam intensity or power  $I_{beam}$ ). If the pickoff ratio is given by  $R$ , the `input_to_intensity` parameter is given by:

$$\text{input\_to\_intensity} = \frac{1 - R}{R} \left( \frac{I_{beam}}{V_{PD}} \right) \frac{5}{1023}. \quad (6)$$

The product `input * input_to_intensity` is then given in units of  $I_{beam}$  and corresponds to the beam power after the stabilisation device.

### 5.3 Device Characterisation

This section describes the experiments carried out with the final version of the PID algorithm on the Arduino, which was attached to the PCB with the electronics described. After initialising the input and output settings, the set up in figure 4 was used for the analysis of the device parameters. In order to find out some meaningful characteristics of the device, it was subjected to two different types of disturbances: sinusoidal and step.

Applying a large amplitude sinusoidal disturbance (57% of the average signal strength) at increasing frequency, the (inverse) gain-bandwidth product was determined<sup>3</sup>. Up to this frequency, the large scale fluctuations are still attenuated. However, effective cancellation of the fluctuations is only successful for much lower frequencies, the upper bound being referred to as cut-off frequency. The experimental data is displayed in figure 9 and the results for different sets of tuning parameters are displayed in table 1.

Considering the settling time and settling behaviour of the signal after the input beam intensity was stepped up, revealed interesting differences in the response of different tuning parameters. As displayed in figure 10, the low gain set up returns to the initial value after overshooting once while the larger gain set up begins a strongly damped oscillation before returning to the set-point in about the same time. Increasing the gain even further leads to singular behaviour and heavy oscillations without reaching the set-point on relevant time scales. Interestingly enough, the oscillatory behaviour after the step reveals what seems to be the maximal correction slope, the slew rate (see also figure 19 in the appendix). Owing to its apparent independence from the gain parameters, improvement of this device parameter will need to address the sampling rate of the algorithm.

On top of the induced disturbances, it was observed that the device itself caused 50 Hz oscillations of amplitude around 1% of the signal strength. Owing to their independence from the incoming beam intensity pattern, they are most likely caused by some power supply nearby, inducing the oscillations into the output electronics. In the most recent, though not yet tested PCB version, capacitors were added in the op-amp supply lines to counteract this phenomenon. If the problem continues, measures have to be taken to shield the PCB from its surroundings.

---

<sup>3</sup>In this case, this refers to the frequency at which the gain of incoming fluctuations rises above unity.

$K_p, K_i, K_d$	gain×bandwidth	cut-off	settling time	slew rate
50, 5, 0	100 Hz	3 Hz	0.02(67) s	0.2 V/ms
100, 10, 0	50 - 100 Hz	10 Hz	0.02(91) s	0.2 V/ms
50, 10, 5	20 - 50 Hz	5 Hz	-	-
300, 50, 30	-	-	inf	0.2 V/ms

Table 1: Device characteristics for different tuning parameters. The settling time is taken as the average over all step responses in figure 21. The estimates of the gain-bandwidth product are based on the analysis of figure 9, while the cut-off frequency and the slew rate are estimates based on the figures 19 and 20 in the appendix D, Test Analysis.

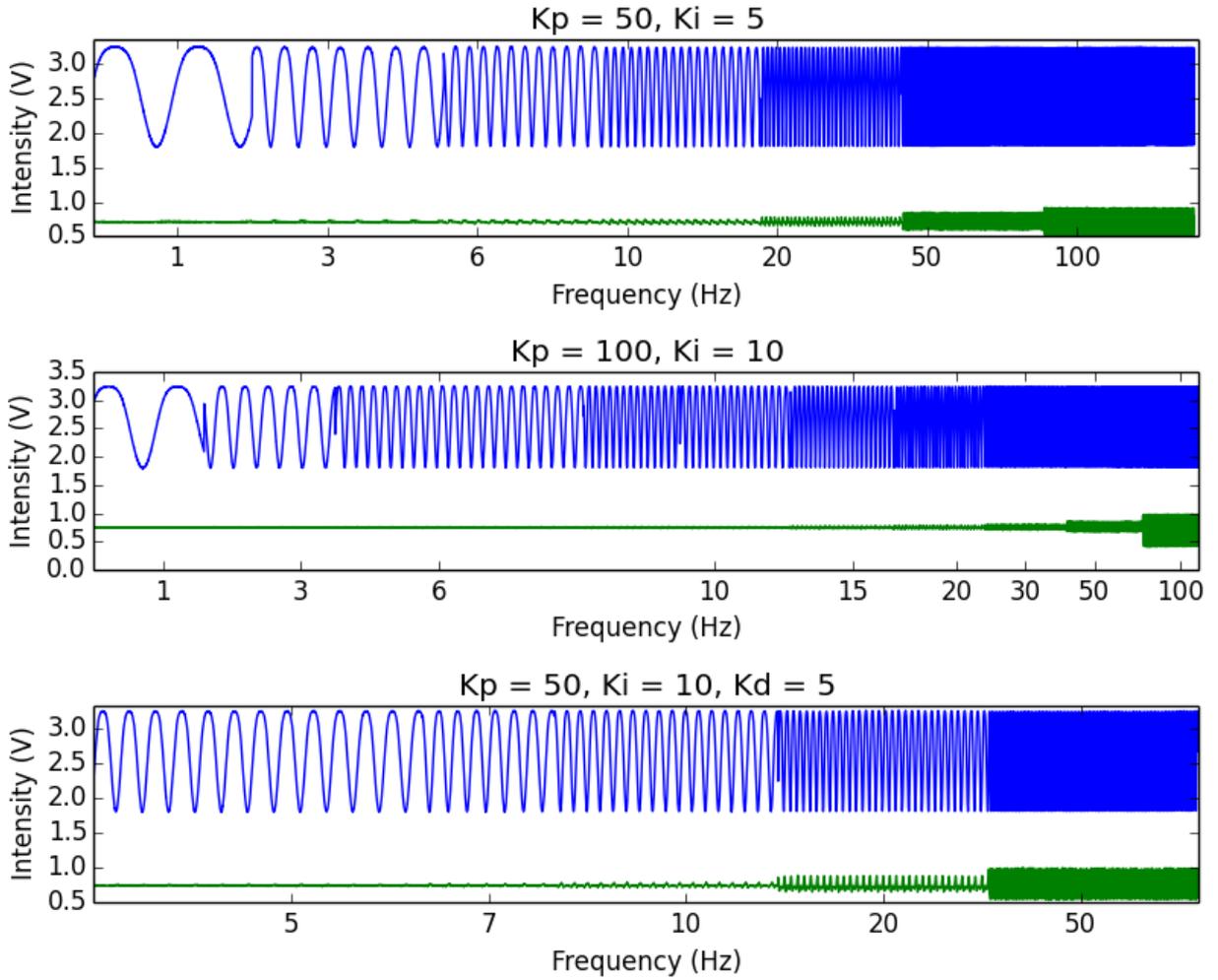


Figure 9: The three plots show the initial laser-beam disturbance (blue) and the stabilised signal (green) for several different parameters  $K_p, K_i, K_d$ . At 100 Hz, the gain (or rather attenuation) in fluctuation in the first configuration ( $K_p = 50, K_i = 5$ ) is just below one:  $\frac{\Delta I_{out}}{\Delta I_{in}} = \frac{0.564}{0.577}$ , with  $\Delta I_{out}, \Delta I_{in}$  given by the relative fluctuation of the output and input respectively. This indicates a gain-bandwidth product of approximately 100 Hz. For both other configurations, this ratio exceeds one, indicating that the gain-bandwidth product is lower than 100 Hz. Note though, that up to 50 Hz, the second configuration is more effective.

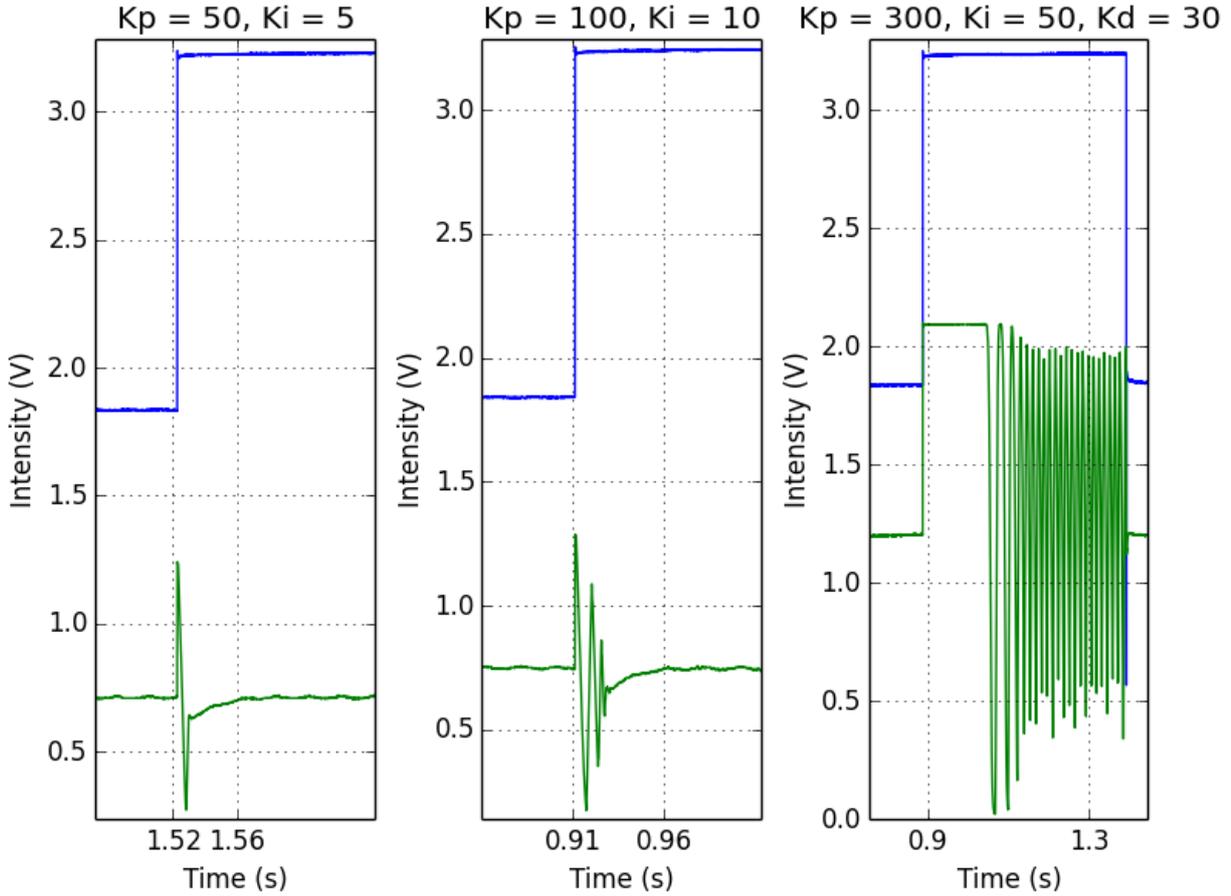


Figure 10: The three plots show the initial laser-beam disturbance (blue) and the stabilised signal (green) for several different parameters  $K_p, K_i, K_d$ . One can see that for larger gain parameters, the settling time suffers from instability since the oscillatory behaviour becomes stronger. Although the settling time is of the same order of magnitude for the first two set ups, the leftmost is clearly the most favourable.

## 6 Conclusion

To summarise, the CIS box designed and built during this semester project was found to be capable of cancelling the intensity fluctuations of a laser beam that is guided through a pm fibre. Those fluctuations were found to be on the time-scale of a few minutes and hence perfectly within the bandwidth of the CIS box. In fact, the architecture of the Arduino UNO allows the device to work effectively up to the several Hz regime and attenuation of fluctuations is even successful up to 10s of Hz. Sudden disturbances and changes in intensity do not lead to instabilities, but are corrected for on a time-scale of a few milliseconds. All in all, the CIS box can be easily adapted to a range of systems and is therefore universally applicable in all experiments that benefit from a stable laser intensity.

At this point, I would also like to thank all members of the lab for contributing all those valuable hints, ideas and objections that made this project work most of the time.

# A

## The PID Algorithm

In the following there will be a detailed discussion of the PID algorithm. The core part of the algorithm deals with the calculation of the correction to the output voltage in every step. It is based upon an open source implementation of a PID to be found on the arduino web page [2]. I will concentrate on explaining only the relevant features of the library. A detailed explanation of the library can be found on the arduino web page [3].

**Libraries Used:** Most importantly, the code is built around the arduino PID library [2], which was modified in order to reduce the update interval. Originally, the library would read in the sample time in milliseconds. Since this becomes inaccurate for low values, it is more suitable to specify the sample time in microseconds, which also allows us to reduce it below 1 ms.

Moreover, the wire-library and the Adafruit\_MPC4725-library were necessary to control the external DAC, which was used to generate the analogue control signal going to the AOM. An explanation of the wiring and programming of the DAC can be found in the Adafruit tutorial [4]. In order to communicate with the 16 column, 2 row liquid crystal display (LCD), the libraries SPI and LiquidCrystal are necessary. The wiring and programming of the LCD was done following the instructions in the tutorial on the digital salvage website [5].

To increase the speed of the algorithm, the standard digitalRead and digitalWrite commands were replaced by the faster digitalWriteFast library to be found in the arduino forum [6].

**Set Up Section:** In order to achieve a high response rate, the ADC clock pre scaler is halved. This raises the clock frequency to 250 KHz and hence reduces the time taken for one analogue read cycle. Reducing the pre scaler even more would increase the speed further, but ultimately this results in inaccuracy of the reading.

Then the settings of the PID library are initialised. The output is renormalised at the end of the algorithm and hence the output limits are dynamically set accordingly. Then the sample time is set to 400  $\mu$ s. This means that the algorithm will not update the output voltage faster and hence the time taken between two calculations is approximately constant. This is important for the calculation of the integral error because the algorithm assumes equal time intervals. Therefore, the sample time can not be reduced arbitrarily, but kept well above the actual time for one computational loop. As illustrated in figure 11, the loop duration was found to be below 400  $\mu$ s, hence justifying the choice. Finally, the gain parameters are set to the values  $K_p, K_i, K_d$  specified in the beginning.

```
myPID.SetOutputLimits(-Input/factor , 4095 * Input/ Voltage / factor - Input,
myPID.SetSampleTime(400);
myPID.SetTunings(Kp, Ki, Kd);
```

The initialisation of the set-point proceeds by setting the transmission of the AOM to an intermediate value by setting the voltage parameter to 2300. This value is fed to the DAC, which maps the numerical range 0-4095 (12 bit) to 0-5 V. Then, the input of the photo-diode wired to the analogue input labelled by A0 of the Arduino UNO is read in and used as the set-point. The set-point as well as the approximate fraction of the beam used are displayed on the LCD:

```
Voltage = 2300;
```

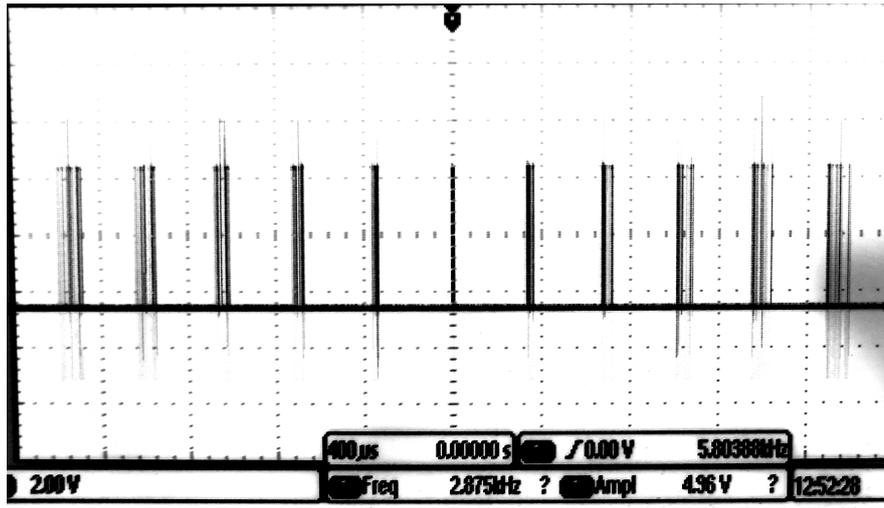


Figure 11: The loop duration of the PID algorithm. To obtain this plot, a digital pulse was generated in every loop while the sample time was reduced to zero. Hence, the pulse interval corresponds to the actual time taken by one computational loop. The spacing of one square on the oscilloscope denotes  $400\ \mu\text{s}$ , such that one can see that the calculation takes less than that. Therefore, setting the sample time to  $400\ \mu\text{s}$  should yield a constant time interval.

```

dac.setVoltage ( Voltage , false );
delay (1000);
Setpoint = analogRead (A0);

```

Finally, the PID library is activated.

**Calculation Section:** This paragraph will only address the core calculation in the automatic lock mode. The other modes of operation and how they work is discussed in the Functionality section.

Firstly, the current photo diode signal is read in to the Input variable:

```
Input = analogRead (A0);
```

If the Input variable is smaller than a minimum reference value specified by `min_signal` (default is  $20 \frac{5}{1023} V$ ), indicating that the beam is blocked or misaligned somewhere, the LCD outputs "No Input", the red warning LED is switched on and the loop is terminated:

```

if (Input < min_signal){
  PID_on = false;
  lcd.clear ();
  lcd.setCursor (0,0);
  lcd.print ("No Input");
  digitalWriteFast (led ,HIGH);
  led_on = true;
  dac.setVoltage (4095, false );
  signal_low = true;
  return;
}

```

Also, the control variable `PID_on` is set to false. When sufficient signal strength is regained, the control variable indicates that the LCD screen showing the parameters intensity and beam fraction needs to be reproduced and all control variables and warnings need to be reset:

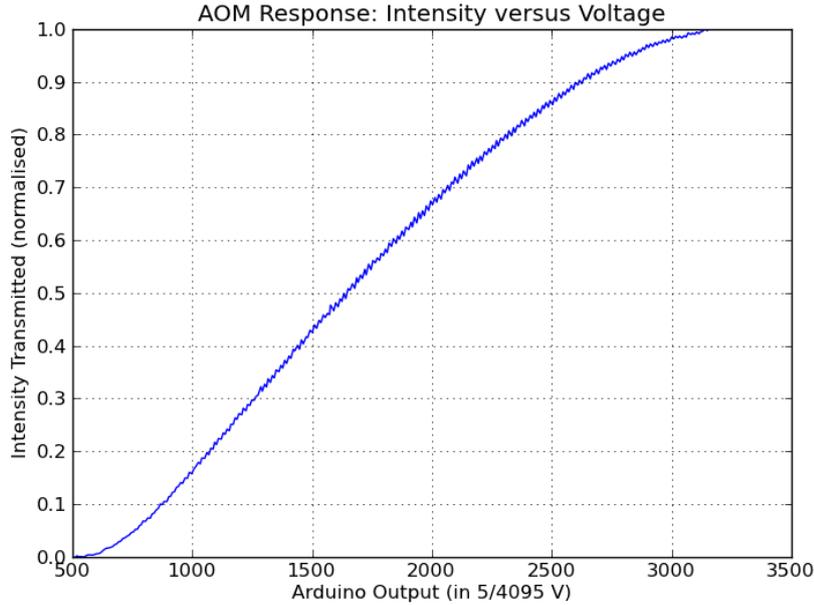


Figure 12: This is the AOM response to a ramp voltage signal generated by the Arduino. In the typical operation region around 50% intensity transmission the relationship is roughly linear. Pushing the beam power closer to the maximum, that is wasting less beam power, decreases the ability to respond reliable to fluctuations. Note that in the final version of the device the arduino output is always in the range 0-5V and then rescaled by an adaptable non-inverting operational amplifier. Hence, the main operation region is ideally around an arduino output of 2700 5/4095 V.

```

else if (PID_on !=true){
    lcd.clear();
    lcd.print(" Setpoint:");
    lcd.print(Setpoint);
    lcd.setCursor(0,1);
    lcd.print(" Signal=");
    lcd.print(Voltage/4095);
    digitalWriteFast(led,LOW);
    led_on = false;
    signal_low = false;
    PID_on = true;
}

```

After passing the above test, the Compute function of the PID library is conditionally employed, that is if the sample time of  $400\ \mu\text{s}$  has passed. The variables `Input`, `Output` and `Setpoint` are directly called and manipulated by the Compute function via references. The function outputs true only if the sample time has passed and hence a new  $\Delta V$ , denoted by the variable `Output`, has been computed. If this is the case, the new  $\Delta V$  is rescaled as described in the section PID Implementation and added to the prior voltage:

```
Voltage = Voltage + Output*(Voltage*factor/(Input));
```

The variable `factor` plays the role of the constant of proportionality  $\tilde{c}$ , fitting together the 10 and 12 bit scales of the ADC and the DAC. With the `Voltage` ranging in  $[0,4095]$ , the `Input` in  $[10,1023]$  and the `Output` being of the order of the `Input`, `factor` was chosen as  $10/4095$ . This number only deals with the proportionality and is arbitrary, though related to

the tuning factors  $K_p$  and  $K_i$ . Those were set by the tuning experiments described above.

**Parameter Checks:** In order to ensure a secure and reliable operation, there need to be several plausibility checks on the input signal. Firstly, one needs to make sure that the input signal is not too small. On the one hand this might correspond to situations when the beam is blocked and hence the attempt to stabilise might lead to instabilities as the beam is unblocked. On the other hand working with low signal strength makes the process more vulnerable to noise and it was found that below the intensity corresponding to an ADC converted value of  $20 \frac{5}{1023} V$ , the stabilisation process leads to instability. This check is implemented as follows:

```
if (Input < min_signal){
    PID_on = false;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("No Input");
    digitalWriteFast(led,HIGH);
    led_on = true;
    dac.setVoltage(4095,false);
    signal_low = true;
    return;
}
```

As the intensity returns to values above the threshold, the previous operation shall resume.

```
else if (PID_on !=true){
    lcd.clear();
    lcd.print("Setpoint:");
    lcd.print(Setpoint);
    lcd.setCursor(0,1);
    lcd.print("Signal:");
    lcd.print(Voltage/4095);
    digitalWriteFast(led,LOW);
    led_on = false;
    signal_low = false;
    PID_on = true;
}
```

Another test to be employed shall be whether the current laser intensity is sufficient to reach the set-point. To avoid the integral error building up during such phases, the PID shall be paused if the voltage is already at the maximum but the set-point is still not reached.

```
if (Voltage = 4095 && Input < Setpoint){
    digitalWriteFast(led,HIGH);
    led_on=true;
    PID_paused = true;
    return;}
//After the input signal reaches the Setpoint again, resume locking
else if (PID_paused){
    PID_paused = false;
    digitalWriteFast(led,LOW);
    led_on = false;}
```

Furthermore, after the new correction signal is calculated, it needs to be kept in the range corresponding to the 12 bit DAC output. Correction signals at the limit hint at stability issues and hence cause the red warning LED to switch.

```
if (Voltage >= 4095){
    Voltage = 4095;
    digitalWriteFast(led ,HIGH);
    led_on = true;
}
else if (led_on){
    digitalWriteFast(led ,LOW);
    led_on = false;
}
```

B

## Analog Components

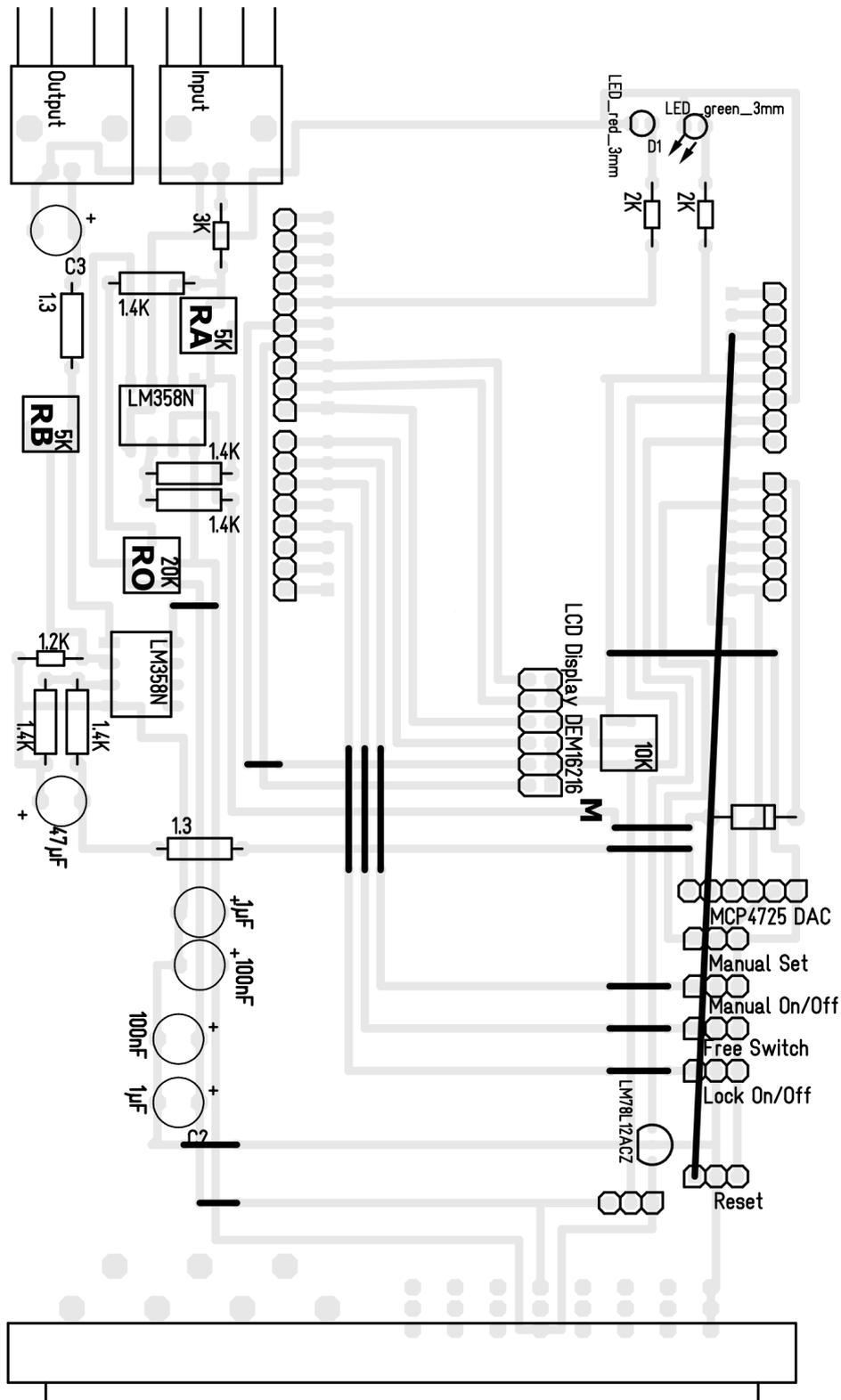


Figure 13: This figure shows the circuit board scheme designed with Target 3001!. The components and sockets for plugs are labelled. The list of components can be found in table 2.



Item	Number (vol.1)	Number (vol.2)
Arduino UNO	1	1
Adafruit MPC4725 DAC	1	1
Op-Amp LM358N	2	2
Voltage Regulator (LM 78L12ACZ (15 to 12 V))	1	1
Male Multipoint Socket Connector H7/F24	1	1
LCD Display DEM 16216 SYH-LY-CYR22	1	1
Toggle Switch	3	3
Resistor	1 (1.2 K)	1 (25 K)
	1 (3 K)	4 (100 K)
	2 (1.3)	2 (1.3)
	5 (1.4 K)	2 (50 K)
	2 (2 K)	2 (1 K)
Potentiometer	1 (10 K)	1 (10 K)
	1 (20 K)	1 (200 K)
	2 (5 K)	2 (100 K)
Capacitor	2 (47 $\mu$ F)	2 (47 $\mu$ F)
	2 (1 $\mu$ F)	2 (1 $\mu$ F)
	2 (100 nF)	2 (100 nF)

Table 2: List of components for the initial design (vol1) and the second design (vol2) of the device.

## C

### Electronics

In figure 16 one can see the set up used to adapt the input range to the Arduino. Roughly speaking, it consists of an additional source of voltage in parallel to the input voltage to remove positive or negative offsets up to  $\pm 15V$  and a non-inverting operational amplifier (split into an inverting gain stage and another voltage inverter). Via nodal analysis at the inverting input of the first operational amplifier, one finds the output voltage  $V_{out}$  as a function of the input voltage  $V_{in}$ , the offset voltage  $V_{off}$ , the fixed resistances  $R_1 = R_2$  and the variable resistance  $RA$ :

$$V_{out} = \frac{RA}{R_1}(V_{in} + V_{off}). \quad (7)$$

The offset voltage can be set by tuning the potentiometer  $R0$  (see the board in figure 13 and the schematic 16) and can be calculated under the assumption of an ideal operational amplifier that leads to a set up as given in 17. Here,  $V_{off}$  is given by:

$$V_{off} = 15V \frac{R_2(R_b - R_a)}{R_b R_2 + R_a R_2 + R_a R_b}. \quad (8)$$

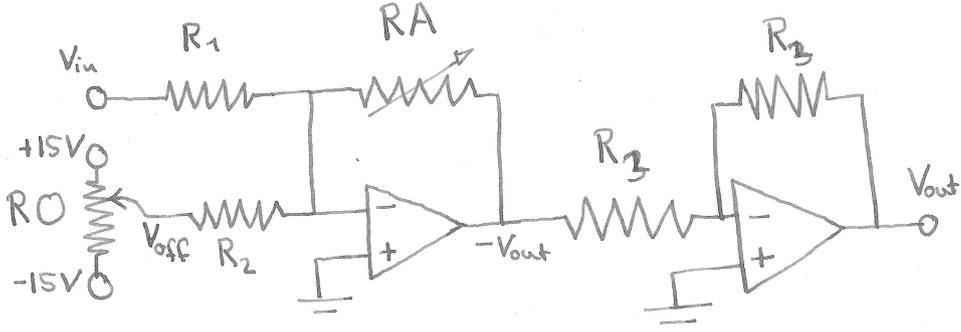
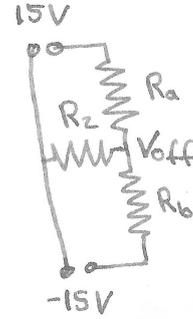


Figure 16: The figure shows a schematic of the input electronics. The elements as used in the tested PCB were  $R_1 = 3 \text{ K}\Omega$ ,  $R_2 = 1.4 \text{ K}\Omega$ ,  $R_O \leq 20 \text{ K}\Omega$ ,  $R_A = 5 \text{ K}\Omega$  and  $R_3 = 1.4 \text{ K}\Omega$ .

Figure 17: If we assume ideal operational amplifiers, the offset stage of the schematic 16 can be reduced to the schematic on the right.



Hence, the set up procedure corresponds to setting :

$$V_{\text{off}} = -(V_{\text{in}})_{\text{min}},$$

$$R_A = 5V \times \frac{R_1}{\Delta V},$$

where  $V_{\text{in}}$  is the minimal input voltage and  $\Delta V$  is the maximal range of voltages.

The output electronics (see figure 18) consist of a voltage divider followed by a non-inverting operational amplifier. For the set up it holds, that:

$$V_{\text{out}} = \frac{1}{2} \frac{R_1 + R_B}{R_1} V_{\text{in}}. \quad (9)$$

The values  $R_1$  and  $R_B$  should allow for a gain of at least 2.5, such that the upper end point of the maximal voltage range is around 6.5 V. In order to suppress high frequency fluctuations induced by the amplification process, both, in front of and after the amplification stage, low-pass filters with a cut-off frequency of about 2.5 KHz are installed. This corresponds to the maximal update rate of the Arduino,  $400 \mu\text{s}$ .

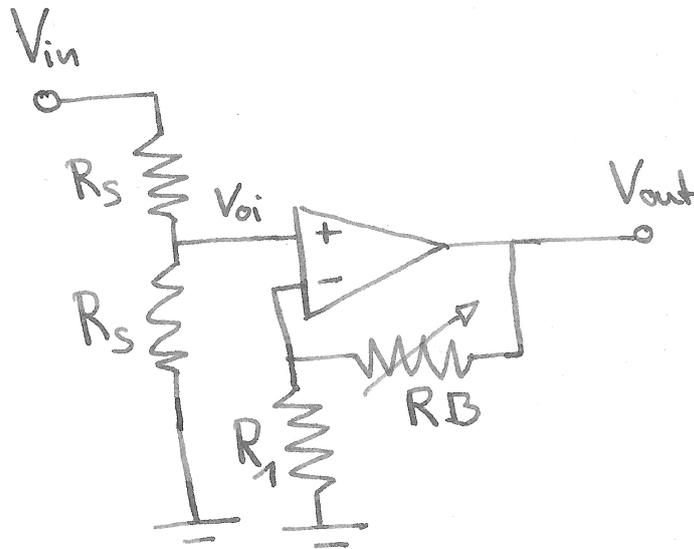


Figure 18: The figure above shows a schematic of the output electronics. In the set up used the values were as follows:  $R_s = 1.4 \text{ K}\Omega$ ,  $R_B \leq 5 \text{ K}\Omega$ ,  $R_1 = 1.2 \text{ K}\Omega$ . This allows for a net gain between 0.5 and 2.5 such that the upper end can be set to anywhere between 2.5 V and 6.25 V.

# D

## Test Analysis

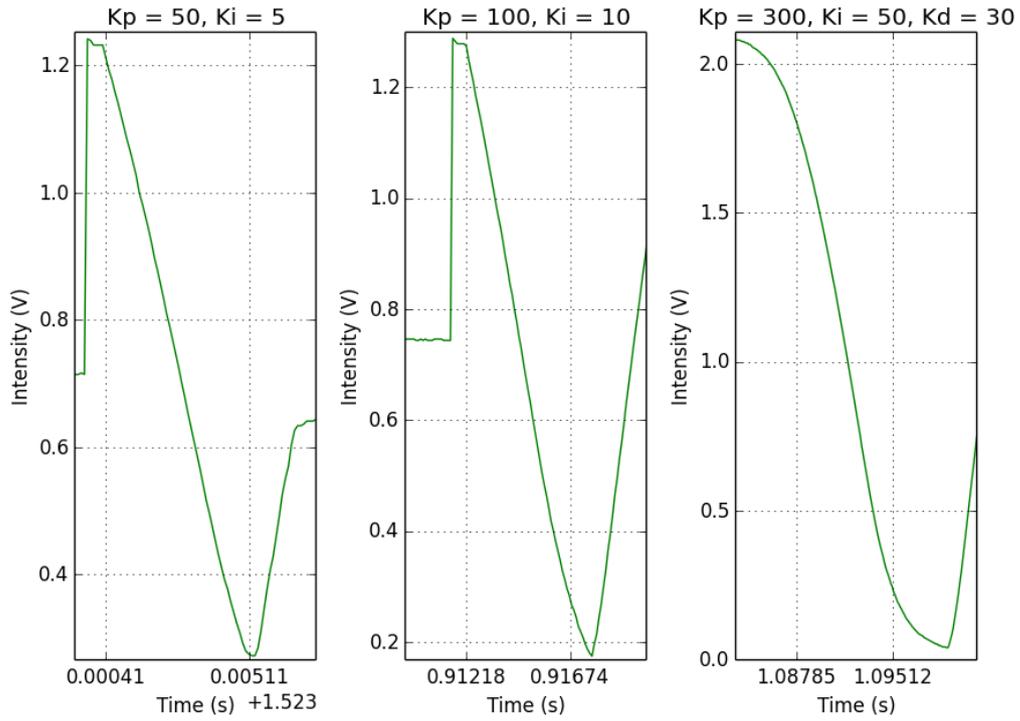


Figure 19: The graph shows the very similar maximal slopes of the signal correction after a step modulation of the incoming beam. One can associate this slope with the slew rate of the device.

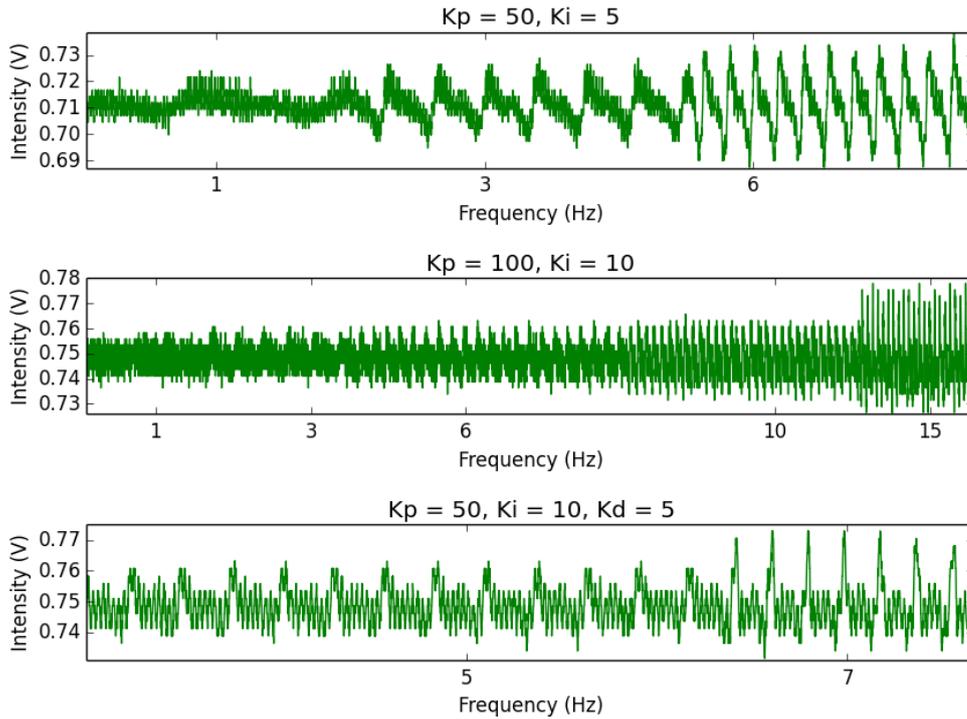


Figure 20: This figure shows a close up of the fluctuations after stabilisation in the low frequency regime. One can see that while the lower frequencies all show a similar amplitude in their fluctuations, the highest frequency visible in the graph respectively causes the fluctuations to stand out from those at lower frequencies. This step up of the fluctuation amplitude at a certain frequency was used as an estimate of the cut-off frequency in the respective cases.

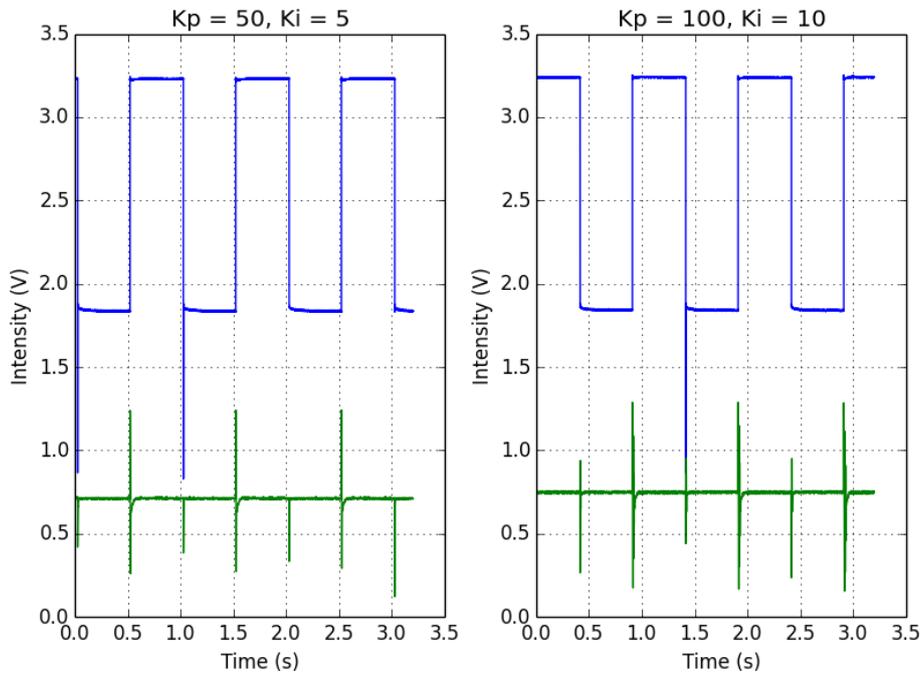


Figure 21: The figure shows the six step responses that were used to find a mean value for the settling time. Note that stepping up the intensity leads to longer settling times with oscillatory behaviour while stepping the intensity down is stabilised significantly faster.

## E

### Additional Programmes for Setting Up

In this section, a simple program to output 5 V from the Adafruit DAC of the Arduino is presented:

```
#include <Wire.h>
#include <Adafruit_MCP4725.h>

Adafruit_MCP4725 dac;

void setup(void) {
  dac.begin(0x62);
}
void loop(void) {
  dac.setVoltage(4095, false);
}
```

## References

- [1] Hudson Lab Wiki, *How to Align Polarization Maintaining Fibers*, [http://hudsonlab.wikia.com/wiki/How\\_to\\_Align\\_Polarization\\_Maintaining\\_Fibers](http://hudsonlab.wikia.com/wiki/How_to_Align_Polarization_Maintaining_Fibers) (18.02.2014)
- [2] Arduino Playground, *PID Library*, <http://playground.arduino.cc/Code/PIDLibrary> (24.02.2014)
- [3] Project Blog, *How to Improve the Beginners PID*, <http://brettbeauregard.com/blog/2011/04/improving-the-beginners-pid-introduction/> (24.02.2014)
- [4] Learn Adafruit, *12-bit-dac Tutorial*, <http://learn.adafruit.com/downloads/pdf/mcp4725-12-bit-dac-tutorial.pdf> (25.02.14)
- [5] digital-salvage, *Arduino and the LiquidCrystal Library*, <http://digital-salvage.net/?p=124> (25.02.14)
- [6] Arduino Forum, *Digital Read Fast*, <http://forum.arduino.cc/index.php/topic,46896.0.html> (25.02.14)
- [7] Horowitz and Hill, *The Art Of Electronics*, Cambridge University Press, 1989