Palacký University in Olomouc
Faculty of Science
Department of Optics

DIPLOMA THESIS

Possibilities of realization of logical functions using optical and optoelectronic methods

Drafted by:	**Tomáš Lamich**
Study program:	N1701 Physics
Study field:	Optics and Optoelectronics
Study form:	Full-time
Diploma thesis supervisor:	Ing. Zdeněk Řehoř Ph.D.
Submission deadline:	June 2020

Univerzita Palackého v Olomouci
Přírodovědecká fakulta
Katedra Optiky

DIPLOMOVÁ PRÁCE

Možnosti realizace logických funkcí optickými a optoelektronickými metodami

Vypracoval:	**Tomáš Lamich**
Studijní program:	N1701 Fyzika
Studijní obor:	Optika a optoelektronika
Forma studia:	Prezenční
Vedoucí diplomové práce:	Ing. Zdeněk Řehoř Ph.D.
Termín odevzdání práce:	Červen 2020

Declaration

I declare, that I have elaborated the submitted thesis independently under the supervision of Ing. Zdeněk Řehoř Ph.D.and that I utilized the sources I cite in the list of sources.

In Olomouc on June 5, 2020

. . . .

Tomáš Lamich

Acknowledgement

I would like to thank my supervisor Ing. Zdeněk Řehoř Ph.D.for outstanding patience and incredibly useful advice he provided during my work. I would also like to thank my colleagues, namely Ondřej Vaculík, Michal Neset, Jiří Fadrý, and Jan Grygar for deeply inspiring discussion in the time between our classes.

Bibliografická identifikace

Jméno a příjmení autora	Tomáš Lamich
Název práce	Možnosti realizace logických funkcí optickými a optoelektronickými metodami
Typ práce	Diplomová
Pracoviště	Katedra optiky
Vedoucí práce	Ing. Zdeněk Řehoř Ph.D.
Rok obhajoby práce	2020
Abstrakt	Tato práce se zabývá řešením problému optického počítání a jeho využití k řešení výpočetních problémů. V úvodu práce jsou vymezeny základní pojmy, které jsou náledně využity k porovnání jednotlivých přístupů v oblasti optického počítání. V další části práce jsou popsány jednotlivé způsoby konstrukce logických funkcí, které jsou následně rozšířeny pro možnosti paralelního počítání. Tato myšlenka je poté využita k zobecnění problému a konstrukci optického integrátoru, použitelného k řešení diferenciálníh rovnic ve 2D. V poslední části práce je integrátor setrojem a jeho funkce ověřena pomocí měření integrace předem připravených funkcí. V samotném závěru jsou uvedeny nápady ke zlepšení a rozšíření tohoto integrátoru.
Klíčová slova	Optická hradla, optický integrátor, plošný integrátor, optický počítač, optické počítání
Počet stran	57
Počet příloh	0
Jazyk	anglický

Bibliographical identification

Autor's first name and surname	Tomáš Lamich
Title	Possibilities of realization of logical functions using optical and optoelectronic methods
Type of thesis	Master
Department	Department of Optics
Supervisor	Ing. Zdeněk Řehoř Ph.D.
The year of presentation	2020
Abstract	This thesis follows up the solution of the problem of optical computation and its use to solve computational problems. The introduction defines the basic concepts that are subsequently used to compare the various approaches in the field of optical computing. The next part of the work describes the various ways of constructing logical functions, which are then extended for the possibility of parallel computation. This idea is then used to generalize the problem and construct an optical integrator that can be used to solve differential equations in 2D. In the last part of the work, the integrator is assembled and its function is verified by measuring the integration of pre-prepared functions. At the very end I proposed ideas for improving and expanding this integrator.
Keywords	Optical gates, optical integrator, spatial integrator, optical computer, optical computing
Number of pages	57
Number of appendices	0
Language	english

Contents

Introduction

While working on my bachelor's thesis covering deterministic chaos in electrical circuits I found many possibilities of use for Chua's circuit. Among them one stood out for me, and that was a way of utilizing Chua's cicuit for random bit generation [1] where the authors showed and utilized very interesting approach to tackling more than one problem.

While studying the phenomena of chaos in electronics I also was more intrigued by the physical representation of information and the principles of computation using electronics. This with combination of my curriculum resulted into the thought of optical computation which promises the possibility of much faster processing units for the computers of the future. This could lead to better mathematical modelling as well as faster predictions since the overall speed of the computation should increase considerably while maintaining the heat output or even lower the heat output of optics based chips.

In this thesis I am going to cover the fundamentals of computation and the difference between electronics a optics based computations with their advantages and disadvantages. From that point I shall be able to describe the most prominent branches of optical computing and compare them. Afterwards I will propose my own design of all optical gates and all integrator, that I demonstrate experimentally and use to solve differential equation in a manner not dissimilar to conventional analogue computers.

Chapter 1

Computing from hardware perspective

In this chapter I am going to present basics of computing principles such as often used functions and the difference between continuous and discrete value computing. Following section is going to cover the topic of physical representation of values based on the platform used for computing as well as some simple technical solutions for basic operations. The last section in this chapter is going to deal with the problematic of optical computing and current trends in this field including their advantages and disadvantages.

1.1 Basic principles of computing

When we talk about computing we need to specify if we talk about discrete value computing widely utilized in computers, which is based on logical operations, or continuous value computing. Continuous value computing, also called analogue computing, is based on operations with continuously variable aspects of certain physical phenomena, through which we can model and solve certain problems. In the following sections I am going to describe basic operations we can do with either the former or the latter mentioned method.

1.1.1 Logical operations

Whenever we want to perform any logical operation we need discreet values of 0 (false) and 1 (true) represented in some way. Some might argue that this statement is false, and bring up either fuzzy logic or higher-valued logic, but fuzzy logic is in principle a continuous variable based system and higher-valued logic (e.g. trinary logic) isn't widely spread and used, therefore I focus only on binary (or Boolean) logic and any further use of the term "logic" and its forms is referring to binary logic.

Considering logical functions we have three basic operations: NOT ($\neg$), AND, and OR, and three secondary operations: XOR, implication ($\rightarrow$), and equivalence ($\leftrightarrow$). The former three are called basic, because we can construct the latter three from their combinations. All of the truth values are listed in (table 1.1).

These operations can be combined to form other, most commonly used ones are NAND, NOR, and NXOR (which gives us an equivalence) where "N" stands for negated.

Table 1.1: Truth table of all mentioned logical operations.

x	y	¬x	x AND y	x OR y	x XOR y	x→y	x↔y
1	1	0	1	1	0	1	1
1	0	0	0	1	1	0	0
0	1	1	0	1	1	1	0
0	0	1	0	0	0	1	1

Now is a good time to introduce the term logical gate. Logical gate is an ideal physical device performing a logical function on one or more binary inputs. The symbol for each gate is to be seen at (figure 1.1).

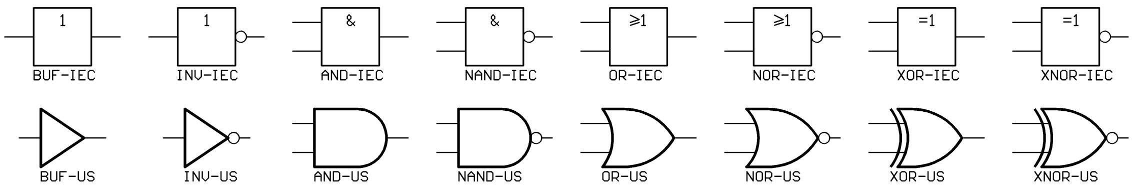

Figure 1.1: EU (top) and US symbols for logical gates [2].

With these basic elements we can build higher logical functions by combining these gates into circuits which can be one-way, where inputs are processed and then sent to the outputs, or cyclical, where outputs of one operation is the input of the next one.

1.1.2 Analogue operations

If we compare the amount of basic operations available for binary and analogue computation, we get incomparably more possibilities in the analogue world. We can say, that in the analogue world we can do any mathematical operation we want. This, however, is rather bold claim, because we also need a platform to perform the operation. Keep in mind, that some operations may not be efficiently feasible on certain platforms.

1.2 encoding

While mathematics is using very strictly defined values, physics we use to represent them does not like to abide by those rules and we need to code our values into some property of used medium and then decode them to interpret the output of our operation.

If we want to use binary system than the encoding can be rather simple, since we need 2 values of single property and a way of switching between those 2 values. The simplest way of encoding this way is intensity encoding where a high value of signal represents either 1 or 0 and low level represents the opposite (0 for the former and 1 for the latter). However we can use other properties as well, phase, polarization, or spin just to name a few. The choice of the property we use for encoding is dependent purely on the medium, and possible operations we can than perform within the constraints of physical interactions we have at hand.

In the analogue realm, the problem of encoding is more complicated because we need to consider much more than just two levels and a few simple operations. We need to prepare the input with enough power to be easily detectable by the output while keeping in mind the limitations of the elements used. For example if we have a differential equation such as this one:

$$a\frac{d^2f(t)}{dt^2} + b\frac{df(t)}{dt} = g(t), \tag{1.1}$$

we can solve it by chaining two integrators which we limit in a way that the limitations on the integrators will follow the $\frac{a}{b}$ ratio.

1.3 Computing in the physical realm

In the previous section I described the basic idea behind encoding values into a physical property. This section is going to describe the basic building blocks and the principles of computation based on the medium we choose. Namely I am going to cover the subject of electronic and optical computation and the possibilities for both digital and analogue operations.

1.3.1 Electronic computing

Both the digital and analogue circuits have the same basic building blocks, which are transistor, resistor, and capacitor.

Basic components

Resistor is the simplest of the three, while the material used for construction may vary, the basic principle remains unchanged. Resistor adds resistance to the electronic current, which can be calculated from the Ohm's law:

$$V = R \cdot I, \tag{1.2}$$

where V stand for voltage, R for resistance, and I for current.

Capacitor is a simple electronic component as well, consisting of two conductors separated by a insulating region. This way capacitor can retain electric charge, therefore it acts as an energy. The capacitance of capacitor can be calculated as

$$C = \frac{Q}{V}, \tag{1.3}$$

where C is capacitance, and Q is charge of the capacitor. There are many types and forms of capacitors for many different uses, however that is not the purpose of this thesis.

The most complicated of the basic elements is the transistor, which can be constructed in two ways. The bipolar transistor (figure 1.2) is based lowering the charge barrier in base by input voltage therefore allowing the carriers to pass through between collector and emitter. MOSFET (figure 1.3) is similar in function, but the medium that allows carriers to pass through the bulk is not carrier injection but an applied field from the gate electrode, which opens a channel through which the carries may pass between source and drain. Even though MOSFET and bipolar transistor work on different principles, they both are nearly the same in function, applied voltage to the gate or base allows carriers to pass which is same function as switching between on and off states.

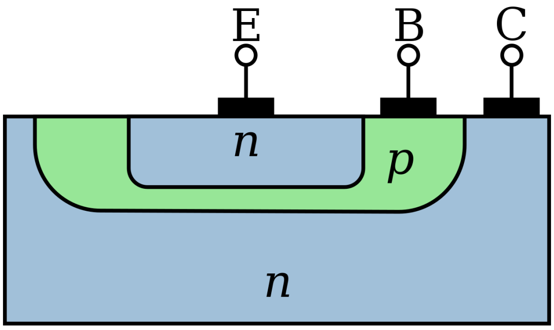

Figure 1.2: Bipolar transistor, E stands for emitter, B for base, and C for colector [3].

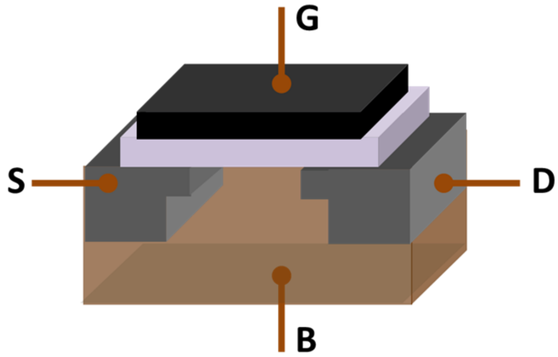

Figure 1.3: MOSFET, G stands for gate, S for source, D for drain, and B for body or bulk [4].

Building simple gates

We can build all the simple gates using transistors. Since the actual construction may differ based on manufacturer and model of the gate, I am going to present the simplest solution based on MOSFET technology. Also I am going to assume that high level of signal is equivalent to logical 1 and low level equivalent to logical 0. If we would like to swap the to the inverted encoding, we can simply add inverter to each of the inputs.

First gate I would like to focus on is the AND gate (figure 1.4). By setting either A, or B on high, you open the respective transistor but not both, which results is the OUT transistor to remain open and the signal goes through towards the ground, but if you set both A, and B high, the OUT transistor closes, resulting in sending the signal towards the output, which we detect.

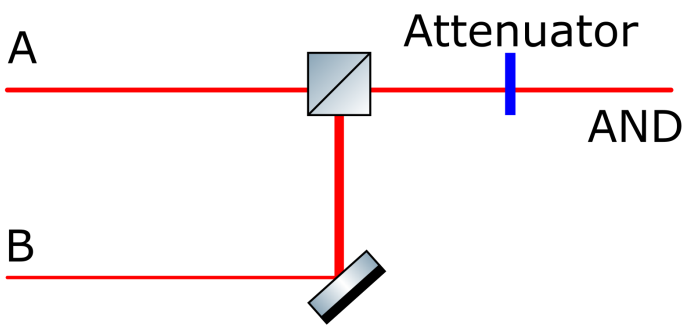

Figure 1.4: AND gate with two inputs A and B.

Next very important is NAND (figure 1.5). The only case when the signal does not go towards the output is when both A and B transistors are open which can be done only when both the inputs are high. In all other cases the output is at the state of logical 1 resulting in the NAND function.

By using the NAND gates we can construct all the other gates, this however usually is not the most efficient way, since chaining gates increases the processing time. For example if we want an inverter all we have to do is connect the inputs of NAND gate.

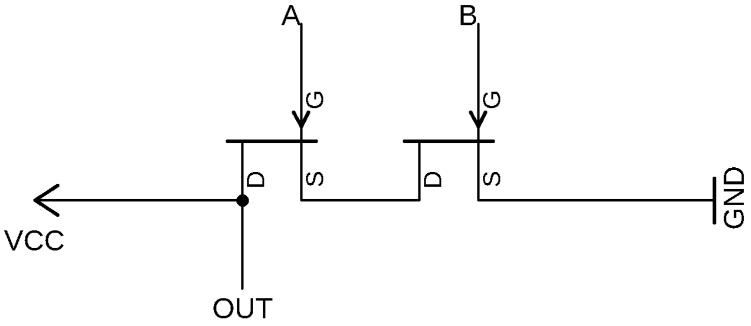

Figure 1.5: NAND gate with two inputs A and B.

Analogue computers

If we want to compute a solution to any differential equation, we need to be able to integrate and differentiate, which we can do with logical operations, however we would introduce many sources of error into the solution, such as rounding error. This can be countered by using good algorithms, or by using analogue computers.

Basic building block of an analogue computer is an operational amplifier, which consists of differential amplifier at the input stage which amplifies the difference between the inputs, voltage amplifier, providing high voltage gain, and an output amplifier, that provides high current gain, current limiting, and short circuit protection. The operational amplifier has two inputs, one of them being inverting and the other regular.

By using a feedback loop we can build a differentiator (figure 1.6). This circuit measures the change in input voltage by measuring the current through the capacitor and then outputs a voltage proportional to that current. The output voltage can be calculated by a formula

$$V_{out} = -RC\frac{dV_{in}}{dt}, \tag{1.4}$$

where R is resistance of the feedback resistor and C is capacity of the input capacitor.

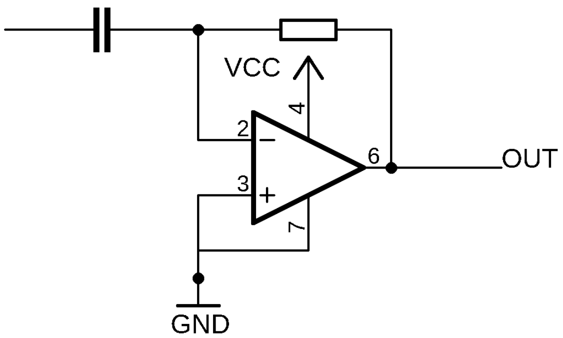

Figure 1.6: Differentiator circuit using operational amplifier.

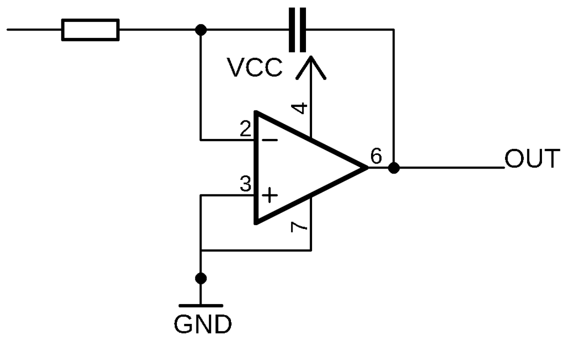

Figure 1.7: Integrator circuit using operational amplifier.

by switching the resistor's and capacitor's position in the circuit we get a integrator (figure 1.7). This circuit will produce an output voltage proportional to the magnitude and duration of input voltage deviated from 0 V. Therefore producing output voltage following equation

$$V_{out} = \int_0^t -\frac{V_{in}(t)}{RC}dt + c, \tag{1.5}$$

where $c = V_{in}(0)$, R is the resistance of the input resistor and C is the capacity of the feedback capacitor.

encoding in electronic computers

If we want to code an information so we can send it through electrical circuit, we have multitude of possibilities, whether it is a frequency of the field, phase, amplitude, or multiple possibilities of pulse shapes, when we want to use the signal in computing, we only use encoding into amplitude (at least for the usual applications, I am describing in this thesis).

Solving problems using electrical computer

Let us have a model problem of solving a differential equation in the shape of

$$a\frac{d^2 f(t)}{dt^2} + b\frac{df(t)}{dt} + cf(t) = g(t), \tag{1.6}$$

where a, b, and c are real parameters. We can either solve this equation analytically using pen and paper, a software (which can use either numerical or symbolic solution), or a analogue computer using two integrators and a detection unit, such as an oscilloscope (figure 1.8).

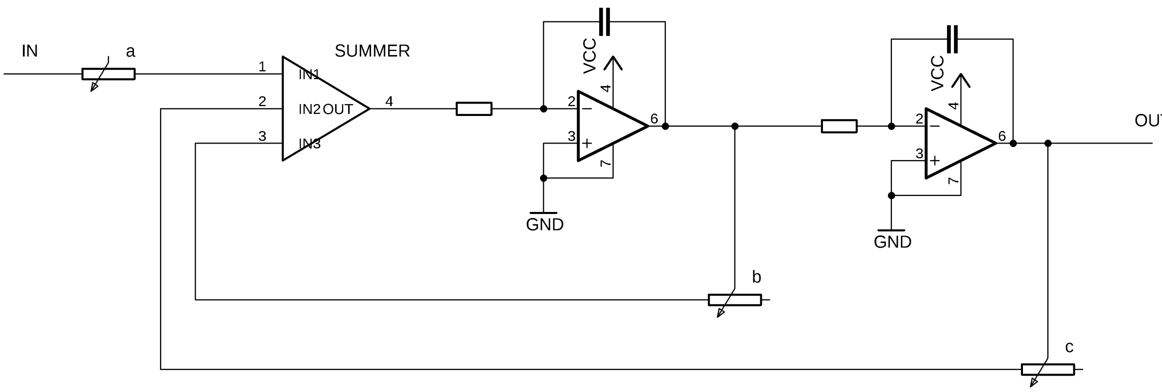

Figure 1.8: Analogue computer capable of solving (1.6) type of equations.

While these computers did not find their way into the wide use by the general public, mainly due to the unintuitive user interface and much more complicated use, there has been a new wave of digital-analogue hybrid computers, using digital for control and data storage while the analogue part performs the operations needed. This new research path promises a specialized, power efficient, and fast computers [5].

1.3.2 Optical computing

To cover not only the approaches but also the capabilities of optical domain, we must broaden our horizons, because there is no major principle on which all of the computers are being build. To add to the, maybe confusing, large number of possible platforms, the problem of encoding information into the physical medium and retrieving that information as well as controlling the entire process are not set in stone, since all of those are in the process of development.

Encoding of information in optical domain

Since optical signal is a for of electromagnetic wave, we can use all the properties we already know from electrical computing and apply them here. This gives us the possibility of encoding information in the intensity, phase and pulse shape. However light has more interesting properties into which we can encode information, such as polarization or wavelength of the signal.

The first and simplest encoding is intensity based encoding, where the level of radiant energy (and thus detected intensity in any units we choose based on our equipment) is directly related to that of encoded information. The basic example can be Low-High intensity encoding, where we have two discreet levels divided by a threshold and these two levels are assigned the values of either 1 or 0. Another approach, more useful in analogue computing, utilize setting a level l, which is assigned a value desired a for the computation and afterwards the information is retrieved in the form of

$$x = a\frac{m}{l}, \tag{1.7}$$

where x is the desired numerical value of the result, m is the measured value of the operation results and l is the reference measurement value.

Second possibility is the encoding into the phase of the optical signal. Here we require a phase-stable light source, such as laser. Again the process is rather straight forward, as we divide the phase into discreet intervals, and assign them their discreet values. The fact that phase has a periodic range in the interval of $[0,\pi]$, makes phase encoding a bit impractical to use for analogue computation, because if we want to use its full range, the highest and the lowest values would be indistinguishable from one another.

We can effectively combine the two aforementioned methods of encoding. Through this we achieve, what is called quadrature amplitude modulation (QAM), which can be best describe using a constellation diagram of one (figure 1.9). For the purposes of

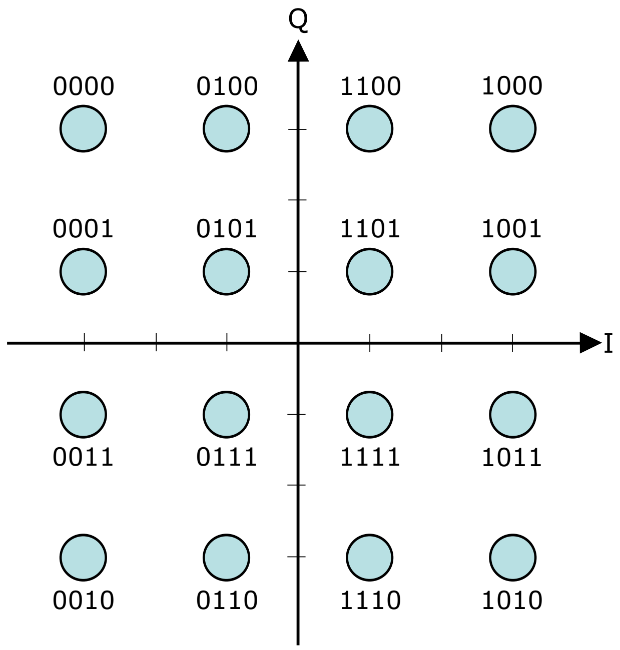

Figure 1.9: 16-QAM constelation diagram including Gray's coding [6].

optical computation, where we want to work with either single bits, or analogue values of a single physical property, the QAM is not a useful encoding option, since it gives us neither of the desired results. Nevertheless QAM is extremely useful for information transmission since we can code multiple bits into one symbol and this way increase the capacity of the transmitted information.

Third comes the encoding into polarisation. For this we require a polarised light in the beginning of the encoding process. We then assign each polarisation state a value, and through polariser on the detection unit, we can retrieve the encoded information as a change in the intensity. Based on the relative angle between the input polariser and the detection polariser, we can detect intensity in the interval of $[I_{max},0]$ corresponding with the relative angle interval of $\left[0,\frac{\pi}{2}\right]$. While being very interesting, this method in the end is no different than intensity coding. However we can use polarisation for intensity encoding and modulation.

Last comes the encoding into different wavelengths. This can be done using non-linear optics effects, such as second harmonic frequency generation, sum frequency generation, or any other frequency altering phenomena. The information retrieval is a simple case of using a wavelength selective element, such as diffraction grating and a spatial detector, whose areas would correspond with the encoded bit values. By using a white light continuum source with a laser for sum frequency generation, we could even get a continuous spectrum of read-out values, based on the position allowing further possibilities [7].

Wavelengths and their use

While is it possible to use wavelengths for encoding, we can use this property of light in another way, that may prove much more useful. It allows us to compute multiple problems simultaneously, thanks to the fact, light is a parallel wave, and thus does not interact with itself, unless both spatial and temporal properties are similar enough.

This would also allow us to build a chip, that does not need multiple layers of metal conductors to provide communication between the elements, but to use different wavelengths, and thus reducing the material consumption. This method is currently used in optical communication, by sending multiple packets of information at the same time, but on different wavelengths and thus increasing the channel transmission capacity.

Elementary units for optical computing

Unsurprisingly, since we are talking about the problem of computing, we want to get the same elementary building block as in the case of electronic computing. These basic unit is an all optical transistor.

The first problem arises with the construction of an all optical transistor, behaving as a optical switch. While there have been attempts to build an all optical transistor, the results were either bulky (Fabry-Perot resonator based design) or otherwise impractical to use (again, Fabry-Perot resonator design based on constructive and destructive interference of incident light). Quite recently, a new design based on organic electronics and their optical properties was published [8]. Given the novelty of this design, and the impracticality of realising an optical computer by using all optical transistors, I decided to look at a higher level of circuitry and choose logical gates, integrator and differentiator as the basic building units.

Regarding optical gates, integrator and differentiator. The possible design solutions are discussed in the next section.

1.4 Design of optical computing units

The previous section delved into the problematic of computing from the perspective of hardware and it is physical properties, based on the medium chosen for computation. The final topic - design possibilities of optical gates, integrators and differentiators - was left open because of its vastness. In this section, I am going to address the most prominent design platforms of optical gates, integrators and differentiators, and in the end I shall summarize the advantages and disadvantages those.

1.4.1 Logical gates

Let us firstly look at the problematic of logical gates. Even though there are more possibilities, I am going to cover Mach-Zehnder interferometer (MZI) based designs, photonic crystal based designs and ring resonator based designs. I chose the first three because a lot of the scientific papers concerned by designing optical gates are covering these three fields.

MZI based logical gates

The underlying principle of all the MZI gates is interference of light. To understand the design principles, let us firstly discus the Mach-Zehnder interferometer (figure 1.10). By

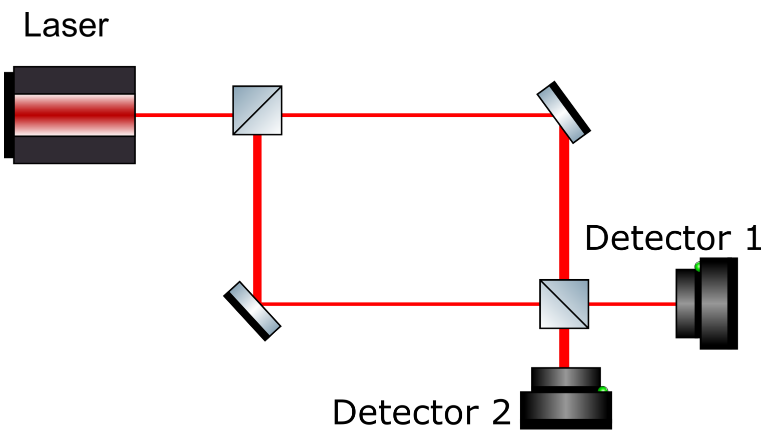

Figure 1.10: Scheme of Mach-Zehnder interferometer (BS stands for beam-splitter).

splitting the laser beam using a beam-splitter we obtain two beams travelling through the branches of our interferometer. These meet at the second beam-splitter and interfere with each other. The detected intensity I follows the equation

$$I = I_1 + I_2 + 2\sqrt{I_1 I_2}\cos\phi, \tag{1.8}$$

where, I_1 and I_2 are the intensities in lower and upper branch of the interferometer respectively and ϕ is the phase difference induce between the two. For our case of 50:50 beam-splitter we obtain that $I_1 = I_2 = I_{in}$ therefore simplifying the equation into the form of

$$I = 2I_{in}\left(1 + \cos\phi\right). \tag{1.9}$$

Without adding any other phase difference, all the intensity will be detected by detector 1 (the phase shift in this case would be $\phi = 0$). Detector 2 would detect that the incoming intensity is equal to zero, which is the result of destructive interference ($\phi = \pi$).

By increasing the value of ϕ by inducing a phase shift on our phase shifter in the lower branch, we will observe that, the second detector starts to detect a non-zero levels of intensity and the intensity detected by detector 1 is decreasing. When we reach $\phi = \frac{\pi}{2}$ both detectors detect same level of intensity that $I_{d1} = I_{d2} = I_{in}$.

We can use this phenomena to construct a simple inverter by shielding one of the outputs and adding a phase shift of π into the signal arm of the interferometer (figure 1.11). The idle branch uses a continuous wave while signal comes in predefined pulses

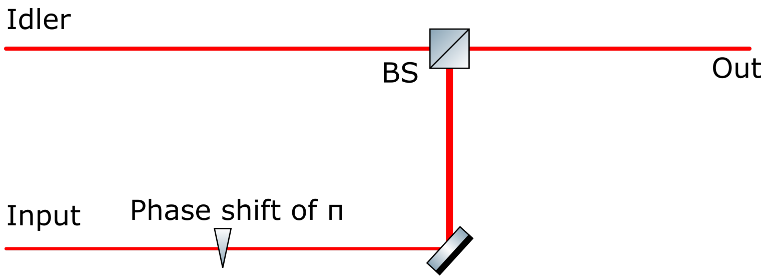

Figure 1.11: Scheme of MZI based inverter (only the second beam-splitter of the interferometer is in the scheme).

and the destructive interference on the beam-splitter is providing the bit flipping.

More advanced architectures usually utilizes a 'push-pull' method utilizing a continuous wave signal that is 'pulled' through an unbalanced interferometer giving us a desired logical function. A good example of such device is a XOR gate [9] (figure 1.12). The output on the right hand side gives us desired XOR function as a series

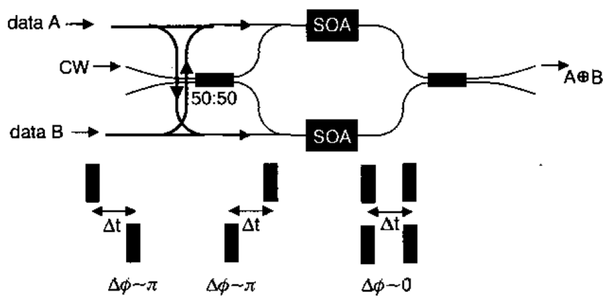

Figure 1.12: Scheme of MZI based XOR (SOA stands for semiconductor optical amplifier) [9].

of continuous wave pulses of the duration Δt. The data streams of A and B always interfere destructively due to the phase shift, that is the result of the delay caused by splitting the signal into both branches.

Another way of constructing MZI based gates is using semiconductor optical amplifiers (SOA) for phase switching, and thus producing either constructive or destructive interference. Through this method an AND gates can be build as is shown in the figure (figure 1.13) [10]. While utilizing two wavelengths for inputs A and B, the input A works as an idle branch, utilizing a high intensity pulses, and the B input behaves as a probe signal, needing only low intensity inputs. If the input A is at zero level, not phase switching happens at the SOAs so the B signal interferes destructively with itself. In the case of both inputs being at the 1 level, the phase of signal B is shifted, so that the interference is constructive, thus resulting into a 1 on the output. In all other cases, either the wavelength is filter out, or no light is present, therefore resulting

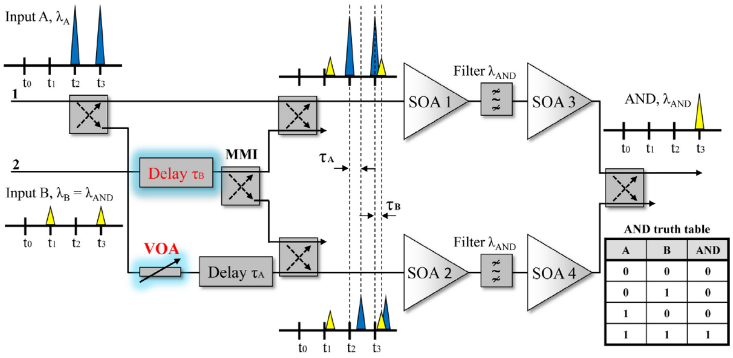

Figure 1.13: Scheme of MZI based AND gate [10].

in 0 level on the output. This gate (figure 1.13) currently holds the record speed of 640 Gbit/s, however has one huge drawback in needing two distinct wavelengths, with only one being on the output. This may prove a challenge if we wanted to connect multiple gates in series, we would need some way of wavelength switching between the two wavelengths.

Photonic crystals based logical gates

Firstly we should consider, what actually is a photonic crystal. By definition it is a periodic structure of materials with two different refractive indexes. This can be done in one, two or even three dimensions. The one dimensional system is widely used today in production of thin layers, such as anti-reflex layers on glasses or camera lens. However, if we need more than one signal beam to interact with each other, one dimension is not enough.

Even though three dimensional photonic crystals give us a lot of promising features, such as dividing layers, using one as a conduit of signal to a gate on a different layer, which is basically a architecture used today in electronic chips.

Using two dimensional photonic crystals allow us to manipulate individual data streams in form of light signals, while maintaining a relatively simple architecture of the photonic crystal itself. This is the reason why we are going to focus solely on two dimensional photonic crystals in this section.

By creating a periodic structure as a photonic crystal, we create a band gap-similar region in optical spectrum due to destructive interference between the transmitted and reflected waves. If we leave a region without such a periodic structure, we create a waveguide-like structure allowing the light to pass through (figure 1.14. We can see

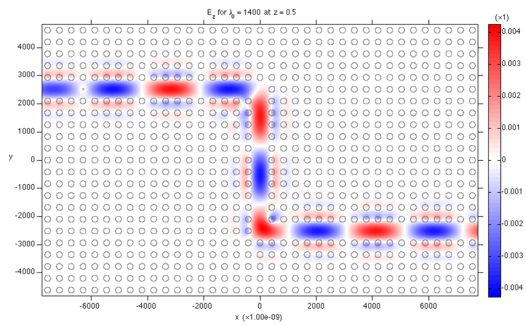

Figure 1.14: A waveguide using a 2D photonic crystal (simulation) [11].

that a close region around the waveguide also allows some of the signal through, this phenomenon is called evanescent wave (or evanescent field) and is utilized in waveguide optics in form of a waveguide beam-splitters or fiber tapping devices.

For the purposes of computing, we can use this evanescent field as well and by optimizing the parameters of inter-channel transmission we can build logical gates. For example we can construct an OR gate by utilizing two inputs A and B, ring cavities and a "Y" shaped branch for the output Y (figure 1.15). The crystal itself is consisting of silicon rods in triangular lattice.

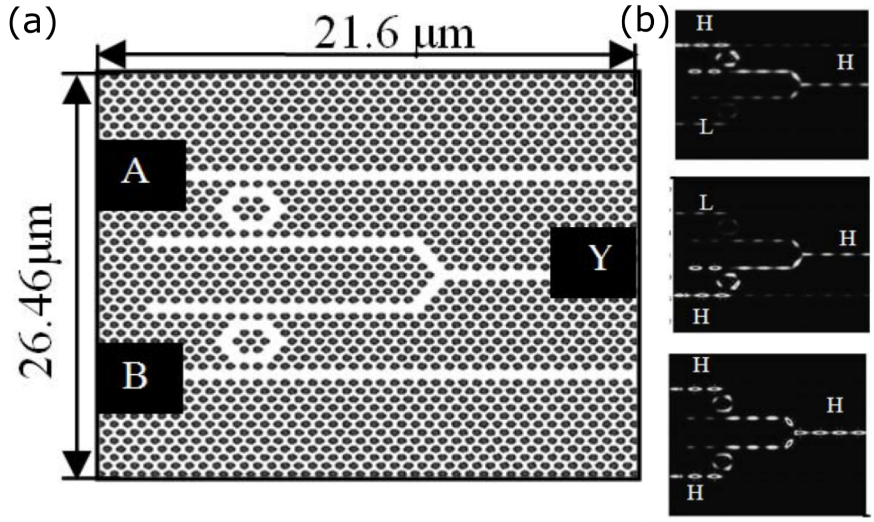

Figure 1.15: (a) proposed OR gate using 2D photonic crystal, (b) steady field distribution in the proposed OR gate ("L" and "H" represent low and high level of signal respectively) [12].

We can see a visualization of steady field distribution in the proposed OR gate (figure 1.15). As we can see, majority of the input gets transferred from the input channel into the output channel through the ring cavity.

By utilizing a second ring cavity and further optimization of parameters, we can build an AND gate in similar fashion (figure 1.16). The visualization of steady field distribution can be seen in a figure (figure 1.16). As we can see, both of these gates provide us with a close representation of the logical functions, they are designed to do. Also the falling time of the AND gate is below 5 ps, which gives us possible speed of up to 200 Gbit/s and for the OR gate the falling time is below 2.2 ps, that allows for speeds of up to 454 Gbit/s [12].

Optimizing of inter-channel transmission is not the only possibility of designing optical gates on 2D photonic crystals. We can also induce an interference on a channel intersection, and by optimizing the parameters of the intersection, we can create the logical gates as well. A good example of this approach is a "Y" port based architecture with an optimized hole in the intersection (the photonic crystal in this case is composed of triangular lattice air holes in silicon bulk) (figure 1.17) [13].

Thanks to optimization of the intersection hole, and reference field applied at port "R", we can achieve all the logical functions. A good example of this is an AND gate (figure 1.18).

The strongest point of this architecture is the application of the reference channel, through which we can control the logical function a gate is producing. however this adds

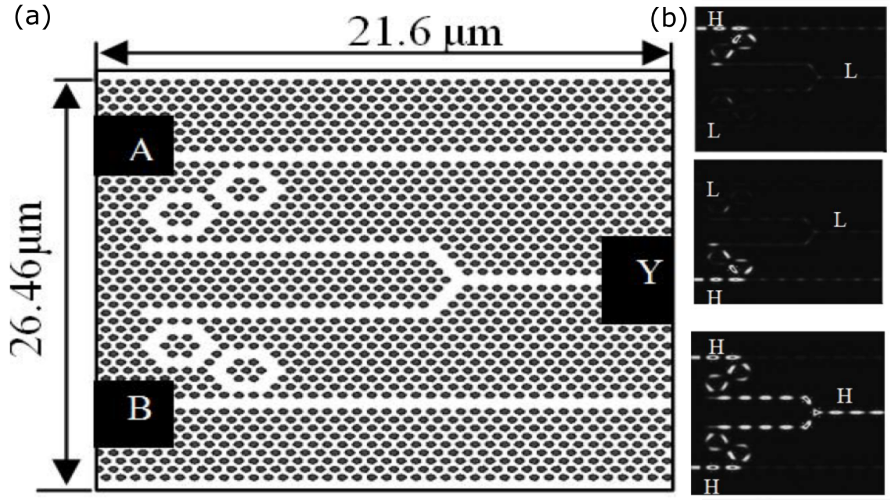

Figure 1.16: (a) proposed AND gate using 2D photonic crystal, (b) steady field distribution in the proposed AND gate (”L” and ”H” represent low and high level of signal respectively) [12].

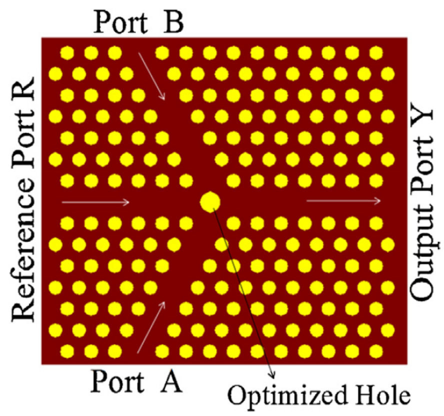

Figure 1.17: Proposed all optical AND gate using a 2D photonic crystal [13].

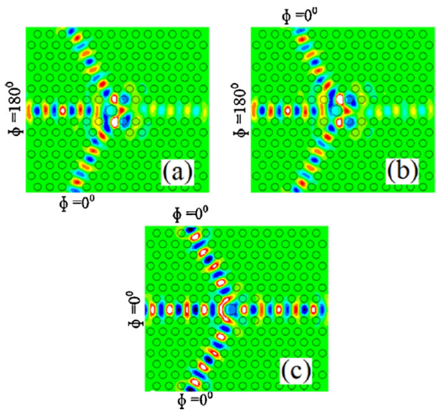

Figure 1.18: Steady field distribution in the proposed AND gate (a) A=1, B=0, R=1; (b) A=0, B=1, R=1; (c) A=1, B=1, R=1 [13].

a requirement of monitoring the input channels, and based on their signal switching the phase of the reference field, which may prove a complicated thing to do.

Microring resonators based gates

This section is based on [14]

A microring resonator is an optoelectronic device, that is in the shape of a ring, its size is typically below tens of µms and acts as an resonator.

Such a device may be manufactured in a similar fashion as any electronic chip, but we require a direct transition material, such as GaAs, or InP. An example of such a device can be seen in the figure (figure 1.19(a)).

This AND gate works on the principle of a two-photon absorption, that generates free carriers. Through proper design methods, we can achieve low switching power for a high speed data streams. Thanks to the multiple resonant frequencies, we can

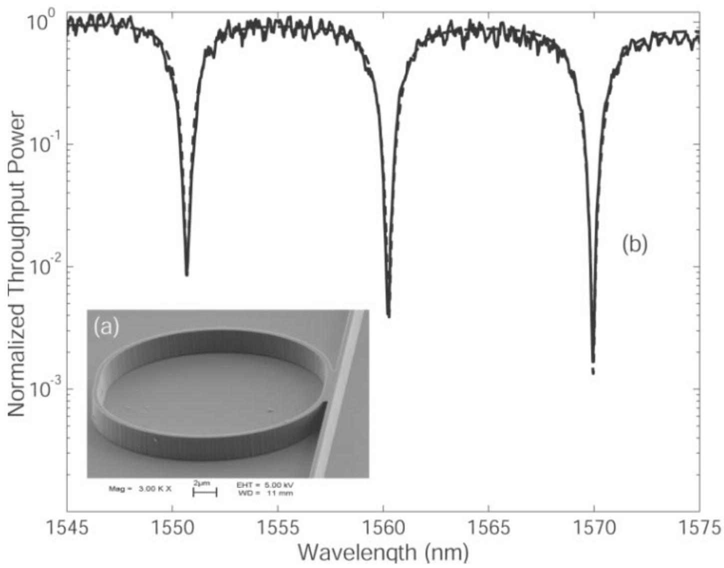

Figure 1.19: (a) Electron microscope image of an InP microring resonator used for AND and NAND logical gates; (b) spectral response of of the throughout port [14].

see in the graph of the spectral response (figure 1.19(b)), we can use a pump pulses that generate free carriers, resulting in temporal decrease in refractive index on the resonator, which results in a blue shift in the resonant frequencies. In the case of critically coupled ring, the probe beam is tuned into resonance, which results in highly attenuated signal on the output and thus giving us a logical 0. In the case of either A, or B signal being 1 on the input, the acquired phase shift is not high enough to switch the probe out of resonance, but in the case of both A, and B being 1 on the input, the amount of generated carriers is enough to switch the probe out of resonance, resulting in a logical 1 on the output. As we can see, this gives us an all optical AND gate. We can easily achieve a NAND functions by switching the initial sate of the probe to be out of resonance.

As we can expect, the speed of these devices would be limited mainly by the lifetime of carriers. However it is possible to achieve speeds up to 30 Gbit/s. This can be radically improved by better optimization of the devices as well as new materials, with lower carrier lifetimes.

Discussion of advantages and disadvantages of mentioned logical gate approaches

Before we approach the specific architecture ideas, we should deal with the elephant in the room. The speed of the gates looks highly promising, and can reach tens or even hundreds of GHz, which compared to current CPU frequency record of 8.722 78 GHz [15] can provide us with a step increase of computational speed by factor of 10 or more. There is a big difference in size of a device, since nowadays we have a technology of manufacturing transistors smaller than 10 nm, the optical gates dimensions are in the range of µms or bigger. From the point of size, the most promising seems gates based on microring resonators, that can be as small as 10 µm in diameter [14]. Photonic crystal based gates we discussed were about twice the size [12] and since I wasn't able to find any dimensions of the MZI gates, but those are primarily based on waveguide optics and therefore are not necessarily integrated on chips, we can expect that the experimental set-ups were considerably larger.

Let us now look at the MZI based gates once more, and discuss the advantages and disadvantages of those. Firstly the speed, currently the record of 640 Gbit/s is held by an MZI gates based on turbo-switched interferometers. This is more than twenty times higher compared to the microring gates we discussed earlier. However the speed is bought by other properties, that may prove problematic.

The first problem that arises, apart from the size, is the need of interference. To get a stable output, we require a stable environment free from mechanical disturbance. This would be a very hard to achieve in a general use environment, such as a desktop PC.

If we move towards the photonic crystal gates, we see that we sacrifice a bit of speed, but we also gain a stability in the output, due to the fact, that there is no interference needed to achieve the logical functions, we want. However there is a risk of needing a high power light sources, which can lead to lowering of the device's lifetime. We may also encounter another problem, that is due to the principle of the gates, and that is lowering of "1", because as the signal progresses through the gate, the output "1" needs to be above threshold of the physical signal, that may result in false zeros as the signal progresses through series of such gates, as the signal drops in intensity.

If we look at the last discussed architecture, we see, that we get a decent speed with a good signal to noise ratio as well as no need for stabilisation, since we do not need an interference of signals. However thanks to the construction of such device, we require to high quality lasers with very well defined spectra and narrow bandwidth. This could increase the cost of a device above the level of usefulness.

In conclusion, all of the mentioned architectures have their advantages and disadvantages, which made me take a slightly different path of using a free space optics. This is going to be elaborated on the next chapter.

1.4.2 Analogue units

In this section I am going to discus the problematic of optical integrators, which are needed for analogue computing, and therefore solving differential equations.

Optical integrator

To integrate any signal, we need a device, that can output a signal proportional to the sum of signal that is on the input port of such device. That poses a question "why is this a problem, since we have a CCDs?" And yes, it is true that charge coupled device (CCD), such a matrix detector in any camera is in sense an integrator. However what we seek a device, that would give us an integrated signal without changing its properties, therefore a signal converted from optical to electrical is not what we seek.

The very simplest way of integrating in optical domain would be to use a Fabry-Pérot resonator, that would retain part of the signal, and leave the rest to pass through. By accumulating a certain intensity of field in the resonator, a proportional part of it would leave through the output mirror as

$$I_{out} = TI_R, \tag{1.10}$$

where I_{out} is the output intensity, T is the transmittance of output mirror and I_R is intensity inside the resonator.

More advanced device may be constructed from microring resonators such as this one (figure 1.20), that provide us with temporal integration as well as give us the possibility to directly add signals thanks to the add channel.

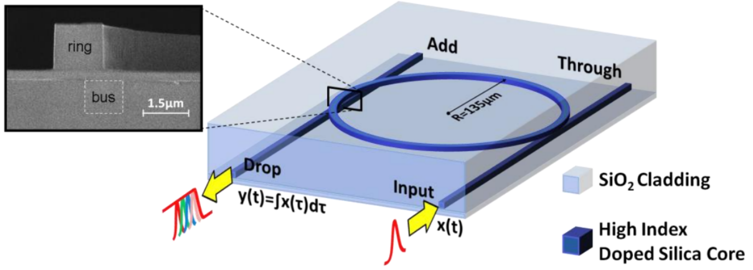

Figure 1.20: Schematic of a microring all optical integrator including and image from electron microscope (without cladding) [16].

We can than construct both first and second order integration as is shown in a figure (figure 1.21). This integrator is not capable of processing speed greater than 400 GHz,

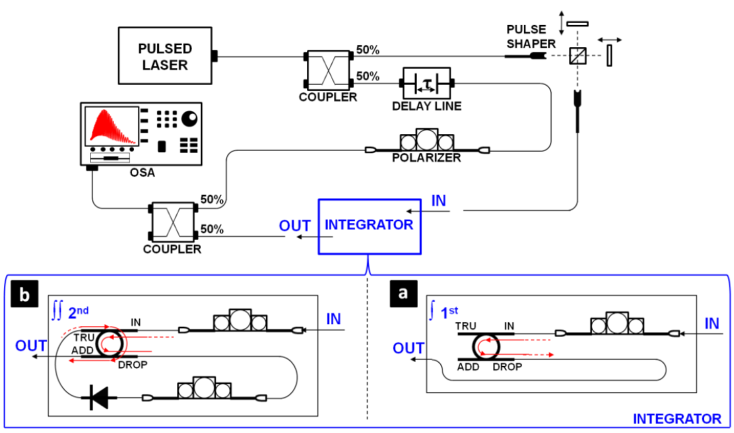

Figure 1.21: Experimental setup for first (a) and second (b) order integration [16].

but also shows, that higher order integration can be achieved without increasing the device complexity and last but least is can be integrated successfully on chip [16].

Chapter 2

Design and analysis of optical computation

In this chapter I am going to cover the design of simple all optical gates which would be incomplete if I did not include design principles. Afterwards I am going to explain why I believe it is better to focus on analogue computing for specialized computation, and why are optical computers more suitable for those applications. In the end of the chapter I am going to present a experimental idea and analyse sources of error expected in the experimental set-up.

2.1 The choice of architecture

But before all that, we need to consider how are we going to achieve our goal of successfully computing using naught but light. In the previous chapter we discussed three possibilities of building logical gates. However now we have to take into consideration manufacturing and building the experiment, and for that we have to be able to produce such devices.

Firstly we must rule out the microring resonator architecture, since to produce these, we would need a facility capable of manufacturing chips on demand. That is the reason why I decided to not pursue this path, which left me with the possibility of utilizing photonic crystals or MZI based architectures.

The former however suffers from the same complications as the aforementioned microrings. Manufacturing photonic crystals can be very difficult, since we require a very precise manufacturing of the 2D photonic crystals as well as a finely tuned laser.

That leaves us the MZI architecture and by extension a free space optics, which we are going to utilize in our experiments. The reason for this decision is mainly the ease of use and build of a free space optics, since we already have the equipment needed. We can also utilize other branches of free space optics, such as Fourier optics and planar coding, that can allows us to do a parallel processing.

2.2 Design principles

In this section I am going to state and discuss design principles, that I used to design my optical computing units.

2.2.1 Primary design principles

The design principles I came up with are three and are quite simple:

1. simplicity,
2. robustness,
3. use of linear optics.

Let me explain those a bit more.

Firstly the principle of simplicity, which can be directly taken as an extension of Occam's razor, is the first and most important of the principles. The reason is, that the simpler the device is going to be, the simpler it is going to be to optimize, replicate, integrate, and use. Since we already are limited by the wavelength of light for the dimensions of the parts, we need to have as little elements as possible, for the chip to be small enough to be effective. Also the replication of more complicated elements can lead to more defective units, and thus increasing the cost of such device.

Secondly, the principle of robustness. For the device to function well in normal conditions (e.g. a desktop computer in home), it needs to be robust. There is a lot of disturbance in such conditions, and therefore we require either a good shielding of the device, or a device not requiring it. We should therefore aim for a device that can give us stable results without needing a lot in terms of stabilization and temperature control.

And lastly, the use of linear optics. This condition is imposed mostly because of the high costs of non-linear elements and a rather finicky requirements on the system as a whole to work ideally. The non-linear elements also usually require quite high intensities that, as stated earlier, can lead to shortening of the lifetime of the device as a whole.

2.2.2 Secondary design principles

In addition to these three main principles I added three sub-principles

1. encoding does not change in the process,
2. wavelength does not change in the process,
3. diminishing of "1" should not happen.

The reason for these three lower principles is to ensure the first two main principles of robustness and simplicity.

The requirement of encoding staying the same on the input as well as the output is a core principle of computing as such. If this principle is broken, the complexity of any device, other than a single unit computers (e.g. single gate), grows and can become unbearable. We can counter it by encoding switches, that would convert any results back to the original encoding, but that is in direct contradiction with the principle of simplicity.

The requirement of signal wavelength to be of constant value is based on the same argument presented for the encoding in the latter paragraph. Again, it can be countered by frequency converters after each operation, but that is again in direct contradiction with the principle of simplicity.

The last requirement of not diminishing "1" is a reasonable for the simple fact, that different lengths of computation, would require different thresholds as well as definitions of "1" and that is not a very good solution. There is of course the possibility of amplifying the signal on the output port back to its original strength, but that may result into false "1" instead of "0" and thus increase the computational errors. And if that is not enough, it is also in direct contradiction with both the principle of simplicity and principle of robustness.

Ultimately I require, that all of the computation happens in the optical part of the device, so that the detection part (that is inevitable for proving that the device works as desired) is only detecting the results. If the detector was part of the computation, the speed of the device would decrease drastically.

2.3 Design of optical gates

In this section I am going to discuss the design of inverter, OR, AND, and a NXOR.

2.3.1 Inverter

To produce inverter in optical domain can be quite a simple feat. If we use encoding into polarisation, we can invert any signal through adding a half-wave plate (HWP) at the angle of 45°, and thus rotating the polarisation between two orthogonal states of horizontal (H) and vertical (V) polarisation (figure 2.1).

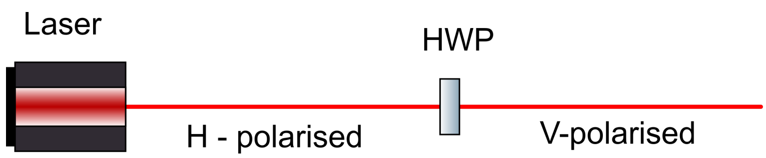

Figure 2.1: Inverter design using HWP and polarization coding.

By encoding "1" as H polarised wave and "0" as V polarised wave (the encoding is arbitrary, and therefore shall serve only as an example), on the output of inverter as this one we get inverted value, we can than detect it by placing polariser in front of a detection unit, that would convert polarisation encoding into intensity encoding, that is much easier to detect.

However using inverter as this one would mean, that all the other gates would have to be polarisation based as well, and to do this the complexity of the gates might get a bit to complicated. For this reason, I decided to stick with intensity encoding, where "1" is represented by "HIGH" level and "0" by "LOW" level of signal.

This decision, however, forced me to utilize an interference scheme for inverter, because that is the only way, how we can subtract intensity of optical signal (figure 2.2).

For this inverter to work properly, we require that the intensity of the idler and the signal is the same, because if it was not, the visibility would get lower, which would result into lower difference of the HIGH and LOW values, which can lead to increase

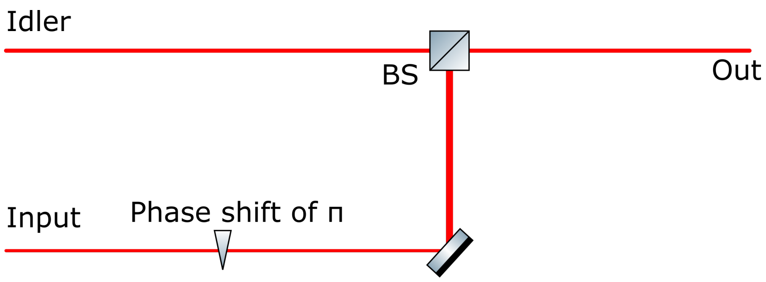

Figure 2.2: Inverter design using MZI architecture and intensity encoding.

of error. We also require well adjusted interference, so that we do not have more than one interference fringe. The principle of the functions is a simple one, by shifting the phase of incoming signal by π, we achieve destructive interference, thus zeroing out the intensity on the output. This provides us that the incoming signal pulse swaps from "1" to "0". If there is no signal, which is equivalent to "0", the idle pulse goes through interrupted, which results in detecting "1" on the output.

2.3.2 OR gate

We can construct OR gate in a very similar fashion to the inverter we discussed in the previous section. There is a minute difference nonetheless. If we construct the gate in this way (figure 2.3), we do not require interference to be part of the computation.

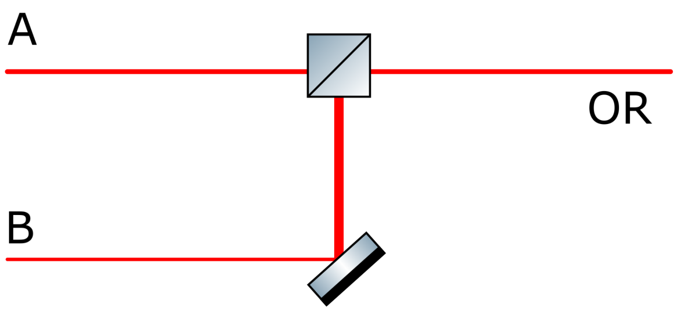

Figure 2.3: OR gate design using MZI-like architecture and intensity encoding.

If either, or both the signals is "1", than we detect a "1" on the output, the only time we get "0" on the output is when the both "A" and "B" is at "0" level. The biggest advantage of this gate, is that the beam-splitter works only as an adder of signals, and since it does not require interference, the stability of the device increases.

We can also construct and OR gate by using lenses or mirrors, whose combination would act as an adder in a similar way as the beam-splitter. However the construction using a single beam-splitter is the simplest one I came up with.

2.3.3 AND gate

The construction of an AND gate may be a bit more complicated. There are possible solutions, using either idle signals or a non-linear optics. One of these we already discussed in previous chapter (figure 1.13), however this gate does not meet the conditions I imposed on the designs in form of the design principles.

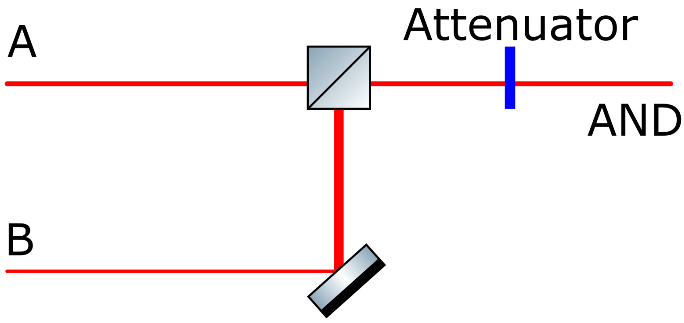

Figure 2.4: AND gate design using MZI-like architecture and intensity encoding.

For a simpler solution, I went for another MZI-like architecture (figure 2.4).

This gate utilizes attenuation of signal in a way, that if either "A" or "B" is "1", than on the output we can detect a signal, however this signal is below the threshold, which then results into "0". If both the signals are "1", the attenuation is not enough to bring the signal below the threshold, so the detection unit detects a "HIGH" level of signal which corresponds with "1".

The problem here is to attenuate the signal well bellow the threshold, so we do not get any false HIGHs, while maintaining the signal as close to the original HIGH level of either "A" or "B", so that we do not diminishing of "1" in the process of computation. The requirements on the precision of maintaining the level of "1" may differ based on the application, but for our purpose we ideally want that output "1" and input "1" are of same detection values.

2.3.4 XNOR gate

This gate is based on the MZI architecture as are the others. The function NXOR is otherwise known as equivalence, and therefore what we are building is a optical comparator for digital signals. To build such a device we require an idle pulse and two inputs "A" and "B" (figure 2.5).

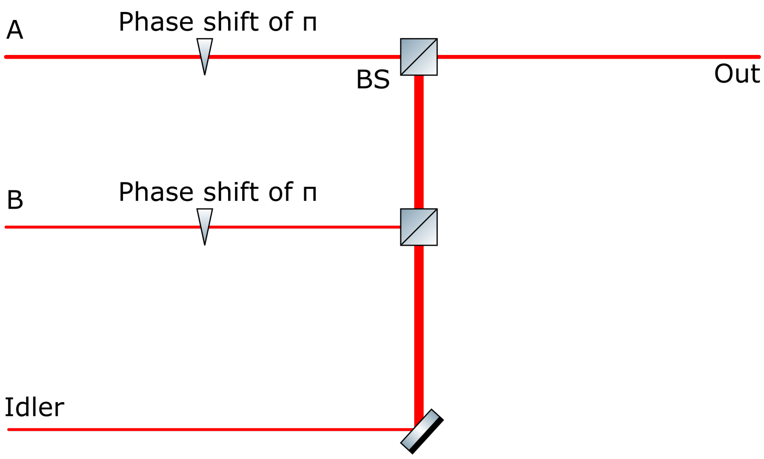

Figure 2.5: NXOR design using MZI architecture and intensity encoding.

In the case of either "A" or "B" being "1", a destructive interference occurs, providing us a "0" on the output. In the case of both "A" and "B" having the same value of either "0" or "1", the interferometer becomes unbalanced, and the output acquires an intensity, that is equal to the "1". This is a result of either, no signal, so there is no wave, that can be subtracted from the idle wave, or having two signals, that after the first interference, the idle signal and "B" signal is subtracted from each other, resulting in "LOW" level, which allows signal "A" pass through the gate unchanged so that we get "1" on the output.

This gate is rather complicated, since it requires synchronization of three pulses and two properly adjusted interference. This may prove a complication resulting in breakdowns, due to disturbances from the environment.

2.3.5 Free-space optics advantages

The free-space optics, while looking a bit impractical from the point of integration and applications, it allows us some other possibilities. The very first and quite a direct extension, the free-space gives is is the ability to expand the signal into more than just a point.

By expanding the signal to a line, we can than use a intensity modulator to encode the information into predefined parts of the line. By doing so, we achieve a parallel computing. This style of parallel computing is unique to free-space optics, because it allows us to process information in a parallel fashion, while using a single device. In electronics, the parallel processing requires a new gate for each bit in the byte.

This expansion, unfortunately, is not free of charge, because to encode information into segments of a line, we than need a detector capable of distinguishing those segments. This may be done in multiple ways, probably the simplest of them is using multiple photodiodes connected in a parallel fashion such a is shown in a figure (figure 2.6). The recovery of information is done by measuring the current going through each

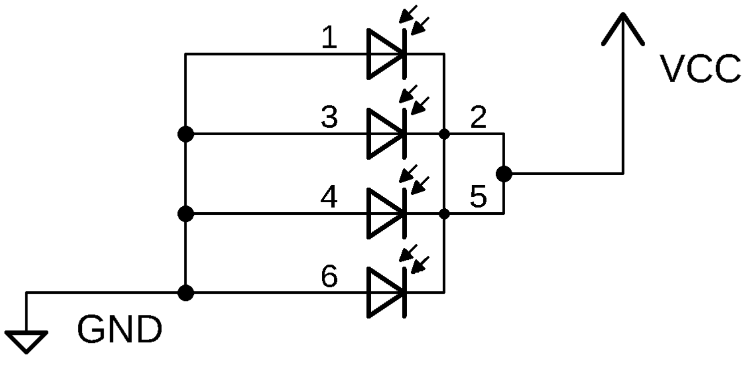

Figure 2.6: Simple 4 bit line detector.

of the channels, if the current measures as "HIGH", we know that the incoming field has also a "HIGH" intensity, thus giving us "1". By measuring a "LOW" level of current, the diode is not open and thus no (or very little) light is hitting the photodiode, which gives us a "0".

This solution of detection is rather crude. thankfully we have much more sophisticated solution, and that is using matrix detectors such as charge-coupled devices (CCD) or complementary metal-oxide-semiconductor (CMOS) detectors. These are commonly found in cameras and allow us not only discerning bit in line, but in a plane. That allows us to increase the dimensions of our expansion of field from 1D line into 2D plane, which we than divide into single bit areas, which we than process through our system and retrieve information from the signal by reading out specific pixels, or bunches of pixel, depending of the fineness of the divisions.

We are limited not only by the physical dimensions of a pixel but also by diffractive limit. Let us firstly focus on the first limit, the pixel dimensions. the reason for this is fairly simple, if we have two bits covering the area of a single pixel, these two bits become indiscernible, and thus we cannot retrieve the encoded information. Therefore, the limit of the size of the pixel.

The diffractive limit comes from the two hole experiment, performed by Thomas Young. If the bits are too small and a situation occurs, where there are two bits, that let the light pass through and an area of dark bits dividing them, these two bits may than behave as point two sources of light that can than interfere with each other. The purpose of this thesis is not to optimize this effect, so I am going to leave the subject here, it is nevertheless important to consider this effect, when optimizing the density of bits in the plane.

Concerning the role of the detector

In this section, I delved into the detection, and it would be good to discuss here the role the detector plays, and the restriction it poses on the system.

Firstly, the role. Detection in my designs, as well as the experiment, only serves for information retrieval and verification of the correctness of the computation. This is of vital need, since we cannot use purely optical set-up due to the fact, that fully optical computers are not available to us, they would however allow us to skip the detection part. Nonetheless the computation itself is purely optical and thus unhindered by the limitations of electronics.

To consider the restrictions of the detection, we have to consider the nature of detection, the speed of the detector and its sensitivity. The sensitivity limits the intensity of the signal we are still able to detect, but we want to use as low power as possible, since it is efficient in both the power consumption aspect, and the aspect of device lifetime. The requirement is for the signal to be of much higher intensity the the noise-equivalent power of the detector.

The speed of the detection limits how fast can we retrieve the data from the computation, but does not limit the computation speed itself! Again, we are encountering the limits of electronics we need to check the results of the computation.

Lastly, the nature of the detection. To be able to decode the information encoded in the signal, we need a proper detection set-up. For example we cannot use a simple PIN diode if we use other than intensity encoding. But since the detection is only for checking the results, using inadequate detection set-up would result only in defective information retrieval, not in error of the computation itself.

2.4 Integrating in the optical domain

In the previous chapter, we discussed the possibilities of integration of optical signal. The first one, was using a CCD that would output electrical signal based of the accumulated charge from the incident light, for this, we would require an uncapped CCD with non-destructive continuous read-out. By uncapped, I mean a CCD that it cannot become saturated in the course of the computation process, otherwise, the computation would not be correct. The requirement of non-destructive read-out is a complicated one, since the CCD detectors work as a one time valve, that converts all the accumulated charge into voltage, that is then read by the electronics, this would not work, if we wanted a time dependency of the integrated signal, we either would need to sample that by incrementing the acquisition time and afterwards recreating the function, which is a rather cumbersome way of computing. The requirement of continuous read-out is simply an extension to the non-destructive read-out. This way of integrating is not really a step forward, and we can as easily use an electronic analogue computer.

By using the resonator integration, we get a true optical integration, that we can work with in the optical domain. However using an external Fabry-Pérot resonator in free-space optics may not be as simple as trying some different approach. This brings us back to the previous section, where we discussed the possibilities of expanding the beam and using it for a parallel computation of matrix arranged bits. We have also have the possibility of encoding a 2D function into the intensity of the field, which may allow us a unique possibility of spatial integration, that is not possible in any other analogue computer we know.

To build such a spatial integrator, we would require a device that would allow us to add all the intensity distributed across the incoming field into a single point in the output plane. Thankfully we have a optical element, that is ideal for such use, and that is biconvex lens, that focus an incoming planar wave into the focus point. The principle can be shown by the principles of geometrical optics (figure 2.7).

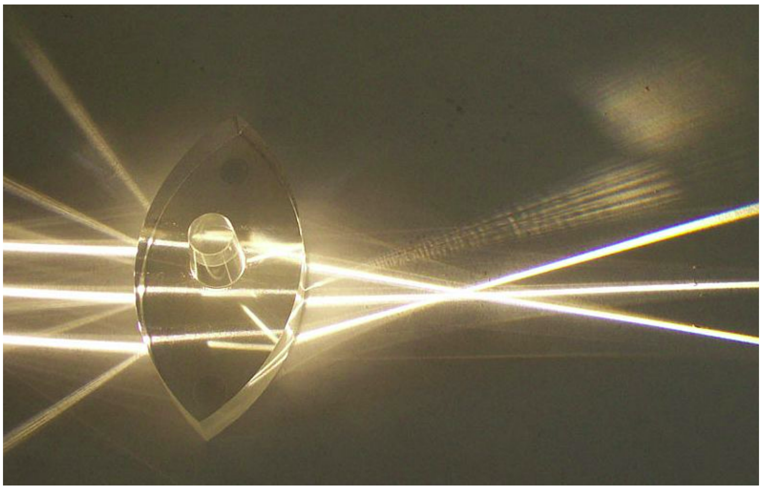

Figure 2.7: Visualization of biconvex lens function [17].

We can see, that the intensity I at the focus point is

$$I = \sum_i I_i, \tag{2.1}$$

where I_i is the intensity of a single beam. Using this approach, would be very similar to numerical approaches, where we sample the function of interest by single beams, and their intensity. However if we generalize this approach and use wave optics, we do not get a set of beams on the input, but a plane wave, that gets transformed by the biconvex lens to convergent spherical wave, converging in the focus point. The intensity at focus point I would then be

$$I = \iint i(x,y)dxdy, \tag{2.2}$$

where $i(x,y)$ is the function describing the intensity distribution of the input plane wave. And that is the operation we were looking for.

Let us now look at the design of the integrator. As I already stated, the very simplest design can be achieve by using a simple biconvex lens, but that may be a little unwieldy, since we would not get any further possibilities of manipulation. For this reason, the design of an all optical integrator, is a bit more complicated, using a focusing screen to add uniformity to the results and an 4f system for focusing on the detector (figure 2.8). By changing the position of the second lens, we are able to change

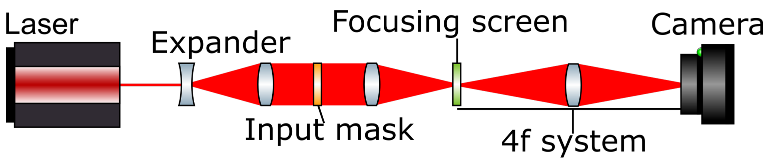

Figure 2.8: Integrator scheme utilizing 4f system for better focusing on the detector.

the size of the spot we then detect. For the lens being twice the focus distance from the focusing screen, the spot on the camera is going to have the same size. Depending on the type of the detector the requirements for the spot size may differ. If we take any single unit detector such as PIN diode based power meters we want the entire spot focused the the active area of the detector, so we collect maximum possible amount of incoming light.

For matrix detectors such as CCD or CMOS cameras, the spot size is a more complicated to discuss. We still want a relatively small spot to be able to utilize other parts of the detector for possible parallel computing. If we want to maximize the amount of parallel data streams, we want the spot to be no larger than 1 pixel. However if we may want defocus the system a little and thus enlarging the spot size for the purpose of compensating the spatial inhomogeneous intensity distribution of the input field as well as a possibility for countering the saturation of the pixels.

It is important to note, that even though we are detecting the results with a camera, the actual process of integrating is all optical and the detector's function is only for us to measure results if we require so, and is thus independent of the process. The biggest advantage of using a camera for detection is that it allows us to compensate for inhomogeneities in the field intensity distribution, by dividing the signal and the field shots as well as offers possible extensions for parallel operations. It also gives us the possibility to directly observe the result and check their correctness.

The disadvantage of having a camera in the chain is that it is rather slow way of detecting the signal. Nevertheless we do not need to be concerned about it, since it is not an active component in the computation.

2.4.1 Preparation of inputs

The preparation of the input function, that we want to integrate can be done in multiple ways. The easiest one is using a printed representation of the function, another way is using a spatial intensity modulator and the last possibility is studying phenomena directly, given that the phenomena are the source of the intensity modulation. Let us look into these possibilities in more detail.

Firstly, let us consider the last possibility. Using some kind of physical phenomenon, that we do not know the precise distribution, it possesses, is not a good approach to testing a new device. However, due to the very high speed of integration, we may be able to analyse these phenomena at a later point in time. For now, this is not useful.

Using a spatial intensity modulator is another possibility. Such a modulator can be fashioned from an LCD, by taking of the backside, providing light, and using only the transparent liquid crystal part of it. By applying voltage to pixels, we can change the transmission coefficient and by doing so change the intensity of output intensity. Using an optoelectronic spatial modulator is the preferred method of encoding information, since it gives us superior control over the the system, however if we lack such a device, we can use a static masks.

The last and simplest method is using any printer and a see-through foil. By doing so, we can easily get a good quality masks, that provide us with a function of interest, that we want to integrate. Mask that is produce in this way can look like this (figure 2.9).

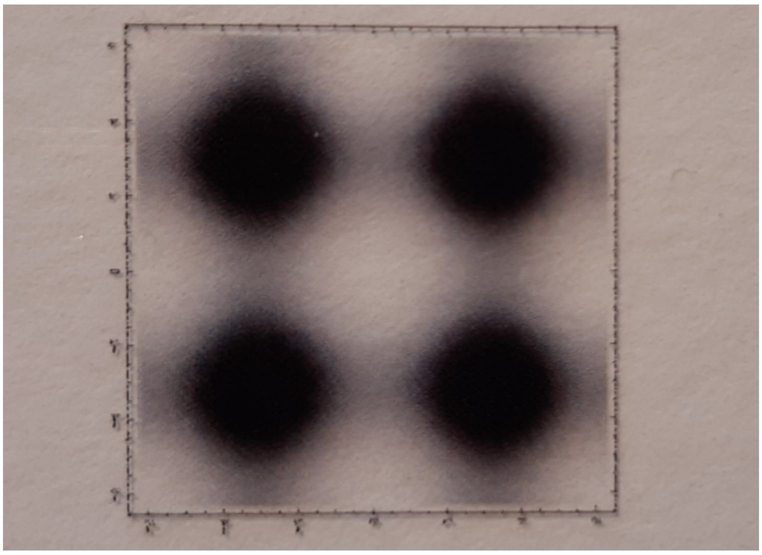

Figure 2.9: An example of an input mask printed on see-through foil using inkjet printer.

The printed masks may suffer from low density of print, that would manifest in a form of low attenuation of the signal in the black parts of the mask. Thankfully, we can counter the influence, by printing different saturations of black, by which, we can compensate for non-zero transmission of black regions. The resolution of such mask is directly proportional to the dots per inch (dpi) of the printer used.

Printed masks have another property, the LCD based modulators lack. The modulation is non-polarising, allowing us to use a non-polarised signal. This gives us more options in choosing the light source, by posing essentially non-existent requirements on it. Only one requirement remains and that is the requirement of absorption of chosen wavelength.

2.4.2 Solving differential equations using the integrator

We can use the integrator to solve differential equations in form of

$$\frac{\partial^2 f(x,y)}{\partial x \partial y} = ag(x,y) + bh(x,y), \tag{2.3}$$

where $f(x,y)$ is the solution we seek, and $g(x,y)$ and $h(x,y)$ are inputs functions. To provide the solution, we would require the scheme to be slightly altered (figure 2.10).

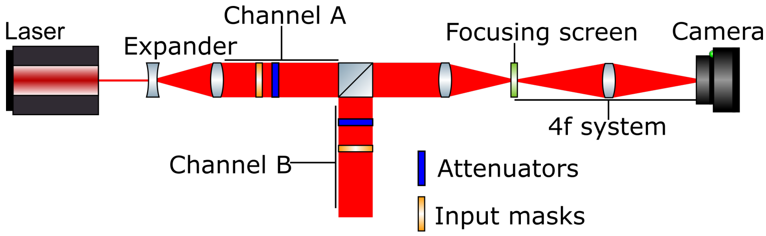

Figure 2.10: Scheme of solver, capable of solving equation (2.3).

We can even expand the scheme to solve a sum of more inputs, just by repeating the process of adding more inputs in the same fashion as input b (vertical input (figure 2.10).

We can also solve any product of multiple functions by adding the input masks to series.

The solution of equation (2.3) can be done by direct integration

$$f(c,d) = \iint_{[x,y]=[a,b]}^{[c,d]} ag(x,y) + bh(x,y)dxdy, \tag{2.4}$$

where $[a,b]$ and $[c,d]$ are coordinates in the 2D plane. We get the solution using the scheme (figure 2.10) by changing the coordinates $[c,d]$ on the modulator.

The 4f system's function in the computation is to allow us manipulation of the integrated signal and through that use of the signal for further manipulation or detection of the results.

The solution of a differential equation depends on its initial conditions, these are directly encoded in the modulation as the boundaries of the modulator.

The proposed scheme does not require a matrix detector, however the matrix detector allows us to copy the scheme into a 2D array of those, thus allowing us to utilize the entire area of the detector. This however would be quite a feat of engineering. A much simpler simpler solution is to divide the input mask into areas representing the functions and using a fly-eye lens array. By doing so, we can substantially increase the computing power of the device. For example, if we have a 15 MPx sensor and dedicate to every result an area of 9 Px (3 × 3 Px) with the detection speed of 0.01 s we get $1.667 \cdot 10^8$ results per second. Keep in mind, that the results are not of logical computation, but of analogue computation, to compare the speed to classical computer, we need to know the range of analogue-digital converter (ADC). For a 16 bit (ADC) we would get the processing power of 2.667 Gbit/s. By using a high-speed camera and better space management, that would allow us to use less pixels per result, we can increase the the computing power of equivalent CPU up to the THz levels.

Chapter 3

Experimental set-up and results

Previous two chapters covered both the theoretical and the design part of this work. This chapter follows the subject of spatial optical integration from the point of setting up the experiment, that is testing the design of the integrator, analysing the set-up and deducing the expected error of measurement and in the end covers the results of the experiment I conducted. But before we start, we should look at one other thing, that can be quite vital for any computation, and that is the protocol of coding and decoding the information.

3.1 Computing protocol

For successful encoding and decoding of information, we need a set reference point. In binary computing, this can be done, by specifying a threshold, but in analogue computing, this can be a bit more complicated. For this experiment, we are going to use the range of values of [0,1], where the value of "1" is equal to the intensity of empty input (empty input is an input without any mask) and "0" is equal to a covered input, so no light gets to the detector.

After setting our reference point, we can approach the integration of any input function, by inserting the mask and detect the intensity. We then divide the reference measurement and the integration as,

$$I = \frac{M}{F}, \tag{3.1}$$

where I is the numerical result, M is the absolute value with mask in the input and F is an empty field measurement in absolute value, by which we receive normalized numerical value of the integration, which can be written as

$$I = \frac{\int\limits_a^b f(x)dx}{\int\limits_a^b \max(f(x))}. \tag{3.2}$$

3.2 Experimental set-up

For the experiment, that can test the function of the integrator, we require a source of light, a focusing screen, a beam expander, in the case of the beam being to narrow for the purpose of encoding the information, input masks, the integrator and a camera. Scheme of the set-up can be seen in the figure (figure 2.8). The measurement is going to be a direct measurement of intensity, using a CCD matrix detector.

3.3 Analysis of the error based on the influence of experimental elements

The elements we need to consider are lenses, a focusing screen, a input mask, a light source, and a camera. For the purpose of this analysis, I am going to divide these into two categories, the passive elements and the active elements.

3.3.1 Passive elements

The passive elements are all but the light source and the camera. Which leaves us with the lenses, input masks and the focusing screen.

Due to the principle of integrating the intensity across the field into a point, it would seem, that we are going to be limited by the quality of the lenses, and require them to have no aberrations, but in truth, the integration is independent on them, since the "amount of light" in the field is not changed by those, just redistributed. And since our computing protocol is based on the division of the integration by an empty field, these are going to be compensated through it.

The focusing screen can only add a constant offset, in the form of attenuation of the signal, but since the same would be true for the empty field, this is compensated by our protocol as well.

Lastly, the masks, which I decided to print on an inkjet printer. These may add a positive offset to the result of the integration, since the black, representing the value of "0" may not be saturated enough, and thus may be leaking light. This can be compensated by preparing different saturation masks, which we can than measure an by a method of linear regression determine the true "0" value, and afterwards compensate the equation (3.1) by deducing this value as

$$I = \frac{M - aF}{(1 - a)F}, \tag{3.3}$$

where a is a number in range [0,1], acquired from measurement of the absorption of the print.

As we can see, with good set-up and a few simple compensations (some of which are inherent in our computing protocol) we can make the effect of the passive elements negligible.

3.3.2 Active elements

There are two active elements, the light source, and the camera.

Light source

Firstly let us look at the influence the light source have. Since we are using an intensity encoding, the only influence on the stability of our measurement is going to be the stability in intensity. We are going to be using a laser module, that is constructed to provide a stable intensity with refresh rate in the range of GHz, so as long as we keep our detection frequency well below the refresh rate, we are going to be integrating across a wide set of pulses, thus reducing the noise towards Gaussian distribution. This source of noise is usually very low compared to the noises, we encounter in the detection units, that has electronic source.

Noises of the camera

Before starting the analysis, it is good to mention, that the character of added noise by the camera is different from the rest of previously discussed noises. The difference is in the fact, that the now we are talking about retrieving the results of the computation, not the computation itself. For this reason, the added noise by the camera does not influence the computation itself, because it has no part in the computation.

The camera is suffering from more noise sources, than all of the other parts combined. There is dark current, image noises, readout noise, amplification noise and analogue-digital converter (ADC) noise.

Before we analyse these, we should get acquainted with the architecture of a CMOS detector and the working principle (the function of CCD detector is similar, but I believe, that CMOS are more conceivable). A CMOS detector may be constructed as it is shown in the figure (figure 3.1).

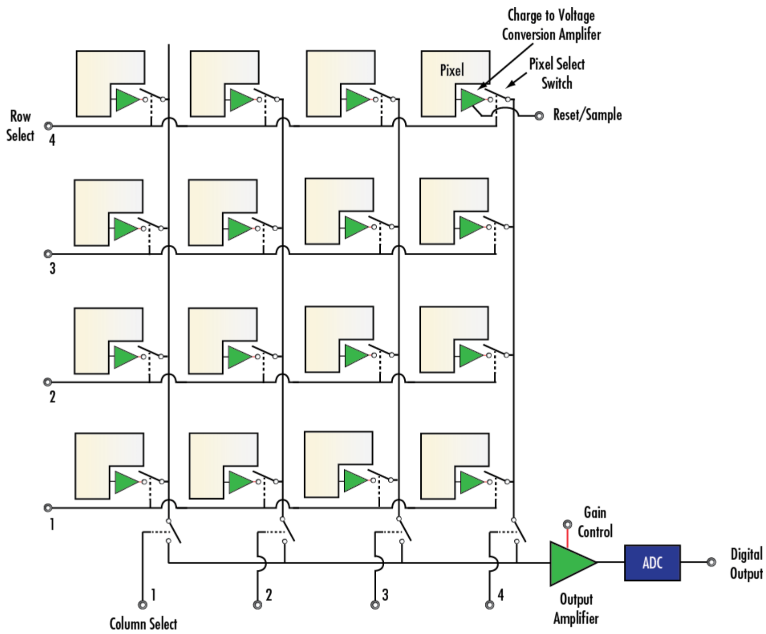

Figure 3.1: An example of a CMOS detector [18].

The incident light creates free carriers in the pixel, which creates charge, this is than converted to voltage and send to the output amplifier, which is afterwards converted from analogue to digital signal and which is then send to a computer.

An image, that is gained this way is obviously suffering from the imaging noise, that is caused by the statistical character of detection, readout noise due to the electronics, an ADC addition to it, a thermal noise, caused by the non-zero temperature of the chip and a vignetting, that is caused by the optical system used for focusing the signal.

An ideal detector, that has been covered, so that no light reaches the chip, would give us a reading of zero. A real detector, in the same scenario, would not, This is caused by thermal noise of the silicone chip, which is called dark current and is caused by electron generation within pixels. This can take place in any point within the pixel. While being caused by the temperature of the detection unit we can counter it effectively by cooling it down using either passive methods, such as liquid nitrogen, or passive radiators, or actively using thermoelectric coolers such as Peltier cells. If we are able to maintain stable temperature throughout the measurement, dark current maintains constant mean value. If we are not able to reduce dark current below the detectable levels, which would be if the level of the variance of dark current was significantly lower than the lowest output bit (output will be discussed in a short time), dark current gives us the lowest possible noise level we will detect thus limiting the precision of our measurement and computation.

If we let light hit our detector, we will unavoidably encounter a new, higher levels of variance, than they should, based on the dark current. This phenomena is called image noise. While we can describe multiple types of image noise, such as film grain, or salt-and-pepper noise. Most of them originate in charge-voltage conversion, analogue-digital conversion, or transmission errors, however two some are tied directly to image capturing. These are periodic noise and shot noise. The former can be caused by an periodic electrical interference during the image capturing. This can manifest as it can be seen in the figure (figure 3.2). Periodic noise can be diminished by using notch filters in the frequency domain. The shot noise, on the other hand is a fundamental noise, we

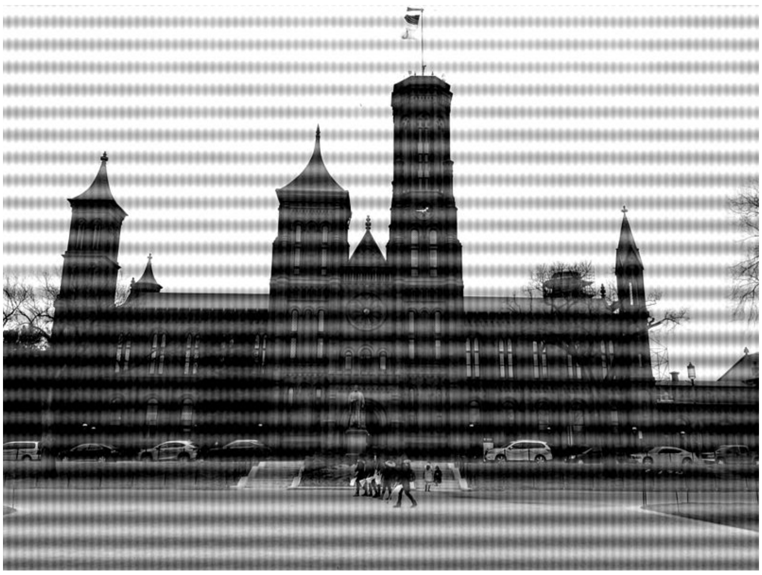

Figure 3.2: Periodic noise example [19].

cannot reduce effectively, because of its nature. It is caused by the discrete character of both light and electrical charge and the statistical character of light detection. Since we know that shot noise follow the Poisson probability distribution, we can assume, that images taken in series will follow the same distribution on each single pixel. Since this follows the Poisson distribution, we can also quantify the signal to noise ratio

$$SNR_{shot} = \sqrt{N_{in}}, \tag{3.4}$$

where N_{in} is the number of incident photons.

Lastly we need to discus the read-out noise. This is a noise of the electronics. I comprises of three noise sources:

1. Charge-Voltage converter and amplifier,
2. Output amplifier,
3. Analogue-digital converter.

If all of the aforementioned elements worked ideally, none of them would add any noise to the signal, however both the dark current, and the image noise would be amplified accordingly.

Lastly let us discuss the influence of vignetting. Vignetting has multiple causes, but in all cases, it causes an intensity drop towards the edges of the detector. Mechanical vignetting is caused by blocking of the incoming light by the mechanical elements of camera lens. This we can easily get rid of, and therefore we can neglect its effect. Then there is optical vignetting, caused by shielding of optical components by precedent optical elements in the camera lens. This effect can be diminished by stopping of the lens through increasing the f-number.

The last, and not diminishable, is the natural vignetting, which is caused by the geometry of the imaging. We can calculate this effect by using the $\cos^4$ law. However in our case we are going to aim for a small area. For example with the area of $\approx 10 \times 10$ px, with one pixel being 6.8×6.8µm, we can expect the decrease in intensity due to natural vignetting approximately 0.007% for the focal length of 5 mm. By increasing the focal length, the influence of natural vignetting diminishes.

3.4 Used equipment

Let us start with the detection part of the set-up. I used an Apogee using KAF-3200ME chip and camera lens (figure 3.3(a)), that was controlled with SharpCap software (figure 3.3(b)) and a power supply (figure 3.3(c)).

The computation part of the set-up consists of laser module (figure 3.4(a)), a beam expander for which I used microscope lens (figure 3.4(b)), gray filters limiting the intensity of the beam (figure 3.4(c)), an input mask (figure 3.4(d)), and the integration lens, for which I used another microscope lens, and a focusing screen (figure 3.4(e)).

Now, let us have a closer look at the components and their use.

The laser module gives us a beam of stable intensity and with very well defined spatial properties. This beam then passes through a microscope lens, than works as a beam expander (figure 3.5), so we can encode information into the spatial distribution of intensity.

After expanding the beam, it passes through gray filters (figure 3.6). These are not necessary, but counters the high intensity, that comes from the laser module, which would otherwise saturate our detector. If we have a light source with intensity, that does not saturate the detector, we can omit these.

This expanded beam then hits the printed mask (figure 2.9), which modulates the function of interest onto the intensity.

Afterwards, the beam goes through another microscope lens, that works as an integrator (figure 3.7), that focuses our beam to the focusing screen (figure 3.8).

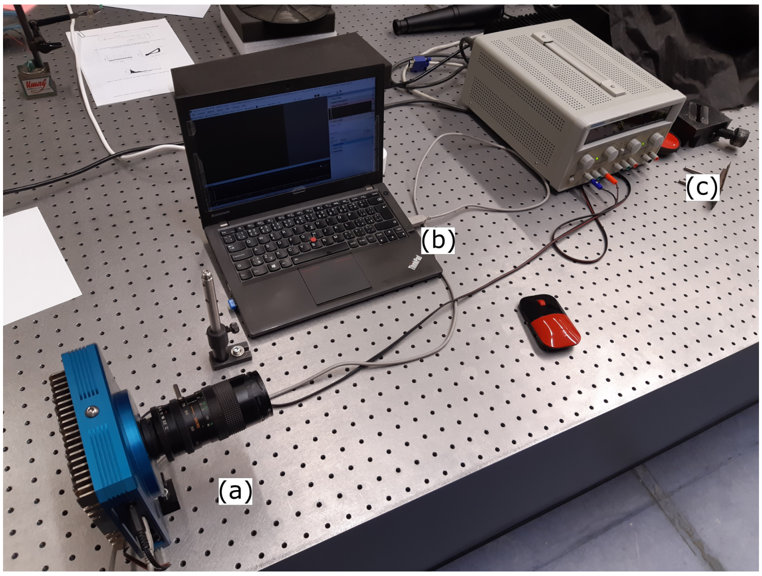

Figure 3.3: Detection part of the experimental set-up (a) Apogee camera with camera lens, (b) laptop using SharpCap for controling the camera, (c) power supply.

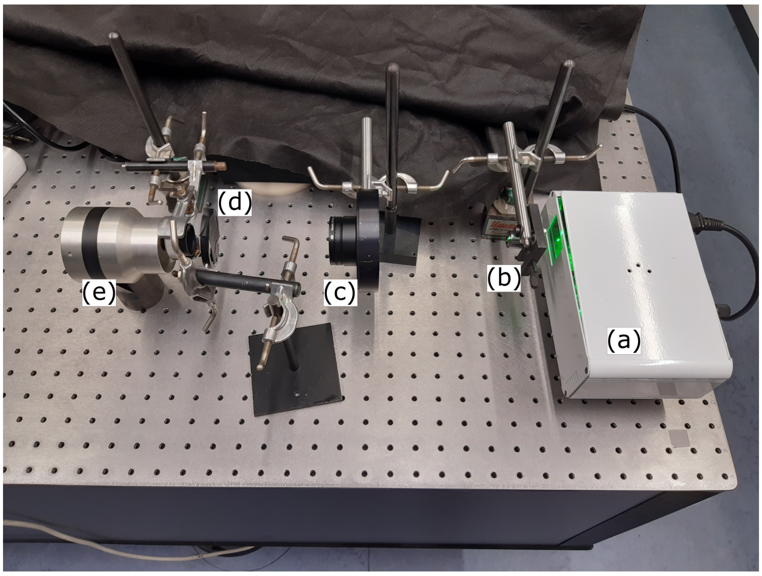

Figure 3.4: Computation part of the experimental set-up (a) Laser module, (b) microscope lens used as an integrator, (c) gray filters used for attenuation of the beam, (d) input mask, (e) microscope lens used as an integration lens and focusing screen.

This focused point, is then focused on the detector through camera lens (figure 3.4), that act as the 4f system and refocuses our result onto the detecting chip. The camera I used, was equipped with a KAF-3200ME chip, that guarantees us a linear detection. The camera is also equipped with thermal controls, which allows us better oversight over the experimental conditions and lowers the dark current of the detector.

The question, whether we really require a laser module as a light source arises. To answer it we decided to swap the laser module for a set-up that would give us a wide beam, so that we would not need a beam expander, and allow us to modulate the input data using optoelectronic modulator. Such a device can be obtained by using a data projector (figure 3.9).

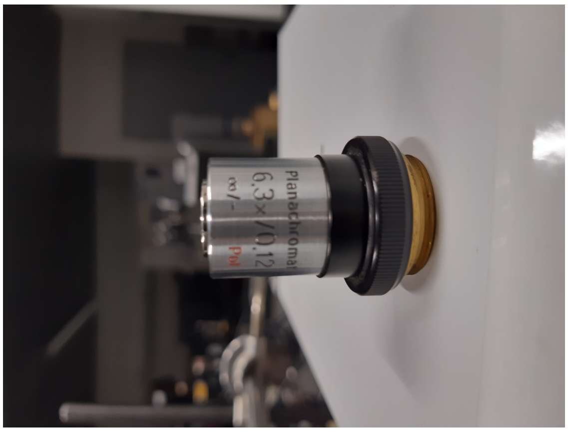

Figure 3.5: Microscope lens I used as a beam expander.

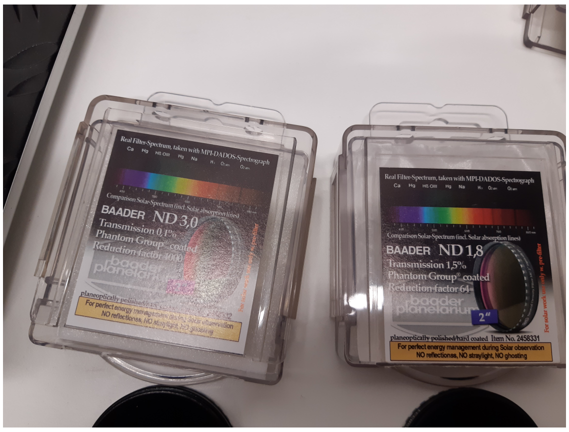

Figure 3.6: Gray filters used for attenuation of the beam for control over saturation.

Figure 3.7: Microscope lens I used as an integrator.

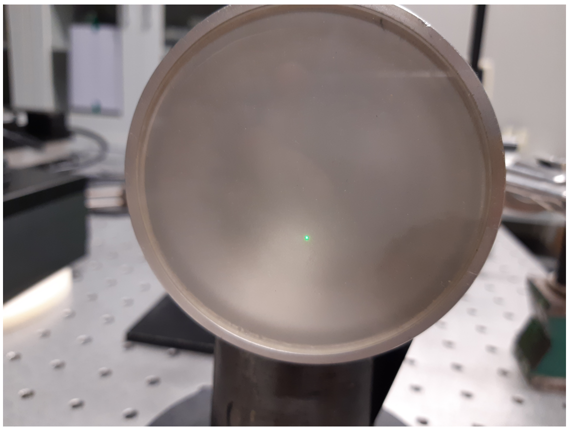

Figure 3.8: Focusing screen used for homogenization of the integration.

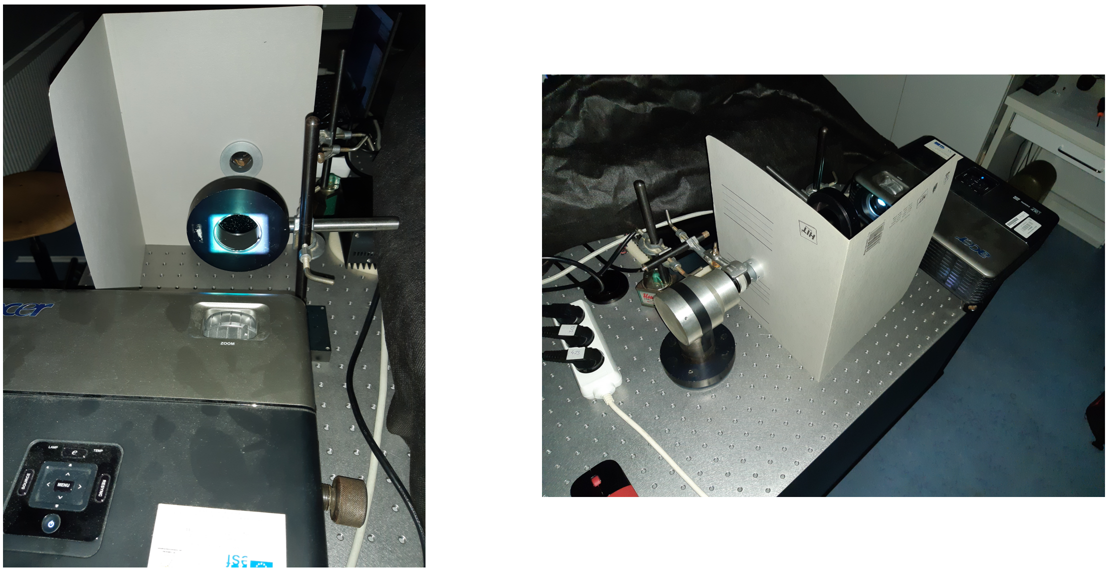

Figure 3.9: Experimental set-up using data projector as a light source and a modulator with a pupil, limiting the surrounding field.

3.5 Error analysis, based on the equipment

As stated in the previous noise analysis, we have many sources of noise that come to our measurement, however, these add together, and what we detect is a sum of all the mentioned parts. In the previous section, where I was analysing the influence of vignetting, all the parameters were taken from the datasheet of the sensor. Therefore the expected error of 0.007% for the focal length of 5 mm due to natural vignetting is accurate prediction for our measurement.

We can also reduce the noise caused by the dark current not only by cooling the camera, but also by subtracting a dark frame from our measured signal.

Nevertheless, we are guaranteed detection inaccuracy in linearity from 1% to 2%, and inaccuracy in uniformity from 1% to 5% by the manufacturer of the chip [20]. These two inaccuracies add up, an we cannot compensate them, as we can do with vignetting. To these construction limits of the camera we have to add all the noise sources, we discussed in section 3.3. Knowing this we expect a slight increase to the uncertainties, therefore we can expect lowest measurement uncertainty of $\approx 2\%$, but it can go up to $\approx 5\%$.

3.6 Measurement procedure

3.6.1 Measurement set-up preparation

For measuring the integration of the input function we need a CCD or CMOS detector, input masks in printed form or planar intensity modulator, beam expander, attenuator in form of either grey filters or polariser (based on the nature of the light source), a laser module and the integrator consisting of a convex lens, a focusing screen, and a 4-f system utilizing either a convex lens or a camera lens.

We require the detector to have as linear response to intensity as possible. If the detection unit does not have a linear response, than we need to address its response function while recovering the solution.

The input masks should be printed in highest possible quality on a transparent foil, which itself would introduce as little variations in the in the output intensity. We should also measure the consistency of the print in the area as well as the decrease in intensity based on the hue of the print.

The light source should provide us with a stable intensity. That is why we decided to use a laser module, since it can provide us with a well controlled intensity and a easy to use interface.

Before measurement, we need to thermally stabilise the chip so we have a constant thermal noise. This can be done either by leaving the detector running in the measurement environment, or by using electronic thermal stabilization. If we have a thermal stabilization on the chip, we want to cool the chip down, thus reducing the thermal noise.

3.6.2 Measurement execution

After building the integrating set-up we first need to address the saturation of the chip. We do this by setting the detection time to a short enough exposition time. We need to make sure, that the measured intensity does not exceed two thirds of the range, since we can expect a big increase in the deviation from linearity.

After setting the exposition time, we need to focus our camera lens, so we have a well defined bright spot close to the middle of the chip. We want the spot to be no larger than $\approx 10 \times 10$ px. The detector chip KAF-3200ME has ≈ 3.2 MPx, allowing us ≈ 3200 parallel operations for the specified spot size.

Now we can proceed to getting our field shot, which we will use to normalize our measured inputs. With the exposition time and proper beam focusing we take a shot, followed by a dark frame, which we subtract from the field shot. We thus obtain our normalized field shot. To increase the accuracy of the measurement, we should take multiple shots (I used 50 shots), and use the average of those, as our reference field shot F.

Afterwards we put in the input mask, and capture the intensity profile the same way, as we did with the empty input. By following the subtraction of the dark frame we get the signal shot S.

In the case of imperfect black on the inputs, caused by the printer, we should add the measurement of the black minimum, which we can do by inserting series of printed grey filters, and measuring the intensity the same way as with function inputs.

3.6.3 Data processing

Firstly we need to compensate for the inhomogeneity of the light field for each input. We can do this simply by using the flat-field correction as

$$I = \frac{S}{F}, \tag{3.5}$$

where S is our signal shot and F is our reference field shot.

In the next step we choose the area of bright pixels where the decrease in intensity of the border should not be more than 10%. If we took the entire detected spot, we would largely decrease the accuracy of our measurement by including the image expansion caused by the diffraction due to the physical dimensions of entrance pupil.

Thereafter we calculate the arithmetic mean of the normalized intensity as

$$I_{out} = \frac{\Sigma I_{xy}}{N}, \tag{3.6}$$

where I_{xy} is the detected intensity in the pixel whose coordinates are $[x,y]$ (the actual values of the coordinates are arbitrary), and N is number of the pixels.

Then we calculate the standard deviation as

$$s = \sqrt{\frac{\Sigma\left(I_{xy} - I_{out}\right)^2}{N-1}}, \tag{3.7}$$

and analyse the data. If any of the pixel's deviations are larger than $3s$, we consider it wrong, and exclude it from our data ensemble.

The last step of our measurement is comparing the measured intensity (which will have value between 0 and 1) and calculation of the same function, as the input, done in Mathematica. If the values match, than our optical integrator works correctly.

If we observe a systematic shift of our values towards higher than expected values, we need to employ the correction of the printed black (described in the last step of measurement) and recalculate the range as

$$I = \frac{S - aF}{(1-a)F}, \tag{3.8}$$

where a is our black correction coefficient.

3.7 Results

In this section, I am going to present the results of the integration, using the integrator I built. Before I am going to approach the functions I integrated, I need to perform the correction on black, because, the prints were not close to blocking all of the light, which was observable by naked eye.

3.7.1 Correction on black

For the purpose of this measurement we need to print improvised gray filters, with progressing saturation of black. I decided to start at 10% black saturation and increase it by 10% up to 50%. I also added a 70% black saturation (figure 3.10) for more data-points closer to the 100% saturation of black, that would still give us high-enough signal.

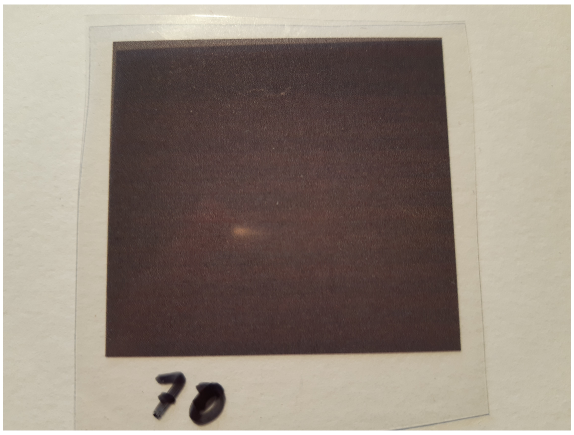

Figure 3.10: Printed 70% black saturation filter.

If I wanted to obtain closer data-point to 100% we might encounter that the intensity of the signal would be too low to distinguish from the background.

By utilizing the method described earlier, I obtained the relative intensity, which I then plotted in a graph, and fitted by linear function (figure 3.11).

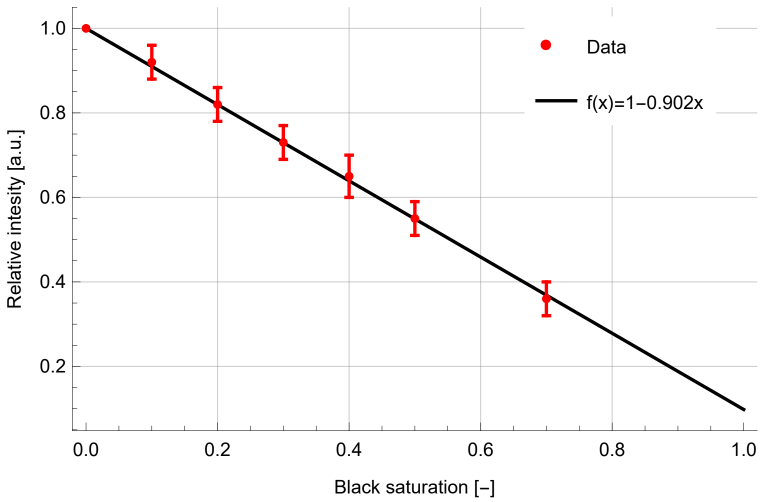

Figure 3.11: Graph of relative intensity versus black saturation including linear fit.

By using linear regression, we now know, that the "0" value is at 9.8% above the true "0". This gives us the black correction coefficient $a = 0.098$.

Since we wanted to use the projector as a modulator, we also needed to measure the intensity of light based on the black saturation. I did so in the same way, as in the previous measurement and fitted the results with exponential function (figure 3.12).

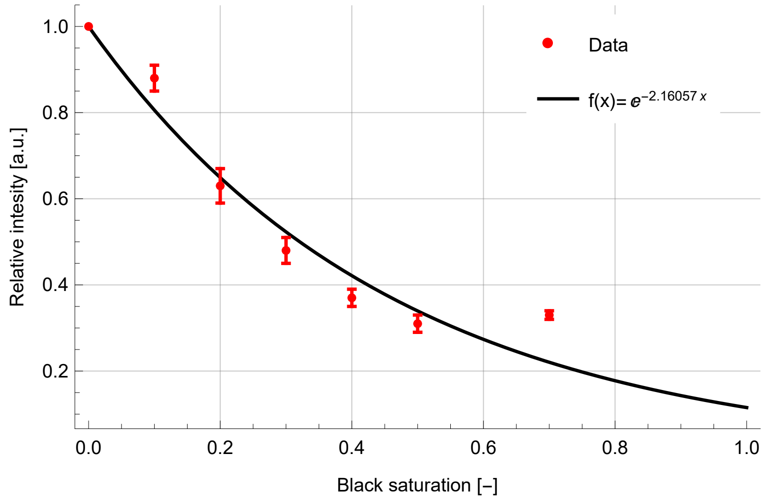

Figure 3.12: Graph of relative intensity versus black saturation including exponential fit.

The reason for using exponential fit instead of any other function is based on experience. It was a needed change to the method since the data obviously do not follow linear function. By extrapolating to 100% saturation of black, we acquire that the relative intensity should be at 11.5% an thus the black correction coefficient $a = 0.115$. As we can see from the graph, the 70% saturation data point is very far from the fit, which warns us, that we should approach this correction with caution.

3.7.2 Measurement of field

To measure the field, we leave the input empty, and measure the absolute value of intensity after integration. I took 50 shots, which I then averaged and compensated for the dark current by subtracting the dark shot. The fields of both the laser and the projector can be seen in figures (figure 3.13) and (figure 3.14) respectively.

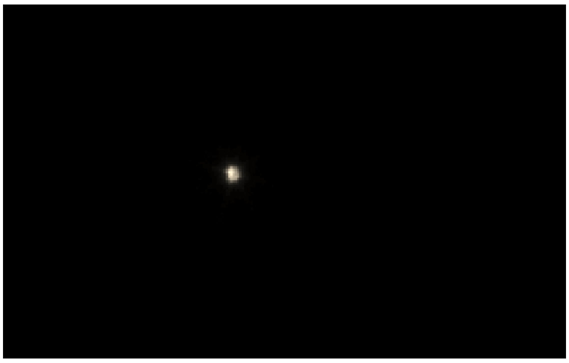

Figure 3.13: Empty field of the laser.

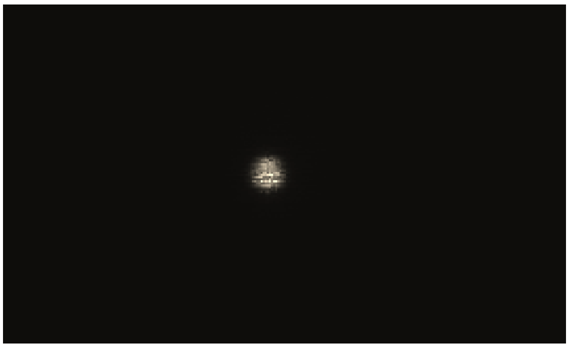

Figure 3.14: Empty field of the projector.

3.7.3 Results of integration

for the purpose of proving that an integrator constructed this way works properly, I choose four functions. Let us look at the results of them one by one now.

$f(x,y) = x + y$

I choose this function for being the simplest, since it represents linear progression through a plane. The mask of the function can be seen in a figure (figure 3.15), and its photo, using the laser beam passing through the mask can be seen in a figure (figure 3.16).

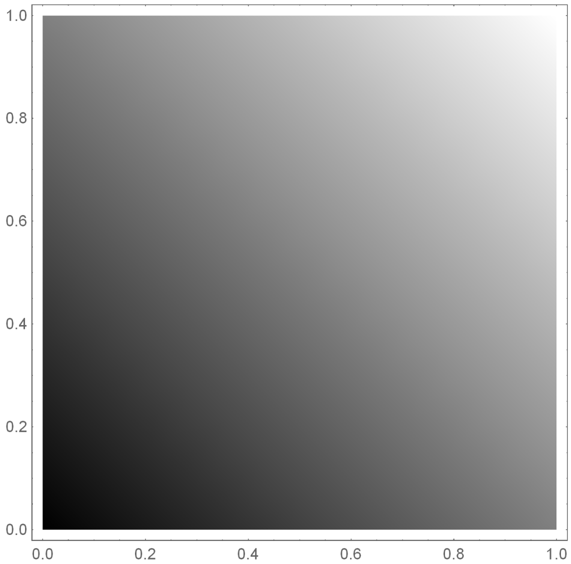

Figure 3.15: Mask representing function $x + y$.

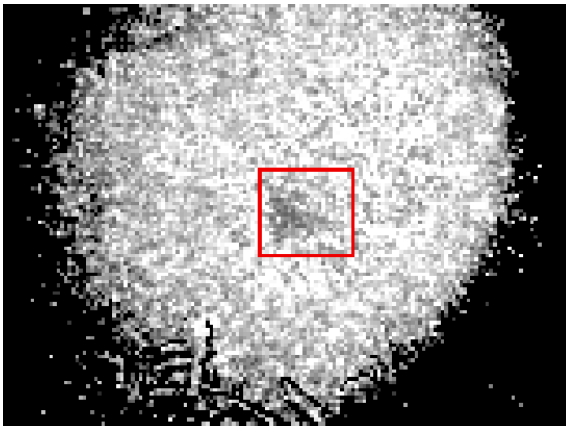

Figure 3.16: A photo of mask representing function $x + y$.

Afterwards, we let the signal pass through our integrator, and by following our method, we acquired the value of $I_{lin} = 0.47 \pm 0.01$. If we compare it to the result of equation

$$I' = \frac{\iint\limits_{x,y=0}^{1} x + y dxdy}{\iint\limits_{x,y=0}^{1} 2dxdy} = 0.5, \tag{3.9}$$

where I' is mathematical solution of the equation, computed using Mathematica, we can seen, that our result is slightly lower than it should be. This was caused, by restricting the pupil by inserting the mask, since the field shot was done without any restrictions to it. We will see, that this is a systematic error, that can be solved by using different modulation, which would not suffer from different pupil sizes during the computation.

If we compare it to the result, we got from the set-up using data projector (processing of the data was unchanged), we get the $I_{linp} = 0.62 \pm 0.04$. This is very far from the true value, and it might have been caused by combination of cutting the field by input pupil (figure 3.9 (left)) and by different spectral response to different wavelengths, which were present due to the usage of white light.

$f(x,y) = xy$

The mask of the function can be seen in a figure (figure 3.17), and its photo, using the laser beam passing through the mask can be seen in a figure (figure 3.18).

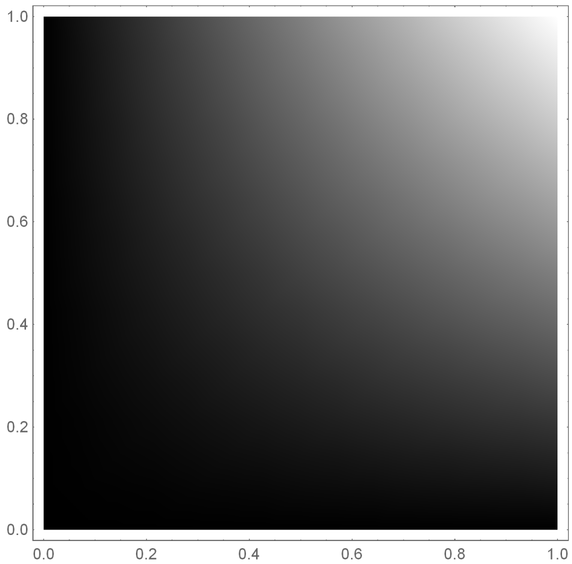

Figure 3.17: Mask representing function xy.

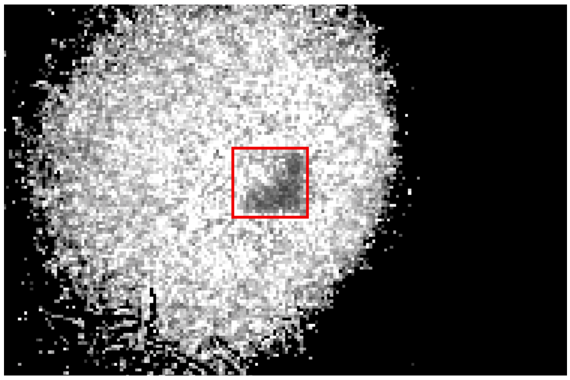

Figure 3.18: A photo of mask representing function xy.

Afterwards, we let the signal pass through our integrator, and by following our method, we acquired the value of $I_{xy} = 0.233 \pm 0.006$. If we compare it to the result of equation

$$I' = \frac{\iint\limits_{x,y=0}^{1} xy dx dy}{\iint\limits_{x,y=0}^{1} 1 dx dy} = 0.25, \tag{3.10}$$

we can seen, that our result is slightly lower than it should be, the cause is the same as in the previous case.

In this case, during data acquisition using the projector, I encountered a technical difficulty, which corrupted the data, and since these data were only for comparison purposes, I decided to not to redo the measurement.

$f(x,y) = \cos(x) + \cos(y) + 2$

The mask of the function can be seen in a figure (figure 3.19), and its photo, using the laser beam passing through the mask can be seen in a figure (figure 3.20). The reason for adding 2, is that the integrator is not capable of processing negative values due to the encoding of information.

Afterwards, we let the signal pass through our integrator, and by following our method, we acquired the value of $I_{cosx+cosy} = 0.47 \pm 0.01$. If we compare it to the result of equation

$$I' = \frac{\iint\limits_{x,y=-2\pi}^{2\pi} \cos(x) + \cos(y) + 2 dx dy}{\iint\limits_{x,y=-2\pi}^{2\pi} 4 dx dy} = 0.5, \tag{3.11}$$

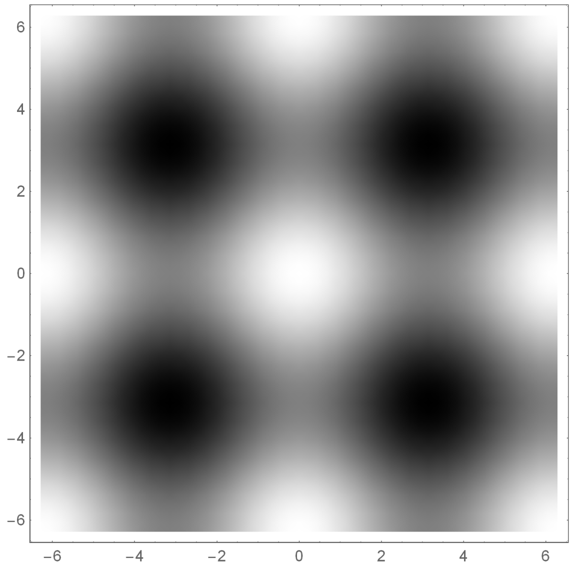

Figure 3.19: Mask representing function $\cos(x) + \cos(y) + 2$.

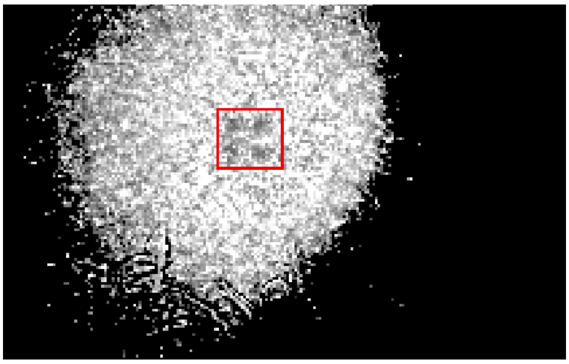

Figure 3.20: A photo of mask representing function $\cos(x) + \cos(y) + 2$.

we can seen, that our result is slightly lower than it should be, the cause is the same as in the previous case.

The result obtained from the set-up utilizing data projector is $I_{(cosx+cosy)p} = 0.37 \pm 0.02$. The cause of the shift here is the same as in the case of the function $x + y$.

$f(x,y) = \cos(x + y) + 1$

The mask of the function can be seen in a figure (figure 3.21), and its photo, using the laser beam passing through the mask can be seen in a figure (figure 3.22). The reason for adding 1, is that the integrator is not capable of processing negative values due to the encoding of information as it was before.

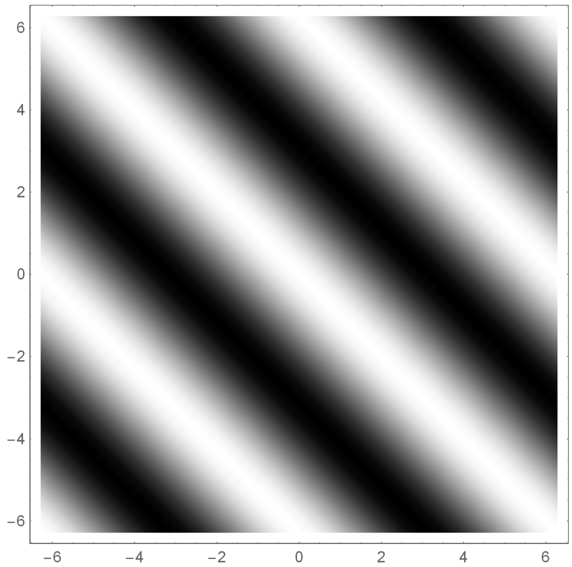

Figure 3.21: Mask representing function $\cos(x + y) + 1$.

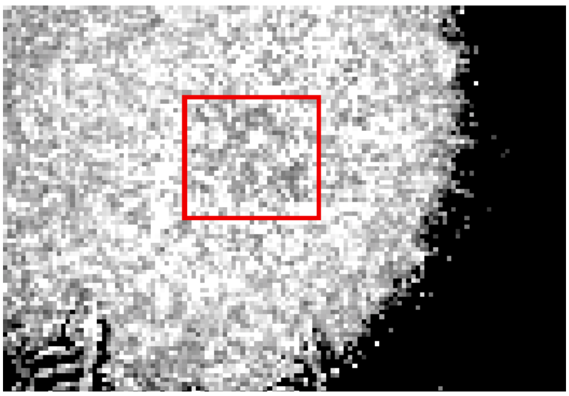

Figure 3.22: A photo of mask representing function $\cos(x + y) + 1$.

Afterwards, we let the signal pass through our integrator, and by following our method, we acquired the value of $I_{cosx+y} = 0.49 \pm 0.02$. If we compare it to the result of equation

$$I' = \frac{\iint\limits_{x,y=-2\pi}^{2\pi} \cos(x+y) + 1 dxdy}{\iint\limits_{x,y=-2\pi}^{2\pi} 2dxdy} = 0.5, \tag{3.12}$$

we can seen, that our result agrees within the uncertainty. The difference between this measurement and the others probably was better placement of the input mask, so that the field was not cropped as much.

The result obtained from the set-up utilizing data projector is $I_{(cosx+cosy)p} = 0.69 \pm 0.03$. The cause of the shift here is the same as in the previous cases.

3.7.4 Summary of the results

We already discussed the results at their respective functions, but for clarity it is better to present them at in a compact form here (table 3.1).

Table 3.1: Results of the computations for both the laser module source and data projector as a source.

Function	Expected result	Laser result	Data projector result
$x+y$	0.5	0.47 ± 0.01	0.62 ± 0.04
xy	0.25	0.233 ± 0.006	no result
$\cos x + \cos y + 2$	0.5	0.47 ± 0.01	0.37 ± 0.02
$\cos(x+y) + 1$	0.5	0.49 ± 0.02	0.69 ± 0.03

As we can see, the results of the computation, using laser module as a source are consistent, but a bit lower, than they should be. This was caused by restricting the pupil when inserting the mask. The only way to counter the error of the measurement is to use better defined modulation, that would keep the pupil size constant through the measurement such as optoelectronic spatial modulator.

The results obtained from the experiment using data projector as a source suffer from large discrepancies. The cause of this is cropping of the input function by circular pupil, which was needed to filter out the surrounding light, for example the cropping of function $\cos x + \cos y + 2$ (figure 3.19) was done in a way that cropped the four maximums at the corners, leaving us with lower result. Another problem with this measurement is calculating the correction parameter, since we do not know the correct response function to the attenuation of the signal, we used the values obtained from the fitted function (figure 3.12) by putting $x = 1$. With that said, using a data projector as a light source with dedicated spatial intensity modulator is a promising path for this platform, but in our experiment suffers heavily from the geometry of the experiment and badly defined correction parameter.

We can try to counter the systematic error, due to the pupil cropping by adding a constant value to the results. We can estimate the value as a mean of differences from the expected results, which is 0.022. Results compensated for this systematic error are listed in a table (table 3.2).

We can also try for better calibration of the measurement using data projector by putting the correction parameter $a = 0.3$ since the data point for 70% saturation of black is close to that value, as well as the previous two values. By doing so, we acquire

Table 3.2: Results compensated for systematic errors of the computations for both the laser module source and data projector as a source.

Function	Expected result	Laser result	Data projector result
$x + y$	0.5	0.49 ± 0.01	0.52 ± 0.04
xy	0.25	0.255 ± 0.006	no result
$\cos x + \cos y + 2$	0.5	0.49 ± 0.01	0.23 ± 0.03
$\cos(x + y) + 1$	0.5	0.51 ± 0.02	0.53 ± 0.05

results that are consistent with the predictions for the both the $x+y$ and $\cos(x+y)+1$, since these functions are not affected by cropping as much as $\cos x + \cos y + 2$ whose result is substantially lower the the predicted value, which is to be expected (table 3.2).

3.8 Limitations of the integrator

Firstly let us consider the technical limitations of the experiment. We encountered systematic error, that decreased the results below the expected value due to the change of the pupil. To counter these limitations we need a spatial intensity modulator that would act as a limiting pupil, thus solving both the problem of different sized pupils for field and signal shots encountered with laser measurement and the problem of cropping the input function, that was present in the data projector part of the measurement.

There is however another limit for our computation. That is the fact, that intensity encoding allows us to only work with positive valued functions. However using laser with high coherence may allow us to code the sign of the value into the phase, that would interfere destructively in the focus point in the case of opposite signs. If we take the example of the function $f(x,y) = \cos x + \cos y$ than we can modulate the intensity of the field by using a mask that represents the absolute value of $f(x,y)$ (figure 3.23) and a phase modulating mask, which represents the sign of $f(x,y)$ (figure 3.24). The encoding of the function's sign leaves the positive values with their original

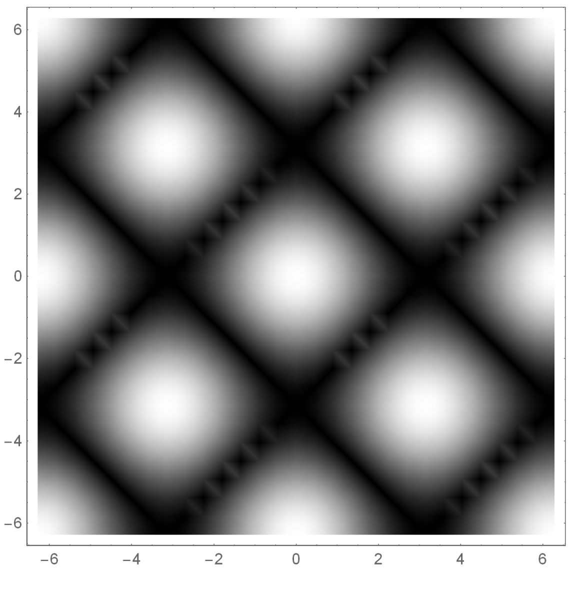

Figure 3.23: Mask representing function $|\cos x + \cos y|$.

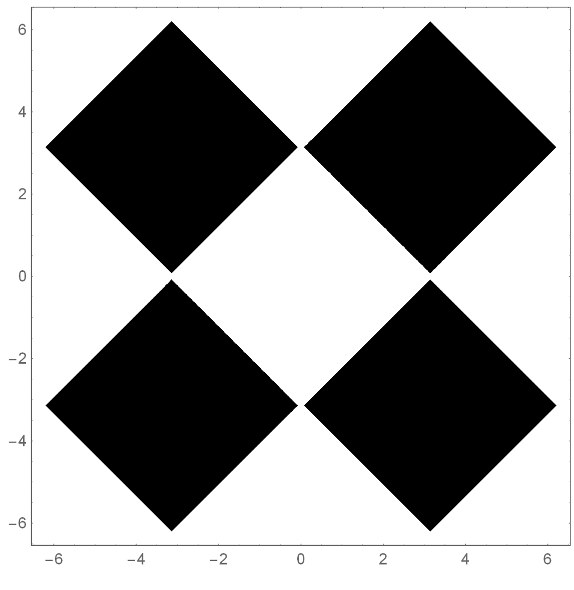

Figure 3.24: Mask representing function $sgn(\cos x + \cos y)$ (black regions represent negative values).

phase and shifts the phase of the negative values by π.

By using this approach, the negative values would interfere destructively with the positive values, and by doing so leaving us the difference between them, which is equivalent of integration of such function.

If we used only the improved modulation, we would be able to integrate across both the positive and negative values, but by using intensity detection, we would be only able to see the absolute value of the computation. To be able to detect even the sign of the result, we require the detection of phase, which can be done using homodyne detection.

3.9 The use of the integrator as a correlator

We can use the constructed device as a 2D optical correlator capable of computing both autocorrelation and cross-correlation functions by a very simple modification.

We need two input masks (or modulators) in series, capable of position shifting, through the relative shift of the two masks, we directly observe a correlation function of the two inputs (figure 3.25).

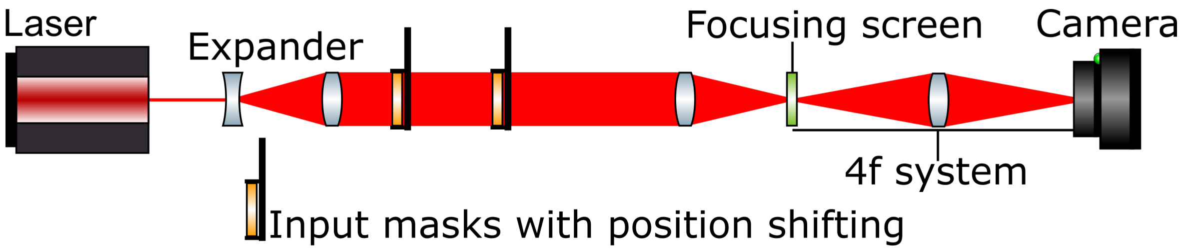

Figure 3.25: Proposal of optical correlator using position shifting.

We can simplify the scheme and quicken the computation by using spatial optical modulators, and fly-eye lens. By modulating copies of the function in different spatial modes of the field and by shifting them accordingly using the two modulators, we would get the correlation function directly as a function of detected intensity at individual pixels of the detector. The only requirement for this is a good definition of the starting point of $[\Delta x, \Delta y] = [0,0]$ is needed. The desired starting pint may differ based on the functions (for example rectangular pupil has has the correlation function symmetrical by the axis x and y, so we need to solve only one quadrant to know the entire function, by doing so, we can use finer sampling of the relative shifting).

Conclusion

In this thesis I choose to deal with the problem of computing using a optical signal.

In the first chapter I described the principles of computing and encoding from the most general point of view. I also described the basic logical functions such as AND, OR, or NOT, just to name a few. I continued by describing analogue operations, mainly differentiation and integration, since these are principal to analogue realm.

I continued by describing the basics of computation from the hardware perspective and as an introduction which brought us to the optical computing, I used electronic approach to describe the thought process of using physical medium for computations. After describing electronic logical gates and integrators and differentiators I have shown a simple schematic, that could be used for solving differential equations of second order (figure 1.8).

I continued the chapter by reviewing the possibilities of constructing optical logical gates. I focused on three most prominent branches, that are Mach-Zhender interferometer (MZI) based gates, photonic crystal based gates and microring resonators based gates. I describe the principle for every branch and in the summary I discussed their advantages and disadvantages. In the last part of first chapter I described optical integrator based on microring resonators.

The second chapter was dedicated to my designs of optical gates, for which I choose MZI-like architecture and intensity encoding based on my design principles, that opened the chapter. I then followed with stating the advantages of free-space optics, which I used for the designs, and its promise in parallel computing. Through the idea of parallel computing I expanded the thought of this thesis to include optical integration and its possibility for differential equation solving (figure 2.10).

I started the last chapter by discussing the computation protocol for encoding and recovering information. Then I analysed the experimental set-up from the perspective of noise coming into the measurement based on fundamental physical laws. This was followed by discussing the used equipment in the set-up (figure **??**) and the influence of the elements from the perspective of measurement uncertainties. I followed by a clear measurement procedure, that I used during the experiment.

Afterwards I presented and discussed the results (table 3.1) of the integration for four functions:

- $f(x,y) = x + y$,
- $f(x,y) = xy$,
- $f(x,y) = \cos x + \cos y + 2$,
- $f(x,y) = \cos(x + y) + 1$.

I then discussed the errors of the computations and found out that the system is suffering from systematic error, that is inherent to the cropping of field by the pupil of the masks.

In the last section I proposed not only a way to deal with the systematic error, but also an idea for further improving the integrator as a whole, that would allow us both encode and recover negative values, by expanding the encoding from intensity encoding to intensity and phase encoding and by expanding the detection form using single camera to using homodyne detection.

Bibliography

[1] YALCIN, M.E., J.A.K. SUYKENS and J. VANDEWALLE. *True Random Bit Generation From a Double-Scroll Attractor*. IEEE Transactions on Circuits and Systems I: Regular Papers [online]. 2004, 51(7), 1395-1404 [cit. 2018-05-18]. DOI: 10.1109/TCSI.2004.830683. ISSN 1057-7122. Available form: http://ieeexplore.ieee.org/document/1310510/

[2] *File:Logic-gate-index.png*. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2005 [cit. 2020-04-15]. Available from: https://commons.wikimedia.org/wiki/File:Logic-gate-index.png

[3] *Bipolar junction transistor*. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2020 [cit. 2020-04-16]. Available form: https://en.wikipedia.org/wiki/Bipolar_junction_transistor

[4] *MOSFET*. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2020 [cit. 2020-04-16]. Available form: https://en.wikipedia.org/wiki/MOSFET

[5] GUO, Ning, Yipeng HUANG, Tao MAI, Sharvil PATIL, Chi CAO, Mingoo SEOK, Simha SETHUMADHAVAN and Yannis TSIVIDIS. *Continuous-time hybrid computation with programmable nonlinearities*. In: ESSCIRC Conference 2015 - 41st European Solid-State Circuits Conference (ESSCIRC) [online]. IEEE, 2015, 2015, s. 279-282 [cit. 2020-04-16]. DOI: 10.1109/ESSCIRC.2015.7313881. ISBN 978-1-4673-7470-5. Available from: http://ieeexplore.ieee.org/document/7313881/

[6] *File:16QAM Gray Coded.svg*. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2006 [cit. 2020-05-14]. Available from: https://commons.wikimedia.org/wiki/File:16QAM_Gray_Coded.svg

[7] NAUTIYAL, Ambika and Prem B. BISHT. *Sum and difference frequency generation of white light continuum with the ps pulses of Nd+3: YAG laser in a thick BBO crystal*. Optics Communications [online]. 2007, 278(1), 175-179 [cit. 2020-05-14]. DOI: 10.1016/j.optcom.2007.05.067. ISSN 00304018. Available from: https://linkinghub.elsevier.com/retrieve/pii/S0030401807005561

[8] Zasedatelev, A.V., Baranikov, A.V., Urbonas, D. et al. *A room-temperature organic polariton transistor*. Nat. Photonics 13, 378–383 (2019). https://doi.org/10.1038/s41566-019-0392-8

[9] WEBB R.P., R.J. MANNING, G.D. MAXWELL and A.J. POUSTIE. *40Gbit/s all-optical XOR gate based on hybrid-integrated Mach-Zehnder interferometer*. Electronics Letters [online]. 2003, 39(1) [cit. 2020-05-17].

DOI: 10.1049/el:20030010. ISSN 00135194. Available from: https://digital-library.theiet.org/content/journals/10.1049/el_20030010

[10] RENDÓN-SALGADO, I., E. RAMÍREZ-CRUZ and R. GUTIÉRREZ-CASTREJÓN. *640 Gb/s all-optical AND gate and wavelength converter using bulk SOA turbo-switched Mach-Zehnder interferometer with improved differential scheme*. Optics & Laser Technology [online]. 2019, 109, 671-681 [cit. 2020-05-17]. DOI: 10.1016/j.optlastec.2018.08.055. ISSN 00303992. Available from: https://linkinghub.elsevier.com/retrieve/pii/S0030399218301774

[11] *MaxwellFDFD* [online]. 2019 [cit. 2020-05-17]. Available from: http://www.mit.edu/ wsshin/maxwellfdfd.html

[12] YOUNIS, Raghda M., Nihal F. F. AREED and Salah S. A. OBAYYA. *Fully Integrated AND and OR Optical Logic Gates*. IEEE Photonics Technology Letters [online]. 2014, 26(19), 1900-1903 [cit. 2020-05-24]. DOI: 10.1109/LPT.2014.2340435. ISSN 1041-1135. Available from: http://ieeexplore.ieee.org/document/6858067/

[13] RANI, Preeti, Yogita KALRA and R.K. SINHA. *Design of all optical logic gates in photonic crystal waveguides*. Optik [online]. 2015, 126(9-10), 950-955 [cit. 2020-05-24]. DOI: 10.1016/j.ijleo.2015.03.003. ISSN 00304026. Available from: https://linkinghub.elsevier.com/retrieve/pii/S003040261500128X

[14] IBRAHIM, T.A., R. GROVER, L.C. KUO, S. KANAKARAJU, L.C. CALHOUN and P.T. HO. *All-optical AND/NAND logic gates using semiconductor microresonators*. IEEE Photonics Technology Letters [online]. 2003, 15(10), 1422-1424 [cit. 2020-05-24]. DOI: 10.1109/LPT.2003.818049. ISSN 1041-1135. Available from: http://ieeexplore.ieee.org/document/1232978/

[15] *CPU FREQUENCY: HALL OF FAME* [online]. 2020 [cit. 2020-05-25]. Available from: https://hwbot.org/benchmark/cpu_frequency/halloffame

[16] FERRERA, Marcello, Yongwoo PARK, Luca RAZZARI, Brent E. LITTLE, Sai T. CHU, Roberto MORANDOTTI, David J. MOSS and José AZAÑA. *All-optical 1st and 2nd order integration on a chip*. Optics Express [online]. 2011, 19(23) [cit. 2020-05-25]. DOI: 10.1364/OE.19.023153. ISSN 1094-4087. Available from: https://www.osapublishing.org/oe/abstract.cfm?uri=oe-19-23-23153

[17] *Lens*. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2020 [cit. 2020-05-28]. Available from: https://en.wikipedia.org/wiki/Lens

[18] *Imaging Electronics 101: Understanding Camera Sensors for Machine Vision Applications*. Edmund Optics [online]. [cit. 2020-05-30]. Available from: https://www.edmundoptics.eu/knowledge-center/application-notes/imaging/understanding-camera-sensors-for-machine-vision-applications/

[19] *Image noise*. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2020 [cit. 2020-05-30]. Available from: https://en.wikipedia.org/wiki/Image_noise

[20] *KAF-3200ME* [online]. Rochester, New York: Eastman Kodak Company, 2002 [cit. 2020-05-30]. Available from: https://www.datasheets360.com/pdf/-9054014344235087618