Detection System and Enclosure Design

for Quantum Random Number Generation

by

Margo Molenaar & Feike Pacilly

Instructor:	R. Ishihara
Teaching Assistant:	S. Nur, J. Varveris, S. Yu
Institution:	Delft University of Technology
Place:	Department of Quantum & Computer Engineering, Faculty of Electrical Engineering, Delft
Project Duration:	04, 2021 - 07, 2021

Abstract

This report focuses on the design and implementation of both a detection system and an enclosure, which are designed to be used in a quantum random number generator. These parts are designed to be manufactured using off-the-shelf components to make the quantum random number generator affordable and accessible to a completely new user-group that cannot afford and operate the currently existing alternatives for quantum random number generation. After implementing the designed systems it is found that although the output of the system is random, true-randomness cannot be guaranteed using off the shelf components, because of the interference of classical noise sources.

Preface

This report is written as part of the bachelor graduation project of electrical engineering. Together with three other electrical engineering students we have been working on a quantum random number generator for the past 2.5 months. The whole group delved into the world of quantum random number generation and electrical system design to come up with the necessary designs to create this quantum random number generator and implement and test its components. We hope that our work will be useful for the field and that it can help to make quantum random number generation more affordable and accessible to a larger group of users.

Furthermore we would like to express our gratitude to our supervisor Ryoichi Ishihara who has helped and supported us through the course of the project. Also the teaching assistants Salahuddin Nur, John Varveris and Shuichang Yu have helped us a lot by sharing their insights, feedback and by being available to ask questions. And at last we would also like to thank our team mates Ricardo Boshuisen, Ashti Kasem and Romario Sobhi, who were great to collaborate with during the project, which we really enjoyed.

Margo Molenaar & Feike Pacilly
Delft, June 2021

Contents

1
Introduction

1.1. Quantum Random Number Generation

Random numbers are becoming significantly important as businesses and services that need random keys are more frequently being used nowadays. Next to the demand for high random number generation rates, the higher demand for better randomness becomes apparent to facilitate high secure services through the generation of long random keys that are nearly impossible to guess [38]. Examples of where random numbers are often used are for cryptography, computer simulations, lotteries and for password generation.

The definition of randomness is that the outcome is unpredictable, that all options have an equal chance to occur and that there is no correlation between past measurements and future measurements [15]. Right now, the most often used type of random number generators are pseudo-random number generators. These random number generators use various hard to predict measurements such as low significant bits of counters or background noise measurements and perform some long complicated algorithms on the measured data to obtain pseudo-random data. Although pseudo-random numbers generated this way are very hard to predict, it is theoretically still possible. Pseudo-random number generators that work this way often have undesirable security flaws since although it is really hard to determine which data has what influence on the output, overloading the sensors which act as seed for the pseudo-random number generators may influence its output and in the worst case make the output easier to predict. Furthermore, at high bitrates some seeds like counters and background noise will become less random and more predictable. Therefore the demand for true-random number generators is increasing as the demand for high throughput and highly-random generators is requested. This is where quantum random number generators (QRNGs) enter the field. To generate random output values, QRNGs make use of quantum effects that are by definition random. This gives QRNGs the potential to generate true-random data that is in no way connected to external factors. Because quantum effects take place in very small time frames, the bitrates that QRNGs can potentially achieve are also very high.

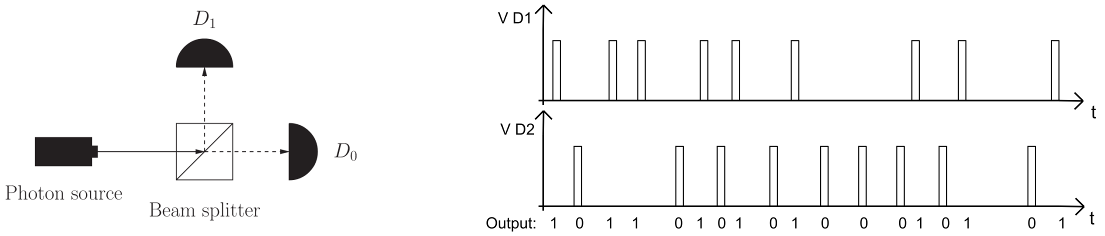
1.2. State of the Art Analysis

While in the middle of a technological revolution, the need for random number generators is greatly increasing as is the need for high security and high throughput ones [17]. This results in a lot of money spent on research and innovation in different field where random numbers are required [16]. Therefore QRNGs are far from new and a lot of research about QRNG has already been published. Actually, QRNGs are already commercially available [42].

There are different types of QRNGs that use different quantum effects to generate random numbers. Two popular effects that are often used are based on spatial superposition of photons and the temporal difference in detection time of photons [42]. Both methods generate output bits (0 or 1) that are by definition true-random, but they do so in a completely different way.

Spatial superposition

The spatial superposition technique makes use of a polarizer that circularly polarizes the light of a laser which is then sent through a beam splitter. In the beam splitter every photon has a 0.5 chance to go to direction 1 and a 0.5 chance to go to direction 2, because the beam splitter will split spin-up photons from spin-down photons [42]. By placing two detectors at both sides, we can measure which path the photon took and then directly translated into a 0 or a 1 as output.



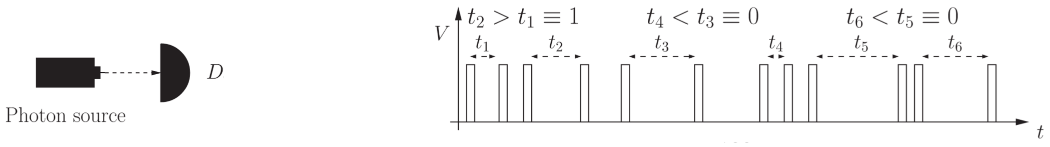
(a) The physical setup of the system.

(b) The signals received at the detectors.

Figure 1.1: Schematic representation of quantum random number generation based on spatial superposition [14].

Arrival time difference

When the difference in detection time is used, single photon pulses are measured and the times between two pulses are compared to either generate a 0 or a 1. It is important in this technique that the two times are completely independent, therefore the time between pulse 1 and 2 is compared to the time between pulse 3 and 4. [14] [42]



(a) The physical setup of the system.

(b) The measured signal at the detector.

Figure 1.2: Schematic representation of quantum random number generation based on time of arrival differences [14].

Both techniques make use of photons for the quantum effects, but different ways of generating these photons can be used. The two most often used methods are either by using some kind of radioactive material that occasionally emits a photon due to radioactive decay and more conventional light sources. Both methods have advantages and disadvantages, but in general a piece of radioactive material is more suited in the situation where a relatively low photon count is necessary while conventional light sources are used when a high photon count is used.

1.3. Problem Definition

Although QRNGs are not new, they are often expensive, not portable and not suited for the everyday user market. Most published papers about QRNGs either only discuss the various techniques that can be used to generate quantum random numbers or make use of single photon detectors based on for example superconducting nanowires. Although this makes it possible to create very high quality QRNGs, it also makes them very expensive and bulky. Setups like these work well in the lab but are not suited for small businesses let alone for private use.

For this reason a bachelor graduation project on electrical engineering was set up to design a portable QRNG based on off-the-shelf components [27]. By only using off-the-shelf components the QRNG will be relatively cheap to produce. This means that by designing such a system, the technology can be made available to a group of users that could otherwise not realistically afford to use it.

1.4. System Subdivision

The bachelor graduation project group tasked to build this QRNG, consists of 5 electrical engineering students. As a whole group and after careful consideration, with the use of a literature study, we chose to build a QRNG based on spatial superposition, as was also proposed in the original project outline [27]. Subsequently, the group was subdivided into two subgroups one consisting of 2 and one consisting of 3 people. Our subgroup was tasked with designing, implementing and testing the detection system and the enclosure for the QRNG.

1.5. Thesis Outline

The main topic of this thesis is the design, implementation and testing of the detection system and the enclosure. In Chapter 2 the requirements for both the detection system and the enclosure will be discussed. Chapter 3 will focus on the design of the detection system while Chapter 4 will focus on the design of the enclosure. Chapter 5 is about how both designs are implemented, tested and integrated into the rest of the system. The final results will be discussed and elaborated on in Chapter 6 and finally Chapter 7 contains the conclusion, future work and recommendations.

2

Programe of Requirements

As stated in Chapter 1 our subgroup is responsible for designing the detection system and the enclosure for the QRNG. To make these concepts more discrete this chapter will discuss the requirements of both these designs. The requirements are subdivided into mandatory requirements, consisting of functional and non-functional requirements, and trade-off requirements.

2.1. Detection System

Together with the other subgroup we have decided to go for a QRNG based on spatial superposition, which means that two detectors will be used. Furthermore since to system should be made with off-the-shelf components, single photon detection is not possible which means that we will be comparing the intensities of both detectors, therefore the QRNG will not be perfect. Based on these assumptions the following requirements have been set up.

2.1.1. Mandatory Requirements

Functional

- **Generate output bits.** The main task of the detection system is to facilitate the generation of random data. To do this the system should be able to generate bits.
- **The design should consist of off-the shelf components.** The problem our bachelor graduation project is set up to solve is that current QRNGs are either too expensive or complex to be used by an everyday users. To actually solve this problem, the system can not contain any exclusive or hard to obtain parts.
- **Fit inside the enclosure.** The detectors should fit within the enclosure that we will design as well. This does mean that the detection system should be small enough to facilitate the portable nature of the enclosure.
- **The prototype should be safe to handle.** Since we will be working on the prototype, the system should not be able to hurt us. This means no lasers more powerful than 5 mW and no voltages higher than 40 V.
- **The output data should be delivered in a way that it can be stored on a PC.** The data generated by the detection system has to be stored on a PC. To make this possible the data should be delivered to the rest of the system in a way that facilitates this.

Non-Functional

- **Have a bit rate of at leas 2 kB/s.** The user of the QRNG should not have to wait longer than about a second to generate a random key. The value for the bit rate is based on the longest key we expect an average user to generate, which is 256 characters. Since each character consists of 8 bits, this is a total of 2048 bits.
- **Have a lifespan of at least 3 years.** For the QRNG to be affordable enough to be personally used, users should not have to buy a new QRNG every year because the old one stopped working. Therefore the lifespan of the detector should be at least 3 years.

- **The detector should work on 5 V.** The whole QRNG will be powered from the 5 V connection with the computer to which it is connected. Therefore the detector should work on this 5 V as well.
- **The detector should not draw more than 50 mA.** A standard USB connection should be capable of at least delivering 200 mA. Since 150 mA of this is reserved for the laser and the microcontroller, the detector should not use more than 50 mA.

2.1.2. Trade-off Requirements

- **Maximize the true randomness of the system.** Although using off-the-shelf components, and therefore a system that only generates true-random bits is not feasible, the quantum randomness of the output bits should be maximized. This also means that the influence of environmental effects on the output should be minimized.
- **Maximize the bit rate.** Although a minimum limit for the bit rate is already set in Section 2.1.1, a higher bit rate will decrease the time the user has to wait on the data.
- **Minimize costs.** The system has to be designed using off-the-shelf components to make the technology affordable and reachable for more users. Minimizing the cost of the detector will help reaching this goal.

2.2. Enclosure

The enclosure is necessary to physically protect the inner hardware and can help to shield the electronics and optics from environmental interference. Although QRNG do not depend on external factors to generate random numbers, these factors might still interfere with the system in the form of noise. Therefore the enclosure should prevent this as much as possible. The following requirements are set up for the enclosure.

2.2.1. Mandatory Requirements

Functional

- **The enclosure should physically protect the QRNG.** Since the QRNG is supposed to be portable, it has to be handleable without a high risk of breaking or damaging the internal components.
- **All parts of the QRNG should fit within the enclosure.** The enclosure should protect the whole QRNG, so all parts should fit within it.
- **Isolate internal voltages.** No internal voltages except for the ground may be accessible when the enclosure is closed. This is necessary to prevent any kind of short or malfunction of the system when something conducting touches the enclosure.
- **Have a lifetime of at least 3 years.** For the QRNG to be affordable to be personally used, users should not have to buy a new one every year because the old one is damaged. Therefore the enclosure should be able to last for at least 3 years.

Non-Functional

- **The size of the enclosure should be less than 40 x 20 x 20 cm.** For the system to be reasonably portable, this is the maximum size the enclosure may have.
- **The enclosure should not weight more than 500 g.** For the QRNG to be portable it should not weight more than 1 kg, since the enclosure is the largest part of the system it may have a relatively high contribution to this, but it should not be heavier than 500 g.

2.2.2. Trade-off Requirements

- **Minimize the transmission of electromagnetic waves.** Environmental light and other electromagnetic waves that enters the system may influence the output of the QRNG. This is not desirable, so amplitude of electromagnetic waves that enter the system should be minimized.
- **Minimize the volume.** The system has to be portable. The smaller the enclosure is the more portable it becomes.

3

Detection System

This chapter will discuss the function, working principle and design of the detection system.

3.1. Introduction

The detection system is responsible for measuring the photons that went to either one of the detectors and derive output bits based on these measurements. Because it is not feasible to measure single photons using off-the-shelf parts, the detectors will instead measure intensities of the light and derive output bits based on fluctuations in these intensities. In this chapter we will discuss how we used this principle to design a system that does exactly this.

The detection system is part of the QRNG. Figure 3.1 shows where in the overall system the detection system is located. The detection system includes everything from the point where the photons touch the photodetector to the function on the microcontroller code that returns random bits.

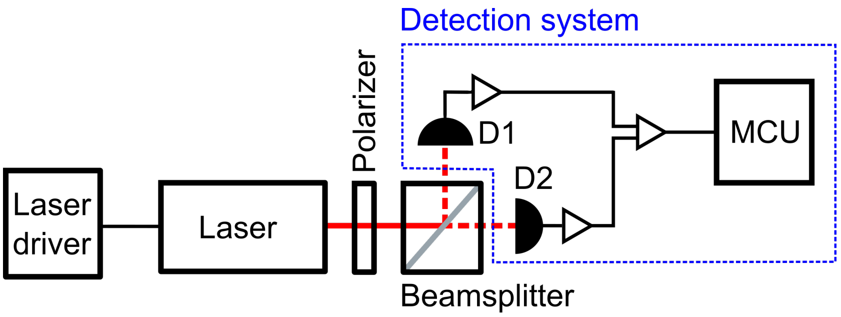


Figure 3.1: Schematic representation of the complete QRNG system. The part of the system which we call the detection system is highlighted in blue.

3.2. Methodology

This section will discuss the basic structure of the detection system, the method used to extract random bits from the quantum effect and the physics behind this effect.

3.2.1. System Structure

As stated, the detection system works by comparing the photon intensities of two detectors. So the first components of the system will be the photodetectors. These will be both be followed by amplifiers and DC-blocking filters that amplify the signal from the photodetectors to a level at which both signals can be compared using a comparator. The output of this comparator will alternate between high and low based on which of the detectors measure the highest intensity. This signal will be measured by the microcontroller on fixed time intervals to obtain a random number. To ensure the signal does not change while the microcontroller is measuring a flip-flop will be used to keep the signal constant. The clock for the flip-flop is generated by the microcontroller. Figure 3.2 shows a schematic representation of the detection system. It must be noted that although the microcontroller is drawn as part of the detection system, not all code on the microcontroller is dedicated to the detection system and therefore only part of the microcontroller should be considered as part of the detection system. What the different parts shown in Figure 3.2 exactly do, why they are necessary and how they are designed will be discussed in Section 3.4 until 3.7.

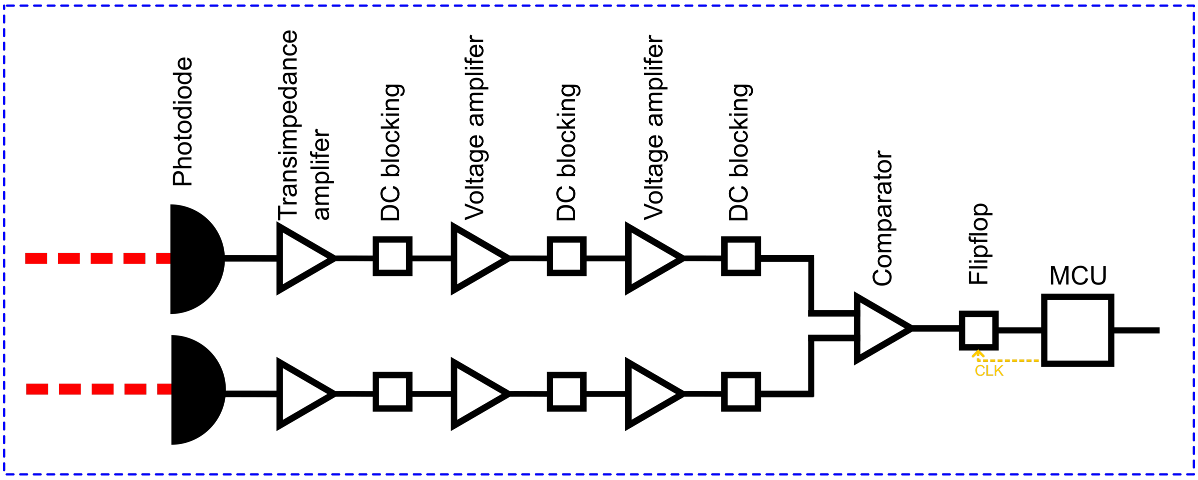


Figure 3.2: A schematic representation of the detection system and the parts of which the it consists. (Note that the photodiodes are drawn side-by-side, instead of under a 90° angle. This is purely done for illustration purposes and does not represent the physical orientation.)

We have chosen to use this setup because it is the most feasible option using off the shelf components. Quite a few papers have been written about the setup based on spatial superposition and about it successfully working, although those typically use single photon detectors. The setup does have some drawbacks which have to be accounted for. One of these drawbacks is a possible bias between both detectors. Since a bias will by definition make the output less random, effort has to be put into preventing a bias between both detectors. An other disadvantage of using the difference in intensities is that when comparing both analog signals, classical noise will enter the system. Although this cannot be prevented, the influence of classical noise on the output of the signal should be minimized. How we tackle these problems will be discussed further on in this chapter.

3.2.2. Physics

In our proposed setup, the quantum randomness is generated by a form of spatial superposition. To achieve this superposition, circular polarized light is send through a beamsplitter which splits the horizontal polarized light from the vertical polarized light. This means that every photon that has passed though the beam splitter either went straight through the beamsplitter or got reflected to the side. As long as the light is circularly polarized at the moment it passes through the beamsplitter, the probability of both of these outcomes happening is 0.5, since the beam splitter splits the spin-up photons from the spin-down photons. Figure 3.3 shows a schematic representation of the polarization states of the light in different parts of the optical system.

3.2.3. Intensity Based

Since measuring single photons using off-the-shelf components is not feasible, the detectors will measure photon intensities instead, this means that the output will be based on which detector measures the most photons per time unit. This does however raise the question about how this influences the randomness of the result.

Inherently, the result of such an experiment will still be random. To prove this we assume that each photon can either go to detector 1 or detector 2, respectively resulting in an output of 1 or 0. Since the

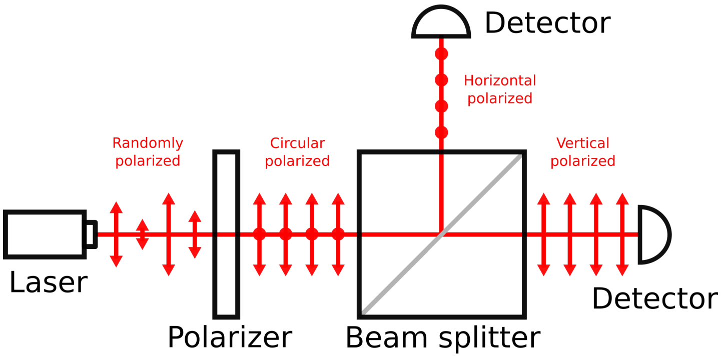


Figure 3.3: Polarization states of the light at different stages in the system.

result of this experiment is random, we call this random variable X with a probability for both outcomes of 0.5.

$$X \in \{0,1\} \quad \text{and} \quad P(X=0) = P(X=1) = 0.5 \tag{3.1}$$

Now we consider an experiment where not only one of the random variables, introduced in Equation 3.1, is measured but a sum of N of these events, this yields Equation 3.2. The output of this experiment is called θ, and follows a binomial distribution.

$$\theta = \sum_{n=0}^{N} X_n \tag{3.2}$$

Since every photon measured by detector 1 is not measured by detector 2 and vise-versa, Equations 3.3 and 3.4 can be derived for both detectors.

$$\theta_{D1} = \sum_{n=0}^{N} X_n \tag{3.3}$$

$$\theta_{D2} = \sum_{n=0}^{N} (1 - X_n) \tag{3.4}$$

In the end we will be comparing θ_{D1} and θ_{D2} to generate bits. So, in order to make sure our output is random as well, we need to check that the probabilities of $\theta_{D1} < \theta_{D2}$ and $\theta_{D1} > \theta_{D2}$ are both equal to 0.5. Using that θ_{D1} and θ_{D2} are complementary and that they both follow binomial distributions it is indeed found that for large enough values of N Equation 3.5 holds. The reason N has to be large enough is because there is a probability that θ_{D1} and θ_{D2} are exactly equal, but for large values of N this probability can be neglected. The derivation of Equation 3.5 can be found in Section A.1.

$$P(\theta_{D1} < \theta_{D2}) = 0.5 \tag{3.5}$$

This means that measuring intensities instead of single photons does result in a truly random output.

3.3. Limitations

Although theoretically the system we will use does produce perfect true random bits, in reality the measurements will be influenced by imperfections in the setup. In this section we will discuss the most important imperfections we should consider and how we can optimize the system to minimize the effect of these imperfections.

3.3.1. Bandwidth

The bandwidth of the amplifiers used to amplify the signal from the photodetectors plays an important role in the randomness of the output of the system. The maximum achievable frequencty that the amplifier can amplify however, is limited. The maximum frequency is limited by the bandwidth of the photodetectors and the bandwidth of the op-amps that will be used to amplify the signal.

As stated using intensity measurements, measuring the sum of multiple independent random events, to determine an output value does result in valid truly random bits, since the probability of either detector having the highest intensity is the same. However, for the result to be truly random it also has to be unpredictable. This is where the bandwidth becomes important.

Maximum Frequency

If the maximum frequency the system can handle is too low, the chance of measuring two equal digits in a row will be higher than the chance of measuring two different digits. This is because after performing a measurement the signal needs time to swing to the opposite state. If the sample frequency is too high compared to the maximum frequency the amplifiers can amplify, this means that part of the random events that decided the output of a measurement also play a role in the second measurement which makes the measurements correlated and therefore not random.

To decide on the maximum frequency the amplifiers should be able to handle, we will make use of the time constant which is denoted by τ. The time constant of the amplifiers is defined as the reciprocal value of the angular cutoff frequency w, this is shown in Equation 3.6.

$$\tau = \frac{1}{\omega_{-3dB}} = \frac{1}{2\pi f_{-3dB}} \tag{3.6}$$

The step response of an amplifier with a time constant τ will approximately be within 1% of its steady state value after a time of 5τ has passed [40]. Therefore we will assume that the dependence of the output state of the amplifiers, on photons that were detected a time of 5τ in the past can be neglected. This makes it possible to calculate a minimum for the bandwidth of the amplifiers. This is done using Equation 3.7 in which T_s is the time between measurements. Based on the bitrate of 2kB/s as specified in the requirements (Chapter 2) a minimum for the bandwidth of 1.6 kHz is found.

$$f_{-3dB} = \frac{1}{2\pi\tau} = \frac{2.5}{\pi T_s} = 1.6\ kHz \tag{3.7}$$

3.3.2. Noise and Randomness

Next to measuring the quantum effect for which the system is designed, it is inevitable that classical noise will enter the system as well. Although most of this classical noise seems random as well, it is not true random but instead pseudorandom. To make sure the output of the QRNG is based as much as possible on the quantum effect and not on the pseudorandom classical noise, the influence of classical noise sources on the output should be minimized as much as possible. In this section multiple classical noise sources will be addressed and we will discuss how the influence of these sources is minimized.

Dark Count

Almost all types of photon sensors experience a phenomenon called dark current or dark count. This refers to the signal strength that is being detected by the sensor while not under illumination. In semiconductor based photodetectors, dark current is related to thermal excitation of the carriers [8]. This adds classical noise to the system which is not desirable. How strong the dark current is depends on the type of photodetector and how it is used. Therefore the best way to minimize dark current is to carefully pick the type of photosensor, which will be done in Section 3.4.

Thermal Noise

Resistive components used in the amplifier will introduce thermal noise into the system caused by agitation of electrons in these components [31]. Especially components that add noise early in the system before the signal is amplified can play a significant role in the classical noise at the output. In Section 3.5 the design of the amplifiers will be discussed, while designing these amplifiers, possible ways to minimize the thermal noise are carefully considered. The total output noise that the amplifiers will get at the output is calculated in Section 3.5.5 by means of a simulation.

Environmental Noise

Another potential source of noise is electromagnetic interference that enters the amplifiers via the wires and connection points. To reduce the effect of this, the length of the connections before the amplifiers will be held short. Furthermore the enclosure will protect the system against interference by completely surrounding the system into a conducting piece of foil. The design of this enclosure will be discussed in Chapter 4.

3.4. Photodiode

There are different kinds of detectors that can be used to detect the incoming photons. Since we will measure light intensities instead of single photons, the most likely candidates are photoresistors, phototransistors and photodiodes. Since the project has to be based on off the shelf components, more advanced types of photodetectors such as avalanche diodes, photomultipliers or superconducting nanowires are outside the scope of the project and therefore not considered. For our system the most important properties the photodetectors should have are:

- High sensitivity
- Applicable for our frequency
- Low noise and low temperature dependence
- Affordable

The photoresistor is the most simple in design and the most affordable option. However the photoresistor is also quite temperature sensitive and suffers from thermal noise. The phototransistor has a high sensitivity but the gain is temperature dependent. The photodiode has a good sensitivity, can handle fast deviations and has a low dark current in reverse bias configuration. However the directivity of photodiodes is higher, which means that more care should be taken when aligning them. [7] [9] [11] [41]

Considering the requirements for the detector and the pros and cons for the possible options, a photodiode is the best option. The photodiode will be operated with a reverse voltage of 0V. This minimizes the noise cause by the dark current, which will minimize the classical interference. This does however also lower the sensitivity of the photodiode [7] [8].

Different types of photodiodes are considered and the most promising types are summarized in Table 3.1. The most important property is the sensitivity at 635 nm, namely the wavelength of the laser. So only the photodiode types with the highest sensitivity are shown in the table. For these photodiodes we looked more closely to the bandwidth, the effective photosensitive area and the power dissipation. Considering those properties we chose to use the S5973 photodiode.

Type	**Bandwidth (MHz)**	**Sensitivity at 635 nm (A/W)**	**Effective photosensitive area (mm^2)**	**Power dissipation (mW)**	
S5971	100	0.43	1.1	50	[13]
S5973	1000	0.43	0.12	50	[13]
S1223	30	0.43	6.6	100	[12]
SFH 213	200	0.43	1	150	[32]
SFH 229	100	0.43	0.31	150	[33]

Table 3.1: Comparison between multiple photodiodes.

3.5. Amplifier Design

In this section the design and design decisions of the amplifier are discussed. The amplifier is responsible for amplifying the signal generated by the photodiode into a signal that can be compared by the comparator.

3.5.1. Voltage reference driver

The voltage reference driver is connected to the 5V power supply and convert this to a stable 2.5V which is used by the transimpedance amplifiers and voltage amplifiers as a reference. Different voltage reference drivers are compared in Table 3.2.
We decided to chose the REF5025AIDG4 because it has the best accuracy while being able to deliver a sufficient output current. Figure 42 from the datasheet [22] already showed the basic circuit for implementing the voltage reference driver and is also shown in Figure 3.4. We used this setup in the final design.

Type	V output (V)	I output (mA)	Noise	Accuracy (%)	
REF3025	2.5	25	80uV	0.2	[18]
REF5025AIDG4	2.5	-+10	3uVpp/V	0.05	[22]
REF6025	2.5	-+ 4	3uVpp/V	0.05	[23]

Table 3.2: Comparison between voltage reference drivers.

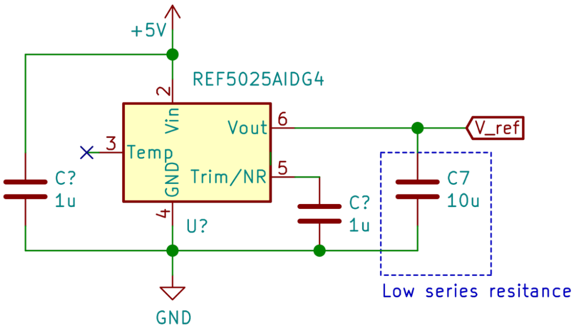


Figure 3.4: The circuit for connecting the voltage reference driver [22].

3.5.2. Transimpedance Amplifier

The photodiodes produce a current based on the photons it absorbs. The first amplifier that will read out the signal from the photodiodes is a transimpedance amplifier, which converts and amplifies the current produced by the photodiodes into a voltage.

To determine what gain the transimpedance amplifier should have, we first calculated the output current of the photodiode based on the expected power it will absorb from the laser. This is done using Equation 3.8 in which I is the signal current from the photodiode, P is the power absorbed by the photodiode, which is half of the power transmitted by the laser, since power is divided by the beamsplitter. S is the sensitivity of the photodiode in A/W at the wavelength of the light transmitted by the laser [13].

$$I = P \cdot S = 2.5 \cdot 10^{-3} \cdot 0.43 = 1.1 \; mA \tag{3.8}$$

All amplifiers will be working on 5V. Although rail-to-rail op-amps can be used, we want to keep a margin of 0.5 V from the supply rail. Furthermore the amplifiers will work with a reference voltage of 2.5V. This means that the maximum output voltage of the op-amps should be 2V by design. Using Equation 3.9 the transimpedance gain, G, is calculated based on the output current of the photodiode, I_{in}, and the maximum output voltage of the op-amp V_{out}.

$$G = \frac{V_{out}}{I_{in}} = \frac{2}{1.1 \cdot 10^{-3}} = 1860 \; \Omega \tag{3.9}$$

The schematic of the transimpedance amplifiers we will be using is shown in Figure 3.5. The transimpedance gain of this amplifier is determined by the feedback resistor. Actually the transimpeadance gain is equal to the value of the resistor. The transimpedance gain we calculated was 1860 Ω, therefore a resistor of 1.5 $k\Omega$ will be used as feedback resistor, since this is the closest standard value that does not exceed the limit.

The capacitor in parallel with the feedback resistor prevents the amplifier from being unstable by increasing the gain margin of the transimpedance amplifier [3]. It does however also act as a low-pass filter and therefore the capacitance should not be too high. Therefore we choose to use a value of 470 pF.

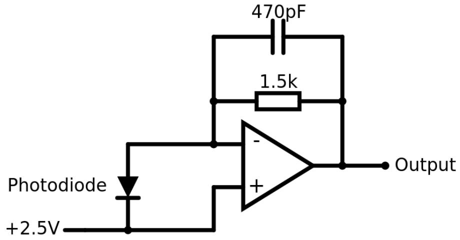


Figure 3.5: The schematic of the transimpedance amplifier.

There are multiple op-amps to choose for the transimpedance amplifier, a summary is given in Table 3.3. It is important that the amplifier can operate under a single 5V supply and that the output is near rail-to-rail. Furthermore we want the current bias, voltage offset and the equivalent input voltage en current noise as low as possible. Comparing those values, the MAX4475AUT would be the best fit to be used for the transimpedance amplifier.

Type	Vout	I bias (pA)	I offset (pA)	V offset (uV)	eq.input V noise (nV/$\sqrt{Hz}$)	eq. input I noise (fA/$\sqrt{Hz}$)	GBW (MHz)	
AD8541ARTZ	R2R	4	0.1	1000	42	100	1	[5]
TS922IPT	R2R	15000	1000	3000	9	-	4	[34]
TS921IDT	R2R	15000	1000	5000	9	-	4	[35]
TSV324IPT	R2R	70000	3000	200	27	-	1.4	[36]
OPA2333AIDR	R2R	70	140	2	-	100	0.35	[21]
OPA2211AIDDA	R2R	60000	25000	50	2	3300	80	[20]
MAX4012EUK	R2R	5400	100	4000	10	1300	-	[29]
MAX4475AUT	R2R	1	1	70	4.5	0.5	42	[26]
MAX44251AKA	R2R	200	400	3	6.2	300	10	[25]
MCP6022T	R2R	1	1	500	8.7	3	10	[30]

Table 3.3: Comparison between multiple op-amps.

3.5.3. DC Blocking

After the current from the photodiode has been amplified and converted into a voltage the offset has to be removed, since we are not interested in the steady state level of the light intensity. The DC blocking is done using a capacitor and placing it in series with the output of the transimpeadance amplifier. To make sure that the side of the capacitor that is not connected to the output of the transimpadance amplifier is balanced around the 2.5 V reference voltage, a resistor is used. This makes the DC blocker into a high-pass filter. The schematic of this is shown in Figure 3.6.

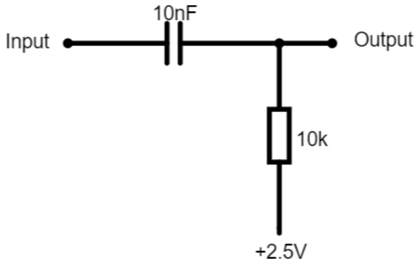


Figure 3.6: The schematic of the DC-blocker, which basically is a RC high-pass filter.

As discussed in section Section 3.3.1, minimum frequency that the amplifiers can amplify should not be too low. Since the DC-blockers are high pass filters, they do limit this frequency. To prevent them from attenuating low frequencies too much, the values of the resistor and the capacitor should be high enough. Furthermore it is important that in the voltage amplifiers that will be discussed in Section 3.5.4, two extra DC-blockers will be used. These will have an influence as well and therefore have to be accounted for when calculating the -3 dB point. The -3 dB frequency of these three high-pass filters together is given by Equation 3.10. The derivation of this equation is shown in Section A.2.

$$f_{-3dB} = \frac{1}{2\sqrt{5}\pi RC} \tag{3.10}$$

Based on the cutoff frequency determined in Section 3.3.1 the minimum value of $R{\cdot}C$ can be calculated, however this does not fix the ratio between the resistance and the capacitance. To do this, the value of the resistor has to be considered. The thermal noise created by the resistor is directly proportional to the resistance, according to Equation 3.11 [31].

$$V_{RMS} = 4kTRB \tag{3.11}$$

According to Equation 3.11, R should be as low as possible, however the op-amp before the DC-blocker should not be loaded to much. Because this will increase the total harmonic distortion and decrease the rail-to-rail characteristics of the op-amp. Based on the datasheets [26] [20] of the op-amps we will be using we decided to use resistors of 10 $k\Omega$ to make sure the output current would stay well below the current the op-amp is designed for. Based on Equation 3.10, the chosen resistance and a frequency of 1 kHz, the value for the capacitance is calculated. This is done in Equation 3.12.

$$C = \frac{1}{2\sqrt{5}\pi R f_{-3dB}} = \frac{1}{2\sqrt{5}\cdot\pi\cdot 1\cdot 10^4\cdot 1\cdot 10^3} = 7.1\ nF \tag{3.12}$$

Based on this we will use a value for C of 10 nF, since this is the closes standard value that is higher than the value we calculated.

3.5.4. Voltage Amplifier

Now that the signal current from the photodiodes has been converted into a voltage and the DC offset has been removed, the signal voltage has to be amplified. For the comparator to work the signals from the detectors should be amplified until their amplitude is approximately in the same range as the supply voltage.

To be able to design the amplifiers, the first step is determining the amplification that is needed. To do this the signal strength generated by the photodiodes due to the quantum effect we are trying to measure has to be calculated. This is done by calculating how many photons reach the detector based on the power transmitted by the laser. Based on the number of photons and the sample frequency defined in the requirements (Chapter 2) the standard deviation of the received number of photons can be calculated and therefore the root mean square of the power generated by these fluctuations. These calculations are explained in more detail in A.3 and the MatLab script used the perform these calculations can be found in Section A.4. Table 3.4 shows the result of the calculations.

Amount of photons	7.99e+10
Photon variance	2.00e+10
RMS Power	4.42 nW
RMS Current	1.90 nA

Table 3.4: Results of the Matlab calculation.

The power and current values we calculated are due to the photon fluctuations and do therefore not include the DC offset, this is exactly what we want however, since we do not amplify this offset. To obtain the RMS input voltage of voltage amplifiers, we multiply the output current of the photodiode by the transimpedance gain of the transimpedance amplifier. This is shown in Equation 3.13 and results in a voltage of 2.85 nV.

$$V_{RMS_{in}} = I_{RMS_{PD}} \cdot G = 1.90 \cdot 10^{-9} \cdot 1500 = 2.85\ \mu V \tag{3.13}$$

To calculate the amplification we also need to determine the output level of the amplifiers. The output level can be determine based on that the output should not saturate the outputs of the op-amps. Since the signal is random however, the level at each point in time will follow a Gaussian distribution [28], which means that there will always be a small chance that the output is saturated. The smaller the RMS voltage of the signal the smaller the chance of the output being saturated at any time. The chance of the output clipping at any point in time was chosen to be less than 0.1 % . We know that the signal will be lower than 3σ 99.9% of the time, since we know that the op-amp saturates when the amplitude is higher than 2 V, we can use this to calculate RMS voltage the signal should have.[4]. This is done in Equation 3.14. The RMS voltage we calculated does not include the 2.5 V offset.

$$V_{RMS_{out}} = \sigma = \frac{2}{3} = 666\ mV \tag{3.14}$$

Now that we know the RMS input voltage and RMS output voltage, the amplification of the voltage amplifiers can be calculated. This is done in Equation 3.15 and results in an amplification of 107 dB.

$$A = \frac{V_{RMS_{out}}}{V_{RMS_{in}}} = \frac{666 \cdot 10^{-3}}{2.85 \cdot 10^{-6}} = 2.34 \cdot 10^5 = 107dB. \tag{3.15}$$

Since this amplification is too high to be amplified by a single op-amp, two stages will be used to amplify the signal. This means that each op-amp only has to amplify half of the total amplification, which is 54 dB.

To amplify the signal we will be using two non-inverting voltage amplifiers. The amplification of these amplifiers is determined by the feedback resistor R_f and the resistor connected to the reference voltage R_{in}. The amplification is given by Equation 3.16 [39].

$$A = 1 + \frac{R_f}{R_{in}} \tag{3.16}$$

Based on this we choose to use a value 150 Ω for R_{in} and a value of 100 $k\Omega$ for R_f. Figure 3.7 shows the schematic of the two non-inverting voltage amplifiers. The DC blockers are added to prevent the input offsets of the op-amps from creating a bias in the output.
To decide which amplifier we want to use for the voltage amplifier, we again looked at the comparison in Table 3.3. For the voltage amplifier the most important specifications are the equivalent input noise voltage and the offset voltage, since those reduce the randomness of the output. Therefore we chose to use the OPA221AIDDA.

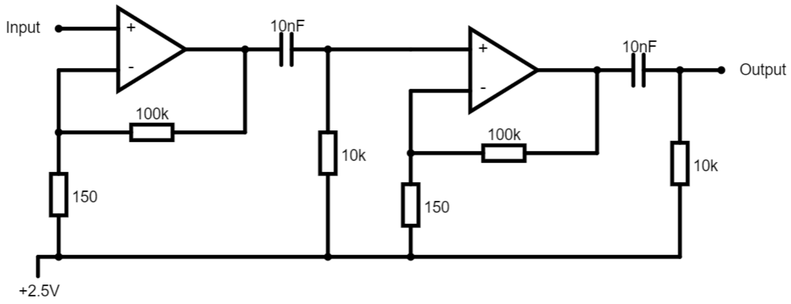


Figure 3.7: The schematic of the voltage amplifier.

3.5.5. Simulation

The check the system up to this point everything from the photodiode until the signal that will be going into the comparator was simulated using LTSpice. To simulate the op-amps, the Spice models provided by the manufacturers were used. To simulate the photodiode, we made use of the equivalent circuit for a photodiode. Two current sources were used for this equivalent circuit, one for the current offset and one that resembles the current created by fluctuations in light intensity. The second one is used as source for the AC analysis.

The schematic used to do the simulations is shown in Figure 3.8. With LTSpice the bodeplot of the system until the comparator is determined. The result is shown in Figure 3.9. Since the bodeplot represents the voltage amplification, the cutoff frequencies are found by plotting a line at -6dB. The cutoff frequencies we found are 1.5 kHz and 100 kHz.

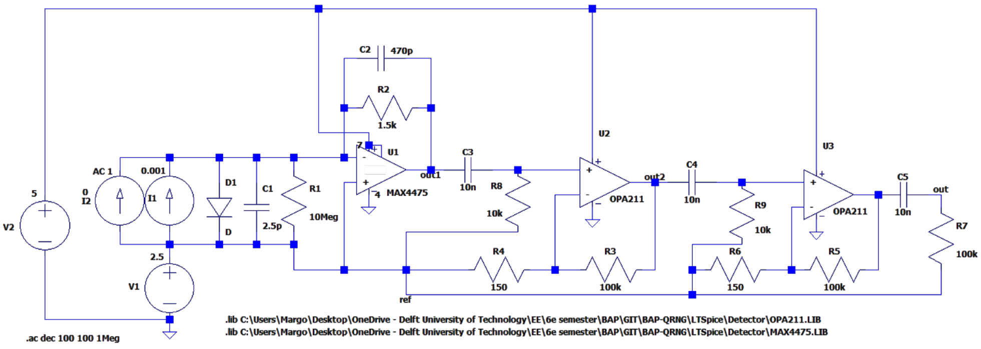

V2
5
AC 1
0
I2
0.001
I1
D1
D
C1
2.5p
R1
10Meg
2.5
V1
C2
470p
R2
1.5k
U1
MAX4475
out1
C3
10n
R8
10k
U2
OPA211
out2
R4
150
R3
100k
ref
C4
10n
R9
10k
U3
OPA211
R6
150
R5
100k
C5
10n
out
R7
100k
.ac dec 100 100 1Meg
.lib C:\Users\Margo\Desktop\OneDrive - Delft University of Technology\EE\6e semester\BAP\GIT\BAP-QRNG\LTSpice\Detector\OPA211.LIB
.lib C:\Users\Margo\Desktop\OneDrive - Delft University of Technology\EE\6e semester\BAP\GIT\BAP-QRNG\LTSpice\Detector\MAX4475.LIB

Figure 3.8: The circuit in LTSpice used for the simulations.

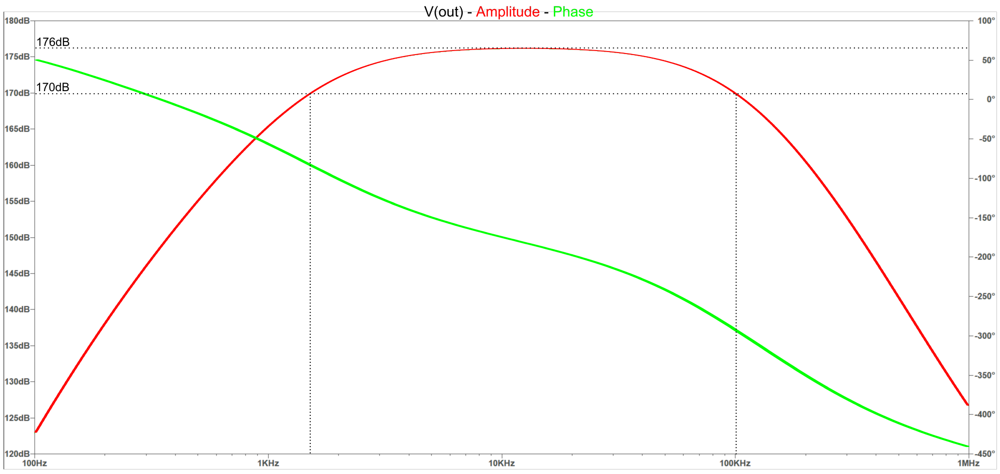


Figure 3.9: The bode plot for the detector circuit

Next to the AC analysis, we also performed a noise analysis using LTSpice. The noise calculated by LTSpice is based on the thermal noise in the resistors and noise specified in the op-amp models. The total RMS noise at the output of the last voltage amplifier was calculated to be 902 mV. This is quite high compared to the signal power we calculated, but is has to be noted that the signal from both detectors will be complementary while the noise will not. Therefore the comparator will increase the accuracy of the measured bits.

3.6. Comparator

The comparator is responsible for comparing the signal from detector 1 to the signal of detector 2. The output of the comparator will be high or low based on which of these signals is the highest.

A comparator can be seen as an amplifier without feedback and a high amplification. This means that theoretically every op-amp which is used without feedback acts as comparator. However op-amps not designed to be used as a comparator will not perform as well as dedicated comparator IC's, especially when they are driven into and out of saturation at high rates [6]. This is why we chose to use a dedicated comparator IC to perform the comparison. Multiple comparators were considered and compared. The most important features we considered to choose between are:

- The rise-time and fall-time
- The input offset
- The power consumption
- The unintentional hysteresis

Based on those features we found 3 good options that are further investigated on propagation time, bias current, offset current and offset voltage. This is shown in Table 3.5. We chose to use the TLV3501AIDG4 comparator, based on the lowest propagation time.

Type	**Rise/fall time (ns)**	**Propagation time (ns)**	**I bias (pA)**	**I offset (pA)**	**V offset (mV)**	
LMV761	1.7	200	0.2	0.001	0.2	[19]
TLV3501AIDG4	1.5	10	2	2	1	[24]
TS3011IYLT	1.1	16	1	1	0.4	[37]

Table 3.5: Comparision between different comparators.

3.7. Microcontroller

The task of the microcontroller is to read out the signal from the comparator and based on this generate random bits. Next to generating these bits, the microcontroller will perform more tasks like converting the bits into ascii characters and generating security keys based on this, however this part of the microcontroller code is not considered part of the detection system and is therefore not treated in this section.

3.7.1. Hardware

The microcontroller will measure the output of the comparator to determine the polarity of the random bits. To prevent the comparator from changing while the microcontroller is measuring the signal, a flip-flop is used. The microcontroller will create a clock signal for the flip-flop. On the rising edge of the clock signal the flip-flop will set its output value according to the output state of the comparator at that time. The the microcontroller will measure the state of the flip-flop at the falling edge of the clock signal to make sure that the flip-flop has reached a steady state output. Since the microcontroller works on 3.3V, the output of the flip-flop will be measured using a 5V tolerant pin of the microcontroller. This means no logic level conversion from 5V to 3.3V is necessary.

3.7.2. Software

The part of the software of the microcontroller that is responsible for reading out the state of the flip-flop and keeping track of the random bits is bundled together in a class called 'RandomBitExtractor'. The code of this class, like all code for the microcontroller, is written in C++. The code can be found in Section A.5.

The code for the detection system will perform the following tasks:

- Set up a timer and create a clock signal for the flip-flop based on this timer.
- The same timer used for the clock signal should also trigger a interrupt at the falling edge, which measures the current state of the flip-flop.
- Keep small buffer of random generated bits. This makes processing the bits to create random characters less complicated, since 8 bits can be read from the buffer at the same time.

Generate clock signal
To generate the clock signal a hardware timer of the microcontroller is setup to run at the desired frequency. This timer is then mapped to a digital pin by setting the pin to low when the timer overflows and setting the pin to high when the timer reaches half of its overflow value.

Read out input data
To readout the state of the flipflop, a overflow trigger is set up based on the timer used to generate the clock signal. This trigger is connected to a callback function in which the state of the flip-flop is measured. Since the output signal of the clock is set to low when the timer flows over, the state of the flip-flow will always be measured on a falling edge.

Buffer with random generated bits
The buffer is implemented by creating an array of bits and keeping track of which bits are valid and which bits are used by means of a buffer pointer. This buffer pointer is an integer which stores the value of the highest index of a bit that is still valid. Every time a bit has to be added to the buffer, the pointer is increased and element of the array with this new index value is overwritten. When a bit is read, the bit to which the buffer pointer points is read and the buffer pointer is decreased.

By implementing the buffer like this, the value of the buffer pointer will always be the same as the number of available bits in the buffer. This means that when multiple bits have to be read, using the buffer pointer it can first be checked whether or not enough bits are available. This simplifies the implementation with the rest of the system.

3.8. Full System Schematic

The circuit for the full detection system is found in Figure 3.10.

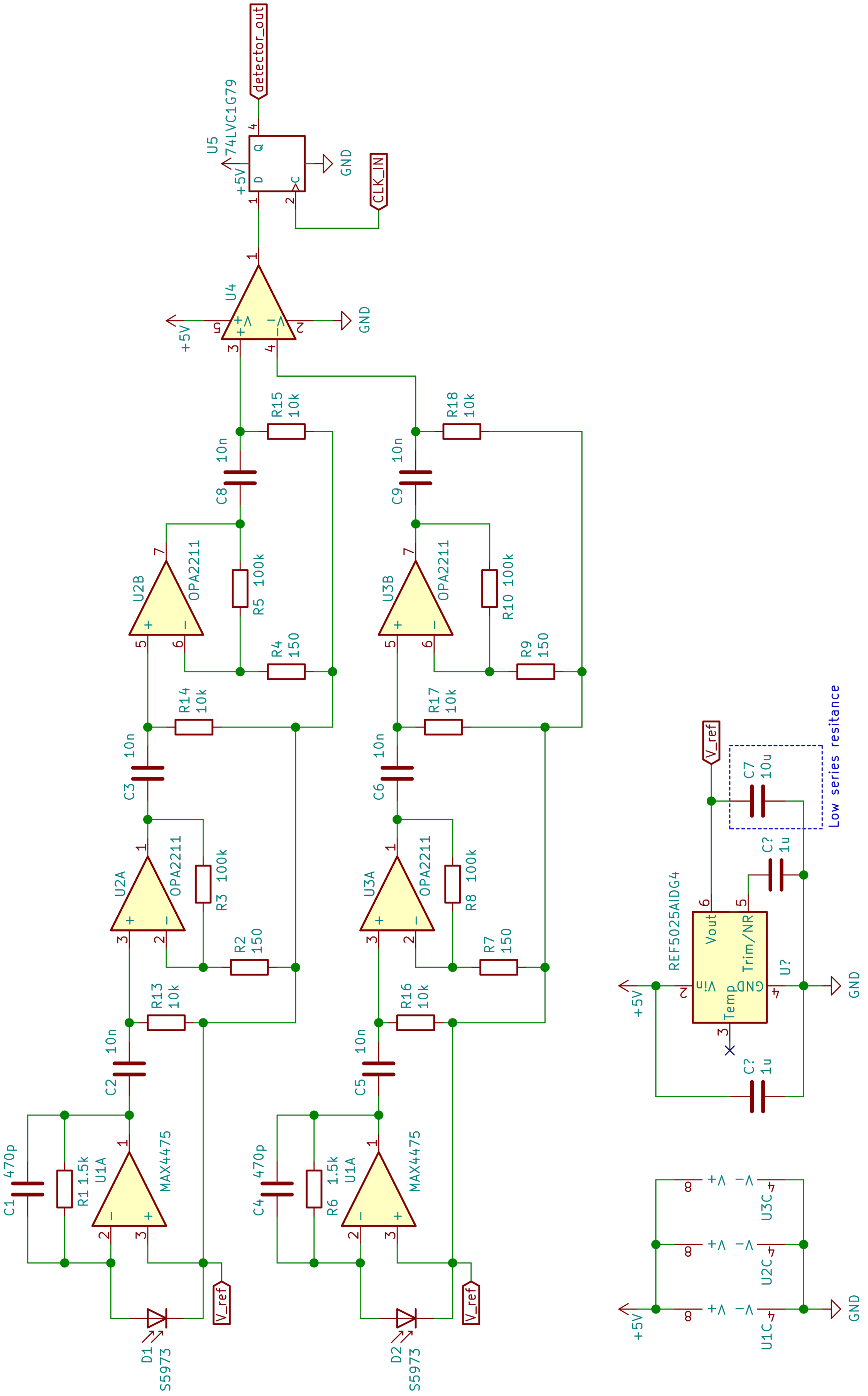


Figure 3.10: The full schematic of the detection system.

4
Enclosure

4.1. Introduction

As stated in the requirements (Chapter 2), the enclosure is necessary to physically protect the system to prevent the internal components of the QRNG from breaking or damaging. Furthermore the enclosure can be used to shield the internal electronics from environmental forces. This chapter will discuss the design and design choices we made for the enclosure.

4.2. Design Choices

The first step for the enclosure was to choose the material out of which we would make it. The important factors for the material are:

- The material should be robust and protect the system.
- The material is not allowed to interfere with the system or conduct electric signals.
- The material should be cheap and durable.

We found a few materials that satisfy all requirements, for example wood, plastic, rubber and polystyrene. From these options we found plastic to be the best option, because it can easily be made in the desired shape with a 3D printer. This is useful, since we already had some experience with designing and printing objects with a 3D printer.

The design will be done using the CAD program Fusion360 from Autodesk [2], since we are most familiar with this software and it offers a free educational license. The design was made based on the CAD models and technical drawings made available by the suppliers of the components. Assembling all these components gave us an idea about the shape and the size the enclosure should have. A render of the assembled components can be seen in Figure 4.1.

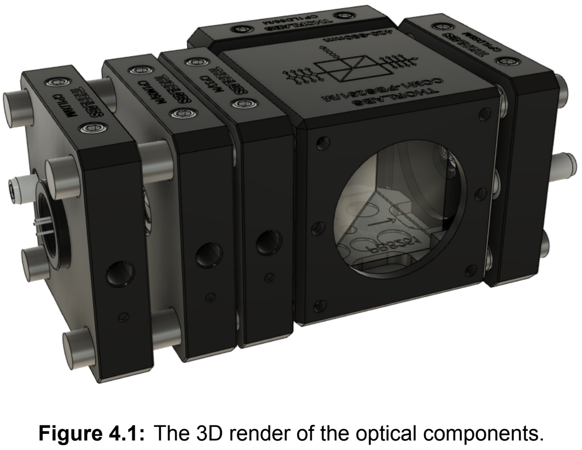


Figure 4.1: The 3D render of the optical components.

Since the system should be portable, we made the system as small as possible while maintaining thick enough walls to keep the enclosure rigid. This lead us to the design shown in Figure 4.2. The size of the design is 17 x 13 x 4.5 cm, which is well below the specified requirements (Chapter 2) and about the size of a lunch-box.

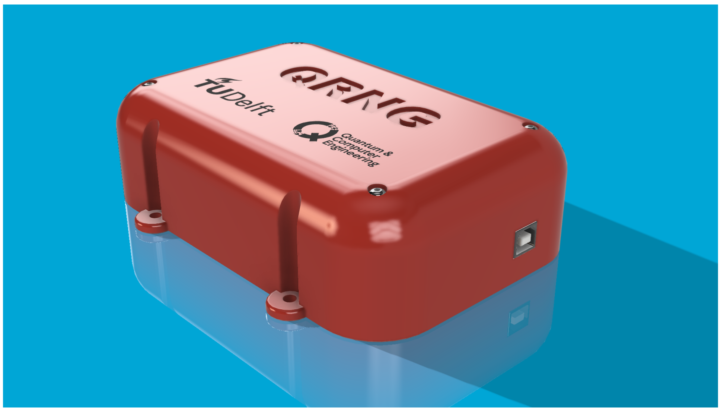


Figure 4.2: Render of the top side of the enclosure.

A trade-off requirement for the enclosure is to maximize the shielding of the systems electronics from environmental noises sources. To do this the enclosure is designed to fit a thin sheet of a conductive material in its shell. By doing so we create a fully enclosed conductive shell around the internal components. Since the shell is conductive, the electromagnetic field inside the enclosure is reduced. A good conductive and flexible material that we will be using for this is aluminium foil.
By splitting the enclosure into an inside part and an outside part and basically sandwiching the aluminium foil in between those parts, also the pars on the inside can not accidentally be shorted by touching the foil. A render of how this will look can be seen in figure Figure 4.3.

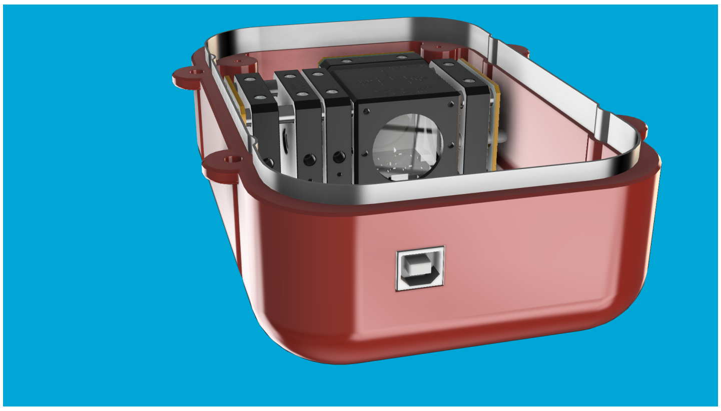


Figure 4.3: Render of the enclosure with shielding foil.

4.3. CAD Design

As stated, the enclosure is designed using CAD. This allows us to carefully inspect and adjust the model according to our needs before actually producing the enclosure.

5

Prototype Implementation

5.1. Introduction

This chapter is about the process of testing, implementing and validating the designs for the detection system and the enclosure.

For the detection system, the components were first tested on breadboards to confirm that they worked as we expected. Next they were implemented by soldering the components on the PCBs designed by the other subgroup and finally the successful implementation of the system is verified.

The enclosure is implemented by 3D-printing the designed parts and assembling all parts together.

5.2. Detector

5.2.1. Testing

The different parts of detector system are first tested on breadboards to check if they are working as we intended before ordering the final PCBs.

Voltage reference

The first part that was tested was the circuit to create a stable 2.5 V that would be used as reference. The input of the voltage regulators is connected to a 5V power supply and the output was measured with an oscilloscope. The output was a constant 2.5 V signal, as it is supposed to be. Figure 5.1 shows the circuit assembled on the breadboard.

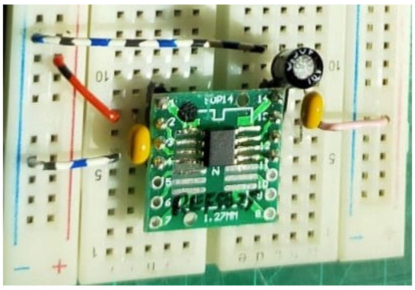


Figure 5.1: Testing the voltage reference driver on a breadboard.

Stage 1: Photodiode, transimpedance amplifier and DC blocker

The next test was to test if the photodiode works properly. Since the output of the photodiode is a small current which we can not measure, we decided to test the photodiode, transimpedance amplifier and DC-blocker at the same time. The transimpedance amplifier is connected to a 5V power supply and the reference voltage of 2.5V from the voltage driver.

We started by measuring the output of the transimpedance amplifier while the photodiode was blocked, so almost no light was absorbed by the photodiode. Before the DC blocker we measured an average voltage of 2.6 V, which is slightly higher than the 2.5 V reference. This slight difference is likely caused by the offset current of the amplifier, but should not be a problem, since the offset will be filtered out by the DC blocker.

Unfortunately we were not able to test this setup in combination with the laser, since the laser was not available at the time. Therefore we used a flashlight to illuminate the photodiode to test whether or not the circuit worked. Using the oscilloscope we confirmed that the output level of the transimpedance amplifier changed based on how much the photodiode was illuminated. We were not able to test the amplification jet since the oscilloscope is not capable of measuring the input signal and we do not know the precise power radiated by the flashlight. The implementation of this circuit on the breadboard can be seen in Figure 5.2.

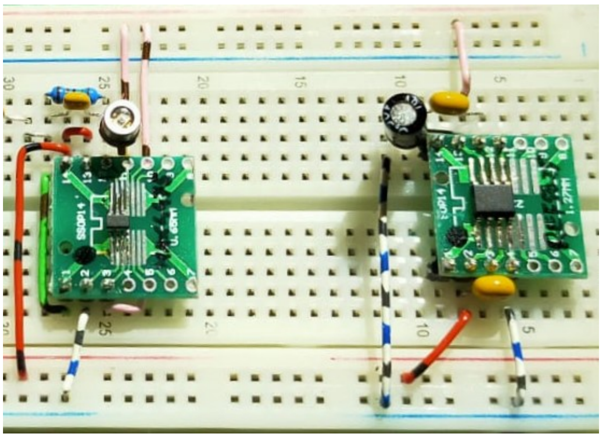


Figure 5.2: Testing the photodiode and transimpedance amplifier on a breadboard.

Stage 2: Voltage amplifiers and DC blockers
In the next test we will amplified the signal with the voltage amplifiers and measured the output after the DC blockage. Again we started measuring the output signal while the photodiode was covered. The setup is shown in Figure 5.3.

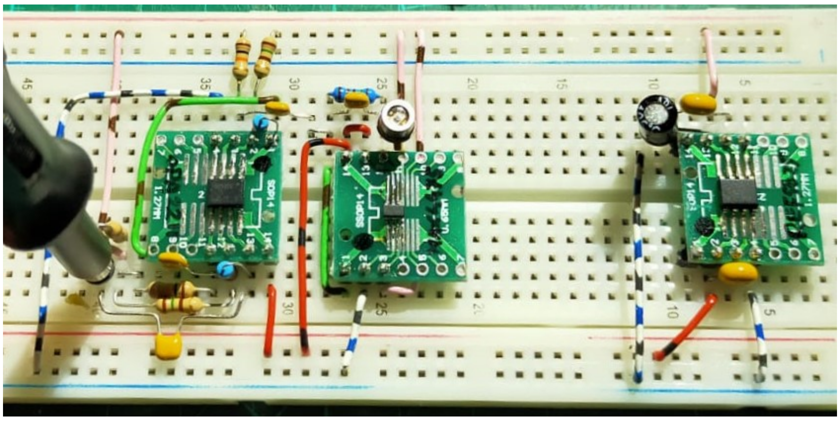
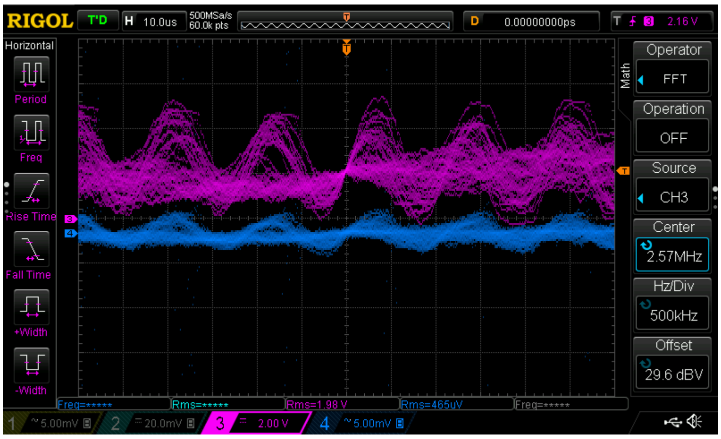


Figure 5.3: Testing the voltage amplifiers on a breadboard.

Without any illumination on the photodiode, we measured the output of the last voltage amplifiers, which is the signal that will eventually be compared by the comparator. The amplitude of the noise we measured was quite a bit higher than the noise we expected based on the simulation. The signal can be seen as the purple signal in Figure 5.4. The two main causes of this are probably:

- The noise contained some kind of interference of a sinusoidal signal of approximately 50 kHz. When we increased the time per division of the oscilloscope, we noticed that this inferference was not constantly present but only for short times that appeared at an interval of 100 Hz. Therefore we expect this interference to be cause by some kind of switching power supply.
- Another reason the noise is probably higher than we anticipated is because the circuit was tested on a breadboard and therefore the distance between components was relatively long. This makes the system extra sensitive to interference from the environment.



H 10.0us
500MSa/s
60.0k pts
D 0.00000000ps
T 3 2.16 V
Operator: FFT
Operation: OFF
Source: CH3
Center: 2.57MHz
Hz/Div: 500kHz
Offset: 29.6 dBV
Freq=***** Rms=***** Rms=1.98 V Rms=465uV Freq=*****
1 ~5.00mV 2 ⎓20.0mV 3 ⎓ 2.00 V 4 ~5.00mV

Figure 5.4: The noise present at the output of both voltage amplifiers. In blue the output of amplifier 1 and in purple the output of amplifier 2.

Because of these two points we are not immediately alarmed by the fact that the noise is so high, but were more interested in the fact that the voltage amplifiers were working because how much they amplified the noise amplitude is constant with the amplification we calculated. Figure 5.4 shows the output signal of the first voltage amplifier and the second voltage amplifier. Based on the amplitudes in this figure, the amplification of the second amplifier is verified.

The Flip-Flop

In order to test if the flip-flop was working correctly, we connected it to the power supply, used the microcontroller to generate a clock and connected a signal generator to its input. With a oscilloscope we simultaneously measured the sinusoidal input, the clock signal and the output of the flip-flop, the result can be found in Figure 5.5. Based on this measurement we were able to verify that the flip-flop was working correctly.

5.2.2. Implementation

For the implementation of the detection system, all components were soldered on the PCBs designed by the other subgroup. Everything was soldered by hand. We did a continuity check using a multi-meter to make sure that everything was connected correctly. During this check we found a few problems that we probably created while entering our schematics into the PCB design software. These mistakes were fixed by breaking the copper tracks and manually reconnecting the wires using a small piece of wire. The results of the implementation of the detection system can be seen in Figure 5.6.

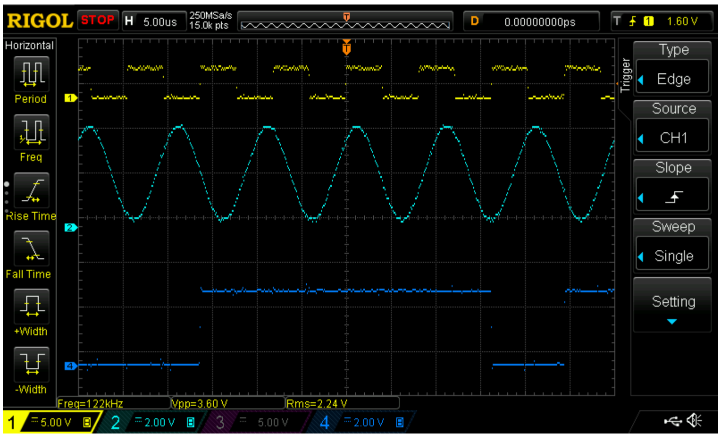
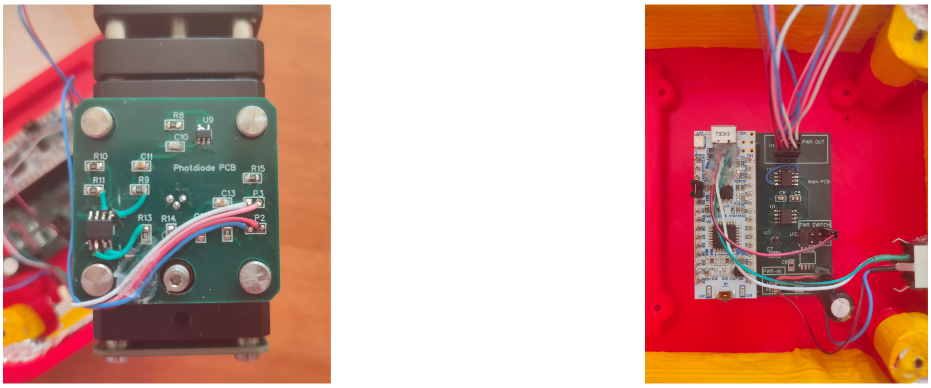


Figure 5.5: Measurement results for flip-flop test. Light blue is the sinusoidal input, yellow is the clock signal and dark blue is the output of the flip-flop.



(a) The PCB of the photodiode, amplifiers and DC blockers.

(b) The PCB with the comparator, flip-flop and microcontroller.

Figure 5.6

5.2.3. Validation results

Now that all components of the detection system have been tested and the complete system has been implemented, the system as a whole can be tested together with the other part of the optical setup. Unfortunately however at the time of writing this report, we have not jet been able to do so. So this will be our main task the following weeks.

5.3. Microcontroller Code

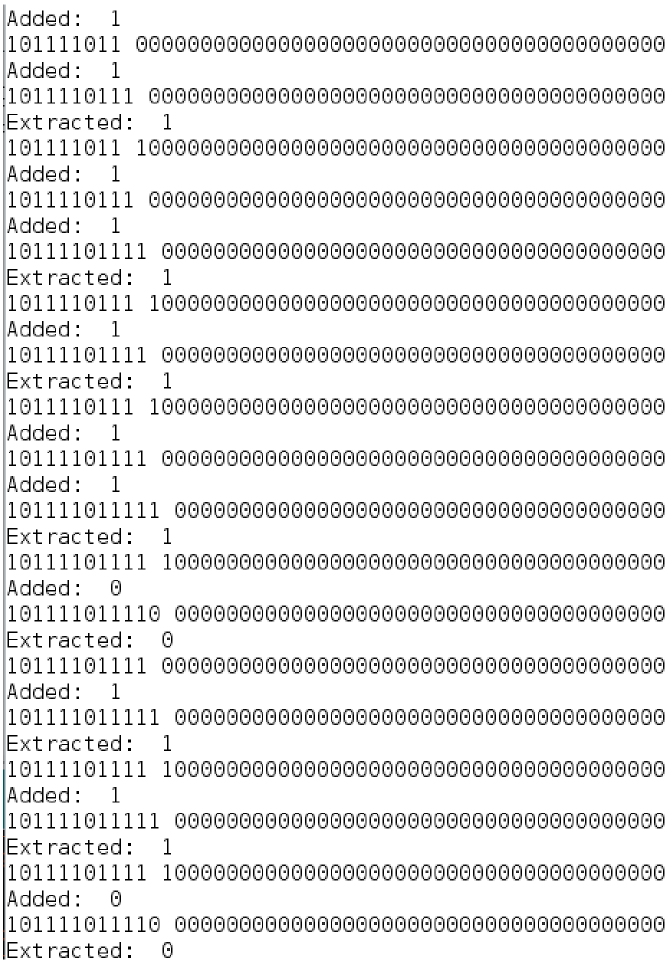
5.3.1. Implementation

The microcontroller code is written in the Arduino IDE [1] and compiled using STM32Duino [10]. The code of the microcontoller that was written for the detector can be found in Section A.5.

5.3.2. Testing

To test the microcontroller code we started by only uploading the part of the code dedicated to generating the clock for the flip-flop. To make sure the measurements are indeed done at the falling clock edge, the callback function was slightly modified to generate a spike. By measuring the output pins with an oscilloscope we confirmed that the clock output was working as expected, and the measurements were indeed performed at the falling edge of the generated clock.

The next thing we tested was the buffer used to store the random generated bits. To do this we made the microcontroller extract bits form the buffer at irregular intervals and print them to the serial monitor. Besides that we printed lines that explained what was going on and the complete content of the buffer. Part of the output of the serial monitor is shown in Figure 5.7. During testing we set the buffers length to 50 since this makes it easier see what is going on, however during normal operation the length can be set much higher. What the length should be depends on how many bits are expected to be read at once.



```
Added:  1
101111011 00000000000000000000000000000000000000000
Added:  1
1011110111 0000000000000000000000000000000000000000
Extracted:  1
101111011 10000000000000000000000000000000000000000
Added:  1
1011110111 0000000000000000000000000000000000000000
Added:  1
10111101111 000000000000000000000000000000000000000
Extracted:  1
1011110111 1000000000000000000000000000000000000000
Added:  1
10111101111 000000000000000000000000000000000000000
Extracted:  1
1011110111 1000000000000000000000000000000000000000
Added:  1
10111101111 000000000000000000000000000000000000000
Added:  1
101111011111 00000000000000000000000000000000000000
Extracted:  1
10111101111 100000000000000000000000000000000000000
Added:  0
101111011110 00000000000000000000000000000000000000
Extracted:  0
10111101111 000000000000000000000000000000000000000
Added:  1
101111011111 00000000000000000000000000000000000000
Extracted:  1
10111101111 100000000000000000000000000000000000000
Added:  1
101111011111 00000000000000000000000000000000000000
Extracted:  1
10111101111 100000000000000000000000000000000000000
Added:  0
101111011110 00000000000000000000000000000000000000
Extracted:  0
```

Figure 5.7: The output of the serial monitor while testing the buffer used to store the measured bits. The space is printed after the bit to which the buffer pointer is currently pointing.

5.4. Enclosure

The enclosure was designed to be 3D-printable, so this is also how we produced it. The enclosure was printed with PLA since this is a very rigid material and one of the easiest materials to print. Once both the inside and the outside were printed, the aluminium foil was wrapped around the inside and folded around the edges to make sure that the aluminium foil made good contact with the aluminium foil of the lid. Next the inner part was carefully pushed into the outer part and the four hex screws at the top were screwed in to fix the inner and outer part together.

Next all components of the QRNG were fitted inside the enclosure to make sure that everything fitted as intended, which turned out to be the case. The results can be seen in Figure 5.8.

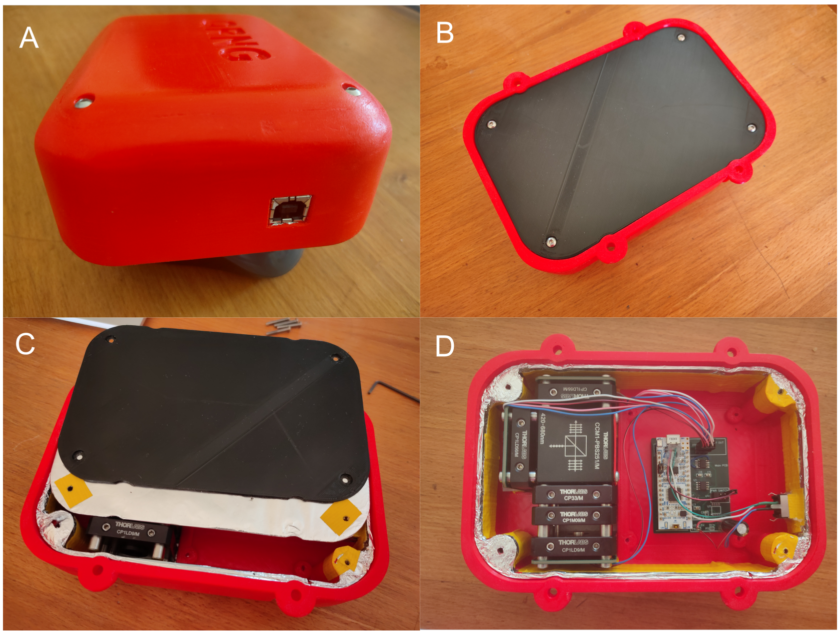


Figure 5.8: Photos of the enclosure. Photo C clearly shows the shielding and photo D shows how all the components fit inside the enclosure.

6 Results

6.1. Detection System

All subsystems of the detection system have been tested and confirmed to work. However it must be noticed that the measured noise at the output of the voltage amplifiers is higher than we would like. Whether or not this noise is also apparent int the final setup has still to be tested.

If we look at the requirements we can conclude that most of the requirements have already been met by design, these are:

- The design does only use off-the shelf components.
- The design fits in the enclosure.
- The detection circuit is safe to handle, since it uses no voltages or laser powers that can do much harm.
- The data is stored in a buffer and can easily be delivered to a PC.
- The detection system works on 5 V.

The other requirements still have to be confirmed by testing the complete system.

6.2. Enclosure

In contrast to the detection system, the enclosure is finished. (Appart from the stickers of the TUDelft logo and the QCE logo that still have to be added.) We designed and implemented an enclosure that protects the QRNG and fits perfectly around all the components and holds the components in place. The enclosure isolates the internal voltages since it is made from plastic and all conducting components such as the USB connector and the screws are connected to the ground. We have no way to test the lifespan of the enclosure, but if feels sturdy and we do not think it will break any time soon. The size of the enclosure is 17X13X4.5 cm and therefore meets the requirements. However the enclosure could be designed smaller, we still implemented a room for a larger PCB or an extra battery in case we found this to be necessary at a later stage. The weight of the enclosure is 270 g which is well below the limit set by the requirements. We can not test the transmission of electromagnetic waves, but when we will test the complete system, we can also test the effect of the enclosure on the measurements.
And at least we also think the enclosure just looks nice.

7 Conclusion

In conclusion the detector system is implemented and the enclosure is finished. Although we still have to test the detection system we are satisfied with the progress we have made and the results we have already achieved. We have made significant steps to designing and make an affordable QRNG based on off-the-shelf components. Based on the noise amplitudes we expect to get however, true randomness of the output can not be guarantied. By designing and implementing the enclosure we made sure that the QRNG is not only portable but can actually be carried around without accidentally damaging the internals.

7.1. Recommendation and Future Work

Our biggest issue during the project was the short amount of time we had. In the beginning of the project we were good on track and even ahead of the planning. However when the delivery of components got delayed, our project also got delayed. During this ”waiting” time, we worked as much as possible on the project and even executed some additional work, but because of this we were not able to continue with the main tasks that still had to be done. And in the end this caused that we still have to test the detection system implementation on the PCBs. In the future this could be avoided by ordering the parts as soon as possible and ordering spares for parts that easily break, since we also had to wait on a replacement for a part that broke during the testing of the sub-systems on the breadboards.

While writing the thesis, we found that a few assumptions and calculations we made were incorrect. We adapted the assumptions and calculations but we were not able to correct all the choices we made based on them, which resulted in the final prototype not always being based on the optimal choices. For example while calculating the gain for the amplifiers, we forgot to calculate the noise and only looked at the signal. As a consequence the signal output might in the end turn out saturated more often than we anticipated. A solution to this is to change the resistors of the non-inverting voltage amplifiers and by doing so reduce the total gain.

Regarding future work, the most important thing that still has to be done is the testing of all parts of the detection system combined. This is also the first thing we will do. When this turns out to be successful, we would also like to further optimize the system. By for example performing optimization algorithms on the measured bits or by trying to increase the signal to noise ratio.

Furthermore we can test the performance of the enclosure based on weight, rigidity and electromagnetic wave transmission.

Other future plans are to compare the final QRNG prototype we have made with other existing RNGs, such as CMOS based RNGs and other QRNGs. Moreover since a QRNG basically is a 1 qubit quantum computer, it would be meaningful to search for ways to extend our QRNG to a quantum computer and what would be required to do so.

A
Appendix

A.1. Calculations on detection distributions

The number of photons measured by each source follows a binomial distribution according to Equation A.1 and Equation A.2.

$$\theta_{D1} = \sum_{n=0}^{N} X_n \quad \text{(A.1)} \qquad \theta_{D2} = \sum_{n=0}^{N} (1 - X_n) \quad \text{(A.2)}$$

Since every photon that is received by detector 1 will not be received by detector 2, Equation A.3 holds.

$$\theta_{D2} = \sum_{n=0}^{N} (1 - X_n) = N - \sum_{n=0}^{N} X_n = N - \theta_{D1} \tag{A.3}$$

This means that we can rewrite the probability $P(\theta_{D1} < \theta_{D2})$ according to Equation A.4.

$$P(\theta_{D1} < \theta_{D2}) = P(\theta_{D1} < N - \theta_{D1}) = P(\theta_{D1} < 0.5 \cdot N) \tag{A.4}$$

Since θ_{D1} follows a binomial distribution, the change of θ_{D1} being less than half of the number of experiments is equal to 0.5, as long as the probability of each individual experiment is 0.5 and the number of experiments is large enough to be able to neglect the change that θ_{D1} and θ_{D2} are exactly the same.

A.2. DC blocking cutoff frequency

For the DC blocking schematic

$$\frac{V_o}{V_i} = \frac{sRC}{1 + sRC} \tag{A.5}$$

Since each detector has 3 DC blocks, the -3dB is calculated by

$$\left| \frac{V_0}{V_i} \right|^2 = \left| \frac{sRC}{1 + sRC} \right|^2 = \left(\frac{1}{2} \right)^3 \tag{A.6}$$

$$\left(\frac{\omega RC}{\sqrt{1 + \omega^2 R^2 C^2}} \right)^2 = \frac{1}{6} \tag{A.7}$$

$$\frac{\omega^2 R^2 C^2}{1 + \omega^2 R^2 C^2} = \frac{1}{6} \tag{A.8}$$

$$\frac{1}{\omega^2 R^2 C^2} = 5 \tag{A.9}$$

$$\omega^2 = \frac{1}{5 R^2 C^2} \tag{A.10}$$

$$\omega = \frac{1}{\sqrt{5}RC} \tag{A.11}$$

$$f_c = \frac{1}{2\sqrt{5}\pi RC} \tag{A.12}$$

A.3. Calculation for the RMS output current of the photodiode

The amount of photons that reach the detector is calculated by

$$N = \frac{P_L/E}{F} \tag{A.13}$$

where P_L is the power of the laser received by the photodiode namely 2.5 mW, F is the frequency of the measurements namely 100 kHz and E is the energy per photon which is calculated by

$$E = \frac{h \cdot c}{\lambda} \tag{A.14}$$

where h is planks constant, c is the speed of light and λ is the wavelenght of light emitted by the laser namely 635 nm.

The standard deviation of the binomial distributed is proportial to the RMS power generated by the fluctuations of photons. This is calculated by

$$SD = \sqrt{Np(1-p)} \tag{A.15}$$

$$P_{RMS} = E \cdot SD \cdot F \tag{A.16}$$

The output current of the photodiode is therefore

$$I_{RMS} = \textit{sensitivity} \cdot P_{RMS} \tag{A.17}$$

where the sensitivity of the photodiode is 0.43 A/W.

A.4. Matlab code to calculate output of photodiode

```matlab
% inputs:
FREQUENCY = 100e3;  % frequency of measurements (Hz)(1/s)
POWER = 2.5e-3;     % assume a 2.5mW Laser (W)(J/s)
LAMBDA = 635e-9;    % 635nm light (m)
PROBABILITY = 0.5;  % binomial probability (-)
SENSITIVITY = 0.43; % photodiode sensitivity (A/W)

% constants:
c = 299792458;      % speed of causality (m/s)
h = 6.62607015e-34; % Planck constant (J/s)

% calculations:
E = h*c/LAMBDA;     % energy of photon (J)
R = POWER/E;        % photon Rate (Hz)(1/s)
n = R/FREQUENCY;    % number of photons per measurement (-)

p = PROBABILITY;
var = n*p*(1-p);    % the variance of the number of photons received by a detector
SD = sqrt(var);     % standard deviation of the number of photons

% The next step is allowed because we can assume that the mean of the
% difference is equal to 0:
RMS = SD;           % the RMS value of the difference in photons between measurements (-)

t = 1/FREQUENCY;    % the sample period
P_RMS = E * RMS/t;    % the RMS power caused by the fluctuations in number of photons (W)

I_RMS = SENSITIVITY * P_RMS;      % the RMS current at the output of the PD (A)
```

A.5. Code for the microcontroller

Main sketch

```
/* Name:        Main sketch
 * Author:      Margo Molenaar and Feike Pacilly
 * Date:        17-06-2020
 * Function:    This is the main sketch that has to run on the mircocontroller for
 *              the detection system. Right now, it simply prints random ascii-characters
 *              to the serial monitor.
 */

// Include external files:
#include "config.h"              // Include the configuration
#include "RandomBitExtractor.h"  // Include the RandomBitExtractor class

// Create objects:
RandomBitExtractor RBE;          // Create an RandomBitExtractor object

// Define system variables:
int16_t LifeLedCounter;   // Create an counter to use for the life led

void setup() {
  #ifdef USE_SERIAL
    // Start the serial communication:
    Serial.begin(115200);        // Start the serial communication...
    Serial.println("Serial started");   // ...And confirm it works
  #endif

// Set the frequency of the clock for the flip-flop:
  RBE.set_timer(DETECTOR_CLOCK_FREQUENCY);

  // Define the outputs:
  pinMode(LED_BUILTIN, OUTPUT);      // Set the life led as output
}

void loop() {
// Blink the life led to indicate the system is operational:
  digitalWrite(LED_BUILTIN, LifeLedCounter++ > 0);
// If enough bits are available in the buffer ...
  if(RBE.CharAvailable()){
    #ifdef USE_SERIAL
// ... Generate and print a random character to the serial monitor:
      Serial.write(RBE.ReadChar());
    #endif
  }
}
 // This is the wrapper function for the detector's callbackfunction:
void DetectorClockCallbackWrapper(){
  RBE.ClockPeriodCallback();
}
```

RandomBitExtractor.h

```
/* Name:        RandomBitExtractor.h
 * Author:      Margo Molenaar and Feike Pacilly
 * Date:        17-06-2020
 * Function:    This is the header file of the class that is used to
 *              generate a clock signal, read out bits and read out
 *              axcii characters based on measured random bits.
 */

#ifndef _RANDOMBITEXTRACTOR_
#define _RANDOMBITEXTRACTOR_

#include "config.h"             // Include the config file
#include "Arduino.h"            // Include all standard Arduino ...
    definitions

class RandomBitExtractor{
  public:                       // Predeclare all public functions:
    RandomBitExtractor();
    void set_timer(uint32_t);
    void ClockPeriodCallback();
    uint16_t DataAvailable();
    uint16_t CharAvailable();
    bool ReadBit();
    char ReadChar();
  private:                      // Predeclaire the private variables:
    uint32_t ClockTimerChannel;
    bool RandomBitsBuffer[DETECTOR_BUFFER_LENGTH];
    uint16_t BufferPointer;
};

#endif
```

RandomBitExtractor.cpp

```
/* Name:        RandomBitExtractor.cpp
 * Author:      Margo Molenaar and Feike Pacilly
 * Date:        17-06-2020
 * Function:    This is the header file of the class that is used to
 *              generate a clock signal, read out bits and read out
 *              axcii characters based on measured random bits.
 */

#include "RandomBitExtractor.h"  // Include the header file

// Define the callback wrapper function defined in the main sketch:
extern void DetectorClockCallbackWrapper();
void ClockPeriodCallback();   // Predefine the clock callbackfunction

RandomBitExtractor::RandomBitExtractor(){   // Define the constructor
// Determine the Timer channel that should be used to set-up a
// clock on the defined clock output pin
  ClockTimerChannel = ...
      STM_PIN_CHANNEL(pinmap_function(digitalPinToPinName(CLOCK_OUTPUT), ...
      PinMap_PWM));
```

```
}

// Define the set_timer function in which the timer for the clock is
// set-up
void RandomBitExtractor::set_timer(uint32_t frequency){
// Create a hardware timer object for timer 16
  HardwareTimer *ClockTimer = new HardwareTimer(TIM16);
// Set the timer to PWM mode with a 50% duty cycle and bind the
// callbackwrapper function to the overflow trigger
  ClockTimer->setPWM(ClockTimerChannel, CLOCK_OUTPUT, frequency, 50, ...
      DetectorClockCallbackWrapper);
  #ifdef USE_SERIAL
// Confirm via the serial monitor that the clock started and print
//its frequency.
    Serial.print(”Clock started at ”);
    Serial.print(frequency/1e3);
    Serial.println(”kHz”);
  #endif
  pinMode(DETECTOR_INPUT, INPUT);    // Set the detector pin as input
}
// Define the calback function that is triggered on the timer
// overflow (falling edge)
void RandomBitExtractor::ClockPeriodCallback(){
  // If the buffer is not jet full...
  if(BufferPointer < DETECTOR_BUFFER_LENGTH)
    // read a random digit and add it to the buffer and add 1 to the
    // buffer pointer
    RandomBitsBuffer[BufferPointer++] = digitalRead(DETECTOR_INPUT);
}
// Define a function used to check how many bits are available
uint16_t RandomBitExtractor::DataAvailable(){
  return BufferPointer; // Return the number of available bits
}
// Define a function used to check how many characters
// can be created using the available bits
uint16_t RandomBitExtractor::CharAvailable(){
// Return how many times 8 bits can be read until the buffer is empty
  return BufferPointer / 8;
}

// Define a function that generates characters based on the random
// bits in the buffer
char RandomBitExtractor::ReadChar(){
  uint8_t val = 0;               // Initialize an integer...
  for(int i = 0; i < 7; i++){    // Then for every bit...
    val += ReadBit();    // Fill in the least significant (right) bit
    val = val << 1;      // And shift all bits to the left
  }
  return val;            // Convert the integer to a char and return it
}

// Define a function to read a bit from the buffer:
bool RandomBitExtractor::ReadBit(){
// Wait until a bit is available in the buffer:
  while(!DataAvailable()){delay(1);}
// Then return it and subtract 1 from the buffer pointer:
  return RandomBitsBuffer[--BufferPointer];
}
```

Config.h

```
/* Name:        Config.h
 * Author:      Margo Molenaar and Feike Pacilly
 * Date:        17-06-2020
 * Function:    This is an configuration file, in this file some of the
 *              values that might differ per system or can be tuned for
 *              optimal performance can be adjusted.
 */

// If defined, the serial monitor to is used to print the output data
#define USE_SERIAL

// Define the frequency of the clock generated for the flip-flop
// (in Hz)
#define DETECTOR_CLOCK_FREQUENCY 100e3
// Define the lenght of the buffer used to store random bits (in bits)
#define DETECTOR_BUFFER_LENGTH 500

// Define pin in/output pins:
#define CLOCK_OUTPUT D5
#define DETECTOR_INPUT A0
```

References

[1] Arduino. *Software | Arduino IDE 1.8.15*. https://www.arduino.cc/en/software. 2021.

[2] Autodesk. *Fusion 360 | Integrated CAD, CAM, CAE, and PCB software*. https://www.autodesk.com/products/fusion-360/overview. 2019.

[3] Bonnie Baker. *Transimpedance Amplifier Design | Digikey*. https://www.digikey.nl/nl/articles/how-to-design-stable-transimpedance-amplifiers-automotive-medical-systems. 2017.

[4] Robert Keim - All about circuits. *How Standard Deviation Relates to Root-Mean-Square Values*. https://www.allaboutcircuits.com/technical-articles/how-standard-deviations-relates-rms-values/. 2020.

[5] Analog devices. *AD8541/AD8542/AD8544 Data Sheet*. http://www.farnell.com/datasheets/1423364.pdf. 2011.

[6] Eric Coates - Learn about Electronics. *Op amps and Comparators*. https://learnabout-electronics.org/Amplifiers/amplifiers62.php. 2020.

[7] Wavelenght Electronics. *Photodiode Basics:Selection Operation*. https://www.teamwavelength.com/download/applicationtechnotes/an-ld17.pdf. 2020.

[8] Rudiger Paschotta - RP photonics encyclopedia. *Dark current*. https://www.rp-photonics.com/dark_current.html. n.d.

[9] Rudiger Paschotta - RP photonics encyclopedia. *Phototransistors*. https://www.rp-photonics.com/phototransistors.html. n.d.

[10] GitHub. *Arduino Core for STM32*. https://github.com/stm32duino/Arduino_Core_STM32. 2021.

[11] Circuit Globe. *Difference Between Photodiode Phototransistor*. https://circuitglobe.com/difference-between-photodiode-and-phototransistor.html. n.d.

[12] Hamamatsu. *Datasheet Si PIN photodiodes S1223 series*. http://www.farnell.com/datasheets/3171157.pdf. 2013.

[13] Hamamatsu. *Datasheet Si PIN photodiodes S5971, S5971, S5973*. http://www.farnell.com/datasheets/3171163.pdf. 2019.

[14] M. Herrero-Collantes and J.C. Garcia-Escartin. “Quantum random number generators”. In: *Rev. Mod. Phys.* 89 (1 Feb. 2017), pp. 16–17. DOI: 10.1103/RevModPhys.89.015004. URL: https://link.aps.org/doi/10.1103/RevModPhys.89.015004.

[15] IDQ. *Quantis white paper - Random number generation using quantum physics*. https://marketing.idquantique.com/acton/attachment/11868/f-0227/1/-/-/-/-/Quantum%20RNG_White%20Paper.pdf. 2010.

[16] IDQ. *Quantum Computing Industry Review: Q1 2020*. https://www.idquantique.com/quantum-computing-industry-review-q1-2020/. 2020.

[17] IDQ. *Quantum Random Number Generation*. https://www.idquantique.com/random-number-generation/applications/. 2021.

[18] Texas Instruments. *DREF30xx 50-ppm/°C Max, 50-μA, CMOS Voltage Reference in SOT-23-3 Datasheet*. https://www.ti.com/lit/ds/symlink/ref3025.pdf?ts=1620645834425&ref_url=https%253A%252F%252Fwww.google.de%252F. 2021.

[19] Texas Instruments. *LMV76x and LMV762Q-Q1 Low-Voltage, Precision Comparator With Push-Pull Output*. https://www.ti.com/lit/ds/symlink/lmv762q-q1.pdf?ts=1620203659361&ref_url=https%253A%252F%252Fwww.google.com%252F. 2021.

[20] Texas Instruments. *OPAx211 1.1-nv/√Hz Noise, Low Power, Precision Operational Amplifiers datasheet*. https://www.ti.com/lit/ds/sbos377l/sbos377l.pdf?ts=1620652941993&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FOPA211. 2021.

[21] Texas Instruments. *OPAx333 1.8-V, microPower, CMOS Operational Amplifiers, Zero-Drift Series datasheet*. https://www.ti.com/lit/ds/symlink/opa2333.pdf?ts=1620211147302&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FOPA2333. 2021.

[22] Texas Instruments. *REF50xx Low-Noise, Very Low Drift, Precision Voltage Reference Datasheet*. https://www.ti.com/lit/ds/symlink/ref5025.pdf?ts=1620722586411&ref_url=https%253A%252F%252Fwww.ti.com%252Fsitesearch%252Fdocs%252Funiversalsearch.tsp%253FsearchTerm%253Dref5025%2526nr%253D567. 2021.

[23] Texas Instruments. *REF60xx High-Precision Voltage Reference With Integrated ADC Drive Buffer Datasheet*. https://www.ti.com/lit/ds/symlink/ref6025.pdf?ts=1620722599650&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FREF6025. 2020.

[24] Texas Instruments. *TLV350x 4.5-ns, Rail-to-Rail, High-Speed Comparator in Microsize Packages datasheet*. https://www.ti.com/lit/ds/symlink/tlv3501.pdf?ts=1620127757684&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FTLV3501. 2021.

[25] MAXIM integrated. *MAX44250/MAX44251/MAX44252 DS*. http://www.farnell.com/datasheets/2001168.pdf. 2015.

[26] MAXIM integrated. *MAX4475 DS*. http://www.farnell.com/datasheets/1913195.pdf. 2012.

[27] R. Ishihara. *BAP project proposal - Mobile Quantum Random Number Generator*. Delft University of Technology: TU Delft, 2021.

[28] Ron Mancini and Bruce Carter. "Op Amp Noise Theory and Applications". In: Elsevier, 2009. Chap. 12. ISBN: 978-1-85617-505-0. URL: https://doi.org/10.1016/B978-1-85617-505-0.X0001-4.

[29] MAXIM. *MAX4012/16/18/20 DS*. http://www.farnell.com/datasheets/28773.pdf. 2004.

[30] Microchip. *MCP6021/1R/2/3/4 Datasheet - Rail-to-Rail Input/Output, 10 MHz Op Amps*. http://www.farnell.com/datasheets/630323.pdf. 2009.

[31] Wendy M. Middleton and Mac E. Van Valkenburg. "Reference Data for Engineers Radio, Electronics, Computer, and Communications". In: Elsevier, 2002. Chap. 34. ISBN: 978-0-7506-7291-7. URL: https://doi.org/10.1016/B978-075067291-7/50036-4.

[32] OSRAM OPTO SEMICONDUCTORS. *Datasheet SFH 213 Radial T1 3/4 Silicon PIN Photodiode*. http://www.farnell.com/datasheets/2711599.pdf. 2018.

[33] OSRAM OPTO SEMICONDUCTORS. *Datasheet SFH 229 Radial T1 3/4 Silicon PIN Photodiode with very short switching time*. http://www.farnell.com/datasheets/2711605.pdf. 2018.

[34] STMicroelectronics. *Datasheet - TS922, TS922A*. https://4donline.ihs.com/images/VipMasterIC/IC/SGST/SGST-S-A0005972394/SGST-S-A0005972394-1.pdf?hkey=6D3A4C79FDBF58556ACFDE234799DDF0. 2018.

[35] STMicroelectronics. *Datasheet TS921 Rail-to-rail high output current single operational amplifier*. http://www.farnell.com/datasheets/1690542.pdf. 2012.

[36] STMicroelectronics. *Datasheet TSV321, TSV358, TSV324, TSV321A, TSV358A, TSV324A,General purpose input/output rail-to-rail low-power operational amplifiers*. http://www.farnell.com/datasheets/1911542.pdf. 2014.

[37] STMicroelectronics. *TS3011 datasheet Rail-to-rail high-speed comparator*. https://4donline.ihs.com/images/VipMasterIC/IC/SGST/SGST-S-A0004145931/SGST-S-A0004145931-1.pdf?hkey=52A5661711E402568146F3353EA87419. 2017.

[38] Inside Quantum Technology. *New IQT Research Report: Quantum Random Number Generators will become a $7.2 Billion Market by 2026*. https://www.globenewswire.com/fr/news-release/2021/01/26/2164365/0/en/New-IQT-Research-Report-Quantum-Random-Number-Generators-will-become-a-7-2-Billion-Market-by-2026.html. 2021.

[39] Electronics Tutorials. *Non-inverting Operational Amplifier*. https://www.electronics-tutorials.ws/opamp/opamp_3.html. 2021.

[40] Electronics tutorials. *RC Charging Circuit and RC Time Constant*. https://www.electronics-tutorials.ws/rc/rc_1.html. n.d.

[41] RF Wireless World. *Difference between photoresistor and photodiode*. https://www.rfwireless-world.com/Terminology/Photoresistor-vs-Photodiode.html. n.d.

[42] Zhu Cao Xiongfeng Ma Xiao Yuan. *Quantum random number generation*. https://doi-org.tudelft.idm.oclc.org/10.1038/npjqi.2016.21. 2016.