Experiment F66: Quantum Encryption - Yago Obispo Gerster | mn304 | yago.obispo_gerster@stud.uni-heidelberg.de

In [ ]:
```
#Import relevant modules
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from iminuit import Minuit
import math
import pandas as pd
```

1. Introduction

1.1 Motivation and goal of the experiment

Over the past years, quantum technology is becoming more and more relevant. Huge tech companies such as IBM and Google and even whole states invert large amounts of money in the development of this technology. In Quantum Computing, for example, IBM's Quantum Computer Chip Condor has more than 1121 qubits and some researchers estimate, it won't take us long to reach quantum supremacy. Although there currently isn't a quantum computer capable of performing (practically) useful tasks, on the theoretical side, algorithms have already been developed for this technology. One of the most famous, Shore's Algorithm, would allow us to break our current encryption really fast, while a classical computer needs some trillion years to crack RSA encryption.

Therefore, scientists have also developed a new form of encryption, also using quantum properties. Although large enough quantum computers aren't there already, the security methods are, and it is the aim of this experiment to get in touch with quantum encryption overall and specially the Eckert 91 protocol by using the polarization of light photons.

1.2 Physical basics

1.2.1 Polarization of Photons as Signal

For the exchange of information, we need to first agree on a $signal$, which is a physical quantity used for the transmission of information. Its concrete values are interpreted as information (i.e. 0 corresponds to one value and 1 to another).

We are looking for a quantity which can exist in a quantum mechanical superposition and be entangled. The polarization of photons turns out to be an appropiate signal. Here we use linear polarized light and assign one bit to vertical and the other bit to horizontal polarization. We can also entangle two photons, so in example if we measure one of them horizontally polarized, the other one necessarily needs to be vertically polarized.

As we are dealing with a two-level-state system, the three pauli Matrices are the corresponding observables and its eigenvectors the respective polarization states basis vectors:

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\sigma_z \text{ :horizontal/vertikal} \qquad |H\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |V\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\sigma_y \text{ :links/rechtszirkular} \qquad |L\rangle = \frac{i}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad |R\rangle = \frac{i}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

$$\sigma_x \text{ :}+45^\circ/-45^\circ \text{diagonal} \qquad |+\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad |-\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

(Figure 1: Pauli-Matrices and their Eigenvectors from the Experiment Script)

As the Pauli-Matrices do not commute, we can only measure a state in one basis at a time.

1.2.2 No-Cloning-Theorem

For our purpose of using a system for quantum communication, the $No\text{-}Cloning\text{-}Theorem$ plays an important role. It states that a quantum state cannot be copied, so a quantum mechanical bit ($qubit$) cannot be duplicated - which comes from the idea, that by performing a measurement, the wavefunction collapses and the state is projected onto the measured state, so the phase of the quantum state is lost.

We can also prove this theorem mathematically:Lets assume an operator U exists, which copies any arbitrary state $|\psi\rangle$ as follows:

$$U\,|\psi 0\rangle = |\psi\psi\rangle$$

For the conservation of probability, this operator needs to be unitary $U = U^{-1}$. By considering the inner product of two states which we want to copy with U, we get:

$$\langle\phi|\,|\psi\rangle = \langle\phi|\,|\psi\rangle \cdot \langle 0|\,|0\rangle = \left\langle \phi 0|U^\dagger U\right|\,|\psi 0\rangle = \langle\phi\phi|\,|\psi\psi\rangle = \langle\phi|\,|\psi\rangle^2$$

$$\Longrightarrow \langle\phi|\,|\psi\rangle^2 = \langle\phi|\,|\psi\rangle \Longrightarrow \langle\phi|\,|\psi\rangle = 0 \vee \langle\phi|\,|\psi\rangle = 1$$

This means, that two quantum states can only be copied, when they are parallel or orthogonal. So we found out that we can ony copy eigenstates of a single basis. If we on the contrary want to copy a superposition state $|\chi\rangle = a\,|\phi\rangle + b\,|\psi\rangle$, we get a contradiction: On one hand applying assumption that U copies $|\chi\rangle$:

$$U\,|\chi 0\rangle = |\chi\chi\rangle = (a\,|\phi\rangle + b\,|\psi\rangle)(a\,|\phi\rangle + b\,|\psi\rangle) = a^2\,|\phi\phi\rangle + b^2\,|\psi\psi\rangle + ab\,|\phi\psi\rangle + ba\,|\psi\phi\rangle$$

On the other hand substituting $|\chi\rangle$ and applying the operator to $|\phi\rangle$ and $|\psi\rangle$ leads to:

$$U\,(a\,|\phi 0\rangle + b\,|\psi 0\rangle) = a\,|\phi\phi\rangle + b\,|\psi\psi\rangle$$

Therefore we get a contradiction. Qubits are therefore of special interest, as a spy will not be able to copy a quantum mechanical state.

1.2.3 EPR Paradox and Bell Inequality

In quantum mechanics, we have to critically check some of the fundamental assumptions used in classical physics, such as $realism$ - this means that the real world is independent of us performing measurements and $locality$ - which means an interaction between two physical systems can only occur if they are spatially connected. We loose realism because of the possibility to create a superposition of two states. Before we do not perform a measurement, the system does not collaps onto one of the possible states and therefore the current state of the system before measurement is not "real" in a sense and in some way only becomes real when we measure it. The Zeno-Paradoxon is a great example - by continuosly measuring an atom in an excited state, we can maintain it in its excited state. Under no measurement the atom would get back to the non-excited state over a period of time. Entanglement makes locality incompatible.

In our case, for the photon polarization we want to considered the $Bell\text{-}States$, which are the maximally entangled quantum states (if we measure one photon, we know to which polarization the other photon collapsed):

$$\left|\psi^{\pm}\right\rangle = \frac{1}{\sqrt{2}}(|H_A V_B\rangle \pm |V_A H_B\rangle)$$

$$\left|\phi^{\pm}\right\rangle = \frac{1}{\sqrt{2}}(|H_A H_B\rangle \pm |V_A V_B\rangle)$$

Here, the indices A and B each correspond to one photon and stand for "Alice" - in literature typically used as the sender of a message and "Bob" - the receiver of the message.

However during the first quantum mechanical discoveries Albert Einstien himself (a convinced realist) refused the fact that realism is lost. In the EPR-Paradoxon, he argumented, that there must be some hidden variables of quantum systems, which determine the collaps of the wavefunction, but which we are only not capable of measuring. Extraodinarily, John Stewart Bell was able to fomulate an equation which proofed Einstein and his supporters to be wrong, known as $Bell's\ Inequality$. The mathematical formulation of the argument is also known as the $CHSH$ inequality.

Under the classical assumptions of local realism, Alice and Bobs measurements do not affect each other and therefore the probability that Alice and Bob individually measure a certain polarization of their individual photon, can be expressed as the product of the individual probabilities (statistical independence). For the expectation value for Alice measuring in direction $\vec{a}$ and Bob measuring in direction $\vec{b}$, we therefore get the separability criterion:

$$E(\vec{a}, \vec{b}) = E(\vec{a}) \cdot E(\vec{b})$$

Now consider the following expression which we will call our parameter of interest S:

$$S = E(\vec{a}, \vec{b}) - E(\vec{a}, \vec{b}') + E(\vec{a}', \vec{b}) + E(\vec{a}', \vec{b}')$$

Plugging in the separability criterion and taking into consideration that there are only two measurement outcomes, represented by the eigenvalues $E(i) = \pm 1$, in the classical approach we get an upper bound for S of 2:

$$|S| \leq 2$$

For the quantum mechanical approach, we do not assume the separability criterion as local realism is seen critically. Instead we consider the observables for measurements in directions $\vec{a}, \vec{a}'$ and $\vec{b}, \vec{b}'$:

$$O_a = \sum_i \sigma_i a_i \quad O_{a'} = \sum_i \sigma_i a'_i \quad O_b = \sum_i \sigma_i b_i \quad O_{b'} = \sum_i \sigma_i b'_i$$

When two measurements are performed, we compute the expectation value of the product of the observables:

$$E(\vec{a}, \vec{b}) = \langle\psi| (\sum_i \sigma_i a_i)(\sum_i \sigma_i b_i) |\psi\rangle = -\vec{a} \cdot \vec{b} = -cos(\theta_{ab})$$

Therefore, the expectation value only depends on the angle between $\vec{a}$ and $\vec{b}$. The maximum value of S is achieved at:

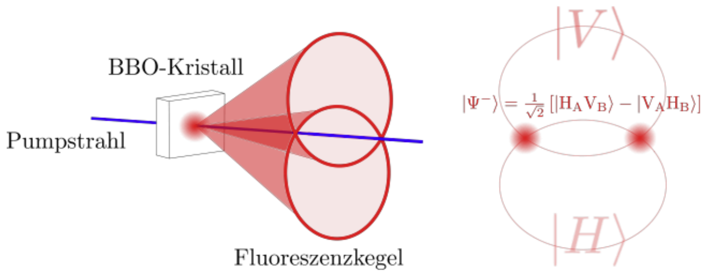
$$S \leq 2\sqrt{2}$$

Therefore, if for the maximizing angles a value which is not in the classical threshold of $S \leq 2$ is measured, cannot be explained with a local-realistic theory and therefore the Hidden Variables theory is proven to be wrong. By measuring the correlations at different angles one can therefore decide if a state behaves classical or quantum mechanical.

1.2.4 Creation of a Bell-State with polarization of photons

In ordered to create a maximally entangled photon pair, we are going to use a laser beam and a so called $birefringent$ crystal (here the BBO beta-barium borate). When projecting the laser beam on it, the BBO has the special property, that because of some non-linear effects, with a small probability, a single incoming photon with frequency ν_P is turned into 2 outgoing fluorescence photons with half the frequency ν_F.

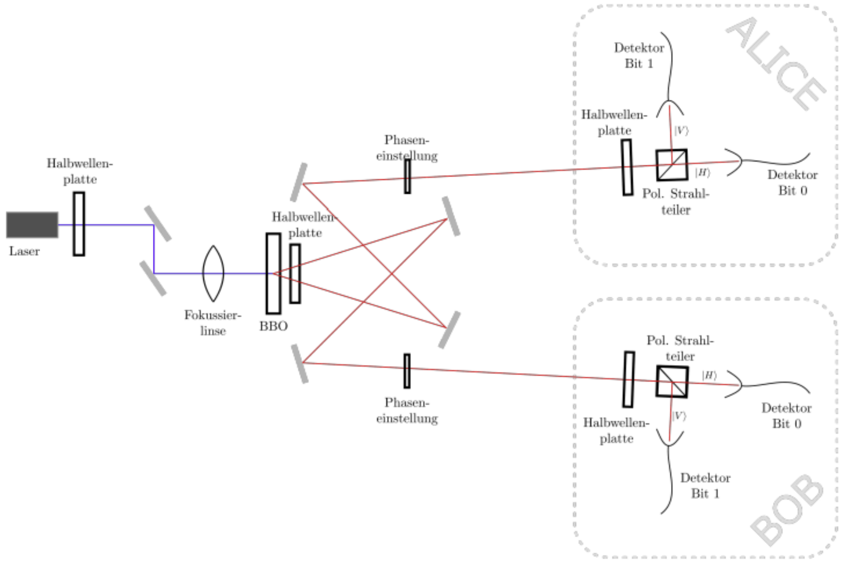
In the type 2 phase matching, for a extraordinarily polarized laser beam, the photons leaving the crystal are one ordinary and one extraordinary polarized - so the pair leaves the crystal in different directions. Therefore 2 emission cones result, which partially overlap. At the intersection of both, we cannot clearly state, which of the two photons is in which of the two possible polarizations, but we are certain that they must be orthogonally polarized to each other. Therefore, by using this method, we were able to construct the maximally entangled Bell-State $|\psi^-\rangle$.



(Figure 2: Creation of Bell-State with BBO from the Experiment Script)

1.2.5 Technical measurement of Bell-Inequality

In order to verify the Bell-Inequality experimentally, the following setup will be used



(Figure 3: Experimental setup from the Experiment Script)

One of the photons of the photon pair goes to Alice and the other to Bob. There each uses a polarizing beam splitter and a single photon detector in order to measure the polarization of the photon. Each detected photon is also attributed a time stamp (very precise because of optimal electronics) and therefore we can assume two photons compose a pair when they arrive at the same times.

A Half-Wave Plate can be used to rotate the polarization by an arbitrary value, so we can measure in different bases.

In practice, instead of measuring the expectation values, we measure the correlation C. After Alice and Bob have measured a Bit-sequence, they compare their results, checking which of them coincide (positive correlation) and where it comes to deviations (negative correlation). The correlation can be determined by using

$$C = \frac{N_{\alpha\beta} + N_{\alpha^\perp\beta^\perp} - N_{\alpha^\perp\beta} - N_{\alpha\beta^\perp}}{N_{\alpha\beta} + N_{\alpha^\perp\beta^\perp} + N_{\alpha^\perp\beta} + N_{\alpha\beta^\perp}}$$

Where N is the respective number of detected events for the different bases. Based on the value of the correlation function, one can describe how similar they are (1 for perfect correlation - Alice and Bob always measure the photons in the same channel, -1 for perfect anticorrelation where they always measure the photons in the opposite channel and 0 for no correlation).

According to Maulu's Law, the number of registered events obeys:

$$N_{\alpha\beta} \propto sin^2(|\alpha - \beta|)$$

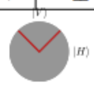
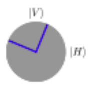
Plugging this relation into our formula for the correlation, we obtain

$$C = -cos(2|\alpha - \beta|)$$

For verifying the Bell inequality, one must use angles, which maximize the S parameter. This turns out to be the case for

$$|\alpha - \beta| = \frac{\pi}{8}$$

Therefore, we will use the following bases for measurement:

Alice: / Bob:		α (0°)	$\alpha_\perp$ (90°)	α' (45°)	$\alpha'_\perp$ (135°)
					
β (22.5°)		$C_{\alpha\beta}$		$C_{\alpha'\beta}$	
$\beta_\perp$ (112.5°)					
β' (67.5°)		$C_{\alpha\beta'}$		$C_{\alpha'\beta'}$	
$\beta'_\perp$ (157.5°)					

(Figure 4: Bases for maximum violation of Bell (Experiment Script))

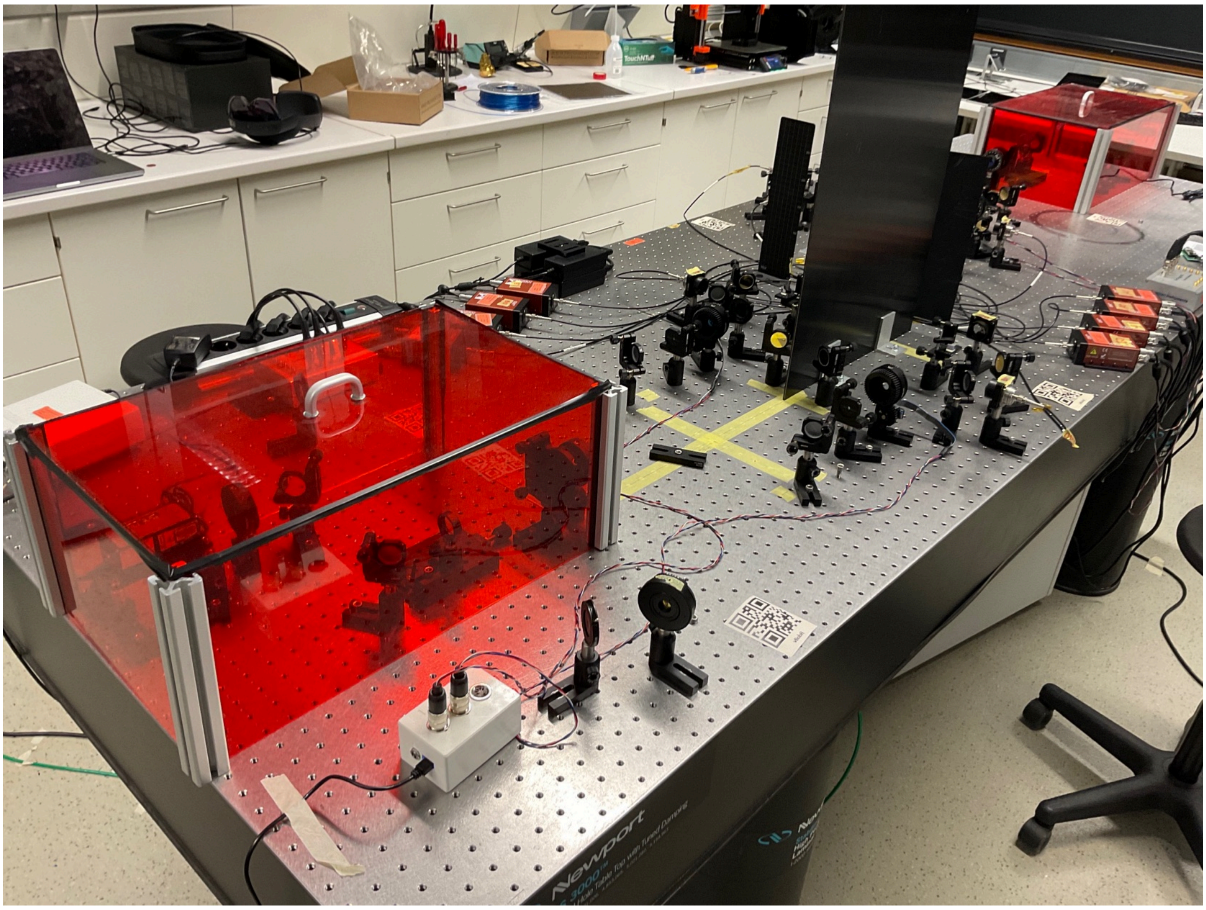
1.2.6 The Ekert91 protocol

The $Ekert91$ protocol can be used for quantum communication and encryption. The process works the following way:

1. Alice and Bob agree that they will only use the four bases for measurement which maximize the S parameter in the Bell-Inequality
2. Both perform a measurement in order to determine the key. Therefore, they randomly (by a quantum random number generator) change between the four measurement bases after each photon measured for a sequence of (25) photons
3. Alice and Bob tell each other publicly, which Bases they have used for the measurement. The bits of the matching bases are then selected for the key
4. In order to check if a spy intercepted their message, they use the measurements performed on other (non-key) Bases and check Bell's Inequality. If the S value is $S \leq 2$ they know, they are being intercepted by a spy. This is due to the fact that the spy cannot copy the send message without destroying it (no-cloning-theorem). After performing a measurement he would need to

send a new pair of bits to Alice and Bob. However he is going to send classical states, which in one basis are going to give a strong anti-correlation (i.e. $|H\rangle\,|V\rangle$) but in another basis no correlation. The S factor will lie in the classical range
5. When Alice wants to send a message, she adds her (Bit) key to the (Bit) message and sends it
6. When Bob receives the message he adds his key to the message, inverts the Bit sequence (because of the anti-correlation of the $|\psi^-\rangle$ state and obtains her message

2. Execution and Evaluation



(Figure 5: Experimental setup)

2.1 Single count rates

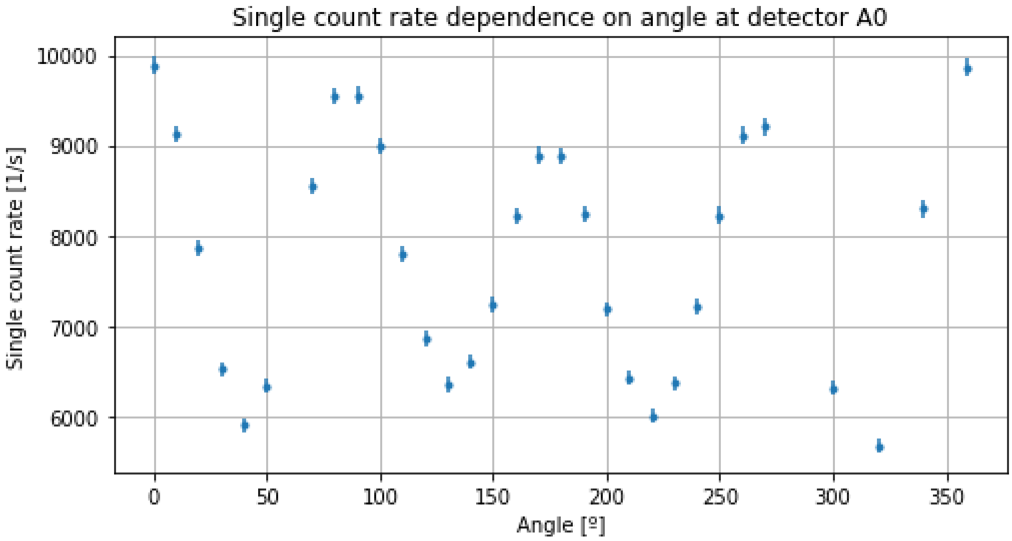
We first only consider the single count rates of the incoming fluorescence photons at Alice and Bob. We do not filter the results for coincidences yet.

In the first step, we took a series of measurements of the count rates for different polarization angles. The program recorded the number of detected photons per 5 seconds, so in order to get the rate, we have to divide the measured values by 5.

In [ ]:
```
#Import measurements: count rates for different polarization angles
degreeA, degreeB, A0, A1, B0, B1, A0B0, A1B0, A0B1, A1B1 =  np.loadtxt(
'2024-08-21_11-11-12_Zufall_Alice (1).txt', unpack = True, skiprows = 1
, delimiter = ';')

#Compute count rate
```

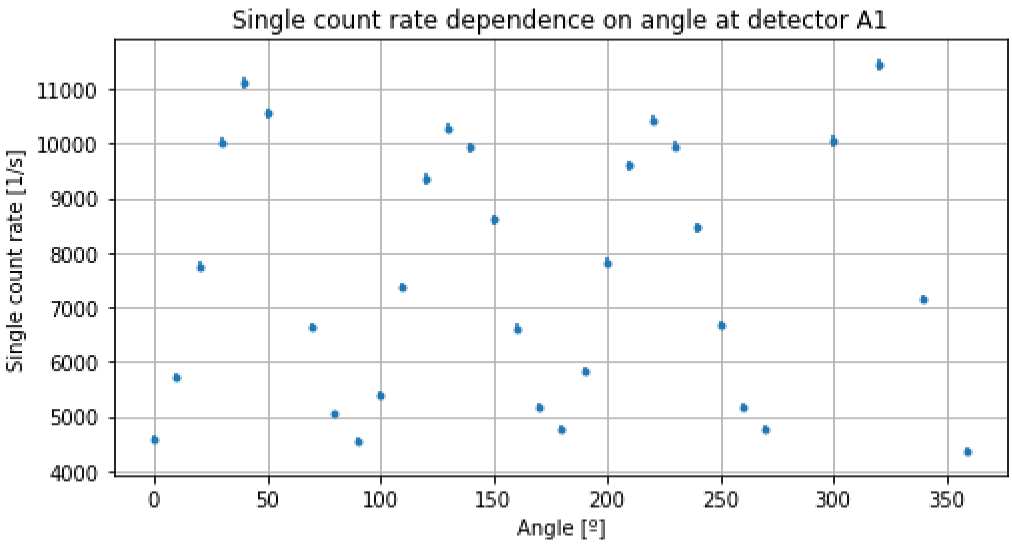
```
A0 = A0/5
A1= A1/5
B0=B0/5
B1 = B1/5

#Plot single count rate against angle Alice
plt.figure(figsize=(8,4))
plt.grid()
plt.title("Single count rate dependence on angle at detector A0")
plt.errorbar(degreeA,A0,yerr=np.sqrt(A0),fmt=".")
plt.xlabel("Angle [º]")
plt.ylabel("Single count rate [1/s]")
```

Out[ ]: Text(0, 0.5, 'Single count rate [1/s]')



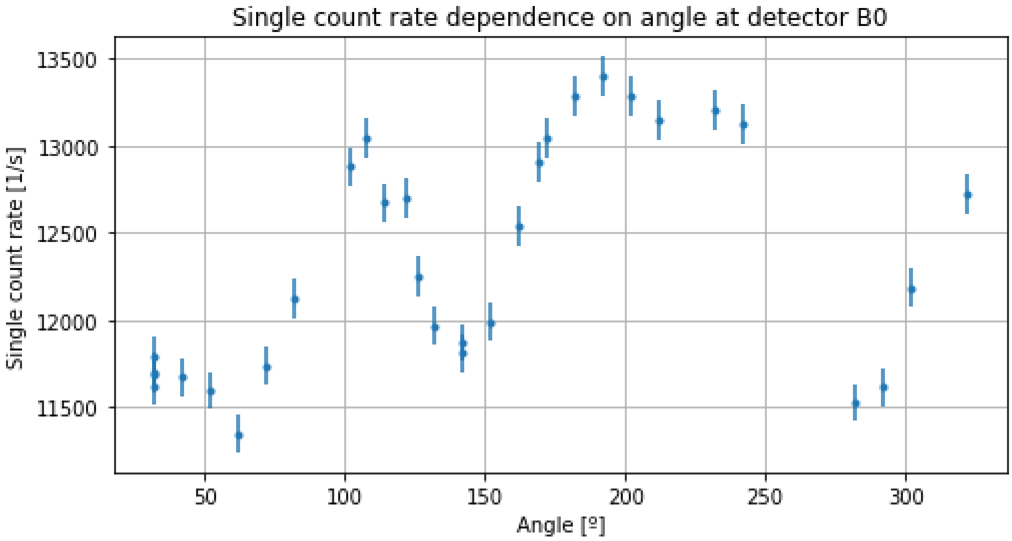
In [ ]:
```
#Plot single count rate against angle Alice
plt.figure(figsize=(8,4))
plt.grid()
plt.title("Single count rate dependence on angle at detector A1")
plt.errorbar(degreeA,A1,yerr=np.sqrt(A1),fmt=".")
plt.xlabel("Angle [º]")
plt.ylabel("Single count rate [1/s]")
```

Out[ ]: Text(0, 0.5, 'Single count rate [1/s]')



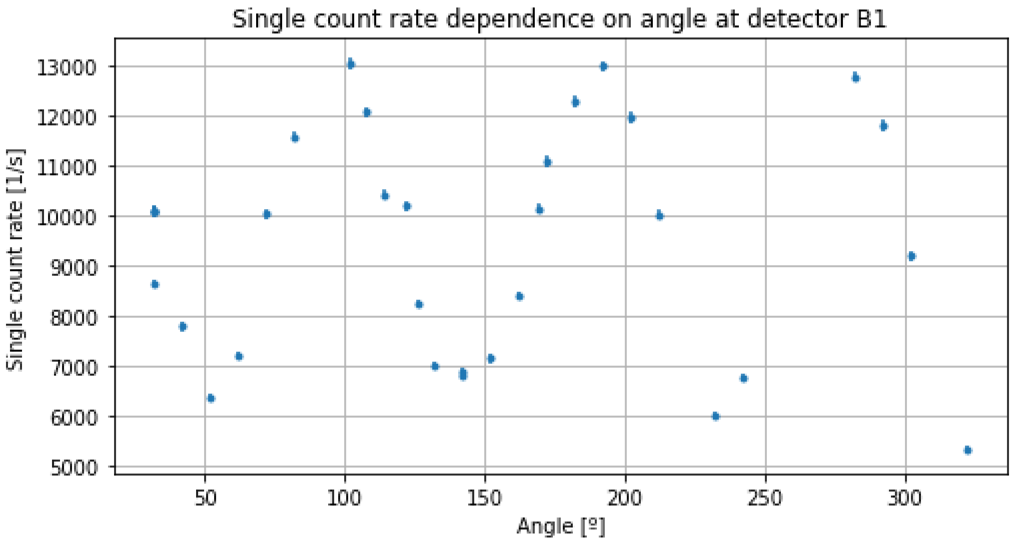
In [ ]:
```
#Plot single count rate against angle Bob
plt.figure(figsize=(8,4))
plt.grid()
plt.title("Single count rate dependence on angle at detector B0")
plt.errorbar(degreeB,B0,yerr=np.sqrt(B0),fmt=".")
plt.xlabel("Angle [º]")
plt.ylabel("Single count rate [1/s]")
```

Out[ ]:
```
Text(0, 0.5, 'Single count rate [1/s]')
```



In [ ]:
```
#Plot single count rate against angle Bob
plt.figure(figsize=(8,4))
plt.grid()
plt.title("Single count rate dependence on angle at detector B1")
plt.errorbar(degreeB,B1,yerr=np.sqrt(B1),fmt=".")
plt.xlabel("Angle [º]")
plt.ylabel("Single count rate [1/s]")
```

Out[ ]:
```
Text(0, 0.5, 'Single count rate [1/s]')
```



The observed single count rates at all detectors show a periodic behavior for all measurement arms, except for B0 where one can still see the periodicity, but for angles between 200° and 280° the periodicity is not as clear. This is most probably due to a systematic error with the experimental setup. For example, the program might had a small delay before stopping the counter, so more photons were actually measured. On the other hand the intensity of the laser might have varied in that time-frame or we could have influenced in some way the experimental setup (by for example touching the table). Therefore, we assume this deviation to be an experimental error and not a theoretical one.

We can further assess the degree of polarization of the incoming photons by computing the contrast for each detector, which is defined by:

$$V = \frac{N_{Max} - N_{min}}{N_{max} + N_{min}}$$

with a respective error of

$$\sigma_V = \sqrt{\left(\frac{2N_{min}\sqrt{N_{max}}}{(N_{min}+N_{max})^2}\right)^2 + \left(\frac{2N_{max}\sqrt{N_{min}}}{(N_{min}+N_{max})^2}\right)^2}$$

In [ ]:

```
#Define function for computing the contrast V and its error
def V(N_max,N_min):
    return (N_max-N_min)/(N_max+N_min)
def sig_V(N_max,N_min):
    return np.sqrt((2*N_min*np.sqrt(N_max))/(N_min+N_max)**2
    + (2*N_max*np.sqrt(N_min))/(N_min+N_max)**2)

#Determine N values
N_max_A0 = np.max(A0)
N_max_A1 = np.max(A1)
N_max_B0 = np.max(B0)
N_max_B1 = np.max(B1)

N_min_A0 = np.min(A0)
N_min_A1 = np.min(A1)
N_min_B0 = np.min(B0)
N_min_B1 = np.min(B1)

#Compute V and sig_V
V_A0 = V(N_max_A0,N_min_A0)
V_A1 = V(N_max_A1,N_min_A1)
V_B0 = V(N_max_B0,N_min_B0)
V_B1 = V(N_max_B1,N_min_B1)

sig_V_A0 = sig_V(N_max_A0,N_min_A0)
sig_V_A1 = sig_V(N_max_A1,N_min_A1)
sig_V_B0 = sig_V(N_max_B0,N_min_B0)
sig_V_B1 = sig_V(N_max_B1,N_min_B1)

print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A0",np.round(V_A0,2),np.round(sig_V_A0,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A1",np.round(V_A1,2),np.round(sig_V_A1,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("B0",np.round(V_B0,2),np.round(sig_V_B0,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("B1",np.round(V_B1,2),np.round(sig_V_B1,2)))
```

```
Contrast V for A0: VA0 = 0.27 +/- 0.1
Contrast V for A1: VA1 = 0.45 +/- 0.1
Contrast V for B0: VB0 = 0.08 +/- 0.09
Contrast V for B1: VB1 = 0.42 +/- 0.1
```

The contrast describes the degree in which the polarization fluctuates. Therefore, for a small contrast, the light does not fluctuate as much and is not as strongly polarized as for higher contrasts.

According to our results, all arms except for the B0 arm, show that the value is not close to zero and the light is therefore slightly polarized.

The problem in taking the contrast as a measure for fluctuation is, that it is not representative for measurements in which the count rates at the maxima and minima vary a lot from one extremum to the other. This is because for computing the contrast, we only take into account the maximum value of counted events and the minimal value. Therefore, if this single value was measured to be wrong (which in a large dataset is likely), the whole contrast turns out to be nonsense.

For the A1 arm in example, the 2 peaks in the middle have a smaller amplitude than the lateral ones. Therefore the contrast turns out the highest. As discussed, due to an experimental error, at arm B0, some of the count rates where measured wrongly and therefore the obtained value for the contrast is not meaningful.

For this reasons, as N_{max} has probably been overestimated and N_{min} undersestimated, the contrast turns out to be greater than it probably is. If there were less deviations between the extrema and we could build a perfect experimental setup, when measuring both light beams independently, the light in each of the arms should be close to not being polarized, as we are not considering an entanglement yet.

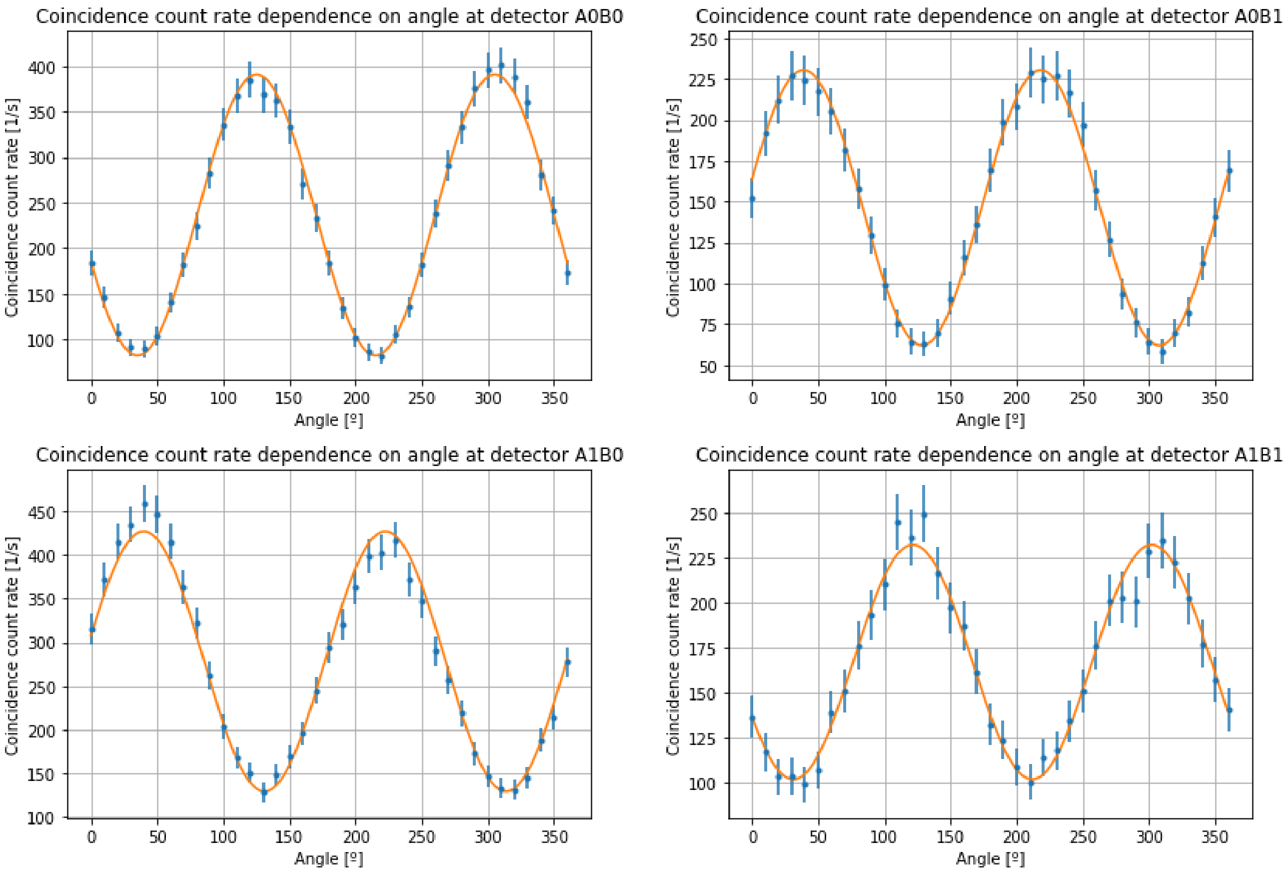
2.2 Coincidence count rates

In the second step of the experiment, a fixed basis was chosen at one of the arms (the 0° and 90° basis) and the polarization direction is varied on the other arm.

In [ ]:

```
#Import coincidence measurements at Alice arm in fixed 0 to 90 degree basis
degreeA, degreeB, A0, A1, B0, B1, A0B0, A1B0, A0B1, A1B1 =  np.loadtxt(
'0and90CoincidenceBob.txt', unpack = True, skiprows = 1
, delimiter = ';')

#Correct degrees
degreeA = degreeA*2
degreeB = degreeB*2

#Compute count rate
A0B0 = A0B0/5
A0B1 = A0B1/5
A1B0 = A1B0/5
A1B1 = A1B1/5

def sinsq(x,a,b,c,d):
    return a*np.sin(np.deg2rad(b*(x-c)))**2 + d
```

```python
poptA0B0,pcovA0B0 = curve_fit(sinsq,degreeB,A0B0,p0=[400,1,0,0])
poptA0B1,pcovA0B1 = curve_fit(sinsq,degreeB,A0B1,p0=[400,1,0,0])
poptA1B0,pcovA1B0 = curve_fit(sinsq,degreeB,A1B0,p0=[400,1,0,0])
poptA1B1,pcovA1B1 = curve_fit(sinsq,degreeB,A1B1,p0=[400,1,0,0])

#Plot coincidence count rate against angle
var_deg_b = np.linspace(0,360,1000)
fig, axs = plt.subplots(2, 2, figsize=(12, 8))

# Plot 1: A0B0
axs[0, 0].grid()
axs[0, 0].set_title("Coincidence count rate dependence on angle at detector A0B0")
axs[0, 0].errorbar(degreeB, A0B0, yerr=np.sqrt(A0B0), fmt=".")
axs[0, 0].plot(var_deg_b, sinsq(var_deg_b, *poptA0B0))
axs[0, 0].set_xlabel("Angle [º]")
axs[0, 0].set_ylabel("Coincidence count rate [1/s]")

# Plot 2: A0B1
axs[0, 1].grid()
axs[0, 1].set_title("Coincidence count rate dependence on angle at detector A0B1")
axs[0, 1].errorbar(degreeB, A0B1, yerr=np.sqrt(A0B1), fmt=".")
axs[0, 1].plot(var_deg_b, sinsq(var_deg_b, *poptA0B1))
axs[0, 1].set_xlabel("Angle [º]")
axs[0, 1].set_ylabel("Coincidence count rate [1/s]")

# Plot 3: A1B0
axs[1, 0].grid()
axs[1, 0].set_title("Coincidence count rate dependence on angle at detector A1B0")
axs[1, 0].errorbar(degreeB, A1B0, yerr=np.sqrt(A1B0), fmt=".")
axs[1, 0].plot(var_deg_b, sinsq(var_deg_b, *poptA1B0))
axs[1, 0].set_xlabel("Angle [º]")
axs[1, 0].set_ylabel("Coincidence count rate [1/s]")

# Plot 4: A1B1
axs[1, 1].grid()
axs[1, 1].set_title("Coincidence count rate dependence on angle at detector A1B1")
axs[1, 1].errorbar(degreeB, A1B1, yerr=np.sqrt(A1B1), fmt=".")
axs[1, 1].plot(var_deg_b, sinsq(var_deg_b, *poptA1B1))
axs[1, 1].set_xlabel("Angle [º]")
axs[1, 1].set_ylabel("Coincidence count rate [1/s]")

# Adjust layout to prevent overlap
plt.tight_layout()

# Show the combined plot
plt.show()
```



We observe the expected results from the experiment script, as we can clearly see the periodic $sin^2(.)$ or $cos^2(.)$ behaviour. We can also see, that the measured values approximatelly agree for A0B0 and A1B1 and also for A0B1 and A1B0.

For A0B1 and A1B0 according to the theory, at 0° both functions should take a maximum because of the anticorrelation of the maximally entangled states. However, in the plots the maximum is shifted by around 40° in both cases. I assume, that the $\lambda/2$ plate was probably calibrated wrongly (although the calibration was done with the programm at the start of the experiment) and therefore in the whole experiment we will obtain values shifted by a constant. The reason for my hypothesis is, that the plots for A0B0 and A1B1 or A1B0 and A0B1 agree very precisely.

One can also observe, that the minima are not completely at a count rate of zero. This can be explained due to other light sources which might have been in the room and had an influence into the experimental setup.

Analogous to the single count rates, we can compute the contrast for each of the arms.

In [ ]:

```
#Compute the contrast for coincidence count rates at 0/90
#Determine N values
N_max_A0B0 = np.max(A0B0)
N_max_A1B0 = np.max(A1B0)
N_max_A0B1 = np.max(A0B1)
N_max_A1B1 = np.max(A1B1)

N_min_A0B0 = np.min(A0B0)
N_min_A1B0 = np.min(A1B0)
N_min_A0B1 = np.min(A0B1)
N_min_A1B1 = np.min(A1B1)
```

```
#Compute V and sig_V
V_A0B0 = V(N_max_A0B0,N_min_A0B0)
V_A1B0 = V(N_max_A1B0,N_min_A1B0)
V_A0B1 = V(N_max_A0B1,N_min_A0B1)
V_A1B1 = V(N_max_A1B1,N_min_A1B1)

sig_V_A0B0 = sig_V(N_max_A0B0,N_min_A0B0)
sig_V_A1B0 = sig_V(N_max_A1B0,N_min_A1B0)
sig_V_A0B1 = sig_V(N_max_A0B1,N_min_A0B1)
sig_V_A1B1 = sig_V(N_max_A1B1,N_min_A1B1)

print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A0B0",np.round(V_A0B0,2),np.round(sig_V_A0B0,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A1B0",np.round(V_A1B0,2),np.round(sig_V_A1B0,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A0B1",np.round(V_A0B1,2),np.round(sig_V_A0B1,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A1B1",np.round(V_A1B1,2),np.round(sig_V_A1B1,2)))
```

```
Contrast V for A0B0: VA0B0 = 0.66 +/- 0.21
Contrast V for A1B0: VA1B0 = 0.56 +/- 0.21
Contrast V for A0B1: VA0B1 = 0.59 +/- 0.25
Contrast V for A1B1: VA1B1 = 0.43 +/- 0.26
```

Here, we can clearly see, that the contrast is closer to 1 in all of the four cases and is therefore much stronger than in the single count rate measurement. The contrast is expected to be 1, as we are dealing with a perfectly correlated state, so that when measuring in the same basis, the count rate for A0B0 and A1B1 should become zero and when measuring in bases with an angle of 90° to each other, A1B0 and A0B1 should become zero. However, as already noted, the value is not exactly zero because of external light sources and a delicate experimental setup - which did not guarantee the perfect anticorrelation of the state $|\psi^-\rangle$.

However, comparing the magnitude of the contrasts in coincidence count rates with the single count rates, the coincidence count rates turn out to be much higher, which emphasizes of the existence anticorrelated, maximally entangled state.

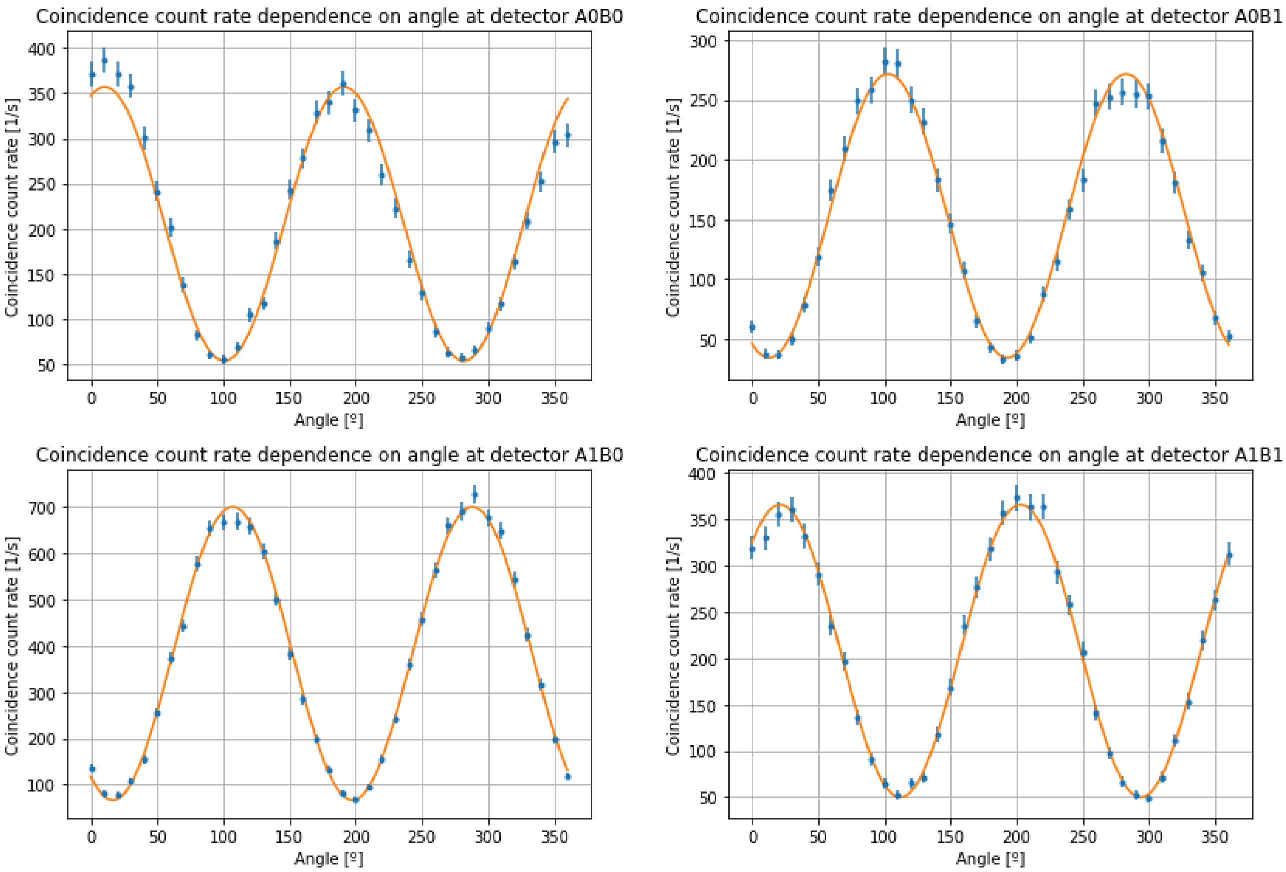
2.2.2 Fixed Bases at -45°/ 45°

We can repeat the same measurements for another fixed basis (45°/ -45°) and then compare the results to the previous measurement. We first plot the measured data, in order to visualize the polarisation and then compute the contrast to quantitatevily assess the degree of polarization.

In [ ]:

```
#Import coincidence measurements in fixed 45/ -45 degree basis
degreeA, degreeB, A0, A1, B0, B1, A0B0, A1B0, A0B1, A1B1 =  np.loadtxt(
'2024-08-21_11-42-02_Korrelation_Bob (1).txt', unpack = True, skiprows = 1
, delimiter = ';')

#Correct degrees
degreeA = degreeA*2
degreeB = degreeB*2

#Compute count rate
```

```python
A0B0 = A0B0/5
A0B1 = A0B1/5
A1B0 = A1B0/5
A1B1 = A1B1/5

def sinsq(x,a,b,c,d):
    return a*np.sin(np.deg2rad(b*(x-c)))**2 + d

poptA0B0,pcovA0B0 = curve_fit(sinsq,degreeB,A0B0,p0=[400,1,0,0])
poptA0B1,pcovA0B1 = curve_fit(sinsq,degreeB,A0B1,p0=[400,1,0,0])
poptA1B0,pcovA1B0 = curve_fit(sinsq,degreeB,A1B0,p0=[400,1,0,0])
poptA1B1,pcovA1B1 = curve_fit(sinsq,degreeB,A1B1,p0=[400,1,0,0])

#Plot coincidence count rate against angle
var_deg_b = np.linspace(0,360,1000)
fig, axs = plt.subplots(2, 2, figsize=(12, 8))

# Plot 1: A0B0
axs[0, 0].grid()
axs[0, 0].set_title("Coincidence count rate dependence on angle at detector A0B0")
axs[0, 0].errorbar(degreeB, A0B0, yerr=np.sqrt(A0B0)/np.sqrt(2), fmt=".")
axs[0, 0].plot(var_deg_b, sinsq(var_deg_b, *poptA0B0))
axs[0, 0].set_xlabel("Angle [º]")
axs[0, 0].set_ylabel("Coincidence count rate [1/s]")

# Plot 2: A0B1
axs[0, 1].grid()
axs[0, 1].set_title("Coincidence count rate dependence on angle at detector A0B1")
axs[0, 1].errorbar(degreeB, A0B1, yerr=np.sqrt(A0B1)/np.sqrt(2), fmt=".")
axs[0, 1].plot(var_deg_b, sinsq(var_deg_b, *poptA0B1))
axs[0, 1].set_xlabel("Angle [º]")
axs[0, 1].set_ylabel("Coincidence count rate [1/s]")

# Plot 3: A1B0
axs[1, 0].grid()
axs[1, 0].set_title("Coincidence count rate dependence on angle at detector A1B0")
axs[1, 0].errorbar(degreeB, A1B0, yerr=np.sqrt(A1B0)/np.sqrt(2), fmt=".")
axs[1, 0].plot(var_deg_b, sinsq(var_deg_b, *poptA1B0))
axs[1, 0].set_xlabel("Angle [º]")
axs[1, 0].set_ylabel("Coincidence count rate [1/s]")

# Plot 4: A1B1
axs[1, 1].grid()
axs[1, 1].set_title("Coincidence count rate dependence on angle at detector A1B1")
axs[1, 1].errorbar(degreeB, A1B1, yerr=np.sqrt(A1B1)/np.sqrt(2), fmt=".")
axs[1, 1].plot(var_deg_b, sinsq(var_deg_b, *poptA1B1))
axs[1, 1].set_xlabel("Angle [º]")
axs[1, 1].set_ylabel("Coincidence count rate [1/s]")

# Adjust layout to prevent overlap
plt.tight_layout()

# Show the combined plot
plt.show()
```



We can again note the periodic $sin(.)^2$ and $cos(.)^2$ behaviour discussed before. The difference between the measurement in the -45°/45° basis and the 0/90 basis is mainly, that the peaks are shifted by $45º$ to the right, which agrees with our expectations as we changed the measurement basis in one of the arms by $45º$ but everything else in the setup remained the same. My hypothesis with the shift due to the $\lambda/2$ plate is confirmed here again.

In [ ]:

```
#Compute the contrast for coincidence count rates at -45/45
#Determine N values
N_max_A0B0 = np.max(A0B0)
N_max_A1B0 = np.max(A1B0)
N_max_A0B1 = np.max(A0B1)
N_max_A1B1 = np.max(A1B1)

N_min_A0B0 = np.min(A0B0)
N_min_A1B0 = np.min(A1B0)
N_min_A0B1 = np.min(A0B1)
N_min_A1B1 = np.min(A1B1)

#Compute V and sig_V
V_A0B0 = V(N_max_A0B0,N_min_A0B0)
V_A1B0 = V(N_max_A1B0,N_min_A1B0)
V_A0B1 = V(N_max_A0B1,N_min_A0B1)
V_A1B1 = V(N_max_A1B1,N_min_A1B1)

sig_V_A0B0 = sig_V(N_max_A0B0,N_min_A0B0)
sig_V_A1B0 = sig_V(N_max_A1B0,N_min_A1B0)
sig_V_A0B1 = sig_V(N_max_A0B1,N_min_A0B1)
sig_V_A1B1 = sig_V(N_max_A1B1,N_min_A1B1)

print("Contrast V for {0}: V{0} = {1} +/- {2}"
```

```
.format("A0B0",np.round(V_A0B0,2),np.round(sig_V_A0B0,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A1B0",np.round(V_A1B0,2),np.round(sig_V_A1B0,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A0B1",np.round(V_A0B1,2),np.round(sig_V_A0B1,2)))
print("Contrast V for {0}: V{0} = {1} +/- {2}"
.format("A1B1",np.round(V_A1B1,2),np.round(sig_V_A1B1,2)))
```

```
Contrast V for A0B0: VA0B0 = 0.75 +/- 0.2
Contrast V for A1B0: VA1B0 = 0.83 +/- 0.16
Contrast V for A0B1: VA0B1 = 0.79 +/- 0.21
Contrast V for A1B1: VA1B1 = 0.76 +/- 0.2
```

We can again observe contrasts close to 1 for the same reasons already discussed in the $0/90$ basis. Compared to the $0/90$ basis, the contrasts are slightly higher, which probably is due to the fact, that in this part of the experiment the external perturbations (for example light sources) did not have a big influence. Therefore, also in this bases, we can confirm the existence of an anticorrelated entangled quantum state and confirm that the setup with the crystal is (except for small systematic deviations) correct.

2.3 Test of Bells inequality in CHSH form

In the third step of the experiment we measured the coincidence count rates the four bases which allow us to test the CHSH inequality (which maximize S). We aim to proof, that we can experimentally obtain a S value over the classical threshold in order to proof Einstein wrong with his local-realism theory.

In [ ]:

```
degreeA, degreeB, A0B0, A1B0, A0B1, A1B1 =  np.loadtxt(
'2024-08-21_12-13-22_Verschraenkung_Alice.txt', unpack = True, skiprows = 1
, delimiter = ';')

#Correct degrees
degreeA = degreeA *2
degreeB = degreeB *2

#Compute count rate
A0B0 = A0B0/5
A0B1 = A0B1/5
A1B0 = A1B0/5
A1B1 = A1B1/5

t = np.array([degreeA,degreeB,A0B0,A1B0,A0B1,A1B1])
df = pd.DataFrame(t)
row_labels = ["Basis Alice [º]","Basis Bob[º]","A0B0 [1/s]"
,"A1B0 [1/s]","A0B1 [1/s]","A1B1 [1/s]"]
df.index = row_labels
df
```

Out[ ]:

	0	1	2	3
Basis Alice [°]	0.0	90.0	90.0	0.0
Basis Bob[°]	46.0	46.0	136.0	136.0
A0B0 [1/s]	443.2	77.4	299.2	98.6
A1B0 [1/s]	54.6	417.4	89.2	269.4
A0B1 [1/s]	79.0	352.6	86.8	232.4
A1B1 [1/s]	258.0	55.4	238.2	67.0

We can compute the correlation function by using

$$C = \frac{N_{\alpha\beta} + N_{\alpha^\perp\beta^\perp} - N_{\alpha^\perp\beta} - N_{\alpha\beta^\perp}}{N_{\alpha\beta} + N_{\alpha^\perp\beta^\perp} + N_{\alpha^\perp\beta} + N_{\alpha\beta^\perp}}$$

$$\triangle C = \frac{\sqrt{4\cdot\left(\sqrt{N_{\alpha\beta}}\cdot(N_\alpha\beta^\perp + N_\alpha^\perp\beta)\right)^2 + 4\cdot\left(\sqrt{N_{\alpha^\perp\beta^\perp}}\cdot(N_\alpha\beta^\perp + N_\alpha^\perp\beta)\right)^2 + 4\cdot\left(\sqrt{N_{\alpha^\perp\beta}}\cdot(N_\alpha\beta + N_\alpha^\perp\beta^\perp)\right)^2 + 4\cdot\left(\sqrt{N_{\alpha\beta^\perp}}\cdot(N_\alpha\beta + N_\alpha^\perp\beta^\perp)\right)^2}}{(N_{\alpha\beta} + N_{\alpha^\perp\beta^\perp} + N_{\alpha^\perp\beta} + N_{\alpha\beta^\perp})^2}$$

For the error of the single $N's$ the square root was taken.

In [ ]:

```
#Determine correlation function C in each case
def C(a,b,c,d):
    return (a+b-c-d)/(a+b+c+d)

def sig_C(a,b,c,d):
    return np.sqrt(4*((c+d)*np.sqrt(a)/(a+b+c+d)**2)**2
    +4*((c+d)*np.sqrt(b)/(a+b+c+d)**2)**2+
    4*((a+b)*np.sqrt(c)/(a+b+c+d)**2)**2
    + 4*((a+b)*np.sqrt(d)/(a+b+c+d)**2)**2)

C_a_b = C(A0B0[0],A1B1[0],A1B0[0],A0B1[0])
sig_C_a_b = sig_C(A0B0[0],A1B1[0],A1B0[0],A0B1[0])

C_a_bp = C(A0B0[3],A1B1[3],A1B0[3],A0B1[3])
sig_C_a_bp = sig_C(A0B0[3],A1B1[3],A1B0[3],A0B1[3])

C_ap_bp = C(A0B0[2],A1B1[2],A1B0[2],A0B1[2])
sig_C_ap_bp = sig_C(A0B0[2],A1B1[2],A1B0[2],A0B1[2])

C_ap_b = C(A0B0[1],A1B1[1],A1B0[1],A0B1[1])
sig_C_ap_b = sig_C(A0B0[1],A1B1[1],A1B0[1],A0B1[1])

print("Correlation function C_alphabeta = {0} +/- {1}"
.format(np.round(C_a_b,2),np.round(sig_C_a_b,2)))
print("Correlation function C_alphabeta' = {0} +/- {1}"
.format(np.round(C_a_bp,2),np.round(sig_C_a_bp,2)))
print("Correlation function C_alpha'beta' = {0} +/- {1}"
.format(np.round(C_ap_bp,2),np.round(sig_C_ap_bp,2)))
print("Correlation function C_alpha'beta = {0} +/- {1}"
.format(np.round(C_ap_b,3),np.round(sig_C_ap_b,3)))
```

```
Correlation function C_alphabeta = 0.68 +/- 0.03
Correlation function C_alphabeta' = -0.5 +/- 0.03
Correlation function C_alpha'beta' = 0.51 +/- 0.03
Correlation function C_alpha'beta = -0.706 +/- 0.024
```

The S parameter can be determined by using:

$$S = C(\alpha, \beta) - C(\alpha, \beta') + C(\alpha', \beta) + C(\alpha', \beta')$$

$$\triangle S = \sqrt{(\triangle C(\alpha, \beta))^2 + (\triangle C(\alpha, \beta'))^2 + (\triangle C(\alpha', \beta))^2 + (\triangle C(\alpha', \beta'))^2}$$

In [ ]:
```
#Determine the S parameter in each case
def S(a,b,c,d):
    return a-b+c+d

def sig_S(sig_a,sig_b,sig_c,sig_d):
    return np.sqrt(sig_a**2 + sig_b**2 +sig_c**2 + sig_d**2)

s_Parameter = S(C_a_b,C_a_bp,C_ap_b,C_ap_bp)
sig_s_Parameter = sig_S(sig_C_a_b,sig_C_a_bp,sig_C_ap_b,sig_C_ap_bp)

print("S-Parameter = {0} +/- {1}"
.format(np.round(s_Parameter,2),np.round(sig_s_Parameter,2)))
```

```
S-Parameter = 0.98 +/- 0.06
```

We obtain an S-parameter close to one. Therefore, the result is in the classical range of $|S| \leq 2$. This is a problem, as we wanted the S-parameter to be grater than 2 in order to argue, that a local-realistic theory cannot be used in order to explain the outcome and with this proof Einsteins theory of hidden variables to be wrong.

The expected value of the S-parameter under perfect experimental circumstances would be $|S| \approx 2\sqrt{2}$ under the chosen (maximizing) bases.

The main explanation for the experimental deviation of the theoretically expected value is the wrongly calibrated $\lambda/2$ plate. Although we carefully chose the calculated bases which maximize the S parameter, in the experiment we used different bases - shifted by a constant, which are not useful for maximizing S. By calibrating the waveplate correctly we probably would get a much higher result.

If the experimental setup could be adjusted, we would have shown the violation of the CHSH inequality. Furthermore, we would have been able to proof in a distinct manner than the one used in the measurement of coincidence count rates and the contrast in different bases, that we successfully created an entangled state, as the explanation of a S parameter over the classical threshold, needs entanglement. By having proven the existence of entanglement, we would have also proven the violation of CHSH, as entanglement is in contradiction to any local realistic theory.

2.4 The Eckert-91 Protocol

In this section, we dedicated ourselves to the main goal of the experiment: being able to communicate messages with the Eckert 91 protocol. Therefore, according to the procedure already described in the introduction, we first need to establish a key. Therefore Alice and Bob randomly take measurements in the bases needed for maximizing the S parameter. The bases at which both agreed are used for the generation of the key and the other ones are discarded.

In the first step, all the measurements are presented, then only the ones where Alice and Bob chose different and same bases.

In [ ]:
```
df = pd.read_csv('2024-08-21_15-04-57_Kryptographie_Bob.txt', sep = ';')
column_labels = ['Degree Alice','Degree Bob','Bit Alice'
,'Bit Bob', 'A0B0', 'A1B0', 'A0B1', 'A1B1']
df.columns = column_labels
df['Degree Alice']*=2
df['Degree Bob']*=2
df
```

Out[ ]:

	Degree Alice	Degree Bob	Bit Alice	Bit Bob	A0B0	A1B0	A0B1	A1B1
0	44	44	1	0	955	970	1191	923
1	44	44	0	1	955	970	1191	923
2	44	44	1	1	955	970	1191	923
3	44	44	0	0	1012	954	1169	939
4	44	44	0	0	1040	965	1227	895
...	...	...	...	...	...	...	...	...
58	116	68	0	0	869	1018	1131	883
59	116	68	0	1	869	1018	1131	883
60	116	68	0	0	869	1018	1131	883
61	0	0	1	1	2135	317	339	1340
62	0	0	0	0	2097	350	324	1279

63 rows × 8 columns

In [ ]:
```
#Plot only the results for different bases used
df[df['Degree Alice'] != df['Degree Bob']]
```

Out[ ]:

	Degree Alice	Degree Bob	Bit Alice	Bit Bob	A0B0	A1B0	A0B1	A1B1
25	0	22	0	0	2434	72	89	1604
26	44	68	0	0	1105	866	1064	883
27	88	68	1	0	907	921	1175	885
28	136	44	1	0	381	1696	1670	425
29	0	22	1	1	2436	102	108	1546
31	44	0	1	0	1872	463	492	1186
32	44	0	1	1	1825	466	509	1246
33	88	0	1	0	1195	921	930	884
34	88	22	0	1	1183	1011	1137	967
35	68	22	1	0	1155	964	1078	1001
36	0	22	1	1	2401	70	104	1544
37	0	44	0	0	2260	241	280	1450
38	44	68	1	1	1160	794	978	891
40	88	22	1	1	1135	1000	1140	1025
41	136	22	0	1	304	1879	1704	376
42	136	44	0	1	395	1787	1624	470
43	0	68	0	0	1332	964	920	799
44	0	44	0	0	2024	432	446	1214
45	0	22	0	0	2434	72	89	1604
46	44	68	0	0	1105	866	1064	883
47	116	68	1	1	869	1018	1131	883
48	116	68	0	1	869	1018	1131	883
49	116	68	1	0	869	1018	1131	883
50	116	68	0	1	869	1018	1131	883
51	116	68	1	0	869	1018	1131	883
52	116	68	0	0	869	1018	1131	883
53	116	68	0	0	869	1018	1131	883
54	116	68	0	0	869	1018	1131	883
55	116	68	1	1	869	1018	1131	883
56	116	68	0	1	869	1018	1131	883
57	116	68	1	1	869	1018	1131	883
58	116	68	0	0	869	1018	1131	883

	Degree Alice	Degree Bob	Bit Alice	Bit Bob	A0B0	A1B0	A0B1	A1B1
59	116	68	0	1	869	1018	1131	883
60	116	68	0	0	869	1018	1131	883

In [ ]:
```
#Plot only the results for the same bases used
df[df['Degree Alice'] == df['Degree Bob']]
```

Out[ ]:

	Degree Alice	Degree Bob	Bit Alice	Bit Bob	A0B0	A1B0	A0B1	A1B1
0	44	44	1	0	955	970	1191	923
1	44	44	0	1	955	970	1191	923
2	44	44	1	1	955	970	1191	923
3	44	44	0	0	1012	954	1169	939
4	44	44	0	0	1040	965	1227	895
5	0	0	1	1	2135	317	339	1340
6	0	0	0	0	2097	350	324	1279
7	68	68	0	1	905	1032	1157	829
8	68	68	0	0	905	1032	1157	829
9	68	68	0	0	905	1032	1157	829
10	68	68	1	1	905	1032	1157	829
11	68	68	1	0	905	1032	1157	829
12	68	68	1	0	905	1032	1157	829
13	68	68	1	0	905	1032	1157	829
14	68	68	1	0	905	1032	1157	829
15	68	68	1	0	905	1032	1157	829
16	68	68	0	1	905	1032	1157	829
17	68	68	1	0	905	1032	1157	829
18	68	68	1	0	905	1032	1157	829
19	68	68	1	0	905	1032	1157	829
20	68	68	1	0	905	1032	1157	829
21	68	68	1	0	869	979	1154	870
22	68	68	1	1	869	979	1154	870
23	68	68	1	0	869	979	1154	870
24	68	68	1	1	869	979	1154	870
30	44	44	0	1	1839	449	627	1292
39	44	44	0	0	1445	637	794	1101
61	0	0	1	1	2135	317	339	1340
62	0	0	0	0	2097	350	324	1279

From this table, we can identify the keys of Alice and Bob, which according to the script are at least of a length of 25 Bits. For Alice: \bold10100100001111110111111110010 For Bob:

$\textcolor{red}{\backslash\text{bold}}01100101001000001000001011010$ In the following we count how many of the digits agree and how many differ:

```
In [ ]: key_Alice = "10100100001111110111111110010"
        key_Bob = "01100101001000001000001011010"

        counter_equal = 0
        for a in range(len(key_Alice)):
            if key_Alice[a]==key_Bob[a]:
                counter_equal +=1
        counter_different = len(key_Alice)-counter_equal
        print("Equal digits: ",counter_equal,"Different digits: ",counter_different)
```

```
Equal digits:  13 Different digits:  16
```

From theory, we expect the number of equal digits to be 0 and the number of different digits to be the length of the generated keys - according to the perfectly anticorrelated nature of the $|\psi^{-}\rangle$ state. However in our experimental setup, we mesured 13 equal digits and 16 different digits. Therefore, we cannot even close see an anticorrelation. Even worse, we seem to have no correlation at all.

This result can be explained by the imperfections in our experimental setup already mentioned. While some external noise factors might also had an influence, the main problem was again the wrongly calibrated waveplate, which lead to the problem, that we measured in bases different than the ones needed for the protocol. The offset by 40° which we observed in the coincidence count rates created an offset, which made our generated keys inusable. In order to proof my assumption, we can consider the correlation function, which according to the script can be computed by using:

$$C = -cos(2|\alpha - \beta|)$$

As the shift due to the waveplate is of a magnitude of $|\alpha - \beta| = 40^{\circ}$, we obtain $C = -0,11$, which is a value very close to zero on the anticorrelated side. This approximatelly agrees with our rations agree:differ, as we observed only slighly more digits which differed from each other.

Theoretically, under perfect experimental circumstances, Alice's key would have been the complement of Bob's, so in order to communicate Alice only needs to sum her key to the message and Bob needs to sum his and invert the obtained (as his key is the complement).

In the next step, we want to check if a spy might have influenced the communication. Therefore, as described, we can use the measurements which where not used for generating the key. With the count rates we can compute the correlation function and the S parameter the same way in which we did in the section of testing Bells inequality:

```
In [ ]: #Compute correlation functions
        df_unterschiedlich = df[df['Degree Alice'] != df['Degree Bob']]
        A0B0 = df_unterschiedlich['A0B0'].values
        A0B1 = df_unterschiedlich['A0B1'].values
        A1B0 = df_unterschiedlich['A1B0'].values
        A1B1 = df_unterschiedlich['A1B1'].values

        #As we performed measurements every 2s
        # and are interested in the rates, we need to normalize
        A0B0 = A0B0/2
        A0B1 = A0B1/2
```

```python
A1B0 = A1B0/2
A1B1 = A1B1/2

C_a_b = C(A0B0[0],A1B1[0],A1B0[0],A0B1[0])
sig_C_a_b = sig_C(A0B0[0],A1B1[0],A1B0[0],A0B1[0])

C_a_bp = C(A0B0[3],A1B1[3],A1B0[3],A0B1[3])
sig_C_a_bp = sig_C(A0B0[3],A1B1[3],A1B0[3],A0B1[3])

C_ap_bp = C(A0B0[2],A1B1[2],A1B0[2],A0B1[2])
sig_C_ap_bp = sig_C(A0B0[2],A1B1[2],A1B0[2],A0B1[2])

C_ap_b = C(A0B0[1],A1B1[1],A1B0[1],A0B1[1])
sig_C_ap_b = sig_C(A0B0[1],A1B1[1],A1B0[1],A0B1[1])

print("Correlation function C_alphabeta = {0} +/- {1}"
.format(np.round(C_a_b,2),np.round(sig_C_a_b,2)))
print("Correlation function C_alphabeta' = {0} +/- {1}"
.format(np.round(C_a_bp,2),np.round(sig_C_a_bp,2)))
print("Correlation function C_alpha'beta' = {0} +/- {1}"
.format(np.round(C_ap_bp,2),np.round(sig_C_ap_bp,2)))
print("Correlation function C_alpha'beta = {0} +/- {1}"
.format(np.round(C_ap_b,3),np.round(sig_C_ap_b,3)))

#Compute S parameter
s_Parameter_2 = S(C_a_b,C_a_bp,C_ap_b,C_ap_bp)
sig_s_Parameter_2 = sig_S(sig_C_a_b,sig_C_a_bp,sig_C_ap_b,sig_C_ap_bp)

print("S-Parameter = {0} +/- {1}"
.format(np.round(s_Parameter_2,2),np.round(sig_s_Parameter_2,2)))
```

```
Correlation function C_alphabeta = 0.92 +/- 0.01
Correlation function C_alphabeta' = -0.61 +/- 0.02
Correlation function C_alpha'beta' = -0.08 +/- 0.02
Correlation function C_alpha'beta = 0.015 +/- 0.023
S-Parameter = 1.47 +/- 0.04
```

We obtain an S parameter under the threshold of 2. Therefore, if Alice and Bob were communicating through our experimental setup, they would think, that they are being intercepted by a spy, as a perfectly anticorrelated state would lead to a value over the classical threshold and should be approximatelly $2\sqrt{2}$ as already discussed. However, they are not being intercerpted by any spy, as we did not place any filter in the experimental setup. The deviation of the measured S-parameter is explained through the wrong calibration of the waveplate. With a perfect calibration they would have got a quantum mechanical S parameter and could assume that they are not being intercepted by a spy according to the no-cloning theorem.

Furthermore, if we compare the estimated S-parameter with the previous one from the coincidence count rate section of $S = 0,98$, we can detect a deviation of around $0,49$. Probably the underlying reason is, that the 2 experiments where conducted at different times. While the coincidence count rates where measured in the morning, the Eckert protocol was done in the afternoon. In between we took a break and the setup was put on standby. In addition, in the afternoon, one of the headsets stopped working because of its battery level, so we had to repeatedly interrupt the experiment and it is possible that those interruptions and possible new calibrations of the waveplate have changed the experimental setup.

2.5 Product-State and influence of a spy

In the last step of the experiment we aim to understand, how the situation previously observed changes when a spy tries to intercept the message. In our experiment, we are going to realize the spy by making use of a polarizing filter at one of the arms. We analyzed the influence of the filter in a qualitative manner together with the tutor.

The core idea behind the polarizing filter is, that when a photon passes it, a measurement in the quantum mechanical sense is performed and therefore the quantum state collapses onto one concrete polarization direction. When performing the bit measurement at Alice and Bobs arm the perfect anticorrelation is broken, as by performing a measurement in one arm, the state in the other arm also collapses. Therefore, after computing the S-parameter with the values displayed we obtained a value which was clearly in the classical range, so we could confirm that a spy was intercepting the message. However, this result was not as surprising, as in the previous steps of the experiment we also obtained values in the classical range (wrong calibration of waveplate).

3. Summary and Discussion

In conclusion, in this experiment we dealed with some of the key methods of quantum cryptography. As some quantum algorithms might become capable of breaking the current encryption, a new communication protocol has been proposed: the Eckert 91 protocol - which we tested in our experiment.

We used the polarization of light as a signal, as we could easily create, manipulate and detect it by using a laser, waveplates or other optical filters. By taking advantage of the non-linear effects of a BBO birefringent crystal, we were able to experimentally entangle the polarization and to create an anticorrelated Bell State $|\psi^-\rangle$.

In the first step, we measured the single count rates and found that after neglecting some experimental deviations the contrast is close to zero, as we were not dealing with an entangled state, but with single photons which could take arbitrary polarizations.

In the next step, we analyzed the coincidence count rates and found a much higher contrast - close to 1, which agreed to our theoretical prediction and verified the entanglement of the fluorescence photons. By measuring the coincidence count rates again, but now at an respective basis angle of 45° to the previous measurement, we could also observe, how the peaks of the coincidence count rates were also shifted to the right by exactly 45°.

This was the first time we were able to detect a systematic error in our experimental setup: the wrong calibration of the waveplates. The first maximum for A0B1 and A1B0 and the minimum of A0B0 and A1B1 of the coincidence count rate was located at around 40° rather than at 0°. We can assume a systematic error, as the graphs agree very precisely in this shift. This error sadly will impact the whole experiment, as the calibration was performed at the beginning and all the measurements were taken with this calibration.

Thirdly, we aimed to test Bells inequality in the CHSH form. Therefore we performed a series of measurements of the coincidence count rates at the bases which maximized the S-parameter. By computing the correlation functions, we were able to calculate S to a value of $S_1 = 0,98 \pm 0,06$. This resulted in a problem, as we wanted to obtain a value over the classical threshold of 2 in order to be able to discard a local realistic theory.

Then, we simulated the Eckert 91 protocol. My partner took the role of Alice and I the one of Bob and we followed the key generation procedure. The generated key was expected to show a perfect anticorrelation, which it did not because of the wrong calibration of the waveplate and did rather show no correlation at all. By taking into account the wrong calibration of 40° we theoretically proofed, that for that case approximately no correlation was expected. As taking into account the wrong calibration into our theoretical predictions described very precisely our observations, the hypothesis of the wrong calibration gains more credibility. Hence, we were not measuring in the maximizing bases and although we might had entangled photons (as proofed in the coincidence count rate section), we were not able to maximize the S parameter and proof Einstein wrong.

The measurements in different bases, which were not used for key generation, were used in order to discard the possibility of having a spy in our experimental setup. Herefore, we calculated the S parameter and obtained $S_2 = 1,47 \pm 0,04$, not a satisfying result, as this would mean a spy was intercepting the message, although there was no type of filter in the setup. Again, this can be explained due to the wrong calibration.

In order to improve the experiment results, the first and most important solution would be to search for an effective way of calibrating the waveplates. In addition, we would need to make sure that the AR-headsets used during the experiment were charged before actually using them. As one of them (mainly at the second part of the experiment), was showing battery problems and turned off a couple of times, we had to interrupt the experiment a few times in order to charge it. Along with the time interruptions there might have been some accidental changes in the software which had an influence on our data. Furthermore, the laser and in general, all the setup was already prepared at the experiment table - in part for security reasons as the laser could be harmful. However, another suggestion to address the question how we could improve our measurements, is to become a bit more familiar with the experimental setup ourselves and not just assume everything to be working well. By this, we could gain a deeper understanding for the systematic errors of our experimental apparatus.

Overall I really had fun preparing the topic and performing the experiment. The setup was very modern and the software used for the experiment very well thought out. However it would have been nice, if we could have checked the correctness of our measurements in an easier way, as the saved files only "appeared" in one section but we could not figure out how to easily open it and sadly were not able to recognize the wrong calibration of the waveplate during the measurements. Although we sadly were not able to experimentally obtain a S parameter over the classical threshold, we learned and gained a strong intuition about which methods can be used to do so and why they work from a theoretical perspective. I want to thank the tutor for his engagement and mentoring during the experiment!