

**ЛАБОРАТОРНЫЙ ПРАКТИКУМ
ПО КВАНТОВОЙ ОПТИКЕ**

Лабораторная работа № 4

ИССЛЕДОВАНИЕ КВАНТОВОГО РАСПРЕДЕЛЕНИЯ КЛЮЧА

Содержание

1 Общие положения	4
1.1 Квантовое распределение ключа.....	4
1.2 Схема распределения квантового ключа на основе интерферометра Маха-Цендера	9
2 Техника безопасности при работе с лазерными источниками излучения.....	12
3. Описание лабораторной установки.....	13
3.1 Двухпроходная автокомпенсационная схема Plug&Play	13
3.2 Ослабление лазерных импульсов	18
3.3 Возможность использования одного детектора одиночных фотонов	20
4. Порядок проведения эксперимента	22
4.1 Настройка времени ожидаемого прихода импульсов на детекторы одиночных фотонов Боба	22
4.2 Настройка напряжения на фазовом модуляторе Боба	27
4.3 Точная настройка временного окна детектора одиночных фотонов.....	29
4.4 Настройка величины задержки приложения напряжения на фазовый модулятор Алисы	31
4.5 Настройка напряжения на фазовом модуляторе Алисы.....	35
4.6 Запуск генерации квантового ключа	39
5 Контрольные вопросы и задания	44
Литература.....	46
Лист регистрации изменений.....	47

Цель работы – на практике ознакомиться с протоколом распределения квантовых ключей BB84 и его реализацией с использованием двухпроходной автокомпенсационной схемы Plug&Play, в том числе, с использованием одного детектора одиночных фотонов.

Приобретаемые компетенции – обучаемые приобретают навыки работы со сложной оптико-электронной системой, осуществляющей квантовое распределение ключей, исследуют характеристики, влияющие на их распределение.

В работе используется: оптическое одномодовое волокно, коннекторы, лазерный модуль, постоянный и переменный оптические аттенюаторы, циркулятор, светоделители, детекторы одиночных фотонов, фазовые модуляторы, линии задержки и хранения, поляризационный светоделитель, синхронизирующий детектор, зеркало Фарадея.

1 Общие положения

1.1 Квантовое распределение ключа

Квантовое распределение ключа — метод передачи ключа, который использует квантовые явления для гарантии безопасной связи. Этот метод позволяет двум сторонам, соединенным по открытому каналу связи, создать общий случайный ключ, который известен только им, и использовать его для шифрования и расшифрования сообщений.

Используя факт существования неизмеримых одновременно величин (квантовый индетерминизм) и состояние суперпозиции, можно осуществлять закрытую передачу данных [1]. Пусть из пункта А в пункт Б требуется передать конфиденциальные данные. В криптографии этому процессу придают некоторую одушевленность, называя передатчик информации Алисой, а приемник – Бобом. Для шифрования данных нужно чтобы и у Алисы, и у Боба были одинаковые ключи, так чтобы при помощи одного из них можно было бы зашифровать данные, а при помощи другого – расшифровать. Проблема в том, как передать ключ от Алисы к Бобу. Если бы был надежный, защищенный от перехвата, способ передачи ключа (в криптографической терминологии – способ распределения ключа), то и не было бы необходимости шифровать данные, их можно было бы передать тем же способом. В данном случае ключ – это просто тоже некоторое количество никому не известной, кроме Алисы и Боба, информации.

Будем рассматривать линейную поляризацию фотона в двумерном гильбертовом пространстве: для того, чтобы задать направление поляризации нам нужна ось, чтобы выделить направление в пространстве и привязать к ней базис. Рассмотрение поляризации в вертикально-горизонтальном базисе будем обозначать значком \oplus , в диагональном-антидиагональном – \otimes . Если фотон поляризован вертикально, т.е. находится в состоянии, описываемом вектором $|\updownarrow\rangle$, и мы проводим измерение в базисе \otimes , то вероятность обнаружить фотон в состоянии $|\nearrow\rangle$ будет равна вероятности обнаружить фотон в состоянии $|\searrow\rangle$, равна 0,5. На рисунке 1 представлено измерение поляризации фотонов в различных базисах как пропускание через фильтр.

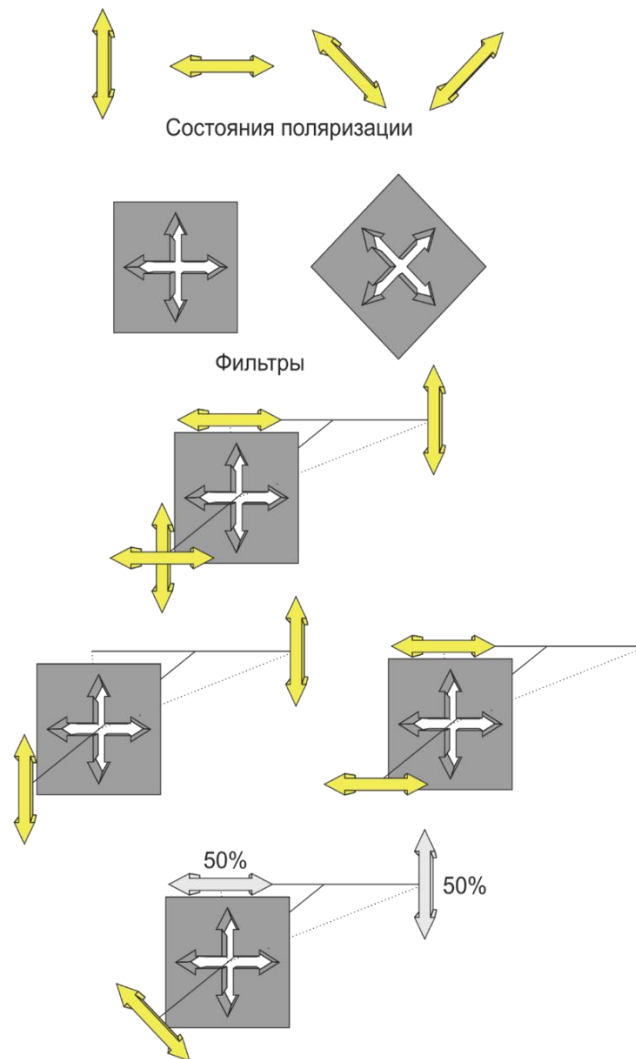


Рисунок 1 Измерение поляризации фотонов в различных базисах как пропускание через фильтр

Математически это выражается следующим образом:

$$\begin{aligned}
 |\updownarrow\rangle &= \frac{1}{\sqrt{2}}|\swarrow\rangle + \frac{1}{\sqrt{2}}|\searrow\rangle \\
 |\leftrightarrow\rangle &= \frac{1}{\sqrt{2}}|\swarrow\rangle - \frac{1}{\sqrt{2}}|\searrow\rangle
 \end{aligned}
 \tag{1}$$

Это означает, что состояние $|\updownarrow\rangle$ состоит из суперпозиции состояний $|\swarrow\rangle$ и $|\searrow\rangle$, причем вероятность обнаружить фотон в состоянии $|\swarrow\rangle$ или $|\searrow\rangle$ равна квадрату множителя (в данном случае $\frac{1}{\sqrt{2}}$) соответствующего состояния. То же

касается и состояния $|\leftrightarrow\rangle$. Здесь знак минус возникает из необходимости некоторого математического согласования, а именно из требования ортогональности. Положение знака минус в этой системе уравнений зависит от выбора направления координатных осей в базисах и выбора направления поворота одного базиса относительно другого. В нашем случае это не играет существенной роли.

Если измерить поляризацию фотона в базисе с неопределенной поляризацией, то в этом базисе она станет определенной, а в другом базисе – перестанет быть определенной.

Алиса, посылая Бобу фотон, должна выбрать базис поляризации. Ограничим этот выбор двумя базисами: Алиса может поляризовать фотон в базисе \oplus (говорят «фотон, приготовленный в базисе \oplus ») или в базисе \otimes . Боб не знает в каком базисе фотон был приготовлен, а Алиса выбирает базисы приготовления случайно. Теперь Боб может провести измерения поляризации фотона как в \oplus , так и в \otimes базисе. Базис измерения Боб тоже выбирает случайно. Если базисы приготовления и измерения поляризации случайно совпадут, то и у Алисы, и у Боба будет одинаковая информация о поляризации фотона. Если Боб случайно выберет для измерения другой базис, не тот в котором фотон приготовила Алиса, то результат измерения поляризации будет случайным и не несущим информацию. Договариваемся, что состоянию $|\updownarrow\rangle$ будет соответствовать логическая 1, состоянию $|\leftrightarrow\rangle$ – логический 0, также и для состояний $|\nearrow\rangle$ и $|\searrow\rangle$ – 1 и 0. Таким образом, Алиса пересылает Бобу серию фотонов. Потом они по открытому каналу сверяют базисы и выбрасывают все те измерения, которые были проделаны в базисах, не совпадающих с базисами приготовления. Таким образом, у Алисы и у Боба должны быть одинаковые последовательности битов, при этом они секретные: результаты своих измерений Боб, разумеется, не сообщает. Этот метод в известной степени бесполезен для передачи информации: Алиса не знает заранее какие биты из последовательности будут выброшены, но он отлично подходит для передачи случайной битовой последовательности, которую можно

будет использовать для шифрования данных. Таким образом можно осуществлять квантовое распределение ключа.

Практическая реализация квантового распределения ключа с поляризационным кодированием может быть осуществлена с использованием устройства, которое управляемо поворачивает поляризацию (ячейка Погкельса) и поляризационного светоделителя PBS [2]. При помощи ячейки Погкельса Боб может осуществлять выбор между базисами \oplus и \otimes , подавая на ячейку соответствующий электрический сигнал. Далее оптический сигнал разделяется поляризационным светоделителем на две взаимно ортогональных поляризационных составляющих, каждая из которых приходит на свой детектор. Если базис, выбранный Бобом, совпадает с базисом Алисы, то должен сработать детектор, соответствующий поляризации светового импульса, номер детектора определит значение бита полученной информации. Если базис был выбран неправильно, то случайным образом сработает один из детекторов. Конечно, если в импульсе не один фотон, а больше, то может сработать два детектора. Собственно, мы даже можем судить по тому, сколько детекторов сработало, правильно или неправильно мы выбрали базис измерения. Но нам нельзя знать заранее насколько правильно мы выбрали базис (точнее не нам нельзя, а потенциальному противнику нельзя дать возможность определить правильность выбора базиса измерения), и для этого нам обязательно нужны однофотонные импульсы.

Теперь введем третий персонаж, который будет подслушивать закрытый канал, традиционно это Ева, от английского eavesdropper – подслушивающий. Точно так же, как и Боб, Ева может измерить поляризацию фотона, но также, как и Боб она не знает в каком базисе он был приготовлен. После измерения фотона Ева пересылает его дальше Бобу (ей же нельзя себя обнаружить). Однако, если она выбирает для измерения не тот базис, в котором фотон приготовила Алиса, то она меняет базис приготовления фотона и, таким образом, в ключ Боба закрадываются ошибки. Правильнее так: количество ошибок в ключе Боба возрастает, потому что когда передача идет на однофотонном уровне, то ошибки возникают неизбежно. У Евы с Бобом базисы различаются – и тот и другой выбирают их случайным образом. Таким образом, у Евы получается меньше полезной информации, чем у

Боба. Ключевым моментом здесь является то обстоятельство, что Еву можно обнаружить по возрастанию количества ошибок в ключе Боба. Пожалуй, это и есть самое важное свойство квантовой криптографии – Ева не может скрыть своего присутствия.

Схематично процедуру квантового распределения ключа можно свести к таблице 1:

Таблица 1

1. Базис Алисы	⊗	⊕	⊕	⊗	⊕	⊗	⊗	⊕	⊗	⊗	⊕
Значение бита Алисы	1	0	1	0	0	1	0	1	1	1	1
Алиса посылает	$ \nearrow\rangle$	$ \leftrightarrow\rangle$	$ \downarrow\rangle$	$ \nwarrow\rangle$	$ \leftrightarrow\rangle$	$ \nearrow\rangle$	$ \nwarrow\rangle$	$ \downarrow\rangle$	$ \nearrow\rangle$	$ \nearrow\rangle$	$ \downarrow\rangle$
2. Базис Боба	⊗	⊕	⊗	⊕	⊕	⊗	⊗	⊗	⊕	⊕	⊕
Значение бита Боба	1	0	0	0	0	1	0	1	0	1	1
3. Тот же базис?	да	да	нет	нет	да	да	да	нет	нет	нет	да
Ключ Алисы	1	0			0	1	0				1
Ключ Боба	1	0			0	1	0				1

Упрощённый алгоритм – протокол распределения квантового ключа – можно представить следующим образом:

1. Алиса случайным образом выбирает базис и поляризацию своих однофотонных импульсов и посылает их Бобу.

2. Для каждого импульса Боб также случайным образом выбирает базис, в котором он измеряет поляризацию импульса. Он либо получает значение бита 0 или 1, либо ничего не регистрирует из-за потерь связи при детектировании. Последовательность всех полученных битов называется сырым ключом.

3. Боб использует открытый канал (например, Интернет), чтобы сообщить Алисе номера измерений, в которых было срабатывание одного детектора и каком базисе проводились эти измерения. При этом Боб не сообщает результат измерения. В тех случаях, когда Алиса и Боб использовали один и тот же базис ⊕ или ⊗, они должны получить одинаковые биты. Последовательность этих битов называется просеянным ключом. При этом, из-за несовершенства процесса

измерения и из-за потенциального подслушивания, возникнет некоторое количество ошибок.

4. Чтобы преобразовать свои частично испорченные, и, возможно, не вполне секретные строки в пригодный к использованию ключ, Алисе и Бобу теперь нужна некоторая обработка. Это выходит за рамки данной лабораторной работы, но, для общего ознакомления, основные шаги таковы: оценить уровень ошибок при передаче; сделать предположение о максимальном количестве информации, которая могла утечь из-за подслушивания; и затем скорректировать все ошибки, в то же время уменьшая количество информации, потенциально доступное Еве, до требуемого уровня. Оставшиеся биты и есть криптографический ключ.

Описанный протокол распределения квантового ключа называется BB84, его изобрели Чарльз Беннет и Жиль Брассард, соответственно, в 1984 году [3]. Протокол нашел свое первое практическое воплощение как раз с использованием поляризованных фотонов. Однако, использование именно этой схемы кодирования, поляризационной схемы, не очень удобно для передачи сигнала по оптоволоконным линиям связи – оно затруднено деполяризациями и флуктуациями двулучепреломления. Гораздо удобнее осуществлять распределение квантового ключа с использованием фазового кодирования.

1.2 Схема распределения квантового ключа на основе интерферометра Маха-Цендера

В описанном выше поляризационном способе кодирования битов информации значение бита определяется поляризацией фотона: поляризован ли он по оси X или Y, ось Y повернута относительно оси X на $\pi/2$, и чтобы индетерминизм работал на нас, мы рассматриваем это в двух максимально ортогональных базисах, один из которых повернут относительно другого на $\pi/4$. С идейной точки зрения, фазовое кодирование аналогично поляризационному. Только рассматривается не поворот плоскости поляризации, а сдвиг импульса по фазе. Здесь тоже есть два базиса, но один из базисов сдвинут на $\pi/2$

относительно другого, а базисные оси отстоят друг от друга по фазе на π . В этой схеме оптический сигнал раскладывается на два импульса, которые потом сходятся вместе и интерферируют. Фазовое кодирование реализуется с использованием интерферометра Маха-Цендера, упрощённо представленного на рисунке 2. И у Алисы и у Боба есть фазовый модулятор. Алиса при помощи фазового модулятора создает разность фаз между импульсами. При соответствующей калибровке интерферометра (по длине оптических линий), если разность фаз между импульсами равна 0, то на детекторе SPD 1 создается конструктивная интерференция – интерференционный максимум, что соответствует значению бита 0. При этом на детекторе SPD 2 реализуется интерференционный минимум – деструктивная интерференция.

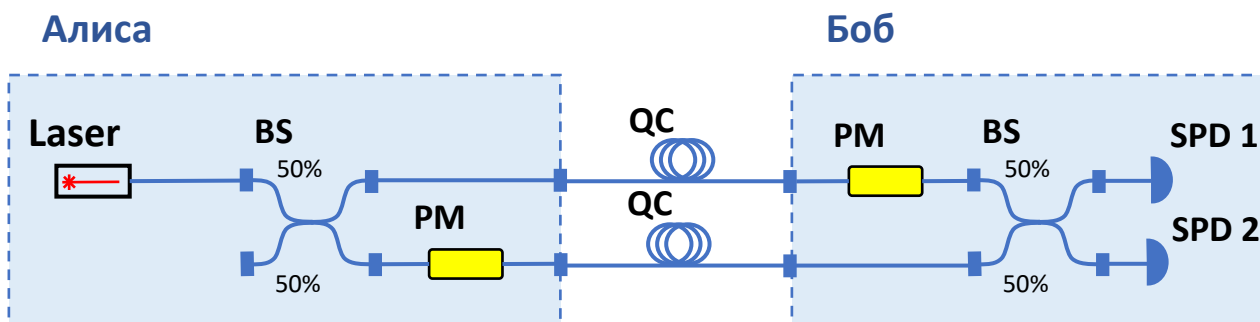


Рисунок 2 Схема реализации фазового кодирования в интерферометре Маха-Цендера

Если Алиса создает фазовый сдвиг равный π , то на детекторе SPD 1 будет деструктивная интерференция, а на детекторе SPD 2 – конструктивная, и это будет соответствовать значению бита 1. Т.е., в зависимости от сдвига фазы, у Боба будет срабатывать либо детектор SPD 1 либо детектор SPD 2, что будет означать либо получение бита 0 или 1.

Теперь надо добавить еще один базис, чтобы сделать передачу ключа секретной. Это обеспечивается введением со стороны Алисы еще двух фазовых сдвигов: $\pi/2$ и $3\pi/2$. Получается, что фазовые сдвиги 0 и π соответствуют одному базису, для определенности, базису \oplus , а сдвиги $\pi/2$ и $3\pi/2$ соответствуют базису \otimes . Боб, в свою очередь, может решать в каком базисе проводить измерения.

Если измерять интерференцию импульсов как они пришли, то это будет измерение в базисе \oplus , а если сдвинуть по фазе один из импульсов на $\pi/2$, то это будет измерение в базисе \otimes . Разумеется, и Боб и Алиса выбирают базис измерения и базис приготовления совершенно случайно. Таким образом, сохраняется вся последовательность действий протокола BB84, представленная в таблице 1.

Надо сказать, что описанная схема распределения ключа, в том виде, как она представлена, так же неудобна в использовании, поскольку очень чувствительна к изменению длин всех оптических линий. Например, малейшее изменение температуры одной из линий QC меняет параметры интерферометра и там, где был интерференционный минимум, может внезапно стать интерференционный максимум и наоборот. Поэтому в лабораторной работе будет использоваться усовершенствованный вариант схемы, который представлен ниже.

2 Техника безопасности при работе с лазерными источниками излучения

При выполнении лабораторных работ, связанных с использованием когерентных лазерных источников излучения, необходимо соблюдать следующие правила техники безопасности:

1. Не смотреть в выходной порт источника, на торцы коннекторов патч-кордов или оптических адаптеров.
2. Контроль чистоты и исправности оптического коннектора или адаптера допускается только при отсутствии в волокне излучения.
3. Для определения активности оптического волокна необходимо использовать измеритель оптической мощности или специальный индикатор излучения.



Внимание! Лазерное излучение, используемое в телекоммуникационных системных и измерительных приборах, невозможно обнаружить визуально!

3. Описание лабораторной установки

Лабораторная работа выполняется с использованием Установки академической модульной квантовой оптики EMQOS 1.0. Исследуемая оптическая схема реализуется на оптических столиках блоков посредством оптико-электронных компонентов из комплекта поставки.

3.1 Двухпроходная автокомпенсационная схема Plug&Play

На рисунке 3 представлена исследуемая двухпроходная автокомпенсационная оптическая схема Plug&Play, наиболее удобная с точки зрения простоты настройки параметров для распределения квантового ключа.

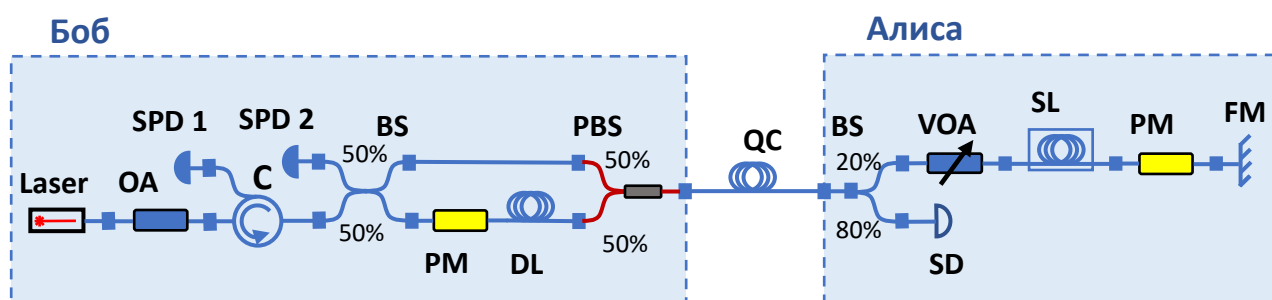


Рисунок 3 Автокомпенсационная двухпроходная схема квантового распределения ключа “Plug&play”

На схеме обозначены:

Laser – лазерный модуль;

OA – optical attenuator – оптический аттенюатор;

C – circulator – оптический циркулятор;

SPD – single photon detector – детектор одиночных фотонов;

SD – synchro detector – синхронизирующий детектор;

BS – beam splitter – неполяризационный светоделитель;

PBS – polarizing beam splitter – поляризационный светоделитель;

PM – phase modulator – фазовый модулятор;

DL – delay line – линия задержки;

VOA – variable optical attenuator – переменный оптический аттенюатор;

SL – storage line – линия хранения;

QC – quantum channel – квантовый канал.

Рассмотрим алгоритм распределения квантового ключа и последовательность событий, происходящих с оптическими импульсами в схеме Plug&Play.

1. Лазерный модуль, расположенный в блоке Боба, генерирует последовательность одинаковых многофотонных оптических импульсов, которую мы будем называть трейном. В нашем случае длина волны оптических импульсов $\lambda = 1550$ нм, импульсы линейно поляризованы. Частота следования лазерных импульсов – это одна из основных характеристик квантового распределения ключа. Обозначим эту величину как ν_0 , соответствующий ей период – T_0 , эти промежутки времени мы будем называть тактами. В трейне n импульсов и трейны следуют с периодом T_t – период трейна. Всего за сеанс формируется пакет из N трейнов. Промежуток между трейнами такой, что трейн успевает пройти расстояние от Боба до Алисы и обратно, и только после этого запускается следующий. О факторах, ограничивающих в данной оптической схеме частоту ν_0 и количество импульсов в трейне, будет сказано ниже. Схема распространения импульсов представлена на рисунке 4.

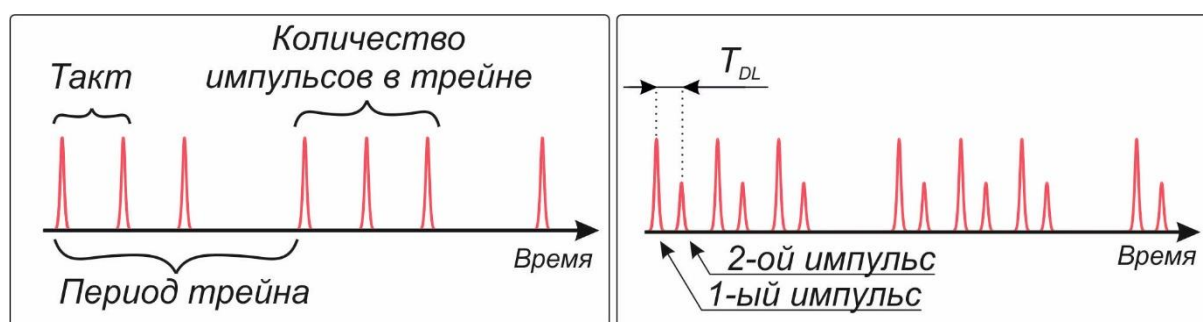


Рисунок 4 Схема распространения импульсов на выходе из лазерного модуля и на выходе из поляризационного светоделителя

2. Блок Боба собран из волокна, сохраняющего поляризацию (PM-волокно). Трейн лазерных импульсов попадает на циркулятор (С), который направляет лазерное излучение на светоделитель. В результате прохождения светоделителя

каждый импульс разделяется пополам и направляется в соответствующие плечи интерферометра.

3. В одном из плечей интерферометра расположена линия задержки (DL) и фазовый модулятор (PM), поэтому на поляризационный светоделитель импульсы, прошедшие через это плечо, приходят позже, следовательно, на выходе из поляризационного светоделителя получается в 2 раза больше импульсов, чем в первоначальном трейне (рисунок 4). Время запаздывания определяется формулой:

$$t_{DL} = \frac{n_0 \Delta l}{c_0}, \quad (2)$$

где Δl – длина линии задержки, $n_0 \approx 1.47$ коэффициент преломления, c_0 – скорость света. На практике удобно использовать вытекающее из этой формулы соотношение: за 5 нс световой импульс проходит 1 м оптоволокна. Из-за описанного смещения импульсы удобно рассматривать парами. Импульсы, преодолевшие большее плечо, имеют меньшую интенсивность из-за потерь, возникающих в линии задержки и фазовом модуляторе. Отметим, что импульсы, выходящие из поляризационного светоделителя Боба (PBS), линейно поляризованы, причем их поляризация попарно ортогональна (за счёт смещения фазы на первом, неполяризационном светоделителе). Дальше трейн покидает Боба и через квантовый канал (QC) направляется к Алисе. Фазовый модулятор в длинном плече интерферометра Боба при прохождении импульсов в сторону Алисы не задействуется.

4. Приходя на оптическую схему Алисы, лазерные импульсы делятся на две неравные части: бóльшая часть оптической мощности идет на синхронизирующий детектор (SD), а меньшая часть используется дальше для обработки. Вся оптическая схема Алисы, как, впрочем, и квантовый канал, состоит из неподдерживающего поляризацию одномодового оптоволокна.

5. Синхродетектор SD служит для синхронизации всех быстрых процессов, в нашем случае – фазовой модуляции. Синхродетектор обозначает время прихода лазерных импульсов на оптическую схему Алисы. Теперь можно вовремя подать электрические сигналы на фазовый (PM) и амплитудный (VOA) модуляторы Алисы – с соответствующей задержкой.

6. До фазового модулятора Алисы импульсы еще должны пройти накопительную линию (SL). Накопительная линия служит для того, чтобы вместить в себя все импульсы трейна. Собственно, длина накопительной линии и определяет количество импульсов в трейне, поэтому накопительная линия длинная – десятки километров. При движении трейна от Боба к Алисе неизбежно возникают множественные отражения на оптических неоднородностях, в основном на оптических разъемах. Эти отражения идут обратно к Бобу, и могут вызвать ложные срабатывания детекторов, информативным же является только сигнал, полученный при отражении трейна от зеркала Фарадея. Следовательно, для защиты от ложных срабатываний необходимо разделить по времени паразитные отражения и отражение от зеркала Фарадея, при этом не должно быть наложения, например, отражений от хвостовых импульсов трейна и уже возвращающихся обратно головных импульсов трейна. Для этого и служит накопительная линия, причем трейн может быть до 2 раз длиннее накопительной линии:

$$nT_0 \leq 2 \frac{n_0 L_{SL}}{c_0}, \quad (3)$$

где L_{SL} – длина накопительной линии. При расчете количества импульсов в трейне необходимо округлять число импульсов в меньшую сторону. Период трейна ограничен снизу длиной квантового канала + накопительная линия. Это ограничение накладывается по той же причине: не должно быть наложения импульсов идущих туда и обратно. Т.е. необходимо дождаться последнего импульса в трейне и только потом посылать новый. При расчете периода трейна необходимо округлять значения в большую сторону.

7. Между накопительной линией и зеркалом Фарадея на один из импульсов каждой пары накладывается фазовый сдвиг фазовым модулятором Алисы, далее отраженные зеркалом Фарадея импульсы возвращаются обратно в накопительную линию. Отражаясь от зеркала Фарадея, лазерный импульс меняет свою поляризацию на ортогональную. Следует отметить, что фазовый сдвиг необходимо накладывать только на один импульс в паре, но при этом этот сдвиг должен накладываться на пути следования и туда, и обратно. Это диктуется особенностями работы фазового модулятора: дело в том, что фазовый модулятор сдвигает только

одну компоненту поляризации, а состояние поляризации у лазерных импульсов может быть непредсказуемым, поскольку они прошли по квантовому каналу, претерпели различные поляризационные искажения, и не известно какое состояние поляризации заходит в фазовый модулятор. Но после отражения от зеркала Фарадея поляризация меняется на ортогональную, и если фазовый модулятор сдвинул одну компоненту поляризации в своем базисе, то на обратном пути он сдвинет как раз другую компоненту. Эти обстоятельства диктуют следующее ограничение: между входом фазового модулятора и зеркалом Фарадея должен находиться только один импульс из пары. Таким образом, получается, что время прохода импульсом расстояния от входа фазового модулятора и обратно не должно превышать времени между оптическими импульсами в паре. Если, к примеру, период следования лазерных импульсов $T_0 = 100$ нс, и при этом после прохождения большего плеча интерферометра Боба второй импульс сдвигается на полпериода, то длина оптоволокну между фазовым модулятором и зеркалом Фарадея не должна превышать 5 метров.

8. От Алисы трейн импульсов направляется по квантовому каналу к Бобу. При этом поляризационные преобразования, которые трейн испытывал по пути к Алисе, компенсируются. В этом заключается преимущество схемы Plug&Play – нет необходимости в постоянной подстройке оптической схемы для компенсации искажений. Но у этой схемы есть и недостатки, например, из-за необходимости проводить передачу трейнами, существенно падает скорость распределения ключа.

9. Поскольку импульсы в трейне, будучи взаимно попарно ортогонально поляризованными, изменили свою поляризацию на противоположную при отражении от зеркала Фарадея, то теперь они проходят каждый по другому плечу интерферометра Боба. Поляризационный светоделитель направляет импульс в длинное плечо, если до этого он прошел по короткому плечу, и наоборот. Если оба фазовых модулятора – Алисы и Боба – не прикладывают сдвига, то на SPD2 реализуется конструктивная интерференция, а на SPD1 – деструктивная. Два базиса обеспечивают секретность передачи ключа. Алиса при помощи фазового модулятора прикладывает четыре фазовых сдвига: 0 , $\frac{\pi}{2}$, π и $\frac{3\pi}{2}$. Получается,

что фазовые сдвиги 0 и π соответствуют одному базису, например \oplus , а сдвиги $\frac{\pi}{2}$ и $\frac{3\pi}{2}$ соответствуют базису \otimes . Боб решает в каком базисе проводить измерения, либо не сдвигая фазу (базис \oplus), либо сдвигая фазу на $\frac{\pi}{2}$, тогда это будет измерение в базисе \otimes . И Боб и Алиса выбирают базис измерения и базис приготовления совершенно случайно.

3.2 Ослабление лазерных импульсов

Квантовое распределение ключа, основанное на протоколе BB84, требует, чтобы оптические импульсы были однофотонными. Если импульсы содержат больше одного фотона, то Ева может проводить атаку разделением числа фотонов и ключ будет скомпрометирован. Из-за сложности создания эффективного источника одиночных фотонов, в реальных условиях квантовые состояния не являются однофотонными. Вместо однофотонных состояний на практике используются ослабленные лазерные импульсы, т.е. когерентные состояния со средним числом фотонов $\mu = 0.1-0.5$ фотона/импульс. При малой интенсивности распределение числа фотонов в импульсе подчиняется статистике Пуассона:

$$\mathbb{P}(n_p, \mu) = \frac{\mu^{n_p}}{n_p!} \exp(-\mu), \quad (4)$$

где n_p – число фотонов в импульсе. На рисунке 5 представлено распределение числа фотонов для разных значений μ .

На рисунке видно, что если у нас в среднем один фотон на импульс, то около 37% импульсов у нас не содержат ни одного фотона, столько же по одному фотону, без малого 20% - два фотона, более 5% - три фотона, и даже у четырёх фотонных импульсов очень заметен вклад. Такое положение вещей совершенно неприемлемо – слишком много импульсов, на которых можно провести атаку расщеплением числа фотонов. Если же использовать импульсы с $\mu = 0.1$, то, как видно из графика, число импульсов, содержащих два фотона и больше пренебрежимо мало. Правда за это приходится платить тем, что более 90%

импульсов не содержат ни одного фотона вообще, из-за чего скорость распределения ключа сильно падает.

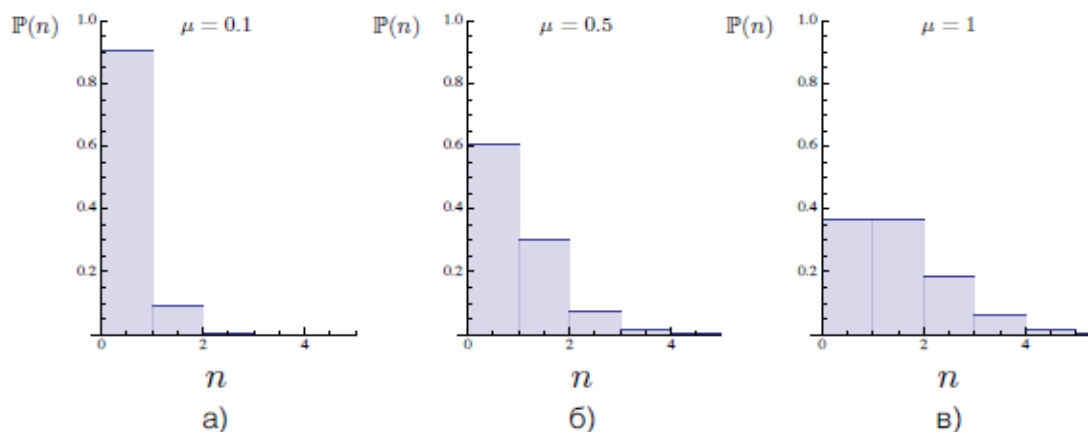


Рисунок 5 Статистика Пуассона лазерного излучения: зависимость вероятности числа фотонов n в импульсе интенсивностью μ

Лазерные импульсы необходимой мощности получают при помощи переменного оптического аттенюатора (VOA), который ослабляет трейн два раза: на пути туда и обратно. Для того, чтобы установить необходимое ослабление на аттенюаторе, например такое, чтобы от Алисы выходили импульсы в среднем, скажем, 0.1 фотона/импульс, необходимо с помощью измерителя мощности определить мощность лазерных импульсов, выходящих из светоделителя Алисы. Но это измерение осложнено тем, что импульсы изначально слабые, а нам нужны еще сильнее ослабленные импульсы и все это выходит за пределы измерений измерителя мощности. Поэтому надо померить более мощные импульсы, а затем дополнительно ослабить мощность в нужное число раз. Лазерный модуль подключен к циркулятору (C) через аттенюатор (OA) или несколько аттенюаторов. Это сделано для того, чтобы не слепить SPD1 мощным светом, что может даже вывести его из строя. Для ослабления лазерных импульсов можно предложить следующую последовательность действий:

1. Отключить оба детектора одиночных фотонов, для надежности лучше отсоединить оптические разъемы.



Обязательно отключите детекторы одиночных фотонов SPD, чтобы не вывести их из строя!

2. Используя источник лазерного излучения измерить коэффициенты затухания в оптических схемах Боба и Алисы. В качестве источника лазерного излучения можно использовать лазер, который установлен в схеме Боба, включённый в режиме непрерывной генерации импульсов. Для расчета коэффициента затухания в оптической схеме используется формула:

$$\alpha_{att} = 10 \cdot \lg \left(\frac{P_{in}}{P_{out}} \right), \quad (5)$$

где P_{in} – мощность излучения, входящего в оптическую схему, P_{out} – мощность излучения на выходе.

3. Мощность при которой в импульсе содержится необходимое нам среднее число фотонов μ , рассчитывается по формуле

$$P_0 = \frac{c_0 \nu_0 h \mu}{\lambda}, \quad (6)$$

где λ - длина волны лазерного излучения, в нашем случае $\lambda = 1550$ нм.



Детекторы одиночных фотонов подключаются к оптической схеме после подключения лазерного модуля через аттенюатор!

3.3 Возможность использования одного детектора одиночных фотонов

Протокол BB84 позволяет осуществлять квантовое распределение ключа с использованием только одного детектора одиночных фотонов (ДОФ). Для этого Бобу необходимо случайным образом выбирать не только базис измерения, но и то, что он собирается проверить в данном базисе – «0» или «1». Если он угадал и с базисом и с битом, то детектор кликнет. Если не угадал с базисом, то детектор кликнет с вероятностью 50%. Если Боб угадал с базисом, но не угадал с битом, то детектор не кликнет, но такие случаи все равно будут выброшены. Данная логика приведена без учета того, что фотона может не быть, что детектор неидеален, конечно, надо и эти обстоятельства иметь в виду.

В случае схемы “Plug and play” на фазовый модулятор Боба случайным образом подается не два значения напряжения, как в случае работы с одним ДОФ-ом, а также как и у Алисы четыре значения, соответствующих сдвигам по фазе 0 , $\pi/2$, π и $3\pi/2$. Для тех событий, в которых Боб зафиксировал клик, он сообщает Алисе по открытому каналу базис измерения: базис I – это сдвиги 0 или π , базис II – сдвиги $\pi/2$ или $3\pi/2$. Информацию, какой бит проверялся в данном базисе Боб оставляет в секрете. После этого Алиса и Боб оставляют себе биты в совпадающих базисах и, таким образом, получают просеянный ключ. При распределении ключа с использованием одного ДОФ ключ получается, по крайней мере, в 2 раза короче, чем при использовании двух ДОФ.

Процедуры настройки для работы с одним ДОФ ничем принципиально не отличаются от таковых для двух ДОФ. Единственное отличие – точная настройка временного окна проводится только для одного детектора, например, для SPD1. Для запуска генерации ключа как у Алисы, так и у Боба хостовая программа переводится в режим работы с одним детектором.

4. Порядок проведения эксперимента

4.1 Настройка времени ожидаемого прихода импульсов на детекторы одиночных фотонов Боба

Включить блоки Алиса, Боб и управляющие ПК. На управляющем ПК Боба запустить файл проекта *Project Bob.lvproj* из состава поставляемого программного обеспечения. В появившемся диалоговом окне (рисунок 6) запустить файл: *\Bob Host Library\Examples\Low Level\rqcBob Channel Test (Low Level Example).vi*.

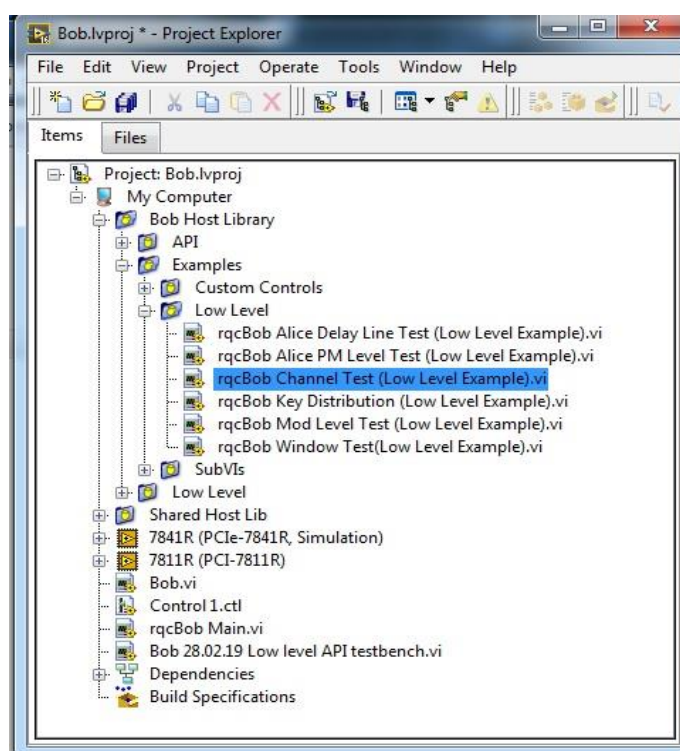


Рисунок 6 Запуск программы *rqcBob Channel Test.vi*

Произвести расчёт периода следования трейна оптических импульсов (для одного импульса в трейне) по формуле

$$T_t = \frac{2(L_{QC} + L_{SL}) \times n_0}{c_0}, \quad (7)$$

где L_{QC} – длина оптического канала, L_{SL} – длина накопительной линии, n_0 – коэффициент преломления в оптическом волокне (для длины волны 1550 нм $n_0 = 1,47$), c_0 – скорость света в вакууме.

Пример – Для оптического канала длиной 10 км и накопительной линии 25 км период трейна будет равен

$$T_t = \frac{2 \times (10^4 \text{ м} + 2,5 \times 10^4 \text{ м}) \times 1,47}{3 \times 10^8 \text{ м} / \text{с}} = 3,43 \times 10^{-4} \text{ с} = 343 \text{ мкс}$$

Определить такт в котором импульс возвращается к Бобу разделив полученное значение периода трейна на величину такта (200 нс) и округлить полученное значение до тысяч в большую сторону. Внести полученное значение в поле переменной *train.period* группы параметров *train.configuration* графического интерфейса программы *rqcBob Channel Test* (рисунок 7).

Пример – Для полученного нами ранее значения периода трейна, такт в котором импульс вернется к «Бобу» будет равен

$$\text{train.period} = \frac{343 \times 10^{-6} \text{ с}}{200 \times 10^{-9} \text{ с}} = 1,715 \times 10^3 = 1715 \rightarrow 2000$$

В остальных полях графического интерфейса программы *rqcBob Channel Test* (рисунок 7) задать следующие параметры:

- *train.num of pulses* – 1 (один импульс в трейне);
- *train.mode* – train;
- *train.num of trains* – 4000 (генерация 4 000 трейнов за сеанс);
- *train.first pulse delay mod* – every train;
- *train.first pulse delay*– 55;
- *modulator.start* – 1500 (такт начала работы модулятора);
- *modulator.stop* – 2000 (такт окончания работы модулятора);
- *window.open* – 0;
- *window.close* – 2000 (значение переменной *train.period*);
- *mod.DAC parameters* – constant;
- *window.close* – 2000 (значение переменной *train.period*);
- *mod.DAC parameters* – constant;
- *mod.DAC levels values* – 150;
- *mod.rnd gen initial seed* – 00000000;
- *mod.clock delay* – constant.

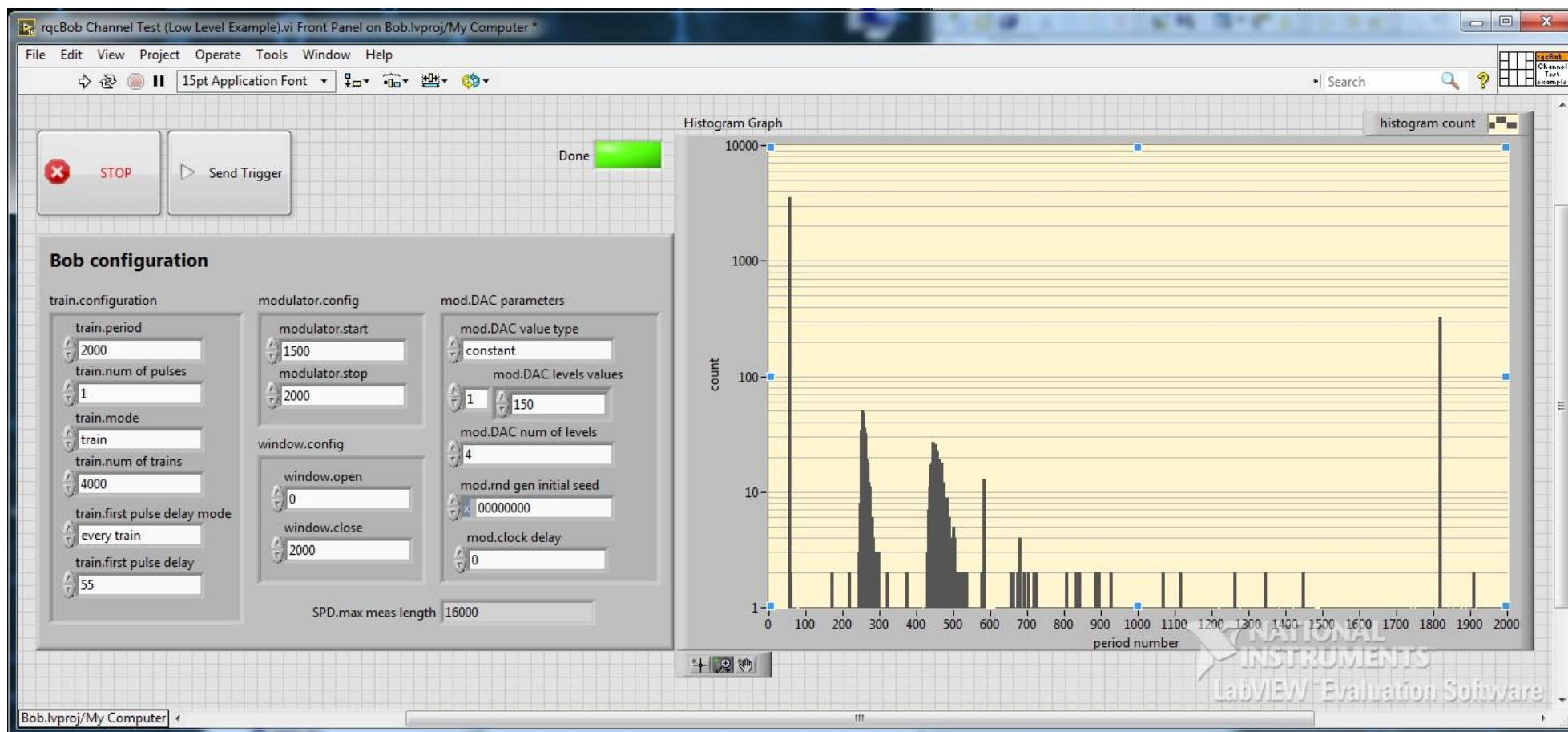


Рисунок 7 Интерфейс программы *rqcBob Channel Testi*

Запустить программу *rqcBob Channel Test* на выполнение, в правой части диалогового окна будет построена диаграмма, на которой по оси абсцисс отражены номера тактов (*period number*), а по оси ординат число срабатываний ДОФ.

Последний импульс с высокой амплитудой на гистограмме – отражение от крайнего элемента в оптической цепи Алисы – зеркала Фарадея.

Примечание – Импульсы на гистограмме могут удваиваться, это происходит по причине регистрации импульсов ДОФ в более мелких временных окнах, чем отображает шкала гистограммы, при этом импульс может быть зарегистрирован в 2 разных окнах.

Пример – На рисунке 7 представлен результат работы программы *rqcBob Channel Test*, отражение от зеркала Фарадея в оптической цепи Алисы между тактами 1800 и 1900.

Увеличить масштаб гистограммы для точного определения значения такта отражения от зеркала Фарадея и внести полученное значение в поле *window.open*.

Увеличить число импульсов в трейне (поле *train.num of pulses*) до 20, и вновь запустить программу. Убедиться, что такт отражения первого импульса трейна от зеркала Фарадея остаётся прежним (рисунок 8).

Пример – На рисунке 8 отражение первого импульса трейна от зеркала Фарадея в оптической цепи Алисы соответствует такту 1817.

Установить значение поля *train.num of pulses* = 1. Значение поля *window.close* установить равным значению *window.open* + 1, запустить программу на выполнение. Убедиться, что такт отражения от зеркала Фарадея остаётся прежним.

Увеличить значения полей *window.close* и *window.open* на единицу. Запустить программу, убедиться, что регистрация отражения импульсов от зеркала Фарадея не происходит.

Полученное значение *window.open* есть искомый такт ожидаемого прихода импульсов на ДОФ Боба, который следует использовать во всех последующих подпрограммах в поле открытия временного окна *window.open*.

Значение поля *window.close*, определяющего такт закрытия временного окна срабатывания ДОФ после прихода последнего импульса в трейне (при числе

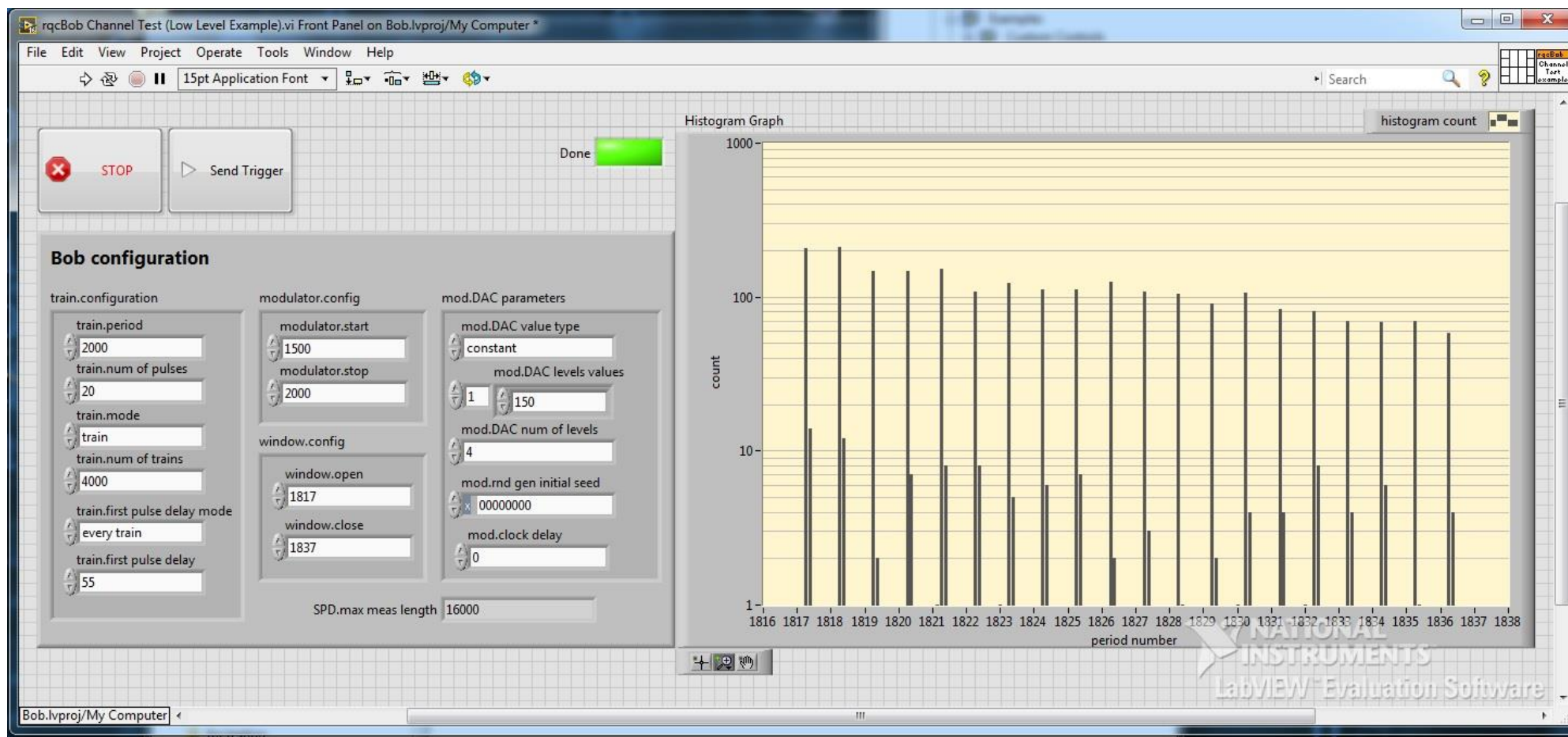


Рисунок 8 Результат работы программы *rqcBob Channel Test*. Отражение первого импульса трейна от зеркала Фарадея в оптической цепи Алисы соответствует такту 1817

импульсов в трейне более единицы), указывается номер такта, полученный суммированием числа импульсов в трейне *train.num of pulses* со значением поля *window.open*.

4.2 Настройка напряжения на фазовом модуляторе Боба

На управляющем ПК Боба запустить файл проекта *Project Bob.lvproj* из состава поставляемого программного обеспечения. В появившемся диалоговом окне (рисунок 6) запустить файл: *\Bob Host Library\Examples\Low Level\rqcBob Mod Level Test (Low Level Example).vi*.

В появившемся графическом интерфейсе программы задать рассчитанный ранее в п. 4.1 период трейна (значение поля *train.period*) и полученный такт ожидаемого прихода импульсов на детекторы одиночных фотонов Боба *window.open*. В остальных полях графического интерфейса программы *rqcBob Mod Level Test* (рисунок 9) задать следующие параметры:

- *train.num of pulses* – 20;
- *train.mode* – train;
- *train.num of trains* – 1000 (генерация 1 000 трейнов за сеанс);
- *train.first pulse delay mod* – every train;*train.first pulse delay*– 55;
- *modulator.start* – 1000 (такт начала работы модулятора);
- *modulator.stop* – 2700 (такт окончания работы модулятора);
- *window.close* – 1837 (будет автоматически рассчитано на основе значений переменных *window.open* и *train.num of pulses*);
- *SPD1.detection window* – 0;
- *DAC Level min* – 0;
- *DAC Level max* – 512;
- *DAC interval* – 8;
- *mod.DAC parameters* – constant;
- *mod.DAC levels values* – 512;
- *mod.DAC num of levels* – 0.

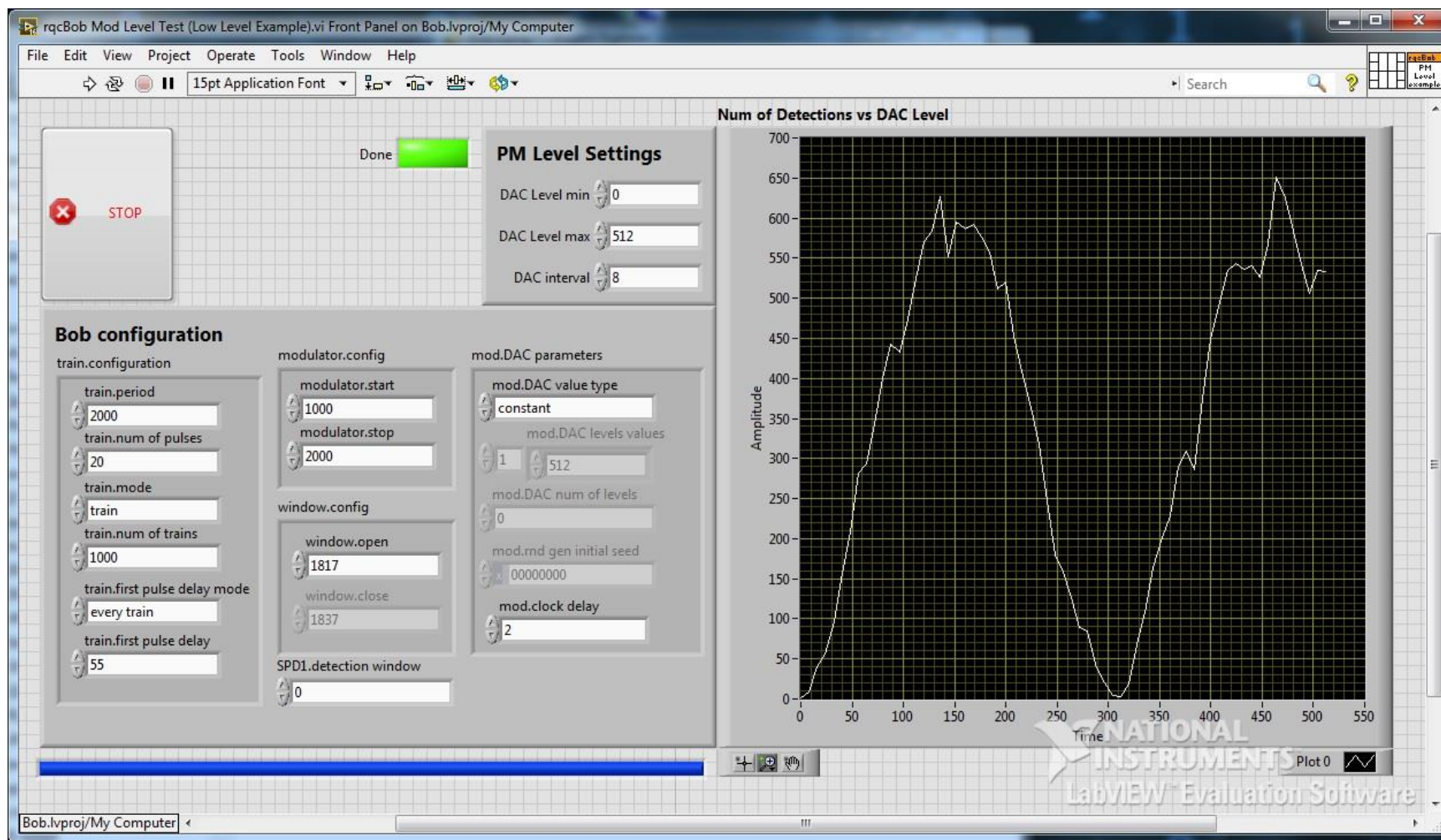


Рисунок 9 Результат работы программы *rqcBob Mod Level1 Test*. Значение *pm.DAC levels values* равно 150 относительных единиц

Запустить программу *rqcBob Mod Levell Test* на выполнение, в правой части диалогового окна будет построена диаграмма, на которой по оси абсцисс в относительных единицах отражена разрядность цифро-аналогового преобразователя драйвера фазового модулятора, а по оси ординат число срабатываний ДОФ.

Определить значение разрядности цифроаналогового модулятора (определяющего величину напряжения на фазовом модуляторе Боба) *mod.DAC levels values* соответствующего сдвигу фазы на π .

Пример – На рисунке 9 представлен результат работы программы *rqcBob Mod Levell Test* значение *mod.DAC levels values* равно 150 относительных единиц.

4.3 Точная настройка временного окна детектора одиночных фотонов

На управляющем ПК Алисы в параметрах сетевого интерфейса определить значение IP-адреса, присвоенного ПК Алисы.

На управляющем ПК Боба в диалоговом окне проекта *Bob.lvproj* запустить файл *\Bob Host Library\Examples\Low Level\rqcBob Window Test (Low Level Example).vi*. В появившемся графическом интерфейсе программы (рисунок 10) задать следующие значения параметров:

- Alice's IP address* – IP-адрес управляющего ПК Алисы;
- *port* – 5020 (номер порта процесса-получателя (приёмника));
- *train.period* – 2000;
- *train.num of pulses* – 20;
- *train.mode* – train;
- *rain.num of trains* – 1000;
- *train.first pulse delay mod* – every train;
- *train.first pulse delay* – 55;
- *modulator.start* – 1800 (такт начала работы модулятора);
- *modulator.stop* – 1900 (такт окончания работы модулятора);
- *window.open* – значение полученное в п. 4.1; *mod.DAC value type* – constant;
- *mod.DAC levels values* – 150;
- *mod.rnd gen initial seed* – 00000000; *mod.clock delay* – 0.

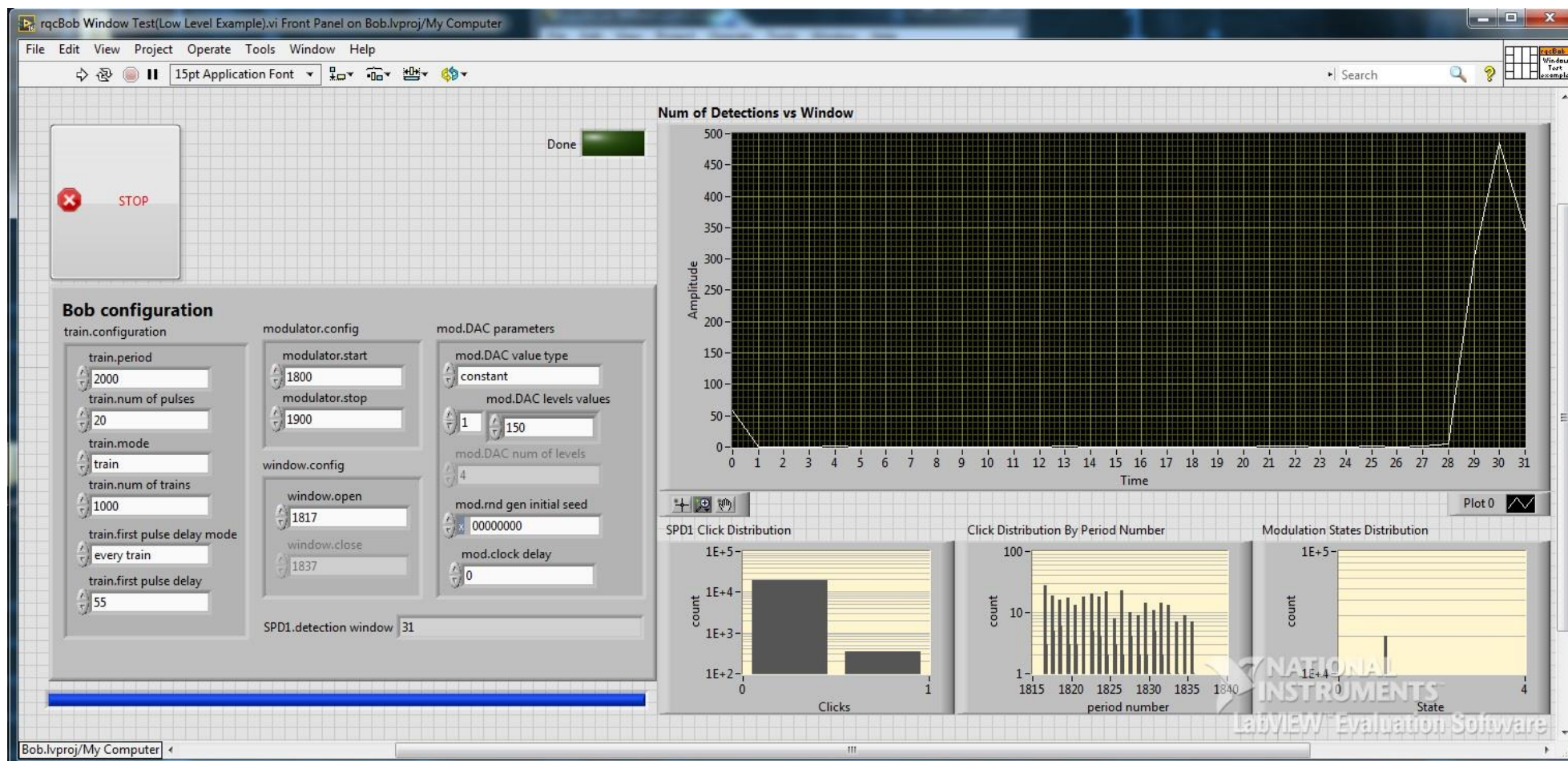


Рисунок 10 Результат работы программы rqcBob Window Test. Номер малого окна срабатывания ДОФ – 30

Запустить программу *rqcBob Window Test* на выполнение, в правой части диалогового окна будет построена диаграмма, на которой по оси абсцисс будут отражены номера малых окон срабатывания ДОФ (от 0 до 39), а по оси ординат число срабатываний ДОФ.

Определить номер малого окна срабатывания ДОФ по максимальному числу срабатываний.

Пример – На рисунке 10 представлен результат работы программы *rqcBob Window Test*, номер малого окна срабатывания ДОФ – 30.

4.4 Настройка величины задержки приложения напряжения на фазовый модулятор Алисы

На управляющем ПК Боба запустить файл *\Bob Host Library\Examples\Low Level\rqcBob Alice Delay Line Test (Low Level Example).vi*. В появившемся графическом интерфейсе программы (рисунок 11) задать следующие значения параметров:

- Alice's IP address* – IP-адрес управляющего ПК Алисы;
- *Port* – 5020 (номер порта процесса-получателя (приёмника));
- *train.period* – 2000;
- *train.num of pulses* – 3 (генерация трёх импульсов в трейне);
- *train.mode* – train;
- *train.num of trains* – 1000;
- *train.first pulse delay mod* – every train;
- *train.first pulse delay* – 55;
- *window.open* – значение полученное в п. 4.1;
- *modulator.start* – 1800 (такт начала работы модулятора);
- *modulator.stop* – 1850 (такт окончания работы модулятора);
- *pm.DAC value type* – constant;
- *pm.DAC levels values* – 0;
- *pm.DAC num of levels* – 0;
- *pm.rnd gen initial seed* – 00000000;

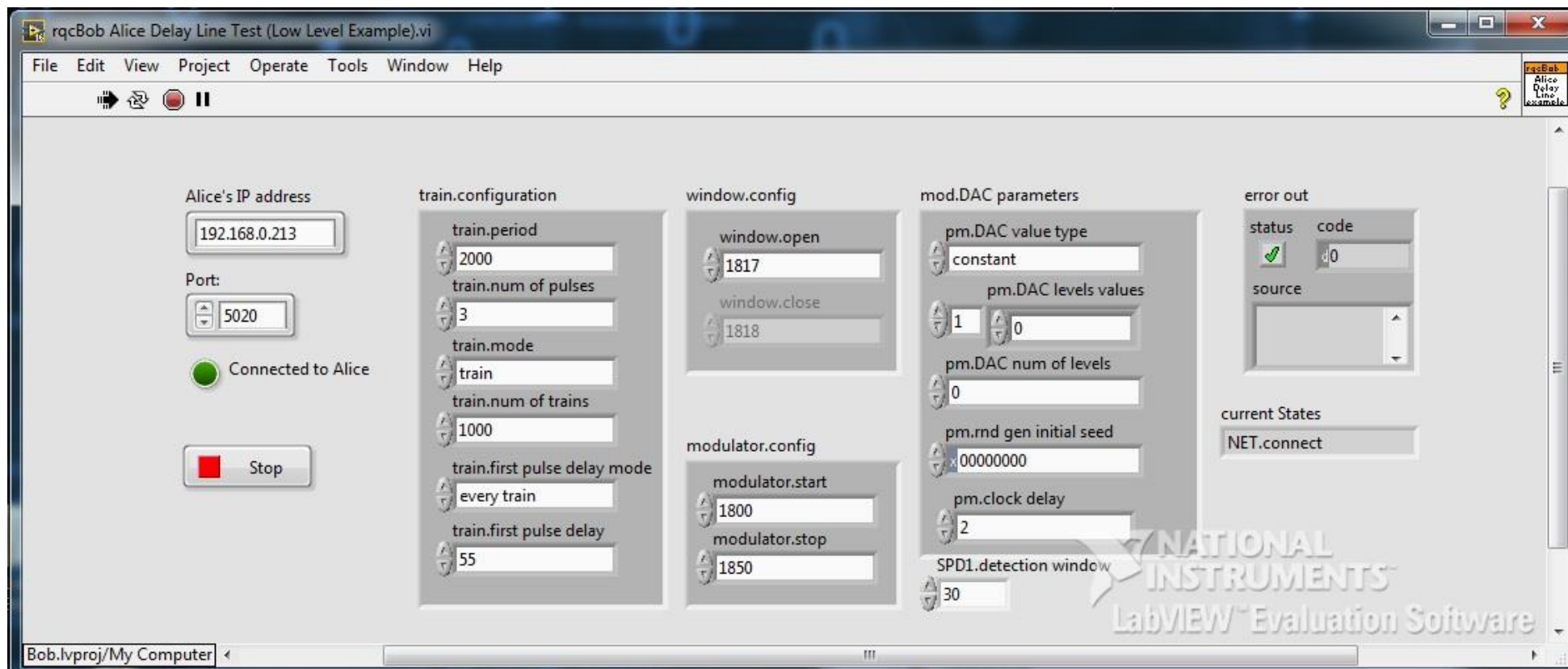


Рисунок 11 Интерфейс программы *rqcBob Alice Delay Line Test*

- *pm.clock delay* – 2;
- *SPD1.detection window* – 30
- *code* (группы *error out*) – 0.

Значение параметра *window.close* –будет автоматически рассчитано на основе значений переменных *window.open* и *train.num of pulses*.

На управляющем ПК Алисы запустить файл проекта *Alice.lvproj* из состава поставляемого программного обеспечения. В появившемся диалоговом окне (рисунок 12) запустить файл: *\Alice Host Library\Examples\Low Level\rqcAlice Delay Line Test (Low Level Example).vi*.

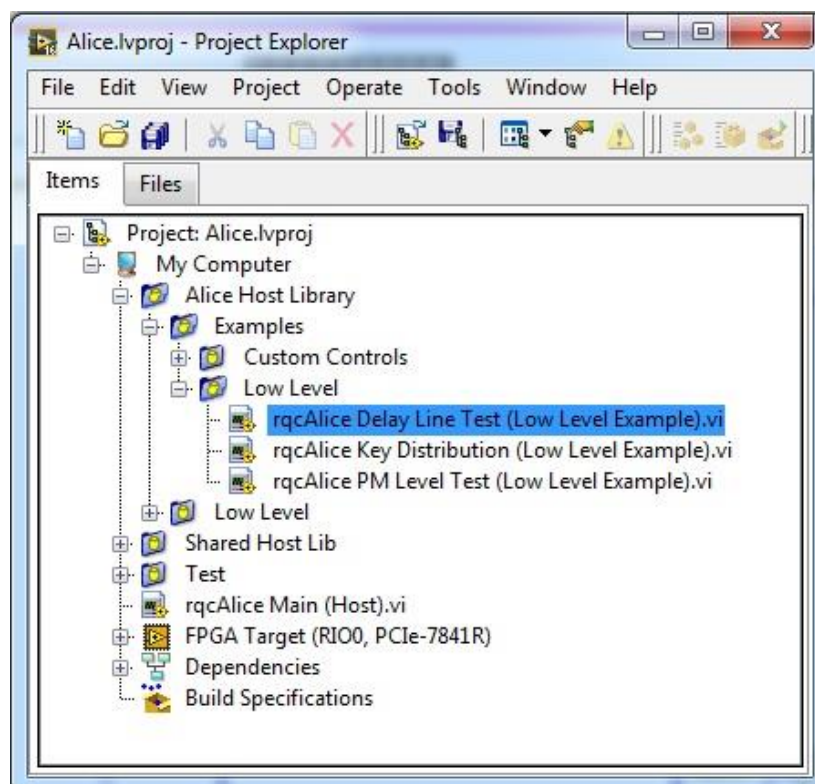


Рисунок 12 Запуск программы *rqcAlice Delay Line Test.vi*

В появившемся графическом интерфейсе программы (рисунок 13) задать следующие значения параметров:

- *port* – 5020 (номер порта процесса-получателя (приёмника));
- *detector pulse width* – 5;
- *DAC clock pulse width* – 5;

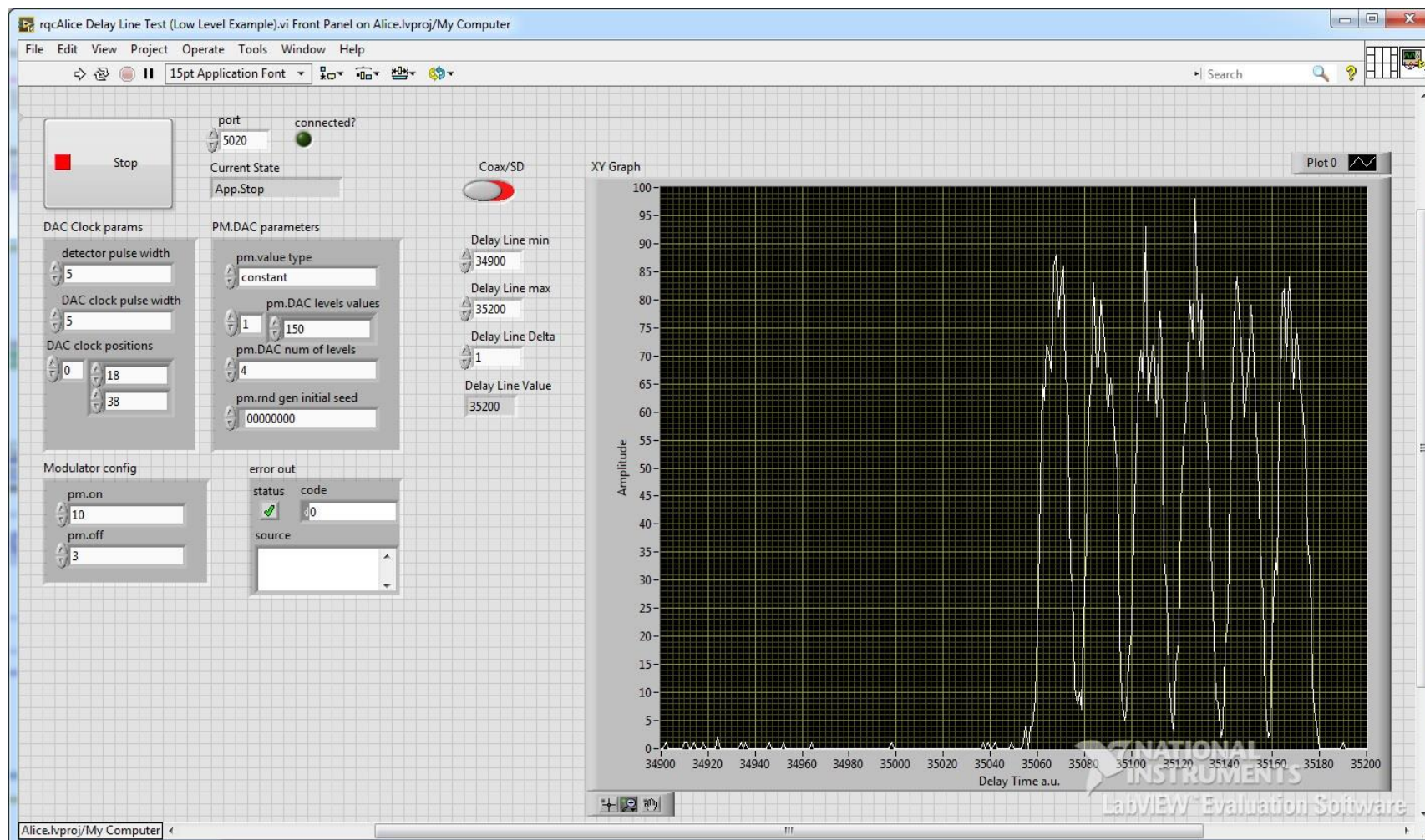


Рисунок 13 Результат работы программы *rqcAlice Delay Line Test*. Величина задержки приложения напряжения на фазовый модулятор «Алисы» – 35170

- *DAC clock positions* – от 18 до 38;
- *pm.value type* – constant;
- *pm.DAC levels values* – 150;
- *pm.DAC num of levels* – 4;
- *pm.rnd gen initial seed* – 00000000;
- *pm.on* – 10;
- *pm.off* – 3;
- *code* (группы *error out*) – 0;
- *Delay Line Delta* – 1.

Значение полей *Delay Line min* и *Delay Line max* определить исходя из расчётного значения длины квантового канала и линии задержки, уменьшив или увеличив его на несколько сот метров.

Пример – Для оптического канала и линии задержки длиной 10 и 25 км их совокупная длина составляет 35 км (35 000 м), соответственно значения полей *Delay Line min* и *Delay Line max* может составлять 34 900 и 35 200 м.

На управляющих ПК Боба и Алисы одновременно запустить на выполнение программы *rqcBob Alice Delay Line Test.vi* и *rqcAlice Delay Line Test.vi* соответственно.

В диалоговом окне программы *rqcAlice Delay Line Test.vi* будет построен график зависимости счёта ДОФ Боба от величины задержки приложения напряжения на фазовый модулятор Алисы. На графике число импульсов в трейне будет удвоено по причине прохождения разных плеч интерферометра Боба. Величину задержки определить как центр крайнего правого максимума.

Пример – На рисунке 13 представлен результат работы программы *rqcAlice Delay Line Test*, величина задержки – 35170.

4.5 Настройка напряжения на фазовом модуляторе Алисы

На управляющем ПК «Боба» в диалоговом окне проекта *Bob.lvproj* запустить файл *\Bob Host Library\Examples\Low Level\rqcBob Alice PM Level Test (Low Level Example).vi*. В появившемся графическом интерфейсе программы (рисунок 14) задать следующие значения параметров:

– *Alice's IP address* – IP-адрес управляющего ПК «Алисы» (определён ранее в п. 4.3);

- *port* – 5020 (номер порта процесса-получателя (приёмника));
- *train.period* – 2000;
- *train.num of pulses* – 20;
- *train.mode* – train;
- *train.num of trains* – 1000;
- *train.first pulse delay mod* – every train;
- *train.first pulse delay* – 55;
- *window.open* – значение полученное в п. 4.1;
- *pm.DAC num of levels* – 0;
- *pm.rnd gen initial seed* – 00000000;
- *pm.clock delay* – 1;
- *SPD1.detection window* – 30
- *code* (группы *error out*) – 0.

Значение параметра *window.close* –будет автоматически рассчитано на основе значений переменных *window.open* и *train.num of pulses*.

На управляющем ПК Алисы в диалоговом окне проекта *Alice.lvproj* запустить файл: *\Alice Host Library\Examples\Low Level\rqcAlice PM Level Test (Low Level Example).vi*.

В появившемся диалоговом окне программы (рисунок 15) задать следующие значения параметров:

- *port* – 5020 (номер порта процесса-получателя (приёмника));
- *detector pulse width* – 5;
- *DAC clock pulse width* – 5;
- *DAC clock positions* – от 18 до 38;
- *pm.rnd gen initial seed* – 00000000;
- *PM min* – 0;
- *PM max* – 512;
- *PM Delta* – 8;
- *PM Value* – 512;
- *Delay Line* – величина задержки, значение получено в п. 4.4;

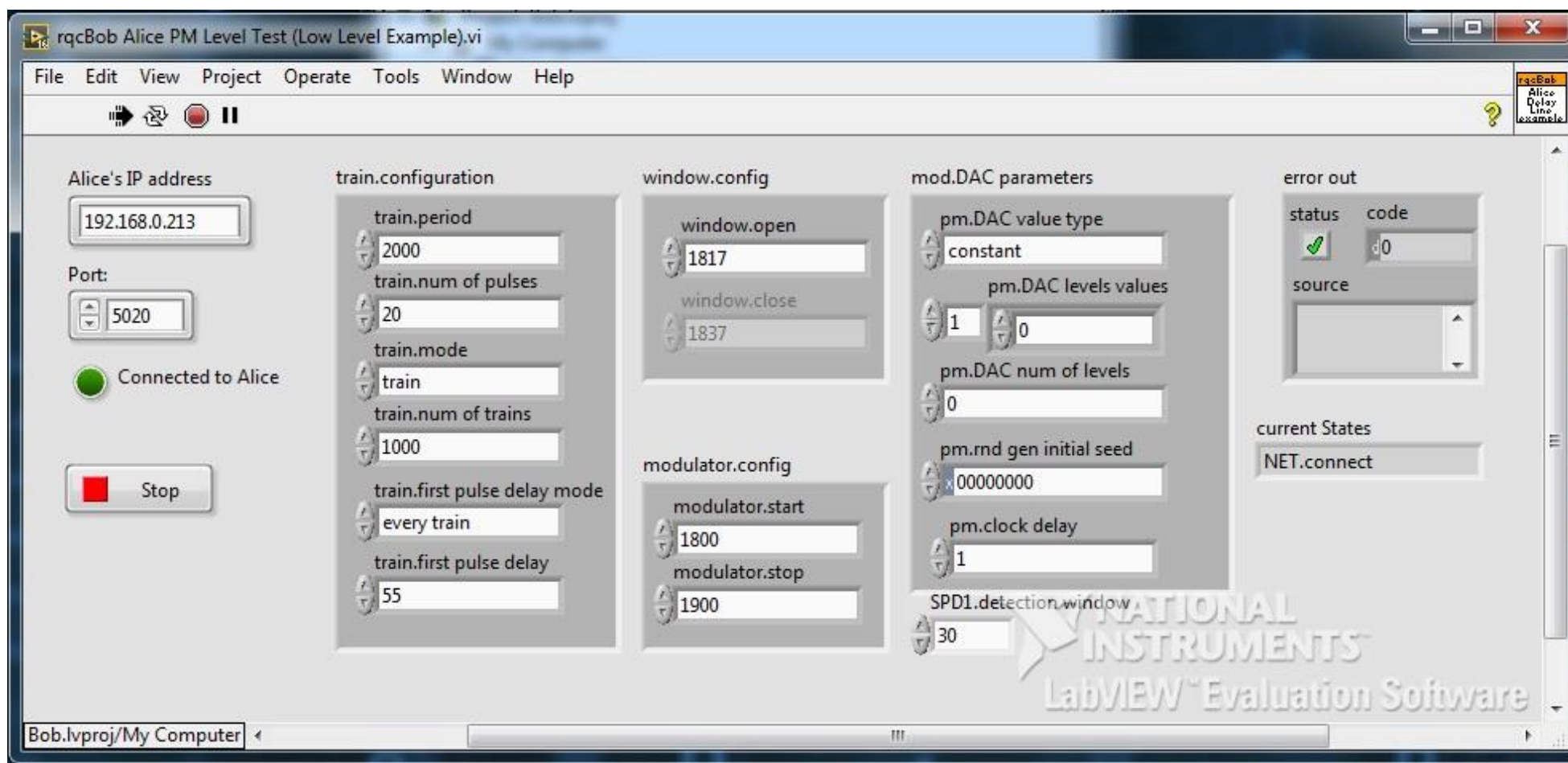


Рисунок 14 Интерфейс программы *rqcBob Alice PM Level Test*

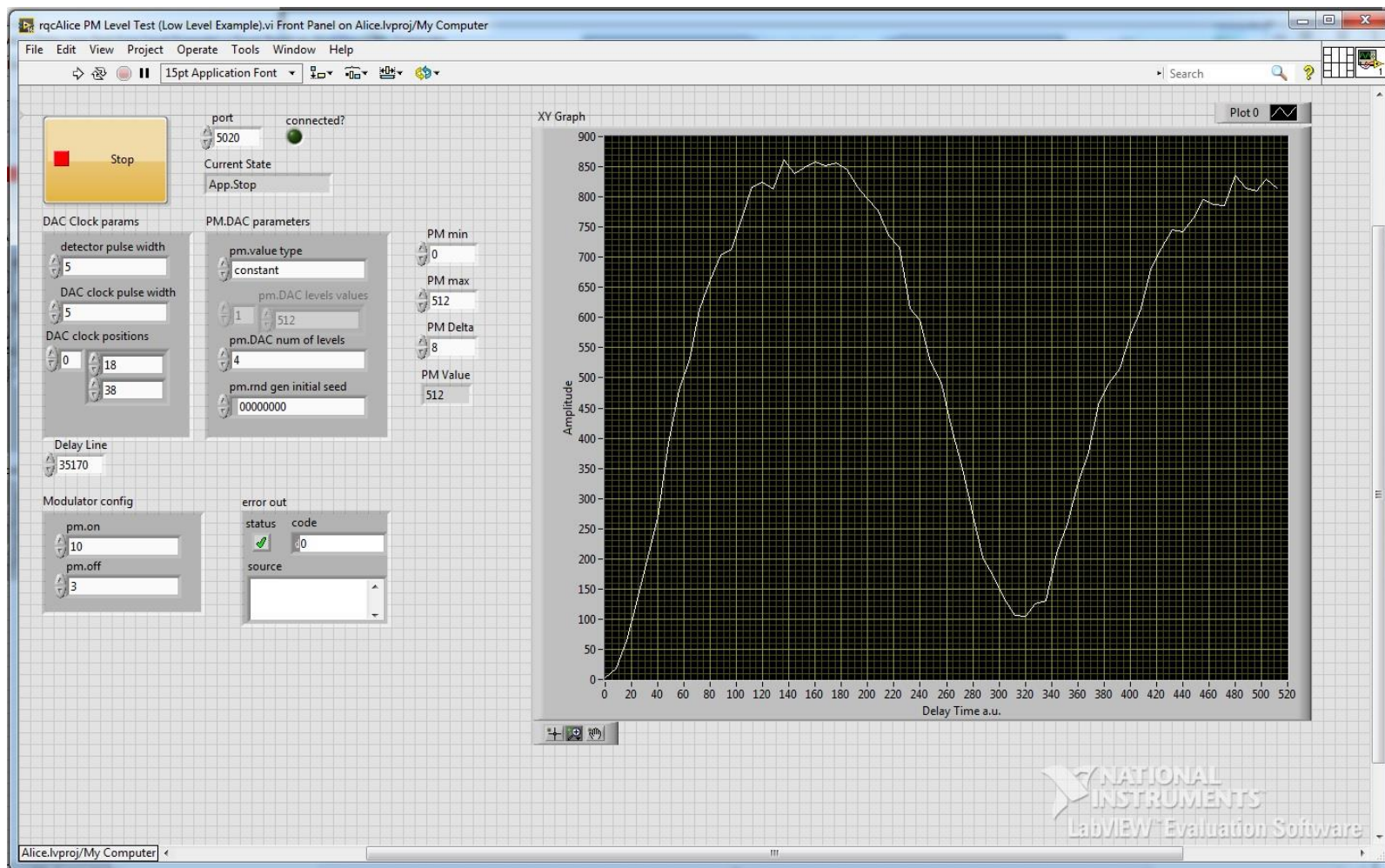


Рисунок 15 Результат работы программы *rqcAlice PM Level Test*. Значение *pm.DAC levels values* равно 160 относительных единиц

- *pm.on* – 10;
- *pm.off* – 3;
- *code* (группы *error out*) – 0.

На управляющих ПК Боба и Алисы одновременно запустить на выполнение программы *rqcBob Alice PM Level Test.vi* и *rqcAlice PM Level Test.vi* соответственно. В диалоговом окне программы *rqcAlice PM Level Test.vi* будет построен график, на котором по оси абсцисс в относительных единицах отражена разрядность цифроаналогового преобразователя драйвера фазового модулятора, а по оси ординат число срабатываний ДОФ.

Определить значение разрядности цифроаналогового модулятора (определяющего величину напряжения на фазовом модуляторе Алисы) *pm.DAC levels values* соответствующего сдвигу фазы на π .

П р и м е р – На рисунке 15 представлен результат работы программы *rqcBob PM Level Test*, значение *pm.DAC levels values* равно 160 относительных единиц.

4.6 Запуск генерации квантового ключа

Ввиду чувствительности оптических элементов схемы к изменению внешних условий, её использование осуществляется непосредственно после процедуры настройки и определения параметров функционирования (п.п. 4.1 – 4.5).

На управляющем ПК Боба в диалоговом окне проекта *Bob.lvproj* запустить файл *\Bob Host Library\Examples\Low Level\rqcBob Key Distribution (Low Level Example).vi*. В появившемся графическом интерфейсе программы (рисунок 16) задать следующие значения параметров:

- *Alice's IP address* – IP-адрес управляющего ПК Алисы (определён в п. 4.3);
- *port* – 5020 (номер порта процесса-получателя (приёмника));
- *pm.DAC value type* – random sequence;
- *pm.DAC levels values* – 75;
- *pm.DAC num of levels* – 4;
- *pm.rnd gen initial seed* – 00000000;
- *pm.clock delay* – 3;
- *window.open* – такт открытия большого окна ДОФ (определён в п. 4.1);

BOB

Current state:
Running FPGA... ■ Stop

LAN settings:
Alice's IP address: 192.168.0.33 Port: 5020 ● Connected to Alice

Key distillation details:
Bob's Filtered items: 3486
Bob's Right Items received: 1 ● Incorrect Alice's data
7841/7811
2-Random (Up)
4-Random (Down)

Data exchange details:
Total pulses expected: 999000 Real pulse count: 999000
Errors: No Error

FPGA settings:

Number of pulses
Train Period

Check D1: 24
Check D2: 0
Open Window: 1817

Train Number: 1000
Train Period: 4400

PM Pi Voltage: 190
WRT: 0
CLK: 5
DAC Level: 0
Start PM: 1530
Stop PM: 3600
PM-D shift: 3
Random On/Off: ●

Key length (bits): 0
Key distribution rate (kbps): 0
Bob's Key:

Key distribution rate (kbps) graph:
Y-axis: Distribution rate, kbps (-1 to 1)
X-axis: Time (0 to 9)
The graph shows a flat line at 0 kbps.

Рисунок 16 Интерфейс программы *rqcBob Key Distribution*

- *train.period* – период трейна (определён в п. 4.1);
- *train.num of pulses* – число импульсов в трейне (в диапазоне от 100 до максимально возможного, определённого в п. 4.1);
- *train.mode* – *train sequence.single*;
- *train.num of trains* – число генераций трейнов за сеанс (определяется пользователем);
- *train.first pulse delay mod* – *every train*;
- *train.first pulse delay* – 55;
- *Check D1* – номер малого окна ДОФ (определён в п. 4.3);
- *Pi Voltage* – значение разрядности цифроаналогового модулятора Боба соответствующего сдвигу фазы на π (определён в п. 4.2 *pm.DAC levels values*);
- *Data shift* – 0;
- переключатель *Random On/Off* – в положение *Off*.

На управляющем ПК Алисы в диалоговом окне проекта *Alice.lvproj* запустить файл: *\Alice Host Library\Examples\Low Level\rqcAlice Key Distribution (Low Level Example).vi*. В появившемся графическом интерфейсе программы (рисунок 17) задать следующие значения параметров:

- *port* – 5020 (номер порта процесса-получателя (приёмника));
- *detector pulse width* – 5;
- *DAC clock pulse width* – 5;
- *DAC clock positions* – от 18 до 38;
- *pm.value type* – *random value*;
- *pm.DAC levels values* – 75;
- *pm.rnd gen initial seed* – 00000000;
- *pm.on* – 9;
- *pm.off* – 2;
- *Pulses in Train* – аналогично значению *train.num of pulses* установленному в программе *rqcBob Key Distribution.vi* управляющего ПК Боба;
- *Train Number* – аналогично значению *train.num of trains* установленному в программе *rqcBob Key Distribution.vi* управляющего ПК Боба;
- *Delay Line* – величины задержки приложения напряжения на фазовый модулятор Алисы (определён в п. 4.4);

Current state:
Running FPGA... Stop

LAN settings:
IP address: 192.168.0.92 Port: 5020 Bob is connected

Key sifting details:
Bob's Filtered Items received: 1979 Right Bob's items sent: 952 Incorrect Bob's data

Data exchange details:
Total pulses expected: 999000 Real pulse count: 999000 Data loss detected

Errors:
No Error

FPGA settings:
Number of pulses in train: 999 Train Number: 1000
Train Period: T_{pulse} T_{off}
QC, BS, VOA, SD, SL, PM, FM
Pi Voltage: 110 PM Data on: 10 PM Data off: 3
DL SPI: 35175 PM DAC Level: 0
PM Random On/Off

QBER:
QBER, %: 2,86017

Key length:
Key length (bits): 944
Output file size, MB: 43,3688

Alice's Key:
1101000010110001100001111010011001100101010001100101111001011100000010010000101101111001010001110
11101101011100110111001010010100000100011011000101110011000111101111011110100000001001011000
0010101000111101100101110001000111001001011101100011100110011101111010000000100101100010
010001010000110110000011101001100111110100000010010100111110110101110011000101000100011011110
011100001111011001110101000111010110011110101000111010111010110011100110010000001010000
01110001001000110111011101010011101101011110010011111100001010010111001111101101010110
110111000100001110100000010001110100001011000111000100101011001000001111000001100100101000
110000000010011110001011101000101011110100101011100100101100101001100101010010011000000
011100101001001101011100110111010011101000111010001000001010001110011011110011110011110011001100011000
11001101000011000100010100100011100001110000100101010000010111011100100

Bob's Key:
1101000010110001100101111010011001100101010001100101110010011000000100110010101111001010001100
11101101011100110110010100101000001000110110001011101100011100100111011110100000001001011000
00101010001110110010101100010100111001001010110000010010001010110000010100100000101000010
01000101000010110000011101001100111110100000010010100111110110101110011000101000100011011110
0111000011110110011101101000111010100111101010001110101001110101011001110011010000001010000
0110000100100011011011101010010111010111100100111110000101100110011111010001010110
11011100010000111010000000100011101000010100011110001001010010100000011111000011001001011000
11000000001001111000110111010001010111101001010111010010011011100101001100101010010011000000
0111001010010011010111001100110111010011001000100000101000111001101111001111001111001100011000
11001101000011000100010100100011100001110000100101010000010111011100100

Рисунок 17 Интерфейс программы rqcAlice Key Distribution

– *Pi Voltage* – значение разрядности цифроаналогового модулятора Алисы соответствующего сдвигу фазы на π (определён в п. 4.2 *pt.DAC levels values*);

– переключатель *Random On/Off* – в положение *Off*.

На управляющих ПК Боба и Алисы одновременно запустить на выполнение программы *rqcBob Key Distribution.vi* и *rqcAlice Key Distribution.vi* соответственно, в диалоговых окнах программ будет отображаться процесс генерации последовательности данных, и её длина.

Для получения последовательности данных максимальной длины настроить номер малого окна ДОФ в программе *rqcBob Key Distribution.vi*, увеличив или уменьшив значение поля *Check D1*.

В случае отсутствия генерации последовательности данных повторить процедуру настройки и определения параметров функционирования оптической схемы (п.п. 4.1 – 4.5).

5 Контрольные вопросы и задания

1. Пусть Ева перехватывает фотоны Алисы и измеряет их в каноническом или диагональном базисе, выбирая случайно. Потом она приготавливает фотон в состоянии, полученном при измерении, и отправляет его Бобу. Какую ошибку обнаружат Алиса и Боб, т.е. какая часть бит их секретных ключах в среднем получится разная?

2. Большое количество фотонов, отправленных Алисой, не доходят до Боба. Но Алиса и Боб не знают потерялись ли фотоны из-за рассеяния в линии или были украдены Евой. Влияет ли это обстоятельство на секретность квантового распределения ключа?

3. Длина накопительной линии в схеме Plug&Play составляет 25 км. Лазерные импульсы следуют с частотой 5МГц. Найдите максимальное значение лазерных импульсов в трейне.

4. Найдите период трейна для квантового канала и накопительной линии по 25 км. Частота следования лазерных импульсов – 5МГц. Количество импульсов в трейне возьмите из предыдущей задачи.

5. Количество фотонов в лазерном импульсе зависит от (выберите правильный ответ):

- а) энергии импульса;
- б) энергии и ширины импульса;
- в) энергии, ширины и формы импульса;
- г) энергии, ширины, формы и поляризации импульса;

6. Лазер генерирует импульсы с длиной волны 1,55 мкм и с частотой следования 5 МГц. Импульсы содержат по 0,1 фотону в среднем. Найдите мощность излучаемого света.

7. Алиса посылает фотоны Бобу, по оптоволоконному каналу длиной 100 км. Потери в оптоволокне – 0,3 дБ/км. Какая часть фотонов, посланных Алисой, достигнет Боба?

8. Чтобы шифровать на лету голос человека методом одноразовых блокнотов нужен ключ, генерируемый со скоростью 5 кбит/с. Пусть лазерные импульсы, следующие с частотой 1 ГГц, содержат 0,1 фотон на импульс; потери в канале 0,3 дБ/км; эффективность детекторов – 10%. Найдите максимальное

расстояние квантового распределения ключа по протоколу BB84 для шифрования голоса. Пренебрегите темновым счетом детекторов и возможными атаками Евы с разделением числа фотонов.

Литература

1. Gisin N. et al. Quantum cryptography // Rev. Mod. Phys. 2002. Vol. 74. P. 145–195.
2. The Physics of Quantum Information: Quantum Cryptography, Quantum Teleportation, Quantum Computation / ed. Bouwmeester D., Ekert A.K., Zeilinger A. Springer, 2000. 315 p.
3. Bennett C.H., Brassard G. Quantum cryptography: Public key distribution and coin tossing // IEEE Int. Conf. on Comput. Sys. and Sign. Proces., Bangalore, India, December 1984. 1984. P. 175–179.

