



Установка, настройка и администрирование.

# инструкция по эксплуатации

версия 1.0 Beta

Июнь 2020 года.

MiniQRNG является зарегистрированной торговой маркой OpenQbit Inc., на условиях свободной лицензии и коммерческого использования. Условия использования:

[www.OpenQbit.com](http://www.OpenQbit.com)

Содержание

1.	Введение.....	3
2.	Что такое блочное программирование?.....	4
3.	Что такое Термукс?.....	4
4.	Что такое Мини QRNG? .....	4
5.	Конфигурация хранения в Termux.....	9
6.	Установка сервера SSH (Secure Shell).....	10
7.	Настройка сервера SSH на мобильном телефоне (смартфоне).....	11
8.	Ambientes Blockly (App Inventor, AppyBuilder y Thunkable). .....	18
9.	Определение и использование блоков в Mini QRNG.....	19
10.	Создание "Аппаратного" устройства QRNG. ....	31
11.	Приложение "OpenQbit Quantum Computing".....	37
12.	Лицензирование и использование программного обеспечения.....	42

## 1. Введение.

Сегодняшняя криптография основана на последовательностях случайных чисел. Используемые в настоящее время генераторы псевдослучайных чисел, похоже, обеспечивают случайные последовательности битов, но в реальности эти последовательности битов имеют определенные закономерности, поэтому существует риск взлома или иных манипуляций с целью заставить информацию перемещаться по публичным и пиратским сетям.

Интеграция источников физической энтропии в генераторы случайных чисел является наиболее распространенным методом преодоления этой угрозы безопасности. Однако классическая физика является причинно-следственной, поэтому непредсказуемость последовательности битов, сгенерированной с помощью классической физики, не может быть доказана.

Квантовая физика, с другой стороны, по сути, случайна. Числа, генерируемые генератором квантовых случайных чисел (QRNG), невозможно предсказать: QRNG является явно непредсказуемым. Таким образом, если в системе безопасности используется генератор квантовых случайных чисел, то даже быстрый суперкомпьютер с быстрой арифметической операцией не может предсказать случайные последовательности битов, используемые этой системой безопасности.

Квантовая физика использует методы, основанные на фундаментальной концепции, называемой энтропией.

### ВИДЫ ЭНТРОПИИ

Существует два общих типа источников энтропии, которые могут быть измерены для генерации истинных случайных чисел. Первый тип включает в себя физический процесс, который трудно или невозможно измерить, или слишком интенсивный с точки зрения вычислений для прогнозирования, или и то, и другое. Это источник каотической энтропии. Распространенным примером, известным большинству людей, является лотерейный автомат. Набор последовательно пронумерованных шариков помещается в камеру и постоянно смешивается, вращая камеру или пропуская через нее воздух. Несколько шаров разрешается выпадать из камеры, а цифры, отмеченные на шарах, представляют собой розыгрыш лотереи. Ничья случайна из-за большого количества взаимодействий между шариками и камерой, что приводит к быстрому увеличению количества возможных движений каждого шарика. Мало того, что сложность этих взаимодействий чрезвычайно высока, нет очевидного способа точно наблюдать или измерять все внутренние переменные шариков, камеры и воздушного потока. Второй, очень разный тип источника энтропии - квантовая механика. Многие микроскопические частицы или волны, такие как фотоны, электроны и протоны,

обладают квантово-механическими свойствами, которые включают в себя вращение, поляризацию, положение и импульс. При правильной конфигурации для получения этих частиц, специфические значения их вращения или поляризации, например, не только неизвестны и теоретически непредсказуемы, но и физически определяются до тех пор, пока не будет произведено измерение.

## 2. Что такое блочное программирование?

**Blockly** - это **визуальный язык программирования**, состоящий из простого набора команд, которые мы можем комбинировать, как кусочки головоломки. Это очень полезный инструмент для тех, кто хочет **научиться программировать** интуитивно понятным и простым способом, или для тех, кто уже умеет программировать и хочет увидеть потенциал этого вида программирования.

Блоки - это форма программирования, где не требуется никакого фона ни на каком компьютерном языке, это потому, что это просто соединение графических блоков, как если бы мы играли в лего или головоломку, вам просто нужна какая-то логика и все!

Любой человек может создавать программы для мобильных телефонов (смартфонов), не связываясь с теми языками программирования, которые трудно понять, просто сложите блоки в графическом виде простым, легким и быстрым способом.

## 3. Что такое Термукс?

Termux - это эмулятор терминалов Android и приложение среды Linux, которое работает напрямую, без необходимости маршрутизации или настройки. Минимальная базовая система устанавливается автоматически.

Мы будем использовать Termux для его стабильности и простоты установки и управления, однако, вы можете использовать установленную среду Ubuntu Linux для Android.

В этой среде Linux у вас будет "ядро" коммуникационных процессов MiniQRNG.

## 4. Что такое Мини QRNG?

Mini QRNG - это программное и аппаратное обеспечение, которое включает в себя три технологических решения для создания QRNG (Quantum Random Number Generators - Квантовые генераторы случайных чисел). Классифицируется следующим образом:

а.- QRNG API. - Генератор квантовых случайных чисел, полученных с внешних серверов.

b.- MiniQRNG / Программное обеспечение. - Генератор случайных квантовых чисел, полученный с использованием физических свойств (квант) камеры мобильного телефона.

с.- MiniQRNG / Оборудование. - Генератор квантовых чисел, полученный с помощью аппаратных средств, основанных на свойствах квантовой физики лазера. Позже мы расскажем вам, как построить его дома по низкой цене.

#### 1. Установка и настройка терминала Termux Terminal.

Для начала нам нужна среда Linux, так как каждая система Android основана на Linux для безопасности и гибкости в инструментах, мы будем использовать терминал "Termux", который содержит ту среду, в которой мы установим инструмент(ы), которые помогут нам создавать QRNG.

Termux - это эмулятор Linux, на котором мы установим необходимые пакеты для создания квантовых чисел.

Одним из главных преимуществ использования терминала Termux является то, что вы можете устанавливать программы без необходимости "поворачивать" мобильный телефон (смартфон), что гарантирует, что гарантия производителя не будет потеряна в связи с данной установкой.

Установка терминала.

Со своего мобильного телефона перейдите в приложение с иконкой Google Play (play.google.com).



Выполните поиск по приложению "Termux", выберите его и запустите процесс установки.



## Запуск приложения Termux.

После запуска нам придется выполнить следующие две команды для выполнения обновления эмулятора операционной системы Linux:

[OpenQbit.com](http://OpenQbit.com)

\$ соответствующее обновление

\$ подходящий апгрейд

Подтвердите все опции Y(Yes)....

Termux

Home \$ apt обновление

\$ apt апгрейд

```
Welcome to Termux!

Wiki:           https://wiki.termux.com
Community forum: https://termux.com/community
Gitter chat:    https://gitter.im/termux/termux
IRC channel:    #termux on freenode

Working with packages:

* Search packages:  pkg search <query>
* Install a package: pkg install <package>
* Upgrade packages:  pkg upgrade

Subscribing to additional repositories:

* Root:   pkg install root-repo
* Unstable: pkg install unstable-repo
* X11:    pkg install x11-repo

Report issues at https://termux.com/issues

$
```

```
Welcome to Termux!

Wiki:           https://wiki.termux.com
Community forum: https://termux.com/community
Gitter chat:    https://gitter.im/termux/termux
IRC channel:    #termux on freenode

Working with packages:

* Search packages:  pkg search <query>
* Install a package: pkg install <package>
* Upgrade packages:  pkg upgrade

Subscribing to additional repositories:

* Root:   pkg install root-repo
* Unstable: pkg install unstable-repo
* X11:    pkg install x11-repo

Report issues at https://termux.com/issues

$ apt update
```

```
Welcome to Termux!

Wiki:           https://wiki.termux.com
Community forum: https://termux.com/community
Gitter chat:    https://gitter.im/termux/termux
IRC channel:    #termux on freenode

Working with packages:

* Search packages:  pkg search <query>
* Install a package: pkg install <package>
* Upgrade packages:  pkg upgrade

Subscribing to additional repositories:

* Root:   pkg install root-repo
* Unstable: pkg install unstable-repo
* X11:    pkg install x11-repo

Report issues at https://termux.com/issues

$ apt upgrade
```



## 5. Конфигурация хранения в Termux.

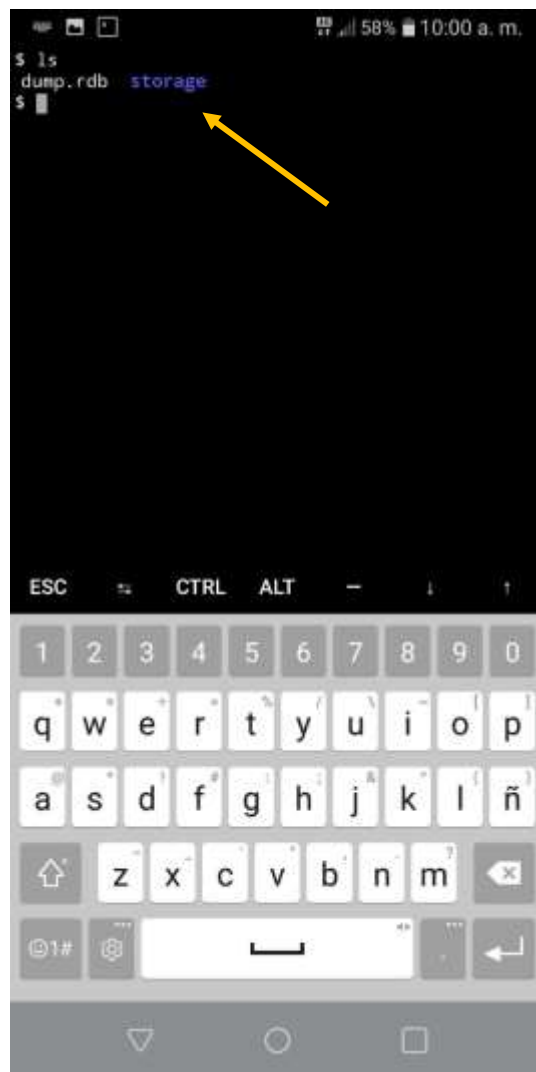
После того, как вы обновили и модернизировали систему Termux, мы начнем настройку просмотра внутреннего хранилища телефона в системе Termux. Это поможет вам иметь возможность обмениваться информацией между Termux и нашей информацией в телефоне.

Это можно сделать просто и быстро, выполнив следующую команду на терминале Termux.

### \$ срочное хранение

При выполнении предыдущей команды появляется окно с просьбой подтвердить создание виртуального **хранилища** (каталога) в Termux. Мы проверяем, отдавая команду:

\$ ls



## 6. Установка сервера SSH (Secure Shell).

\$ подходящая установка openssh

\$ подходящий монтажный путепровод

\$ apt install openssh

\$

apt install sshpass



```
$ apt install openssh
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
krb5 libdns libldb libedit termux-auth
The following NEW packages will be installed:
krb5 libdns libldb libedit openssh termux-auth
0 upgraded, 6 newly installed, 0 to remove and 0
not upgraded.
Need to get 2255 kB of archives.
After this operation, 11.9 MB of additional disk
space will be used.
Do you want to continue? [Y/n] Y
Get:1 https://dl.bintray.com/termux/termux-packa
ges-24 stable/main arm libldb arm 18.1.32-4 [465
kB]
Get:2 https://dl.bintray.com/termux/termux-packa
ges-24 stable/main arm krb5 arm 1.18.1 [839 kB]
24% [2 krb5 131 kB/839 kB 16%]
```



```
$ apt install sshpass
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
sshpass
0 upgraded, 1 newly installed, 0 to remove and 0
not upgraded.
Need to get 7158 B of archives.
After this operation, 57.3 kB of additional disk
space will be used.
0% [Working]
```

Мы завершили установку коммуникационной сети для локального SSH-сервера на мобильном смартфоне.

## 7. Настройка сервера SSH на мобильном телефоне (смартфоне).

Мы дадим возможность серверу SSH в мобильном телефоне подключаться с нашего ПК к мобильному телефону и работать быстрее и комфортнее, а также это послужит нам проверкой того, что услуга сервера SSH в мобильном телефоне работает корректно, так как мы будем использовать его в общении с Mini QRNG.

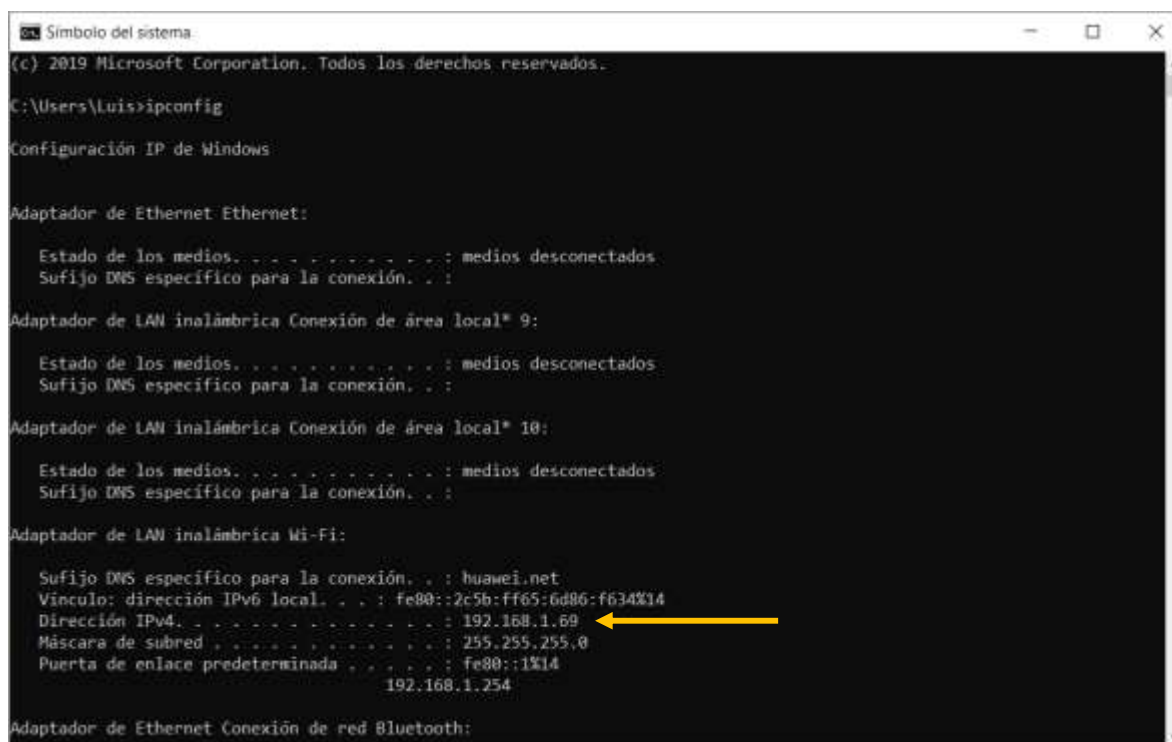
Первое, что нам нужно сделать, это подключить мобильный и компьютер к **одной и той же WiFi сети**, чтобы они могли видеть друг друга. IP-адреса или адреса должны быть похожи на 192.168.XXX.XXX. Значения XXX - это переменные числа, которые присваиваются случайным образом в каждом компьютере.

Этот пример был протестирован на мобильном телефоне LG Q6 и ПК с Windows 10 Home.

Проверьте IP или адрес, который ПК подключил к WiFi, мы должны открыть терминал в Windows.

На нижней панели, где находится поисковая лупа, напишите cmd и нажмите клавишу Enter. Откроется терминал и в нем мы напишем команду:

C:\Имя пользователя> ipconfig



```
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Luis>ipconfig

Configuración IP de Windows

Adaptador de Ethernet Ethernet:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :

Adaptador de LAN inalámbrica Conexión de área local* 9:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :

Adaptador de LAN inalámbrica Conexión de área local* 10:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :

Adaptador de LAN inalámbrica Wi-Fi:

    Sufijo DNS específico para la conexión. . : huawei.net
    Vínculo: dirección IPv6 local. . . : fe80::2c5b:ff65:6d86:f634%14
    Dirección IPv4. . . . . : 192.168.1.69
    Máscara de subred. . . . . : 255.255.255.0
    Puerta de enlace predeterminada. . . : fe80::1%14
                                      192.168.1.254

Adaptador de Ethernet Conexión de red Bluetooth:
```

Он покажет нам IP-адрес, присвоенный ПК в этом файле - 192.168.1.69, но, скорее всего, он будет разным в каждом случае.

ЗАМЕЧАНИЕ: Адрес, на котором написано "IPv4 адрес" должен быть взят, чтобы его не путать со шлюзом.

Теперь в случае с мобильным телефоном в терминале Termux мы должны ввести следующую команду, чтобы узнать имя нашего пользователя, которое мы будем использовать для подключения к SSH-серверу, у которого есть наш телефон, мы выполняем следующую команду:

**\$ whoami**

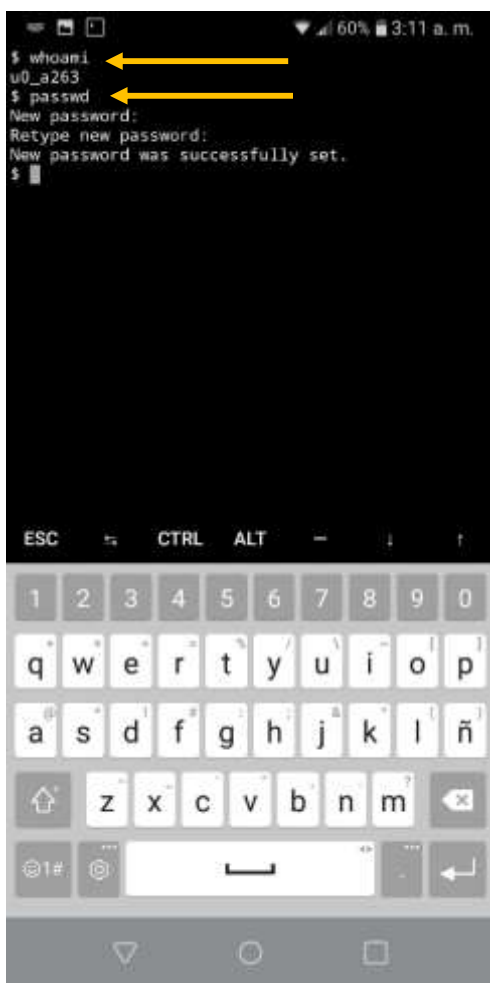
Позже мы должны дать пароль этому пользователю, поэтому мы должны выполнить следующую команду:

**\$ пас**

Он попросит нас ввести пароль и нажмет Enter, снова попросит нас ввести пароль, который мы подтверждаем и нажмет Enter, если он был **успешно "Новый пароль был успешно установлен"**, в случае пометки ошибки возможно, что пароль был набран неправильно. Выполните процедуру еще раз.

А затем, чтобы узнать, какой IP у нас есть в Termux, мы вводим следующую команду, IP после слова "inet":

**\$ ifconfig -a**



Теперь пришло время запустить службу SSH-сервера на вашем телефоне, чтобы вы могли получать сеансы с вашего ПК. Мы выполняем следующую команду в терминале Термих, эта команда не дает никакого результата.


шшд



Теперь нам придется установить на ПК программу, которая будет взаимодействовать с SSH-сервером телефона с ПК.

Надо зайти на <https://www.putty.org>.

Выберите, где находится ссылка "Вы можете скачать PuTTY здесь".




The screenshot shows a web browser window with the address bar displaying <https://www.putty.org>. The main content area features a small image of the PuTTY application window on the left. To the right of the image, the heading "Download PuTTY" is followed by a paragraph: "PuTTY is an SSH and telnet client, developed originally by Simon Tatham for the with source code and is developed and supported by a group of volunteers." Below this, the text "You can download PuTTY [here](#)." is displayed, with a yellow arrow pointing to the word "here".

---

Below suggestions are independent of the authors of PuTTY. They are *not* to be seen a

---



The screenshot shows the Bitvise SSH Client website. It includes a screenshot of the Bitvise SSH Client application window on the left. The main text area has the heading "Bitvise SSH Client" followed by a paragraph: "Bitvise SSH Client is an SSH and SFTP client for Windows. It is developed and supported prof supports all features supported by PuTTY, as well as the following:"

- graphical SFTP file transfer;
- single-click Remote Desktop tunneling;
- auto-reconnecting capability;
- dynamic port forwarding through an integrated proxy;
- an FTP-to-SFTP protocol bridge.

Bitvise SSH Client is **free to use**. You can [download it here](#).

Выбирайте 32-битную версию, не имеет значения, будет ли ваша система 64-битной или нет.

## Download PuTTY: latest release

[Home](#) | [FAQ](#) | [Feedback](#) | [Licence](#) | [Updates](#) | [Mirrors](#)  
Download: [Stable](#) | [Snapshot](#) | [Docs](#) | [Contact](#)

This page contains download links for the latest released version of PuTTY. Currently this is 0.73, released on 2019-09-29.

When new releases come out, this page will update to contain the latest, so this is a good page to bookmark or link to. Alternative

Release versions of PuTTY are versions we think are reasonably likely to work well. However, they are often not the most up-to-date, to see if the problem has already been fixed in those versions.

**Package files**

You probably want one of these. They include versions of all the PuTTY utilities.

(Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

**MSI ('Windows Installer')**

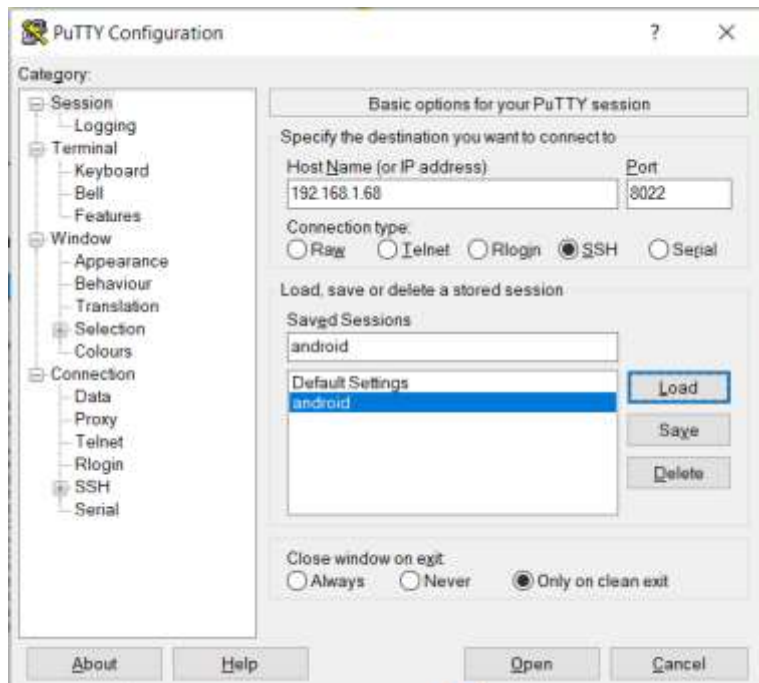
32-bit:	<a href="#">putty-0.73-installer.msi</a>	(or by FTP)	(signature)
64-bit:	<a href="#">putty-64bit-0.73-installer.msi</a>	(or by FTP)	(signature)

**Unix source archive**

.tar.gz:	<a href="#">putty-0.73.tar.gz</a>	(or by FTP)	(signature)
----------	-----------------------------------	-------------	-------------

После загрузки на компьютер запустите его и установите с параметрами по умолчанию. Затем запустите приложение PuTTY.

В этой сессии мы введем данные с нашего Openssh сервера, который мы установили в мобильный телефон.



Введите IP-адрес мобильного телефона.

Имя хоста или IP-адрес:

192.168.1.68 (пример IP)

Порт:

8022 (порт по умолчанию мобильного SSH-сервера).

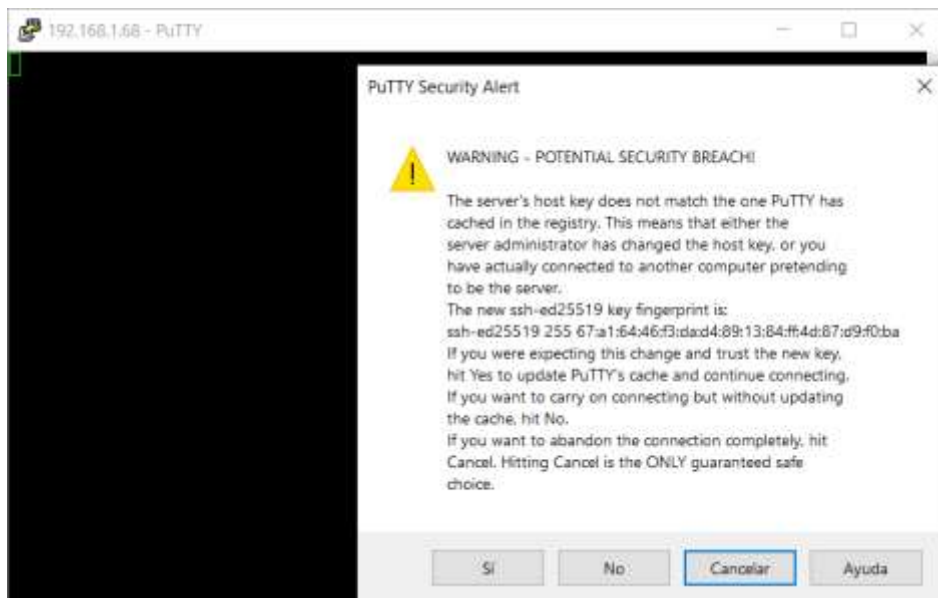
Мы можем дать название сессии в разделе "Сохраненные сессии" и

нажать кнопку "Сохранить".



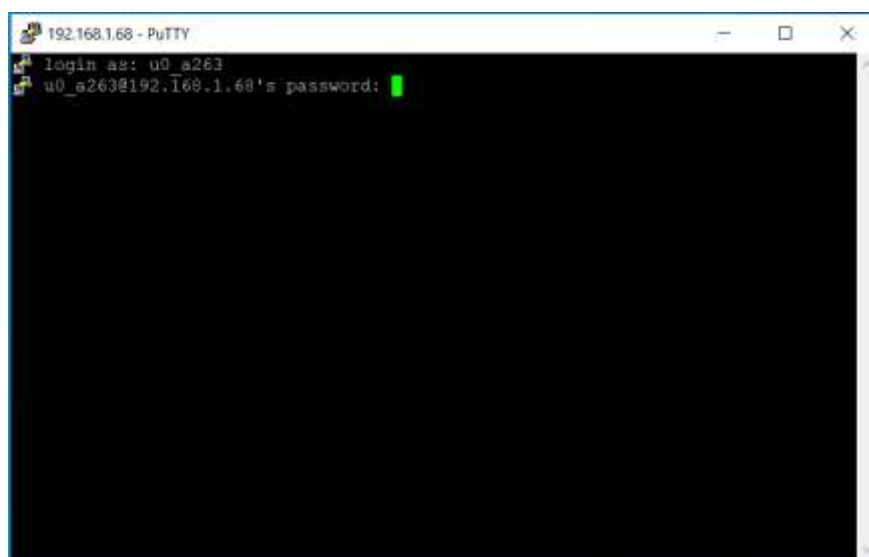
Позже в нижней части мы нажимаем, чтобы открыть соединение с сервером, давая кнопку "Открыть".

При первом подключении к компьютеру вас **попросят подтвердить** ключ шифрования информации, нажав кнопку "Да".



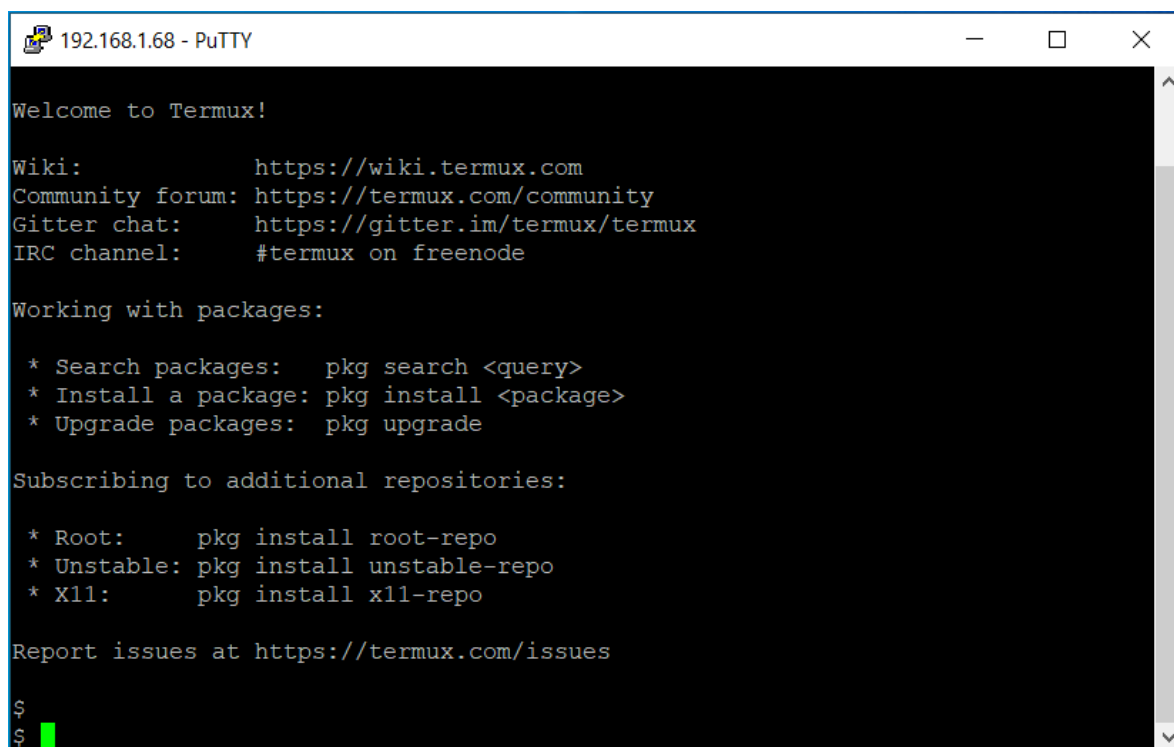
Позже нас спросят о пользователе, с которым мы собираемся связаться. Мы будем использовать ранее полученную информацию (пользователь и пароль).

**Вход как:** мы должны ввести нашего пользователя и дать Enter, затем мы попросим пароль еще раз дать кнопку Enter.





Если данные были верны, то мы находимся в сеансе SSH (Secure Shell), выполненном с компьютера (клиента) на телефоне (SSH Server).



```
192.168.1.68 - PuTTY
Welcome to Termux!

Wiki:          https://wiki.termux.com
Community forum: https://termux.com/community
Gitter chat:   https://gitter.im/termux/termux
IRC channel:   #termux on freenode

Working with packages:

* Search packages:  pkg search <query>
* Install a package: pkg install <package>
* Upgrade packages: pkg upgrade

Subscribing to additional repositories:

* Root:    pkg install root-repo
* Unstable: pkg install unstable-repo
* X11:     pkg install x11-repo

Report issues at https://termux.com/issues

$
$
```

**ВАЖНОЕ ЗАМЕЧАНИЕ:** Помните, что IP (адрес) компьютера и IP (адрес) мобильного телефона, подключенного к одному и тому же WiFi, вероятно, будет меняться каждый раз, когда мы отсоединяемся и переподключаемся, поэтому мы должны дважды проверить, какие адреса есть у каждого устройства, это обеспечит успех соединения между устройствами через SSH-сервер телефона и ПК (Клиент).

## 8. Ambientes Blockly (App Inventor, AppyBuilder y Thunkable).

App Inventor - это среда разработки программного обеспечения, созданная Google Labs для создания приложений для операционной системы Android. Пользователь может, визуально и из набора основных инструментов, связать ряд блоков для создания приложения. Система бесплатна и может быть легко загружена из Интернета. Приложения, созданные с помощью App Inventor, очень легко создавать, так как не требуется знание какого-либо языка программирования.

Все текущие среды, использующие технологию Blockly, такие как AppyBuilder и Thunkable, среди прочих, имеют свою бесплатную версию, их способ использования может быть через интернет на различных сайтах, или они также могут быть установлены дома.

Блоки, входящие в архитектуру Mini BloclyChain, были протестированы в App Inventor и AppyBuilder, но из-за оптимизации кода они должны работать на других платформах.

Онлайн-версии:

App Inventor.

<https://appinventor.mit.edu/>

AppyBuilder.

<http://appybuilder.com/>

Спасибо.

<https://thunkable.com/>

Версия, которая будет установлена на вашем компьютере (ПК):

<https://sites.google.com/site/aprendeappinventor/instala-app-inventor>

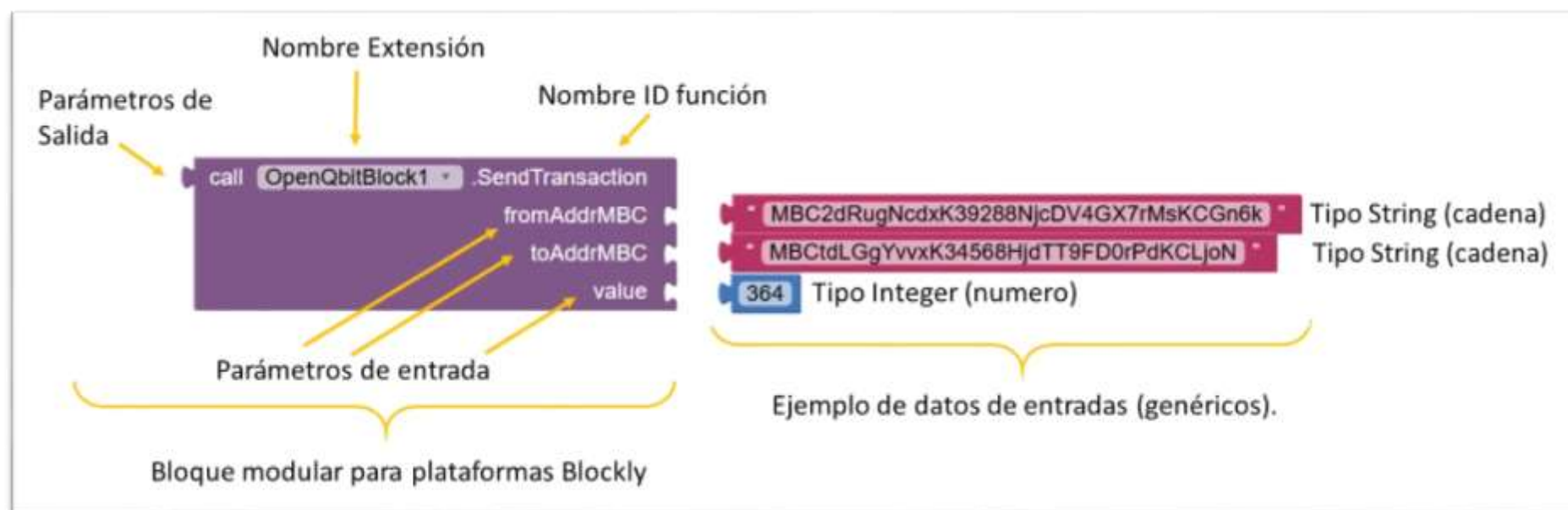
Среда для разработчиков блочных блоков.

<https://editor.appybuilder.com/login.php>

## 9. Определение и использование блоков в Mini QRNG.

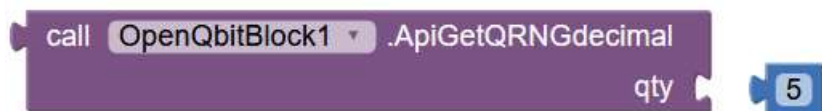
Мы начнем с объяснения распределения данных, которые будут иметь все блоки, их синтаксис использования и настройки.

В следующем примере мы видим модульный блок и его входные и выходные параметры, а также типы входных данных, эти данные могут быть типа String (строка символов) или Integer (целое или десятичное число). Мы показываем, как он используется и настраиваем его для правильного функционирования.



Каждый блок модуля будет иметь свое описание и будет назван в случае, если он имеет обязательные или необязательные зависимости от других блоков, используемых в качестве входных параметров, будет объявлен процесс интеграции. Начнем с блоков расширения **OpenQbitQRNGwithSSH**.

Блок для генерации десятичных случайных квантовых чисел - (**ApiGetQRNGdecimal**)



Входные параметры: **qty** <Integer>

Выходные параметры: Дает количество "qty" случайных квантовых десятичных чисел, введенных во входные числа в диапазоне 0 и 1 в формате JSON.

Пример:

qty = 5; вывод: {"результат": [0.5843012986202495, 0.7746497687824652, 0.05951126805960929, 0.1986079055812694, 0.03689783439899279]} }

Описание: API генератора квантовых случайных чисел (QRNG)

Блок для генерации десятичных случайных квантовых чисел - (**ApiGetQRNGdecimal**)



Входные параметры: **qty** <Integer>, **min** <Integer>, **max** <max>.

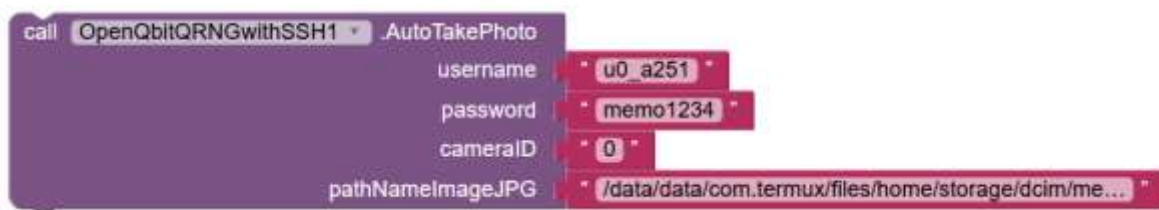
Выходные параметры: Дает количество "qty" случайных квантовых чисел, введенных во входные данные, числа находятся в диапазоне min и max в формате JSON.

Пример:

qty = 8, min = 1, max = 100; выход: {"результат": [3, 53, 11, 2, 66, 44, 9, 78]}

Описание: API генератора квантовых случайных чисел (QRNG)

Блокировать, чтобы сделать снимок автоматически - (**AutoTakePhone**)



Входные параметры: имя пользователя < Строка>, пароль < Строка>, идентификатор камеры < Строка>, pathNameImageJPG < Строка>.

Обязательная зависимость: для использования этого блока необходимо выполнить две программные зависимости; установить в терминале Termux модуль Termux-API. Этот модуль содержит процесс автоматической фотосъемки и загрузки SSH-сервера, который был ранее установлен.

Выходные параметры: передает фотографию (изображение) в формате JPG по указанному пути. На пути вы должны

Описание: Создает фото в формате JPG автоматически без вмешательства пользователя.

Для установки Termux-API в терминале Termux должна быть выполнена следующая команда:

\$ pkg установка термикс-апи

```
$ pkg install termux-api
Ign:2 https://dl.bintray.com/grimler/game-packag
es-24 games InRelease
Ign:3 https://dl.bintray.com/grimler/science-pac
kages-24 science InRelease
Ign:1 https://dl.bintray.com/termux/termux-packa
ges-24 stable InRelease
Get:5 https://dl.bintray.com/grimler/game-packag
es-24 games Release [5344 B]
Get:6 https://dl.bintray.com/grimler/science-pac
kages-24 science Release [6191 B]
Get:4 https://dl.bintray.com/termux/termux-packa
ges-24 stable Release [8255 B]
Get:7 https://dl.bintray.com/grimler/game-packag
es-24 games Release.gpg [475 B]
Get:8 https://dl.bintray.com/grimler/science-pac
kages-24 science Release.gpg [475 B]
Get:9 https://dl.bintray.com/termux/termux-packa
ges-24 stable Release.gpg [821 B]
0% [8 Release.gpg gpgv 6191 B]
```

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be upgraded:
  termux-api
1 upgraded, 0 newly installed, 0 to remove and 7
4 not upgraded.
Need to get 21.2 kB of archives.
After this operation, 4096 B of additional disk
space will be used.
Get:1 https://dl.bintray.com/termux/termux-packa
ges-24 stable/main arm termux-api arm 0.50-1 [21
.2 kB]
Fetched 21.2 kB in 1s (18.7 kB/s)
(Reading database ... 25317 files and directorie
s currently installed.)
Preparing to unpack .../termux-api_0.50-1_arm.de
b ...
Unpacking termux-api (0.50-1) over (0.50) ...
Setting up termux-api (0.50-1) ...
$
```

Чтобы узнать номер, количество и расположение идентификаторов (идентификаторов фотообъективов) вашего мобильного устройства (смартфона), выполните следующую команду в терминале Termux.

\$ termux-camera-info

```
$ termux-camera-info
[
  {
    "id": "0",
    "facing": "back",
    "jpeg_output_sizes": [
      {
        "width": 4160,
        "height": 3120
      },
      {
        "width": 4160,
        "height": 2340
      },
      {
        "width": 4160,
        "height": 2080
      }
    ],
    "width": 4160,
    "height": 2080
  },
  {
    "id": "1",
    "facing": "front",
    "jpeg_output_sizes": [
      {
        "width": 3264,
        "height": 2448
      },
      {
        "width": 2560,
        "height": 1920
      }
    ],
    "width": 2560,
    "height": 1920
  }
]
```

```
    "manual_sensor",
    "manual_post_processing",
    5,
    6,
    4,
    7,
    "raw"
  ],
  {
    "id": "1",
    "facing": "front",
    "jpeg_output_sizes": [
      {
        "width": 2560,
        "height": 1920
      },
      {
        "width": 2560,
        "height": 1600
      },
      {
        "width": 2560,
        "height": 1600
      }
    ],
    "width": 2560,
    "height": 1600
  }
]
```

В нашем примере смартфон LG Q6, который мы использовали, имеет два идентификатора "0" сзади и "1" спереди.

Теперь давайте протестируем API, чтобы сделать снимок, используя ID "0" заднего объектива, и дадим ему альбомное имя в нашем случае test.jpg.

Помните, что API доставляет фотографии только в формате JPG:

### \$ termux-камера-фото -с 0 test.jpg

Предыдущая команда должна была автоматически создать файл с именем test.jpg, если мы можем использовать блок (AutoTakePhoto), не забудьте запустить наш локальный SSH-сервер командой: \$ sshd

**ПРИМЕЧАНИЕ:** В переменной pathNameImageJPG следует учитывать путь внутри терминала Termux для доступа к хранилищу смартфона:

/data/data/com.termux/files/home/storage/dcim/example.jpg

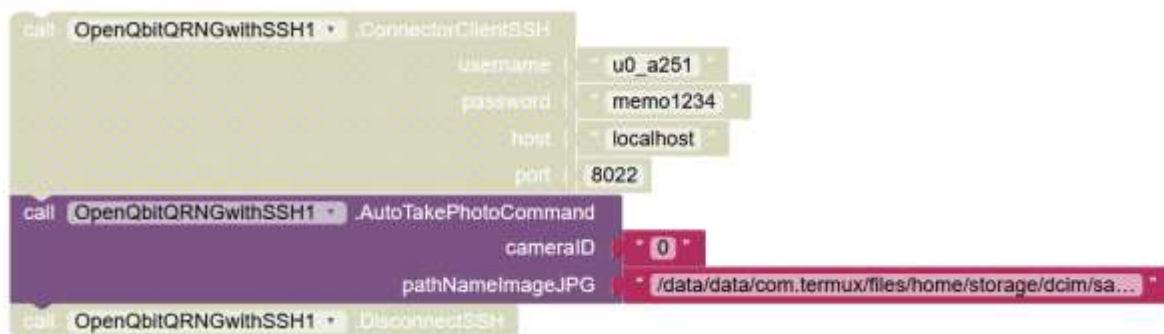
Предыдущий маршрут на андроиде будет таким же:

/mnt/sdcard/dcim/example.jpg

Однако следует помнить, что в терминале Termux действительный маршрут для просмотра включения мобильного телефона всегда должен рассматриваться как маршрут по умолчанию:

/данные/данные/ком.termux/файлы/домашний/хранилище

Блок для автоматической фотосъемки ТОЛЬКО КОМАНДЫ - (AutoTakePhoneCommand)



Входные параметры: cameraID <String>, pathNameImageJPG <String>.

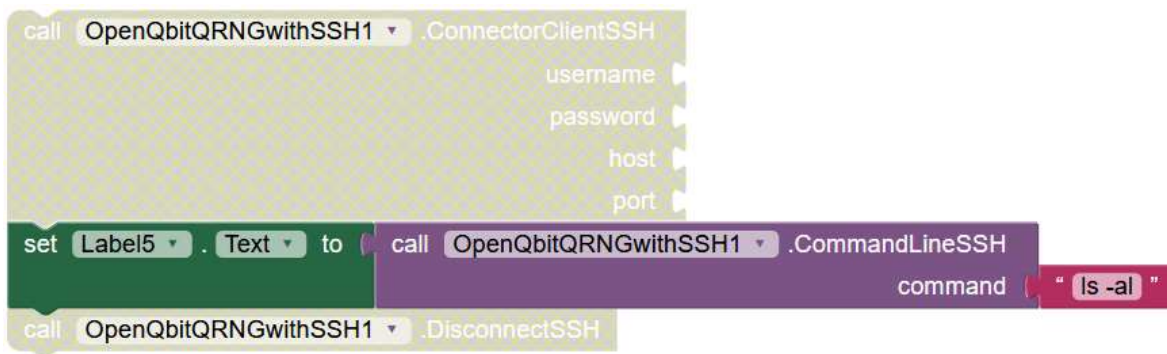
Обязательная зависимость: Блок (ConnectorClientSSH), Блок (DisconnectSSH).

Выходные параметры: передает фотографию (изображение) в формате JPG по указанному пути.



Описание: Создает фото в формате JPG автоматически без вмешательства пользователя. Однако, отличие от блока (AutoTakePhoto), этот блок содержит только команду создания фотографии, и вам нужно, чтобы блок сначала подключился к серверу SSH (**ConnectorClientSSH**), а затем использовал блок (**DisconnectSSH**).

Блок для выполнения команды в терминале Termux - (**CommandLineSSH**)



Входные параметры: **команда** <String>

Обязательная зависимость: Блок (**ConnectorClientSSH**), Блок (**DisconnectSSH**).

Выходные параметры: выполнить команду, введенную в терминале Termux.

Описание: Введенная команда выполняется и блок необходим сначала для подключения к серверу SSH (**ConnectorClientSSH**), а затем для использования блока (**DisconnectSSH**).

Блок для подключения к удаленному или локальному SSH-серверу - (**ConnectorClientSSH**).



Входные параметры: **имя пользователя** <string>, **пароль** <string>, **хост** <string>, **порт** <integer>.

Выходные параметры: если соединение с ssh-сервером терминала Termux успешно, то выдается сообщение; **"Connect SSH"**, если не успешно, выдается сообщение **NULL**.

Описание: Коммуникационный блок для подключения выбранного SSH-сервера к терминалу Termux, через протокол связи SSH (Secure Shell).



Блок декодирования файла с алгоритмом Base64 (**DecoderFileBase64**).

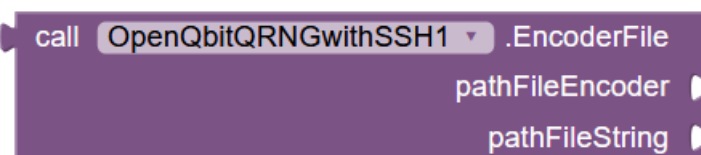


Входные параметры: **pathFileBase64** <String>, **pathFileOrigin** <String>.

Выходные параметры: Исходный файл, который был введен в блоке (**EncoderFileBase64**)

Описание: Файл Base64 преобразуется в исходный файл, который был вставлен в блок (**EncoderFileBase64**).

Блок преобразует файл в формат Base64 - (**EncoderFile**)

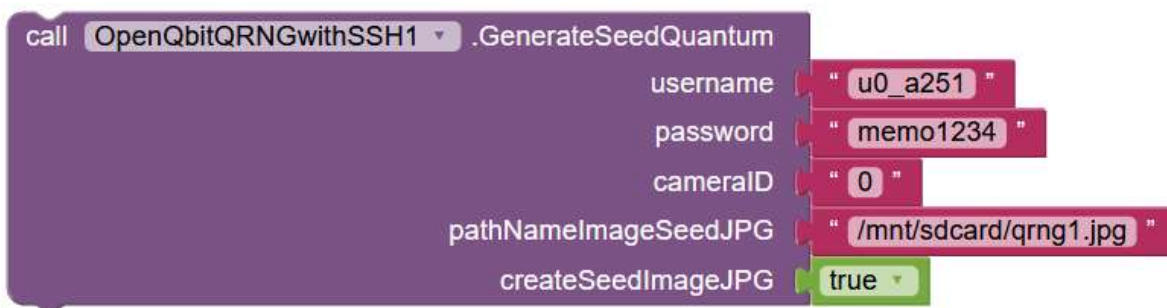


Входные параметры: **pathFileOrigin** <String> , **pathFileBase64** <String>.

Выходные параметры: файл в кодировке Base64.

Описание: Преобразует исходный файл любого формата в файл Base64. Имена файлов могут быть произвольными и выбираться пользователем.

Блок для генерации QRNG (Генератор квантовых случайных чисел) - (**Генерировать посевной квант**)



Входные параметры: **имя пользователя** <String>, **пароль** <String>, **идентификатор камеры** <String>, **pathNameImageJPG** <String>. **createSeedImageJPG** <Boolean>.

Если значение Булева является "True", когда блок выполняется каждый раз при создании нового посадочного JPG-изображения с именем введенного пути. Если значение Boolean равно "False", то мы деактивируем опцию съемки JPG изображения (фото) и можем вручную указать, где находится изображение в нашем выборе, которое может быть любого формата.

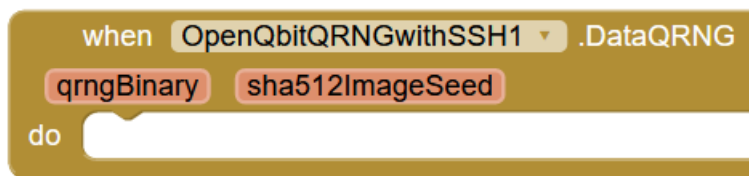
ПРИМЕЧАНИЕ: Наилучшие результаты для генерации QRNG основаны на "RAW" отформатированном изображении. Пример формата DNG.

Обязательная зависимость: вышеупомянутый Termux-API должен быть установлен в блоке (**AutoTakePhoto**).

Выходные параметры: событие выполняется (**DataQRNG**) и дает нам два значения:

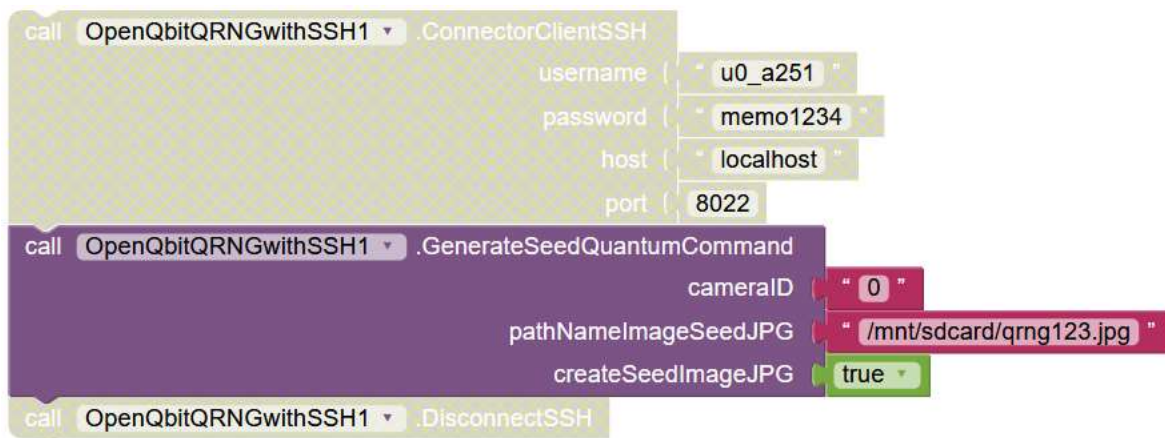
qrngBinary.- Строка случайных двоичных чисел.

Sha512ImageSeed - Sha512 из изображения семян JPG, откуда пришли случайные числа.



Описание: Генерирует квантовые случайные числа (QRNG) через оптический датчик камеры мобильного телефона. Алгоритм основан на захвате случайных фотографий, и алгоритм применяется к нему для доставки строки двоичных чисел.

Блок для генерации QRNG (Quantum Random Number Generator) - (GenerateSeedQuantumCommand).



Входные параметры: **cameraID** <String>, **pathNameImageJPG** <String>. **createSeedImageJPG** <Boolean>.

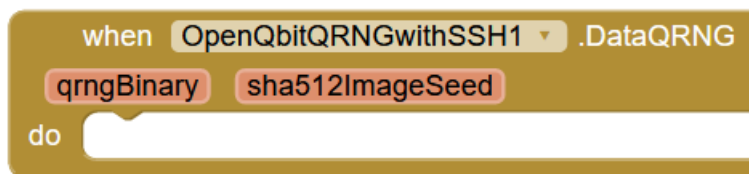
Если значение Булева является "True", когда блок выполняется каждый раз при создании нового посадочного JPG-изображения с именем введенного пути. Если значение Boolean равно "False", то мы деактивируем опцию съемки JPG изображения (фото) и можем вручную указать, где находится изображение в нашем выборе, которое может быть любого формата.

Обязательная зависимость: Блок (**ConnectorClientSSH**), Блок (**DisconnectSSH**).

Выходные параметры: событие выполняется (**DataQRNG**) и дает нам два значения:

**qrngBinary**.- Строка случайных двоичных чисел.

**Sha512ImageSeed** - Sha512 из изображения семян JPG, откуда пришли случайные числа.



Описание: Генерирует квантовые случайные числа (QRNG) через оптический датчик камеры мобильного телефона. Алгоритм основан на захвате случайных фотографий, и алгоритм применяется к нему для доставки строки двоичных чисел.

Однако, отличие от блока (**GenerateSeedQuantum**), этот блок содержит только команду на выполнение алгоритма реакции QRNG и вам нужно, чтобы блок сначала подключился к SSH серверу (**ConnectClientSSH**), а затем использовал блок (**DisconnectSSH**).

Блок для получения энтропии Шеннона на изображении (фото) - (**GetShannonEntropyFile**).



Входные параметры: **имя пользователя** <String>, **пароль** <String>, **pathFileImage** <String>.

Обязательная зависимость: модуль `Shannon_entropy` должен быть установлен в терминале Termux.

Выходные параметры: Показывает энтропию изображения.

Пример:

Выход: 8.94596789873

Описание: Дает энтропию изображения. Энтропия является фундаментальным параметром для генерации случайных чисел хорошего качества, чем выше энтропия, тем лучше результаты.

Для установки модуля энтропии `Shannon` сначала необходимо установить пакет `Python`, а затем установить модуль `Pillow` и `Shannon_entropy` со следующими командами в терминале Termux.

\$ подходящая установка Python

\$ установка трубопроводов Подушка

\$ трубопровод установка Shannon\_entropy

```
$ apt install python
Reading package lists... Done
Building dependency tree
Reading state information... Done
python is already the newest version (3.8.3).
0 upgraded, 0 newly installed, 0 to remove and 0
not upgraded.
$
```

```
$ pip install Pillow
Requirement already satisfied: Pillow in /data/d
ata/com.termux/files/usr/lib/python3.8/site-pack
ages (7.2.0)
$
```

```
$ pip install shannon_entropy
Requirement already satisfied: shannon_entropy i
n /data/data/com.termux/files/usr/lib/python3.8/
site-packages (0.2.1)
Requirement already satisfied: Pillow in /data/d
ata/com.termux/files/usr/lib/python3.8/site-pack
ages (from shannon_entropy) (7.2.0)
$
```

Затем мы должны создать в каталоге "Home" компании Termux следующий Python-файл под названием "entropy.py" со следующим кодом внутри.

```
Из PIL импорта Изображение

импортная математика

От импорта Shannon_entropy *
```

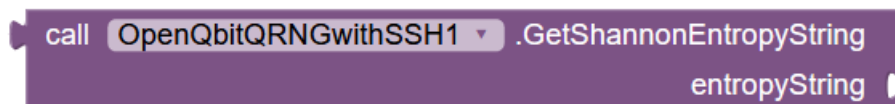
Импортные системы

```
Img=Image.open(sys.argv[1])  
  
print(Shannon_entropy(img))
```

Мы сохраняем файл и имеем наше окружение для использования с блоком (GetShannonEntropyFile).

**Совет:** Фактически, с помощью этой установки Python вы можете создавать свои собственные программы на этом языке и запускать их через блок (ConnectorClientSSH).

Блок для получения энтропии Шеннона из строки - (GetShannonEntropyString).



Входные параметры: энтропияString>String>

Выходные параметры: Показывает энтропию строки символов.

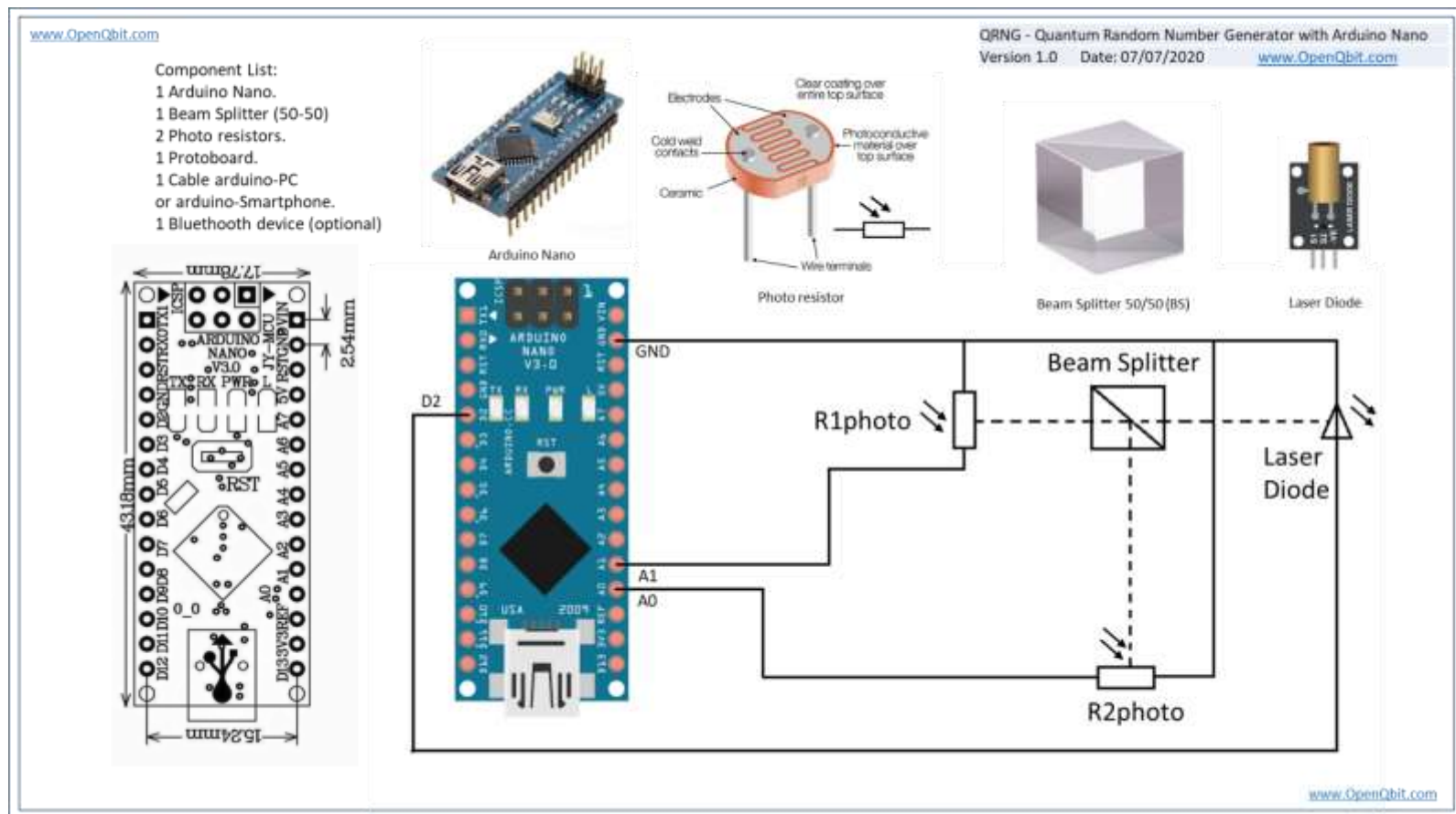
Пример:

Выход: 5.76002345671

Описание: Дает нам энтропию строки символов. Энтропия является фундаментальным параметром для генерации случайных чисел хорошего качества, чем выше энтропия, тем лучше результаты.

## 10. Создание "Аппаратного" устройства QRNG.

Теперь мы создадим физическое "аппаратное" устройство для генерации квантовых случайных чисел (QRNG) с недорогими компонентами, которые можно легко собрать дома.



[www.OpenQbit.com](http://www.OpenQbit.com)

## QRNGv1.0.ino

Software  
Program to arduino nano.

```

/* OpenQbitQRNG Firmware V1.0
*Author: Guillermo Vidal
*Copyright © 2020 OpenQbit, Inc.
*License: MIT
*/

```

```

int triggerQ = 2; // This pin will pulse our quantum circuit
int QuA0Pin = A0; // This pin measures the horizontal polarized photons
int QuA1Pin = A1; // This pin measures the vertically polarized photons
float Qu0 = 0;
float Qu1 = 0;

void setup() {
  // Just setting up triggerPin and serial connection
  pinMode(triggerQ, OUTPUT); // sets the digital pin 2 as output
  Serial.begin(9600);
}

int Random() {
  // Pulse the laser
  digitalWrite(triggerQ, HIGH);
  delay(300);
  digitalWrite(triggerQ, LOW);
  delay(300);
  // Read the photoresistors
  Qu0 = analogRead(QuA0Pin);
  Qu1 = analogRead(QuA1Pin);
  // Determine random bit
  if(Qu0>Qu1) { // More photons in the Qu0 mode, return 0
    return 0;
  } if(Qu0 < Qu1) { // More photons in the Qu1 mode, return 1
    return 1;
  } else {
    /* The same number of photons are in both modes!
    This is actually not an uncommon occurrence, for our
    purposes we will simply run the function recursively until
    a random bit can be generated.
    */
    Random();
  }
}

void loop() {
  Serial.print(Random());
}

```

## Output console

0010110101011110101011010.....


QRNG - Quantum Random Number Generator with Arduino Nano  
Version 1.0    Date: 07/07/2020    [www.OpenQbit.com](http://www.OpenQbit.com)

[www.OpenQbit.com](http://www.OpenQbit.com)



## Мини QRNG - DIY - "Сделай это сам".

Компиляция программы QRNGv10.ino и загрузка в тяжелые нано-...



```
QRNGv10 Arduino 1.8.10
Archivo Editar Programa Herramientas Ayuda

QRNGv10

int triggerQ = 2; // This pin will pulse our quantum circuit.
int Qu0Pin = A0; // This pin measures the horizontal polarised photons
int Qu1Pin = A1; // This pin measures the vertically polarised photons
float Qu0 = 0;
float Qu1 = 0;

void setup() {
  // Just setting up triggerPin and serial connection
  pinMode(triggerQ, OUTPUT); // sets the digital pin 2 as output
  Serial.begin(5000);
}

int Random() {
  // Pulse the Laser
  digitalWrite(triggerQ, HIGH);
  delay(300);
  digitalWrite(triggerQ, LOW);
  delay(300);
  // Read the photoreistors
  Qu0 = analogRead(Qu0Pin);
  Qu1 = analogRead(Qu1Pin);
  // Determine random bit
  if(Qu0>Qu1) { // More photons in the Qu0 mode, return 0
    return 0;
  } if(Qu0 < Qu1) { // More photons in the Qu1 mode, return 1
    return 1;
  } else {
    // The same number of photons are in both modes!
    // This is actually not an uncommon occurrence, but our
    // purposes we will simply run the function sequentially until
    // a random bit can be generated.
    return Random();
  }
}

void loop() {
  Serial.println(Random());
}
```

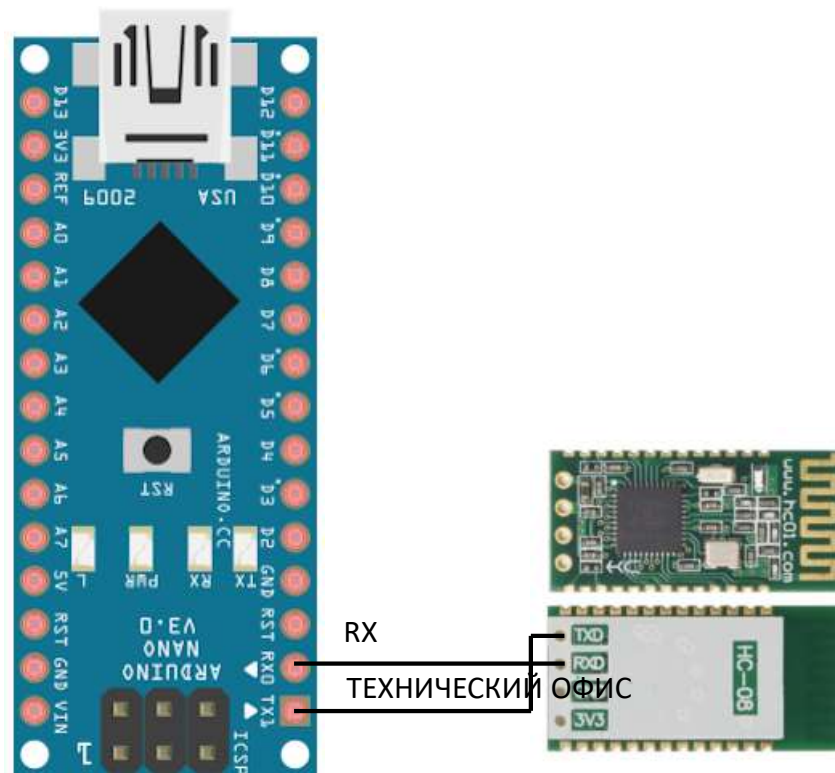
Compilado

El Sketch usa 2352 bytes (7%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.  
Las variables Globales usan 198 bytes (1%) de la memoria dinámica, dejando 1840 bytes para las variables locales. El máximo es 2048 bytes.

40 ArduinoGetters Ltd en CDM

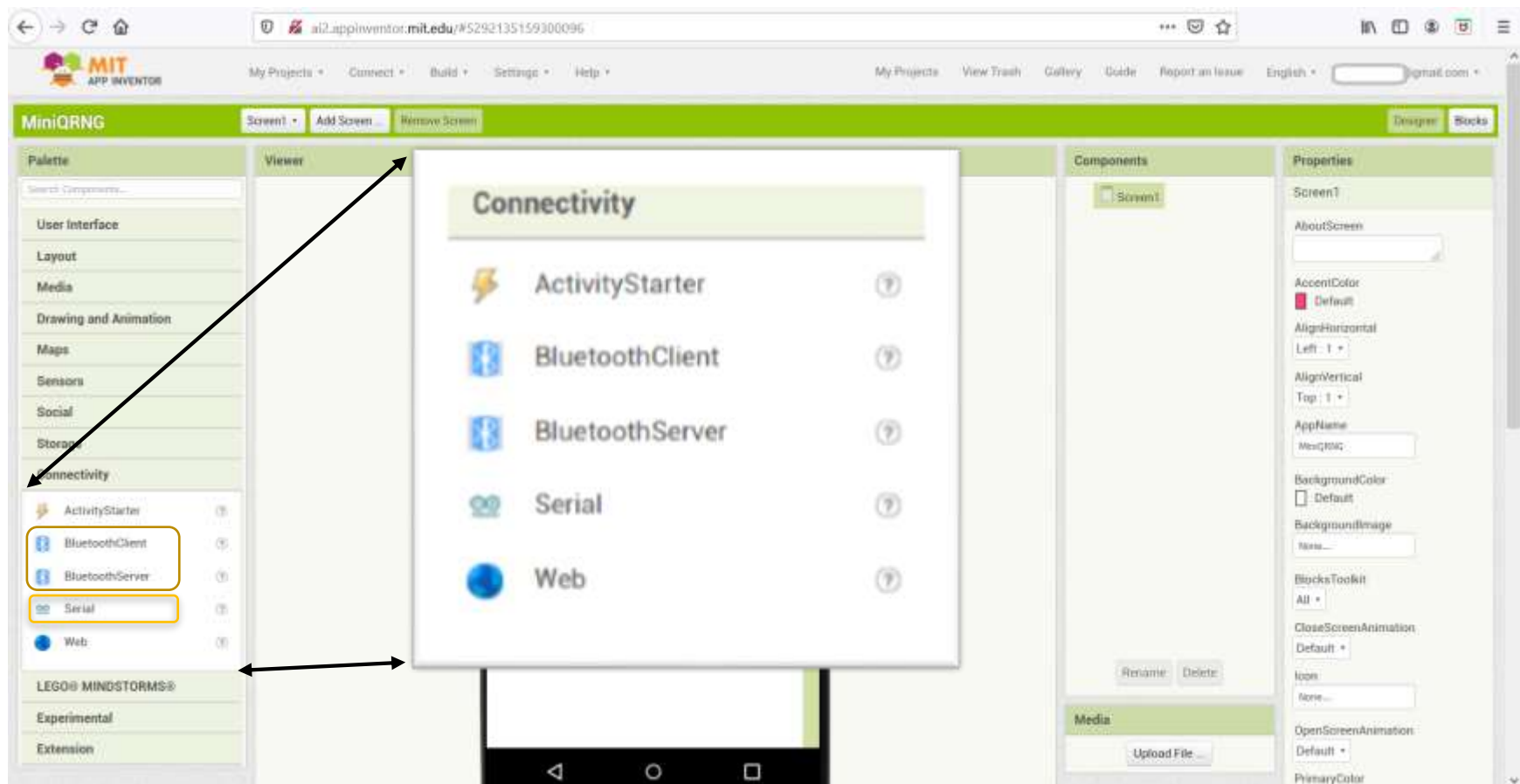
Есть два способа связи с напряженным нано, один - через последовательный порт, другой - через соединение Bluetooth.

Для очень простого Bluetooth-соединения нам достаточно купить модуль HC-08 или аналогичный и подключить его следующим образом:



## Мини QRNG - DIY - "Сделай это сам".

Следующие компоненты последовательного интерфейса или Bluetooth могут быть использованы для подключения App Inventor к Arduino:



Теперь скомпилированная и загруженная программа QRNGv10.ino только не общается со сложными нано для сохранения данных (квантовые случайные числа) они будут в двоичном формате, однако, полученные данные легко могут быть переданы в другой формат, такой как шестнадцатеричный или десятичный в зависимости от конечного требования.

Наконец, чтобы посмотреть пример того, как работает последовательное или Bluetooth-соединение, вот несколько ссылок.

Помните, что все через Blockly программирования для тестирования с App Inventor это уже есть блоки для связи с arduino серийный или другой блочной системы может быть похожим на Bluetooth tarves онлайн.

<http://kio4.com/appinventor/9A0 bluetooth RXTX.htm>

<http://kio4.com/appinventor/index.htm#bluetooth>

<https://community.appinventor.mit.edu/>

## 11. Приложение "OpenQbit Quantum Computing".

### Как работают квантовые вычисления? <sup>(2)</sup>

Цифровое преобразование приводит к изменениям в мире быстрее, чем когда-либо. Вы бы поверили, что цифровая эра вот-вот закончится? **Цифровая грамотность** уже была определена как область, в которой открытые знания и доступные возможности для изучения технологий являются насущной необходимостью для устранения пробелов в социальном и экономическом развитии. Обучение на основе ключевых концепций цифровой эры станет еще более критичным с неизбежным приходом еще одной новой технологической волны, способной трансформировать существующие модели с поразительной скоростью и мощностью: **квантовые технологии**.

В этой статье мы сравниваем основные понятия традиционных и квантовых вычислений, а также начинаем изучать их применение в других смежных областях.

Что такое квантовые технологии?

На протяжении всей истории люди разрабатывали технологии, понимая, как работает природа с помощью науки. Между 1900 и 1930 годами, изучение некоторых физических явлений, которые еще не были хорошо изучены, привело к созданию новой физической теории, **Квантовая механика**. Эта теория описывает и объясняет функционирование микроскопического мира, естественную среду обитания молекул, атомов или электронов. Благодаря этой теории удалось не только объяснить эти явления, но и понять, что субатомная реальность работает совершенно интуитивно, почти магически, и что в микроскопическом мире происходят события, которые не происходят в макроскопическом мире.


Эти квантовые **свойства** включают в себя квантовое суперпозиционирование, квантовую связь и квантовую телепортацию.

- **Квантовая суперпозиция** описывает, как частица может находиться в разных состояниях одновременно.
- **Квантовая связь** описывает, как две частицы, находящиеся так далеко друг от друга, могут быть соотнесены таким образом, что при взаимодействии с одной из них другая осознает это.
- **Квантовая телепортация** использует квантовую связь для передачи информации из одного места в другое в пространстве без необходимости путешествовать через нее.

Квантовые технологии основаны на этих квантовых свойствах субатомной природы.

В этом случае сегодня понимание микроскопического мира через Квантовую механику позволяет нам изобретать и проектировать технологии, способные улучшить жизнь людей. Существует много и очень разных технологий, использующих квантовые явления, и некоторые из них, такие как лазеры или магнитно-резонансная томография (МРТ), существуют у нас уже более полувека. Однако в настоящее время мы являемся свидетелями технологической революции в таких областях, как квантовые вычисления, квантовая информация, квантовое моделирование, квантовая оптика, квантовая метрология, квантовые часы или квантовые датчики.

Что такое квантовые вычисления? Во-первых, вы должны понять классические



**FIGURA 1.**  
Ejemplos de caracteres en lenguaje binario.

Caracter	Bits
7	111
A	01000001
\$	00100100
:)	0011101000101001

вычисления.

Чтобы понять, как работают квантовые компьютеры, удобно сначала объяснить, как работают компьютеры, которыми мы пользуемся каждый день и которые мы будем называть в этом документе цифровыми или классическими компьютерами. В них, как и в других электронных устройствах, таких как планшетные компьютеры или мобильные телефоны, биты используются в качестве основных блоков памяти. Это означает, что программы и приложения кодируются в битах, то есть в двоичном языке нулей и единиц. Каждый раз, когда мы взаимодействуем с любым из этих устройств, например, нажимая клавишу на клавиатуре, внутри компьютера создаются, уничтожаются и/или модифицируются строки нулей и единиц.

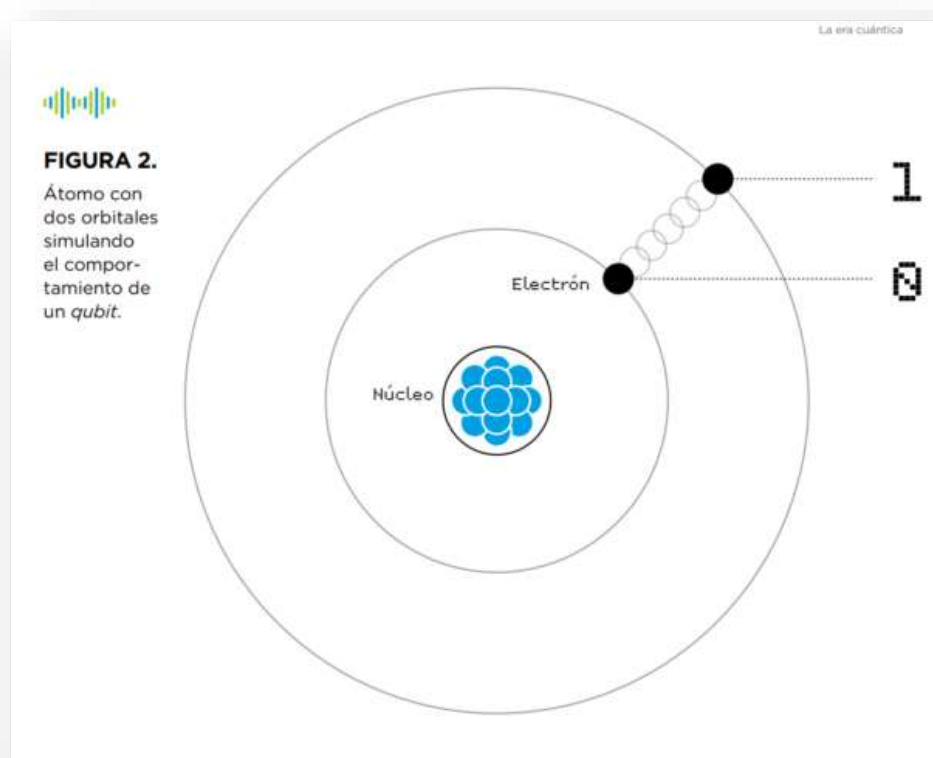
Интересный вопрос - что это за нули и физические нули внутри компьютера? Ноль и одно состояние соответствуют электрическому току, который циркулирует или не циркулирует через микроскопические части, называемые транзисторами, которые действуют как переключатели. Когда ток не протекает, транзистор "выключен" и соответствует биту 0, а когда он протекает, то "включен" и соответствует биту 1.

Проще говоря, биты 0 и 1 как будто соответствуют дыркам, так что пустая дырка - это бит 0, а дырка, занятая электроном, - это бит 1. Поэтому эти устройства называются

электроникой. В качестве примера на рисунке 1 показана запись некоторых символов в двоичном виде. Теперь, когда у нас есть представление о том, как работают современные компьютеры, давайте попробуем понять, как работают кванты.

### От битов к кубитам

Фундаментальной единицей информации в квантовых вычислениях является квантовый бит или кбит. Кубиты по определению являются двухуровневыми квантовыми системами - здесь мы увидим примеры, - которые, как и биты, могут быть на низком уровне, который соответствует состоянию низкого возбуждения или энергии, определяемой как 0, или на высоком уровне, который соответствует состоянию высокого возбуждения или определяемой как 1. Однако, и в этом заключается фундаментальное различие с классическими вычислениями, кбиты могут также находиться в любом из бесконечных промежуточных состояний между 0 и 1, например, в состоянии, которое составляет половину 0 и половину 1, или в трех четвертях 0 и четверти 1.



Квантовые алгоритмы, экспоненциально более мощные и эффективные вычисления

Целью квантовых компьютеров является использование этих квантовых свойств *кбитов*, как квантовых систем, которыми они являются, для запуска квантовых алгоритмов, которые используют перекрытие и чередование, чтобы обеспечить гораздо большую вычислительную мощность, чем классика. Важно отметить, что [OpenQbit.com](http://OpenQbit.com)

реальное изменение парадигмы заключается не в том, чтобы делать то же самое, что делают цифровые или классические компьютеры - нынешние, но более быстрые, как это можно прочесть во многих статьях, а в том, что квантовые алгоритмы позволяют выполнять определенные операции совершенно по-другому, что во многих случаях оказывается более эффективным - то есть, за гораздо меньшее время или с использованием гораздо меньшего количества вычислительных ресурсов.

Давайте посмотрим на конкретный пример. Давайте представим, что мы находимся в Боготе и хотим знать лучший маршрут, чтобы добраться до Лимы из миллиона вариантов ( $N=1,000,000$ ). Чтобы использовать компьютеры для поиска оптимального маршрута, необходимо оцифровать 1 000 000 вариантов, что подразумевает их перевод на битовый язык для классического компьютера и в *квиты* для квантового компьютера. В то время как классический компьютер должен будет идти один за другим, анализируя все пути до тех пор, пока не найдет нужный, квантовый компьютер использует процесс, известный как квантовый параллелизм, который позволяет ему рассматривать все пути одновременно. Это означает, что если классическому компьютеру нужен порядок  $N/2$  шагов или итераций, т. е. 500 000 попыток, то квантовый компьютер найдет оптимальный путь только после операций с  $\sqrt{N}$  в реестре, т. е. 1000 попыток.

В предыдущем случае преимущество квадратично, но в других случаях оно даже экспоненциально, что означает, что при  $n$  *кубитах* мы можем получить вычислительную мощность, эквивалентную  $2^n$  бит. В качестве примера можно привести то, что с помощью примерно 270 кубитов мы могли бы иметь в квантовом компьютере больше базовых состояний - более разных и одновременных символьных строк - чем количество атомов во Вселенной, которое оценивается примерно в 280. Другим примером является то, что с помощью квантового компьютера в диапазоне от 2000 до 2500 *кубитов* мы могли бы сломать практически всю используемую сегодня криптографию (так называемая криптография с открытым ключом).

Почему важно знать о квантовых технологиях?

Мы находимся в момент цифровой трансформации, когда различные новые технологии, такие как блок-цепь, искусственный интеллект, беспилотники, интернет вещей, виртуальная реальность, 5G, 3D-принтеры, роботы или автономные транспортные средства имеют все больше и больше присутствия в различных областях и секторах. Эти технологии, призванные повысить качество жизни человека, ускоряя развитие и создавая социальное воздействие, сегодня развиваются параллельно. Лишь в редких случаях мы видим компании, разрабатывающие продукты, использующие комбинации двух или более из этих технологий, такие как блок-цепь и IoT или беспилотные летательные аппараты и искусственный интеллект. Несмотря на то, что им суждено сойтись, тем самым создавая экспоненциально большее влияние, начальная стадия разработки, в которой они находятся, и нехватка разработчиков и людей с



техническими профилями означают, что конвергенция все еще остается нерешенной задачей.

Ожидается, что в силу своего разрушительного потенциала квантовые технологии не только сблизятся со всеми этими новыми технологиями, но и окажут сквозное воздействие практически на все из них. Квантовые вычисления будут угрожать аутентификации, обмену и безопасному хранению данных, оказывая серьезное влияние на те технологии, в которых криптография играет более значимую роль, такие как кибербезопасность или блок-цепочка, а также оказывают незначительное негативное влияние, но также должны учитываться в таких технологиях, как 5G, IoT или беспилотные летательные аппараты.

Хочешь попрактиковаться в квантовых вычислениях?

Десятки квантовых компьютерных симуляторов уже доступны в сети с различными уже используемыми языками программирования, такими как C, C++, Java, Matlab, Maxima, Python или Octave. Также появились новые языки, такие как Q#, запущенные Microsoft. Вы можете исследовать и играть с виртуальной квантовой машиной через такие платформы, как IBM и Rigetti.

Mini QRNG создан компанией OpenQbit.com, которая специализируется на разработке технологий на основе квантовых вычислений для различных типов секторов, как частного, так и государственного.

*Почему Mini QRNG отличается от других QRNG просто потому, что система была создана по модульному принципу и может быть легко собрана дома по достаточно низкой цене.*

(1) <https://blogs.iadb.org/conocimiento-abierto/es/como-funciona-la-computacion-cuantica/>

## 12. Лицензирование и использование программного обеспечения.

Android

<https://source.android.com/setup/start/licenses>

Termux

<https://github.com/termux/termux-app/blob/master/LICENSE.md>

Узел

<https://raw.githubusercontent.com/nodejs/node/master/LICENSE>

Питон

<https://www.python.org/download/releases/2.7/license/>

OpenSSH

<https://www.openssh.com/features.html>

Шпатлевка SSH

<https://www.chiark.greenend.org.uk/~sgtatham/putty/licence.html>

Изобретатель приложения MIT 2 Компаньон и Изобретатель приложения Блокнотно

<https://appinventor.mit.edu/about/termsofservice>

Внешние расширения:

JSOOTOOLS

<https://thunkableblocks.blogspot.com/2017/07/jsontools-extension.html>

Лицензионные открытые и коммерческие версии системы QRNG Mini см. на официальном сайте <http://www.openqbit.com>.

Mini QRNG, Mini BlocklyChain, MiniBlockly, BlocklyCode, MiniBlockMiniChain, QBlockly son marcas registradas por OpenQbit.

Мини QRNG является общественным достоянием.

Весь код и документация в Mini QRNG была посвящена общественному достоянию авторов. Все авторы кода и представители компаний, на которые они работают, подписали affidavits, посвященные их вкладу в общественное достояние, и оригиналы этих affidavits хранятся в сейфе в штаб-квартире OpenQbit в Мексике. Любой человек может свободно публиковать, использовать или распространять оригинальные расширения Mini QRNG (OpenQbit), как в виде исходного кода, так и в виде скомпилированных двоичных файлов, для любых целей, коммерческих или некоммерческих, и любыми средствами.

Предыдущий пункт касается кода и документации, поставляемых в Mini QRNG, тех частей библиотеки Mini QRNG, которые фактически группируются и поставляются с более широким приложением. Некоторые сценарии, используемые в процессе

компиляции (например, "конфигурационные" сценарии, генерируемые autoconf), могут быть включены в другие лицензии с открытым исходным кодом. Однако ни один из этих сценариев компиляции не превращает его в окончательную библиотеку QRNG Mini, поэтому лицензии, связанные с этими сценариями, не должны быть фактором при оценке ваших прав на копирование и использование библиотеки QRNG Mini.

Весь код доставки в Mini QRNG был написан с нуля. Ни один код не был взят из других проектов или из открытого интернета. Каждая строка кода может быть прослежена до ее первоначального автора, и все эти авторы имеют в файле посвящения в общественное достояние. Таким образом, база кодов Мини QRNG чиста и не загрязнена кодами лицензированными из других проектов с открытым исходным кодом, а не из открытых источников.

Mini QRNG является открытым исходным кодом, что означает, что вы можете сделать столько копий, сколько вы хотите, и делать то, что вы хотите с этими копиями, без ограничений. Но Mini QRNG не является открытым исходным кодом. Для того чтобы сохранить Mini QRNG в общественном достоянии и обеспечить, чтобы код не был заражен проприетарным или лицензионным контентом, проект не принимает патчи от неизвестных людей. Весь код в Mini QRNG является оригинальным, так как он был написан специально для использования в Mini QRNG. Ни один код не был скопирован из неизвестных источников в Интернете.

Мини QRNG является общественным достоянием и не требует лицензирования. Тем не менее, некоторые организации хотят получить юридические доказательства своего права на использование Мини QRNG. Обстоятельства, при которых это происходит, включают в себя следующее:

- Ваша компания хочет получить компенсацию по искам о нарушении авторских прав.
- Вы используете Mini QRNG в юрисдикции, которая не признает общественное достояние.
- Вы используете Mini QRNG в юрисдикции, которая не признает за автором права посвящать свое произведение в общественное достояние.
- Вы хотите иметь реальный юридический документ в качестве доказательства того, что вы имеете законное право использовать и распространять Mini QRNG.
- Ваш юридический отдел говорит вам, что вы должны купить лицензию.

Если какое-либо из вышеперечисленных обстоятельств относится к Вам, OpenQbit, компания, которая использует всех разработчиков Mini QRNG, продаст Вам Mini QRNG Гарантию Титула. Гарантия на право собственности - это юридический документ, в котором говорится, что заявленные авторы Mini QRNG являются истинными авторами, и что авторы имеют законное право посвятить Mini QRNG в общественное достояние, и что OpenQbit будет решительно защищать себя от претензий по лицензированию. Все

доходы от продажи гарантий на право собственности Mini QRNG используются для финансирования постоянного улучшения и поддержки Mini QRNG.

#### Внесенный код

Чтобы сохранить Mini QRNG полностью бесплатным и без роялти, проект не принимает патчи. Если вы хотите внести предложенное изменение и включить патч в качестве доказательства концепции, это было бы здорово. Однако, не обижайся, если мы перепишем твой патч с нуля. Тип некоммерческой лицензии или лицензии с открытым исходным кодом, использующий ее в данном режиме, а также некоторые подобные лицензии без приобретения поддержки как для индивидуального, так и для корпоративного использования, независимо от размера компании, будут регулироваться следующими юридическими нормами.

Отказ от гарантии. Если это не требуется действующим законодательством или не согласовано в письменной форме, Лицензиар предоставляет Работу (и каждый Поставщик предоставляет свой Вклад) "КАК ЕСТЬ", **БЕЗ ГАРАНТИЙ ИЛИ УСЛОВИЙ ЛЮБОГО РОДА, как** явных, так и подразумеваемых, включая, без ограничений, любые гарантии или условия ТИТУЛА, НЕОТВЕТСТВЕННОСТИ, ТОВАРНОЙ ПРИГОДНОСТИ ИЛИ ПРИГОДНОСТИ ДЛЯ КОНКРЕТНЫХ ЦЕЛЕЙ. Вы единственно ответственны за определение правильного использования или перераспределения Работы и за принятие любых рисков, связанных с вашим осуществлением разрешений по этой Лицензии.

Любые финансовые или иные убытки, понесенные в результате использования данного программного обеспечения, будут нести пострадавшие стороны. Все юридические споры стороны будут подавать в суд только в юрисдикции города Мехико, страны Мехико.

Для коммерческой поддержки, использования и лицензирования должно быть заключено соглашение или контракт между OpenQbit или его компанией и заинтересованной стороной.

Условия дистрибьюторского маркетинга могут быть изменены без предварительного уведомления, пожалуйста, перейдите на официальный сайт [www.openqbit.com](http://www.openqbit.com) для ознакомления с любыми изменениями в положениях о поддержке и лицензировании некоммерческих и коммерческих.

Любое физическое или юридическое лицо, пользователь, частное или публичное лицо любого юридического характера или из любой части мира, которое просто использует программное обеспечение, без каких-либо условий принимает положения,

установленные в данном документе, а также положения, которые могут быть изменены в любое время на портале [www.openqbit.com](http://www.openqbit.com) без предварительного уведомления и могут быть применены по усмотрению OpenQbit в некоммерческом или коммерческом использовании.

Любые вопросы и информация о Mini QRNG должны быть направлены в сообщество App Inventor или в различные сообщества системы Blockly, как они есть: AppBuilder, Trunkable и т.д. и / или на электронную почту [opensource@openqbit.com](mailto:opensource@openqbit.com) для запроса вопросов может занять от 3 до 5 рабочих дней, чтобы ответить на них.

Поддержка с коммерческим использованием.

[support@openqbit.com](mailto:support@openqbit.com)

Продажа для коммерческого использования.

[sales@openqbit.com](mailto:sales@openqbit.com)

Правовая информация и вопросы или проблемы лицензирования

[legal@openqbit.com](mailto:legal@openqbit.com)

