



Installation, Konfiguration und Verwaltung.

Benutzerhandbuch

Version 1.0 Beta

Juni 2020.

MiniQRNG ist ein eingetragenes Warenzeichen von OpenQbit Inc. unter einer freien und kommerziellen Lizenz. Nutzungsbedingungen unter: www.OpenQbit.com

Inhalt

1.	Einleitung.	3
2.	Was ist blockweise Programmierung?	4
3.	Was ist Termux?	4
4.	Was ist Mini-QRNG?	4
5.	Speicherkonfiguration innerhalb von Termux.	8
6.	SSH (Secure Shell)-Server-Installation.	9
7.	Konfiguration des SSH-Servers auf dem Mobiltelefon (Smartphone).	11
8.	Ambientes Blockly (App Inventor, AppyBuilder und Thunkable).	19
9.	Definition und Verwendung von Blöcken in Mini QRNG.	21
10.	Erstellung eines "Hardware"-Gerätes eines QRNG.	33
11.	Anhang "OpenQbit Quantum Computing".	39
12.	Lizenzierung und Nutzung von Software.	44

1. Einleitung.

Die heutige Kryptographie basiert auf Zufallszahlenfolgen. Die derzeit verwendeten Pseudozufallszahlengeneratoren scheinen zufällige Bitsequenzen zu liefern, aber in Wirklichkeit weisen diese Bitsequenzen bestimmte Muster auf, so dass die Gefahr besteht, gehackt oder anderweitig manipuliert zu werden, damit die Informationen über öffentliche und raubkopierte Netzwerke weitergeleitet werden.

Die Integration von Quellen physikalischer Entropie in Zufallszahlengeneratoren ist die gängigste Methode zur Überwindung dieser Sicherheitsbedrohung. Die klassische Physik ist jedoch kausal, so dass die Unvorhersagbarkeit einer mit der klassischen Physik erzeugten Bitfolge nicht bewiesen werden kann.

Die Quantenphysik hingegen ist im Wesentlichen zufällig. Die von einem Quantenzufallszahlengenerator (QRNG) erzeugten Zahlen können nicht vorhergesagt werden: QRNG ist nachweislich unvorhersehbar. Wenn also ein Quanten-Zufallszahlengenerator in einem Sicherheitssystem verwendet wird, kann selbst ein schneller Supercomputer mit schneller arithmetischer Operation die von diesem Sicherheitssystem verwendeten zufälligen Bitfolgen nicht vorhersagen.

Die Quantenphysik verwendet Methoden, die auf einem fundamentalen Konzept namens Entropie basieren.

ARTEN DER ENTROPIE

Es gibt zwei allgemeine Arten von Entropiequellen, die gemessen werden können, um echte Zufallszahlen zu erzeugen. Der erste Typ umfasst einen physikalischen Prozess, der schwer oder unmöglich zu messen oder zu rechenintensiv ist, um ihn vorherzusagen, oder beides. Dies ist eine Quelle der Caoticentropie. Ein gängiges Beispiel, das den meisten Menschen bekannt ist, ist eine Lotterie-Maschine. Ein Satz von fortlaufend nummerierten Kugeln wird in eine Kammer gelegt und durch Drehen der Kammer oder durch Blasen von Luft durch die Kammer ständig miteinander vermischt. Mehrere der Kugeln dürfen aus der Kammer herausfallen, und die auf den Kugeln markierten Zahlen stellen die Lottoziehung dar. Die Auslosung erfolgt nach dem Zufallsprinzip aufgrund der großen Anzahl von Interaktionen zwischen den Bällen und der Kamera, wodurch die Anzahl der möglichen Bewegungen jedes einzelnen Balls rasch zunimmt. Nicht nur ist die Komplexität dieser Interaktionen extrem hoch, es gibt auch keine offensichtliche Möglichkeit, alle internen Variablen der Kugeln, der Kamera und des Luftstroms genau zu beobachten oder zu messen. Eine zweite, ganz andere Art von Entropiequelle ist die Quantenmechanik. Viele mikroskopische Teilchen oder Wellen, wie Photonen, Elektronen und Protonen, haben quantenmechanische Eigenschaften, zu denen Rotation, Polarisierung, Position und Impuls gehören. Bei richtiger Konfiguration zur Erzeugung dieser Teilchen sind beispielsweise die spezifischen Werte ihrer Rotation oder

Polarisation nicht nur unbekannt und theoretisch nicht vorhersagbar, sondern werden bis zur Messung physikalisch bestimmt.

2. Was ist blockweise Programmierung?

Blockly ist eine **visuelle Programmiersprache**, die sich aus einem einfachen Satz von Befehlen zusammensetzt, die wir kombinieren können, als wären sie die Teile eines Puzzles. Es ist ein sehr nützliches Werkzeug für diejenigen, die **lernen** wollen, **wie man auf** intuitive und einfache Weise **programmiert**, oder für diejenigen, die bereits programmieren können und das Potenzial dieser Art der Programmierung erkennen wollen.

Blockly ist eine Form des Programmierens, bei der man keinen Hintergrund in irgendeiner Computersprache benötigt, weil es nur das Zusammenfügen von Grafikblöcken ist, als ob wir Lego oder ein Puzzle spielen würden, man braucht nur etwas Logik und das war's!

Jeder kann Programme für Mobiltelefone (Smartphones) erstellen, ohne sich mit diesen schwer verständlichen Programmiersprachen herumschlagen zu müssen, indem er einfach Blöcke in grafischer Form zusammensetzt.

3. Was ist Termux?

Termux ist ein Android-Terminalemulator und eine Anwendung für Linux-Umgebungen, die direkt ohne Routing oder Konfiguration arbeitet. Ein minimales Basissystem wird automatisch installiert.

Wir werden Termux wegen seiner Stabilität und der einfachen Installation und Verwaltung verwenden, Sie können jedoch eine installierte Umgebung von Ubuntu Linux für Android verwenden.

In dieser Linux-Umgebung haben Sie den "Kern" der Kommunikationsprozesse des MiniQRNG.

4. Was ist Mini-QRNG?

Mini QRNG ist Software und Hardware, die drei technologische Lösungen zur Erstellung von QRNGs (Quantum Random Number Generators) umfasst. Wie folgt klassifiziert:

- a.- QRNG-API. - Quantenzufallszahlengenerator, der von externen Servern bezogen wird.
- b.- MiniQRNG / Software. - Zufallsquantenzahlengenerator, der aus den physikalischen Eigenschaften (Quantum) der Handykamera gewonnen wird.

c.- MiniQRNG / Hardware. - Quantenzahlgenerator, der mit Hilfe von Hardware erhalten wird, die auf den quantenphysikalischen Eigenschaften eines Lasers basiert. Später werden wir Ihnen sagen, wie Sie es zu Hause zu niedrigen Kosten bauen können.

1. Installation und Konfiguration von Termux Terminal.

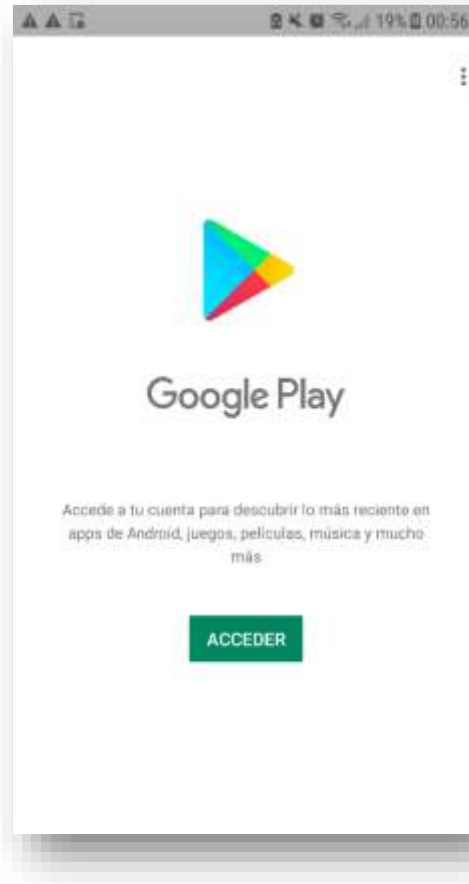
Zuerst brauchen wir eine Linux-Umgebung, da jedes Android-System auf Linux basiert, um Sicherheit und Flexibilität bei den Tools zu gewährleisten. Wir werden das "Termux"-Terminal verwenden, das diese Umgebung enthält, in der wir das/die Tool(s) installieren werden, das/die uns bei der Erstellung von QRNGs hilft.

Termux ist ein Linux-Emulator, in dem wir die notwendigen Pakete zur Erzeugung von Quantenzahlen installieren werden.

Einer der Hauptvorteile der Verwendung von Termux besteht darin, dass Sie Programme installieren können, ohne das Mobiltelefon (Smartphone) "drehen" zu müssen. Dadurch wird sichergestellt, dass durch diese Installation keine Herstellergarantie verloren geht.

Termux-Installation.

Rufen Sie von Ihrem Handy aus die Anwendung mit dem Google Play-Symbol auf (play.google.com).



Suchen Sie nach der Anwendung "Termux", wählen Sie diese aus und starten Sie den Installationsprozess.



Start der Termux-Anwendung.

Nach dem Start müssen wir die folgenden beiden Befehle ausführen, um Updates des Linux-Betriebssystememulators durchzuführen:

[OpenQbit.com](https://openqbit.com)

\$ apt Aktualisierung

\$ apt Upgrade

Bestätigen Sie alle Optionen Y(Ja)...

Termux

Home \$ apt Aktualisierung

\$ apt Upgrade



```
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:

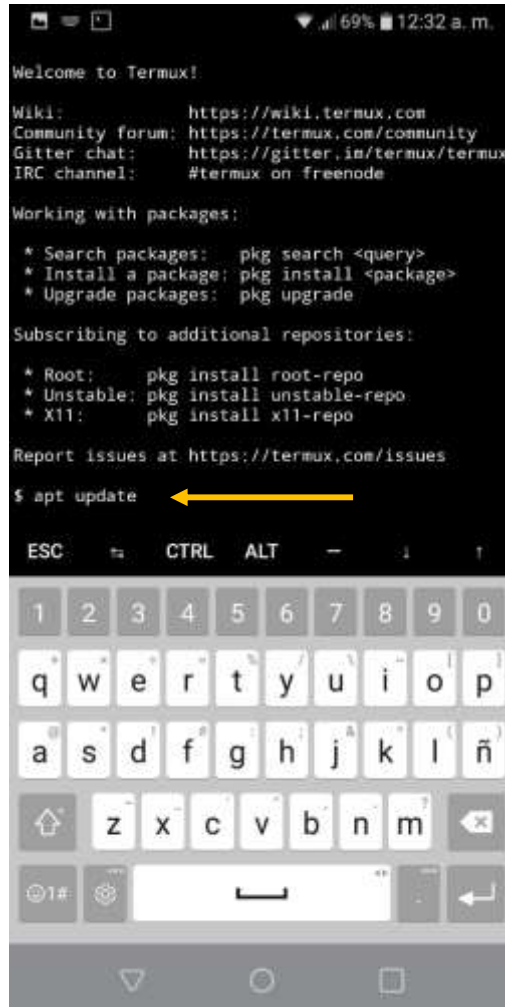
* Search packages:  pkg search <query>
* Install a package: pkg install <package>
* Upgrade packages: pkg upgrade

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. Speicherkonfiguration innerhalb von Termux.

Nachdem Sie das Termux-System aktualisiert und aufgerüstet haben, beginnen wir mit der Konfiguration, wie der interne Speicher des Telefons im Termux-System angezeigt werden

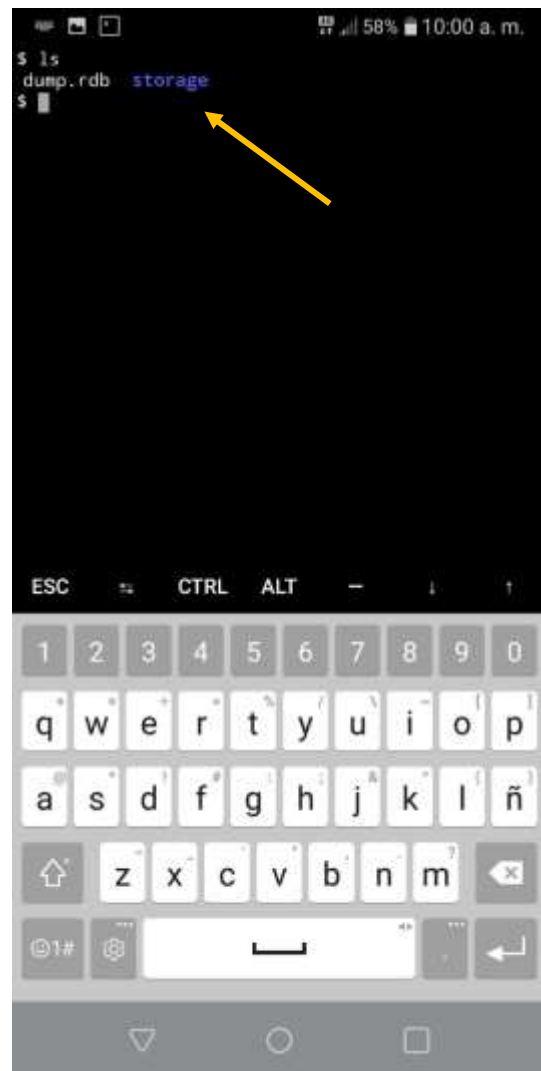
soll, damit Sie Informationen zwischen Termux und unseren Informationen im Telefon austauschen können.

Dies kann einfach und schnell durch Ausführen des folgenden Befehls auf einem Termux-Terminal erfolgen.

\$ termux-setup-storage

Wenn Sie den vorherigen Befehl ausführen, erscheint ein Fenster, in dem Sie aufgefordert werden, die Erstellung eines virtuellen **Speichers** (Verzeichnisses) in Termux zu bestätigen. Wir verifizieren, indem wir den Befehl geben:

\$ ls



6. SSH (Secure Shell)-Server-Installation.

\$ apt install openssh

\$ apt install sshpass

\$ 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 ldns libdb libedit termux-auth
The following NEW packages will be installed:
  krb5 ldns libdb 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 libdb 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]
```

Wir haben die Installation des Kommunikationsnetzes für localhost SSH-Server auf mobilen Smartphones abgeschlossen.

7. Konfiguration des SSH-Servers auf dem Mobiltelefon (Smartphone).

Wir werden es dem SSH-Server im Mobiltelefon ermöglichen, sich von unserem PC aus mit dem Mobiltelefon zu verbinden und schneller und komfortabler arbeiten zu können, außerdem wird er uns dazu dienen, zu überprüfen, ob der Dienst des SSH-Servers im Mobiltelefon korrekt funktioniert, da wir ihn bei der Kommunikation mit Mini QRNG verwenden werden.

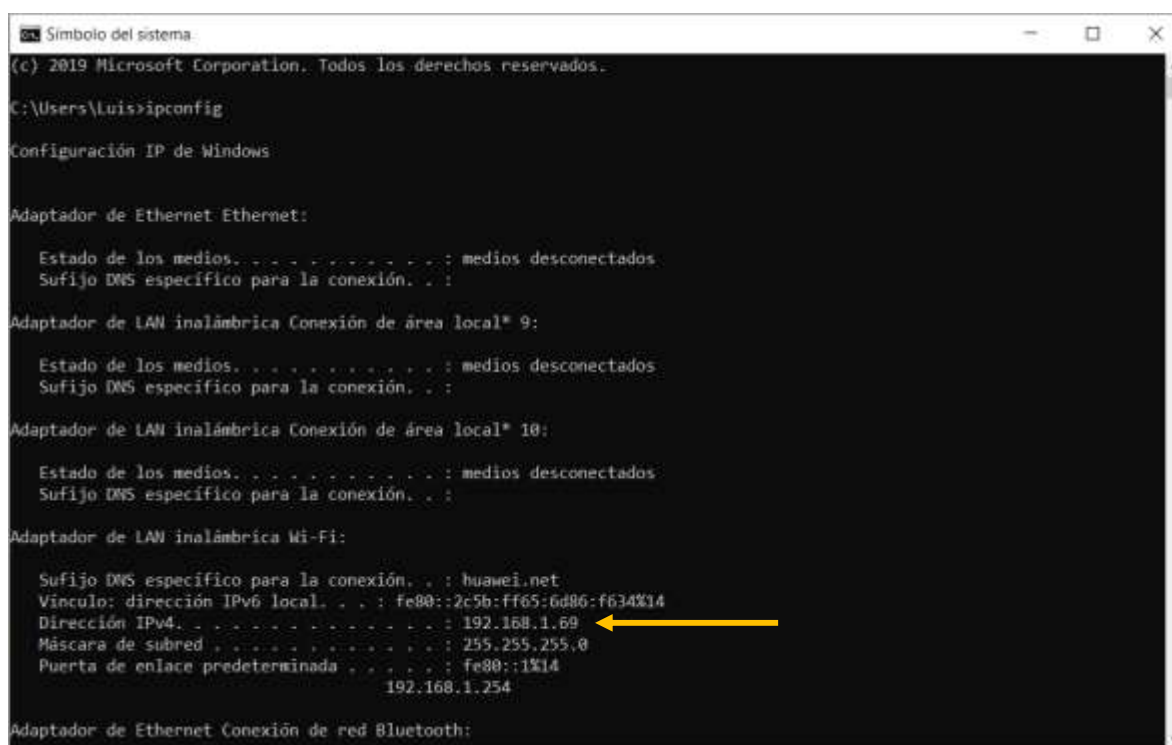
Als erstes müssen wir das Handy und den PC an dasselbe **WiFi-Netzwerk** anschließen, damit sie sich gegenseitig sehen können. Die IPs oder Adressen müssen ähnlich wie 192.168.XXX.XXX sein, die XXX-Werte sind variable Zahlen, die in jedem Computer zufällig zugewiesen werden.

Dieses Beispiel wurde auf einem LG Q6-Mobiltelefon und einem PC mit Windows 10 Home getestet.

Überprüfen Sie die IP oder die Adresse, die der PC mit dem WiFi verbunden hat, wir müssen ein Terminal in Windows öffnen.

Schreiben Sie in das untere Feld, in dem die Suchlupe steht, cmd und drücken Sie die Eingabetaste. Es öffnet sich ein Terminal, in das wir den Befehl schreiben:

C:Benutzername> 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:ff65:6d06: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:
```

Es wird uns zeigen, dass die dem PC darin zugewiesene IP 192.168.1.69 ist, aber dies ist höchstwahrscheinlich von Fall zu Fall unterschiedlich.

ANMERKUNG: Die Adresse, bei der "IPv4-Adresse" steht, ist zu nehmen, nicht zu verwechseln mit dem Gateway.

Nun müssen wir im Falle des Mobiltelefons im Termux-Terminal den folgenden Befehl eingeben, um den Namen unseres Benutzers zu erfahren, mit dem wir uns mit dem SSH-Server verbinden werden, der unser Telefon hat, führen wir folgenden Befehl aus:

\$ whoami

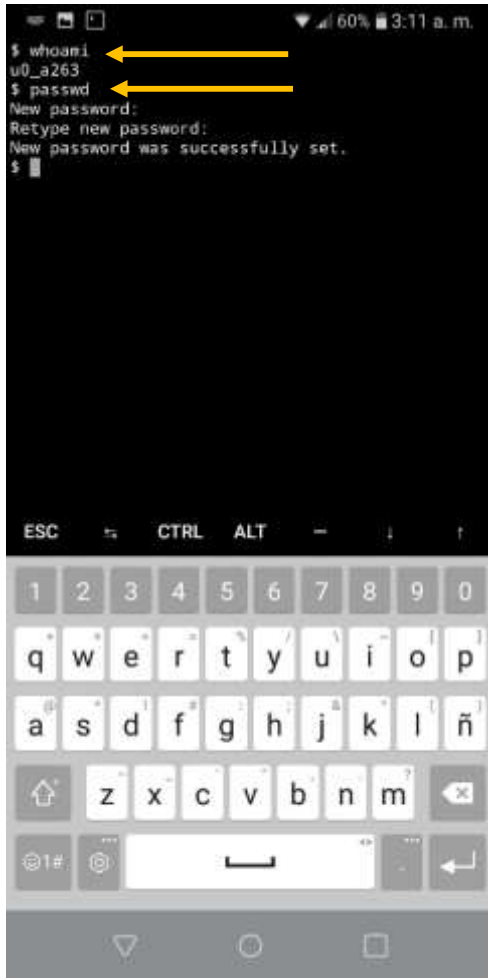
Später müssen wir diesem Benutzer ein Passwort geben, so dass wir den folgenden Befehl ausführen müssen:

\$ passwd

Es fordert uns auf, ein Passwort einzugeben und die Eingabetaste zu drücken, wieder fragt es uns nach dem Passwort, wir bestätigen es und drücken die Eingabetaste, wenn es **erfolgreich war "Neues Passwort wurde erfolgreich gesetzt"** im Falle der Markierung ist es möglich, dass das Passwort nicht korrekt eingegeben wurde. Führen Sie das Verfahren erneut durch.

Und dann, um zu wissen, welche IP wir in Termux haben, geben wir den folgenden Befehl ein, die IP steht nach dem Wort **"inet"**:

\$ ifconfig -a



Jetzt

ist es an der Zeit, den SSH-Server-Dienst auf Ihrem Telefon zu starten, damit Sie Sitzungen von Ihrem PC empfangen können. Wir führen den folgenden Befehl im Termux-Terminal aus, dieser Befehl liefert kein Ergebnis.

\$ sshd



Jetzt müssen wir auf dem PC ein Programm installieren, das vom PC aus mit dem SSH-Server des Telefons kommuniziert.

Wir müssen zu <https://www.putty.org> gehen.

Wählen Sie, wo der Link "Sie können PuTTY hier herunterladen" ist



The screenshot shows the PuTTY website homepage. At the top is a navigation bar with a search icon and a language dropdown menu. Below this is a large banner with the text "Download PuTTY" and a description: "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." A yellow arrow points to the link "here" in the text "You can download PuTTY here." Below the banner is a section titled "Below suggestions are independent of the authors of PuTTY. They are not to be seen a..."



Bitvise SSH Client

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](#).

Wählen Sie die 32-Bit-Version, es spielt keine Rolle, ob Ihr System mit 64-Bit funktioniert.

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, see the [development snapshots](#).

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')

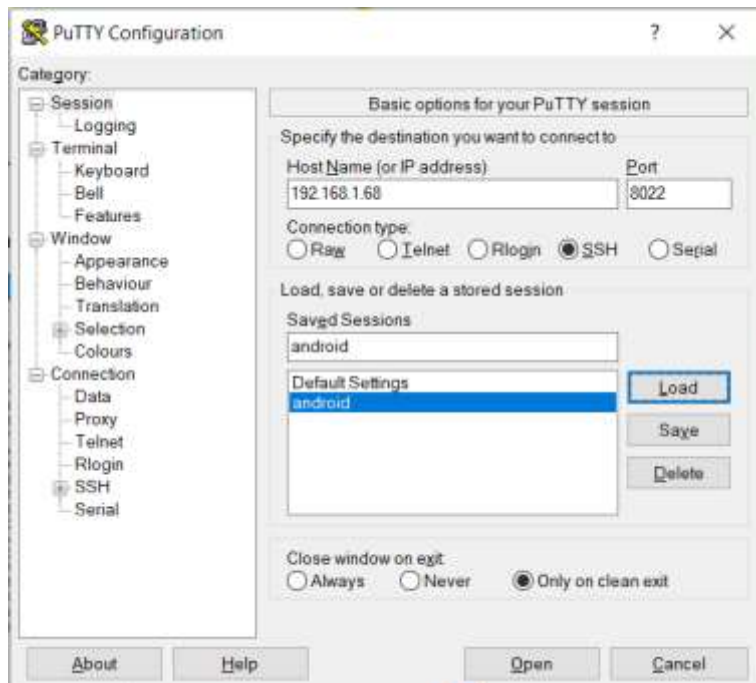
32-bit:	putty-0.73-installer.msi	(or by FTP)	(signature)
64-bit:	putty-64bit-0.73-installer.msi	(or by FTP)	(signature)

Unix source archive

.tar.gz:	putty-0.73.tar.gz	(or by FTP)	(signature)
----------	-----------------------------------	-------------	-------------

Sobald es auf Ihren PC heruntergeladen wurde, führen Sie es aus und installieren es mit den Standardoptionen. Starten Sie dann die PuTTY-Anwendung.

In dieser Sitzung geben wir die Daten von unserem Openssh-Server ein, den wir auf dem Mobiltelefon installiert haben.



Geben Sie die IP des Mobiltelefons ein.

HostName oder IP-Adresse:

192.168.1.68 (IP-Beispiel)

Hafen:

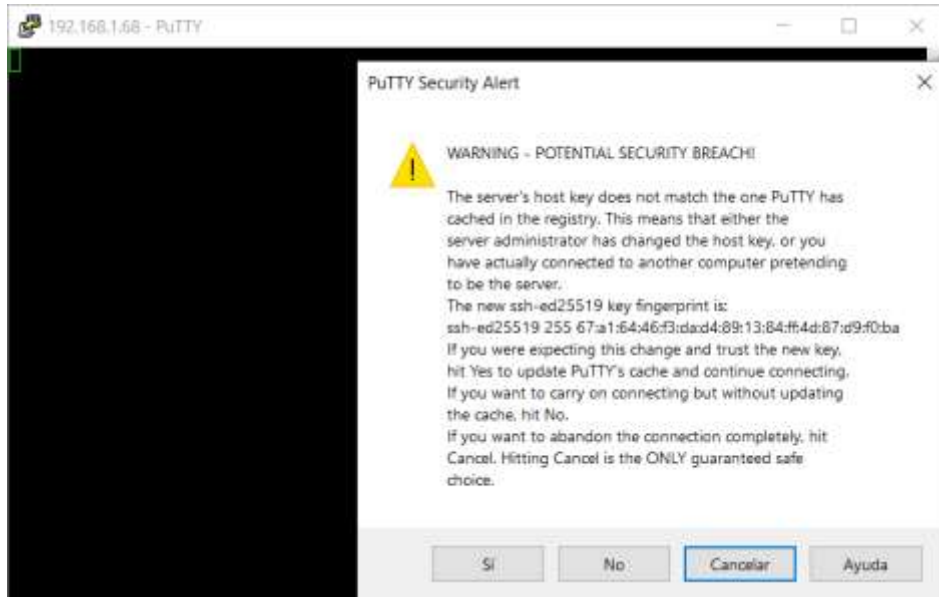
8022 (Standard-Port des mobilen SSH-Servers).

Wir können der Sitzung unter "Gespeicherte Sitzungen" einen Namen geben und auf

die Schaltfläche Speichern klicken.

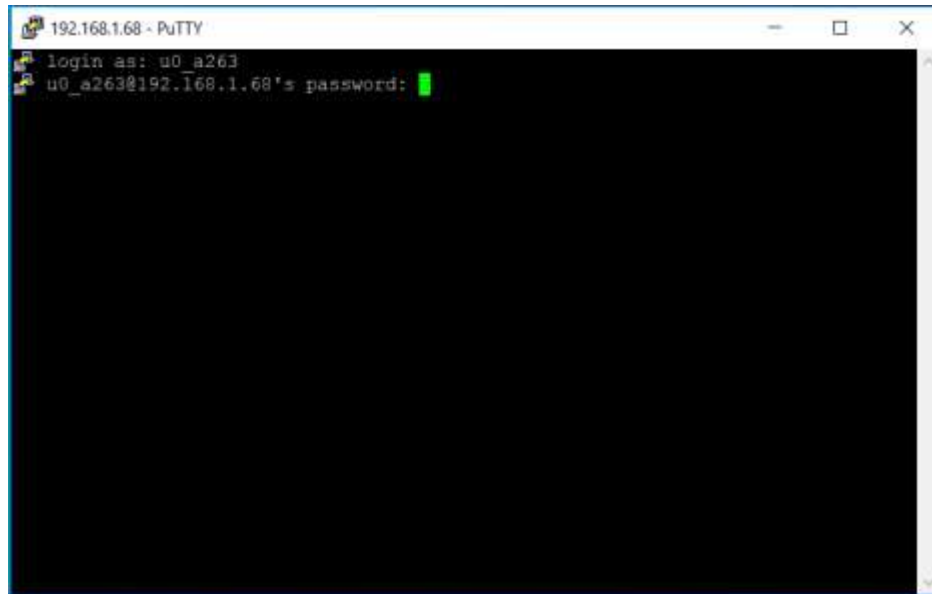
Später im unteren Teil drücken wir auf , um eine Verbindung zum Server zu öffnen, indem wir den Knopf "Öffnen" drücken.

Auf dem PC, wenn Sie zum ersten Mal eine Verbindung herstellen, werden Sie zur Bestätigung des Informationsverschlüsselungsschlüssels **aufgefordert**, indem Sie auf die Schaltfläche "Ja" klicken.

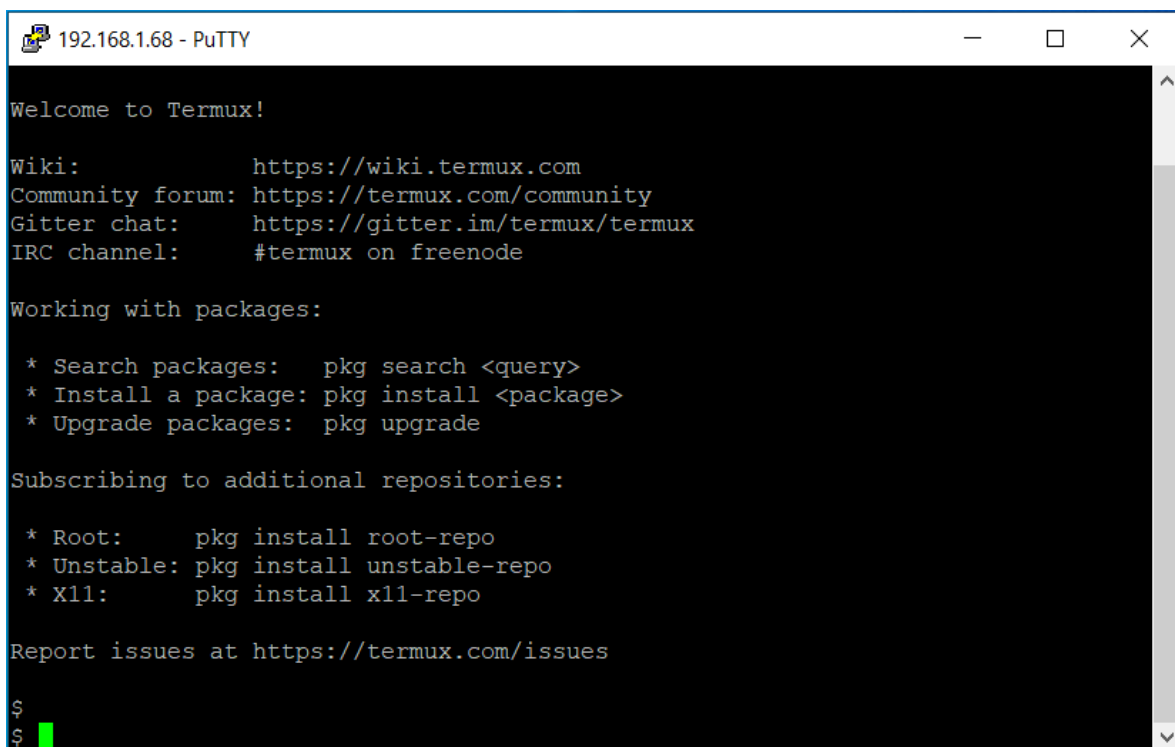


Später werden wir nach dem Benutzer gefragt, mit dem wir uns verbinden wollen. Wir werden die Informationen verwenden, die wir zuvor erhalten haben (Benutzer und Passwort).

Im **Login als:** müssen wir unseren Benutzer eingeben und Enter geben, dann werden wir nach dem Passwort fragen und wieder Enter geben.



Wenn die Daten korrekt waren, befinden wir uns in einer SSH-Sitzung (Secure Shell), die vom PC (Client) aus am Telefon (SSH-Server) durchgeführt wird.



WICHTIGER HINWEIS: Denken Sie daran, dass sich die IP (Adresse) des PCs und die IP (Adresse) des Mobiltelefons, das an dasselbe WiFi angeschlossen ist, wahrscheinlich jedes

Mal ändern werden, wenn wir die Verbindung trennen und wieder herstellen, so dass wir doppelt prüfen müssen, welche Adressen jedes Gerät hat, dies wird den Erfolg der Verbindung zwischen den Geräten über den SSH-Server des Telefons und den PC (Client) sicherstellen.

8. Ambientes Blockly (App Inventor, AppyBuilder und Thunkable).

App Inventor ist eine Software-Entwicklungsumgebung, die von Google Labs erstellt wurde, um Anwendungen für das Android-Betriebssystem zu erstellen. Der Benutzer kann visuell und mit Hilfe eines Satzes von Basiswerkzeugen eine Reihe von Blöcken miteinander verknüpfen, um die Anwendung zu erstellen. Das System ist kostenlos und kann einfach aus dem Internet heruntergeladen werden. Mit App Inventor erstellte Anwendungen sind sehr einfach zu erstellen, da keine Kenntnisse einer Programmiersprache erforderlich sind.

Alle aktuellen Umgebungen, die die Technologie von Blockly nutzen, wie z.B. AppyBuilder und Thunkable, haben ihre kostenlose Version, ihre Nutzung kann über das Internet in ihren verschiedenen Sites erfolgen oder auch zu Hause installiert werden.

Die Blöcke, aus denen sich die Mini BloclChain-Architektur zusammensetzt, wurden in App inventor und AppyBuilder getestet, aber aufgrund ihrer Code-Optimierung sollten sie auf den anderen Plattformen funktionieren.

Online-Versionen:

App-Erfinder.

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

AppyBuilder.

<http://appybuilder.com/>

Denkbar.

<https://thunkable.com/>

Version, die auf Ihrem Computer (PC) installiert werden soll:

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

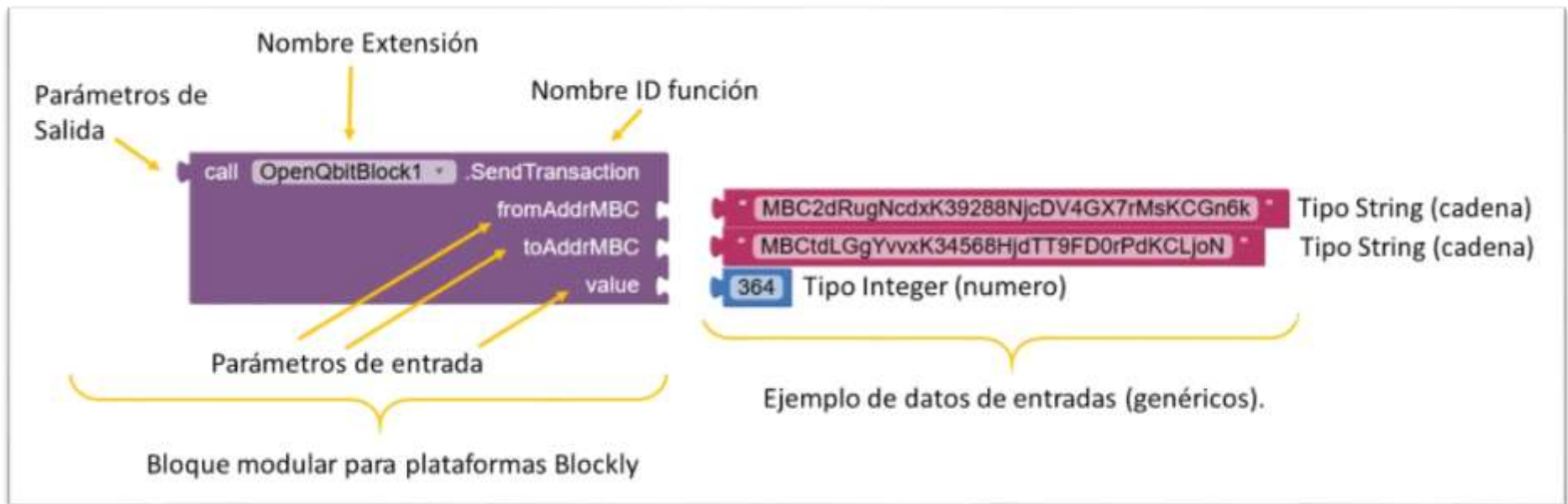
Umgebung für Entwickler von Blockly-Blöcken.

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

9. Definition und Verwendung von Blöcken in Mini QRNG.

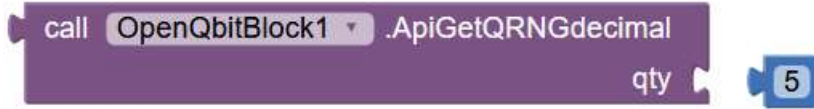
Wir beginnen damit, die Verteilung der Daten, die alle Blöcke haben werden, ihre Syntax der Verwendung und Konfiguration zu erklären.

Im folgenden Beispiel sehen wir einen modularen Block und seine Ein- und Ausgabeparameter sowie die Arten von Eingabedaten, diese Daten können vom Typ String (Zeichenfolge) oder Integer (ganzzahlig oder dezimal) sein. Wir zeigen, wie sie benutzt wird, und konfigurieren sie für ihr einwandfreies Funktionieren.



Jeder Modulblock hat seine eigene Beschreibung und wird benannt, falls er eine obligatorische oder optionale Abhängigkeit(en) von anderen Blöcken als Eingabeparameter hat, wird der Integrationsprozess angekündigt. Beginnen wir mit den Blöcken des **OpenQbitQRNG** mit der **ErweiterungSSH**.

Block zur Erzeugung von dezimalen Zufallsquantenzahlen - (**ApiGetQRNGdecimal**)



Eingabeparameter: **Menge** <Ganzzahl>

Ausgabeparameter: Gibt die Menge "qty" von zufälligen Quanten-Dezimalzahlen an, die in die Eingabezahlen eingegeben werden, die im Bereich von 0 und 1 im JSON-Format liegen.

Beispiel:

Anzahl = 5; Ausgabe: {"Ergebnis": [0,5843012986202495, 0,7746497687824652, 0,05951126805960929, 0,1986079079055812694, 0,0368978343989999279]}

Beschreibung: Quantum Random Number Generator (QRNG) API

Block zur Erzeugung von dezimalen Zufallsquantenzahlen - (**ApiGetQRNGdecimal**)



Eingabeparameter: **Menge** <Ganzzahl>, min <Ganzzahl>, max <Max>

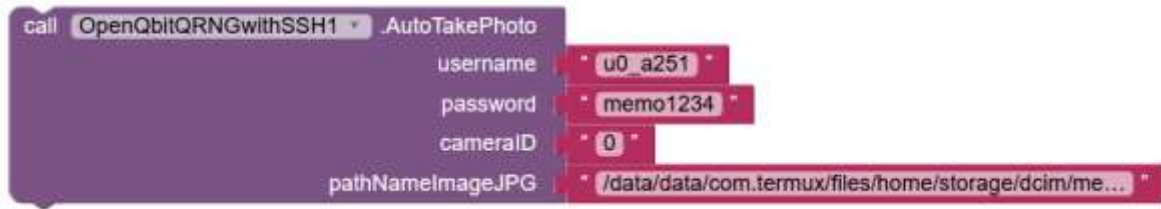
Ausgabeparameter: Gibt die Menge "qty" zufälliger Quantenganzzahlen an, die in die Eingabe eingegeben wurden, wobei die Zahlen im Bereich von min und max im JSON-Format liegen.

Beispiel:

Anzahl = 8, min = 1, max = 100; Ausgabe: {"Ergebnis": [3, 53, 11, 2, 66, 44, 9, 78]}

Beschreibung: Quantum Random Number Generator (QRNG) API

Blockieren, um automatisch ein Bild aufzunehmen - (AutoTakePhone)



Eingabeparameter: **Benutzername** < Zeichenfolge>, **Passwort** < Zeichenfolge>, **KameraID** < Zeichenfolge>, **PfadNameImageJPG** < Zeichenfolge>.

Obligatorische Abhängigkeit: Um diesen Block zu verwenden, müssen Sie zwei Software-Abhängigkeiten erfüllen; installieren Sie im Termux-Terminal das Termux-API-Modul. Dieses Modul enthält den Prozess zur automatischen Aufnahme von Bildern und zum Hochladen des zuvor installierten SSH-Servers.

Ausgabeparameter: Es liefert ein Foto (Bild) im JPG-Format im angegebenen Pfad. Auf dem Weg müssen Sie

Beschreibung: Erstellt automatisch ein JPG-Foto ohne Benutzereingriff.

Um die Termux-API zu installieren, muss der folgende Befehl im Termux-Terminal ausgeführt werden:

\$ pkg termux-api installieren

```

$ 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) ...
$

```

Um die Anzahl, die Anzahl und die Position der IDs (Fotoobjektiv-Kennungen) Ihres mobilen Geräts (Smartphone) herauszufinden, führen Sie den folgenden Befehl im Termux-Terminal aus.

\$ termux-camera-info



The screenshot shows a Termux terminal window on an Android phone. The status bar at the top shows 80% battery and the time 11:35 p.m. The terminal displays the command `$ termux-camera-info` and its output, which is a JSON array. The first element of the array is for ID "0", which is facing "back". It lists four JPEG output sizes: {width: 4160, height: 3120}, {width: 4160, height: 2340}, {width: 4160, height: 2080}, and {width: 3264, height: 2448}. Yellow arrows point to the `"id": "0"`, `"facing": "back"`, and the first size object. Below the terminal is a virtual keyboard with a home button, a search button, and a back button.

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



The screenshot shows a Termux terminal window on an Android phone. The status bar at the top shows 80% battery and the time 11:36 p.m. The terminal displays the command `$ termux-camera-info` and its output, which is a JSON array. The first element of the array is for ID "1", which is facing "front". It lists four JPEG output sizes: {width: 2560, height: 1920}, {width: 2560, height: 1600}, and {width: 2560, height: 1600}. Yellow arrows point to the `"id": "1"` and `"facing": "front"`. Below the terminal is a virtual keyboard with a home button, a search button, and a back button.

```
$ termux-camera-info
[
  {
    "id": "1",
    "facing": "front",
    "jpeg_output_sizes": [
      {
        "width": 2560,
        "height": 1920
      },
      {
        "width": 2560,
        "height": 1600
      },
      {
        "width": 2560,
        "height": 1600
      }
    ]
  }
]
```

In unserem Beispiel hat der von uns verwendete LG Q6 Smartphone zwei IDs "0" auf der Rückseite und "1" auf der Vorderseite.

Testen wir nun die API, um ein Bild mit der ID "0" der hinteren Linse aufzunehmen und ihr in unserem Fall test.jpg einen albitären Namen zu geben

Denken Sie daran, dass die API nur Fotos im JPG-Format liefert:

\$ termux-Kamera-Foto -c 0 test.jpg

Der vorhergehende Befehl muss automatisch eine Datei mit dem Namen test.jpg erstellt haben, wenn ja, können wir den Block (AutoTakePhoto) verwenden, vergessen Sie nicht, unseren lokalen SSH-Server mit dem Befehl: **\$ sshd** zu starten

HINWEIS: In der Variable pathNameImageJPG sollte berücksichtigt werden, dass der Pfad innerhalb des Termux-Terminals für den Zugriff auf den Smartphone-Speicher verwendet werden sollte:

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

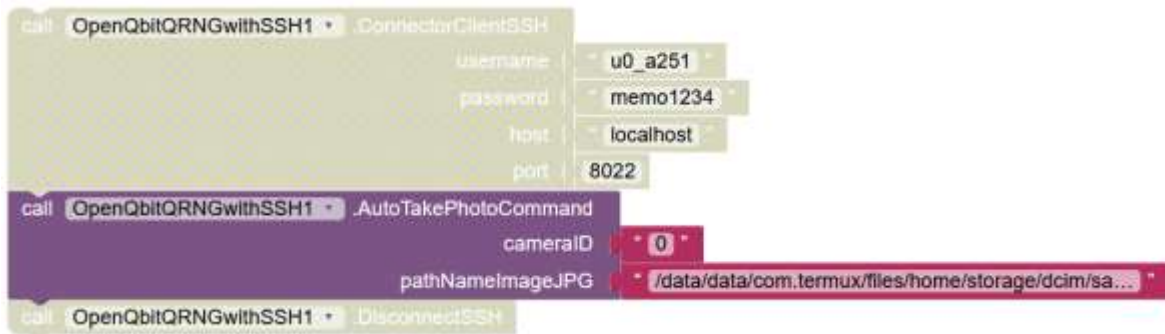
Die bisherige Route auf einem Androiden wäre die gleiche wie die vorherige:

/mnt/sdcard/dcim/beispiel.jpg

Wir dürfen jedoch nicht vergessen, dass im Termux-Terminal immer die gültige Route zum Anzeigen des Einschaltens des Mobiltelefons als Standardroute betrachtet werden muss:

/data/data/com.termux/files/home/storage

Blockieren, um ein Foto automatisch aufzunehmen NUR BEFEHL - (AutoTakePhoneCommand)



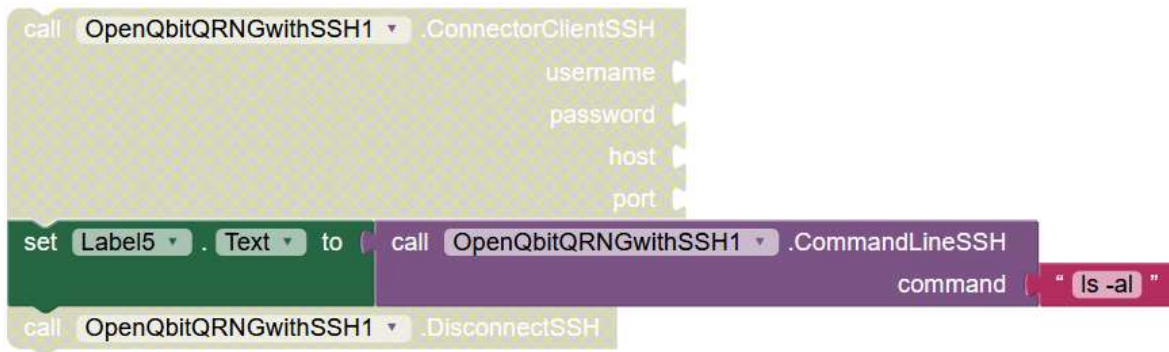
Eingabeparameter: **KameraID** <Zeichenkette>, **PfadNameImageJPG** <Zeichenkette>.

Obligatorische Abhängigkeit: Block (**ConnectorClientSSH**), Block (**DisconnectSSH**).

Ausgabeparameter: Es liefert ein Foto (Bild) im JPG-Format im angegebenen Pfad.

Beschreibung: Erstellt automatisch ein JPG-Foto ohne Benutzereingriff. Der Unterschied zum Block (AutoTakePhoto) besteht jedoch darin, dass dieser Block nur den Befehl zum Erstellen des Fotos enthält und Sie den Block zunächst benötigen, um eine Verbindung mit dem SSH-Server herzustellen (**ConnectClientSSH**) und dann den Block zu verwenden (**DisconnectSSH**).

Block zur Befehlsausführung im Termux-Terminal - (**CommandLineSSH**)



Eingabeparameter: **Befehl** <Zeichenkette>

Obligatorische Abhängigkeit: Block (**ConnectorClientSSH**), Block (**DisconnectSSH**).

Ausgabeparameter: Führen Sie den im Termux-Terminal eingegebenen Befehl aus.

Beschreibung: Ein eingegebener Befehl wird ausgeführt, und der Block wird zuerst für die Verbindung zum SSH-Server (**ConnectClientSSH**) und dann für die Verwendung des Blocks (**DisconnectSSH**) benötigt.

Blockieren der Verbindung zu einem entfernten oder lokalen SSH-Server - (**ConnectorClientSSH**).



Eingabeparameter: **Benutzername** <string>, **Passwort** <string>, **Host** <string>, **Port**<integer>

Ausgabeparameter: Wenn die Verbindung mit dem ssh-Server des Termux-Terminals erfolgreich ist, gibt er uns eine Meldung; "**Connect SSH**", wenn sie nicht erfolgreich ist, gibt er uns eine **NULL-Meldung**.

Beschreibung: Kommunikationsblock zur Verbindung des gewählten SSH-Servers mit dem Termux-Terminal über das Kommunikationsprotokoll SSH (Secure Shell).

Block zum Dekodieren einer Datei mit Base64-Algorithmus (**DecoderFileBase64**).

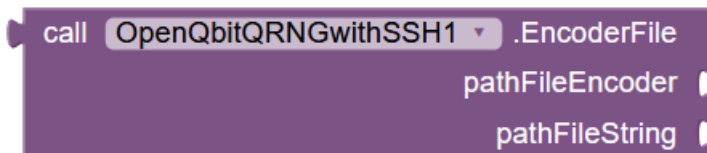


Eingabeparameter: **pathFileBase64** <Zeichenkette>, pathFileOrigin <Zeichenkette>

Ausgabeparameter: Quelldatei, die in den Block eingegeben wurde (**EncoderFileBase64**)

Beschreibung: Eine Base64-Datei wird in die Originaldatei konvertiert, die in den Block eingefügt wurde (**EncoderFileBase64**).

Block konvertiert eine Datei in das Base64-Format - (**EncoderFile**)

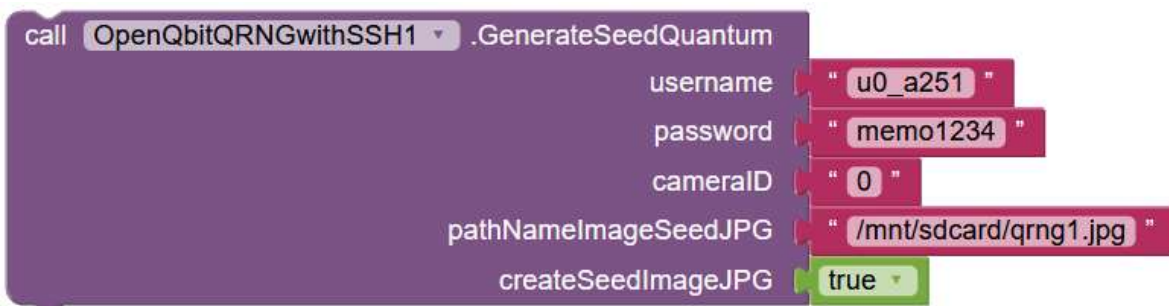


Eingabeparameter: **pathFileOrigin** <String> , pathFileBase64 <String>

Ausgabeparameter: Base64-kodierte Datei.

Beschreibung: Konvertiert eine Quelldatei in einem beliebigen Format in eine Base64-Datei. Die Dateinamen können beliebig sein und vom Benutzer gewählt werden.

Block zur Erzeugung von QRNG (Quantum Random Number Generator) - (**GenerateSeedQuantum**)



Eingabeparameter: **Benutzername** <Zeichenkette>, **Passwort** <Zeichenkette>, **KameraID** <Zeichenkette>, **PfadNameImageJPG** <Zeichenkette>. **createSeedImageJPG** <Boolean>

Wenn der boolesche Wert bei jeder Ausführung des Blocks "True" ist, wird jedes Mal ein neues JPG-Bild mit dem Namen des eingegebenen Pfades erzeugt. Wenn der Boolean-Wert "False" ist, deaktivieren wir die Option, ein JPG-Bild (Foto) aufzunehmen, und wir können

manuell angeben, wo sich ein Bild in unserer Auswahl befindet, das jedes beliebige Format haben kann.

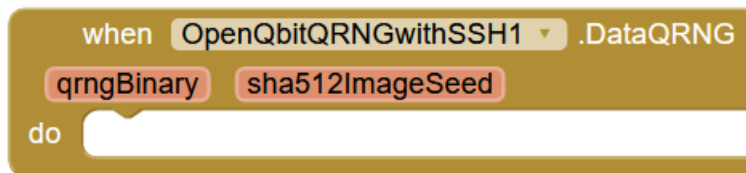
HINWEIS: Die besten Ergebnisse für die Erzeugung eines QRNG basieren auf einem "RAW"-formatierten Bild. Beispiel DNG-Format.

Obligatorische Abhängigkeit: Die oben genannte Termux-API muss im Block installiert sein (**AutoTakePhoto**).

Ausgabeparameter: Das Ereignis wird ausgeführt (**DataQRNG**) und liefert uns zwei Werte:

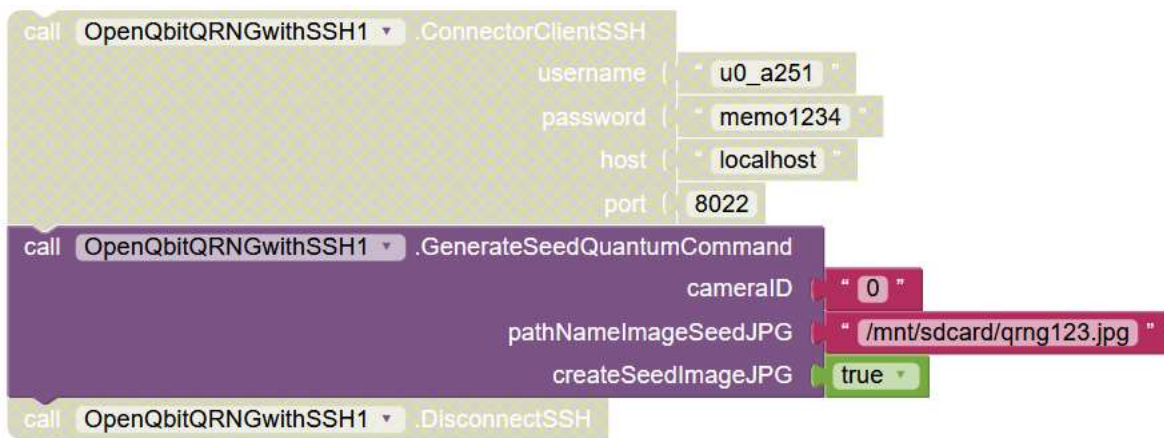
qrngBinary.- Eine Kette von zufälligen Binärzahlen

Sha512ImageSeed - Sha512 aus dem JPG-Seed-Bild, aus dem die Zufallszahlen stammen.



Beschreibung: Es generiert Quantenzufallszahlen (QRNG) durch den optischen Sensor der Kamera des Mobiltelefons. Der Algorithmus basiert auf der Erfassung von Zufallsfotos und der Algorithmus wird darauf angewendet, um eine Kette von Binärzahlen zu liefern.

Block zur Erzeugung von QRNG (Quantum Random Number Generator) - (GenerateSeedQuantumCommand).



Eingabeparameter: KameraID <Zeichenkette>, PfadNameImageJPG <Zeichenkette>. createSeedImageJPG <Blean>

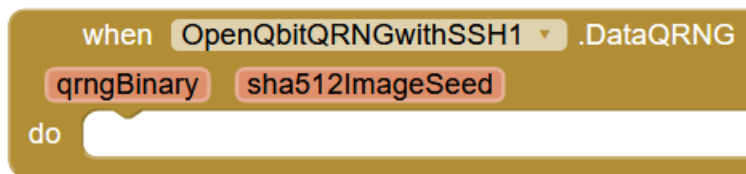
Wenn der boolesche Wert bei jeder Ausführung des Blocks "True" ist, wird jedes Mal ein neues JPG-Bild mit dem Namen des eingegebenen Pfades erzeugt. Wenn der Boolean-Wert "False" ist, deaktivieren wir die Option, ein JPG-Bild (Foto) aufzunehmen, und wir können manuell angeben, wo sich ein Bild in unserer Auswahl befindet, das jedes beliebige Format haben kann.

Obligatorische Abhängigkeit: Block (**ConnectorClientSSH**), Block (**DisconnectSSH**).

Ausgabeparameter: Das Ereignis wird ausgeführt (**DataQRNG**) und liefert uns zwei Werte:

qrngBinary.- Eine Kette von zufälligen Binärzahlen

Sha512ImageSeed - Sha512 aus dem JPG-Seed-Bild, aus dem die Zufallszahlen stammen.



Beschreibung: Es generiert Quantenzufallszahlen (QRNG) durch den optischen Sensor der Kamera des Mobiltelefons. Der Algorithmus basiert auf der Erfassung von Zufallsfotos und der Algorithmus wird darauf angewendet, um eine Kette von Binärzahlen zu liefern.

Der Unterschied im Vergleich zum Block (**GenerateSeedQuantum**) besteht jedoch darin, dass dieser Block nur den Befehl zur Ausführung des QRNG-Reaktionsalgorithmus enthält und Sie den Block zunächst benötigen, um eine Verbindung zum SSH-Server herzustellen (**ConnectClientSSH**) und dann den Block zu verwenden (**DisconnectSSH**).

Block zum Erhalten der Shannon-Entropie in einem Bild (Foto) - (**GetShannonEntropyFile**).



Eingabeparameter: **Benutzername** <Zeichenkette> , **Passwort** <Zeichenkette> , **PfadDateiBild** <Zeichenkette>

Obligatorische Abhängigkeit: Das Modul Shannon_entropy muss auf dem Termux-Terminal installiert werden.

Ausgabeparameter: Liefert die Entropie eines Bildes

Beispiel:

Ausgabe: 8,94596789873

Beschreibung: Sie gibt uns die Entropie eines Bildes an. Die Entropie ist der grundlegende Parameter für die Generierung von Zufallszahlen guter Qualität, je höher die Entropie, desto besser die Ergebnisse.

Um das Shannon-Entropie-Modul zu installieren, müssen wir zuerst das Python-Paket installieren und dann das Pillow- und das Shannon_entropy-Modul mit den folgenden Befehlen im Termux-Terminal installieren.

\$ apt Python installieren

\$ pip installieren Kissen

\$ pip installieren 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)
$
```

Dann müssen wir in Termux's "Home"-Verzeichnis die folgende Python-Datei namens "entropy.py" mit folgendem Code darin erstellen.

```
Aus PIL-Import Bild
```

```
Mathe importieren
```

```
Aus Shannon_Entropie importieren *
```

```
Sys importieren
```

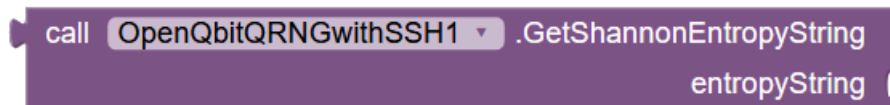


```
Img=Bild.öffnen(sys.argv[1])  
  
drucken(Shannon_Entropie(img))
```

Wir speichern die Datei und haben unsere Umgebung, die wir mit dem Block verwenden können (**GetShannonEntropyFile**).

Tipp: Tatsächlich können Sie mit dieser Python-Installation Ihre eigenen Programme in dieser Sprache erstellen und sie durch den Block laufen lassen (ConnectorClientSSH).

Block zum Erhalten der Shannon-Entropie aus einem String - (**GetShannonEntropyString**).



Eingabeparameter: EntropieString>String>

Ausgabeparameter: Liefert die Entropie einer Zeichenkette.

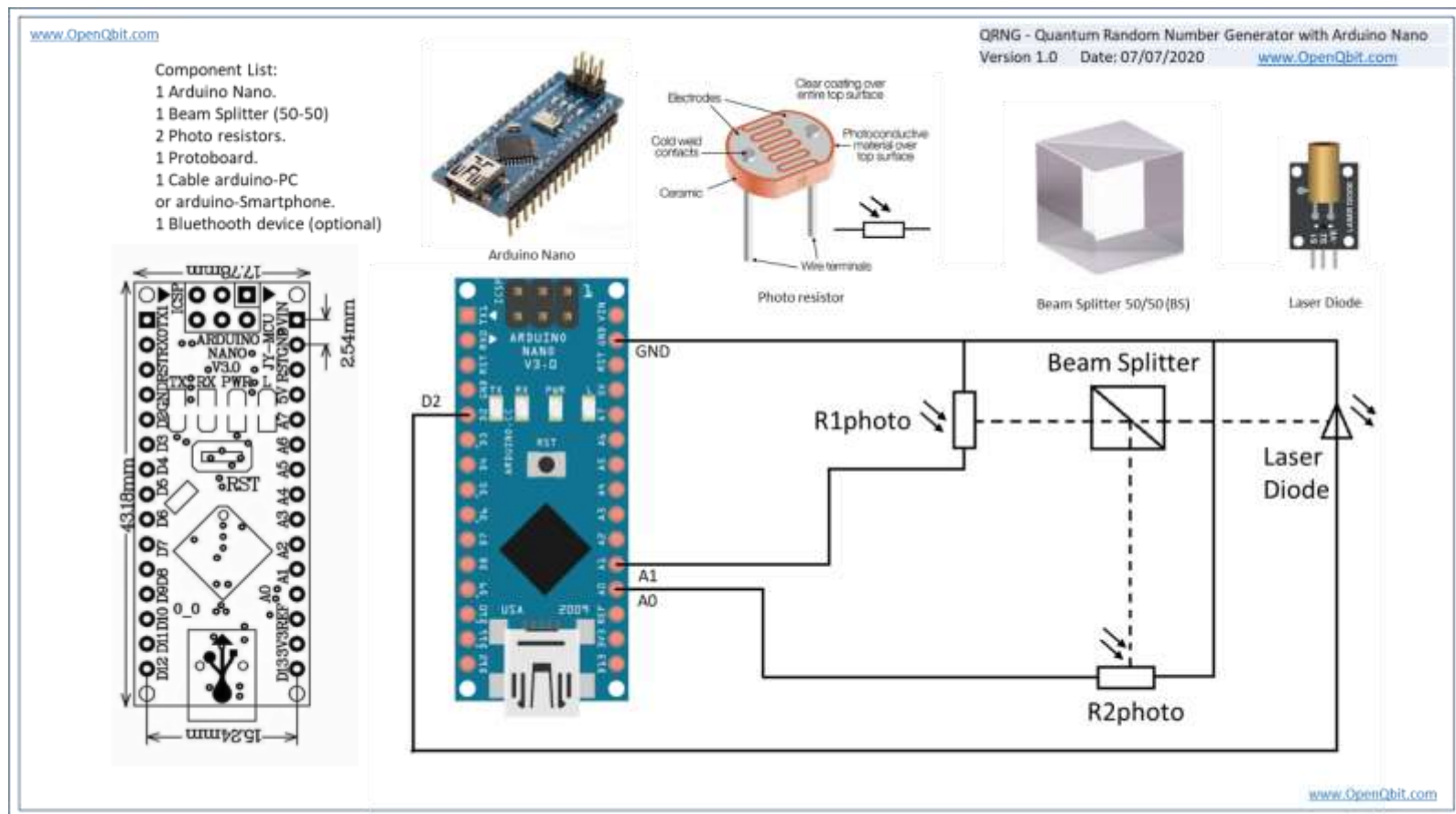
Beispiel:

Ausgabe: 5,76002345671

Beschreibung: Sie gibt uns die Entropie einer Zeichenkette an. Die Entropie ist der grundlegende Parameter für die Generierung von Zufallszahlen guter Qualität, je höher die Entropie, desto besser die Ergebnisse.

10. Erstellung eines "Hardware"-Gerätes eines QRNG.

Wir werden nun ein physisches "Hardware"-Gerät zur Erzeugung von Quanten-Zufallszahlen (QRNG) mit kostengünstigen Komponenten schaffen, die zu Hause leicht zusammengebaut werden können.



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

www.OpenQbit.com

Kompilieren des Programms QRNGv10.ino und Hochladen auf mühsame nano...



```
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
    // program will simply run the function repeatedly until
    // a random bit can be generated.
    return Random();
  }
}

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

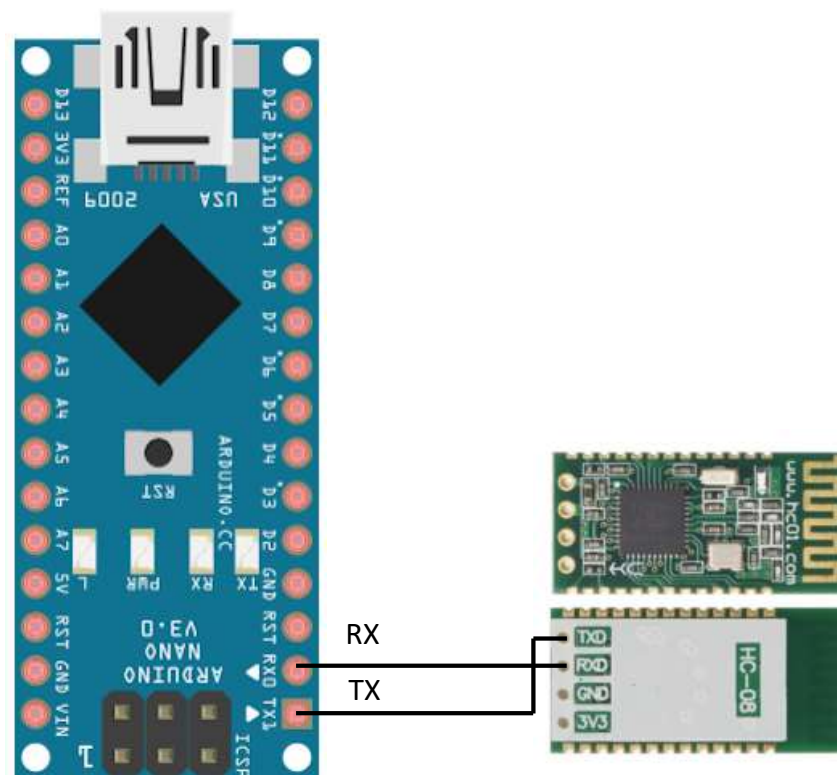
Compilado

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.

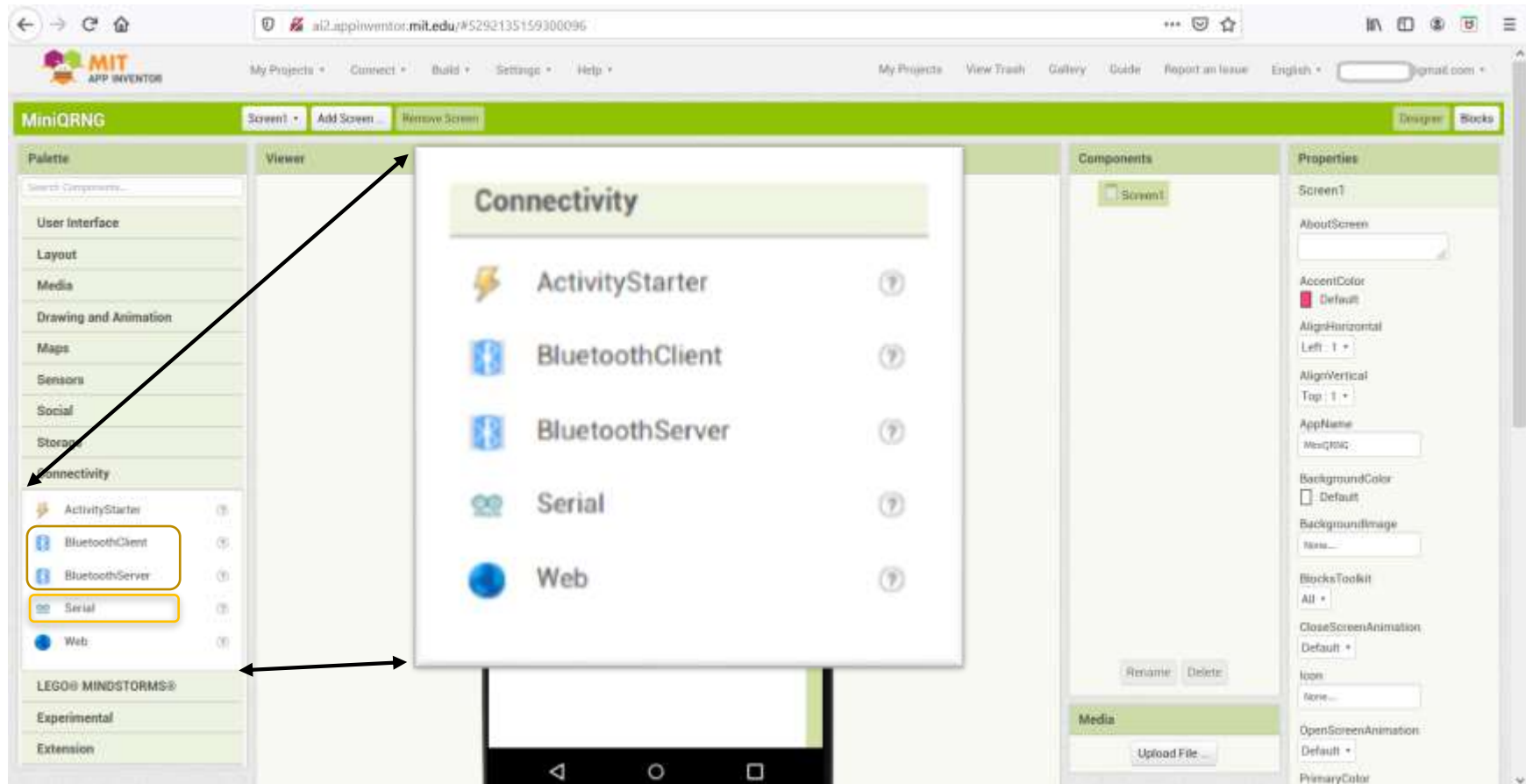
40 ArduinoGetters Ltd en CDM

Es gibt zwei Möglichkeiten, mit dem anstrengenden Nano zu kommunizieren, eine über die serielle Schnittstelle und die andere über eine Bluetooth-Verbindung.

Da die Bluetooth-Verbindung sehr einfach ist, müssen wir nur das HC-08-Modul oder ein ähnliches Modul kaufen und es wie folgt anschließen:



Die folgenden seriellen oder Bluetooth-Komponenten können verwendet werden, um App Inventor mit Arduino zu verbinden:



Jetzt kompiliert und geladen das Programm QRNGv10.ino nur fehlende Kommunikation mit dem mühsamen Nano, um die Daten (Quantenzufallszahlen) zu speichern diese werden im Binärformat sein, jedoch können die erhaltenen Daten leicht in ein anderes Format wie hexadezimal oder dezimal je nach der endgültigen Anforderung übergeben werden.

Um schließlich ein Beispiel zu sehen, wie die serielle oder Bluetooth-Verbindung funktioniert, finden Sie hier einige Referenz-Links.

Denken Sie daran, dass alles durch blockweise Programmierung mit App Inventor getestet werden muss, da dieser bereits Blöcke für die Kommunikation mit arduino seriell oder anderen blockweisen Systemen hat, die zu ähnlichen Bluetooth-Tarves online sein können.

http://kio4.com/erfinder/9A0_bluetooth_RXTX.htm

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

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

11. Anhang "OpenQbit Quantum Computing".

Wie funktioniert das Quantencomputing? ⁽²⁾

Der digitale Wandel bringt schneller als je zuvor Veränderungen in der Welt mit sich. Würden Sie glauben, dass das digitale Zeitalter kurz vor dem Ende steht? **Die digitale Kompetenz** wurde bereits als ein Bereich identifiziert, in dem offenes Wissen und zugängliche Möglichkeiten zum Erlernen von Technologie dringend erforderlich sind, um Lücken in der sozialen und wirtschaftlichen Entwicklung zu schließen. Das Lernen von den Schlüsselkonzepten des digitalen Zeitalters wird mit der bevorstehenden Ankunft einer weiteren neuen technologischen Welle, die in der Lage ist, bestehende Modelle mit erstaunlicher Geschwindigkeit und Kraft zu transformieren, noch kritischer werden: die **Quantentechnologien**.

In diesem Artikel vergleichen wir die grundlegenden Konzepte des traditionellen Rechnens und des Quantencomputings; und wir beginnen auch mit der Untersuchung ihrer Anwendung in anderen verwandten Bereichen.

Was sind Quantentechnologien?

Im Laufe der Geschichte hat der Mensch die Technik entwickelt, da er durch die Wissenschaft verstanden hat, wie die Natur funktioniert. Zwischen 1900 und 1930 führte die Untersuchung einiger physikalischer Phänomene, die noch nicht gut verstanden waren, zu einer neuen physikalischen Theorie, der **Quantenmechanik**. Diese Theorie beschreibt und erklärt die Funktionsweise der mikroskopischen Welt, dem natürlichen Lebensraum von Molekülen, Atomen oder Elektronen. Dank dieser Theorie war es nicht nur möglich, diese Phänomene zu erklären, sondern auch zu verstehen, dass die subatomare Realität auf eine völlig kontraintuitive, fast magische Weise funktioniert und dass in der mikroskopischen Welt Ereignisse stattfinden, die in der makroskopischen Welt nicht vorkommen.

Zu diesen **Quanteneigenschaften** gehören Quantenüberlagerung, Quantenverschränkung und Quantenteleportation.

- **Die Quantenüberlagerung** beschreibt, wie sich ein Teilchen gleichzeitig in verschiedenen Zuständen befinden kann.
- **Die Quantenverschränkung** beschreibt, wie zwei beliebig weit voneinander entfernte Teilchen so korreliert werden können, dass bei der Wechselwirkung mit dem einen das andere sich dessen bewusst wird.
- **Die Quantenteleportation** nutzt die Quantenverschränkung, um Informationen von einem Ort im Raum zu einem anderen zu senden, ohne ihn durchqueren zu müssen.

Quantentechnologien basieren auf diesen Quanteneigenschaften der subatomaren Natur.

In diesem Fall erlaubt uns heute das Verständnis der mikroskopischen Welt durch die Quantenmechanik, Technologien zu erfinden und zu entwerfen, die das Leben der Menschen verbessern können. Es gibt viele und sehr unterschiedliche Technologien, die sich Quantenphänomene zunutze machen, und einige von ihnen, wie z.B. Laser oder Magnetresonanztomographie (MRI), begleiten uns seit mehr als einem halben Jahrhundert. Derzeit erleben wir jedoch eine technologische Revolution in Bereichen wie Quantencomputer, Quanteninformation, Quantensimulation, Quantenoptik, Quantenmetrologie, Quantenuhren oder Quantensensoren.

Was ist Quanteninformatik? Zuerst müssen Sie das klassische Rechnen verstehen.




FIGURA 1.
Ejemplos de caracteres en lenguaje binario.

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

Um zu verstehen, wie Quantencomputer funktionieren, ist es sinnvoll, zunächst zu erklären, wie die Computer funktionieren, die wir täglich benutzen und die wir in diesem Dokument als digitale oder klassische Computer bezeichnen. Diese, wie auch die übrigen elektronischen Geräte wie Tablets oder Mobiltelefone, verwenden Bits als die grundlegenden Speichereinheiten. Dies bedeutet, dass Programme und Anwendungen in Bits kodiert sind, d.h. in binärer Sprache mit Nullen und Einsen. Jedes Mal, wenn wir mit einem dieser Geräte interagieren, zum Beispiel durch Drücken einer Taste auf der Tastatur, werden im Computer Zeichenketten aus Nullen und Einsen erzeugt, zerstört und/oder verändert.

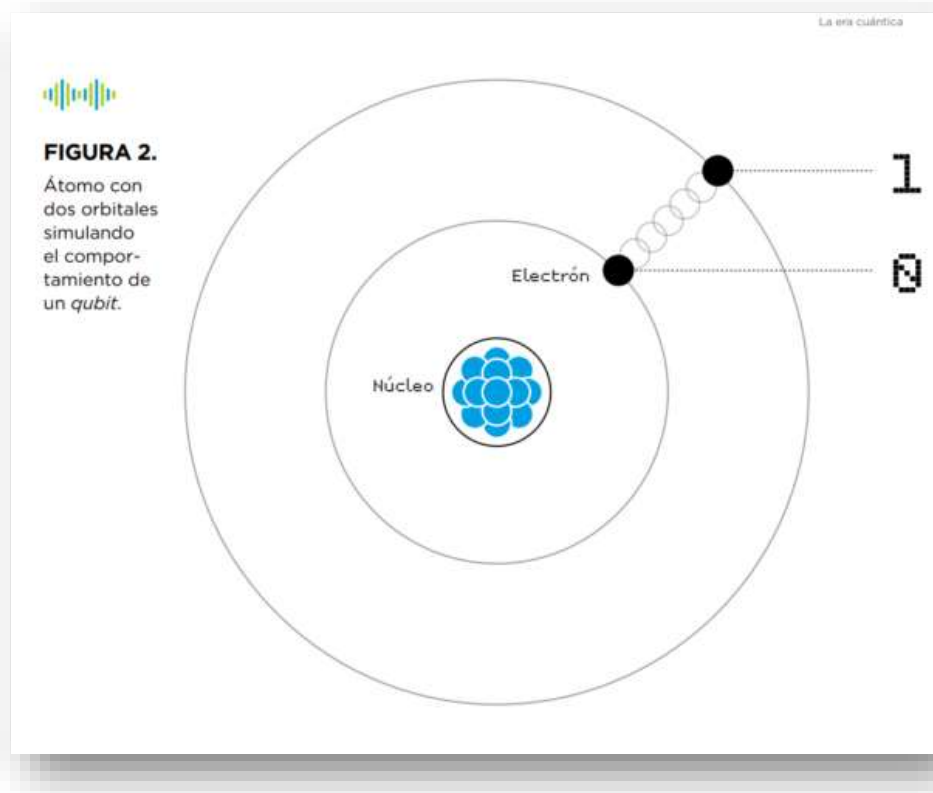
Die interessante Frage ist, was diese Nullen und Einsen physisch im Computer sind. Der Null- und der Eins-Zustand entsprechen elektrischem Strom, der durch mikroskopisch kleine Teile, die als Transistoren bezeichnet werden und als Schalter fungieren, zirkuliert oder nicht. Wenn kein Strom fließt, ist der Transistor "aus" und entspricht Bit 0, und wenn er fließt, ist er "an" und entspricht Bit 1.

Einfacher gesagt, es ist so, als ob die Bits 0 und 1 Löchern entsprechen, so dass ein leeres Loch ein Bit 0 und ein von einem Elektron besetztes Loch ein Bit 1 ist. Deshalb werden diese Geräte als Elektronik bezeichnet. Als Beispiel zeigt Abbildung 1 die binäre Schrift einiger

Zeichen. Da wir nun eine Vorstellung davon haben, wie die heutigen Computer funktionieren, wollen wir versuchen zu verstehen, wie die Quanten funktionieren.

Von Bits zu Qubits

Die grundlegende Informationseinheit im Quantencomputing ist das Quantenbit oder Qubit. Qubits sind per Definition zweistufige Quantensysteme - wir werden hier Beispiele sehen -, die sich wie Bits auf dem niedrigen Niveau befinden können, was einem Zustand niedriger Anregung oder Energie entspricht, der als 0 definiert ist, oder auf dem hohen Niveau, was einem Zustand höherer Anregung entspricht oder als 1 definiert ist. Allerdings, und hier liegt der grundlegende Unterschied zum klassischen Rechnen, können Qubits auch in jedem der unendlichen Zwischenzustände zwischen 0 und 1 liegen, wie z.B. in einem Zustand, der halb 0 und halb 1 oder drei Viertel von 0 und ein Viertel von 1 ist.



Quantenalgorithmen, exponentiell leistungsfähigeres und effizienteres Rechnen

Der Zweck von Quantencomputern ist es, diese Quanteneigenschaften der *Qubits* als Quantensysteme, die sie sind, zu nutzen, um Quantenalgorithmen auszuführen, die Überlappung und Verschachtelung verwenden, um eine viel höhere Rechenleistung als die Klassiker zu erreichen. Es ist wichtig, darauf hinzuweisen, dass der wirkliche Paradigmenwechsel nicht darin besteht, das Gleiche zu tun wie digitale oder klassische Computer - die aktuellen -, sondern schneller, wie in vielen Artikeln zu lesen ist, sondern dass Quantenalgorithmen es erlauben, bestimmte Operationen auf eine völlig andere Art und

[OpenQubit.com](https://openqubit.com) Página 41 | 47

Weise durchzuführen, die sich in vielen Fällen als effizienter erweist - d.h. in viel weniger Zeit oder mit viel weniger Rechenressourcen -.

Schauen wir uns ein konkretes Beispiel dafür an, was dies beinhaltet. Stellen wir uns vor, wir sind in Bogotá, und wir wollen die beste Route nach Lima aus einer Million Möglichkeiten kennen, um dorthin zu gelangen ($N=1.000.000$). Um mit Hilfe von Computern den optimalen Weg zu finden, müssen wir 1.000.000 Optionen digitalisieren, was bedeutet, sie für den klassischen Computer in Bitsprache und für den Quantencomputer in *Qubits* zu übersetzen. Während ein klassischer Computer einen nach dem anderen alle Pfade analysieren müsste, bis er den gewünschten gefunden hat, nutzt ein Quantencomputer den als Quantenparallelität bekannten Prozess, der es ihm erlaubt, alle Pfade auf einmal zu berücksichtigen. Dies bedeutet, dass, während der klassische Computer die Reihenfolge von $N/2$ Schritten oder Iterationen benötigt, d.h. 500.000 Versuche, der Quantencomputer den optimalen Pfad nach nur \sqrt{N} Operationen auf dem Register findet, d.h. nach 1.000 Versuchen.

Im vorherigen Fall ist der Vorteil quadratisch, in anderen Fällen ist er sogar exponentiell, was bedeutet, dass wir mit n *Qubits* eine Rechenkapazität von 2^n Bits erreichen können. Um dies zu veranschaulichen, ist es üblich zu zählen, dass wir mit etwa 270 Qubits in einem Quantencomputer mehr Grundzustände - mehr unterschiedliche und gleichzeitige Zeichenketten - haben könnten als die Anzahl der Atome im Universum, die auf etwa 280 geschätzt wird. Ein weiteres Beispiel ist, dass man schätzt, dass wir mit einem Quantencomputer mit 2000 bis 2500 *Qubits* praktisch die gesamte heute verwendete Kryptographie (die sogenannte Public-Key-Kryptographie) brechen könnten.

Warum ist es wichtig, etwas über Quantentechnologie zu wissen?

Wir befinden uns in einem Moment der digitalen Transformation, in dem verschiedene aufkommende Technologien wie Blockchain, künstliche Intelligenz, Drohnen, Internet der Dinge, virtuelle Realität, 5G, 3D-Drucker, Roboter oder autonome Fahrzeuge immer mehr Präsenz in verschiedenen Bereichen und Sektoren haben. Diese Technologien, die dazu berufen sind, die Lebensqualität des Menschen zu verbessern, die Entwicklung zu beschleunigen und soziale Auswirkungen zu erzeugen, schreiten heute parallel voran. Nur selten sehen wir Unternehmen, die Produkte entwickeln, die Kombinationen aus zwei oder mehr dieser Technologien nutzen, wie z.B. Blockchain und IoT oder Drohnen und künstliche Intelligenz. Obwohl sie dazu bestimmt sind, zu konvergieren und damit eine exponentiell größere Wirkung zu erzielen, ist die Konvergenz aufgrund des Anfangsstadiums der Entwicklung, in dem sie sich befinden, und des Mangels an Entwicklern und Personen mit technischem Profil noch immer eine offene Aufgabe.

Aufgrund ihres disruptiven Potenzials wird erwartet, dass die Quantentechnologien nicht nur mit all diesen neuen Technologien konvergieren, sondern einen Querschnittseinfluss auf praktisch alle von ihnen haben werden. Die Quanteninformatik wird die Authentifizierung,

den Austausch und die sichere Speicherung von Daten bedrohen, was einen großen Einfluss auf jene Technologien hat, bei denen die Kryptographie eine wichtigere Rolle spielt, wie z.B. Cybersicherheit oder Blockchain, und einen geringen negativen Einfluss hat, aber auch in Technologien wie 5G, IoT oder Drohnen in Betracht gezogen werden muss.

Wollen Sie sich im Quantencomputing üben?

Dutzende von Quantencomputer-Simulatoren sind bereits im Netz verfügbar, wobei verschiedene Programmiersprachen wie C, C++, Java, Matlab, Maxima, Python oder Octave bereits im Einsatz sind. Auch neue Sprachen wie Q#, die von Microsoft eingeführt wurden. Sie können eine virtuelle Quantenmaschine über Plattformen wie IBM und Rigetti erforschen und mit ihr spielen.

Mini QRNG wurde von der Firma OpenQbit.com ins Leben gerufen, die sich auf die Entwicklung von auf Quantencomputing basierender Technologie für verschiedene Arten von privaten und öffentlichen Sektoren konzentriert.

Warum Mini QRNG sich von anderen QRNGs unterscheidet, einfach deshalb, weil das System als modulares System konzipiert wurde und zu relativ geringen Kosten einfach zu Hause zusammengebaut werden kann.

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

12. Lizenzierung und Nutzung von Software.

Android

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

Termux

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

Knotenpunkt

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

Python

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

OpenSSH

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

Kitt SSH

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

MIT App Inventor 2 Companion und App Inventor Blockly

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

Externe Erweiterungen:

JSOONTOOLS

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

Die Lizenzierung von Open Source- und kommerziellen Versionen des QRNG Mini-Systems finden Sie auf der offiziellen Website <http://www.openqbit.com>

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

Mini QRNG ist öffentliches Eigentum.

Der gesamte Code und die Dokumentation in Mini QRNG wurde von den Autoren der Public Domain gewidmet. Alle Code-Autoren und Vertreter der Unternehmen, für die sie arbeiten, haben eidesstattliche Erklärungen unterzeichnet, in denen sie ihre Beiträge der Öffentlichkeit zur Verfügung stellen, und die Originale dieser eidesstattlichen Erklärungen werden in einem Safe in der Zentrale von OpenQbit Mexiko aufbewahrt. Jedem steht es frei, die originalen Mini QRNG (OpenQbit)-Erweiterungen zu veröffentlichen, zu verwenden oder zu verteilen, entweder als Quellcode oder als kompilierte Binärdateien, für jeden Zweck, ob kommerziell oder nicht kommerziell, und mit allen Mitteln.

Der vorstehende Absatz bezieht sich auf den Code und die Dokumentation, die in Mini QRNG geliefert werden, die Teile der Mini QRNG-Bibliothek, die tatsächlich mit einer größeren Anwendung gruppiert und ausgeliefert werden. Einige Skripte, die als Teil des Kompilierungsprozesses verwendet werden (z.B. von autoconf generierte "Konfigurations"-Skripte), können in anderen Open-Source-Lizenzen enthalten sein. Jedoch schafft es keines dieser Kompilierungs-skripte in die endgültige QRNG-Mini-Bibliothek, so dass die mit diesen Skripten verbundenen Lizenzen kein Faktor bei der Bewertung Ihrer Rechte zum Kopieren und Verwenden der QRNG-Mini-Bibliothek sein sollten.

Der gesamte lieferbare Code in Mini QRNG wurde von Grund auf neu geschrieben. Es wurde kein Code von anderen Projekten oder aus dem offenen Internet übernommen. Jede Codezeile kann bis zu ihrem ursprünglichen Autor zurückverfolgt werden, und alle diese Autoren haben Public-Domain-Widmungen in den Akten. Daher ist die QRNG Mini-Code-Basis sauber und nicht durch Code verunreinigt, der von anderen Open-Source-Projekten lizenziert wurde, nicht durch offene Beiträge

Mini QRNG ist Open Source, was bedeutet, dass Sie so viele Kopien machen können, wie Sie wollen, und mit diesen Kopien tun können, was Sie wollen, ohne Einschränkung. Aber Mini QRNG ist nicht Open Source. Um Mini QRNG in der öffentlichen Domäne zu halten und sicherzustellen, dass der Code nicht mit proprietären oder lizenzierten Inhalten verunreinigt wird, akzeptiert das Projekt keine Patches von unbekannten Personen. Der gesamte Code in Mini QRNG ist original, da er speziell für die Verwendung durch Mini QRNG geschrieben wurde. Es wurde kein Code aus unbekannten Quellen im Internet kopiert.

Mini QRNG ist öffentlich zugänglich und erfordert keine Lizenz. Einige Organisationen wollen jedoch einen legalen Beweis für ihr Recht, Mini QRNG zu benutzen. Zu den Umständen, unter denen dies geschieht, gehören die folgenden:

- Ihr Unternehmen will eine Entschädigung für Ansprüche wegen Urheberrechtsverletzungen.
- Sie verwenden Mini QRNG in einer Gerichtsbarkeit, die den öffentlichen Bereich nicht anerkennt.
- Sie verwenden Mini-QRNG in einer Gerichtsbarkeit, die das Recht eines Autors, seine Arbeit der Public Domain zu widmen, nicht anerkennt.
- Sie möchten ein greifbares Rechtsdokument als Beweis dafür haben, dass Sie das Recht haben, Mini QRNG zu verwenden und zu verbreiten.
- Ihre Rechtsabteilung teilt Ihnen mit, dass Sie eine Lizenz kaufen müssen.

Wenn einer der oben genannten Umstände auf Sie zutrifft, wird OpenQbit, die Firma, die alle Mini QRNG-Entwickler beschäftigt, Ihnen eine Mini QRNG-Titelgarantie verkaufen. Eine Titelgarantie ist ein Rechtsdokument, das besagt, dass die beanspruchten Autoren von Mini-QRNG die wahren Autoren sind, und dass die Autoren das Recht haben, das Mini-QRNG der Public Domain zu widmen, und dass OpenQbit sich energisch gegen die Lizenzansprüche

verteidigen wird. Alle Erlöse aus dem Verkauf von Mini QRNG-Titelgarantien werden zur Finanzierung der kontinuierlichen Verbesserung und Unterstützung von Mini QRNG verwendet.

Beisteuernder Code

Um Mini QRNG völlig frei und lizenzfrei zu halten, akzeptiert das Projekt keine Patches. Wenn Sie einen Änderungsvorschlag machen und einen Patch als Proof-of-Concept beifügen möchten, wäre das großartig. Seien Sie jedoch nicht beleidigt, wenn wir Ihren Patch von Grund auf neu schreiben. Die Art der nicht-kommerziellen oder Open-Source-Lizenz, die es in dieser und ähnlichen Modalität ohne den Kauf von Support für Einzelpersonen oder Unternehmen unabhängig von der Größe des Unternehmens verwendet, wird durch die folgenden rechtlichen Prämissen geregelt.

Gewährleistungsausschluss. Sofern nicht durch geltendes Recht gefordert oder schriftlich vereinbart, stellt der Lizenzgeber den Schutzgegenstand (und jeder Beitragende stellt seine Beiträge zur Verfügung) "SO WIE SIE SIND", **OHNE GEWÄHRLEISTUNGEN ODER BEDINGUNGEN IRGEND EINER ART, weder** ausdrücklich noch stillschweigend, einschließlich, aber nicht beschränkt auf Garantien oder Bedingungen des TITELS, der NICHTVERLETZUNG, der MARKTGÄNGIGKEIT ODER DER EIGNUNG FÜR EINEN BESTIMMTEN ZWECK. Sie sind allein verantwortlich für die Bestimmung der korrekten Verwendung oder Weitergabe des Schutzgegenstandes und für die Übernahme aller Risiken, die mit der Ausübung der Berechtigungen unter dieser Lizenz verbunden sind.

Alle finanziellen oder sonstigen Verluste, die durch die Verwendung dieser Software entstehen, werden von der betroffenen Partei getragen. Alle Rechtsstreitigkeiten werden von den Parteien nur den Gerichten in der Gerichtsbarkeit von Mexiko-Stadt, Land Mexiko, vorgelegt.

Für kommerzielle Unterstützung, Nutzung und Lizenzierung muss eine Vereinbarung oder ein Vertrag zwischen OpenQbit oder seinem Unternehmen und dem Interessenten abgeschlossen werden.

Die Bedingungen des Vertriebsmarketings können sich ohne Vorankündigung ändern. Bitte gehen Sie auf die offizielle Website www.openqbit.com, um alle Änderungen der Support- und Lizenzbestimmungen für nicht-kommerzielle und kommerzielle Zwecke zu sehen.

Jede Person, Benutzer, private oder öffentliche Körperschaft jeglicher rechtlicher Natur oder aus irgendeinem Teil der Welt, die die Software einfach nur benutzt, akzeptiert ohne Bedingungen die in diesem Dokument festgelegten Klauseln und diejenigen, die jederzeit im Portal von www.openqbit.com ohne vorherige Ankündigung geändert werden können und

die nach dem Ermessen von OpenQbit in nicht-kommerzieller oder kommerzieller Nutzung angewendet werden können.

Alle Fragen und Informationen über Mini QRNG sollten an die App Inventor-Gemeinschaft oder an die verschiedenen Blockly-System-Gemeinschaften gerichtet werden, so wie sie sind: AppBuilder, Trunkable, etc. und/oder an die E-Mail opensource@openqbit.com für die Anforderung von Fragen kann es 3 bis 5 Werktage dauern, bis sie beantwortet werden.

Unterstützung bei der kommerziellen Nutzung.

support@openqbit.com

Verkauf zur kommerziellen Nutzung.

sales@openqbit.com

Rechtliche Informationen und Lizenzfragen oder Bedenken

legal@openqbit.com

