IBB Ingenieur-Büro Bergmann, Sonnenweg 3, 15537 Grünheide, info@ibbergmann.org

BENUTZERHANDBUCH

des Physikalischen Zufallsgenerators PRG420

Version 1.0

Autor: **Frank Bergmann**

Letzte Änderung: **01.08.2017 11:42**

1 Inhaltsverzeichnis

1 Inhaltsverzeichnis 2
2 Copyright 3
3 Bedeutung von Zufallszahlen 4
4 Kryptografisch sichere Zufallszahlen 4
5 Der hybride Zufallsgenerator PRG420 5
6 Technische Daten des PRG420 6
7 Prinzip der Rauscherzeugung 7
8 Generierung des Zufallssignals 7
9 Entropie 9
10 Generierung von Zufallszahlen 9
11 Sicherheitsfunktionen 10
11.1 Tot-Test 10
11.2 Permanenter Online-Test 10
12 Statistische Qualität 11
13 Sicherheitshinweise 11
14 Anwendungen 12
15 Einsatzumgebung 12
16 Funktionen der Leuchtdioden 12
17 Schnittstelle und Treiber 13
18 Verwendung der PRG100-Software 13
19 Literatur 15

2 Copyright

IBB Ingenieurbüro Bergmann

Sonnenweg 3

D-15537 Grünheide

Der rechtmäßige Erwerb des Physikalischen Zufallsgenerators PRG420 erlaubt eine Nutzung ausschließlich entsprechend Lizenzvertrag.

Schutzrechte

Für wesentliche Schaltungsdetails des Physikalischen Zufallsgenerators sind Schutzrechte eingetragen.

Patente:

- Deutsches Patent DE 102 23 252 vom 18.06 2003
- Europäisches Patent EP 150 98 38 vom 15.03.2006

Haftung

Bei der Erarbeitung dieser Dokumentation wurde größter Wert auf die Vollständigkeit und Richtigkeit des Inhalts gelegt. Es kann dennoch keine Garantie für die Vollständigkeit und Richtigkeit übernommen werden.

Für Hinweise zu dieser Dokumentation sind wir dankbar.

Hotline

Die Hotline des Herstellers erreichen Sie unter 0172 308 6554.

Warenzeichen

MS Windows ist eingetragenes Warenzeichen der Microsoft Corp.

3 Bedeutung von Zufallszahlen

Gute Zufallszahlen sind das Fundament vieler kryptographischer Verfahren und Protokolle. Es ist wichtig, dass die verwendeten Zufallszahlen nicht vorhersagbar sind. Solche Zufallszahlen zu erzeugen, fällt Computern naturgemäß schwer. Zahlreiche Meldungen kritisieren Lücken, Schwächen und Manipulationen bei der Erzeugung von Zufallszahlen für kryptografische Verfahren.

Bei allen Meldungen geht es nicht um spezielle, bedeutungslose Applikationen, sondern um millionenfach installierte Standardprogramme in professionellen Anwendungen. Vor allem dort, wo kontinuierlich viele Zufallszahlen benötigt werden (Netzwerke, Kommunikationssysteme), sind statistische Angriffe auf schwache Zufallsgeneratoren am erfolgreichsten.

Nach dem Kerckhoff-Prinzip (die Sicherheit soll nur auf der Geheimhaltung des Schlüssels beruhen, nicht auf der Geheimhaltung des kryptographischen Algorithmus) benötigt jede Art von Verschlüsselung eine geheime Komponente, die unter keinen Umständen vorhersagbar oder rekonstruierbar sein darf: der aus Zufallszahlen gebildete Schlüssel. Diese Zufallszahlen werden in den bekannten IT-Sicherheitsapplikationen aus Pseudozufallszahlen gebildet. Quelle der Generierung von Pseudozufall ist ein so genannter Seed (ein Startwert, bestehend aus Passwort, Timer-Register, Tastaturanschlägen, Mausbewegungen usw.), mit dem ein mathematisch-kryptografischer Algorithmus eine statistisch gut verteilte Zufallsfolge erzeugt. Aber die gesamte Sicherheit der per Pseudozufall erzeugten geheimen Schlüssel hängt *ausschließlich* von dieser Anfangsinitialisierung ab. Die Anfangsinitialisierung ist bei richtiger Wahl der Quelle der einzige wirklich zufällige Parameter, alles Weitere ist *deterministisch* und somit berechenbar. Eine schwache Anfangsinitialisierung (trivialer Seed) ist im statistischen Ergebnis nicht erkennbar, aber ein effizienter Angriffspunkt der Kryptoanalysis.

Auch professionelle Entwickler nutzen als Seed für Pseudozufall oftmals das Timerregister in der Annahme: wer will denn schon wissen, in welcher Sekunde das Register ausgelesen wurde. Für einen Angreifer kein Problem, denn ein Jahr hat ca. 32 Millionen Sekunden. Und um diese mit der totalen Probiermethode (brute force) durchzutesten, benötigt man nur eine durchschnittliche Rechenleistung. Wird der gleiche Seed mehrfach verwendet, so entstehen schlüsselgleiche Geheimtexte. Ein sicherer Erfolg für die Kryptoanalysis.

4 Kryptografisch sichere Zufallszahlen

Mit dem PRG420 steht ein professioneller Zufallsgenerator der Klasse PTG.3 (hybrider Zufallsgenerator) für die permanente Generierung von kryptografisch sicherem Zufall zur Verfügung. Dieser Zufallsgenerator hat ein USB-Interface (virtuelle COM) und **arbeitet ohne Kommando-Interface**. Nach PON werden kontinuierlich Zufallszahlen mit hoher Geschwindigkeit ausgegeben. Ein permanent im Hintergrund laufendes Sicherheitssystem garantiert, dass bei Ausfall oder Manipulation der Rauschquellen die Zufallsausgabe sofort eingestellt und solange weiter getestet wird, bis alle Qualitätskriterien wieder eingehalten werden.

Die generierten Zufallszahlen sind garantiert kryptografisch sicher und werden vorzugsweise zur Entropieerhöhung und -bereitstellung in Sicherheitsapplikationen verwendet. Unter Linux gibt es bekanntlich immer wieder Probleme mit der Bereitstellung von Zufall im Entropie-Pool. Besonders kritisch wird die Situation, wenn keine Eingabegeräte an einem Server zur Verfügung stehen.

Die Qualität der vom PRG420 erzeugten Zufallszahlen ist qualitativ unvergleichlich höher und sicherer, als jede andere Art der Zufallserzeugung. Zum Verifizieren dieser Aussage stehen diverse Analysen des Zufallsgenerators, auch unter erhöhter thermischer Belastung, zur Verfügung. Die hohe Ausgabegeschwindigkeit des

PRG420 ermöglicht eine Füllung des Entropie-Pools (4096 Bit) in max. 5,74ms. Eine kryptografische Nachbearbeitung der ausgegebenen Zufallsdaten ist prinzipiell nicht erforderlich.

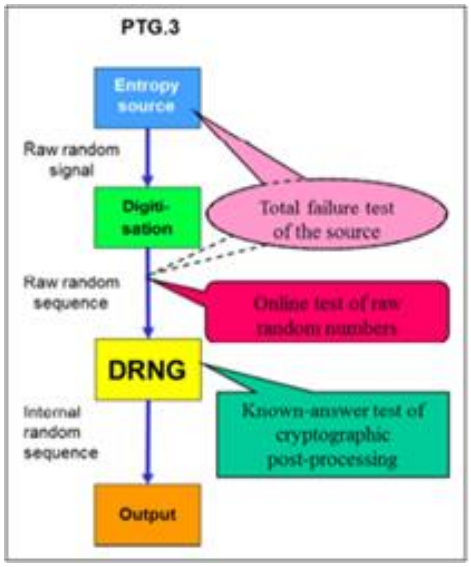
5 Der hybride Zufallsgenerator PRG420

Bei dem PRG420 handelt es sich um einen physikalischen Zufallsgenerator mit einer USB-Schnittstelle (virtuelle COM). Der PRG420 ist voll hardware-kompatibel zum universellen PRG310. Der PRG420 ist mit einem Micro-USB-Anschluss ausgestattet.

Abbildung: PRG420

Der Zufallsgenerator ist in einem stabilen Metallgehäuse untergebracht. Kern des PRG420 ist ein patentierter physikalischer Zufallsgenerator des IBB:

- Deutsches Patent DE 102 23 252 vom 18.06 2003
- Europäisches Patent EP 150 98 38 vom 15.03.2006



Hybride Zufallszahlengeneratoren vereinen Sicherheitseigenschaften von deterministischen und physikalischen Zufallszahlengeneratoren und besitzen neben einer starken Rauschquelle eine starke kryptographische Nachbearbeitung mit Gedächtnis. Die Klasse PTG.3 stellt nach Algorithmenkatalog 2015 **Fehler! erweisquelle konnte nicht gefunden werden.** der Bundesnetzagentur (BNetzA) die *stärkste Funktionalitätsklasse* dar.

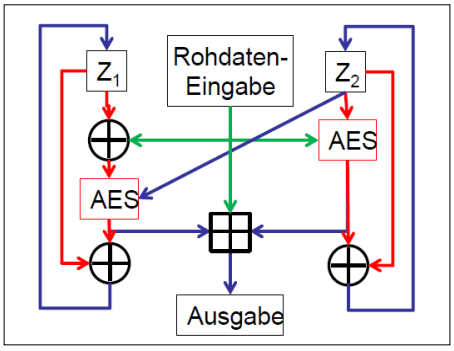
Für die Erzeugung und Bewertung von kryptografisch sicheren Zufallszahlen sind eindeutige Normen und Regeln vorgegeben. Diese sind neben dem Algorithmenkatalog in den AIS31-Dokumenten des Bundesamtes für Sicherheit in der Informationstechnik (BSI) festgelegt und bestimmen die notwendigen Eigenschaften von unbearbeiteten Zufallszahlen (Rohdaten): „*Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren*“ **Fehler! erweisquelle konnte nicht gefunden werden.**. Demnach muss ein

kryptografisch sicherer Zufallsgenerator z.B. eine Entropie von mehr als 7,97 Bit/Byte haben. Weiterhin müssen die Zufallsbits bestimmten Eigenschaften genügen, wie die Gleichverteilung, Bitunabhängigkeit und die Bewertung von 2-, 3-, und 4-Bit Zufallsfolgen.

Grundsätzlich verlangt die Bundesnetzagentur im aktuellen Algorithmenkatalog:

„Für Zertifizierungsdienstanbieter wird die Verwendung von Zufallsgeneratoren der Funktionalitätsklassen *PTG.3* und DRG.4 im Grundsatz *ab 2015 verpflichtend* werden, sowohl allgemein bei der Erzeugung von Langzeitschlüsseln als auch bei der Erzeugung von Ephemeralschlüsseln."

Die kryptografische Nachbereitung der Rohdaten nach PTG.3 sind im Schema rechts dargestellt. Die Rohdaten-Eingabe erfolgt mit 16 Byte Zufallszahlen des physikalischen Zufallsgenerators und ergeben nach der Verarbeitung 16 Byte Ausgabedaten. Es wird also *explizit kein Pseudozufall* generiert. Einschätzungen zur Qualität geben statistische Analysen für Rohdaten und Zufallsdaten der Klasse PTG.3 auf der Homepage des Autors.



6 Technische Daten des PRG420

Abmessungen:	85*55*16 mm
Versorgungsspannung:	5V (+/- 10%) aus USB-Port
Stromaufnahme:	max. 200mA
Temperaturbereich:	funktionell und statistisch stabil von -20°C..+85°C
Schnittstelle:	USB1.1-Interface über virtuellen COM, 921.600 bps, Protokoll 8,N,1
Anschluss:	Micro-USB, Typ B
Qualitätssicherung:	automatischer Selbstabgleich von Verstärkung und Digitalisierung
	Tot-Test zur Überwachung der Rauschquellen
	Abschaltung der Zufallsausgabe bei Ausfall einer Rauschquelle
	Permanenter Online-Test pro Sekunde zur statistischen Überwachung des Zufallssignals
Entropie:	>7,99 Bits/Byte (ermittelt aus Zufalls-Rohdaten nach AIS31)
0/1-Verhältnis:	garantiert im Bereich 0,49..0,51 (Rohdaten > 8.000 Bit)
Ausgabegeschwindigkeit:	ca. 712 Kbit/s netto
Entropie-Pool füllen:	innerhalb 5,74ms (=4096 Bit)

7 Prinzip der Rauscherzeugung

Rauschen ist ein physikalisches Phänomen und stellt eine Störgröße mit breitem unspezifischem Frequenzspektrum dar. Dieses Frequenzspektrum besteht aus der Überlagerung mehrerer Schwingungen oder Wellen mit unterschiedlicher Amplitude und Frequenz beziehungsweise Wellenlänge. Diese Eigenschaften wurden erstmalig 1918 durch Walter Schottky beschrieben. Später wurde das thermische Rauschen experimentell durch John Bertrand Johnson verifiziert. Eine Modellvorstellung der spektralen Leistungsdichte des thermischen Rauschens erfolgte durch Harry Nyquist.

Das in diesem Zufallsgenerator verwendete 1/f-Rauschen bezeichnet ein Rauschen, dass mit steigender Frequenz abnimmt, die Amplitudenverteilung ist umgekehrt proportional zur Frequenz (~ 1/f). Die verwendeten Rauschquellen sind Z-Dioden, die in Sperrrichtung betrieben werden. Rauschen entsteht hier durch den Lawineneffekt (Avalancheeffekt) in der pn-Sperrschicht des Halbleiterbauelements (Dioden und Transistoren). Dioden und Transistoren lassen sich durch ein kontrolliertes Avalanche-Verhalten vor Zerstörung durch Überspannungen schützen.

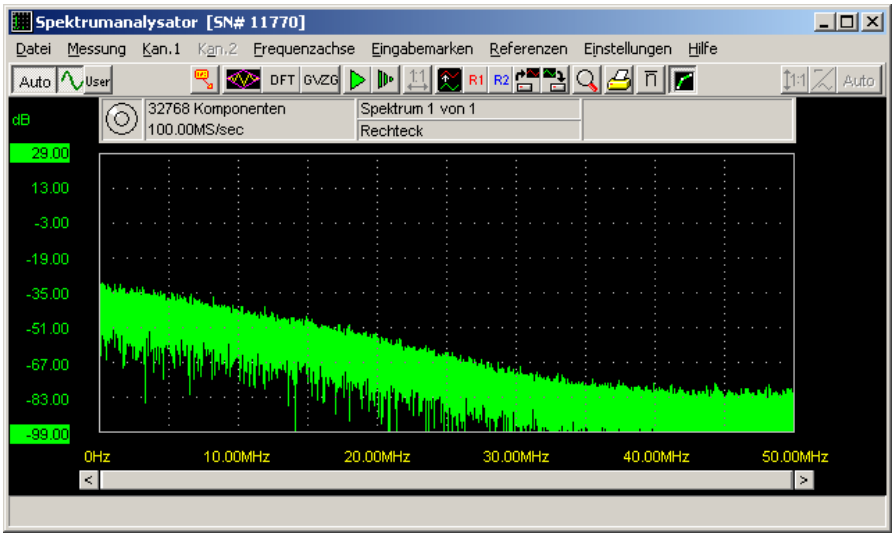


Abbildung: Typisches 1/f-Rauschen nach der Verstärkung

8 Generierung des Zufallssignals

Die Rauschsignale zweier Z-Dioden werden einem Differenzverstärker zugeführt, der eine ca. 300-fache Verstärkung realisiert und durch seine hohe Gleichtaktunterdrückung Störsignale und Artefakte sehr wirksam eliminiert. Ein schneller Schmitt-Trigger mit einer Hysteresis von ca. 0,3V digitalisiert das verstärkte Rauschsignal und führt es einem Porteingang des Mikrocontrollers zu.

Das Blockschaltbild zeigt wesentliche Komponente der 8 implementierten Zufallsgeneratoren:

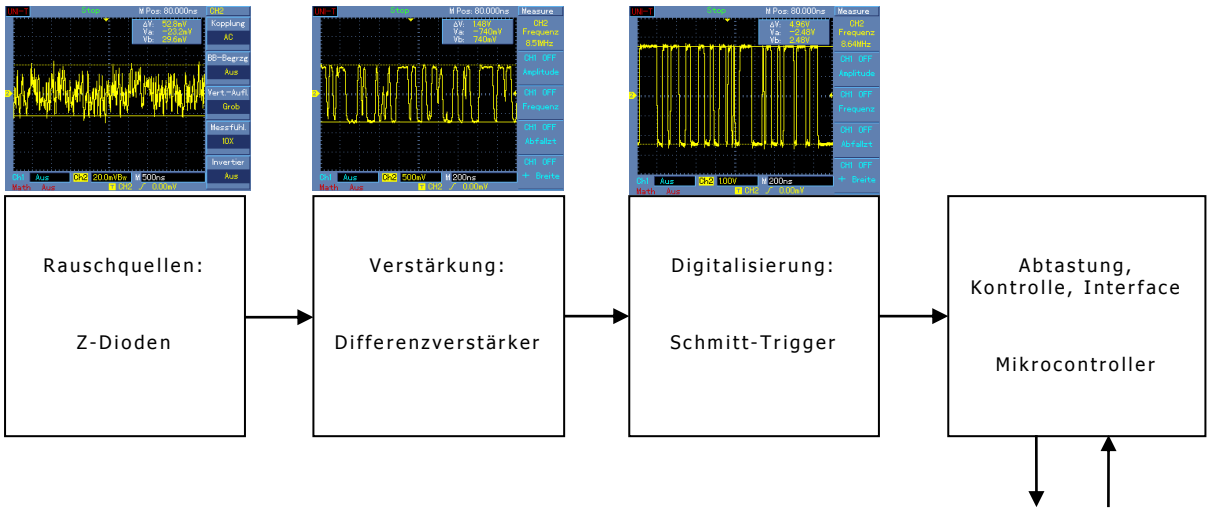


Abbildung: Prinzipschaltbild der Zufallserzeugung

Um eine hohe Ausgabegeschwindigkeit der Zufallszahlen zu erreichen, sind im PRG420 8 Zufallsgeneratoren implementiert, die sequentiell abgetastet werden. Im Bild unten ist das Prinzip erkennbar. M1 bis M8 sind die autonom arbeitenden Zufallsgeneratoren. Ein zentraler Mikrocontroller fragt mit „RXD" den jeweiligen Zufallsgenerator ab und erhält 128 Bit kryptografisch sicheren Zufall. Das „TXD"-Signal entspricht der PRG420-Ausgabe des USB-Ports.

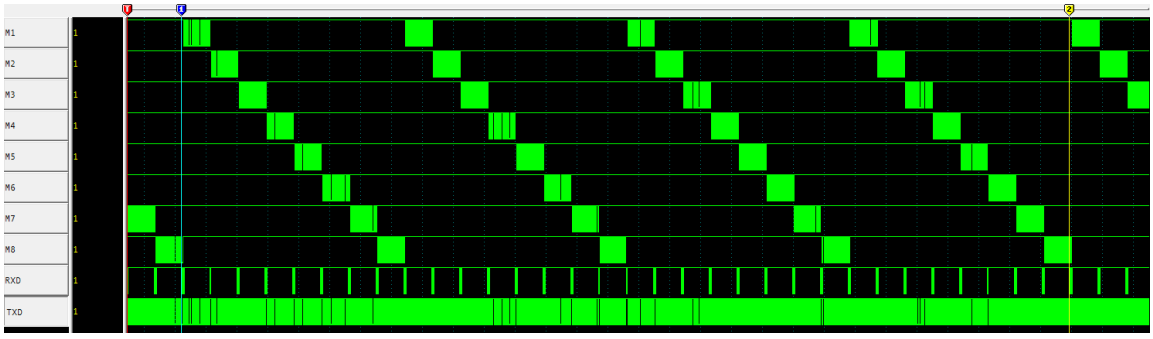


Abbildung: sequentielle Abtastung der Zufallszahlen

Der markierte Bereich (blau-gelb) oben zeigt die Zeit zur Generierung von 4096 Zufallsbit zur Füllung des Entropie-Pools unter Linux und beträgt 5,74ms (entspricht 712Kbit/s netto).

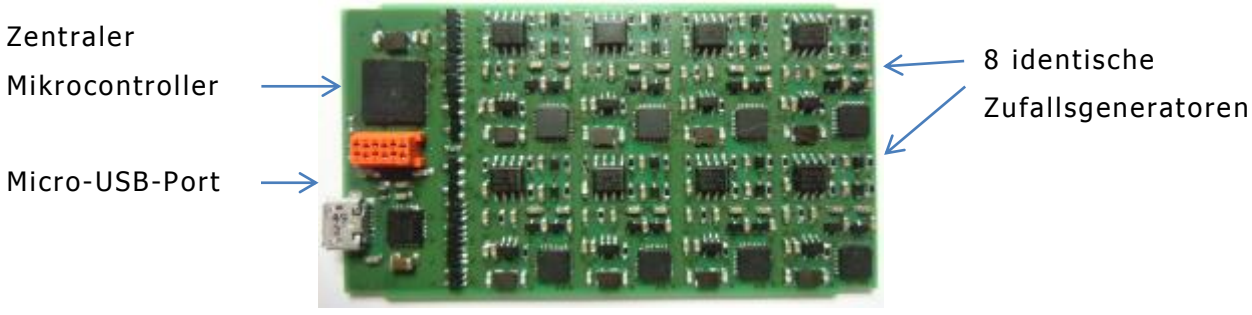


Abbildung: bestückte Musterplatine

9 Entropie

Die Entropie der Zufallsrohdaten ist die entscheidende Eigenschaft eines echten Zufallsgenerators und sollte so hoch als möglich sein. Die reproduzierbare Generierung von Zufallsdaten mit hoher Entropie der PRG420-Applikationen zeigen beispiellose Werte. Da der PRG420 ohne Kommando-Interface arbeitet, wurde zur Analyse der Entropie die kryptografische Nachbearbeitung nach PTG.3 abgeschaltet.

Folgende Entropiewerte (nach AIS31) der Rohdaten wurden mit dem PRG420 ermittelt (jeweils 10Mbyte Zufallszahlen):

PRG420	Mittelwert 0/1	Entropie nach AIS31
Bei +20°C	0.499710	8.00155318
Bei +85°C	0.498630	8.00099945
Bei -20°C	0.500080	7.99676296

Abbildung: Entropie der Rohdaten des PRG420

Details zu den AIS31-Analysen befinden sich auf der Homepage des Autors (www.ibbergmann.org).

10 Generierung von Zufallszahlen

Grundlage der Entwicklung des PRG420 ist ein stochastisches Modell, mit dem die Leistungsfähigkeit zur Generierung kryptografisch sicherer Zufallszahlen begründet wird. Dieses Modell erklärt:

- die robuste und hohe Entropie der Rauschquelle
- das Prinzip der Abtastung des analogen Rauschsignals,
- die permanente Überwachung der Rauschquelle durch Frequenzmessung
- die kryptografische Nachbearbeitung durch Mayer-Einwegfunktionen
- den permanenten statistischen Online-Test

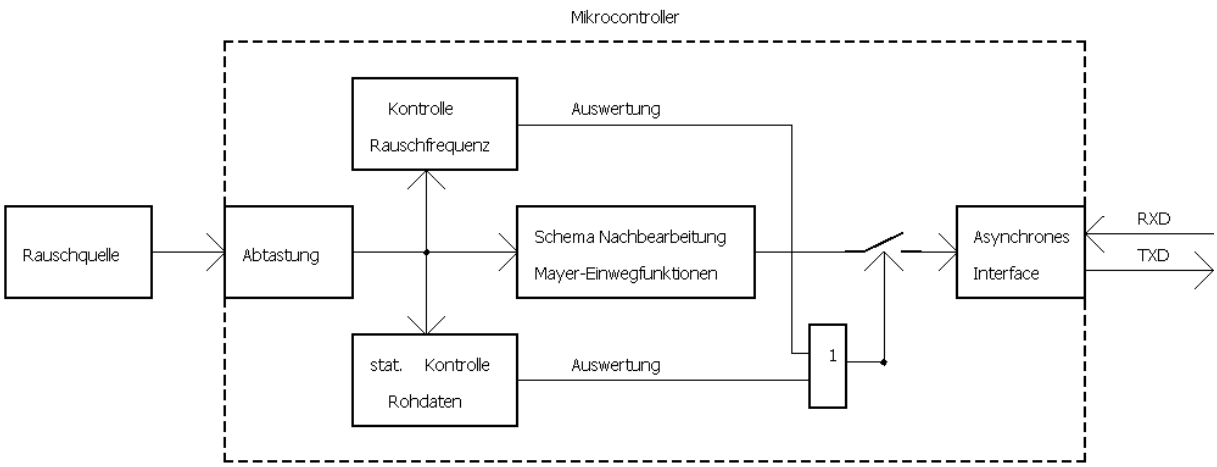


Abbildung: Schema zum stochastischen Modell jedes der 8 Zufallsgeneratoren

11 Sicherheitsfunktionen

11.1 Tot-Test

Es ist ein Kontrollsystem installiert, welches das digitalisierte Rauschsignal am Anschluss des Mikrocontrollers überwacht. Unmittelbar vor und nach jeder Byte-Generierung des digitalisierten Rauschsignals wird eine Funktion aufgerufen, die folgende Aufgabe hat:

- Es wird die Zeit ermittelt die erforderlich ist, um vier wechselnde Flanken des digitalisierten Rauschsignals zu erfassen
- Wird nach 16µs (entspricht 125 KHz) diese Bedingung nicht erfüllt, wird eine Fehlermeldung generiert, die zur sofortigen Einstellung aller Zufallsausgaben führt
- Dieser Zustand ist nur durch ein positives Ergebnis des weiter laufenden Tot-Test aufzulösen
- Typische Zeiten, um die Bedingung zu erfüllen, sind 2..8µs

Sollte eine der Rauschquellen ausfallen entsteht am Eingang des Mikrocontrollers ein Mäander von ca. 20ms und wird durch den Tot-Test eindeutig als Fehlzustand erkannt.

Wird von einem der 8 implementierten Zufallsgeneratoren ein Fehler detektiert, stoppt auch die Ausgabe des zentralen Mikrocontrollers. Dieser Fehlerzustand wird durch die Leuchtdioden angezeigt.

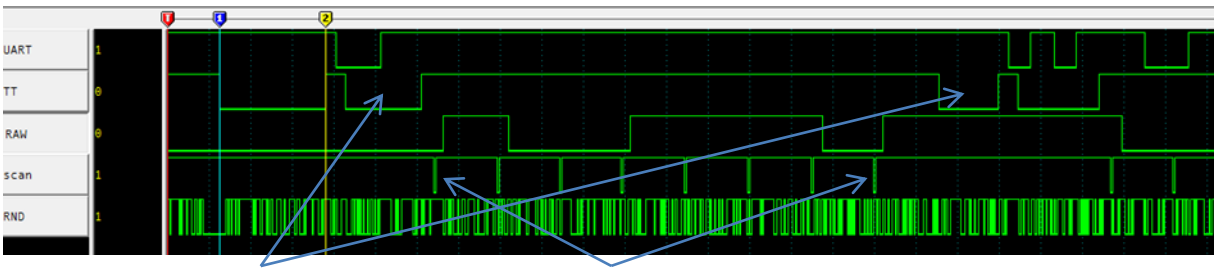


Abbildung: Tot-Test Abtastung (8x) des digitalisierten Rauschsignals

11.2 Permanenter Online-Test

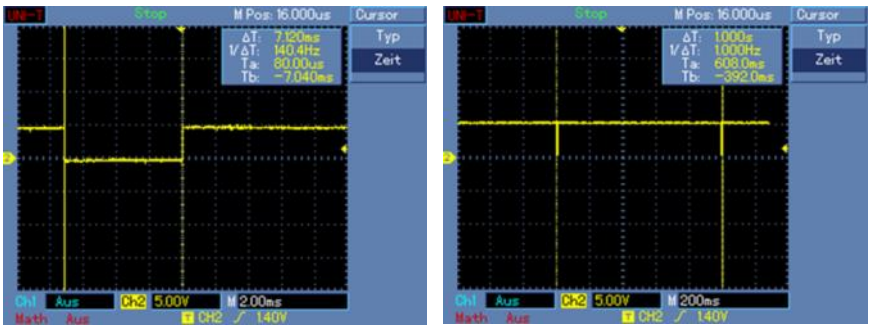
Im permanenten Halbbytetest (zyklisch im Abstand von 1 Sekunde) zur Kontrolle der statistischen Qualität der Zufallsrohdaten werden folgende Kriterien geprüft:

- Sind die Werte innerhalb der statistischen Vorgaben, wird kein Fehler generiert
- Ist einer der 16 Werte im Halbbytetest gleich Null, wird von einem Totalausfall mindestens einer Rauschquelle ausgegangen und jede Zufallsausgabe blockiert.
- Dieser Zustand ist nur durch ein positives Ergebnis des weiter laufenden Online-Test aufzulösen

Wird von einem der 8 implementierten Zufallsgeneratoren ein Fehler detektiert, stoppt auch die Ausgabe des zentralen Mikrocontrollers. Dieser Fehlerzustand wird durch die Leuchtdioden angezeigt.

Der permanente Online-Test dauert 7,12ms (links)

Im Abstand von 1 Sekunde (rechts)



12 Statistische Qualität

Dieser physikalische Zufallsgenerator generiert kontinuierlich Zufallsbits in herausragender statistischer Qualität. Eine Qualitätsaussage zu einem solchen Produkt ist aber nicht durch einen einzelnen statistischen Test möglich. In Zusammenarbeit mit Mathematikern und Kryptologen hat sich die Summe aus folgenden statistischen Tests für eine sichere Qualitätsaussage eines physikalischen Zufallsgenerators bewährt:

- Statistische Forderungen der AIS31 für Rohdaten (keine digitale Nachbearbeitung; nichts beschreibt besser die Eigenschaften eines physikalischen Zufallsgenerators, als seine Rohdaten!)
- NIST-Test-Suite (Summe verschiedener Test der USA-Sicherheitsbehörde)
- Diehard-Test-Suite nach George Marsaglia
- Proprietärer statistischer Basistest

Zur Evaluierung wurden umfangreiche statistische Tests durchgeführt. So wurden die ausgewählten Tests auf mehrere erzeugte Bitfolgen angewendet. Keine dieser Testfolgen konnte Unterschiede zu einem idealen Zufallszahlengenerator aufzeigen. Darüber hinaus wurden Testergebnisse mit Untersuchungen von kryptographisch starken Pseudo-Zufallszahlengeneratoren, wie dem DES-, AES- und SH1-Generator verglichen. Die Vergleiche zeigten keine signifikanten Unterschiede zu den Testergebnissen dieser deterministischen Zufallsgeneratoren.

Details zu den Analysen befinden sich auf der Homepage des Autors (www.ibbergmann.org).

13 Sicherheitshinweise

Um eine sichere Generierung von Zufallszahlen zu gewährleisten, sind folgende Sicherheitshinweise zu beachten:

- Zugriff und Nutzung in sicherheitsrelevanten Bereichen ist nur autorisierten Personen zu gestatten
- Der Einsatz des PRG420 in einem geschlossenen Gehäuse erschwert eine Manipulation der Zufallserzeugung und ist zu bevorzugen

14 Anwendungen

Der stetig steigende Bedarf an Zufallszahlen in vielen Bereichen von Wissenschaft und Technik erfordert eine zuverlässige und stabile Generierung von Zufallszahlen mit physikalischem Zufall und sicherer Gewährleistung aller erforderlichen statistischen und funktionellen Normen. Damit sind vor allem Entwickler von Sicherheitsapplikationen in der Lage, Wirksamkeit und Widerstand gegen Angriffe besser zu kalkulieren. Besonders hohe Ansprüche an die Statistik von Zufallszahlen werden für kryptografisch sichere Zufallszahlen gestellt.

Insbesondere die Implementierung unter Linux zur permanenten und schnellen Entropie-Erhöhung, ist eine wesentliche Anwendung.

Exemplarische Beispiele für den Einsatz des PRG420:

- Netzwerksicherheit
- Transaktionen beim Homebanking (SSL-Verschlüsselung)
- Dateiverschlüsselung mit Security-Applikationen (Speicher-Sticks)
- einfachere Administration und sichere Verschlüsselung bei drahtloser Datenübertragung: WLAN, Bluetooth, GSM, ZiggBee, Industriedatenfunk
- Internet-Verschlüsselung
- elektronischer Zahlungsverkehr
- Erstellung von PKI-Zertifikaten
- OneTimePad-Verfahren

Weitere Informationen im Kontext finden Sie im Dokument:

„Professionelle_Zufallsgeneratoren.pdf"

15 Einsatzumgebung

Der PRG420 ist für den permanenten Einsatz in beliebigen PC-Systemen entwickelt und getestet worden. Auch bei erhöhtem Industriestandard (-20°C bis +85°C) bleiben die Entropiewerte sehr hoch und unterschreiten die Vorgaben aus den AIS31-Forderungen nicht. Getestet wurden Applikationen in einem Temperaturbereich von -60 bis +110°C. Alle Grenzwerte der Zufallsrohdaten (Entropie, Halbbytetest) wurden nicht überschritten. Statistische Analysen, auch für diese Temperaturbereiche, befinden sich auf der Homepage des Autors.

16 Funktionen der Leuchtdioden

Die auf der Platine befindlichen Leuchtdioden reflektieren die jeweils ablaufenden Funktionen und Zustände des PRG420. Das Blinken der Leuchtdioden erfolgt immer im Sekundentakt. Synchron zur Änderung des Blinkens (ein→aus und aus→ein) wird der permanente Online-Test gestartet.

OK (grün)	BUSY (gelb)	Zustand
Blinkt	Ein	Selbsttest ok, Zufallszahlen werden ausgegeben
Ein	Aus	Fehler im internen Test, keine Zufalls-Ausgabe

17 Schnittstelle und Treiber

Die Schnittstelle des PRG420 basiert auf einen Chip der Firma FTDI (http://www.ftdichip.com/FTDrivers.htm). Auf der Internetseite werden alle aktuellen Betriebssysteme unterstützt, wobei WINDOWS ab Vista bereits den Chip automatisch erkennt und unterstützt.

Ist der PRG420 erkannt, wird ein unikater virtueller Treiber mit einer nummerierten COM eingerichtet. In der Systemsteuerung ist die eingerichtete COM-Schnittstelle eindeutig spezifiziert.

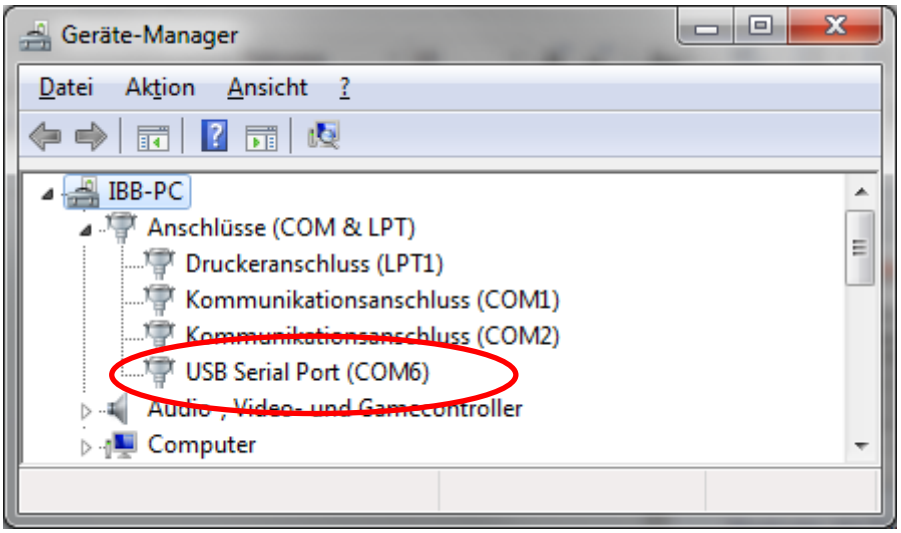


Abbildung: nach dem Einstecken des PRG420 eingerichteter virtueller COM-Port

18 Verwendung der PRG100-Software

Der PRG420 ist vollkompatibel zum PRG320 und kann in der kostenlos angebotenen Software PRG100 unter dieser Einstellung verwendet werden.

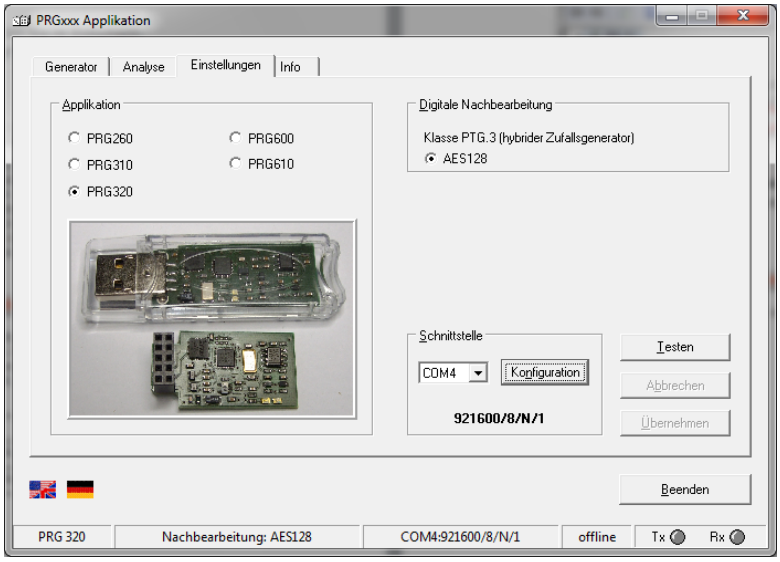


Abbildung: Einstellungen mit der PRG100-Software

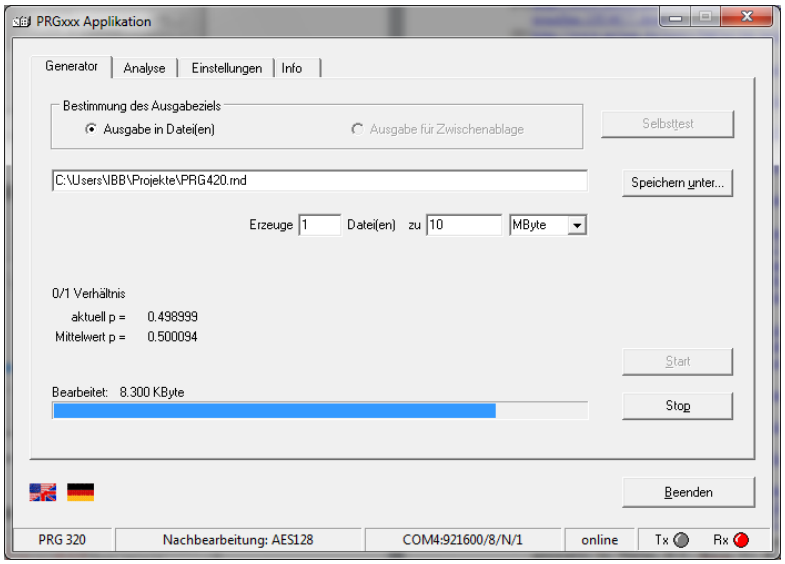


Abbildung: Generierung einer 10Mbyte-Datei

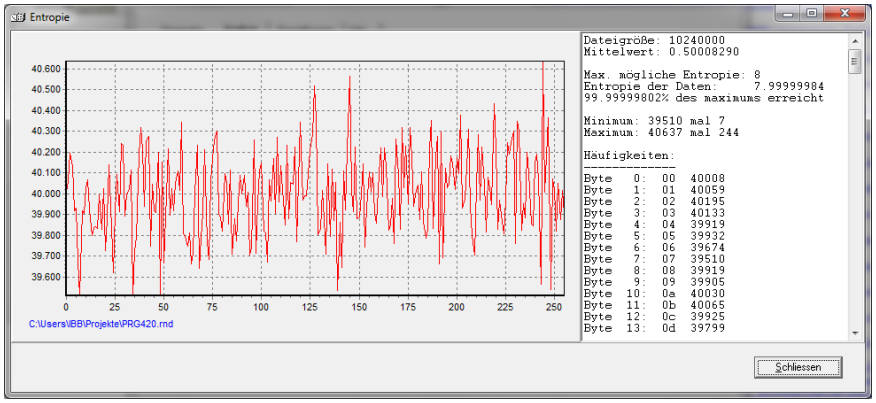


Abbildung: Grafische Analyse der erzeugten Datei

19 Literatur

(1) http://www.golem.de/news/bsafe-rsa-security-warnt-vor-nsa-zufallsgenerator-1309-101727.html

(2) http://www.golem.de/news/elliptische-kurven-die-herkunft-der-nist-kurven-1309-101567.html

(3) http://www.golem.de/news/verschluesselung-nist-raet-von-dual-ec-drbg-wegen-moeglicher-nsa-backdoor-ab-1309-101521.html

(4) http://www.heise.de/newsticker/meldung/NIST-laesst-Zufalls-Generatoren-neu-pruefen-1954677.html

(5) http://www.golem.de/news/fehler-im-zufallsgenerator-netbsd-erzeugt-schwache-schluessel-1303-98350.html

(6) http://www.heise.de/newsticker/meldung/PHP-stuempert-bei-Zufallszahlen-967062.html

(7) http://www.heise.de/security/meldung/Mathematiker-entlarvt-schwache-DKIM-Schluessel-1736107.html

(8) http://www.heise.de/security/meldung/MIPS-Router-mit-Entropieproblemen-1953097.html

(9) http://www.heise.de/security/meldung/OpenSSL-erzeugt-zu-oft-den-gleichen-Zufall-1942299.html

(10) http://www.heise.de/newsticker/meldung/Verschluesselungsstandard-unter-Backdoor-Verdacht-196659.html

(11) http://www.heise.de/newsticker/meldung/Zu-wenig-Zufall-im-Zufallszahlengenerator-von-OpenBSD-178124.html

(12) http://www.heise.de/newsticker/meldung/Apache-Tool-erzeugt-Passwort-Hashes-mit-vorhersagbaren-Salts-180864.html

(13) http://www.heise.de/security/artikel/Gute-Zahlen-schlechte-Zahlen-270078.html

(14) http://www.heise.de/security/meldung/NSA-Affaere-Generatoren-fuer-Zufallszahlen-unter-der-Lupe-1953716.html

(15) http://www.golem.de/news/freebsd-misstrauen-bei-rngs-von-intel-und-via-1312-103305.html

(16) http://www.heise.de/newsticker/meldung/JavaScript-Engine-V8-Vorsicht-vor-Math-random-3010353.html

(17) http://www.golem.de/news/verschluesselung-2013-das-jahr-der-kryptokalypse-1312-103617-2.html

(18) https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_31_pdf.html

(19) (http://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/QES/Veroeffentlichungen/Algorithmen/2015Algorithmenkatalog.pdf;jsessionid=2EFC232C278CA15479747CF5FB36D32C?__blob=publicationFile&v=1

(20) Dichtl, M.: How to predict the output of a hardware random number generator. In: Walter, D.C., Ko»c, C» .K., Paar, C. (eds.) CHES 2003. LNCS, vol. 2779, pp. 181{188.

(21) https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_31_Functionality_classes_for_random_number_generators_e.pdf?__blob=publicationFile