

# Red Pitaya Messautomatisierung

---

## Red Pitaya Messautomatisierung

- Einleitung

- DCsweep

  - Probleme

    - Generierung der Gleichspannung

    - Größere und kleinere Spannungen als +1V, -1V

- Amplitudengang

  - Probleme

    - Messbereich

- Phasengang

  - Bestimmung der Periode

  - Bestimmung der Phasenverschiebung

  - Probleme

    - Rauschen am Ausgang

- Fazit

- Quellen

## Einleitung

---

Der Red Pitaya verfügt über eine SCPI (Standard Command for programmable Instrumentation) Schnittstelle, die über remote Control von Programmen wie Matlab, Python, LabView oder SCILAB gesteuert werden kann. Hierdurch können komplexe Signale generiert und über den Output ausgegeben werden. Des Weiteren können über die 'fast analog Inputs' Messdaten aufgezeichnet und in Form von Rohdaten abgespeichert werden. Da die Messdatenerfassung mittels Matlab, Python, LabView oder SCILAB durchgeführt wird, lassen sich die aufgezeichneten Messdaten hervorragend aufbereiten.

Eine ausführliche Dokumentation mit Beispielen, der Befehlskatalog und eine Anleitung für den Verbindungsaufbau zwischen Red Pitaya und Mess-PC ist unter folgendem Link zu finden:

<https://redpitaya.readthedocs.io/en/latest/appsFeatures/remoteControl/remoteControl.html>

Im Rahmen des "Projekts Analog System Lab Kit Pro Texas" wurden drei Skripte mittels Python für die Messautomatisierung geschrieben und benutzt. Auf den folgenden Seiten werde ich näher auf die Funktionsweise, Probleme bei der Entwicklung und auf evtl. Verbesserungen der Skripte eingehen.

# DCsweep

---

Das DCsweep Skript kann zur Ermittlung der Übertragungseigenschaften von Systemen genutzt werden wenn das Gleichspannungsverhalten untersucht werden soll.

Hierfür werden durch den Benutzer ein Start- und Endspannung vorgegeben. Die Spannungen dürfen dabei nicht größer als 1V und nicht kleiner -1V sein. Hierbei ist eine Eingabe der Messpunkte notwendig. Dadurch wird die Schrittweite zwischen Start- und Endwert errechnet.

Im folgenden ein Beispiel. Es wird ein Startwert von -1V und ein Endwert von +1V mit 11 Messpunkten gewählt. In diesem Fall werden folgende Gleichspannungen generiert und durch den 'fast analog Output' ausgegeben.

$V_{out}$  : -1.0V, -0.8V, -0.6V, -0.4V, -0.2V, 0V, 0.2V, 0.4V, 0.6V, 0.8V, 1.0V

## Probleme

### Generierung der Gleichspannung

In der oben genannten Dokumentation ist nicht beschrieben wie eine Gleichspannung generiert werden kann. Hierfür muss die Wellenform auf 'square' und die Frequenz auf '0' gesetzt werden. Nach setzen der genannten Parameter lassen sich dann über das Einstellen der Amplitude verschiedene Gleichspannungen generieren.

Jedoch lässt sich nicht ohne Weiters eine negative Spannung erzeugen. Hierfür ist das einstellen eines Offsets notwendig. Angenommen eine Gleichspannung von -0.5V soll erzeugt werden. In diesem Beispiel ist ein Offset von -1.0V und eine Amplitude von 0.5V zu wählen.

### Größere und kleinere Spannungen als +1V, -1V

Ein Weiters Problem ist, dass mit dem Red Pitaya sich keine größeren Spannungen als +1V und keine kleineren als -1.0V erzeugen lassen. Doch da sich das Projekt größtenteils mit Operationsverstärkern beschäftigt ist es naheliegend ein nicht-invertierendes Verstärker vorzuschalten um größere oder kleinere Spannungen zu generieren.

Des Weiteren ist darauf zu achten, dass die Messspitzen auf das Verhältnis 1:10 einzustellen sind. Hierdurch wird der Messbereich von -1V - +1V auf -10V - +10V vergrößert.

## Amplitudengang

---

Mit dem Amplitudengang Skript lässt sich das Übertragungsverhalten von Systemen bei verschiedenen Frequenzen messen. Hierbei wird das Verhältnis von Ausgangsspannung zu Eingangsspannung von der Startfrequenz bis zur Endfrequenz ermittelt. Somit lässt sich bestimmen bei welchen Frequenzen das System ein verstärkendes oder dämpfendes Verhalten aufweist. Die Start- und Endfrequenzen sind durch den Benutzer zu wählen. Dabei ist, wie beim DCsweep eine Eingabe der Messpunkte notwendig.

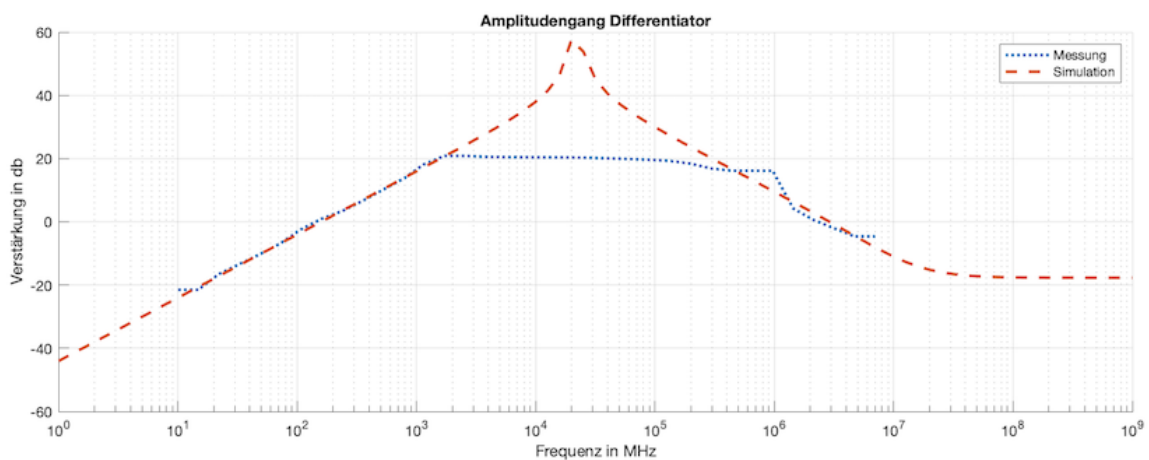
Auch in dieser Messung sind die Messspitzen auf das Verhältnis 1:10 einzustellen.

# Probleme

## Messbereich

Da das Verhältnis an den Messspitzen nicht größer als 1:10 eingestellt werden kann wird in manchen Fällen das Ende des Messbereichs erreicht. In der folgenden Abbildung ist der Amplitudengang eines Differentiator zu sehen. Es wurde ein Frequenzsweep mit einem Sinus  $2V_{pp}$  durchgeführt. In der Abbildung ist zu erkennen, dass die Messung nicht ganz mit der Simulation übereinstimmt. Ursache dafür ist, dass eine Verstärkung größer als 20dB nicht gemessen werden kann da das Ende des Messbereichs erreicht wird.

Dieser Fehler könnte durch eine kleinere Amplitude im Eingangssignal verringert werden. Das ist jedoch keine optimale Lösung.



## Phasengang

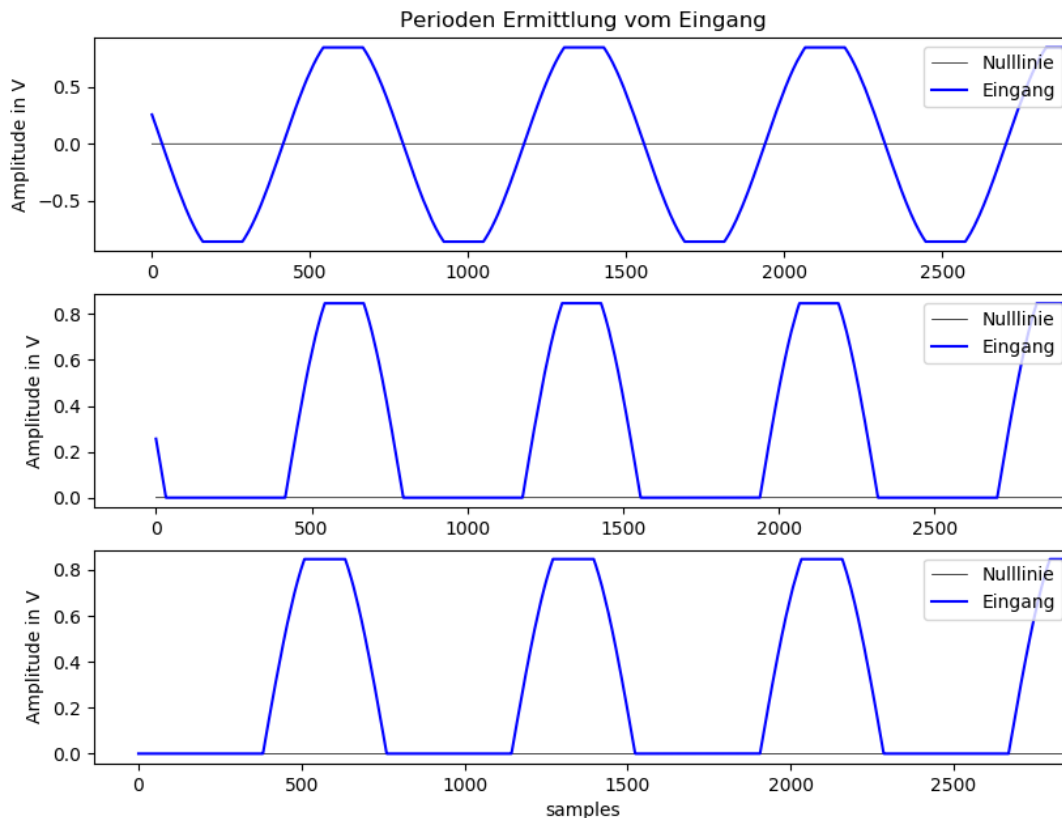
Mit dem Phasengang Skript lässt sich die Phasenverschiebung zwischen Eingangs- und Ausgangssignal messen und aufzeichnen. Durch die Detektion der Nulldurchgänge vom Eingangs- und Ausgangssignal wird die Phasenverschiebung ermittelt.

Für die Messung sind die Messspitzen auf das Verhältnis 1:1 einzustellen.

## Bestimmung der Periode

Zur Bestimmung der Periode sind Weitere Aufbereitungen an den Messdaten notwendig. Die erforderlichen Schritte sind in folgender Abbildung zu sehen. Im obersten Graphen ist das unbearbeitete Eingangssignal dargestellt. Im ersten Schritt ist festzustellen ob die ersten Samples größer oder kleiner null sind. Im unteren Dargestellten Fall sind die ersten Samples größer null und somit werden die Rohdaten folgendermaßen aufbereitet. Alle Werte die kleiner null sind, werden durch den Wert null ersetzt. Wie in der mittleren Abbildung zu sehen. Anschließend werden im zweiten Schritt die ersten Samples die größer null sind ausgeschitten. Siehe unterste Abbildung.

Durch die Aufbereitung der Messdaten wurden die unteren Halbwellen des Sinus durch nullen ersetzt und dies entspricht genau einer halben Periode. Zur Berechnung der Periode werden nun die Samples gezählt bis der Eingang größer null ist und das Ergebnis wird mit zwei Multipliziert. Dies entspricht der Periodendauer in Samples.



## Bestimmung der Phasenverschiebung

Da die Periodendauer im Schritt zuvor bestimmt worden ist, wird nun der Zeitpunkt des ersten Nulldurchgangs sowohl vom Eingang als auch vom Ausgang detektiert und miteinander verglichen. In der untern Abbildung tritt der erste Nulldurchgang vom Eingang bei Sample 191 und vom Ausgang bei Sample 382 auf. Die Periodendauer beträgt 764 Sampleles. Mit folgenden Formeln lassen sich die Phasenverschiebungen berechnen. Hierbei sind vier Fallunterscheidungen zu beachten.

Input, Output ersten samples  $> 0$  :  $\phi = ((Input - Output)/T) * 360$

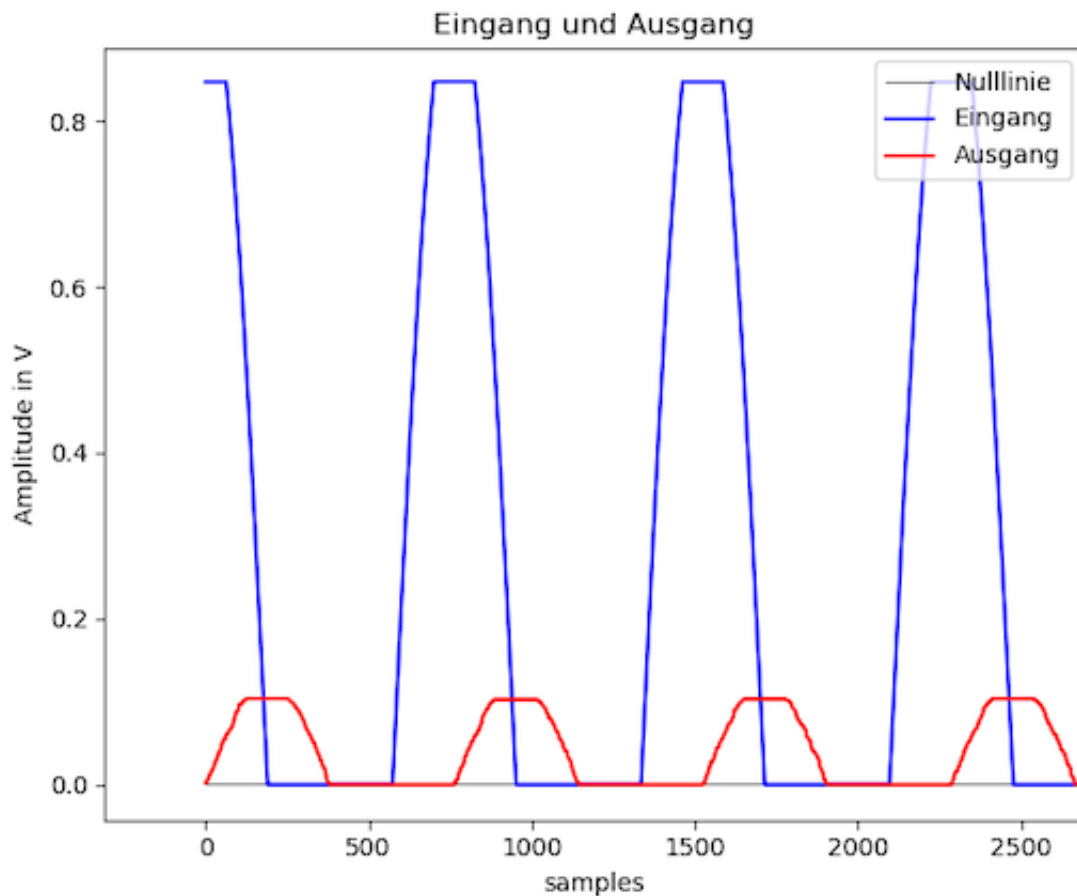
Input, Output ersten samples  $< 0$  :  $\phi = ((Input - Output)/T) * 360$

Input er. samp.  $< 0$ , Output er. samp.  $> 0$  :  $\phi = ((Input - (Output + T/2))/T) * 360$

Input er. samp.  $> 0$ , Output er. samp.  $< 0$  :  $\phi = ((Input - (Output + T/2))/T) * 360$

In unserem Beispiel erhalten wir für die Phasenverschiebung:

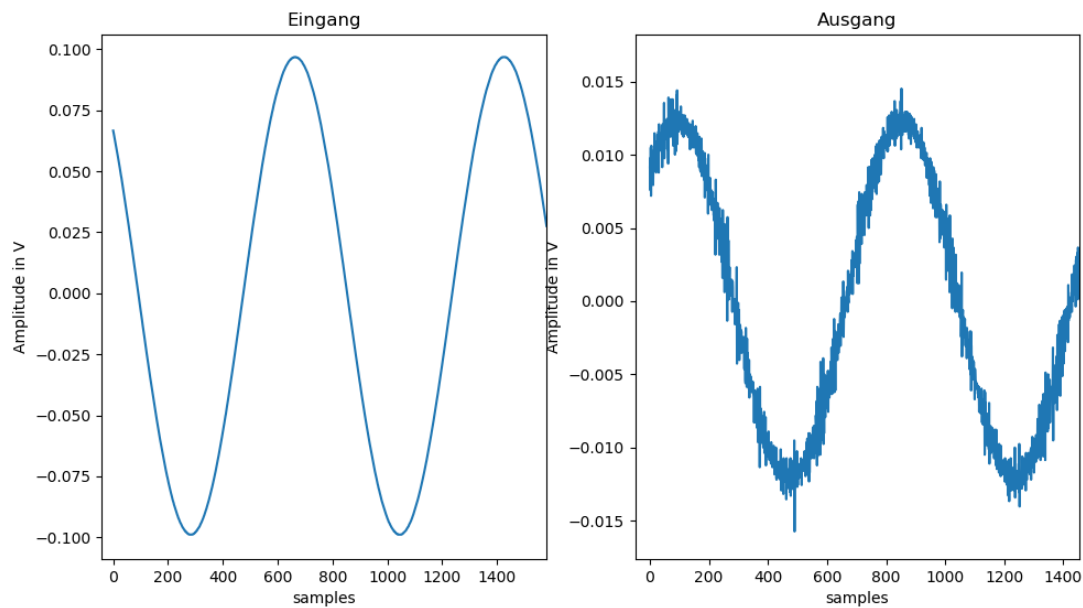
$$\phi = ((191 - 382)/764) * 360 = -90^\circ$$



## Probleme

### Rauschen am Ausgang

Trotz eines reinen Sinus am Eingang wird das Ausgangssignal sehr verrauscht. Wie in folgender Abbildung zu sehen. Das Rauschen hat zur Folge, dass die Nulldurchgänge nicht zuverlässig detektiert werden können.



Ein Auszug aus dem Python Variablenraum zeigt wie im Ausgangssignal innerhalb weniger sample mehrere Nulldurchgänge stattfinden. Dies macht eine Detektion und Vergleich der Nullstellen nach dem oben beschriebenen Schema unmöglich bzw. unzuverlässig.

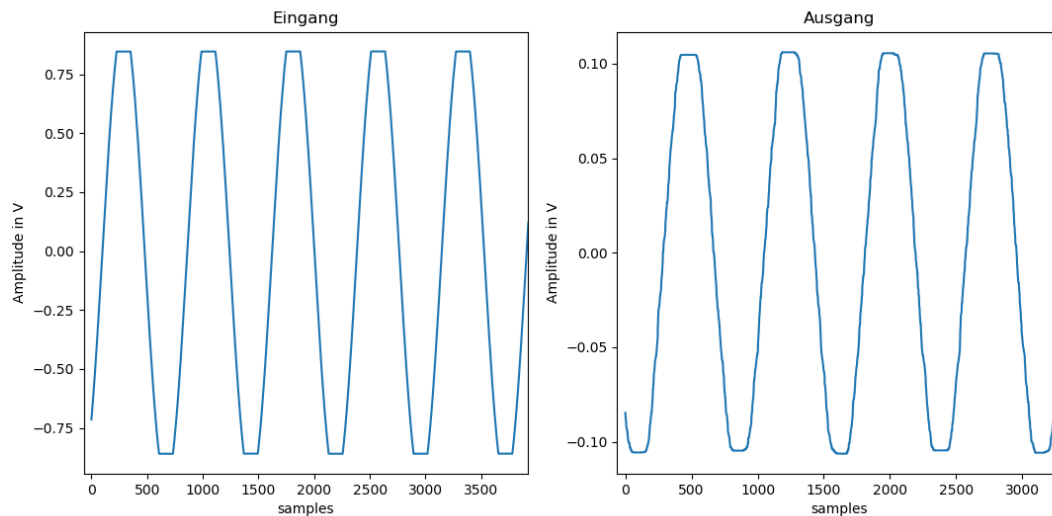
277	0.00146484
278	0.000854492
279	-0.000732422
280	0.00134277
281	0.000854492
282	-0.000488281
283	0.00012207
284	0.00012207
285	0.00012207
286	-0.000366211

Um den Effekt des Rauschens zu beseitigen können die Rohdaten mit Hilfe von Python aufbereitet werden. Eine Möglichkeit ist die lineare Regression. Dies hat jedoch zur Folge, dass bei hohen Frequenzen das Ausgangssignal stark im Verlauf verfälscht wird und eine Weitere Auswertung der Daten nicht möglich ist.

Die zweite Möglichkeit ist ein Moving Average Filter womit die Werte geglättet werden. Hierdurch werden aber die ersten Messwerte unbrauchbar. Dies stellt jedoch kein Problem dar, denn werden sowohl vom Eingang als auch vom Ausgang die erste Werte ausgeschnitten, hat dieser Fehler keinen weiteren Einfluss auf die Auswertung.

Des Weiteren sollten die Messspitzen auf das Tastverhältnis von 1:1 eingestellt werden um den oben genannten Effekt weiter zu verringern.

In der folgenden Abbildung ist der Eingang und Ausgang nach Aufbereitung der Daten und einem Tastverhältnis von 1:1 zu sehen. Es ist eine deutliche Verbesserung zu erkennen.



## Fazit

---

Mit den oben beschriebenen Skripten wurde im Rahmen des Projekts "Projekts Analog System Lab Kit Pro Texas " drei Experimente gemessen und ausgewertet. Das grundsätzliche Übertragungsverhalten wurde ersichtlich jedoch kam es zum Teil zu starken Abweichungen. Zu nennen wäre da die Amplitudenmessung, die auf 20dB begrenzt war.

Bei der Phasenmessung kam es innerhalb einer Messreihe zu einzelnen sporadischen Fehlmessungen.

Aus den genannten Gründen ist eine exakte Messung mittels Red Pitaya nicht möglich. Jedoch für den Erhalt des groben Übertragungsverhalten ist die Messautomatisierung gut geeignet.

## Quellen

---

<https://redpitaya.readthedocs.io/en/latest/appsFeatures/remoteControl/remoteControl.html>