



TU Berlin Fakultät IV  
Institut für Telekommunikationssysteme  
Fachgebiet Nachrichtenübertragung  
Praktikum Nachrichtenübertragung

# **Praktikum 06**

## **Digitale bertragungstechnik: Digitale Empfänger**

Dirk Babendererde (321 836)  
Thomas Kapa (325 219)

2. Juli 2012

Gruppe:

Betreuer: Lieven Lange

## Inhaltsverzeichnis

<b>1</b>	<b>Vorbereitung</b>	<b>1</b>
<b>2</b>	<b>Labordurchführung</b>	<b>1</b>
2.1	Encoderkennlinie . . . . .	1
2.2	Quantisierungsfehler, Kodieren, Dekodieren . . . . .	1
<b>3</b>	<b>Auswertung &amp; Theorie</b>	<b>2</b>
3.1	Encoderkennlinie . . . . .	2
3.2	Quantisierungsfehler . . . . .	2

# 1 Vorbereitung

Abb. 1: Testkennlinie des PCM-Analyse Scriptes

In Abbildung 1 ist die PCM-Encoder-Kennlinie einer Messreihe zu sehen, die mit einem nicht linearen Quantisierer quantisiert wurde. Dabei werden die Werte um 0 herum wesentlich genauer und die Werte im größeren Spannungsbereich seltener abgetastet. Dies kann von Vorteil sein, wenn die Werte um 0 häufiger auftreten und somit für die Messung interessanter sind.

TODO:

Blockschaltbild? Erklärung?

## 2 Labordurchführung

### 2.1 Encoderkennlinie

Es soll die PCM-Encoder-Kennlinie aufgenommen werden. Dazu wird das PCM-Encoder-Modul des ETT101 genutzt. Zunächst wird als Clocksignal an den Eingang CLK das 100 kHz DIGITAL Signal des Master Signals-Modul gelegt. Anschließend wird als Eingang in INPUT 1 ein symmetrisches Dreiecksignal mit einer Amplitude von 2,5 Volt angelegt (2,5 Volt, damit für über 2 und unter -2 Volt die Codewörter 1111 1111 und 0000 0000 ausgegeben werden) und der Schalter auf PCM gestellt. Die beiden Ausgänge FS (Rahmensignal) und PCM (Pulsecode) werden auf die beiden Eingänge des Addierers gegeben. Da beide Signale 5 Volt high und 0 Volt low ausgeben, wird die Verstärkung für das PCM-Signal auf 0 gestellt und die Verstärkung für das Rahmensignal auf 8/5 gestellt, um die Anforderungen aus der Aufgabenstellung zu erfüllen. Mit dem Picoscope werden die Summe aus Rahmensignal und Pulscodesignal und eine steigende Flanke des Zeitignals gemessen und angezeigt.

### 2.2 Quantisierungsfehler, Kodieren, Dekodieren

Um den Quantisierungsfehler bestimmen zu können, werden das Eingangssignal und das dekodierte Signal benötigt. Zur Dekodierung wird das PCM Decoder-Modul genutzt. Die Eingänge FS, PCM DATA und CLK werden mit ihren jeweiligen Gegenstücken des PCM Encoder-Moduls verbunden. Anschließend wird das Signal am OUTPUT 1 zusammen mit dem Eingangssignal dem Picoscope zugeführt.

## 3 Auswertung & Theorie

### 3.1 Encoderkennlinie

Abb. 2: PCM-Encoderkennlinie der steigenden Dreiecksflanke

In Abbildung 2 erkennt man im Gegensatz zu der Abbildung 1 in der Vorbereitungsaufgabe eine lineare Encoder-Kennlinie. Die im Labor mit dem PCM-Modul aufgenommenen Messwerte wurden, also mit einem linearen Quantisierer aufgenommen.

### 3.2 Quantisierungsfehler

Das Signal besitzt in dieser Form noch einen Mittelwert (Offset), welcher mit der Funktion `mean` in Matlab ermittelt werden kann und vom Signal abgezogen wird. Da weiterhin das Signal durch den Kodier- und Dekodiervorgang eine Dämpfung und einen Delay erfährt, wird das dekodierte Signal mit Hilfe von Matlab verstärkt und mit der Kreuzkorrelation um die erfahrene Verzögerung verschoben. Das Ergebnis ist in Abb. 4 zu sehen.

Abb. 3: 100 kHz Sinus noch verschoben

Abb. 4: 100 kHz Sinus angepasst

Anschließend können Sendesignal und dekodiertes Signal voneinander subtrahiert werden und man erhält den Quantisierungsfehler. Die Quantisierungsfehler sind in den Abbildungen 5, 6, 7 und 8 zu sehen.

Abb. 5: Quantisierungsfehler 100 kHz Sinus

Abb. 6: Quantisierungsfehler 100 kHz Dreieck

Abb. 7: Quantisierungsfehler 8 kHz Sinus

Abb. 8: Quantisierungsfehler 8 kHz Dreieck

Es ist gut zu erkennen, dass wie zu erwarten für die Abtastung mit 8 kHz ein von der Amplitude her wesentlich größerer Fehler auftritt als bei 100 kHz. Der Quantisierungsfehler für kleinere Abtastfrequenzen ist größer, da ein Wert des S&H Gliedes länger gehalten wird und somit über die Zeit mehr vom Originalsignal abweicht.

Mit dem Matlabbefehl `hist` wird der Quantisierungsfehler als Häufigkeitsverteilung geplottet. Die Ergebnisse sind in Abbildung 9, 10, 11 und 12 zu sehen.

Abb. 9: 100 kHz Dreieck Quantisierungsfehler Histplot

Abb. 10: 100 kHz Sinus Quantisierungsfehler Histplot

Abb. 11: 8 kHz Dreieck Quantisierungsfehler Histplot

Abb. 12: 8 kHz Sinus Quantisierungsfehler Histplot

Der Quantisierungsfehler, bzw. das Quantisierungsrauschen ist wie erwartet gaußverteilt. In den Histplots ist wieder zu erkennen, dass bei der Abtastung mit 100 kHz die Amplituden des Quantisierungsfehlers geringer ist als bei der mit 8 kHz. Des weiteren sind die Histplots bei 100 kHz schmaler, was bedeutet, dass die kleinen Fehler um 0V wesentlich häufiger auftreteten, was durch die geringere Breite der s&H Stufen zu erklären ist.

Die Definition des Leistungsdichtspektrums ist in Formel 1 beschrieben:

$$S_{uu}(\omega) := \mathcal{F}\{r_{uu}\}(\tau) = \int_{-\infty}^{\infty} r_{uu}(t) e^{-j\omega t} dt \quad (1)$$

Für das Leistungsdichtespektrum wird die Autokorrelation des Quantisierungsfehlers mit dem Befehl `xcorr` aus Matlab gebildet und das Ergebnis dann fouriertransformiert (Abb. 13, 14, 15 und 16).

Abb. 13: 100 kHz Dreieck Quantisierungsfehler LDS

Abb. 14: 100 kHz Sinus Quantisierungsfehler LDS

Abb. 15: 8 kHz Dreieck Quantisierungsfehler LDS

Abb. 16: 8 kHz Sinus Quantisierungsfehler LDS

An den Spektren kann man erkennen, dass sich die Amplituden für die Abtastung mit 8 kHz deutlich höher sind, da der Quantisierungsfehler für eine kleinere Abtastfrequenz größer ist. Aus größeren Abtastfehlern ergibt sich, dass die Leistung des Quantisierungsrauschens, vor allem in den Hauptfrequenzen (200 bzw. 70 kHz), bei 8 kHz deutlich höher ist. Der SNR ist bei kleineren Abtastraten also schlechter.