



TU Berlin Fakultät IV
Institut für Telekommunikationssysteme
Fachgebiet Nachrichtenübertragung
Praktikum Nachrichtenübertragung

Praktikum 05

Pulsamplitudenmodulation und nichtideale Abtastung

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

26. Juni 2012

Gruppe:

Betreuer: Lieven Lange

Inhaltsverzeichnis

1	Vorbereitung	1
2	Labordurchführung	1
2.1	Encoderkennlinie	1
2.2	Quantisierungsfehler, Kodieren, Dekodieren	2
3	Auswertung & Theorie	2
3.1	Encoderkennlinie	2
3.2	Quantisierungsfehler	2

1 Vorbereitung

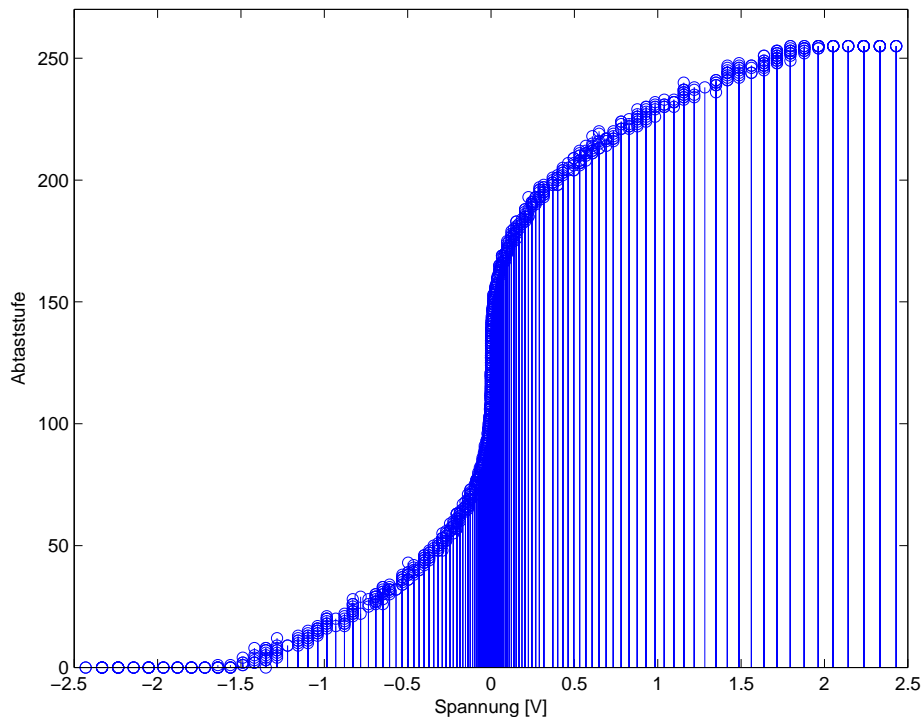


Abb. 1: Testkennlinie des PCM-Analyse Scriptes

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 Pulscode-Signal 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 kann das Signal am OUTPUT 1 zusammen mit dem Eingangssignal dem Picoscope zugeführt.

3 Auswertung & Theorie

TODO:

Thommy: Stimmt es, dass bei 8kHz in der Matlabdatei $f_T = 8000$; stehen muss? (so sind die Plots jetzt gemacht)

3.1 Encoderkennlinie

dsf

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. 3 zu sehen.

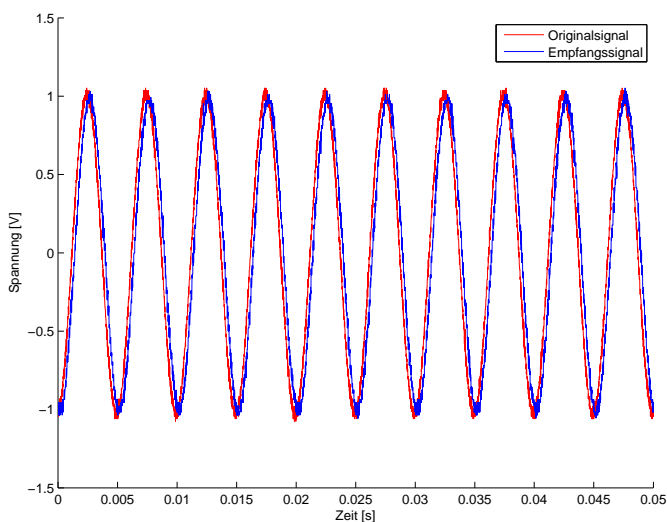


Abb. 2: 100 kHz Sinus noch verschoben

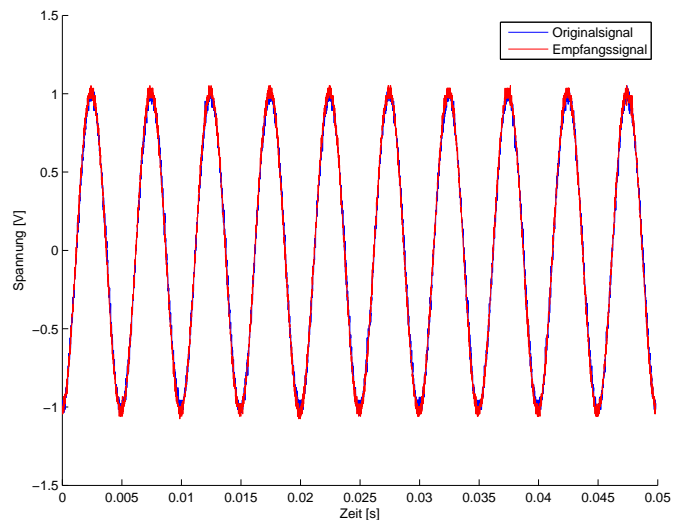


Abb. 3: 100 kHz Sinus angepasst

Anschließend können Sendesignal und dekodiertes Signal voneinander subtrahiert werden und man erhält den Quantisierungsfehler.

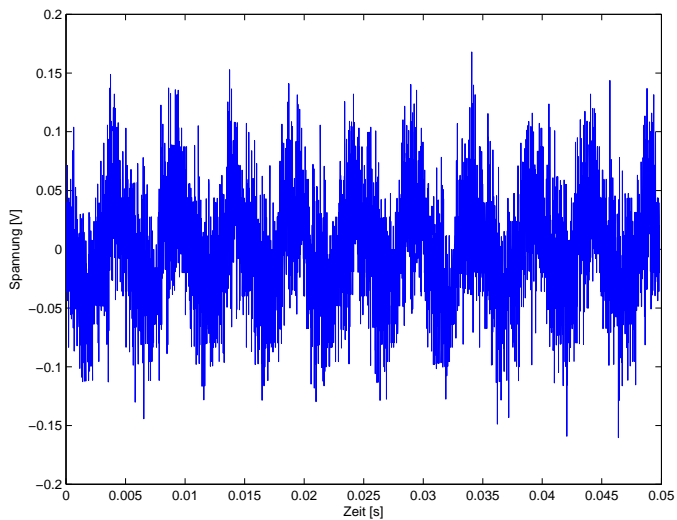


Abb. 4: Quantisierungsfehler 100 kHz Sinus

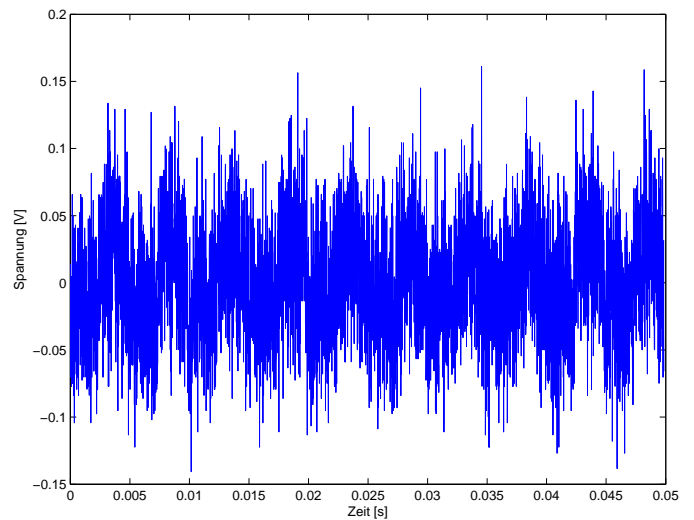


Abb. 5: Quantisierungsfehler 100 kHz Dreieck

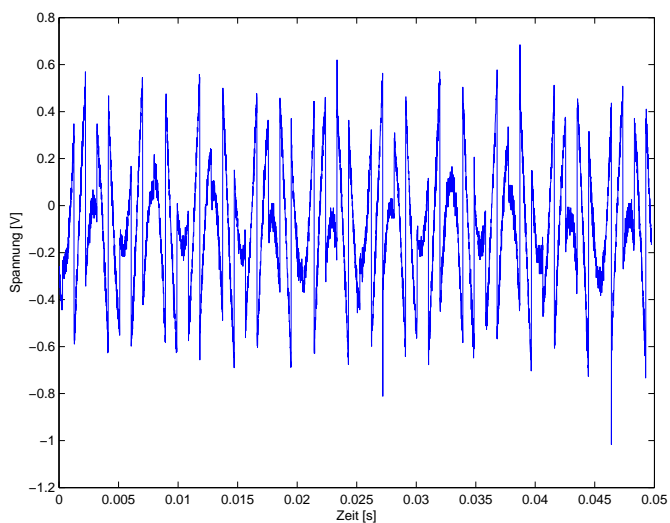


Abb. 6: Quantisierungsfehler 8 kHz Sinus

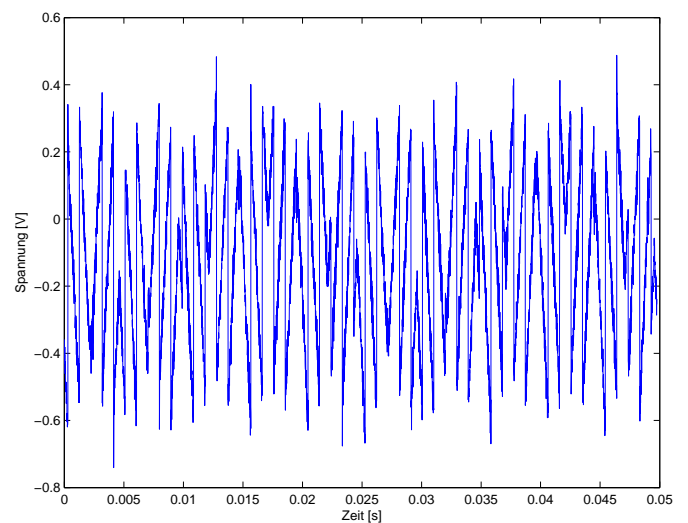


Abb. 7: Quantisierungsfehler 8 kHz Dreieck

Mit dem Matlabbefehl `hist` wird der Quantisierungsfehler als Häufigkeitsverteilung geplottet. Die Ergebnisse sind in Abbildung 8, 9, 10 und 11 zu sehen.

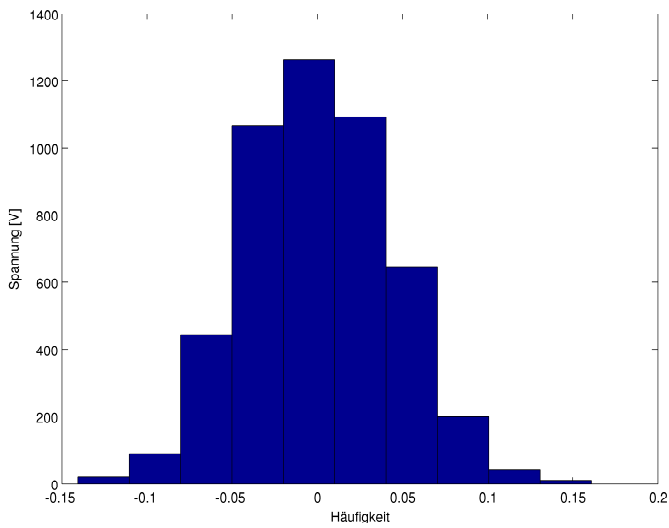


Abb. 8: 100 kHz Dreieck Quantisierungsfehler Histplot

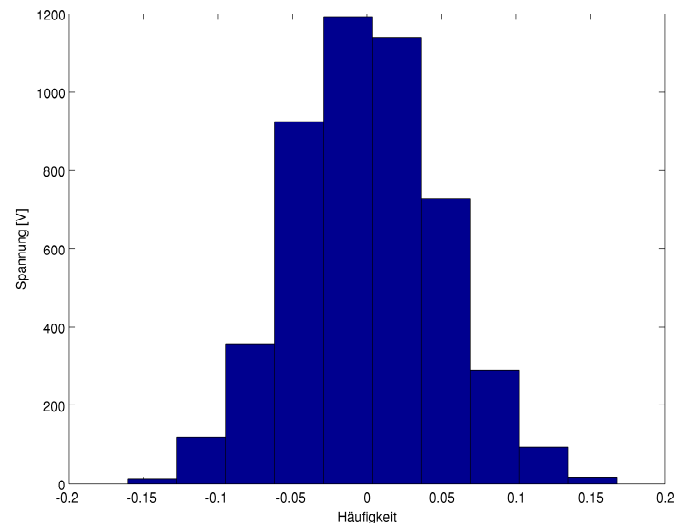


Abb. 9: 100 kHz Sinus Quantisierungsfehler Histplot

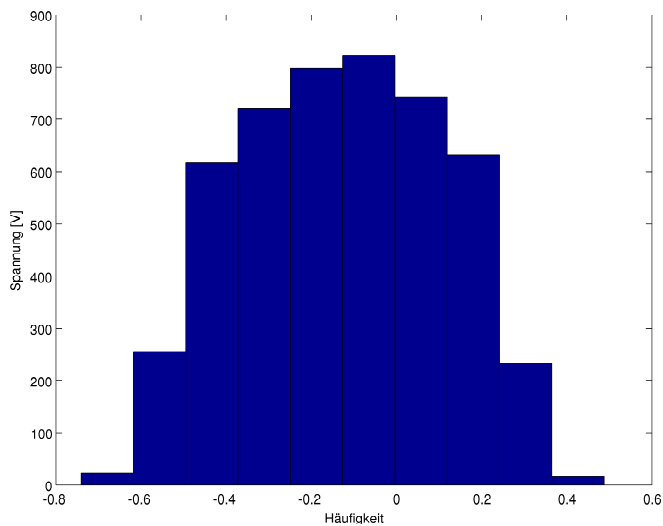


Abb. 10: 8 kHz Dreieck Quantisierungsfehler Histplot

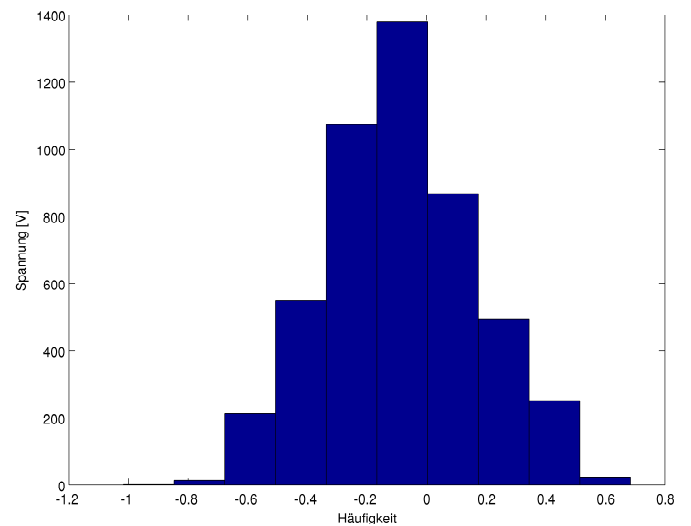


Abb. 11: 8 kHz Sinus Quantisierungsfehler Histplot

Für das Leistungsdichtespektrum wird die Autokorrelation des Quantisierungsfehlers mit dem Befehl `xcorr` aus matlab gebildet und das Ergebnis dann Fouriertransformiert (Abb. 12 und 13).

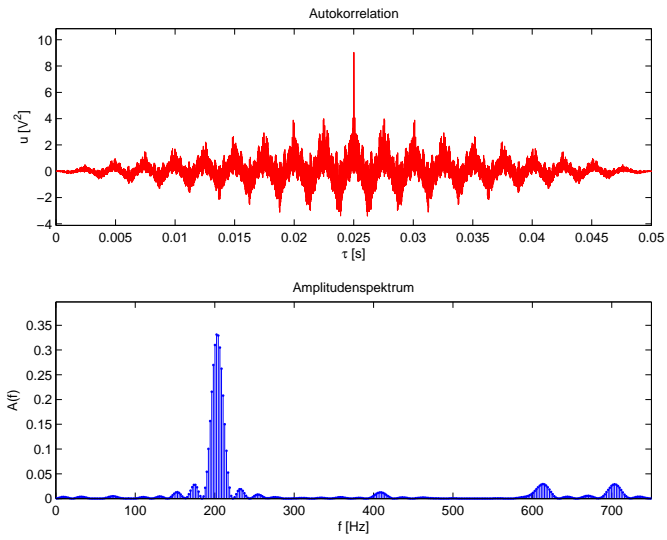


Abb. 12: 100 kHz Dreieck Quantisierungsfehler LDS

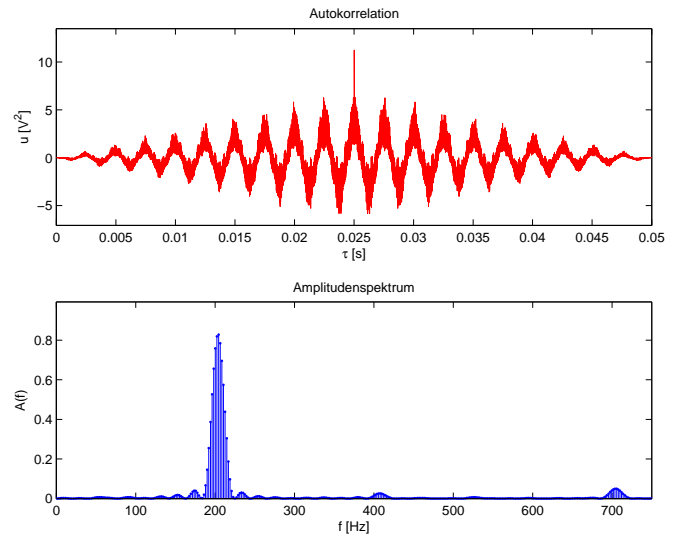


Abb. 13: 100 kHz Sinus Quantisierungsfehler LDS

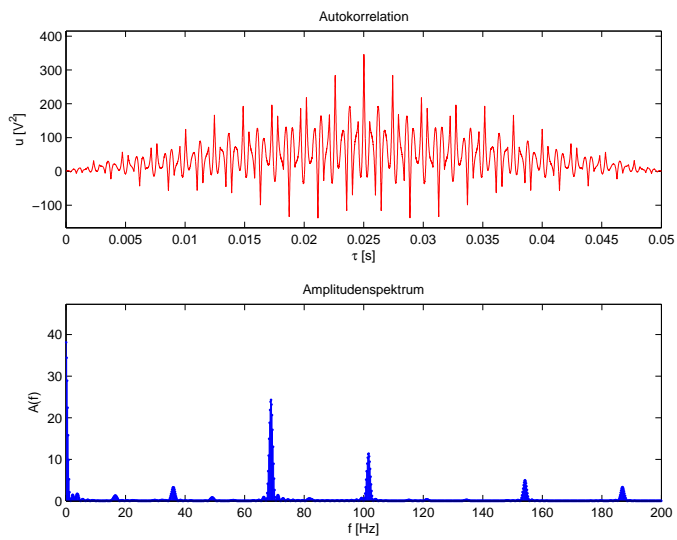


Abb. 14: 8 kHz Dreieck Quantisierungsfehler LDS

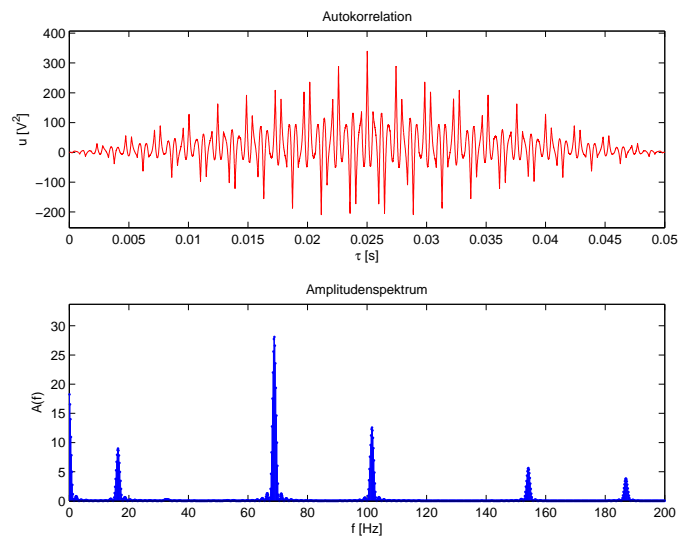


Abb. 15: 8 kHz Sinus Quantisierungsfehler LDS