

Proseminar Computer und Musik –
Analog-Digital-Wandlung

Sebastian Reuter

9. Januar 2018

Inhaltsverzeichnis

1	Einführung	2
2	Fundamentale Prinzipien	2
2.1	Problemstellung	2
2.2	Sampling	3
2.3	Aliasing	3
2.4	Quantisierung	5
2.5	Kanal-Multiplexing	7
2.6	Bitrate	8
3	Pulse-Code-Modulation-Transmitter	9
4	Zusammenfassung	11

1 Einführung

Wir leben in einer Zeit, in der Computer und von Mikroprozessoren gesteuerte Geräte in immer mehr Bereiche eindringen, die bis dahin meist ohne Computer ausgekommen sind. Häufig bedeutet das, dass Computer mit Einflüssen und Daten umgehen müssen, die zunächst nicht digital erfasst werden können. Dazu zählen etwa Musik- und andere Tonaufnahmen, aber auch Sensordaten, analoge Bilddaten und vieles mehr. Diese Daten liegen meist nur in analoger Form vor, weshalb sie vor der Verarbeitung mit dem Computer zunächst digitalisiert werden müssen. Diese Aufgabe übernimmt ein Analog-Digital-Wandler, mit dessen Grundlagen wir uns in dieser Arbeit beschäftigen wollen.

Denn obwohl erste theoretische Ansätze für die Umsetzung eines solchen Bauteils schon seit 1938 existieren [8], sind Analog-Digital-Wandler in der heutigen Zeit so wichtig wie nie zuvor: Professionelle Audio-Anwendungen, wie sie bei Studioaufnahmen und Live-Konzerten vorkommen, aber auch alltägliche Dinge, wie Telefonie und Fernseh-Übertragungen wären ohne Analog-Digital-Wandlung heute kaum noch vorstellbar.

2 Fundamentale Prinzipien

Bevor wir uns aber der genauen Umsetzung, beziehungsweise der Implementierung von Analog-Digital-Wandlern (kurz ADC, von engl. „analogue-digital-converter“) im Einzelfall widmen können, müssen wir uns zunächst mit der vorhandenen Problemstellung und einigen fundamentalen Prinzipien zur Lösung dieser zuwenden.

2.1 Problemstellung

Das Problem, welchem wir uns mit dieser Arbeit widmen wollen, besteht darin, ein analoges Eingangssignal unter Verwendung eines elektrischen Bauteils (dem Analog-Digital-Wandler) in ein digitales Ausgangssignal zu überführen. Dazu müssen wir jedoch erst genauer charakterisieren, was wir unter einem analogen beziehungsweise digitalen Signal verstehen.

Mit einem analogen Signal bezeichnen wir im Folgenden ein elektrisches Signal, welches sowohl einen zeitkontinuierlichen, als auch einen spannungskontinuierlichen Wertebereich annehmen kann. Andererseits ist mit einem digitalen Signal ebenfalls ein elektrisches Signal gemeint, welches nur diskrete Zeit- und Spannungswerte darstellt. [5] Dabei ist anzumerken, dass digitale Signale meist nicht als zeitdiskrete Signale vorliegen, sondern es sich vielmehr um zeitkontinuierliche Signale handelt, die jedoch stets abschnittsweise einen konstanten, diskreten (Spannungs-)Wert annehmen.

Da zum Beispiel Computer und andere mikroprozessorgesteuerte Geräte nicht (beziehungsweise nur schlecht) mit kontinuierlichen Wertebereichen umgehen können, wird es also unser hauptsächliches Problem sein, ein analoges und damit zeit- und spannungskontinuierliches Signal in ein digitales, diskretes Signal zu überführen. Dabei wollen wir darauf achten, dass unser überführtes (auch: gewandeltes) Signal nur einem geringen Informationsverlust unterliegt, dass man also das ursprüngliche analoge Signal wieder eindeutig aus den digitalen Daten rekonstruieren kann. Im Bereich Audio-Digitalisierung entspricht dies eben einem geringen Qualitätsverlust im Vergleich zum ursprünglich analogen Signal. Dazu werden im Wesentlichen die im Folgenden diskutierten Methoden des Samplings und der Quantisierung zum Einsatz kommen.

2.2 Sampling

Mit Hilfe von Sampling wollen wir nun zuerst die *Zeitkontinuität* des analogen Eingangssignals behandeln.

Sampling bedeutet dabei das regelmäßige, gleichförmige Abtasten (also Messen) der Spannung des analogen Eingangssignal mit einer festgelegten Sampling-Frequenz f_s . Da dies vor dem Hintergrund einer späteren *eindeutigen* Rekonstruktion des analogen Eingangssignal aus den digitalisierten Daten geschehen soll, ist darauf zu achten, dass eine ausreichende Anzahl an Samples (Abtastpunkten) erstellt wird, um eben jene eindeutige Rekonstruktion im Nachhinein zu ermöglichen. Aus diesem Grund sollte die Sampling-Frequenz f_s nicht beliebig gewählt werden, sondern entsprechend des sogenannten Shannon-Nyquist-Theorems (nach Claude Shannon (1928) und Harry Nyquist (1948)) festgelegt werden.

Theorem (Shannon-Nyquist-Theorem bzw. WKS-Theorem). *Die Sampling-Frequenz f_s sollte mindestens doppelt so groß sein, wie die höchste Frequenz f_a , die im Signal vorkommt [3], sodass also gilt:*

$$f_s \geq 2 \cdot f_a$$

Die Einhaltung dieses Theorems garantiert nun, dass unser digitalisiertes Signal nachher genug Stützstellen enthält, um das analoge Signal wieder eindeutig zu rekonstruieren. [6]

2.3 Aliasing

Wird das im vorherigen Abschnitt erwähnte Shannon-Nyquist-Theorem bei der Wahl der Sampling-Frequenz f_s nicht hinreichend beachtet, so können Nebeneffekte auftreten, die die eindeutige Rekonstruktion des analogen Signals aus den

digitalisierten Daten unmöglich macht. Diese Nebeneffekte werden *Alias-Effekte* oder schlicht *Aliasing* genannt und lassen sich am besten an einem Beispiel verdeutlichen:

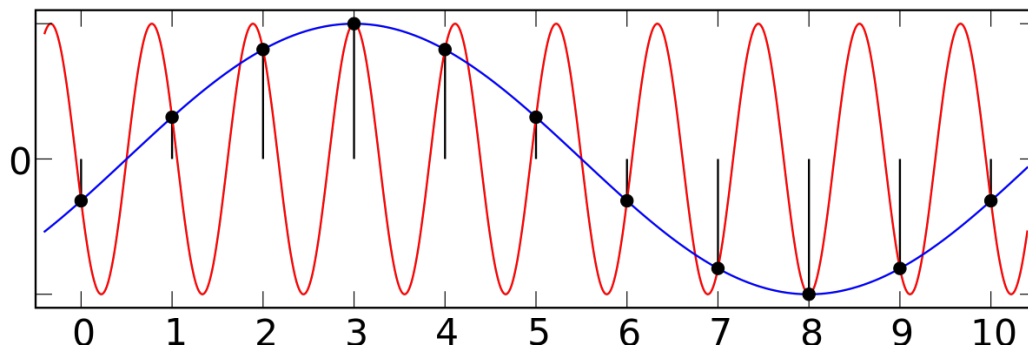


Abb. 1: Alias-Effekt.

Quelle: <https://commons.wikimedia.org/wiki/File:AliasingSines.svg>

In Abb. 1 ist der Spannungsverlauf eines analogen Eingangssignals (rot) in Abhängigkeit der Zeit abgetragen, welches um den Spannungswert 0V oszilliert. Bei diesem analogen Eingangssignal handelt es sich offensichtlich um eine gleichförmige Sinus-Schwingung mit einer konstanten Frequenz f_a . Die schwarzen Punkte auf dem Signalverlauf dieses Eingangssignals sollen die Sampling-Zeitpunkte sowie die jeweils gemessenen Spannungswerte darstellen. Wie man leicht erkennen kann, sind diese Sampling-Zeitpunkte in regelmäßigen Zeitabständen hinsichtlich einer Sampling-Frequenz f_s anzutreffen. Wie man nun jedoch durch genaues Hinsehen leicht bemerkt, gilt hier nicht $f_s \geq 2 \cdot f_a$, wie vom Shannon-Nyquist-Theorem gefordert, sondern es gilt vielmehr $f_s \approx f_a$.

Wir stellen uns nun vor, dass wir das Sampling abgeschlossen haben und uns statt des analogen Eingangssignals nur noch die Samples (also die schwarzen Punkte) zur Verfügung stehen. Wie in Abb. 1 angedeutet, gibt es nun zwei Möglichkeiten, aus den gegebenen Samples wieder unser analoges Eingangssignal zu rekonstruieren: Sowohl die rote, als auch die blaue Kurve stellen mögliche Rekonstruktionen dar. Da unser Eingangssignal jedoch durch die rote Kurve beschrieben wurde, handelt es sich bei der blauen Kurve um einen Nebeneffekt, der durch eine unzureichende Sampling-Frequenz verursacht wurde. Wir haben somit die Eindeutigkeit in den digitalisierten Daten verloren.

Wie das gerade diskutierte Beispiel zeigt ist es also wichtig, die Sampling-Frequenz, mit der wir die Analog-Digital-Wandlung durchführen, an das entsprechende Eingangssignal anzupassen. Unser Ziel ist es jedoch - gerade im Audio-Bereich - nicht für jedes mögliche Eingangssignal einen anderen Analog-Digital-Wandler zu ver-

wenden; wir wollen vielmehr *einen* Analog-Digital-Wandler für (in einem gewissen Spektrum) jedes mögliche Eingangssignal verwenden. Als Folge dessen müssen wir, um dem Shannon-Nyquist-Theorem gerecht zu werden, einen Tiefpass-Filter (LPF, von engl. Low Pass Filter) verwenden, der vor dem Sampling sicherstellt, dass jegliche Frequenzen $f > \frac{f_s}{2}$ aus dem Eingangssignal herausgefiltert werden, sodass beim anschließenden Sampling keine Alias-Effekte auftreten. [2]

2.4 Quantisierung

Nachdem wir in den vorherigen zwei Abschnitten die Zeitkontinuität unseres analogen Eingangssignals behandelt haben, bleibt nun noch die Spannungsbeziehungsweise Wertekontinuität, mit der wir uns beschäftigen müssen: Die Spannungswerte der einzelnen Samples (Abtastdaten), die wir durch das Sampling erhalten, liegen typischerweise in einem abgeschlossenen, aber kontinuierlichen Spannungsintervall $[V_{min}, V_{max}]$. Unser Ziel wird es sein, ebenjenes kontinuierliche Spannungsintervall auf ein diskretes Spannungsintervall abzubilden und dabei die jeweiligen Spannungsstufen durch ein Binärwort fester Länge n_s zu kodieren. Da wir mit einem Binärwort der Länge n_s genau 2^{n_s} Spannungswerte kodieren können, legt die Wahl von n_s den Detailgrad der Quantisierung, beziehungsweise die Auflösung unseres Analog-Digital-Wandlers fest. [2] Die Zahl n_s ist also eine charakteristische Größe unseres Analog-Digital-Wandlers und wird daher auch oftmals als *Sample-Bit-Depth* beziehungsweise kurz *Sample-Depth* bezeichnet. [7] Da die Wahl der Sample-Depth und damit auch der Auflösung eines Analog-Digital-Wandlers maßgeblich die Menge der bei der Analog-Digital-Wandlung anfallenden Daten bestimmt (siehe dazu Unterabschnitt 2.6), muss zur Wahl der Sample-Depth ein Kompromiss aus Qualität der Quantisierung und anfallender Datenmenge gewählt werden. Eine kleine Sample-Depth entspricht dabei einer geringen Datenmenge bei schlechter Qualität, während eine große Sample-Depth eher einer guten Qualität bei jedoch größerer Datenmenge entspricht.

Nachdem also eine geeignete Sample-Depth festgelegt wurde, mit der unser Analog-Digital-Wandler arbeiten soll, muss nun noch die Art der Binärkodierung, sowie die zugrundeliegende Transferfunktion gewählt werden.

Als Kodierungsarten eignen sich etwa eine ‚Standard‘-Binärkodierung mit Offset, eine Darstellung als Einer-/Zweierkomplement oder auch eine Kodierung im Gray-Code (mit Offset) [2], wobei sich Unterschiede in den Kodierungsmethoden insbesondere dann deutlich abzeichnen, falls $V_{min} < 0$ gilt. (Mit Offset ist bei den oben genannten Kodierungsmöglichkeiten gemeint, dass V_{min} mit der 0 der jeweiligen Kodierungsmethode identifiziert wird.)

Die Transferfunktion gibt dann an, in welcher Weise das kontinuierliche Intervall $[V_{min}, V_{max}]$ auf die $2^{\text{Sample-Depth}} = 2^{n_s}$ diskreten Quantisierungsstufen abgebildet

wird. Üblich ist beispielsweise eine lineare Transferfunktion, die das gesamte Spannungsintervall in 2^{n_s} Teilintervalle gleicher Länge aufteilt und jedes dieser Teilintervalle mit einem Binärkode identifiziert [2] (siehe Abb. 2). Jedoch sind auch andere Transferfunktionen möglich und teilweise sinnvoll, wie wir in Abschnitt 3 noch sehen werden.

Für einen 3-Bit ADC mit ‚Standard‘-Binärcodierung mit Offset wäre zum Beispiel die Transferfunktion in Abb. 2 denkbar:

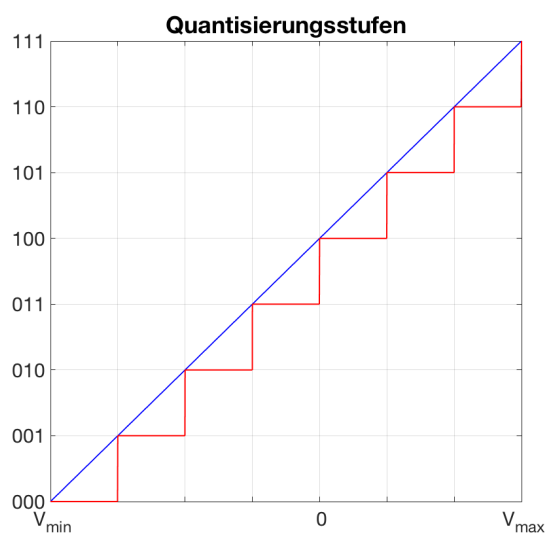


Abb. 2: Transferfunktion.

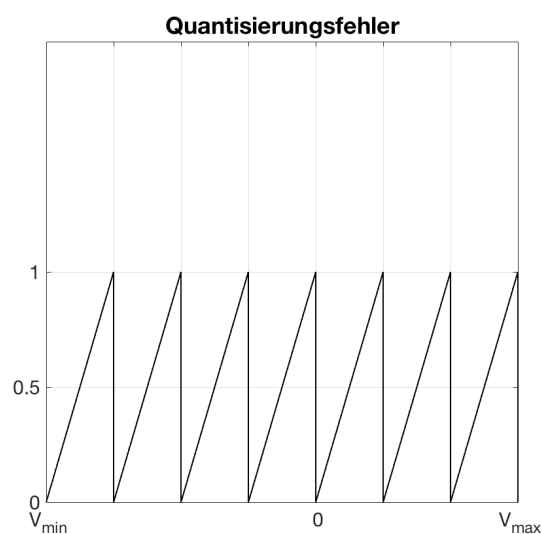


Abb. 3: Quantisierungsfehler zu Abb. 2.

Dort sieht man auf der Abszisse das Spannungsintervall $[V_{min}, V_{max}]$ und auf der Ordinate die entsprechenden Binärcodierungen in ‚Standard‘-Binärcodierung mit Offset abgetragen. Die *blaue* Kurve stellt den idealen Verlauf einer linearen Transferfunktion dar, mit der wir jeden Zwischenspannungswert auf eine eindeutige Binärcodierung abbilden würden. Da dadurch jedoch die Kontinuirlichkeit des Spannungsintervalls erhalten bliebe, müssen wir stattdessen die *rote* Treppenfunktion verwenden, um das Spannungsintervall $[V_{min}, V_{max}]$ auf die $2^3 = 8$ zur Verfügung stehenden Binärcodierungen abzubilden. Durch die Verwendung dieser Treppenfunktion ist das Urbild einer Binärcodierung unter der Transferfunktion nicht mehr eindeutig, was zu einem Fehler in den digitalisierten Daten führt. Dieser Fehler ist offensichtlich kurz vor dem Sprung auf die nächste ‚Treppenstufe‘ am

größten und steigt dort auf nahezu 1 LSB¹ an, wie man auch in Abb. 3 erkennen kann. Dieses Problem behandeln wir, indem wir die rote Treppenfunktion leicht verschieben, sodass der Übergang zur nächsten Binärcodierung in der Mitte statt am Rand der jeweiligen Spannungs-Unterintervalle, die durch Aufteilung des Spannungsintervalls $[V_{min}, V_{max}]$ in $2^{n_s} = 2^3 = 8$ gleichgroße Teilintervalle entstehen, durchgeführt wird [3] (vgl. Abb. 4). Dadurch verringert sich der Quantisierungsfehler auf höchstens 0.5 LSB (vgl. Abb. 5).

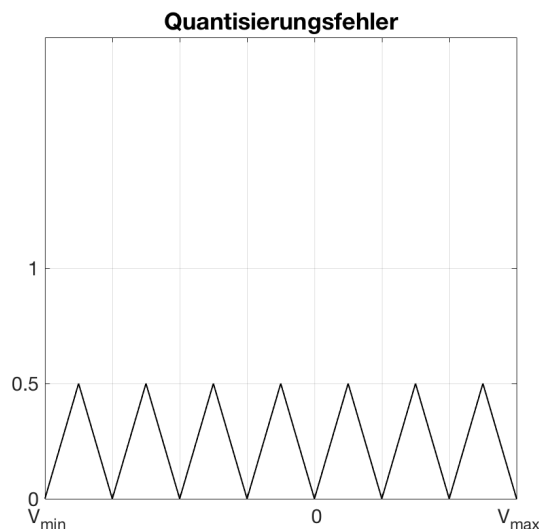
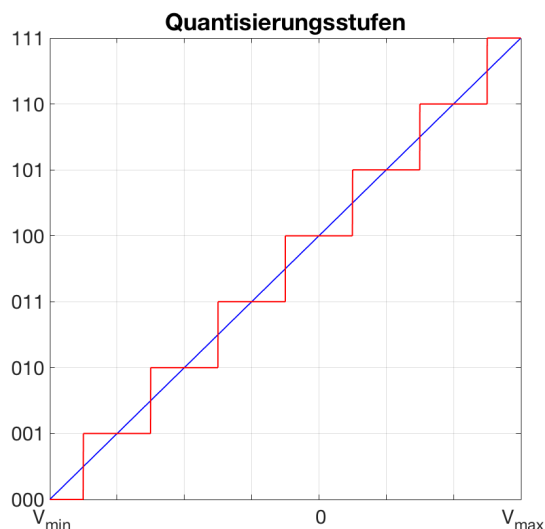


Abb. 4: Transferfunktion mit verringertem Quantisierungsfehler.

Abb. 5: Quantisierungsfehler zu Abb. 4.

2.5 Kanal-Multiplexing

Mit dem Sampling- und dem Quantisierungsprinzip haben wir also unser Hauptproblem gelöst: Wir haben ein zeit- sowie wertekontinuierliches Signal in ein rein diskretes Signal überführen können. Nun wollen wir uns einen weiteren Vorteil der digitalen Signalübertragung zu Nutze machen: Die Möglichkeit des Kanal-Multiplexings.

Der Begriff „Multiplexverfahren“ bezeichnet im Allgemeinen die gleichzeitige Übertragung mehrerer Quellsignale über einen Übertragungskanal. [4] In unserem Fall der digitalen Audio-Übertragung wollen wir damit die gleichzeitige Übertragung

¹LSB $\hat{=}$ Least Significant Bit. Mit 1 LSB ist die Spannungsdifferenz zwischen zwei aufeinanderfolgenden Spannungsstufen gemeint. Es gilt also (bei einer linearen Transferfunktion) $1 \text{ LSB} = \frac{V_{max} - V_{min}}{2^{n_s}}$ [2]

mehrer Audiokanäle über einen einzigen Übertragungskanal (etwa eine elektrische Leitung oder einen Funkkanal) erreichen.

Dazu bietet es sich an, das sogenannte *Time-Division-Multiplexing* [4] zu verwenden, indem wir uns die oft hohe Geschwindigkeit der zur AD-Wandlung verwendeten elektrischen Bauteile zu Nutze machen:

Pro Sample, also pro abgetastetem, quantisiertem Spannungswert, steht uns zur Verarbeitung dessen ein Zeitfenster T_{sample} zur Verfügung, welches durch die verwendete Sampling-Frequenz f_s festgelegt wird. Es gilt dabei der Zusammenhang $T_{\text{sample}} = \frac{1}{f_s}$. Wird durch Verwendung schneller elektrischer Bauteile dieses Zeitfenster nicht vollständig zur Verarbeitung des Samples benötigt – sondern nur ein Bruchteil T_{conv} dieses Zeitfensters, so entsteht eine Leerlaufzeit $T_{\text{idle}} = T_{\text{sample}} - T_{\text{conv}}$, die ungenutzt verstreicht. Statt diese Leerlaufzeit nun tatsächlich vollständig ungenutzt verstreichen zu lassen, ist es der Ansatz des Time-Division-Multiplexings, ebenjene Leerlaufzeit für Umwandlung weiterer Audiokanäle zu verwenden. [1] Statt einem Audiokanal können wir unter Verwendung dieses Prinzips durch eine geschickte Einteilung des verfügbaren Zeitfensters T_{sample} also bis zu n Audiokanäle über einen Übertragungskanal übertragen, wobei für n der Zusammenhang $n = \lfloor \frac{T_{\text{sample}}}{T_{\text{conv}}} \rfloor$ gilt.

2.6 Bitrate

Nachdem wir nun also mit den fundamentalen Prinzipien *Sampling*, *Quantisierung* und *Kanal-Multiplexing* vertraut sind, wollen wir zum Abschluss dieses Abschnitts noch Überlegungen zu der bei der Analog-Digital-Wandlung anfallenden Datenmenge anstellen. Dies soll dabei jedoch unabhängig von der Länge des gewandelten Audiomaterials geschehen, weshalb wir statt der anfallenden Datenmenge die *Bitrate* bei einer Analog-Digital-Umwandlung betrachten wollen. Die Bitrate, gemessen in $\frac{\text{Bit}}{s}$, gibt dabei an, welche Datenmenge pro Zeiteinheit (hier 1 Sekunde) anfällt. Aus diesem Wissen können wir dann ableiten, welche Geschwindigkeitsanforderungen wir an die von uns verwendete Hardware stellen müssen, um zum Beispiel die digitalisierten Daten ohne aufwändiges Zwischenpuffern auf einer Festplatte abspeichern zu können oder etwa um das digitale Signal in Echtzeit übertragen zu können.

Bei den Datenpaketen, die bei einer AD-Wandlung anfallen, handelt es sich offensichtlich genau um die mittels Sampling und Quantisierung erstellten Samples. In einer Sekunde fallen somit $\{f_s\}$ Datenpakete der Größe n_s Bit an. Da diese Datenmenge für jeden gewandelten Audiokanal anfällt, müssen wir das Ergebnis noch mit der Anzahl der vorhandenen Audiokanäle multiplizieren, sodass wir für

die Bitrate des digitalen Audiosignals erhalten [1]:

$$\begin{aligned}\text{Bitrate} &= \text{Sampling-Frequenz} \cdot \text{Sample-Depth} \cdot \text{Kanalanzahl} \\ &= f_s \cdot n_s \cdot \text{Kanalanzahl}\end{aligned}$$

An dieser Formel erkennt man insbesondere, dass eine hohe Qualität der durchgeführten AD-Wandlung – also eine hohe Auflösung und damit große Sample-Depth – stark mit der benötigten Bitrate und damit der bei der Wandlung anfallenden Datenmenge korreliert. Wie bereits erwähnt ist die Sample-Depth eines Analog-Digital-Wandlers somit stets mit Bedacht als Kompromiss zwischen anfallender Datenmenge und Qualität der Wandlung zu wählen.

3 Pulse-Code-Modulation-Transmitter

Im vorausgegangenen Abschnitt wurden die theoretischen Grundlagen eingeführt, die zum Verständnis der Analog-Digital-Wandlung notwendig sind. Im folgenden Abschnitt wollen wir auf Basis dieser Grundlagen nun an einem Fallbeispiel nach H. S. Black und J. O. Edson (siehe dazu [1]) die tatsächliche Realisierung eines Analog-Digital-Wandlers beleuchten.

Der von Edson und Black bereits 1947 beschriebene Analog-Digital-Wandler arbeitet mit dem „*Pulse-Code-Modulation*“-Verfahren (kurz PCM-Verfahren), bei dem die nach den obigen Prinzipien erstellten Samples direkt als Binärworte (ohne etwa Datenkompression anzuwenden) übertragen beziehungsweise gespeichert werden. Der Wandler wurde hauptsächlich für den Bereich der Telekommunikation konstruiert und weicht daher – und wegen seines hohen Alters – etwas von den heutzutage verwendeten Analog-Digital-Wandlern ab: Aufgrund beschränkter technischer Möglichkeiten handelt es sich bei dem von Black und Edson „PCM-Transmitter“ getauften AD-Wandler um einen 5-Bit-ADC mit einer Sampling-Frequenz $f_s = 8000\text{Hz}$ und einer *logarithmischen* Transferfunktion. [1] Zum Vergleich: Üblich sind heutzutage etwa bei CDs eine Sample-Depth von 16 Bit [7] bei einer Sampling-Frequenz von 44.1kHz. Besonders hervorzuheben ist die logarithmische Transferfunktion bei Black und Edsons PCM-Transmitter: Um bei einer vergleichsweise geringen Auflösung des AD-Wandlers (5 Bit Sample-Depth) noch eine akzeptable Sprachqualität zu erzielen, ist die Transferfunktion so gewählt, dass in betragsmäßig kleinen Spannungsbereichen, die leisen Tönen entsprechen, mehr Quantisierungsstufen zur Verfügung stehen, als in betragsmäßig großen Spannungsbereichen. So kann erreicht werden, dass auch flüsternde Stimmen mit einer ausreichenden Genauigkeit dargestellt werden können. [1]

In Abb. 6 ist der von Edson und Black konstruierte AD-Wandler schematisch für einen der acht Audiokanäle dargestellt. Links ist dabei der Eingang für das

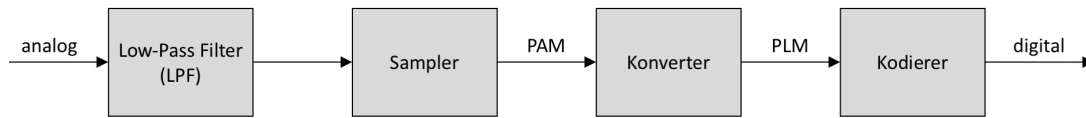


Abb. 6: PCM-Transmitter nach Edson und Black, 1947.

analoge, zu wandelnde Signal und rechts der Ausgang für das digitalisierte Signal dargestellt.

Da der PCM-Transmitter mit einer festen Sampling-Frequenz $f_s = 8000\text{Hz}$ arbeiten soll, wird – um das Shannon-Nyquist-Theorem einhalten zu können – zunächst ein Tiefpass-Filter (LPF) mit einer Trennfrequenz von 4000Hz auf das analoge Eingangssignal angewandt (siehe Unterabschnitt 2.3).

Anschließend durchläuft das analoge Eingangssignal den Sampler, welcher als ein sich schnell öffnendes und schließendes „Gate“ [1] (etwa ein Transistor oder eine Elektronenröhre) realisiert ist. Die gewünschte Frequenz von 8kHz wird dabei durch einen externen Taktgeber, eine sogenannte ‚Clock‘, erzielt, welche mit einer Frequenz von 320kHz getaktet ist und durch 5- und 8-stufige Ringzähler auf die gewünschte Frequenz herunterskaliert werden kann. Als Ergebnis liefert der Sampler dann ein sogenanntes *puls-amplituden*-moduliertes Signal (PAM), welches aus kurzen, gleichlangen Pulsen unterschiedlicher Spannung besteht, wobei die Spannung des Pulses gerade der Spannung des analogen Eingangssignals zum Zeitpunkt des Samplings entspricht.

Dieses *puls-amplituden*-modulierte Signal wird im Anschluss durch den Konverter in ein *puls-längen*-moduliertes Signal (PLM) überführt. Bei jenem PLM-Signal wird die Spannung des Eingangssignals zum Samplingzeitpunkt nun nicht mehr durch die Spannungsamplitude des Pulses, sondern durch die Pulslänge dargestellt. Zur Konvertierung wird dabei ein Kondensator verwendet, der durch den Spannungspuls aufgeladen und dann über einen Widerstand entladen wird. Da die zum Entladen des Kondensators benötigte Zeit logarithmisch von der Ladespannung abhängt [3], erhalten wir so den logarithmischen Zusammenhang, der bei der Transferfunktion gewünscht ist.

Das auf diese Art erzeugte PLM-Signal wird anschließend im Kodierer nochmals durch ein sich öffnendes und schließendes Gate in sehr kurze, aber jeweils gleichlange Pulse („genannt „Pips“ [1],) zerteilt, die dann von einem Binärzähler gezählt werden. Die Länge der kurzen Pulse wird dabei gerade so gewählt, dass jedes beliebige aus einer angelegten Eingangsspannung resultierende PLM-Signal in mindestens 0 und höchstens 31 Pips zerlegt werden kann, sodass der Binärzähler direkt den entsprechenden 5-Bit-Code berechnet.

Abschließend wird das Ergebnis des Binärzählers noch entsprechend verstärkt und dann zu einer Multiplex-Einheit weitergeleitet, die die passende Synchronisierung

mit den Wandlungsergebnissen der anderen sieben Audiokanäle, für die das beschriebene Verfahren analog abläuft, sicherstellt.

Das vollständig digitalisierte Signal kann dann, etwa über eine Telefonleitung, zu einem sogenannten PCM-Receiver übertragen werden, welcher dual funktioniert und die beschriebenen Schritte (fast) genau in umgekehrter Reihenfolge ausführt, um das digitale Signal zurück in ein analoges Signal zu überführen. [1]

4 Zusammenfassung

Wir haben gesehen, dass zur Umsetzung eines Analog-Digital-Wandlers hauptsächlich zwei grundsätzliche Prinzipien angewandt werden müssen: Mit dem Sampling-Prinzip (in Verbindung mit dem Shannon-Nyquist-Theorem) wird die Zeitkontinuität eines analogen Eingangssignals behandelt, während wir durch Quantisierung die Spannungs-, beziehungsweise Wertekontinuität angehen können. Mithilfe beider Prinzipien können wir also das in zwei Arten kontinuierliche analoge Eingangssignal ohne schwerwiegende Informationsverluste in ein vollkommen diskretes, digitales Ausgangssignal überführen. Die Anwendung dieser Prinzipien haben wir dann beispielhaft am PCM-Transmitter nach H. S. Black und J. O. Edson betrachtet, wobei wir natürlich auch andere Beispiele (wie etwa einen Tracking-Wandler, SAR-Wandler, usw.) hätten wählen können.

Die auf diese Weise digitalisierten (Audio-)Daten können nun vielfältig weiterverarbeitet werden: Wir können die Daten speichern, duplizieren, manipulieren, übertragen und gegebenenfalls – bei Audiodaten spätestens zur Wiedergabe – mit einem Digital-Analog-Wandler (DA-Wandler) zurück in ein analoges Signal überführen.

Literatur

- [1] H. S. Black and J. O. Edson. Pulse code modulation. *Transactions of the American Institute of Electrical Engineers*, 66(1):895–899, 1947.
- [2] W. Kester, D. Sheingold, and J. Bryant. Fundamentals of sampled data systems. *Data Conversion Handbook*, 2004.
- [3] Prof. Dr.-Ing. Stefan Kowalewski. *Einführung in die Technische Informatik - Kapitel 14 (WS16/17)*. 2017.
- [4] Jens Ohm and Hans Dieter Lüke. *Signalübertragung: Grundlagen der digitalen und analogen Nachrichtenübertragungssysteme*. Springer-verlag, 2014.
- [5] Johann Siegl and Edgar Zoicher. *Schaltungstechnik-analog und gemischt analog/digital: Entwicklungsmethodik, Funktionsschaltungen, Funktionsprimitive von Schaltkreisen*. Springer-Verlag, 2014.
- [6] Michael Unser. Sampling-50 years after shannon. *Proceedings of the IEEE*, 88(4):569–587, 2000.
- [7] Wikipedia. Audio bit depth — wikipedia, the free encyclopedia, 2017. [Online; accessed 11-December-2017].
- [8] Wikipedia. Digital recording — wikipedia, the free encyclopedia, 2017. [Online; accessed 14-November-2017].