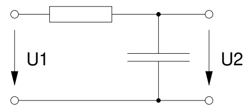
Digitale Signalverarbeitung

- Motivation
- analoge vs. digitale Signalverarbeitung
- Literatur

- Zahlenfolgen, LTI-Systeme
- Abtasttheorem
- Spektrum, FFT, Übertragungsfunktion
- Quantisierung, AD/DA Konverter, Dithering

- Audio-Algorithmen
- digitale Filter
- Dynamikbeeinflussung
- Raumsimulation, Hall

DSP: Motivation



analoge Signalverarbeitung:

- Modellierung mit Differentialgleichungen
- direkte Umsetzung mit diskreten Bauteilen
- hauptsächlich mit "analoger" Elektronik
- Widerstände, Kondensatoren, Spulen, Transistoren

aber:

- entweder geringe Genauigkeit (z.B. 10%) oder sehr teuer
- Alterung, Exemplarstreuungen, aufwendiger Abgleich
- wirtschaftlich nur für geringe Anzahl der Bauelemente
- Datenspeicherung (analog) problematisch

=> komplexe Algorithmen nicht umsetzbar
=> digitale Verarbeitung verspricht Abhilfe

DSP: Literatur

K.v.d.Heide, Skripte zu "Signalverarbeitung" und "Nachrichtentechnik", FB Informatik, Uni HH, tech-www.informatik.uni-hamburg.de/lehre/ (Matlab)

Mathworks, Inc., Matlab 5.3 User and Toolbox Manuals, www.matlab.com

U. Zölzer, Digitale Audiosignalverarbeitung, Teubner 1996

P.Gerdsen, P.Kröger, Digitale Signalverarbeitung in der Nachrichtenübertragung, Springer 1997

R.W.Hamming, Digital Filters, Prentice Hall, 1983

W.H.Press, B.P.Flannery, S.A.Teukolsky, W.T. Vetterling, Numerical Recipes, Cambridge Univ. Press

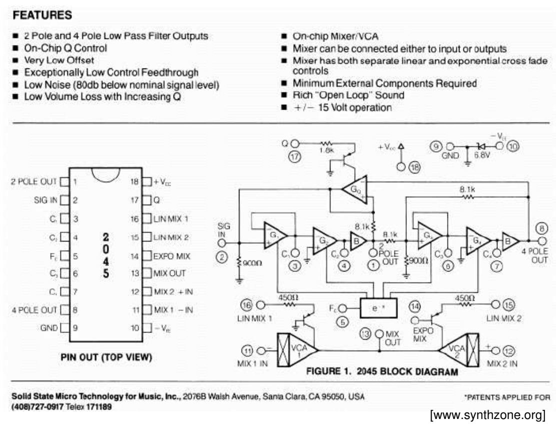
U. Tietze, Ch. Schenk, Halbleiter-Schaltungstechnik, Springer 1993 (analoge Schaltungen)

IEEE Journal Signal Processing

diverse Konferenzen zum Thema, Audio u.a. Audio Engineering Convention

DSP-Datenbücher (www.motorola.com, www.analog.com)

Beispiel für eine Audio-Analogschaltung



[www.synthzone.org]

- SSM-2045: Analogfilter mit Opamps
- typische Komplexität: einige Dutzend Komponenten
- Probleme mit Rauschen und Stabilität

DSP: Definition

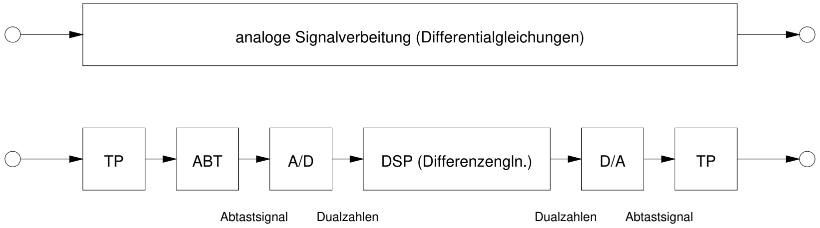
"digital signal processing" (digitale Signalverarbeitung):

Verbeitung von	für:
• zeitdiskreten	Nachrichtenübertragung
• wertdiskreten	Bildverarbeitung
• Zahlenfolgen	Audioverarbeitung
auf Digitalrechnern	usw.

- als Ersatz analoger Verfahren
- Differenzen- statt Differentialgleichungen

- oft mit Echtzeitanforderungen
- im Prinzip beliebige Genauigkeit
- oft mit Spezialhardware, etwa Signalprozessoren / ASICs

DSP: vs. analoge Verarbeitung



"analoge"	Signalverarbeitung	"digitale"
• zeitkontinuierlich		• zeitdiskret (Abtastung)
• wertkontinuierlich		• wertdiskret (Quantisierung)
• Differentialgleichungen		• Differenzengleichungen
• Drift, Rauschen, ...		• Überlauf, Rundungsfehler

DSP: Eigenschaften

Verarbeitung zeit- und wertdiskreter Zahlenfolgen:

- als Ersatz für zeit/wert-kontinuierliche Verarbeitung
- wenn Abtasttheorem erfüllt (genügend hohe Abtastrate)

- beliebige Genauigkeit (Wortbreite anpassen)
- optimale Stabilität (z.B. keine Temperaturabhängigkeit)
- kein Abgleich notwendig (z.B. Nullpunktseinstellung)
- keine Alterung, keine Toleranzen

- auch für sehr komplexe Algorithmen
- die analog nicht (wirtschaftlich) realisierbar sind
- insbesondere: Datenspeicherung

- VLSI-Technologie erlaubt sehr hohe Performance

DSP: Audio-Anwendungen

Signalverarbeitung im Audio-Bereich:

- Datenspeicherung (CD, DVD, ...)
- Datenkompression (MP3, AC3, MiniDisc, ...)
- Datenübertragung (DSR, GSM, . . .)

  jeweils für Musik und Sprache

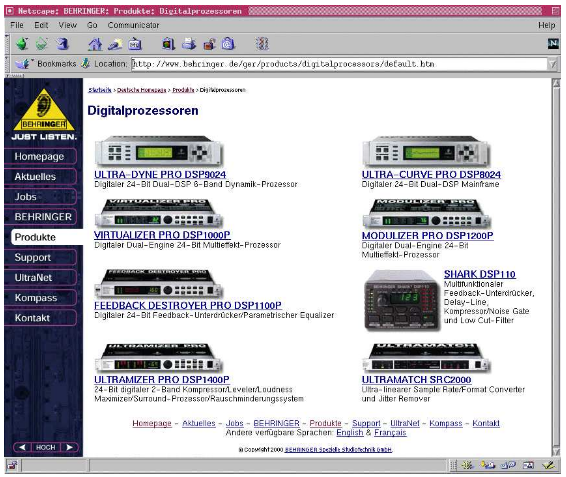
- Musikproduktion (digitales Tonstudio)
- Musiksynthese (z.B. virtuell analog)
- Audioeffekte (z.B. Hall)
- Surround-Verfahren (z.B. DirectX mit HRTF)

- Samplefrequenz typ. 44.1 / 48 KHz
- hohe Genauigkeit erforderlich: 16 .. 24 bit

DSP: Beispiel Mischpult

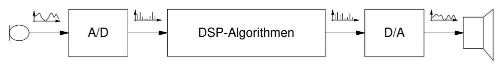
- parallele "Kanalzüge":
  Vorverstärker für Mikrophonsignale
  Pegelregelung (gain)
  Klangregler und Dynamikregler
- Mischung der Einzelsignale und Routing auf "Busse"
- Mischung der Bussignale

DSP: Auswahl Digitalgeräte

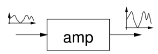


Spektrum: Filter, Effekte, Kompressor, Feedbackkiller, ...

DSP: Verstärker, Mixer

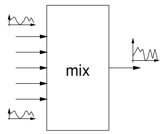


- digitaler Verstärker




```
output[t] = input[t] * gain;
```

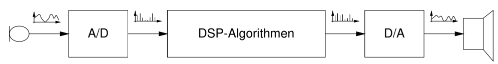
- digitaler Mixer:




```
output[t] = 0;
for( int i=0; i < n_inputs; i++) {
  output[t] += input[i][t] * gain[i];
}
```

- viele MAC-Operationen (multiply-accumulate)
- Problem Aussteuerung vs. Überlauf => saturation arithmetic

DSP: Echo, Hall, und mehr



- Verzögerung: auf alte Abtastwerte zurückgreifen

```
delay[t] = input[t]*gain + input[t-t2]*gain2;
```

- Rekursion möglich: Echo, Hall, Raumsimulation

```
hall[t] = input[t]*gain + hall[t-t2]*gain2;
```

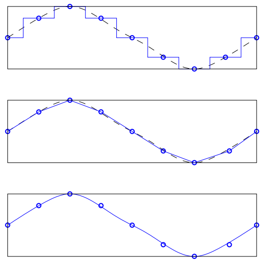
- Algorithmen für viele Anwendungen:
  Verzögerung, Echo, Hall, Overdrive, Verzerrung, ...
  Filter, Formatfilter, Tonhöhenänderung, Tempoänderung, ...

DSP: Sampling und Synthese

- Sampling: Abtastwerte abspeichern

```
sample[t] = input[t];
```

- Samples direkt abspielbar (CD, Spiele)



keine/lineare/Spline-Interpolation

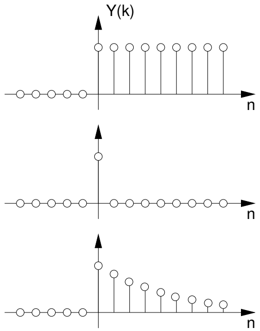
- "Wavetable"-Synthesizer:

```
output[t] = sample[t*pitch]
            interpolate( sample[] )
            filter( interpolate( sample[] ))
            effects( filter( interpolate( sample[] )))
```

- Interpolation: nearest / linear / splines / ...

Zahlenfolgen, Elementarfolgen

Wertemenge {Y(k)}
Zuordnung: Index k -> Y(k)



wichtige "Elementarfolgen":

Impulsfolge
Sprungfolge
Exponentialfolge
Sinusfolge

lineare Systeme:

- Signale aus Elementarfolgen zusammensetzen
- Systemverhalten entsprechend berechenbar

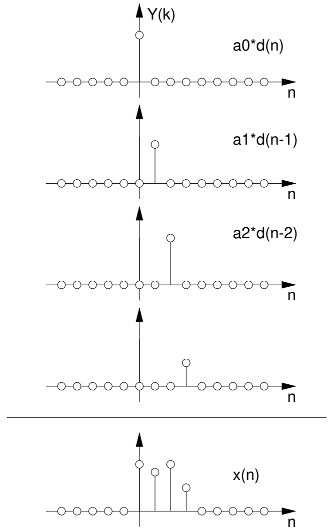
Impulsfolge:

Konstruktion beliebiger Zahlenfolgen:

- Summe
- skalierter
- zeitverschobener
  Impulsfolgen

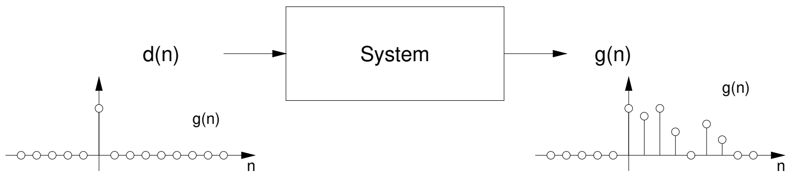
x(n) = a0 * d(n) + a1 * d(n-1) + . . .

gleichwertig: Konstruktion mit
- skalierten, zeitverschobenen
- Sprungfolgen



Impulsantwort

"Impulsantwort" := Ausgangsfolge als Reaktion auf die Impulsfolge



=> liefert vollständige Beschreibung von LTI-Systemen
"linear, time-invariant"
z.B. Verstärker, Filter, usw.
=> auch als lineare Approximation nichtlinearer Systeme

Differenzengleichungen

Berechnung des Ausgangswerts Y(n)

- aus gegenwärtigem Eingangswert X(n)
- aus früheren Eingangswerten X(n-k)
- aus früheren Ausgangswerten Y(n-k) (Rekursion)

- besonders wichtig: linear, konstante Koeffizienten, 2. Ordnung

  $y(n) = A0\ x(n) + A1\ x(n-1) + A2\ x(n-2) - B1\ y(n-1) - B2\ y(n-2)$

  Ordnung N: B1 .. BN kommen vor, A1 .. AM, M < N

- Übertragungsfunktion H(z) durch Z-Transformation:

  $Y(z) = A0\ X(z) + A1\ X(z)/z + A2\ X(z)/z^2 - B1\ Y(z)/z + B2\ Y(z) / z^2$

Fouriertransformation, diskret

- siehe Matlab-Skript Signalverarbeitung [vdHeide]

Fouriertransformation

Fouriertransformation:

für periodische, zeit- und wertkontinuierliche Funktionen

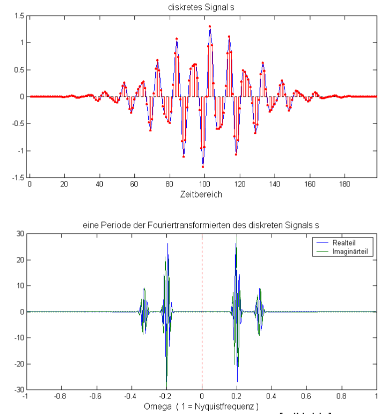
auch für periodische, zeit/wert-diskrete Funktionen

Fensterung erzwingt periodische Fortsetzbarkeit

Zahlenfolgen X(n) und Spektrum X*(f) sind
äquivalente Beschreibung von diskreten Signalen

- siehe Matlab-Skript Signalverarbeitung

Zeit- und Frequenzbereich



[vdHeide]

- Fouriertransformation
- FFT
- Fensterfunktionen

- siehe Matlab-Skript Signalverarbeitung [vdHeide]

Abtasttheorem

macht die Verarbeitung von Zahlenfolgen überhaupt Sinn ?

Abtasttheorem:

ein auf die halbe Abtastfrequenz (=Nyquistfrequenz)
bandbegrenztes Signal kann nach der Abtastung
perfekt rekonstruiert werden

- Grundlage aller digitalen Signalverarbeitung
- Beweis mit Fouriertransformation

- siehe [Shannon 48]
- siehe Matlab-Skript Signalverarbeitung

Abtasttheorem

nur im Bereich

$$-f_{gs} < f < +f_{gs} \qquad (7.8)$$

von null verschieden ist und innerhalb dieses Bereiches mit $\underline{U}_{per}(f)$ übereinstimmt, kann für die Zeitfunktion auch

$$u(t) = \int_{-f_{gs}}^{f_{gs}} \left\{ \sum_{n=-\infty}^{\infty} \underline{c}_n \exp(-j\, 2\pi\, n \frac{f}{2f_{gs}}) \right\} \exp(j\, 2\pi f t)\, df \qquad (7.9)$$

geschrieben werden. Man faßt die e-Funktionen zusammen und bildet die Summe der Integrale. So erhält man:

$$u(t) = \sum_{n=-\infty}^{\infty} \underline{c}_n \int_{-f_{gs}}^{f_{gs}} \exp(-j\, 2\pi f[t - \frac{n}{2f_{gs}}])\, df \quad . \qquad (7.10)$$

Die Lösung des Integrals in dieser Summe ergibt:

$$\int_{-f_{gs}}^{f_{gs}} \exp(-j\, 2\pi f[t - \frac{n}{2f_{gs}}])\, df = 2\, f_{gs} \frac{\sin(2\pi f_{gs}[t - \frac{n}{2f_{gs}}])}{2\pi f_{gs}[t - \frac{n}{2f_{gs}}]} \quad . \qquad (7.11)$$

Zur Bestimmung der Fourier-Koeffizienten $\underline{c}_n$ wird

$$t = k \frac{1}{2\, f_{gs}} \qquad k \text{ ganze Zahl} \qquad (7.12)$$

gesetzt. Dann sind in der Summe (7.10) alle Terme mit $n \neq k$ null. Damit gilt für

Abtasttheorem

7.4 Abtasttheorem für Zeitfunktionen

Ein reelles bandbegrenztes Signal u(t) hat eine Signalgrenzfrequenz f_{gs}. Für die Fourier-Transformierte $\underline{U}(f)$ dieses Signals gilt:

$$\underline{U}(f) = 0 \text{ für } |f| > f_{gs} \quad . \qquad (7.5)$$

Durch periodische Wiederholung von $\underline{U}(f)$ mit der Periode $2\, f_{gs}$ entsteht eine periodische Frequenzfunktion, die in eine Fourier-Reihe

$$\underline{U}_{per}(f) = \sum_{n=-\infty}^{\infty} \underline{c}_n \exp(-j\, 2\pi\, n \frac{f}{2\, f_{gs}}) \qquad (7.6)$$

entwickelt werden kann. Die Zeitfunktion

$$u(t) = \int_{\infty}^{\infty} \underline{U}(f) \exp(j\, 2\pi\, ft)\, df \qquad (7.7)$$

ergibt sich als inverse Fourier-Transformierte der Frequenzfunktion $\underline{U}(f)$. Da diese

[Gerdsen u. Kröger]

Abtasttheorem

die Koeffizienten:

$$\underline{c}_n = \frac{1}{f_{gs}}\; u(n \frac{1}{2\, f_{gs}}) \quad . \qquad (7.13)$$

So entsteht aus (7.10) die Interpolationsformel

$$u(t) = \sum_{n=-\infty}^{\infty} u(n \frac{1}{2\, f_{gs}}) \frac{\sin(2\pi f_{gs}[t - \frac{n}{2f_{gs}}])}{2\pi f_{gs}[t - \frac{n}{2f_{gs}}]} \quad , \qquad (7.14)$$

die aussagt, daß ein bandbegrenztes Signal u(t) mit der Signalgrenzfrequenz f_{gs} vollständig beschrieben ist durch seine Abtastwerte zu den Zeiten

$$t_k = k \frac{1}{2\, f_{gs}} \qquad k \text{ ganze Zahl} \quad . \qquad (7.15)$$

Dieses Abtasttheorem für Zeitfunktionen ist von grundlegender Bedeutung für die digitale Signalverarbeitung.

Abtasttheorem: Schema

Eingangsspektrum

Eingangstiefpaß

bandbegrenztes Signal

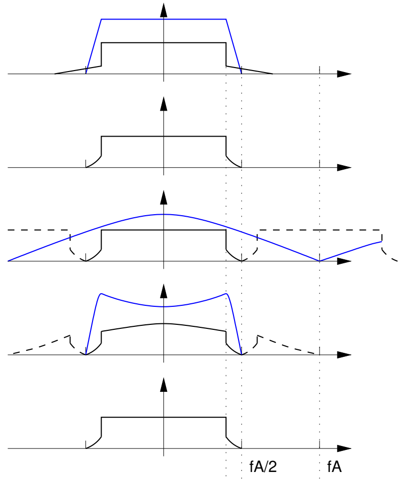
Abtastung

Abtastung sin(x)/x

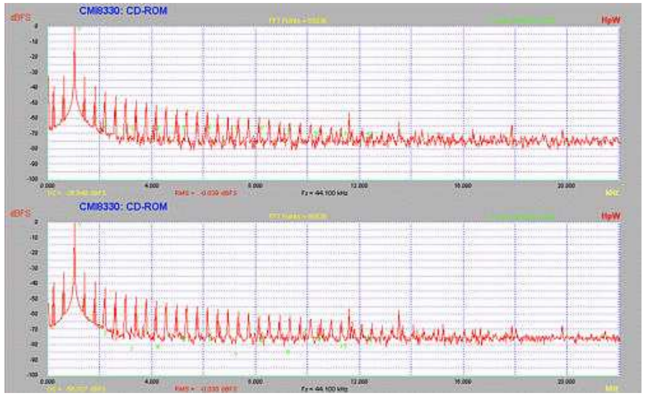
Rekonstruktion

Ausgangstiefpaß

Ausgangsspektrum

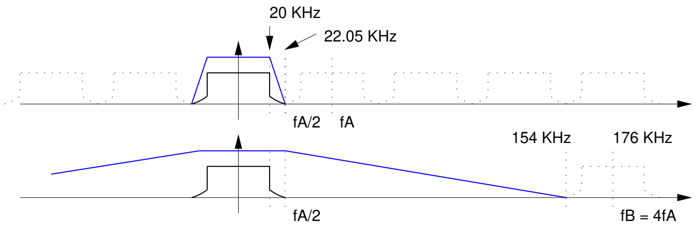


Abtasttheorem: Aliasing-Fehler . . .



- Meßwerte der CMI 8330 Soundkarte [ct 10/98]
- Sinussignal und massive Anti-Aliasing Fehler (Spiegelfrequenzen)
- massiver Qualitätsverlust durch Stümperei

Abtasttheorem: Überabtastung



Abtasttheorem:

- vollständige Rekonstruktion des Signals, aber:
- erfordert analogen Tiefpaß sehr hoher Qualität
- Beispiel CD: Dämpfung 96 dB von 20 KHz bis 22.05 KHz
- typische analoge Filter: 6 .. 24 dB / Oktave

Oversampling:

- digitale Neu-Abtastung mit höherer Rate (z.B. 4x, 8x)
- digitale Tiefpaßfilterung beseitigt die Spiegelfrequenzen
- einfacher analoger Tiefpaß genügt

Quantisierungsfehler

wertdiskret, Zahlendarstellung mit endlicher Wortbreite:

- Quantisierungsfehler von Koeffizienten
- Überlauf bei Additionen
- Auslöschung bei Subtraktionen
- Genauigkeitsverlust bei Multiplikationen

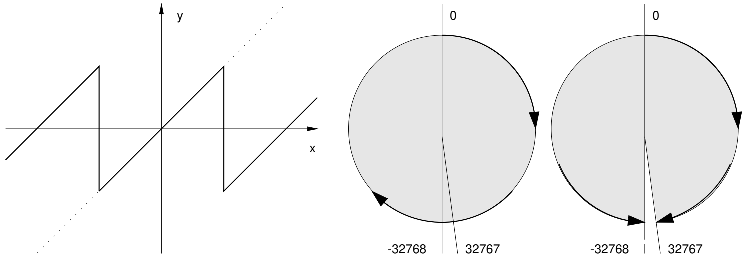
- z.B. andere Lage von Filter-Polen
- dadurch evtl. instabil (!)

- Mathematik: siehe Zölzer, Kap. 2

zeitdiskret, Nyquist-Bedingung

- Vorsicht mit Spiegelfrequenzen

Übersteuerung

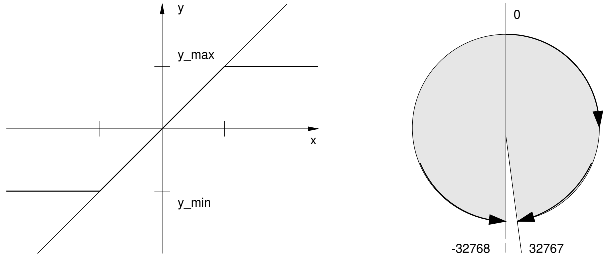


was passiert bei Überlauf / Übersteuerung?

- "wraparound" / "saturation" / "exception"

- arithmetische Ausnahmen unerwünscht (z.B. Echtzeitanforderung)
- normale Binärarithmetik liefert wraparound (0, 1, 2, 3, -4, -3, . . .)
  => maximaler Fehler (Verzerrungen)
- deshalb "saturation" oft besser für DSP geeignet

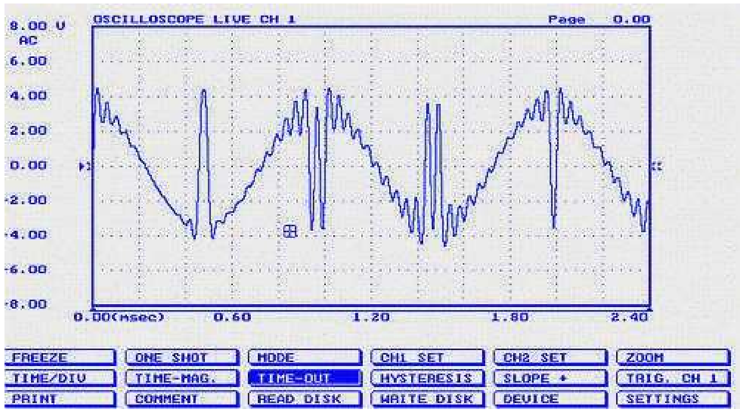
Saturation-Arithmetik



- saturation := "Sättigung"
- Resultat wird auf darstellbaren Zahlenbereich begrenzt
- kein "wraparound", sondern Weiterrechnen mit max/min. Wert

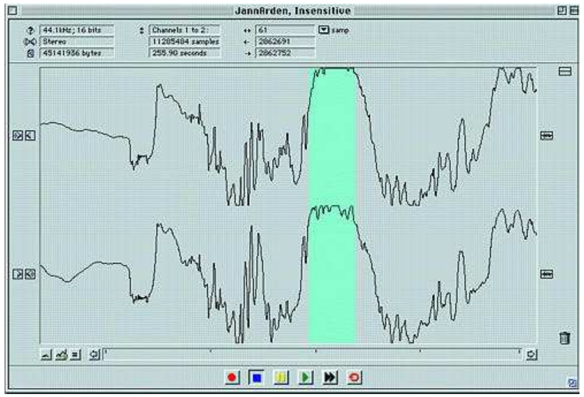
- in vielen Signalprozessoren verfügbar
- auch in MMX

Überlauf: wraparound. . .



- Beispiel für katastrophalen Überlauf mit wraparound:
  A/D-Eingang der 24-bit Audiokarte "Event GINA" [ct xx/98]

Überlauf: übersteuerte CD



- Beispiel für "saturation" auf übersteuerter CD
- weniger störend als wraparound

[ct 21/98 120]

A/D-Wandlung

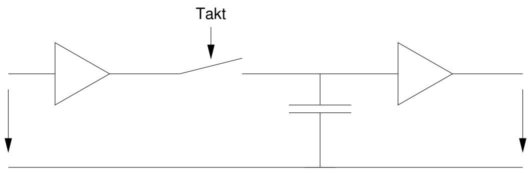
Digitalisierung eines Eingangssignals:

- Zeitquantisierung (Abtaster, Sample&Hold)
- Amplitudenquantisierung (A/D-Konverter)
- Codierung (meistens 2-Komplement)

- nur für bandbegrenzte Eingangssignale
- analoger Tiefpaß vor dem Eingang erforderlich

- diverse Verfahren
- Trade-off zwischen Performance / Genauigkeit / Kosten
- für Audio: 24-bit Wandler bereits erhältlich

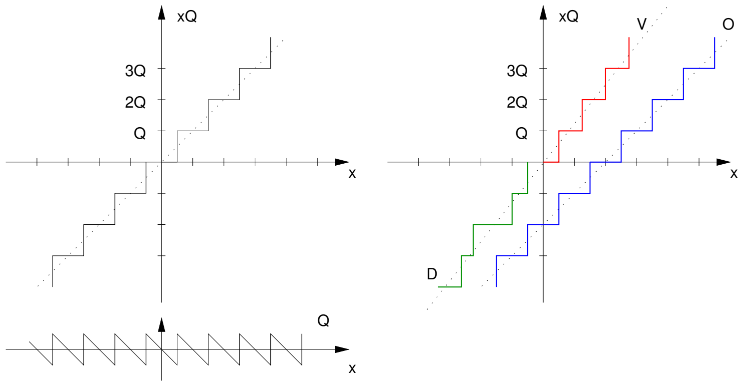
A/D: Abtaster (sample and hold)



Sample: Schalter periodisch schliessen
Hold: Ladung auf Kondensator gespeichert

- zwei Verstärker zur Entkopplung

A/D: Parameter



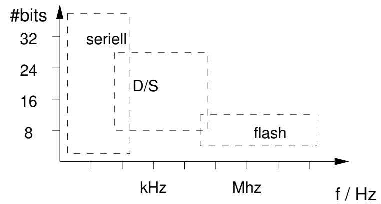
- Quantisierungsfehler — Q
- Offset / Nullpunktsfehler — O
- Verstärkungsfehler (falsche Steigung) — V
- differentielle / integrale Nichtlinearität — D

A/D: Verfahren

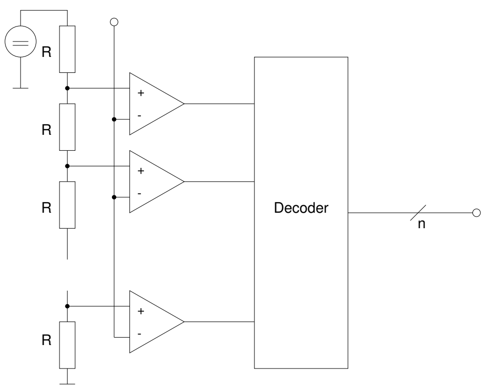
Vielzahl möglicher Verfahren:

- Parallel-Umsetzung ("Flash") schnell, teuer, < 10 Bit
- serielle Umsetzung langsam, billig, sehr genau
- Sigma-Delta-Umsetzer komplex, genau
- Kombinationen mehrerer Verfahren

- siehe Tietze/Schenk, Zölzer, Datenblätter

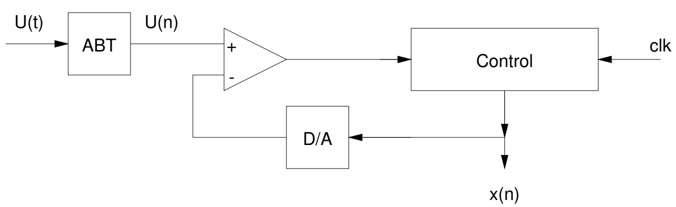


A/D: Parallel-Umsetzer

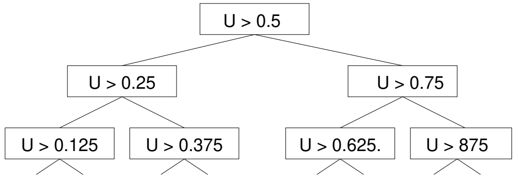


- erfordert 2**n Komparatoren
- sehr schnell
- aber nur für geringe Auflösungen (z.B. 8-bit)

A/D: Wägeverfahren



- Controller, Komparator, D/A-Umsetzer (s.u.)
- Ausgangswert durchläuft Binärcode, 1-bit pro Takt



A/D: Analog Devices AD 1878

PRODUCT OVERVIEW

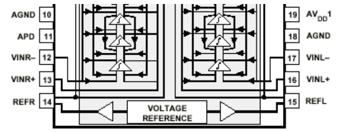
The AD1879 is a two-channel, 18-bit oversampling ADC based on ΣΔ technology and intended primarily for digital audio applications. The AD1878 is identical to the 18-bit AD1879 except that it outputs 16-bit data words. Statements in this data sheet should be read as applying to both parts unless otherwise noted.

Each input channel of these ADCs is fully differential. Each data conversion channel consists of a fifth order one-bit noise shaping modulator and a digital decimation filter. An on-chip voltage reference provides a voltage source to both channels stable over temperature and time. Digital output data from both channels is time-multiplexed to a single, flexible serial interface. The AD1878/AD1879 accepts a $256 \times F_S$ input master clock.

Input signals are sampled at $64 \times F_S$ on switched-capacitors, eliminating external sample-and-hold amplifiers and minimizing the requirements for antialias filtering at the input. With simplified antialiasing, linear phase can be preserved across the passband. The AD1878/AD1879's proprietary fifth-order differential switched-capacitor modulator architecture shapes the one-bit comparator's quantization noise out of the audio passband. The high order of the modulator randomizes the modulator output, reducing idle tones in the AD1878/AD1879 to very low levels. The AD1878/AD1879's differential architecture provides increased dynamic range and excellent common-mode rejection characteristics. Because its modulator is single-bit, AD1878/AD1879 is inherently monotonic and has no mechanism for producing differential linearity errors.

The digital decimation filters are single-stage, 4095-tap finite impulse response filters for filtering the modulator's high frequency quantization noise and reducing the $64 \times F_S$ single-bit output data rate to a F_S word rate. They provide linear

*Protected by U.S. Patent Numbers 5055843, 5126653, and others pending.



phase and a narrow transition band that permits the digitization of 20 kHz signals while preventing aliasing into the passband even when using a 44.1 kHz sampling frequency. Passband ripple is less the 0.001 dB, and stopband attenuation exceeds 115 dB.

The flexible serial output port produces data in twos-complement, MSB-first format. Input and output signals are to TTL and CMOS-compatible logic levels. The port is configured by pin selections. The AD1878/AD1879 can operate in either master or slave mode. Each 16-/18-bit output word of a stereo pair can be formatted within a 32-bit field as either right-justified, I²S-compatible, or at user-selected positions. The output can also be truncated to 16-bits by formatting into a 16-bit field.

The AD1878/AD1879 consists of two integrated circuits in a single ceramic 28-pin DIP package. The modulators and reference are fabricated in a BiCMOS process; the decimator and output port, in a 1.0 µm CMOS process. Separating these functions reduces digital crosstalk to the analog circuitry. Analog and digital supply connections are separated to further isolate the analog circuitry from the digital supplies.

The AD1878/AD1879 operates from ±5 V power supplies over the temperature range of –25°C to +70°C.

A/D: Analog Devices AD 1878

AD1878/AD1879*

FUNCTIONAL BLOCK DIAGRAM

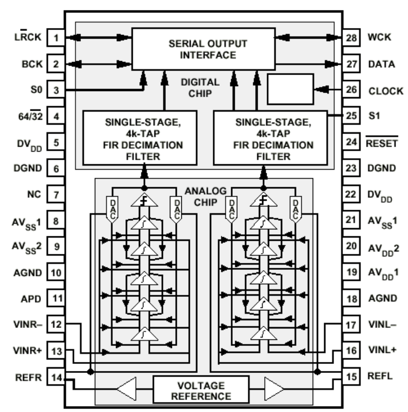


Figure 20. AD1879 S/(THD+N)—1 kHz Tone at –0.5 dBFS (4k-Point FFT)

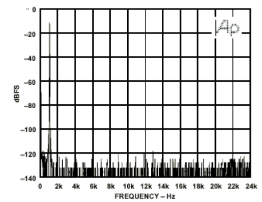


Figure 21. AD1879 S/(THD+N)—1 kHz Tone at –10 dBFS (4k-Point FFT)

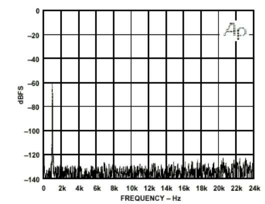
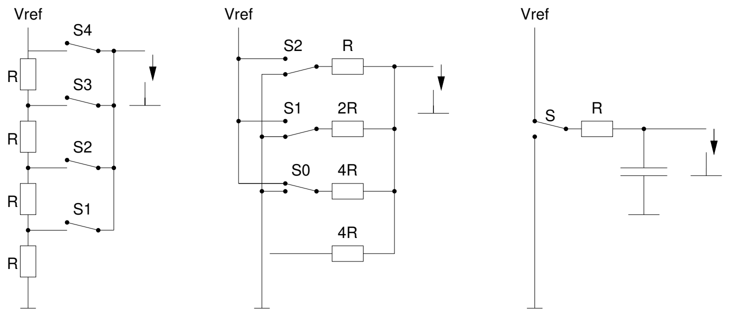


Figure 22. AD1879 S/(THD+N)—1 kHz Tone at –60 dBFS (4k-Point FFT)

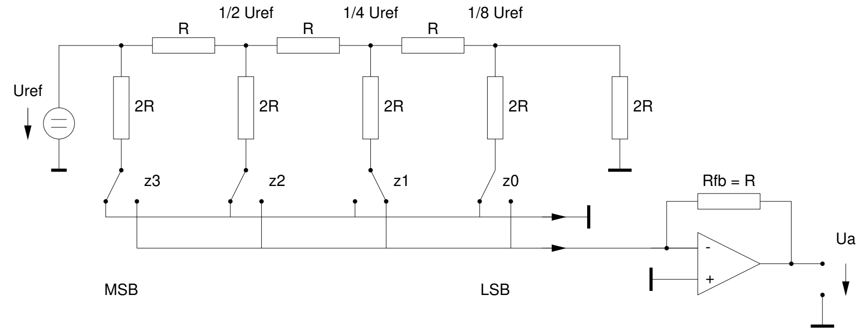
D/A-Wandlung

D/A: Umwandlung numerischer Werte in analoge Werte (Spannungen)

- Parallelverfahren
- Wägeverfahren
- Zählverfahren



D/A: R-2R Netzwerk



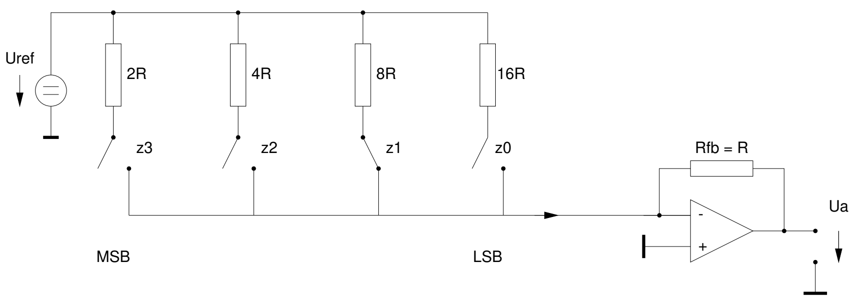
- VLSI: Herstellung unterschiedlicher Widerstände schwierig
- aber gleiche Widerstände mit hoher Genauigkeit

=> fortgesetzte Spannungsteilung mit R-2R Netzwerk

```
Ua = -Uref *(Rfb/16R) * (8z3 + 4z2 + 2z1 + z0)
```

- in CMOS-Technologie oft verwendet

D/A: Widerstandsnetzwerk



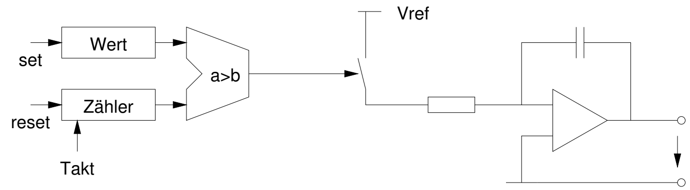
- Summation gewichteter Ströme (Kirchhoff)

```
Ua = -(Uref/16R) * (8z3 + 4z2 + 2z1 + z0)
```

- erfordert Widerstände mit hoher Genauigkeit
- datenabhängige Belastung von Uref

=> kaum realisisierbar

D/A: serieller Umsetzer

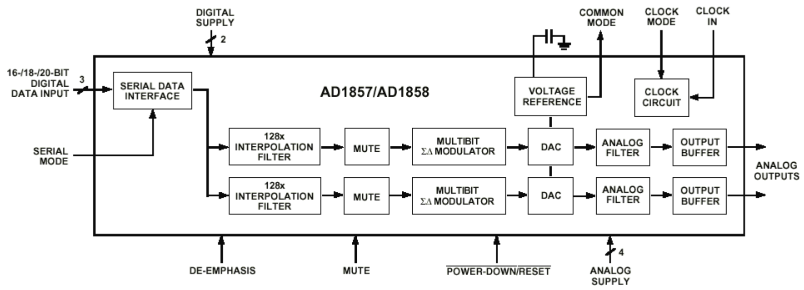


- "Pulsbreitenmodulation"
- Kondensator wird geladen, solange Wert > Zählerstand

- Taktrate des Zählers: 2**n fache der Samplefrequenz
- langsam
- geringer (analoger) Hardwareaufwand

D/A: Beispiel AD 1858

FUNCTIONAL BLOCK DIAGRAM



- Sigma-Delta Verfahren

Quantisierungsprobleme

[Zölzer]

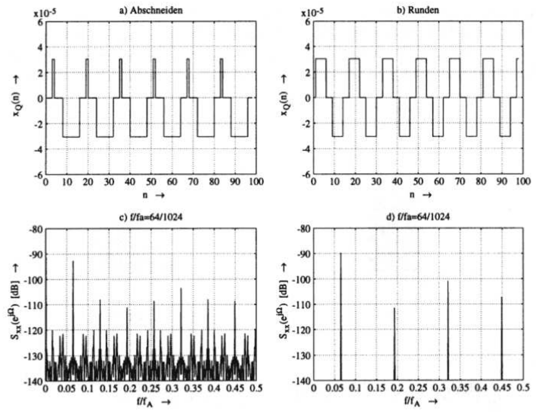


Bild 2.19: 1 Bit Aussteuerung des Quantisierers bei Abschneiden a/c und Runden b/d

- Quantisierung erzeugt störende "Nebensignale"
- Verbesserung durch Hinzufügen von Rauschen: "Dithering"

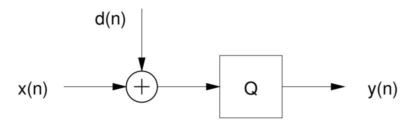
D/A: Beispiel AD 1858

ANALOG PERFORMANCE

	Min	Typ	Max	Units
AD1857 Resolution		18		Bits
AD1858 Resolution		16		Bits
Dynamic Range (20 Hz to 20 kHz, –60 dB Input)				
No A-Weight Filter		91		dB
With A-Weight Filter		94		dB
Total Harmonic Distortion + Noise		–90	**–85**	dB
		0.003	**0.006**	%
Analog Outputs				
Single-Ended Output Range (± Full Scale)	2.8	3.0	3.2	V p-p
Output Impedance at Each Output Pin		<200		Ω
Output Capacitance at Each Output Pin			20	pF
Out-of-Band Energy (0.5 × F_S to 100 kHz)			–72.5	dB
CMOUT	**2.1**	2.25	**2.4**	V
DC Accuracy				
Gain Error		±3.0	**±7.5**	%
Interchannel Gain Mismatch		0.01	**±0.2**	dB
Gain Drift		150	300	ppm/°C
Interchannel Crosstalk (EIAJ method)		–120	–100	dB
Interchannel Phase Deviation		±0.1		Degrees
Mute Attenuation		–100	–90	dB
De-emphasis Gain Error			±0.1	dB

- derzeit bis 24-bit Wandler erhältlich

Dither: Prinzip



		bits
x(n)	Eingangssignal	w+r
d(n)	Zufallssignal (dither)	s < r
y(n)	Ausgangssignal	w

- Q: Re-Quantisierung des Signals, oder D/A-Wandler

- Amplitude des Dithers kleiner als LSB des Eingangssignals
- Verteilung (Statistik) des Dithers frei wählbar: noise shaping
- Mathematik siehe Zölzer S.38 ff

Dithering: Varianten, Realisierung

- gleichverteilte Zufallszahlen d(n)

RECT	d_RECT(n)	= d1(n)
TRI	d_TRI(n)	= d1(n) + d2(n)
Hochpaß	d_HP(n)	= d1(n) - d1(n-1)

Wertebereich |d_RECT| = Q/2
Wertebereich |d_TRI| = Q

- höhere Rauschleistung des TRI/HP-Dithers
- aber besserer Klang
- Details: Zölzer S. 38 ff

- "noise shaping" für psychoakustisch optimales Dithering

Dither: bei 0.25 bit Aussteuerung

[Zölzer]

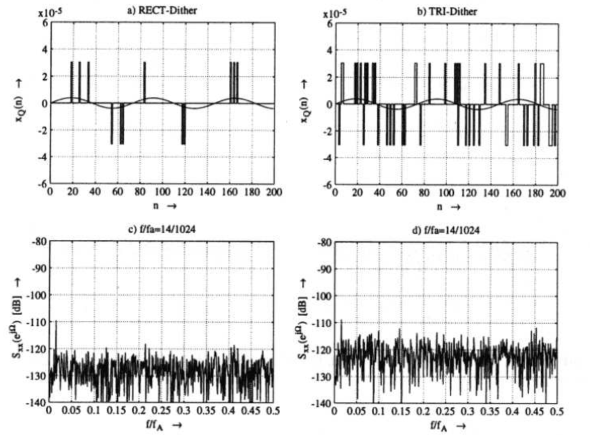


Bild 2.21: 0.25 Bit Aussteuerung - Rundungskennlinie mit RECT-Dither a/c und TRI-Dither b/d

Dither: bei 1 bit Aussteuerung

[Zölzer]

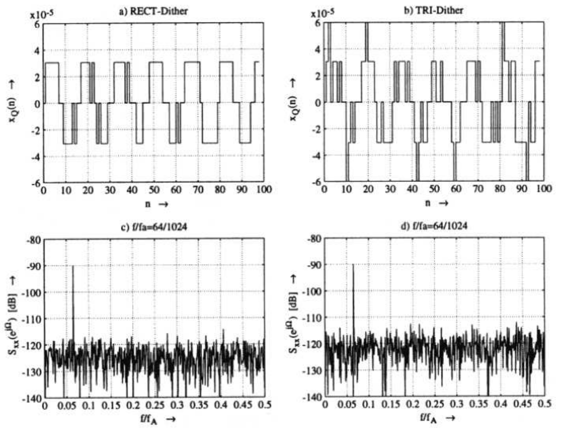


Bild 2.20: 1 Bit Aussteuerung - Rundungskennlinie mit RECT-Dither a/c und TRI-Dither b/d

Dither: Noise-Shaping

- ungleichmäßiger Verlauf der Hörschwelle

=> Idee: Verteilung des Rauschens an Hörschwelle anpassen

- möglichst wenig Rauschen bei ca. 4 KHz
- dafür höheres Rauschen bei tiefen/hohen Frequenzen

- höherer Rechenaufwand für das Dithering
- aber Gewinn von ca. 2.5 bis 6 Bit an Auflösung (!)
- z.B. effektiv 20 bit mit 18 bit D/A Wandlern

- auch beim CD-Mastering: z.B. Sony Super Bit Mapping

- Mathematik: siehe Zölzer, Kapitel 2.3