



## Voransichts- Version

Für Bezug des Originals  
siehe FAQ auf  
[www.embeddedFORTH.de](http://www.embeddedFORTH.de)



Master Station 1967

- 1 Inhalt, Impressum
- 2 IIR-Filter Überblick
- 4 2pol Allpol Filter
- 6 Verbesserte Sättigungslogik
- 9 Shift-Koeffizienten für FIR
- 10 Koordinatenwandler
- 13 DC-Blocker
- 14 Dynamikkompressor
- 16 Diodenkennlinie
- 17 Analoger Kompressor
- 18 Schnellere Pullups
- 19 Leaky Integrator
- 20 nanoFLASHER HC08
- 24 Multiplizierer

## READ . ME

Der Beitrag über Reed-Solomon Codes wurde nicht rechtzeitig fertig und folgt im nächsten Heft.

Die Listings sind in nanoFORTH geschrieben. Für die Konvertierung in andere FORTH-Varianten sollte man im nanoFORTH-Manual nachlesen das in der F08-Version verfügbar ist.

# Digitale Filter: IIR

Für vergleichbare Frequenzgänge benötigen IIR-Filter weniger Speicher und Multiplikationen als reine FIR-Filter. Jedoch muß die Wortbreite größer sein, der Phasengang ist nichtlinear. Bei unpassenden Koeffizienten oder auch nur ungünstiger Quantisierung können sie durch die Rückkopplungen instabil werden.

Ein „Infinite Impulse Response“-Filter besteht für allgemeinen Frequenzgang aus der Serienschaltung eines FIR- und eines reinen IIR-Blocks ( Bild 2 ). Das FIR „all zero“-Filter steuert Nullstellen bei, das IIR „all pole“ Filter die Polstellen.

## Struktur

Für digitale Signalverarbeitung ist die „Direktform“ üblich. Aus Typ I ( Bild 1 ) kann man durch Umzeichnen bzw. Veränderung der Anordnung ( Bild 2, 3 ) den Typ II ( Bild 4 ) gewinnen der eine minimale Zahl von Speicherstellen benötigt. In der Literatur finden sich noch andere Darstellungen, hier z.B. eine Variante von Typ I ( Bild 5 ).

Letztlich haben aber alle diese Varianten gleiche Koeffizienten und identisches Verhalten. Beim Übergang auf Festkommarechnung können Quantisierung der Koeffizienten und Rundung bei der Berechnung zu unterschiedlichem Verhalten führen, so daß beide Grundformen ihre Berechtigung haben.

Koeffizienten im IIR-Block sind typisch negativ. In der Literatur werden statt der Addierer deshalb oft

Subtrahierer dargestellt und die Koeffizienten sind dann positive Betragzahlen. Hier wird diese Darstellung vermieden.

Regelungstechnik bevorzugt andere Darstellungen. Die „zweite kanonische Struktur“ bzw. „Regler-Normalform“ ( Bild 6 ) ist zwar wieder nur eine anders gezeichnete Variante von Direktform Typ II. Die „erste kanonische Struktur“ bzw. „Beobachter-Normalform“ ( Bild 7 ) ist wirklich neu.

## Transponiert

Sie heißt in der DSP-Literatur meist „transposed form“ weil sie durch eine Transponierung der entsprechenden Matrix gebildet werden kann. Es gibt diese veränderte Anordnung auch für Teilfilter wie FIR ( Bild 8 ), aber die praktische Bedeutung ist hier gering.

Wenn man in der Direktform I ( Bild 9 ) die Pfeilrichtungen des Signalflusses überall umdreht [1] und auch an den Knoten die Addierer neu verteilt ( Bild 10 ) hat man Änderung praktisch schon bewältigt. Das Tauschen der Reihenfolge der Blöcke ( Bild 11 ) führt dann zur kanonischen Form I.

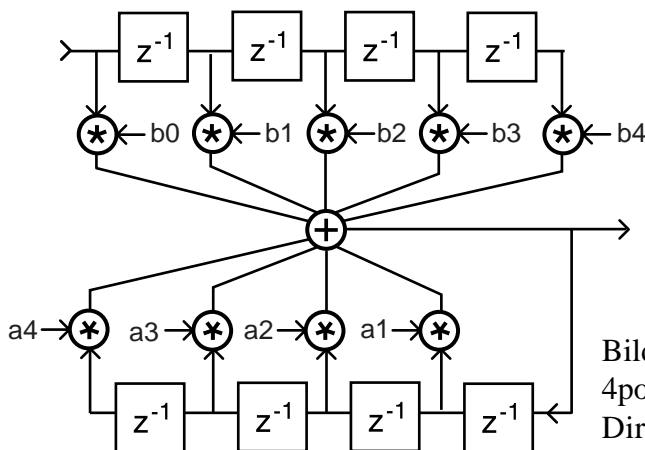


Bild 5:  
4pol IIR  
Direktform  
Typ I

Bild 1: 4pol IIR

Direktform Typ I

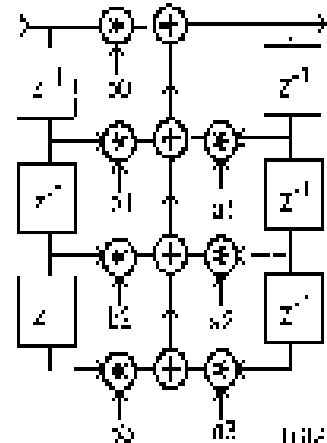


Bild 2

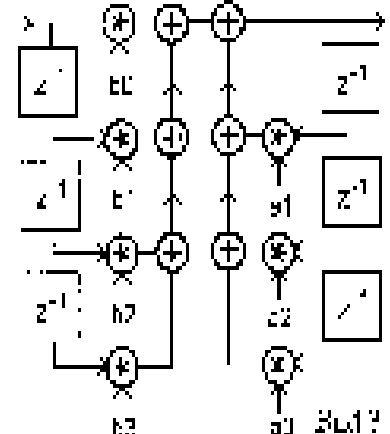


Bild 3

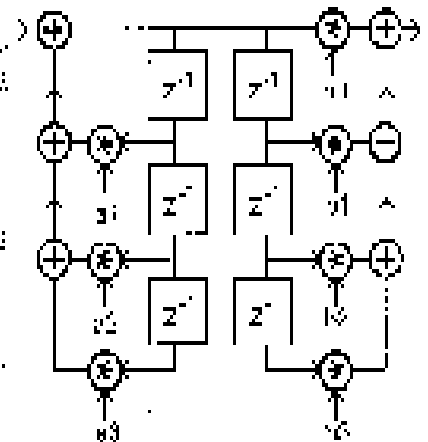


Bild 4: Direktform Typ II

Bild 6: 4pol IIR

Blockform II: Regler Normalform: 3. kanonische Form.

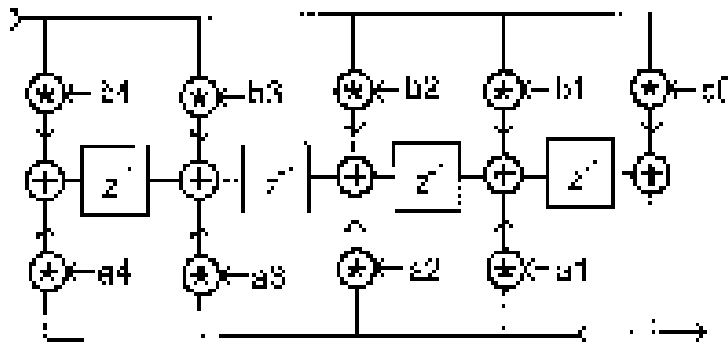
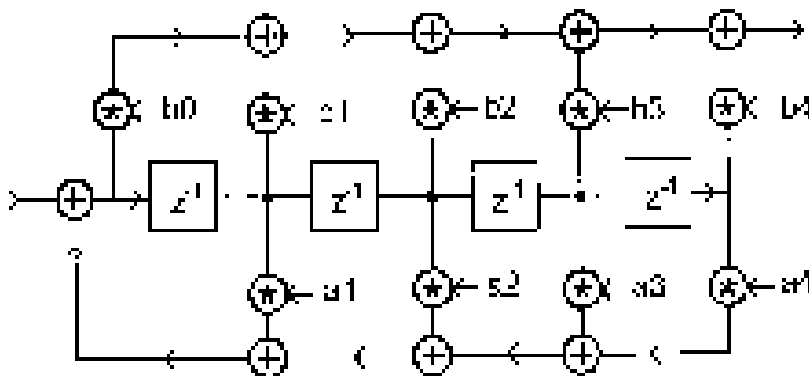


Bild 7: Transparenzform

Blockform Normalform: 1. kanonische Form



Teilfilter

Mit Modulare von Filter  
 Teilern: Ordnung direkt realisieren  
 Bei Teilern wird man sich an  
 Bandbreite u. zu verhalten im  
 Filter 1 und 2 Ordnung. Zusätzlich  
 sind die Kaskaden bzw. serienschalt-  
 ung als „1. kanonische Form“  
 (Bild 13) und die Parallelsetzung  
 als „2. kanonische Form“ (Bild

14) bezeichnet. Die beschriebene  
 Umsetzung in und zwischen diesen  
 Anordnungen hat wegen der  
 Zustandsverhaltens eine eigene  
 spezielle Bedeutung

[ ] Operation, Schalter  
 „Digital Signal Processing“  
 Prozedur Bild 14/15

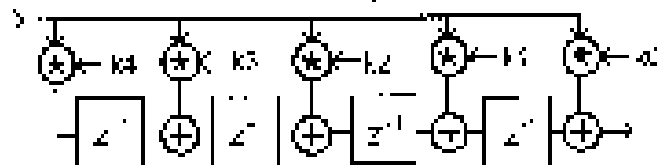
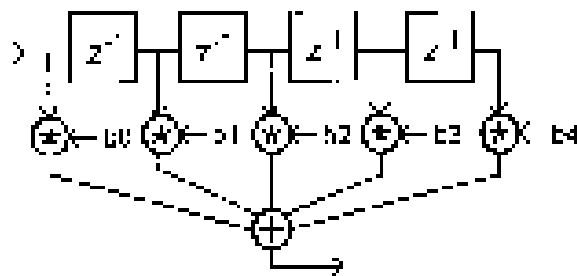


Bild 8: FIR

in Direct Form und  
 transparenzform

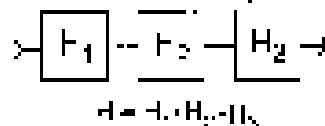


Bild 13  
 Kaskaden-  
 schaltung

Bild 9: 2pol IIR

in Transparenzform

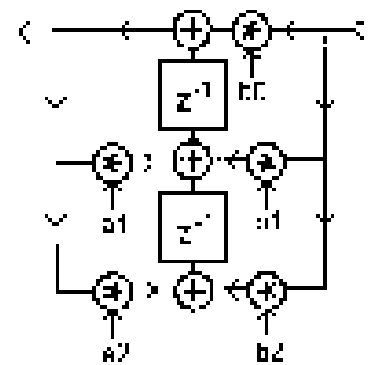
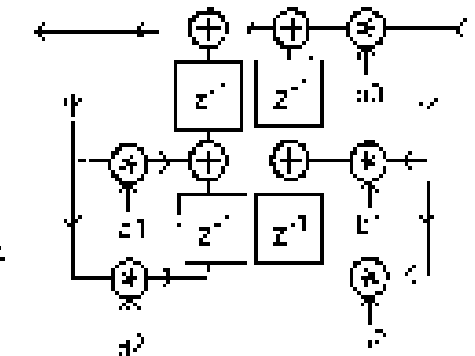
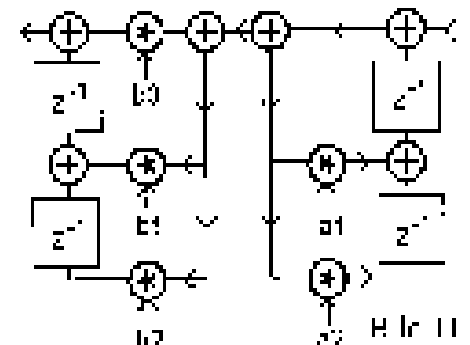
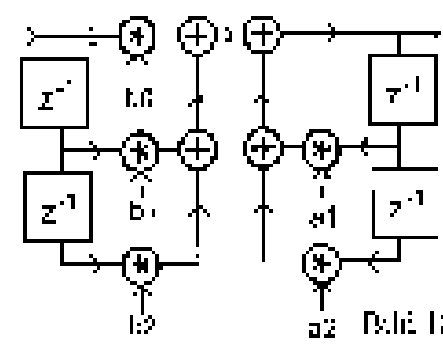


Bild 12: Transparenzform

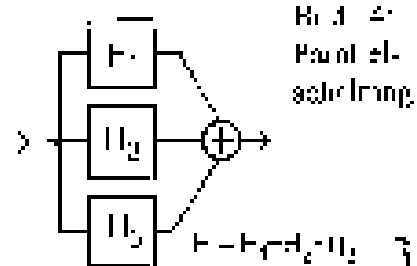


Bild 14  
 Parallel-  
 schaltung

# 2pol Allpol Filter

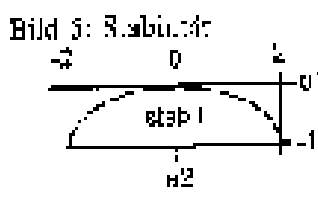
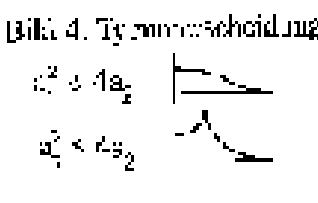
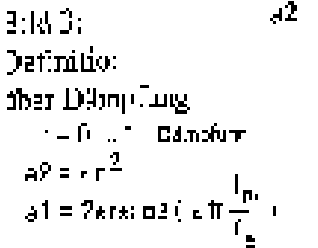
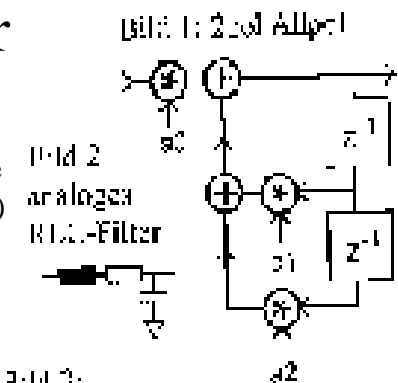
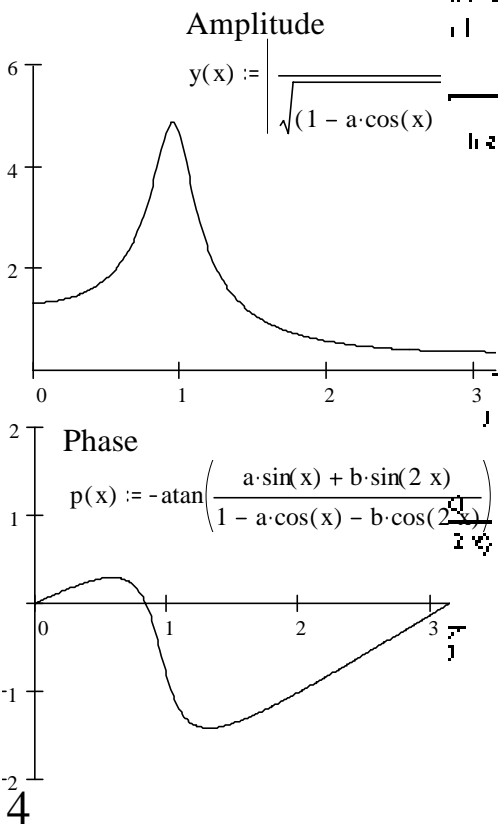
Die abgemagerte Form des Bi-quads ( Bild 1 ) wirkt ähnlich wie ein analoges RLC-Filter ( Bild 2 ) und ist speziell für Resonatoren, also schmale Bandpaßfilter vorteilhaft.

Die beiden Koeffizienten a1 und a2 legen gemeinsam sowohl die Resonanzfrequenz fm als auch die Dämpfung r fest ( Bild 3 ). Es gibt auch die weniger bedeutende Einstellung ohne Resonanz ( Bild 4 ). Sie kann zwei kaskadierte einpolige Filter ersetzen.

Die beiden Koeffizienten a1 und a2 sind so anzulegen daß keine Eigenschwingung auftritt ( Bild 5 ). Entsprechend der Amplitudenüberhöhung ( Bild 6 ) muß man mit der Vormultiplikation a0 das Eingangssignal soweit reduzieren daß kein Überlauf eintritt.

Oft sind Koeffizienten größer 1,0 bei der Implementierung ungünstig, es gibt deshalb Varianten die das Problem mit Shifts umgehen ( Bild 8 ).

Bild 6: Plots & Kennwerte mit Mathcad



x := 2,001 7 ;  
 a1 := 1 a2 := -0,45

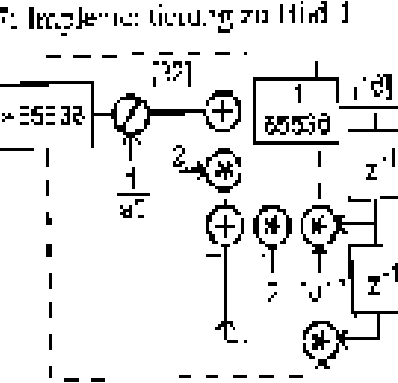
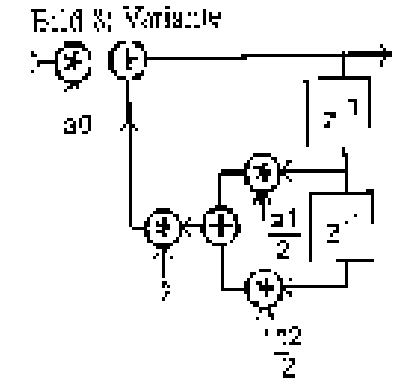
Resonanzfrequenz  
 $f_r = \frac{1}{2\pi} \arccos\left(\frac{a_1}{a_2}\right)$

Tr = 0,968

Amplitude bei dr  
 $G_{dr} = \frac{1}{1 - a_1}$

Amplitude bei Resonanzfrequenz  
 $G_{res} = \frac{1}{\sqrt{(1 - a_1)^2 + (a_2)^2}}$

Gr = 4,891

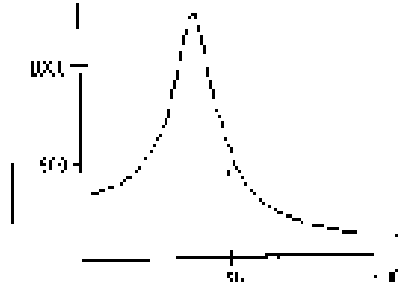


(1)  $1 - z^{-1}$   
 (2)  $1 - z^{-1}$   
 $a_1 = 0 = 2 \cdot 0,5$   
 $a_2 = 0,75$   
 $a_2^2 = 0,5625 = 2 \cdot 0,28125$

A,LPOL1  
 Freq: 250  
 Type: RESONANT FILTER  
 1000



Bild 8: Impulsantwort und LFT von Bild 7



# Implementierung

Die numerische Glue. in hier Diskrete, kann man mit mehr oder weniger Genauigkeit 16 Bit rechnen. Angenommen wenn man die Knotenwerte durch 32 Bit mal schrittweisen darstellen kann, was es bei 1,5 und 0,5 der Fall ist.

Ansonsten muß die Division am Eingang durch eine interne Schaltung im Rechner auf 32 Bit ausgeglichen werden. Bild 13 Skalierung mit 32 Bit wird mit Umrechnen angewandt durch führt die Umwandlung auf 32 Bit geht nur mit zwei Multiplikatoren über die 32 Bit Ergebnis liefert. Dann Faktoren 1,0 erzielbar sind. Bild 14 für 16 Bit gegeben ist, wenn der Wert 1,5 ist, dann man das Ergebnis geteilt für 32 erreicht.

Bild 12: Barpaß-Verzögerung

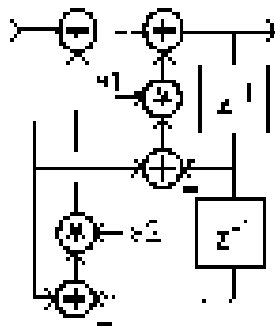
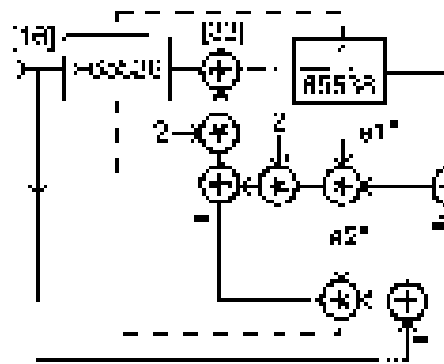


Bild 13: Implementierung Bild 12



$$*a1 = 32767 * 0.5 = 16383$$

$$*a2 = 32767 * 0.75 = 24575$$

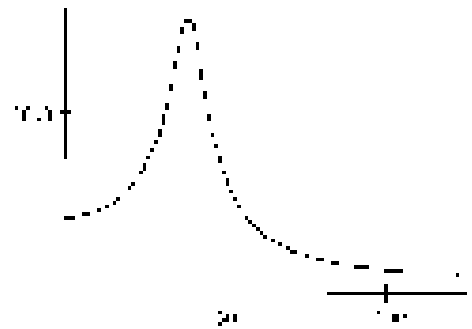


Bild 14: Impulsantwort, um Teil zu Bild 12

Bild 9: Barpaß-Verzögerung

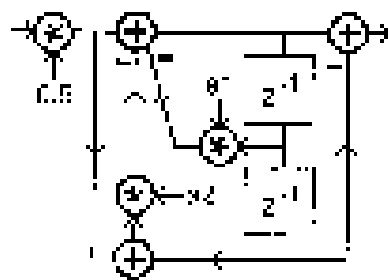
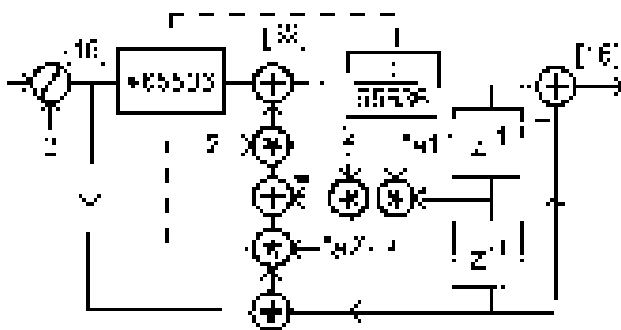


Bild 10: Implementierung, zu Bild 9



$$*a1 = 32767 * 0.5 = 16383$$

$$*a2 = 32767 * 0.75 = 24575$$

## Harmonik

Man kann die Barpaß-Verzögerung auch durch mehr Additionen umsetzen wenn man die Punkte verändert (Bild 11). Sie müssen jetzt nicht 16 und nicht 32 Bit verwenden für diese beiden Werten, da keine bilinear Skalierung erforderlich ist. Die Anzahl der Punkte kann kleiner sein.

## Barpaß + 100%

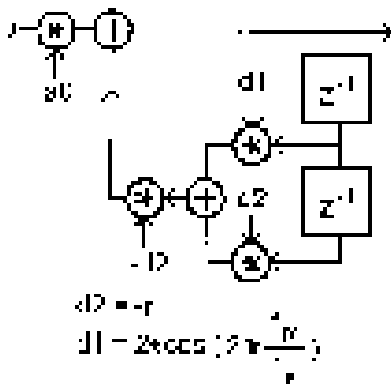
Es gibt eine ähnlich modulierte Verzögerung die 2X mehr verwendet (Bild 15).



Bild 11: Impulsantwort und RT zu Bild 10



Bild 15:  
Dämpfung und Frequenz  
getrennt einstellbar



$$d1 = -r$$

$$d2 = 2\cos\left(2\pi \frac{N}{r}\right)$$

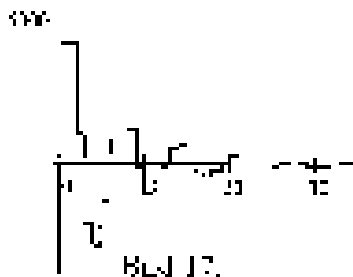
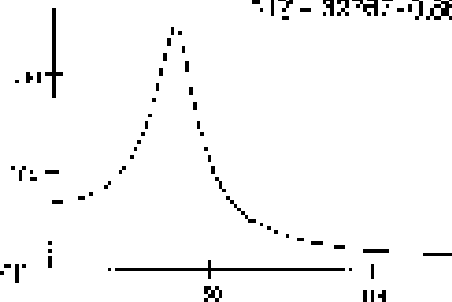


Bild 17:

Impulsantwort mit FFT



$$d1 = 91.1 = -0.1547$$

$$d2 = 22.667 = 1.04715 = -0.8518$$

$$d3 = \sqrt{2} = 0.800$$

$$d4 = 32.787 = 0.86612 = 0.5000$$

Adaptiv

Will man die Leistungsleistung  
Drehzahl ein adaptives Filter  
verändern, wird das Filter keine Rechen-  
funktionale und Dämpfung getrennt  
eingestellt. Das ist nur durch eine  
Multiplikation möglich (Bild 15)

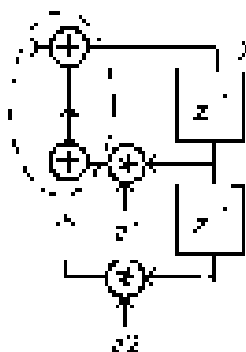


Bild 1: Filter

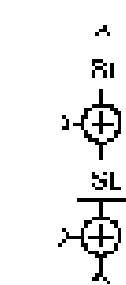


Bild 2:  
konventioneller  
Addierer mit  
Sättigungslogik

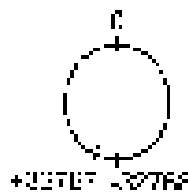


Bild 3:  
Kreisdiagramm  
Zur-Kompliment

Bild 4:  
Einschleifen  
Addieren zweier  
positiver Zahlen

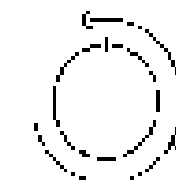
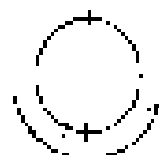


Bild 5:  
Addieren einer  
weiteren  
positiven Zahl

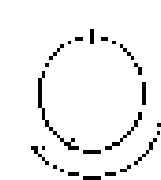


Bild 6:  
Addieren einer  
weiteren  
negativen Zahl

## Verbesserte Sättigungslogik

Der Vorteil einer (Bild 1) mussen Zahlen im 2er Komplement format addiert werden. Ummittelbar danach wird auf Überlauf geprüft und gegebenenfalls Sättigungswerte eingelesen (Bild 2)

Meinste Überlauf ist am besten über ein Kreisdiagramm darzustellen (Bild 3), hier mit dem Beispiel von zwei positiven Zahlen die tatsächlich von einem negativen

von Summe über den Bild 4. Addieren kann man eine weitere Zahl, kann das Ergebnis entweder weiterhin als Zahl (Bild 5) oder es man im Sprung über die 0 über den einen weiteren verfügen und das Ergebnis in ein wieder kommen (Bild 6)

Technische Nutzung von 8 in Computern vor Beginn der 60er Jahre Addition durch den Fehler mit 00 in den um über bei 255 zu kommen. Überlauf in den Komplement. Nur wenn 255 addierte Bild 1 SL wird man durch abschließend die Sättigungslogik (Bild 2)

Heute ist es der Rechner zu Dem mit bzw. die Rechner und die



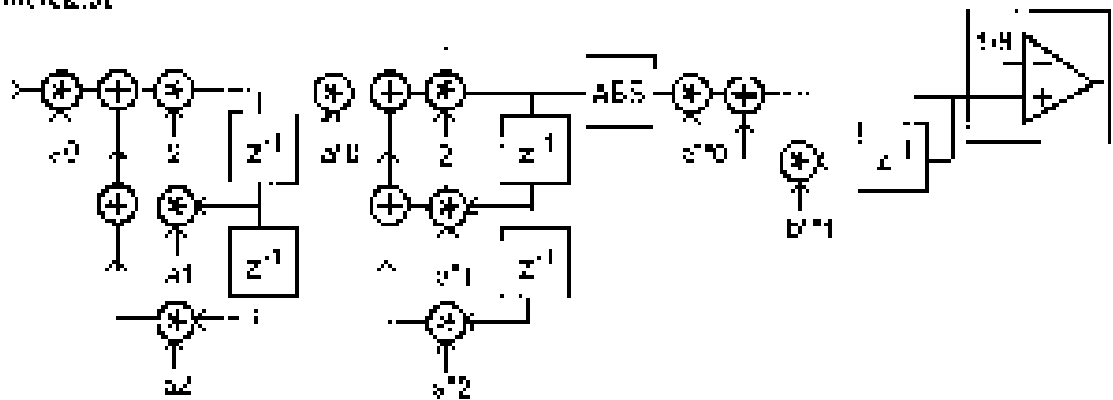
Bild 7:  
reduzierte  
Sättigungslogik

Verwendung des Sättigungslogik  
beinhaltet die Sättigungslogik  
abgegeben wird. Rechner. Wie sich  
mit der konventionellen Methode  
und die Reize

Verwendet man die Rechner  
Überlauf in 255 in den Komplement  
addiert. (FF) 255 (FF) 255



Bild 23: antefilter.at



Anwendung

Die beiden 6000-Filter werden hier von Kaskaden enthalten ein 2. Grades Filter mit Signalfrequenz hier 2000 Hz zu erhalten (Bild 23). Die Koeffizienten werden immer der Anwender einstellbar. Formel 18. Bei 2000 Hz Komplement. SampleRate oder 8000 Hz. Der rechte Teil mit Gleichrichter, Tiefpaß und Komparator auf dem Schaltungsplan ist IIR + FIR für hier nicht von Interesse.

Wenn man die Faktor 0.9 (vgl. Bild 23) entfernt, das Addition umkehrt und dann auch mit 0.9 definiert. Dann der Überlauf bei Addition ist die damit verbunden, dass Zeichen durch Quantisierung ist die.

Die Kaskadierung von Kaskaden Filtern mit überlappenden Zeitintervallen führt zu einer Senkung der Jitterbreite (Bild 24, 25). Das ursprüngliche Filter kann relativ geringe Güte haben und es ist leicht implementierbar sein. Auch wenn Kaskadierung mit Multi- und Additionen, realisiert werden steigt der Programmieraufwand. Aber allerdings erhöht sich natürlich die Rechenzeit. Wie in Softwareprozess ist, aber die Rechenzeit in Hardware kein Problem war.

Rechtlich wurde auch einen anderen Filter, man kann die Koeffizienten leicht unterschiedlich anlegen und dann die Jitterbreite etwas verkleinern und so die ursprüngliche Güte verbessern. Entsprechend der Formel zum mehr einem zweiten Bandpassfilter (Bild 26).

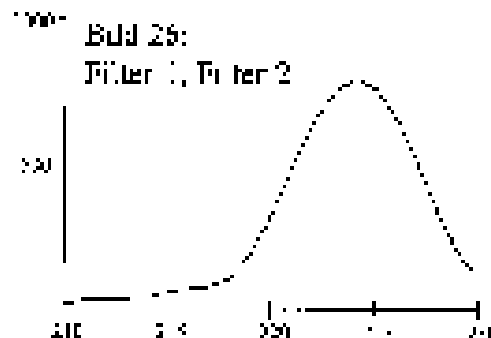
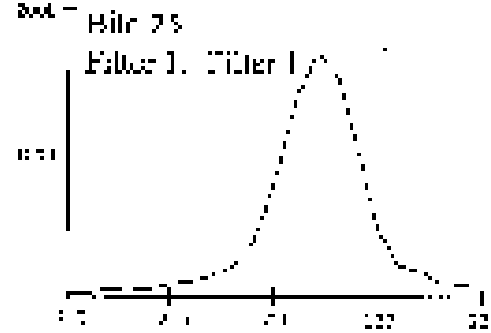
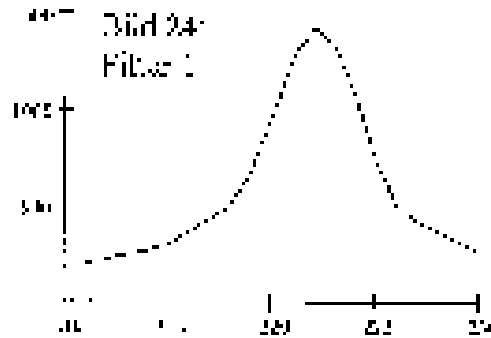
Die Form der Impulsantwort ist offensichtlich, ist jeweils die Reihenfolgezeit proportional (Bild 24, 25, 26). Deshalb sind FIR für diese Anwendung ungeeignet. Es wurde die Methode eine 1024 Punkte FFT verwendet. Wenn  $f_s = 20000$  Hz.

[1] Application Note „Digital Music Tone Detection Filter Tuning“ Intel® International 1997

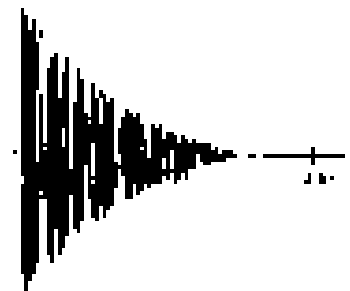
Tabelle 1: Koeffizienten 2. Grades

$a_0$	=	100	/	20000
$a_1$	=	0.450	/	10000
$a_2$	=	-161.87	/	20000
Koeffizienten A:				
$b_0$	=	0.20	/	20000
$b_1$	=	0.070	/	10000
$b_2$	=	-0.18	/	20000

FFT



Impulsantwort





# Koordinatenwandler

Die Umwandlung von polaren auf kartesisch Koordinaten bzw. umgekehrt (Bild 1) ist eine häufig benötigte, aber nicht leicht realisierbare Grundfunktion.

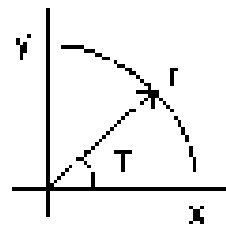


Bild 1: Formeln

$$\begin{aligned}
 & \text{Polar} = r, \phi \\
 & \swarrow \quad \searrow \\
 & x = r \cos \phi \quad \phi = \arctan \frac{y}{x} \\
 & y = r \sin \phi \quad r = \sqrt{x^2 + y^2} \\
 & \nwarrow \quad \nearrow \\
 & \text{Polar} = r, \phi
 \end{aligned}$$

In der Nachrichtentechnik ist diese Beschreibung für gefundene Werte in speziellen Fällen notwendig. In der Regel sind oft spezielle Werte gegeben und muss durch die FFT bestimmte Anwendungen wie z.B. Filter in digitalen Signalprozessoren erforderlich. Anwendungen von Geschwindigkeit gehen.

Diese ist in Sprung-RTI (K2038) implementiert. In der Praxis ist die FFT / FFT-Block 1997 (FFT) verwendet. Kartesische auf polare:  $x, y = 16 \text{ Bit}$ ,  $r = 16 \text{ Bit}$ ,  $\phi = 16 \text{ Bit}$ . Polare auf kartesisch:  $r = 16 \text{ Bit}$ ,  $\phi = 16 \text{ Bit}$ ,  $x, y = 16 \text{ Bit}$ .

Die FFT-Block (FFT) ist ein Block, der die FFT (Fast Fourier Transform) berechnet. In der Praxis ist die FFT / FFT-Block 1997 (FFT) verwendet. Kartesische auf polare:  $x, y = 16 \text{ Bit}$ ,  $r = 16 \text{ Bit}$ ,  $\phi = 16 \text{ Bit}$ .

## Polare auf kartesisch

Wie in Bild 2 dargestellt wird eine Operation auf Sinusfunktionen durchgeführt. Die Komponenten sind durch die FFT (Fast Fourier Transform) in die Zeitdomäne überführt. Die FFT (Fast Fourier Transform) ist ein Block, der die FFT (Fast Fourier Transform) berechnet.

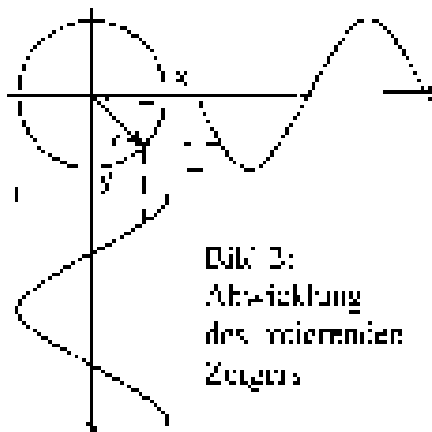


Bild 2: Abwicklung des Interferenzzeitraums

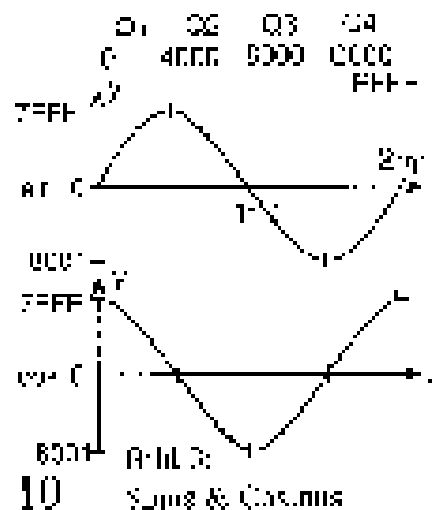
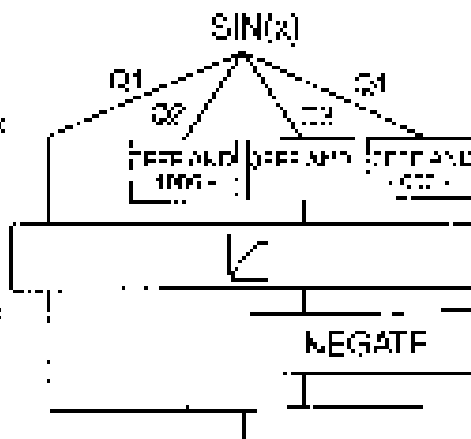


Bild 3: Sinus & Cosinus

genug, um die FFT (Fast Fourier Transform) zu berechnen. Die FFT (Fast Fourier Transform) ist ein Block, der die FFT (Fast Fourier Transform) berechnet. In der Praxis ist die FFT / FFT-Block 1997 (FFT) verwendet. Kartesische auf polare:  $x, y = 16 \text{ Bit}$ ,  $r = 16 \text{ Bit}$ ,  $\phi = 16 \text{ Bit}$ . Polare auf kartesisch:  $r = 16 \text{ Bit}$ ,  $\phi = 16 \text{ Bit}$ ,  $x, y = 16 \text{ Bit}$ .

Wenn man die Sinusfunktion in Bild 3 in die Zeitdomäne überführt, erhält man die FFT (Fast Fourier Transform). Die FFT (Fast Fourier Transform) ist ein Block, der die FFT (Fast Fourier Transform) berechnet.

Wenn die Wandler nur für eine einzelne Anwendung benötigt wird, kann man die folgende Matrix nutzen, um die Zeitdomäne zu eliminieren. Die Matrix ist in Bild 4 dargestellt.



verkalkuliert wird. Die Grundfunktion ist in Bild 4 dargestellt. Die Grundfunktion ist ein Block, der die Grundfunktion berechnet.

## Kartesisch auf polar

Diese Richtung ist ungenau, da die Grundfunktion in Bild 4 dargestellt ist. Die Grundfunktion ist ein Block, der die Grundfunktion berechnet. In der Praxis ist die FFT / FFT-Block 1997 (FFT) verwendet. Kartesische auf polare:  $x, y = 16 \text{ Bit}$ ,  $r = 16 \text{ Bit}$ ,  $\phi = 16 \text{ Bit}$ .

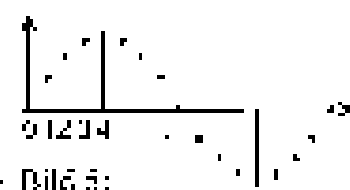


Bild 5: Organisationstabelle

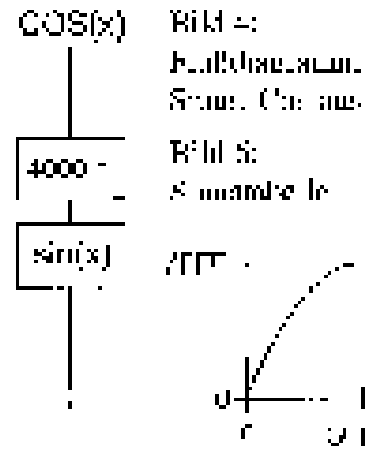


Bild 6: Halbbrücken-Sinus, Cosinus

Bild 7: Sinusfunktion



## Analogrechner

Man kann das Rechnen mit analoger Elektronik (sinus) (Bild 14) leicht zu lernen sein, wenn auf konventionelle Weise x, T, und y, da man Multiplizierer und Funktionsgeneratoren benutzt (Bild 15). In einem werden diese typischerweise als Multiplizierer oder Verstärker deren Werts umrechnen können, sondern Anordnungen auf dem Van der Pol-Prinzip für die Gleichungen wie Bild 16 im Handbuch ab.

Inversen sind zu Multiplizieren auf polar. Dazu werden zwei solche einfache Resolver verwendet, zusätzlich einer Regelkreise die den Winkel schneidet. Auf ein Summenprodukt auf Null ausgelegt wird (Bild 17). Die Detailschaltung zeigt Bild 18 im Handbuch, eine zusätzliche Anordnung im Handbuch.

## Tracking

Für Anwendungen in denen Nachführung genügt kann das Verfahren in Software für Integrierte, da AN-Systeme und die SORT-Belegung kann helfen. Bei einem anderen Beispiel wird eine Variable für T abhängig vom Wert einer Variable x, y, z durch internen oder externen Wert verändert, so der neue Zielwert erreicht ist. Wegen Regelungsfehler wird diese nicht auf Null ausgelegt, sondern die Suche nach dem Wert. Man muß jedoch auf Änderung der Regelungsanordnung und nicht dann die. Eine Anordnung in Abbildung von Bild 19.

Bild 16: sin/cos Resolver

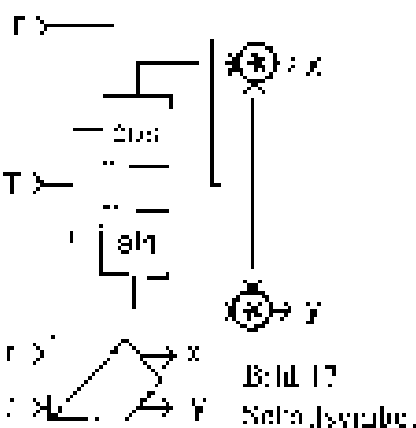


Bild 17: Sphärische

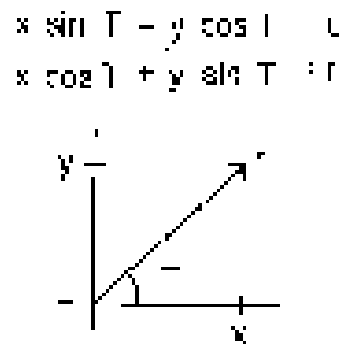
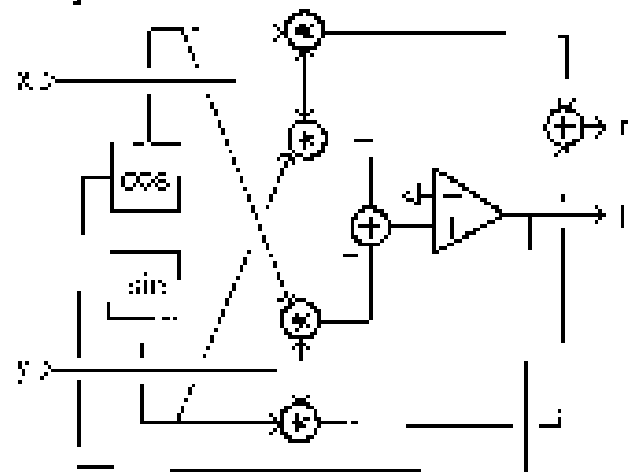


Bild 18: Konversion auf polar

Bild 19: unidirektive Schaltung



dem Wert ist erforderlich, damit die Suche nicht außer Kontrolle gerät und ein falscher Wert angenommen wird. Die Bestimmung kontinuierlich zu suchen und darf nicht ungleich sein, wobei der bestmögliche Wert der Suche ist. Das Verfahren besteht aus den Anwendungen genügt, wo der folgende Wert in weiterer Schritte anwendbar ist.

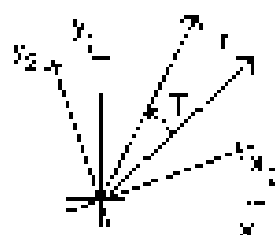
## Achsenrotation

Eine ähnliche Schaltung (Bild 20, 21) schreibe es einen kartesischen Vektor in einen anderen um.

Bild 20: Achsenrotation

$$x_1 \cos T + y_1 \sin T = x_2$$

$$y_1 \cos T - x_1 \sin T = y_2$$

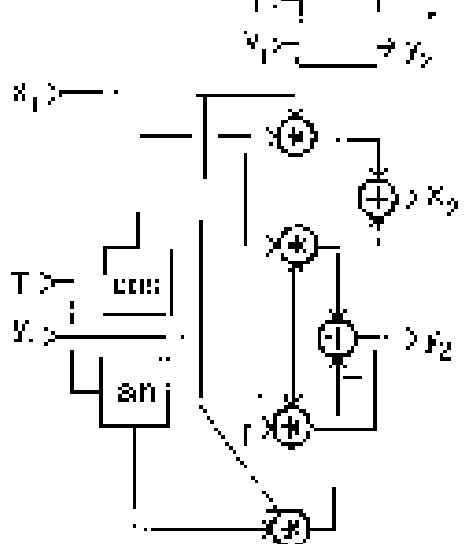


denen Maß die in den Zeiger beginnt im Uhrzeigersinn.

Man beachte jedoch, daß sich die Richtung ändert, wenn man sich von einem Wert zum nächsten bewegt.

Zur Beschreibung der Tracking Variable empfiehlt es sich den Sinus, V, zu verwenden, und die Anordnung in zwei Achsen der Schaltung zu verwenden (Bild 21).

Bild 21: Blockschaltbild





# Dynamik- kompressor

Abgeleitet von PCM-Codern [1] bietet sich für Signalverarbeitung auf 8 Bit Controllern mit 8 Bit A/D-Wandler eine Umcodierung über eine 256 Byte Tabelle auf 4, 5 oder 6 Bit an.

Durch ein ROM ist man in der Wahl der Kennlinie technisch nicht beschränkt. Praktisch ist aber doch der Rückgriff auf A-law und u-law. Einfacher berechenbar ist die Formel für u-law ( Bild 1 ). Üblich und wirksam ist der Wert  $u=255$ . Mittels Mathcad kann man die Kennlinie berechnen und auch gleich die Tabelle erzeugen.

Bild 1:  
u-Law  
Kennlinien

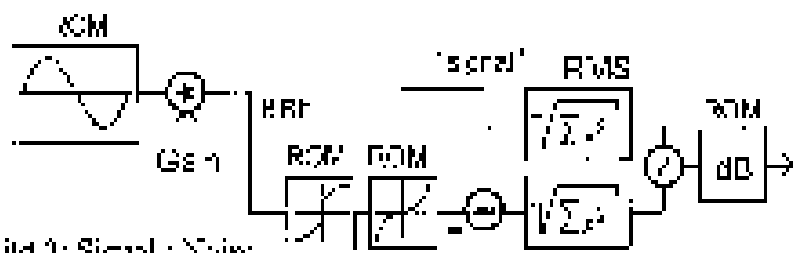
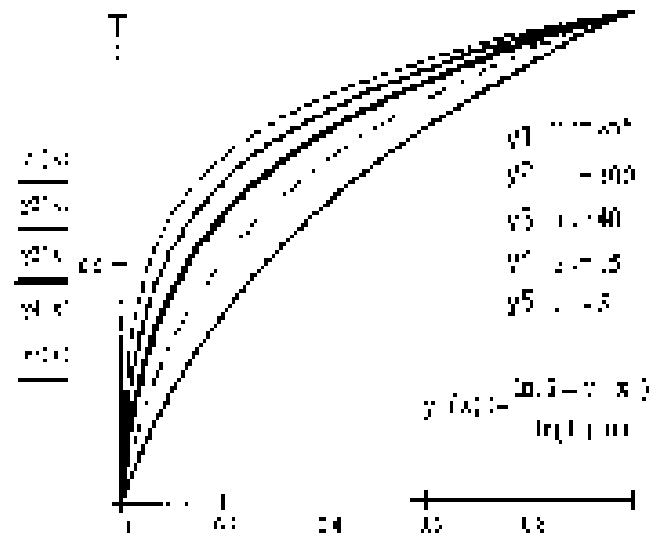


Bild 2: Signal : Noise  
Messung mit Sinus.

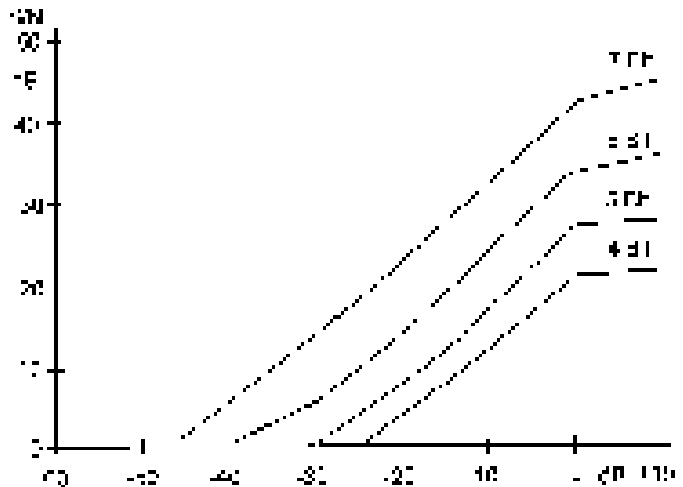
## S/N-Plots

Für die quantisierte Kennlinie will man nun den S/N-Verlauf wissen. Die Simulation des analogen Meßaufbaus ( Bild 2 ) in Software ist recht einfach. Als Signalquelle dient eine Sinus-Tabelle die 256 mit 16 Bit aufgelöste Samples enthält. Die Verstärkung wird in 5dB Schritten verändert. Als 0dB wird die volle Aussteuerung gewählt, Werte wie +5dB oder +10dB übersteuern bereits.

Da der Kompressor-Tabelle die Expander-Tabelle nachgeschaltet ist, wird näherungsweise das Referenzsignal wiederhergestellt. Differenzbildung zum Referenzsignal ergibt das gesuchte Rauschen. Aus Kompatibilität zum analogen System bildet man den Effektivwert beider Signale. Die Samples werden quadriert und addiert, aus der Summe zieht man die Wurzel. Die Division dient zu Verhältnisbildung, Umwandlung in dB ist über eine 256 Byte Tabelle einfach.

Übersteuerung ( +5dB, +10dB ) ist für S/N nicht sofort schädlich ( Tabelle 1 , Bild 3 ).

Bild 3:  
Leistungswert  
Plot  
zu Tabelle 1



Statistik

Für die Kennlinie ist die zeitliche Abfolge der Kompressionswerte über den 256-Byte-Tafel nicht unterscheidbar. Man kann die Amplitudenwerte der 256 Samples in 5dB-Schritten in 52 Klassen einteilen ( Bild 4 ).

Das Meßverfahren basiert immer noch auf der Abweichung gegenüber der Häufigkeit, die bei einem Analog-Analog-Äquator beobachtet werden kann. Die statistische Analyse der Häufigkeit ist die Grundlage für die statistische Analyse der Abweichung. Die Häufigkeit ist die Grundlage für die statistische Analyse der Abweichung.

Tabelle 1 S/N in dB

	1	5	10	15
0-10	15	16	17	18
10-15	21	22	23	24
15-20	27	28	29	30
20-25	33	34	35	36
25-30	39	40	41	42
30-35	45	46	47	48
35-40	51	52	53	54
40-45	57	58	59	60
45-50	63	64	65	66
50-55	69	70	71	72
55-60	75	76	77	78



# Diodenkennlinien

Der MC98DIP68C-P12 kann mit etwas Beschaltung (Bild 1) zur Datenerfassung dienen.

Sowohl PWM als auch A/D-Wandler haben die 5V-Versorgung als Referenz. Die Spannungswandlung hat deshalb die Aufgabe Messstromung für 0V/1 durch den A/D-Wandler. Der genaue Wert 1,25V = 5mV/entscheid.

## PWM

Diese Skalierung soll gemäß der Wertebereiche des A/D-Wandlers möglich sein. Die Periodendauer soll nicht zu groß sein, die Frequenz aber 50/1000 Hz betragen, die Punkte des PWM sollte zum 1000 Hz sein für Programmierung im A/D-Wandler. Die Periode ist in einem Taktzyklus angegeben. Die in der Tabelle angegebenen Werte sind die Werte, die die A/D-Wandler für die PWM sind. Im Bereich der Werte 1 bis 255 sind die Werte, die den Fehlerwert jeweils anzuweisen.

Hoch oder Lowwertigkeit. Man könnte annehmen, es ist nicht möglich, die Werte von 0 bis 255 zu erzeugen, ist es nicht möglich.

Der MC98DIP68C-P12 ermöglicht die Programmierung der Stromwerte. Die Werte sind in der Tabelle angegeben. Die Werte sind in der Tabelle angegeben. Die Werte sind in der Tabelle angegeben.

## Messung

Der PWM-Wert ergibt eine Kurve von 0 bis 255 hoch und auch jeden Schritt wieder nach einer Messung mit dem A/D-Wandler. Ein Austausch der Widerstände wird der Wandler dabei nicht erforderlich. Solange bis 0 bis 255 Werte sind. Ein Wert von 0 bis 255 ist die Messung. Die Messung ist die Messung.

Bild 1. Schaltung

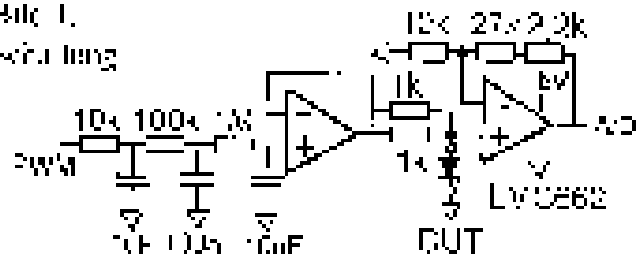


Tabelle 2: PWM

high	low
100	007
101	008
102	009
103	010
104	011
105	012
106	013
107	014
108	015
109	016
110	017
111	018
112	019
113	020
114	021
115	022
116	023
117	024
118	025
119	026
120	027
121	028
122	029
123	030
124	031
125	032
126	033
127	034
128	035
129	036
130	037
131	038
132	039
133	040
134	041
135	042
136	043
137	044
138	045
139	046
140	047
141	048
142	049
143	050
144	051
145	052
146	053
147	054
148	055
149	056
150	057
151	058
152	059
153	060
154	061
155	062
156	063
157	064
158	065
159	066
160	067
161	068
162	069
163	070
164	071
165	072
166	073
167	074
168	075
169	076
170	077
171	078
172	079
173	080
174	081
175	082
176	083
177	084
178	085
179	086
180	087
181	088
182	089
183	090
184	091
185	092
186	093
187	094
188	095
189	096
190	097
191	098
192	099
193	100
194	101
195	102
196	103
197	104
198	105
199	106
200	107
201	108
202	109
203	110
204	111
205	112
206	113
207	114
208	115
209	116
210	117
211	118
212	119
213	120
214	121
215	122
216	123
217	124
218	125
219	126
220	127
221	128
222	129
223	130
224	131
225	132
226	133
227	134
228	135
229	136
230	137
231	138
232	139
233	140
234	141
235	142
236	143
237	144
238	145
239	146
240	147
241	148
242	149
243	150
244	151
245	152
246	153
247	154
248	155
249	156
250	157
251	158
252	159
253	160
254	161
255	162

Tabelle 3: Daten im I/O-Format

high	low
100	007
101	008
102	009
103	010
104	011
105	012
106	013
107	014
108	015
109	016
110	017
111	018
112	019
113	020
114	021
115	022
116	023
117	024
118	025
119	026
120	027
121	028
122	029
123	030
124	031
125	032
126	033
127	034
128	035
129	036
130	037
131	038
132	039
133	040
134	041
135	042
136	043
137	044
138	045
139	046
140	047
141	048
142	049
143	050
144	051
145	052
146	053
147	054
148	055
149	056
150	057
151	058
152	059
153	060
154	061
155	062
156	063
157	064
158	065
159	066
160	067
161	068
162	069
163	070
164	071
165	072
166	073
167	074
168	075
169	076
170	077
171	078
172	079
173	080
174	081
175	082
176	083
177	084
178	085
179	086
180	087
181	088
182	089
183	090
184	091
185	092
186	093
187	094
188	095
189	096
190	097
191	098
192	099
193	100
194	101
195	102
196	103
197	104
198	105
199	106
200	107
201	108
202	109
203	110
204	111
205	112
206	113
207	114
208	115
209	116
210	117
211	118
212	119
213	120
214	121
215	122
216	123
217	124
218	125
219	126
220	127
221	128
222	129
223	130
224	131
225	132
226	133
227	134
228	135
229	136
230	137
231	138
232	139
233	140
234	141
235	142
236	143
237	144
238	145
239	146
240	147
241	148
242	149
243	150
244	151
245	152
246	153
247	154
248	155
249	156
250	157
251	158
252	159
253	160
254	161
255	162

schwand der vom PWM eingepulsten Spannung. Die Werte sind in der Tabelle angegeben.

## 1. Messung der PWM-Werte

Die Werte sind in der Tabelle angegeben. Die Werte sind in der Tabelle angegeben. Die Werte sind in der Tabelle angegeben.

Tabelle 3: Daten im I/O-Format

high	low
100	007
101	008
102	009
103	010
104	011
105	012
106	013
107	014
108	015
109	016
110	017
111	018
112	019
113	020
114	021
115	022
116	023
117	024
118	025
119	026
120	027
121	028
122	029
123	030
124	031
125	032
126	033
127	034
128	035
129	036
130	037
131	038
132	039
133	040
134	041
135	042
136	043
137	044
138	045
139	046
140	047
141	048
142	049
143	050
144	051
145	052
146	053
147	054
148	055
149	056
150	057
151	058
152	059
153	060
154	061
155	062
156	063
157	064
158	065
159	066
160	067
161	068
162	069
163	070
164	071
165	072
166	073
167	074
168	075
169	076
170	077
171	078
172	079
173	080
174	081
175	082
176	083
177	084
178	085
179	086
180	087
181	088
182	089
183	090
184	091
185	092
186	093
187	094
188	095
189	096
190	097
191	098
192	099
193	100
194	101
195	102
196	103
197	104
198	105
199	106
200	107
201	108
202	109
203	110
204	111
205	112
206	113
207	114
208	115
209	116
210	117
211	118
212	119
213	120
214	121
215	122
216	123
217	124
218	125
219	126
220	127
221	128
222	129
223	130

# Analoger Dynamikkompressor

Wenn man vor den 8-Bit A/D-Wandler in einem Controller eine Schaltung mit passender nichtlinearer Kennlinie (z.B. für etwa 255 Stufen) kann man sich hoch wie bei PCM-Chips (1) in analogen Anwendungen die Dynamik schöpfen. Im Controller ist dazu eine 512-Byte-Tabelle nötig, die die Werte wieder freizugibt.

Bild 1:

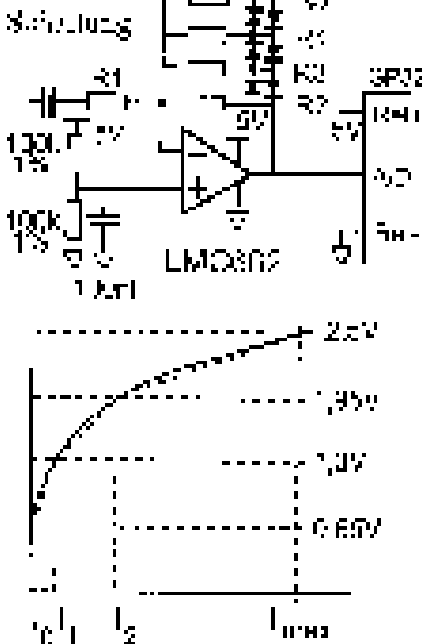


Bild 2:

Kennlinie mit Nachführung für Dither

Mit einem variablen Ditherwert wird die Kennlinie (Bild 2) umgeben, die auch mit 5V-Schwung verwendbar ist.

## Grundrechnung

Wegen der Symmetrie kann man sich auf eine Hälfte der Kennlinie beschränken (Bild 2). Diese wird dann in drei Geradenstücken ausgenutzt, deren Steigungswerte durch die Kennlinie an der Schwellenspannung der Dither (hier: also 0,65V) abgelesen werden. Man kann die Kennlinie dann in drei Stufen aussteigend mit 0,65V und einem Widerstand produzieren (Bild 3). Dann ist es anhand der Zeitpunkte (Bild 2) möglich, schrittweise per Hornerrechnung Näherungswerte der Widerstände zu bestimmen (Bild 4).

## Brückenmodell

Die CMOS-Chips stellen allerdings nur bei 5V am I/O-Pin die Versorgungsspannung und die Rückkopplungen jeweils sind damit löslich. Man muss sich um die Invertierung des Signals bei Totzeit handeln, ist die Nachführung auf eine Nachspannung in der Endschaltung zu ignorieren. Deshalb ist als nächster Schritt eine Testschaltung mit 10kΩ (1) mit der per D/A-Wandler (2, 3) die Kennlinie eingepreist wird, die die Schaltung passiv realisiert (Bild 6). Wenn das Ergebnis per A/D-Wandler eingetragener, kann der Vergleich zum Sollwert aus einer Tabelle (4) erlösen, und die neuen D/A-Werte des Systems angepasst werden. Da die Kennlinie (Bild 2) erhalten wird, hier nach weiterer Verfeinerung und Feinabstimmung (5) (Bild 7).

Die Kennlinie ist dann ungenügend zum Nachvollzug, symmetrisch. Wenn nur die 5V-Bestandsdaten Komposition aus Widerständen

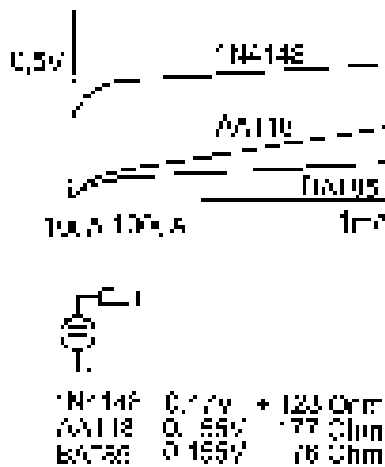
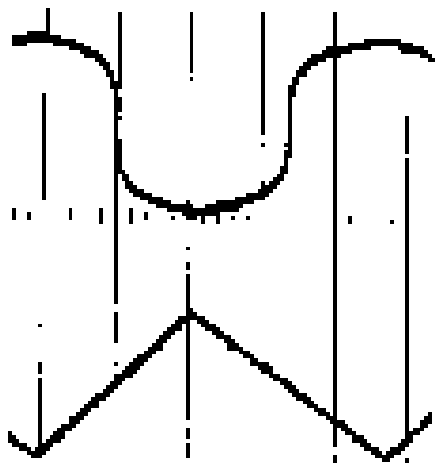


Bild 3:

Ditherwert linear und verschobene Nachführung

Bild 5: Eingangssignale, Kompressionsraten



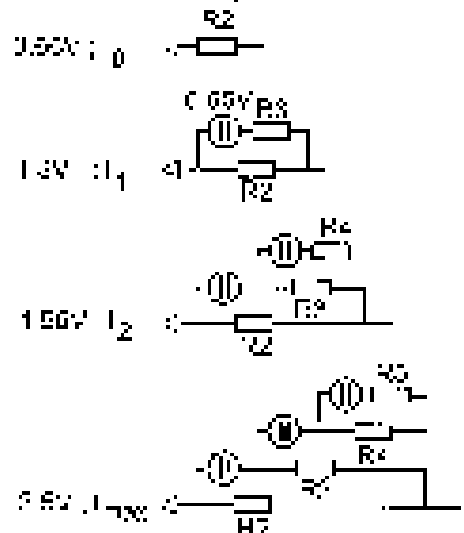
Am Ende des Signals, die nur bei der Startzeit aus der Frequenzumkehrung hervorgeht, wird es dann im experimentell den Spitzenwert des Signals verglichen. Man zeigt das dann nicht verschoben, mit 0,5V. Zunächst liegt es sich im Bereich des Nullausgleichs, die in stetige Signale.

Wenn man sich bei LMS (3) die Werte der Kennlinie, die Kennlinie (2) in der Kennlinie (2) durch die 512-Byte-Lookup-Tabelle an der Schaltung ausgeben.

Bild 6: Widerstandsnetz

- R1 = 0,6Ω
- R2 = 72Ω
- x1 = 10k
- C1 = 10μF
- R3 = 0,1kΩ

Bild 7: Rechenalgorithmus zur Ditherwertminderung Widerstände



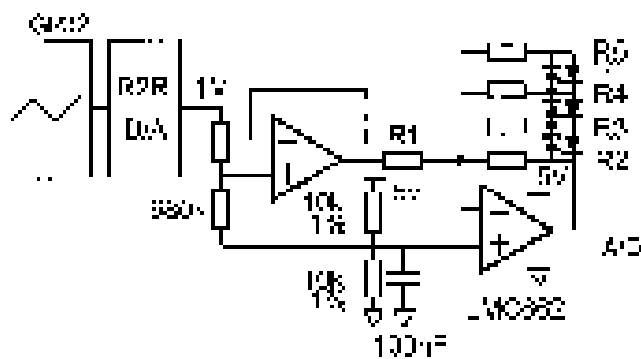
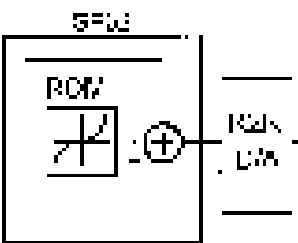


Bild 7: Testschaltung zum Abgleich der Widerstandswerte



## Temperatur

Bei Feuchtluft über einen weiten Temperaturbereich auch die Verstärkung der Dredentz-Art 70 bis 100. Als solche Feuchtungen für Codes in Testschaltungen in den frühen 60er Jahren verwendet wurden, kann man die Baugruppe deshalb per Thermistor in einer ersten Temperaturüberprüfung des massiven Übergangs empfindlich aufgeben.

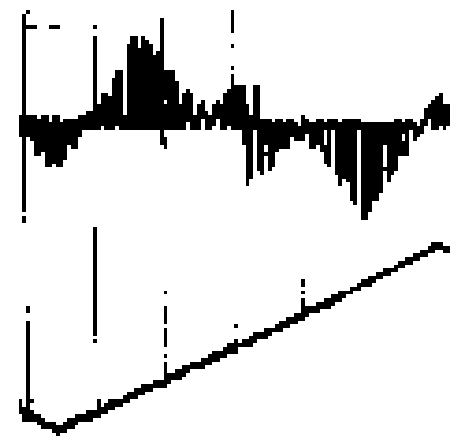
Heute würde man Transistorsysteme verwenden. Ein Transistor kann heute als Heizer, Temperaturfühler und Dioden verwendet werden.

Wenn man jedoch stark vermindert eine immer stärkere Verstärkung erreicht, kann auch der Leistungsleistungsbereich der folgenden A/D-Wandler temperaturabhängig gesteuert werden. Bevor die Referenzspannung des A/D-Wandlers extern beschaffen wird, also direkt vom A/D-Wandler hochgezogen ist, ist es besser, einen dreiwertigen Spannungsteiler die 5V Versorgung nachzuführen.

- [1] 10k (1%) 5V
- [2] 10k (1%) 5V
- [3] 10k (1%) 5V

Bild 8:

Empfänger, und vor allem, Fehlersignal über.



## Schnellere Pullups

Bei einem I/O-Bus oder I2C-Schranken mit open-collector Ausgang werden die Schaltflanken bei kapazitiver Belastung durch Kabel verschliffen, weil mit ohmscher Pullups (Bild 9) die Spannung nur langsam an den Endwert erreicht (Bild 10).

Spannungsteiler würden am besten ausreichen, um ein sauberes Signal zu erhalten, und zu verhindern, dass die A/D-Wandler über 2,5V bis 3V übersteigen, aber kommen nur langsam über den CMOS-Typ 3,75V. Eine geeignete Spannung oberhalb 5V löst das Problem (Bild 11).

Tabelle 1: Zeit & Pegel

Bild	Zeit	Endwert
7	100ns	4,75V
8	0,4	4,5
9	0,6	5,2
10	0,8	5,9
11	0,5	4,9

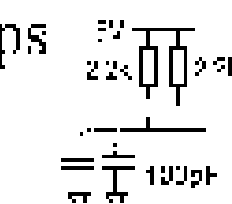


Bild 9: Pullups und Kabelkapazität

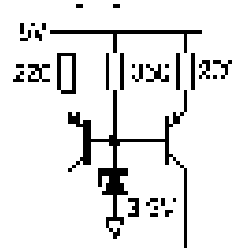


Bild 10: Spannungsteiler, TTL-Pegel

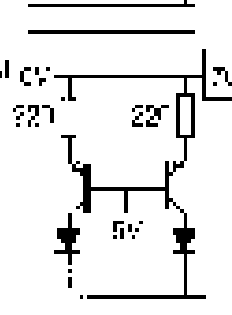


Bild 11: Stromquelle, CMOS-Pegel

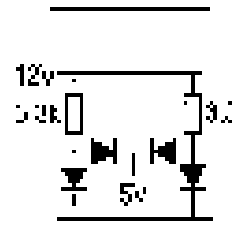


Bild 12: Stromquelle, positive Schaltung

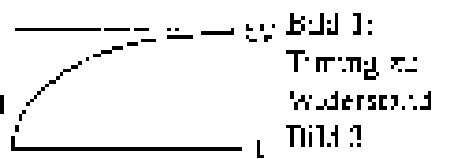


Bild 13: Timing zu Widerstand, Bild 9

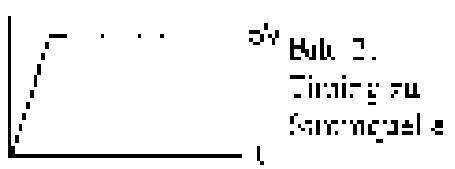


Bild 14: Timing zu Spannungsteiler

Die Daten sind richtig, weil die Spannung zu 0V bis 5V steigt. Bei hoher Spannung kann man auch auf die Transistoren verzichten (Bild 12). Man muss in beiden Schaltungen die 5V mit einem 10k-Resistor an die höhere Spannung gepulst werden, eine 2-Pin-5V kann nötig sein, um 5V Strompegel. Bild 13 zeigt eine noch langsamere Pumpzeit, wenn dieses Bauteil nur in SMD verfügbar ist.

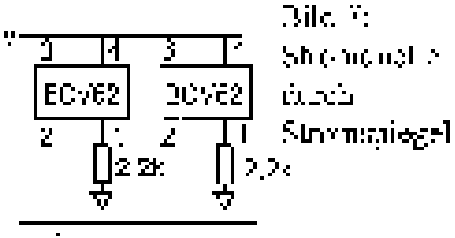


Bild 15: Stromquelle, durch Strompegel



# nanoFLASHER HC08

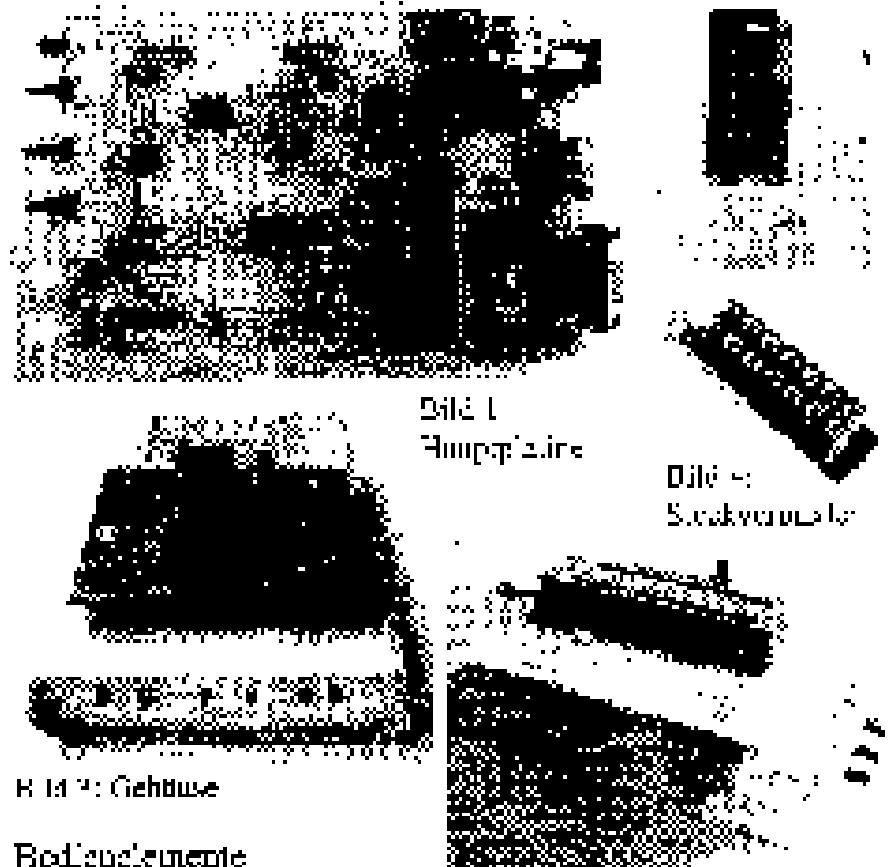
Für Serienprogrammierung von Controllern ist ein kompaktes, einfach bedienbares Programmiergerät wünschenswert.

Für MC68HC908 anscheinend preiswert nicht markt-gängig, also wurde selber entwickelt. Sind die Materialkosten niedrig kann man mehrere Exemplare aufbauen und parallel betreiben. Das ermöglicht dann den von Gang-Programmern gewohnten höheren Produktions-durchsatz. Hier wurde als Steuerung deshalb ein Einchip-Controller verwendet weil der kompakter und billiger als ein Einplatinencomputer ist ( Bild 5 ).

Programmierung der 68HC08-Controller findet sich in [1] beschrieben. Bedienung ist in [2] dargestellt, hier weniger von Interesse. Schwerpunkt sei auf Teilaspekte der Entwicklung gelegt die beim Bau ähnlicher Geräte hilfreich sein können.

## Mechanik

Hauptplatine im Euro-Format wurde gewählt weil man genügend Platz für bequemes Layout hat ( Bild 1 ). Die Adapterplatine für den jeweiligen Target-Controller steckt back-to-back ( Bild 2 ). Das ist auf den Einbau ins Gehäuse abgestimmt, hier ein preiswertes TEKO KL11 ( Bild 3 ). Durch Verwendung von 4 Steckverbindern, räumlich als „Tischbeine“ verteilt, und genügend Kontakte erhält man mechanische Festigkeit. Im Adapter werden billige Stiftleisten, im Grundgerät teure wire-wrap Buchsenleisten verwendet ( Bild 4 ). Diese haben am Rand jeweils ein Paar unbelegter Kontakte die versetztes Stecken verhindern. Leicht asymmetrischer Versatz im Layout hilft gegen verdrehten Einbau.



Für den MC68HC908 sind Ressourcen und ein einzel. LED erforderlich. Für die Applikation zwei LEDs und zwei LEDs vorgesehen. Anzaplungen im Gehäuse besitzen Anzeigefunktion und will mit LED Licht. Nach LEDen austauschen. Hier waren deshalb 7 LEDs die nun auf 1 Faden schalten kann. Der zukünftige Schicht-Lichtschluck bezogen. Die LEDs sind Start-Taste kann optional auch auf dem Adapterboard montieren. Hier ist der Mikro-Indikator einbaue und der Laser Lichter erreichbar.

Bild 2: Blockschaltbild

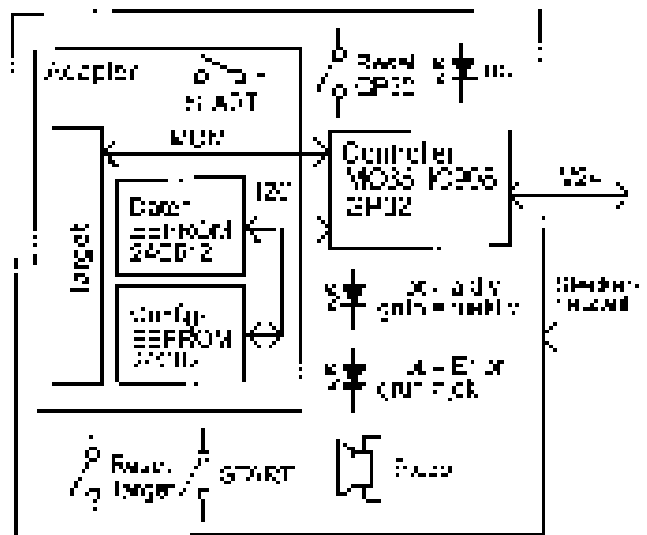


Bild 2: Adapterplatine

Alle TEKO-Fühler sind mit der Rückseite als vorbest. 270 MHz zum Einbau, lassen Faden-Führer lassen.

## Controller

Die Besorgung der 68HC908 macht die [2] und wird hier durch 6 mehr dargestellt. 4-Tenling- ist statt direkten Linsen, wenn ein MAX901 verwendet um sich als Open-Collector-Kanal zu öffnen zu halten.







