

Implementierung und Untersuchung eines  
bitgenauen Modells für einen digitalen MSK  
Transceiver Chip

Michael Hanrath

Juli 1999

170180  
Diplomarbeit  
Betreuer:  
Dipl.-Ing. Ralf Mehlan  
Prof. Dr. H. Meyr

# Inhaltsverzeichnis

<b>1</b>	<b>Der Transmitter</b>	<b>4</b>
1.1	ideales MSK-Signal . . . . .	6
1.1.1	mathematische Beschreibung . . . . .	6
1.1.2	Spektrum . . . . .	6
1.1.3	Erzeugung . . . . .	8
1.2	Mischung und Bandbegrenzung . . . . .	10
1.2.1	Erzeugung des Bandpaß-Signals . . . . .	10
1.2.2	ILO . . . . .	11
1.2.3	Fehler im Spektrum . . . . .	12
1.2.4	Auswertung . . . . .	14
1.3	zeitdiskreter MSK-Modulator . . . . .	16
1.3.1	Erzeugung . . . . .	16
1.3.2	Spektrum . . . . .	17
1.3.3	Fehler im Spektrum . . . . .	19
1.3.4	Auswertung . . . . .	20
1.3.5	Vergleich im Zeitbereich . . . . .	20
1.4	Eigenschaften des Interpolators . . . . .	24
1.4.1	Tiefpaßcharakter des rect-Interpolators . . . . .	25
1.4.2	si-Verzerrungen . . . . .	26
1.4.3	1/si-Filter . . . . .	27
1.4.4	Auswirkungen der si-Verzerrungen . . . . .	28
1.4.5	Varianz des Phasenfehlers . . . . .	30
1.5	Bitfehlerraten ohne Quantisierung . . . . .	31
1.6	digitale Erzeugung (wertdiskret) . . . . .	33
1.6.1	effektive (DA-) Auflösung . . . . .	33
1.6.1.1	Auswirkung der Quantisierung auf die Phase . . . . .	34
1.6.1.2	Tiefpaßfilterung der verzerrten Phase . . . . .	35
1.6.1.3	Quantisierung im Bandpaßbereich . . . . .	37
1.6.1.4	Verwandtschaft mit Spezialform von OQPSK . . . . .	39
1.6.2	Phasengenerator . . . . .	39
1.6.3	Sinustabelle . . . . .	40
1.6.3.1	polar $\rightarrow$ rechtwinklig . . . . .	40
1.6.3.2	Mischoszillator . . . . .	40
1.6.4	Implementierungsvorschlag . . . . .	41

1.7	Bitfehlerraten nach DA-Quantisierung . . . . .	41
1.7.1	Modell 1 . . . . .	43
1.7.2	Modell 2 und 3 . . . . .	44
1.8	Dämpfungsverluste durch den ILO . . . . .	45
1.8.1	Austausch: Spektraldämpfung gegen BER . . . . .	49
<b>2</b>	<b>Der Receiver</b>	<b>51</b>
2.1	Das Empfangsfilter $H(z)$ . . . . .	51
2.1.1	Beseitigung des dc-Offsets . . . . .	53
2.1.2	Herausfiltern der Misch-Spiegelfrequenzen . . . . .	53
2.2	Wortbreiten der Verarbeitungsstufen . . . . .	54
2.2.1	AD-Wandler . . . . .	55
2.2.1.1	Bitfehlerraten mit Frequenzoffset und Taktfehler	55
2.2.1.2	Eingangswertebereich und dc-Offset . . . . .	56
2.2.2	Mischung . . . . .	56
2.2.3	FIR-Empfangsfilter . . . . .	58
2.2.4	R→P-Umwandlung nach Empfangsfilter . . . . .	58
2.3	Der Schätzer . . . . .	60
2.3.1	Theorie . . . . .	60
2.3.1.1	Taktoffsetschätzung . . . . .	60
2.3.1.2	Frequenzoffsetschätzung . . . . .	60
2.3.1.3	modifizierte Frequenzoffsetschätzung für kleine $\Delta f$ . . . . .	62
2.3.2	Implementationsaspekte . . . . .	63
2.3.2.1	Wortbreiten im Schätzer . . . . .	63
2.3.2.2	Architekturen . . . . .	67
2.4	Rahmensynchronisation und Präambeldetektion . . . . .	71
2.4.1	Architekturen . . . . .	71
2.4.1.1	komplette Präambeldetektion mit festen Schätz- werten . . . . .	71
2.4.1.2	stark vereinfachtes Verfahren . . . . .	72
2.4.2	Präambeldetektion ohne Taktoffsetschätzwert . . . . .	72
2.4.2.1	Hard-Decision vor Korrelation . . . . .	72
2.4.2.2	Soft-Decision vor Korrelation . . . . .	75
2.4.2.3	Korrelation mit Differenzphasenverlauf . . . . .	75
2.4.3	Optimierung der Präambel . . . . .	75
2.4.3.1	Korrelationseigenschaften . . . . .	75
2.4.3.2	Schätzerperformance . . . . .	76
2.4.4	Präambeldetektion mit Taktoffsetschätzwert . . . . .	82
2.4.4.1	Dezimationsverfahren . . . . .	82
2.4.4.2	Summationsverfahren . . . . .	83
2.4.5	Präambeldetektionswahrscheinlichkeiten . . . . .	84
2.4.5.1	Missed Hit . . . . .	84
2.4.5.2	False-Alarm . . . . .	86
2.5	Abschließende Simulation auf bandbegrenztem AWGN-Kanal . .	87
2.6	Zusammenfassung wichtiger Ergebnisse . . . . .	88

<b>3 Kanaleigenschaften und Fehlerkorrektur</b>	<b>91</b>
3.1 Kanaleigenschaften . . . . .	91
3.1.1 mittleres SNR . . . . .	91
3.1.2 Fading während der Paketübertragung . . . . .	94
3.1.3 Simulationsergebnisse mit DLR-Kanälen . . . . .	95
3.2 Fehlerkorrekturverfahren . . . . .	97
3.2.1 Allgemeines zur Codierung . . . . .	99
3.2.2 Hamming-Codes . . . . .	100
3.2.2.1 Theorie . . . . .	100
3.2.2.2 Implementation . . . . .	101
3.2.3 BCH-Codes . . . . .	102
3.2.3.1 Theorie . . . . .	102
3.2.3.2 Implementation . . . . .	106
3.2.4 Nutzung von Erasure-Informationen . . . . .	110

# Kapitel 1

## Der Transmitter

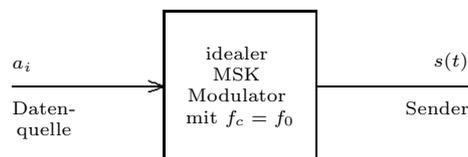
In diesem Kapitel werden die Eigenschaften des MSK-Senders untersucht, die sich aufgrund der digitalen Implementierung ergeben. Abbildung 1 zeigt einen idealen MSK-Modulator, der direkt das Bandpaßsignal mit der Mittenfrequenz von  $f_0 = 64$  GHz erzeugt. Ziel ist es, diesen idealen MSK-Modulator mit Mitteln nachzubilden, die eine kostengünstige und möglichst weitgehend digitale Implementation zulassen.

Abbildung 1.1 zeigt die Struktur des Senders, wie sie implementiert werden soll, und wie sie sich für eine theoretische Analyse eignet. Die Erzeugung des MSK-Signals durch Mischung der Trägerfrequenz mit den Quadraturkomponenten soll noch im digitalen Teil erfolgen. Da es nicht möglich ist, im digitalen Teil direkt ein MSK-Signal zu erzeugen, dessen Mittenfrequenz  $f_0$  bei 64 GHz liegt, muß ein Umweg über zwei Zwischenfrequenzen gewählt werden. Das 64 GHz Signal, das zur Antenne geht, wird durch einen Injection Logic Oscillator generiert. Dieser erfordert neben der Trägerfrequenz ein Ansteuerungssignal, dessen Frequenz  $f_2$  im Bereich von 70 MHz liegt. Um nicht zu hohe Abtastraten zu erfordern, wird eine zweite Mischstufe mit der Zwischenfrequenz  $f_2$  nötig, die das Signal des digitalen MSK-Modulators der Mittenfrequenz  $f_c = f_1$  dann auf  $f_2$  transformiert.

Die Zwischenfrequenz  $f_1$  kann größenordnungsmäßig im Bereich zwischen  $\frac{1}{T} \dots \frac{10}{T}$  liegen, wobei  $\frac{1}{T}$  die Symbolrate darstellt.

Durch diese Struktur des Senders ergeben sich im Vergleich zum idealen Signal  $s(t)$  mehrere Fehler im Spektrum des Signals  $\acute{s}(t)$ .

### 1. Fehler durch Mischen



captionidealer MSK Modulator

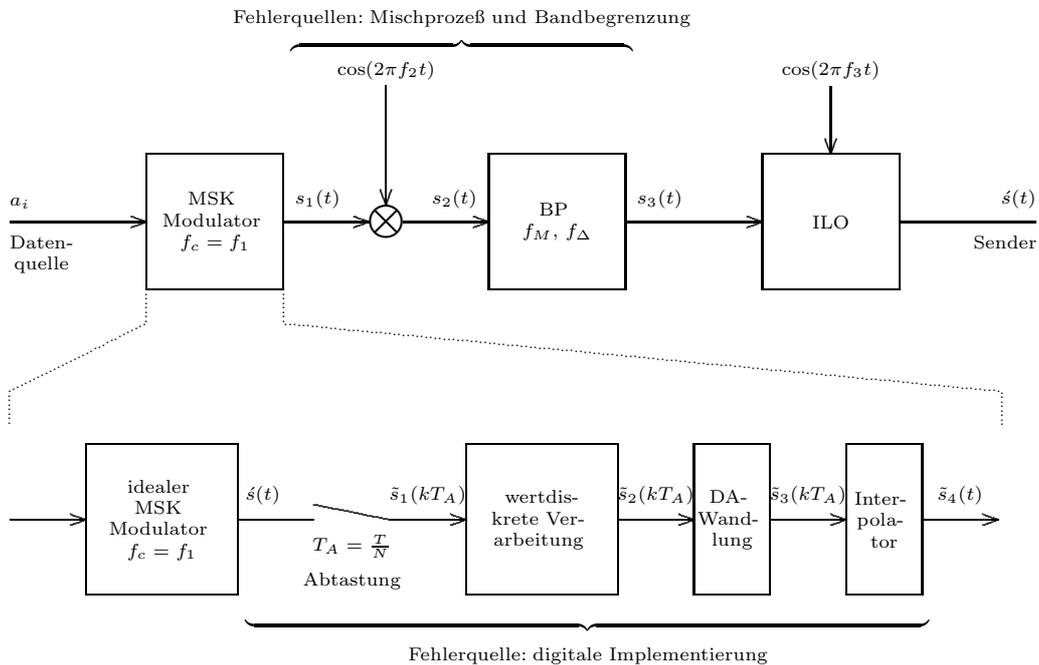


Abbildung 1.1: nichtidealer MSK Modulator

- (a) Überlappungseffekte im Spektrum
- (b) Bandbegrenzung

## 2. Fehler durch digitale Implementation

- (a) Abtastung mit dem  $N$ -fachen der Symbolrate  $T_A = \frac{N}{T}$
- (b) diskrete Realisierung des Modulators
- (c) Ausgangsquantisierung und si-Verzerrungen des DA-Wandlers

Die Untersuchung dieser einzelnen Fehlerquellen gliedert dieses Kapitel. Dabei ist die Reihenfolge der einzelnen Fehlerursachen von Bedeutung, weil einzelne Komponenten in der Signalverarbeitung die Bedingung der Zeitinvarianz bzw. der Linearität nicht erfüllen.

Das Modell des MSK-Senders beinhaltet zu viele Parameter, um sie alle gleichzeitig untersuchen zu können. Daher werden bestimmte Parameter mit den von ihnen abhängigen Fehlern zusammengefaßt untersucht. Während sich Abschnitt 1.1 mit dem idealen Fall eines MSK-Modulators beschäftigt und für alle weiteren Abschnitte als Vergleichsbasis dient, beschäftigt sich Abschnitt 1.2 mit einem Modulator, der dem oberen Teil in Abbildung 1.1 entspricht. Er weist daher noch keine digitalen Komponenten auf, dient jedoch wiederum für die nachfolgenden Abschnitte als Vergleichsbasis.

## 1.1 ideales MSK-Signal

### 1.1.1 mathematische Beschreibung

Das MSK-Signal läßt sich schreiben als:

$$s(t) = \Re\{e^{j(2\pi f_c t + \varphi(t))}\} \quad (1.1)$$

Darin ist  $f_c$  die Mittenfrequenz des MSK-Signals und

$$\varphi(t) = 2\pi h \sum_i b_i q(t - iT) \quad (1.2)$$

mit

$$h = \frac{1}{2} \quad (1.3)$$

$$q(t) = \begin{cases} 0 & t < 0 \\ \frac{t}{2T} & \text{für } 0 \leq t \leq T \\ \frac{1}{2} & t > T \end{cases} \quad (1.4)$$

$$b_i \in \{-1, 1\} \quad (1.5)$$

Dabei erfolgt eine Abbildung

$$b_i = \begin{cases} -1 & \text{für } a_i = 0 \\ +1 & \text{für } a_i = 1 \end{cases} \quad (1.6)$$

In Abbildung 1.2 ist dieser funktionale Zusammenhang  $q(t)$  dargestellt. Abbildung 1.3 zeigt ein MSK-Signal für die Datensequenz  $\{01011110\}$ .

### 1.1.2 Spektrum

Die Berechnung des Spektrums des MSK-Signals erfolgt auf direktem Wege durch Einsetzen von (1.1) in die Fouriertransformation.

$$H(f) = \lim_{T' \rightarrow \infty} \frac{1}{2T'} \int_{-T'}^{T'} s(t) \cdot e^{-j2\pi f t} dt \quad (1.7)$$

[ ...

$$H(f) = \lim_{n \rightarrow \infty} \frac{1}{(2n+1)T} \sum_{i=-n}^n e^{2\pi j b_i} \cdot \left[ \frac{1}{2\pi j(m_i - f)} e^{j2\pi(m_i - f)t} \right]_{iT}^{(i+t)T} \quad (1.8)$$

]

In [1] ist das Leistungsdichtespektrum für ein MSK-Signal berechnet, das durch weiße Daten erzeugt wurde. Es wird von der Funktion

$$s(t) = \Re\{v(t)e^{j2\pi f_c t}\} \quad (1.9)$$

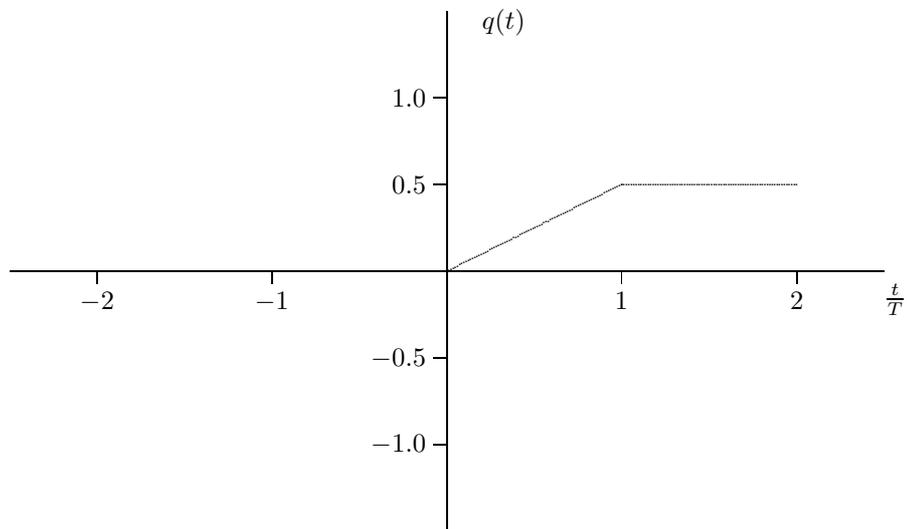
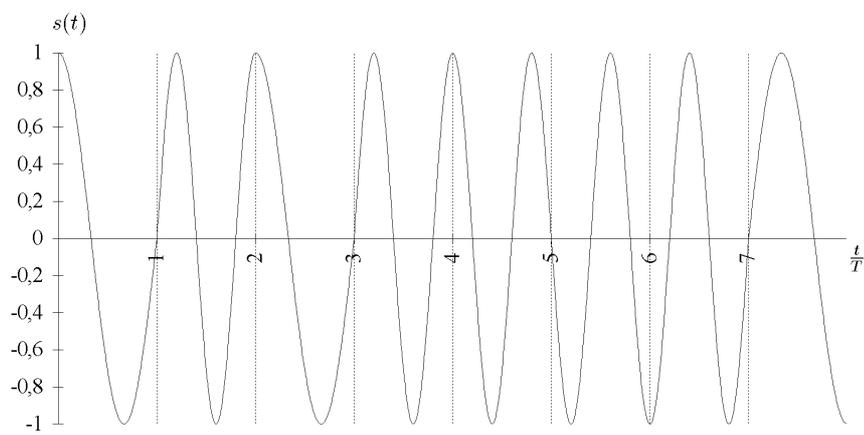
Abbildung 1.2: Modulationsfunktion  $q(t)$ 

Abbildung 1.3: ideales MSK-Signal

ausgegangen. Durch die Wahl von  $v(t) = e^{j\varphi(t)}$  sind (1.1) und (1.9) äquivalent. Die Berechnung erfolgt in [1] indirekt über die Autokorrelationsfunktion von  $s(t)$ .

$$\varphi_{ss}(\tau) = s(\tau) * s(-\tau) = \Re\{\varphi_{vv}(\tau)e^{j2\pi f_c \tau}\} \quad (1.10)$$

Die Fourier-Transformation von (1.10) liefert das gesuchte  $\Phi_{ss}(f)$ :

$$\Phi_{ss}(f) = \frac{1}{2}\Phi_{vv}(f) * (\delta(f + f_c) + \delta(f - f_c)) = \frac{1}{2}(\Phi_{vv}(f - f_c) + \Phi_{vv}(f + f_c)) \quad (1.11)$$

Darin ist  $\Phi_{vv}(f)$  das Leistungsdichtespektrum von  $v(t)$ .  $\Phi_{vv}(f)$  wird mit einigem Aufwand in [1] berechnet und wird zu

$$\Phi_{vv}(f) = A^2 T \frac{16}{\pi^2} \left( \frac{\cos(2\pi f T)}{1 - 16f^2 T^2} \right)^2 \quad (1.12)$$

bestimmt. Darin ist  $T$  die Symboldauer und  $A$  die Amplitude des Signals.

Für  $\Phi_{ss}(f)$  ergibt sich damit

$$\Phi_{ss}(f) = A^2 T \frac{8}{\pi^2} \left( \left( \frac{\cos(2\pi(f - f_c)T)}{1 - 16(f - f_c)^2 T^2} \right)^2 + \left( \frac{\cos(2\pi(f + f_c)T)}{1 - 16(f + f_c)^2 T^2} \right)^2 \right) \quad (1.13)$$

Für den idealen MSK-Modulator nach Abbildung 1 gilt für die Mittenfrequenz  $f_c$ :

$$f_c = f_0 = 64 \text{ GHz}. \quad (1.14)$$

Abbildung 1.4 zeigt das Leistungsdichtespektrum  $\Phi_{vv}(f)$  eines weißen idealen MSK-Signals im Tiefpaßbereich.

Abbildung 1.5 zeigt das gleiche Spektrum ideal hochgemischt auf die Trägerfrequenz. Um einen Eindruck von der Bandbreiteneffizienz zu bekommen, ist der Maßstab auf der Ordinate nicht logarithmisch. Dabei wurde auf das Maximum normiert. Diese Normierung erfolgt auch bei allen weiteren Spektren.

### 1.1.3 Erzeugung

Da MSK zu den Phasenmodulationsverfahren gehört ist die direkte Implementierung von (1.1) aufwendig. Daher ist in [1] und [2] eine Realisation beschrieben, die auf den trigonometrischen Additionstheoremen beruht. Ausgehend von (1.1) folgt:

$$\begin{aligned} s(t) &= \Re\{e^{j2\pi f_c t} \cdot e^{j\varphi(t)}\} \\ &= \cos(2\pi f_c t) \cdot \cos(\varphi(t)) - \sin(2\pi f_c t) \cdot \sin(\varphi(t)) \end{aligned} \quad (1.15)$$

Dies bedeutet, daß die Phasenmodulation durch 2 Multiplikationen mit sin- bzw. cos-Signalen und eine Addition realisiert werden kann. Abbildung 1.6 zeigt diesen Modulator.

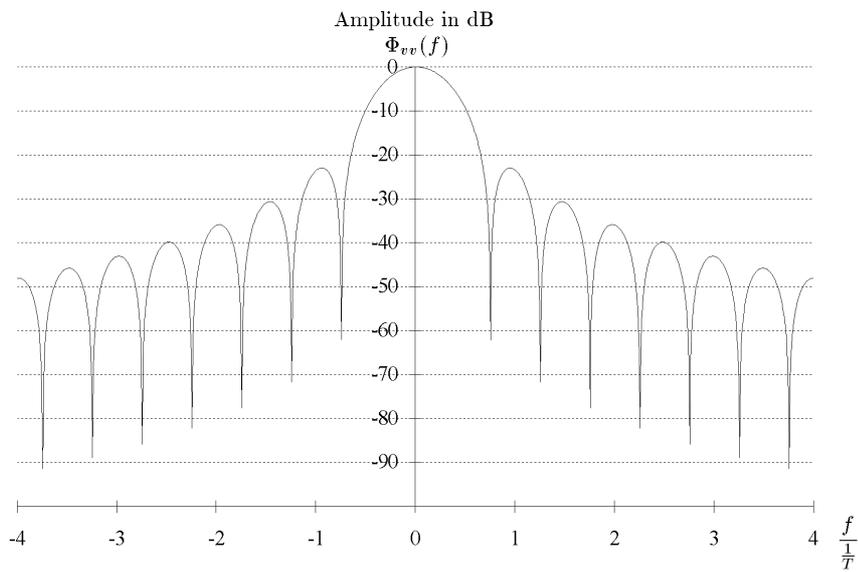


Abbildung 1.4: ideales MSK-Leistungsdichtespektrum im Tiefpaßbereich

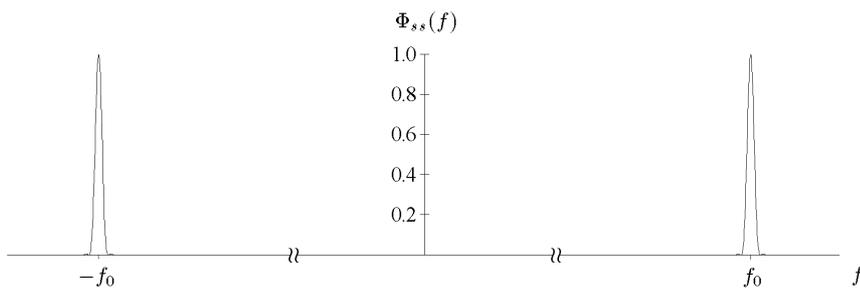


Abbildung 1.5: ideales MSK-Leistungsdichtespektrum auf Trägerfrequenz

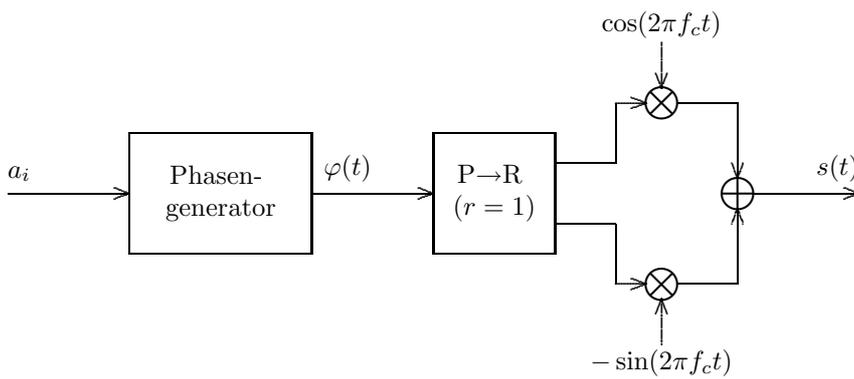
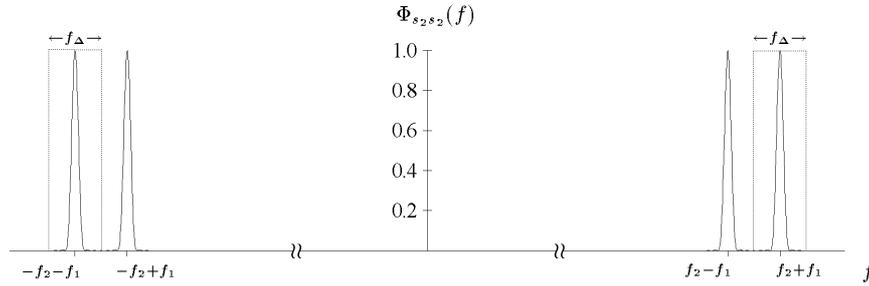


Abbildung 1.6: idealer MSK-Modulator

Abbildung 1.7: Leistungsdichtespektrum nach Mischung auf  $f_2$ 

## 1.2 Mischung und Bandbegrenzung

Der obere Teil in Abbildung 1.1 wird nun zunächst unter der Annahme untersucht, daß der MSK-Modulator am Eingang ideal ist. Die Expansion desselben erfolgt im nächsten Abschnitt.

### 1.2.1 Erzeugung des Bandpaß-Signals

Zunächst erfolgt eine Transformation des MSK-Tiefpaßsignals in den Bandpaßbereich auf die Mittenfrequenz  $f_1$ . Dies entspricht den Operationen in Abschnitt 1.1.2.

$$\Phi_{s_1 s_1}(f) = \frac{1}{2} \Phi_{vv}(f) * (\delta(f - f_1) + \delta(f + f_1)) \quad (1.16)$$

Danach schließt sich das Mischen mit der Zwischenfrequenz  $f_2$  an.

$$\begin{aligned} \Phi_{s_2 s_2}(f) &= \frac{1}{2} \Phi_{s_1 s_1}(f) * (\delta(f - f_2) + \delta(f + f_2)) \\ &= \frac{1}{4} \Phi_{vv}(f) * (\delta(f - f_1) + \delta(f + f_1)) * (\delta(f - f_2) + \delta(f + f_2)) \\ &= \frac{1}{4} (\Phi_{vv}(f - (f_2 + f_1)) + \Phi_{vv}(f - (f_2 - f_1)) \\ &\quad + \Phi_{vv}(f + (f_2 + f_1)) + \Phi_{vv}(f + (f_2 - f_1))) \end{aligned} \quad (1.17)$$

Nun folgt die Bandpaßfilterung:

$$\Phi_{s_3 s_3}(f) = \Phi_{s_2 s_2}(f) \cdot \left( \text{rect} \left( \frac{f - f_M}{f_\Delta} \right) + \text{rect} \left( \frac{f + f_M}{f_\Delta} \right) \right) \quad (1.18)$$

mit  $f_M = f_2 + f_1$ .

Abbildung 1.7 zeigt  $\Phi_{s_2 s_2}(f)$  auf das Maximum normiert, wobei  $f_1 = \frac{2}{T}$  und  $f_\Delta = \frac{4}{T}$  angenommen wurde, um das Spektrum zeichnen zu können. Zusätzlich ist noch die Lage des Bandpasses gestrichelt eingezeichnet, so daß auch  $\Phi_{s_3 s_3}(f)$  erkennbar ist.

## 1.2.2 ILO

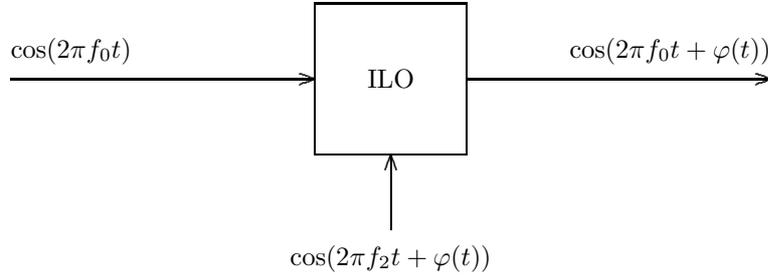


Abbildung 1.8: Funktionsweise des ILO

Abbildung 1.8 zeigt die Funktionsweise des ILO im Zeitbereich. Dies soll nun im Frequenzbereich untersucht werden.

$$\begin{aligned}
 h(t) &= \cos(2\pi f' t + \varphi(t)) \\
 &= \Re\{e^{j(2\pi f' t + \varphi(t))}\} \\
 &= \frac{1}{2} \left( e^{j2\pi f' t} \cdot e^{j\varphi(t)} + e^{-j2\pi f' t} \cdot e^{-j\varphi(t)} \right) \quad (1.19)
 \end{aligned}$$

$$H(f) = \frac{1}{2} (\delta(f - f') * H_{j\varphi}(f) + \delta(f + f') * H_{-j\varphi}(f)) \quad (1.20)$$

Die Fourier-Transformation von  $e^{\pm j\varphi(t)}$  ergibt:

$$H_{j\varphi}(f) = \int_{-\infty}^{\infty} e^{j(\varphi(t) - 2\pi f t)} dt \quad (1.21)$$

$$H_{-j\varphi}(f) = \int_{-\infty}^{\infty} e^{-j(\varphi(t) + 2\pi f t)} dt \quad (1.22)$$

$$\begin{aligned}
 &\implies \\
 H_{j\varphi}(f) &= H_{-j\varphi}^*(-f) \implies \\
 H_{-j\varphi}(f) &= H_{j\varphi}^*(-f) \quad (1.23)
 \end{aligned}$$

Wird nun (1.23) auf (1.19) angewandt, folgt aus (1.20):

$$H(f) = \frac{1}{2} (\delta(f - f') * H_{j\varphi}(f) + \delta(f + f') * H_{j\varphi}^*(-f)) \quad (1.24)$$

Aus (1.24) ergeben sich die für reelle Stoßantworten notwendigen Symmetrien in  $H(f)$ .

Wird für  $f'$   $f_2$  bzw.  $f_3$  eingesetzt, folgt, daß der ILO das Spektrum  $H_{\varphi}(f)$  von  $\varphi(t)$  auf der Frequenzachse von  $f_2$  nach  $f_3$  verschiebt. Es handelt sich dabei um

keinen normalen Mischprozeß, da keine Spiegelfrequenzen auftreten. Der ILO substituiert daher eine an seinem Signaleingang vorhandene Trägerfrequenz ( $f_2$ ) durch seine eigene ( $f_0$ ).

In Abbildung 1.7 ist zu erkennen, daß  $\Phi_{s_3s_3}$  eine neue Mittenfrequenz  $f_2 + f_1$  besitzt. Diese wird durch den ILO verschoben. Für (1.18) bedeutet dies, daß  $f_2 + f_1$  durch  $f_3$  substituiert werden muß. Damit folgt:

$$\begin{aligned} \Phi_{\dot{s}\dot{s}}(f) &= \frac{1}{4}(\Phi_{vv}(f - f_3) + \Phi_{vv}(f - (f_3 - 2f_1)) \\ &\quad + \Phi_{vv}(f + f_3) + \Phi_{vv}(f + (f_3 - 2f_1))) \\ &\quad \cdot \left( \text{rect} \left( \frac{f - f_{M'}}{f_{\Delta}} \right) + \text{rect} \left( \frac{f + f_{M'}}{f_{\Delta}} \right) \right) \end{aligned} \quad (1.25)$$

mit  $f_{M'} = f_3$ .

### 1.2.3 Fehler im Spektrum

Der Fehler  $E$  im Spektrum sei:

$$\begin{aligned} E &= \int_{-\infty}^{\infty} |\Phi_{ss}(f) - \Phi_{\dot{s}\dot{s}}(f)| df \\ &= 2 \int_0^{\infty} |\Phi_{ss}(f) - \Phi_{\dot{s}\dot{s}}(f)| df \end{aligned} \quad (1.26)$$

Wegen der völligen Symmetrie zu  $f = 0$  in sämtlichen Spektren ist diese Umformung möglich.

Die Integration des MSK-Leistungsdichtespektrums liefert einen recht unhandlichen Ausdruck:

$$\begin{aligned} \int \left( \frac{\cos(2\pi x)}{1 - 16x^2} \right)^2 dx &= \\ \frac{1}{32} &\left( Ci(4\pi x - \pi) + \pi \left( Si(4\pi x - \pi) - \frac{\cos(4\pi x)}{4\pi x - \pi} \right) \right. \\ &- Ci(4\pi x + \pi) + \pi \left( Si(4\pi x + \pi) - \frac{\cos(4\pi x)}{4\pi x + \pi} \right) \\ &\left. - \ln(4\pi x - \pi) + \ln(4\pi x + \pi) - \frac{1}{4x - 1} - \frac{1}{4x + 1} \right) + C \end{aligned}$$

Daher werde diese Stammfunktion im folgenden nicht explizit ausgeschrieben sondern nur numerisch ausgewertet, zumal sie den nicht ausintegrierbaren Integralsinus und -cosinus enthält.

Zur Vereinfachung der Rechnung werde bei  $\Phi_{ss}$  und  $\Phi_{\dot{s}\dot{s}}$  der linke Teil des Spektrums unberücksichtigt gelassen. Dies ist jedoch keine schwerwiegende Veränderung, da die Amplitude des linken Teils für  $f_0 = 64$  GHz und  $\frac{1}{T} \approx 1$  MHz auf  $-228$  dB abgefallen ist. Zusätzlich wird zum Vergleich eine Amplitudenkorrektur durchgeführt.

$$\implies \Phi_{ss}(f) \mapsto \tilde{\Phi}_{ss}(f) = \Phi_{vv}(f - f_3) \quad (1.27)$$

und

$$\begin{aligned} \implies \Phi_{\dot{s}\dot{s}}(f) \mapsto \tilde{\Phi}_{\dot{s}\dot{s}}(f) &= (\Phi_{vv}(f - f_3) + \Phi_{vv}(f - (f_3 - 2f_1))) \\ &\cdot \text{rect}\left(\frac{f - f_{M'}}{f_\Delta}\right) \end{aligned} \quad (1.28)$$

Damit wird

$$E \approx 2 \int_0^\infty |\tilde{\Phi}_{ss}(f) - \tilde{\Phi}_{\dot{s}\dot{s}}(f)| df. \quad (1.29)$$

Dieses Integral kann in zwei Anteile aufgespalten werden: Einen außerhalb und einen innerhalb des Bandpasses. Derjenige außerhalb berechnet sich zu

$$\begin{aligned} E_1 &= 2 \int_0^{f_3 - \frac{f_\Delta}{2}} \Phi_{vv}(f - f_3) df + 2 \int_{f_3 + \frac{f_\Delta}{2}}^\infty \Phi_{vv}(f - f_3) df \\ &= 2 \int_{-f_3}^{-\frac{f_\Delta}{2}} \Phi_{vv}(f) df + 2 \int_{\frac{f_\Delta}{2}}^\infty \Phi_{vv}(f) df \end{aligned} \quad (1.30)$$

$E_1$  ist unabhängig von  $f_1$  und es gilt

$$\lim_{f_\Delta \rightarrow \infty} E_1 = 0 \quad (1.31)$$

sowie

$$\lim_{f_\Delta \rightarrow 0} E_1 = 1. \quad (1.32)$$

Zusätzlich ist  $E_1$  bezüglich  $f_\Delta$  monoton fallend:

$$\frac{d}{df_\Delta} E_1(f_\Delta) < 0 \quad (1.33)$$

Der Fehler innerhalb des Bandpasses:

$$\begin{aligned} E_2 &= 2 \int_{f_3 - \frac{f_\Delta}{2}}^{f_3 + \frac{f_\Delta}{2}} \Phi_{vv}(f - (f_3 - 2f_1)) df \\ &= 2 \int_{2f_1 - \frac{f_\Delta}{2}}^{2f_1 + \frac{f_\Delta}{2}} \Phi_{vv}(f) df \end{aligned} \quad (1.34)$$

Und es gilt:

$$\lim_{f_{\Delta} \rightarrow \infty} E_2 = 1 \quad (1.35)$$

Zusätzlich ist  $E_2$  bezüglich  $f_{\Delta}$  monoton steigend:

$$\frac{d}{df_{\Delta}} E_2(f_{\Delta}) > 0 \quad (1.36)$$

Der Fehler wird noch in dB umgerechnet:

$$\frac{E}{\text{dB}} = 10 \lg \left( \frac{E_1 + E_2}{\int_{-\infty}^{\infty} \Phi_{ss} df} \right) = 10 \lg(E_1 + E_2) \quad (1.37)$$

### 1.2.4 Auswertung

Zwei Parameter in (1.37) können frei variiert werden. Es sind dies die Mittenfrequenz  $f_1$  des MSK-Modulators sowie die Bandbreite  $f_{\Delta}$ .

Die Abbildungen 1.9, 1.10 und 1.11 zeigen (1.37) für  $f_1 = \frac{1}{T} \dots \frac{8}{T}$  und  $f_{\Delta} = \frac{1}{T} \dots \frac{8}{T}$ . Die ersten beiden zeigen  $E_1$  und  $E_2$  getrennt, die dritte zeigt die Summe  $E = E_1 + E_2$ .

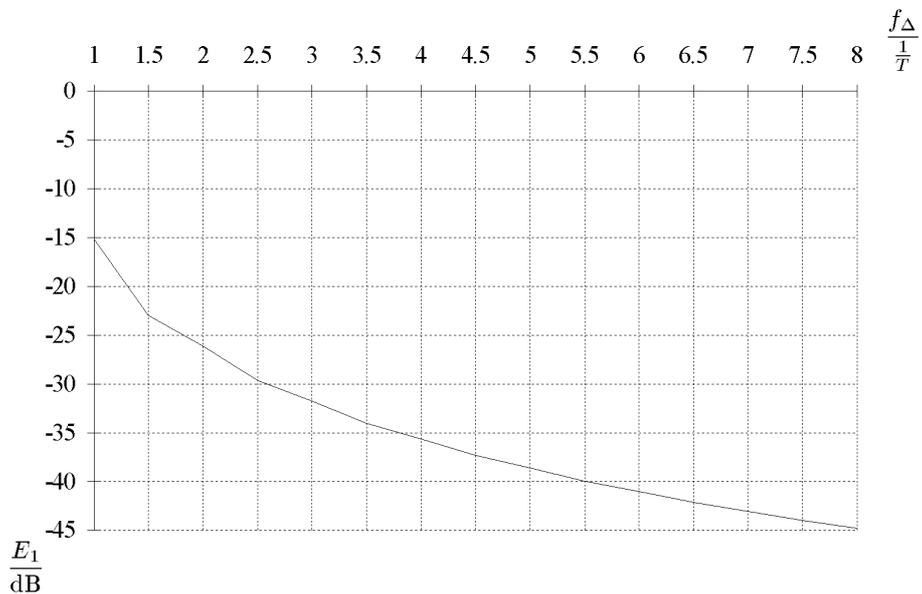
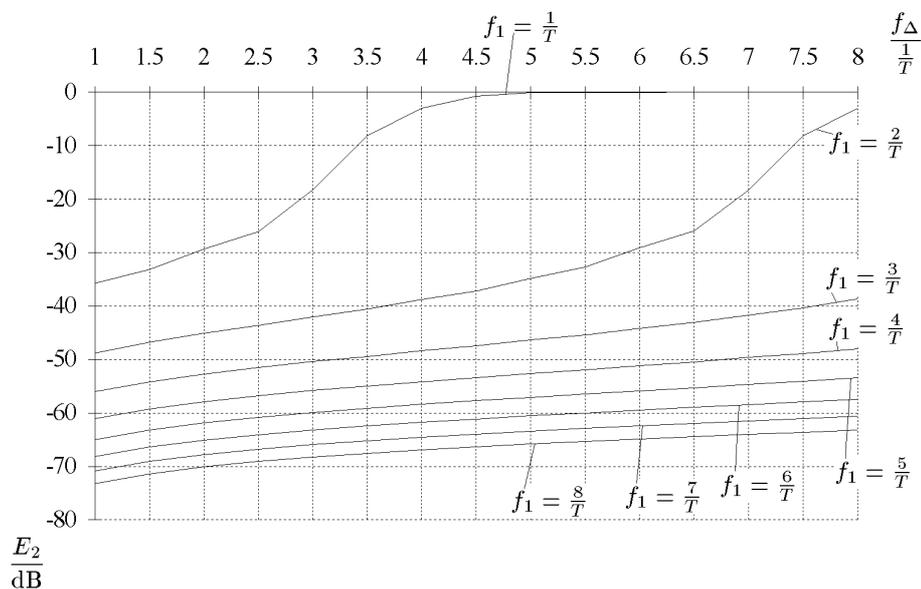


Abbildung 1.9: Fehler  $E_1$  außerhalb des Bandpasses

Es sind verschiedene Eigenschaften von (1.37) zu erkennen.

Abbildung 1.10: Fehler  $E_2$  innerhalb des Bandpasses

1. Wählt man die Bandbreite  $f_\Delta$  bei kleinem  $f_1$  zu groß, liegen die beiden Hauptmaxima rechts in Abbildung 1.7 zu eng aneinander, so daß sich das Maximum bei  $f = f_2 - f_1$  innerhalb des Bandpasses befindet.  $\Rightarrow E_2$  wird groß.
2. Wird die Bandbreite klein gewählt, sind zwar geringere Überlappungseffekte zu beobachten, jedoch wird sehr viel des Spektrums einfach weggeschnitten.  $\Rightarrow E_1$  wird groß.
3. Zwischen diesen Extrema für  $f_\Delta$  liegt ein Minimum des Fehlers, dessen Lage durch die Wahl  $f_1$  bestimmt wird. Als eine Abschätzung für die Lage des Minimums kann  $f_\Delta \approx 2f_1$  dienen. Das heißt, die untere Grenzfrequenz des Bandpasses liegt genau zwischen den beiden Hauptmaxima im Spektrum. Dieser Fall ist in Abbildung 1.7 eingezeichnet. Tabelle 3 zeigt die Minima für ausgewählte  $f_1$ .

$f_1$	$f_\Delta$	$E_{\min}$ in dB
1.0	2.2	-24.5
2.0	4.4	-34.2
3.0	6.5	-39.6
4.0	8.6	-43.3

Tabelle 1.1: Minima des Fehlers  $E$  für vorgegebene  $f_1$

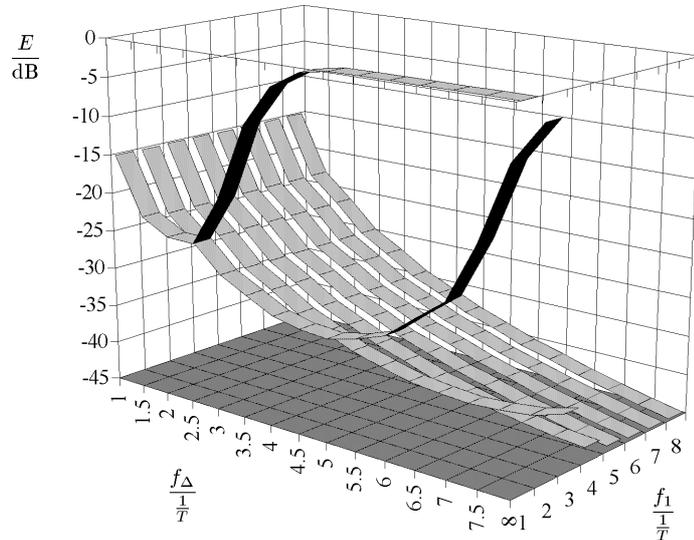


Abbildung 1.11: Fehler durch Überlappen und Mischen

Bei diesen Ergebnissen sind zwei Dinge zu beachten: Es gibt keinen idealen Bandpaß. Daher sind die Daten in Tabelle 3 nur Richtwerte, die durch die spezielle Gestalt der Übertragungsfunktion des Bandpasses beeinflusst werden. Zweitens ist die Wahl der Fehlerfunktion (1.26) willkürlich geschehen. Die Teilfehler  $E_1$  und  $E_2$  werden gleich gewichtet. Dies könnte jedoch nicht die optimale Wichtung sein.

### 1.3 zeitdiskreter MSK-Modulator

Bis jetzt wurde nur der obere Teil von Abbildung 1.1 betrachtet und der am Eingang stehende MSK-Modulator als ideal angenommen. Diese Annahme soll in diesem und den folgenden Abschnitten schrittweise fallengelassen werden. Der erste Schritt ist, den Modulator zeitdiskret zu realisieren. Wie im unteren Teil von Abbildung 1.1 zu sehen, wird einem hypothetischen idealen Modulator ein Abtaster nachgeschaltet. Dabei entsteht das Signal  $\tilde{s}_1(kT_A)$ .

#### 1.3.1 Erzeugung

Das System in Abbildung 1.6 enthält bis auf den Phasengenerator keine Komponenten mit Gedächtnis. Das heißt, daß der Ausgabewert in einem Takt nur von den Eingabewerten in diesem Takt abhängt. An der Abtastung des Signals  $s(t)$  ändert sich daher nichts, wenn auch vor den Multiplikatoren bei beiden

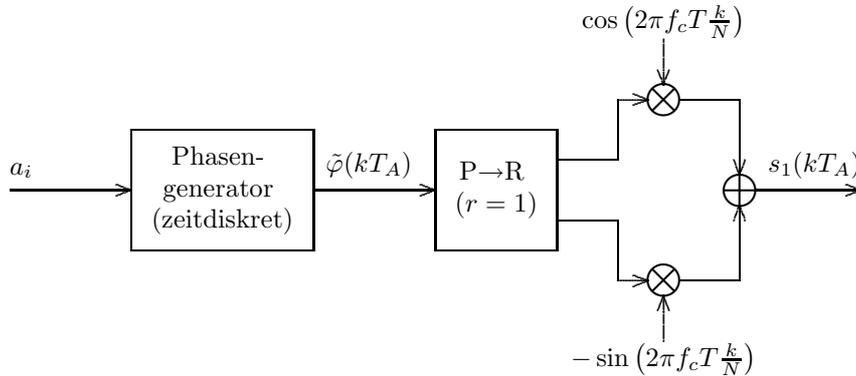


Abbildung 1.12: MSK-Modulator, zeitdiskret

Faktoren Abtaster eingesetzt werden, und wenn zusätzlich noch die Funktion  $\varphi(t)$  abgetastet wird. Lediglich beim Phasengenerator muß sichergestellt sein, daß in den Abtastzeitpunkten gilt

$$\tilde{\varphi}(kT_A) = \varphi(kT_A). \quad (1.38)$$

Dann entspricht das System in Abbildung 1.12 dem in Abbildung 1.6 mit nachgeschaltetem Abtaster. Es gilt daher

$$s_1(t) = \dot{s}(t) \cdot \frac{1}{T_A} \text{III} \left( \frac{t}{T_A} \right) \quad (1.39)$$

Die Multiplikation mit den cos- und sin-Folgen gestaltet sich für

$$\frac{N}{f_c T} = 4 \quad (1.40)$$

besonders einfach, denn es gilt

$$\cos\left(\frac{\pi}{2}k\right) = \{1 \ 0 \ -1 \ 0 \ \dots\} \text{ und } \sin\left(\frac{\pi}{2}k\right) = \{0 \ 1 \ 0 \ -1 \ \dots\}.$$

### 1.3.2 Spektrum

Damit folgt für das Spektrum

$$S_1(f) = \dot{S}(f) * \text{III}(T_A f) = \frac{1}{T_A} \sum_{n=-\infty}^{\infty} \dot{S} \left( f - \frac{n}{T_A} \right) \quad (1.41)$$

Es gilt  $\Phi_{\dot{s}\dot{s}}(f) \sim |\dot{S}(f)|^2$ . Damit folgt für das abgetastete Leistungsdichtespektrum  $\Phi_{1ss}(f)$

$$\Phi_{1ss}(f) = \frac{1}{T_A} \sum_{n=-\infty}^{\infty} \Phi_{\dot{s}\dot{s}} \left( f - \frac{n}{T_A} \right). \quad (1.42)$$

Wobei wieder gilt:

$$\Phi_{\tilde{s}\tilde{s}}(f) = \frac{1}{2}\Phi_{vv}(f) * (\delta(f - f_1) + \delta(f + f_1)) \quad (1.43)$$

Das Spektrum wird periodisch wiederholt. Da das MSK-Spektrum  $\Phi_{ss}$  nicht bandbegrenzt ist, entsteht durch die direkte Abtastung auf jeden Fall Aliasing. Abbildung 1.13 zeigt das Spektrum des Signals  $\tilde{s}_1(t)$  nach MSK-Modulation mit  $f_c = f_1 = \frac{2}{T}$  und Abtastung mit  $N = 16$ .

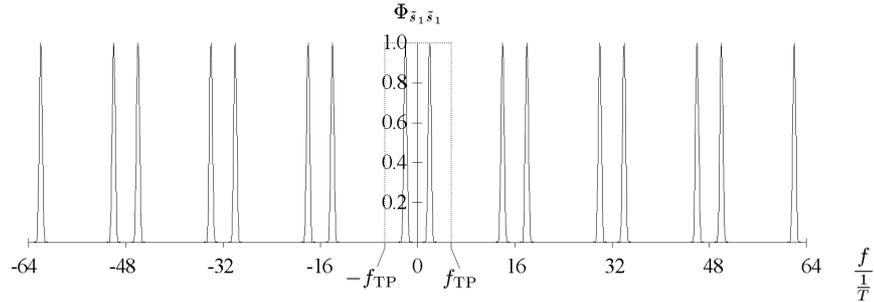


Abbildung 1.13: MSK-Spektrum für  $f_1 = \frac{2}{T}$  nach Abtastung mit  $N = 16$

In diesem Abschnitt wird zunächst eine ideale Umwandlung des Signals  $\tilde{s}_1(t)$  durch einen idealen Tiefpaß ins Zeitkontinuierliche angenommen. Die Aliasing-Effekte bleiben berücksichtigt. In Abbildung 1.13 ist bereits ein Tiefpaßfilter eingezeichnet. Die Grenzfrequenz  $f_{TP}$  sollte nicht kleiner sein als die Bandbreite  $f_\Delta$  des späteren Bandpasses, da sonst das Spektrum unsymmetrisch abgeschnitten würde. Wird  $f_{TP}$  zu groß gewählt, erscheinen nach dem Mischen auf  $f_2$  im Bereich des Bandpasses Spiegelfrequenzen von der linken Seite. Dies kann durch die Wahl von  $f_{TP} < 2f_2$  verhindert werden. Um eine möglichst unkritische Weiterverarbeitung zu ermöglichen, sollte die Grenzfrequenz des Tiefpasses jedoch relativ niedrig liegen.  $\Rightarrow f_{TP} \gtrsim f_\Delta$

In Abschnitt 1.3 soll gelten (ideale Tiefpaß-Interpolation):

$$\begin{aligned} \Phi_{\tilde{s}_4\tilde{s}_4}(f) &= \Phi_{s_1s_1}(f) = \frac{1}{T_A} \left( \sum_{n=-\infty}^{\infty} \Phi_{\tilde{s}\tilde{s}} \left( f - \frac{n}{T_A} \right) \right) \cdot \text{rect} \left( \frac{f}{2f_{TP}} \right) \\ &= \frac{1}{2T_A} \left( \sum_{n=-\infty}^{\infty} \Phi_{vv} \left( f - f_1 - \frac{n}{T_A} \right) + \Phi_{vv} \left( f + f_1 - \frac{n}{T_A} \right) \right) \\ &\quad \cdot \text{rect} \left( \frac{f}{2f_{TP}} \right) \end{aligned} \quad (1.44)$$

Nun kann die Berechnung des Fehlers im Spektrum analog zu Abschnitt 1.2 erfolgen, wenn (1.16) durch (1.44) ersetzt wird.

$$\Phi_{s_2s_2}(f) = \frac{1}{2}\Phi_{\tilde{s}_4\tilde{s}_4}(f) * (\delta(f - f_2) + \delta(f + f_2)) \quad (1.45)$$

Bandpaßfilterung:

$$\begin{aligned}
\Phi_{s_3 s_3}(f) &= \Phi_{s_2 s_2}(f) \cdot \left( \text{rect} \left( \frac{f - f_M}{f_\Delta} \right) + \text{rect} \left( \frac{f + f_M}{f_\Delta} \right) \right) \\
&= \frac{1}{4T_A} \left[ \left( \sum_{n=-\infty}^{\infty} \Phi_{vv} \left( f - (f_2 + f_1) - \frac{n}{T_A} \right) + \Phi_{vv} \left( f - (f_2 - f_1) - \frac{n}{T_A} \right) \right) \right. \\
&\quad \cdot \text{rect} \left( \frac{f - f_2}{2f_{\text{TP}}} \right) \\
&\quad + \left( \sum_{n=-\infty}^{\infty} \Phi_{vv} \left( f + (f_2 + f_1) - \frac{n}{T_A} \right) + \Phi_{vv} \left( f + (f_2 - f_1) - \frac{n}{T_A} \right) \right) \\
&\quad \left. \cdot \text{rect} \left( \frac{f + f_2}{2f_{\text{TP}}} \right) \right] \\
&\quad \cdot \left( \text{rect} \left( \frac{f - f_M}{f_\Delta} \right) + \text{rect} \left( \frac{f + f_M}{f_\Delta} \right) \right) \tag{1.46}
\end{aligned}$$

wieder mit  $f_M = f_2 + f_1$ . Der sich anschließende ILO bewirkt wieder eine Substitution der Ausdrücke  $f_2 + f_1$  durch  $f_3$ . Es soll aus oben genannten Gründen wieder nur der rechte Teil des Spektrums berücksichtigt werden. Da zusätzlich  $f_\Delta < 2f_{\text{TP}}$  vorausgesetzt wurde, kann in dem entstehenden Ausdruck der Term

$$\text{rect} \left( \frac{f - f_2}{2f_{\text{TP}}} \right) \tag{1.47}$$

entfallen. Es ergibt sich:

$$\begin{aligned}
\tilde{\Phi}_{s's'} &= \frac{1}{4T_A} \left( \sum_{n=-\infty}^{\infty} \Phi_{vv} \left( f - f_3 - \frac{n}{T_A} \right) + \Phi_{vv} \left( f - (f_3 - 2f_1) - \frac{n}{T_A} \right) \right) \\
&\quad \cdot \text{rect} \left( \frac{f - f_{M'}}{f_\Delta} \right) \tag{1.48}
\end{aligned}$$

mit  $f_{M'} = f_3$

### 1.3.3 Fehler im Spektrum

Es werde wieder eine Fehlerfunktion wie in Abschnitt 1.2.3 entsprechend (1.26) definiert. Es ergibt sich wieder:

$$E \approx 2 \int_0^{\infty} |\tilde{\Phi}_{ss}(f) - \tilde{\Phi}_{s's'}(f)| df. \tag{1.49}$$

Der Fehler  $E_1$  außerhalb des Bandpasses hat sich nicht verändert. Für den Fehler  $E_2$  gilt, nach Amplitudenkorrektur:

$$E_2 = \int_{2f_1 - \frac{f_\Delta}{2}}^{2f_1 + \frac{f_\Delta}{2}} \Phi_{vv}(f) df + \int_{-\frac{f_\Delta}{2}}^{\frac{f_\Delta}{2}} \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} \Phi_{vv}\left(f - \frac{n}{T_A}\right) + \Phi_{vv}\left(f + 2f_1 - \frac{n}{T_A}\right) df \quad (1.50)$$

Ein Vergleich mit dem vorigen Abschnitt zeigt, daß sich der Fehler  $E_2$  summativ aus dem Fehler ohne Abtasten und einem Aliasing-Fehler zusammensetzt.

### 1.3.4 Auswertung

In den Abbildungen 1.14 und 1.15 ist der Fehler innerhalb des Bandpasses  $E_2$  gegen die Filterbandbreite  $f_\Delta$  aufgetragen. Die Kurvenschar besitzt als Parameter die Zahl der Abtastwerte pro Symbol  $N$ . Die einzelnen Diagramme unterscheiden sich in der Wahl der ersten Zwischenfrequenz  $f_1$ .

Das erste Diagramm mit  $f_1 = \frac{1}{T}$  enthält die Abtastraten  $N = 4 \dots 16$  in Zweierschritten. Es hebt sich nur die Kurve für  $N = 4$  deutlich ab. Das zweite Diagramm mit  $f_1 = \frac{1.5}{T}$  zeigt deutlich die Vergrößerung des Fehlers  $E_2$ , wenn die erste Zwischenfrequenz bei zu niedriger Abtastrate angehoben wird. Anschaulich wird die Mittenfrequenz des Bandpasses verschoben und zieht das erste Aliasing-Spektrum mit ein. Es existieren Zwischenfrequenzen  $f_1$ , ab denen das erste Aliasing-Spektrum bereits wieder aus dem Bereich des Bandpasses herauswandert, bis das nächste folgt. Daher beginnt der Fehler  $E_2$  für kleine  $N$  und große  $f_1$  zu oszillieren. Da keine verünftigen Ergebnisse auf diese Weise erzielt werden können, sind in den folgenden Diagramm nur die Abtastraten  $N = 4f_1T \dots 16$  eingezeichnet. Dabei erfüllt jeweils die oberste Kurve die Bedingung (1.40), kann daher besonders leicht implementiert werden.

Bei der Auswahl einer geeigneten Kombination der Parameter  $f_1$ ,  $f_\Delta$  und  $N$  muß berücksichtigt werden, daß der Fehler durch Abschneiden für kleine  $f_\Delta$  besonders groß wird. Da sich jedoch im Empfänger ein Filter mit der Bandbreite  $B \approx \frac{1.5}{T}$  befindet, erscheint ein Wahl von allzu großen Filterbandbreiten im Sender nicht sinnvoll. Unter Berücksichtigung von (1.40) läßt sich Tabelle 1.3.4 aufstellen:

[

### 1.3.5 Vergleich im Zeitbereich

Bisher wurde die Untersuchung der Fehlereinflüsse auf den Frequenzbereich beschränkt. Im Empfänger kommt es jedoch darauf an, die Phase des empfangenen Signals zu gewinnen. Daher wird nun der Einfluß der Abtastung und der

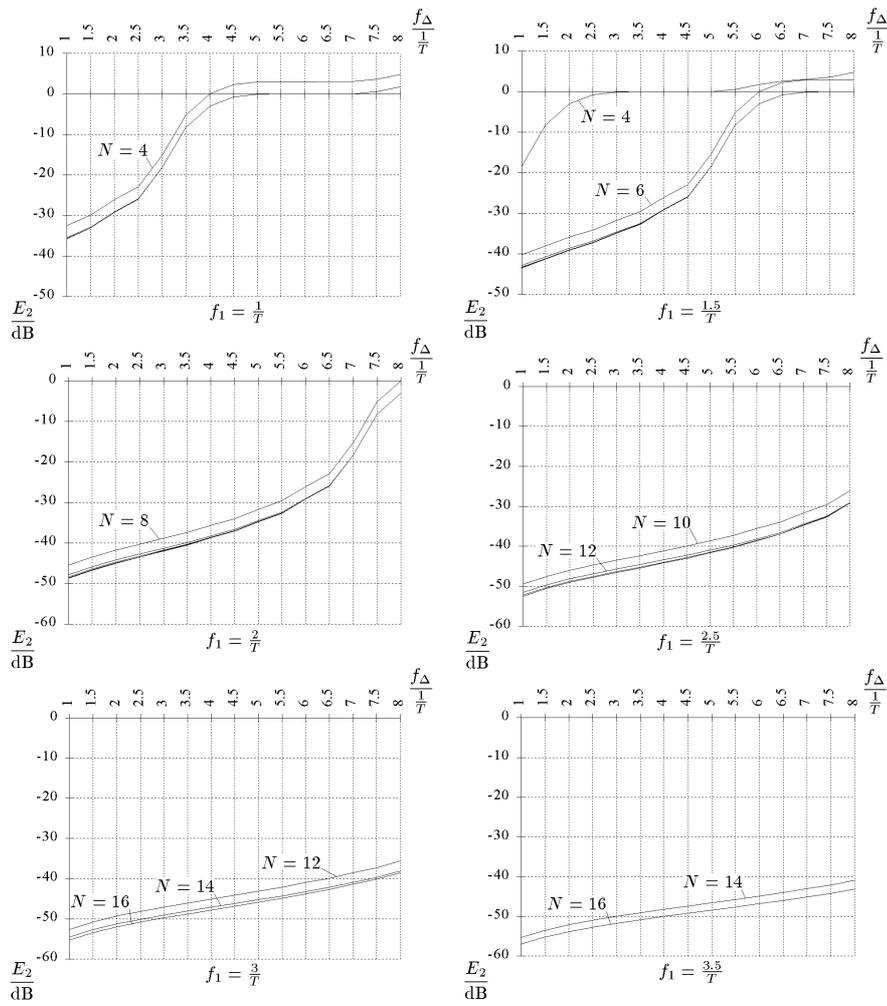
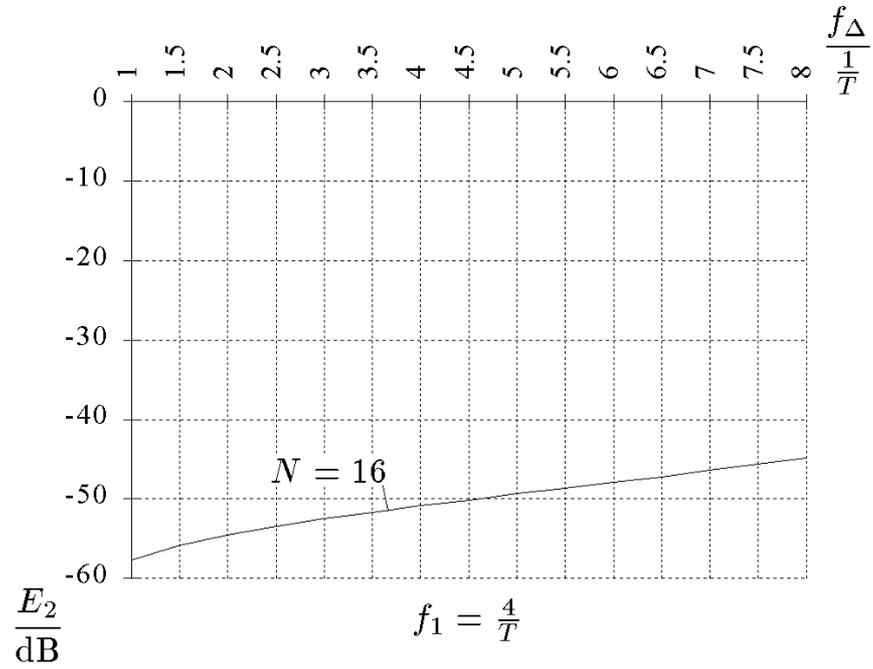
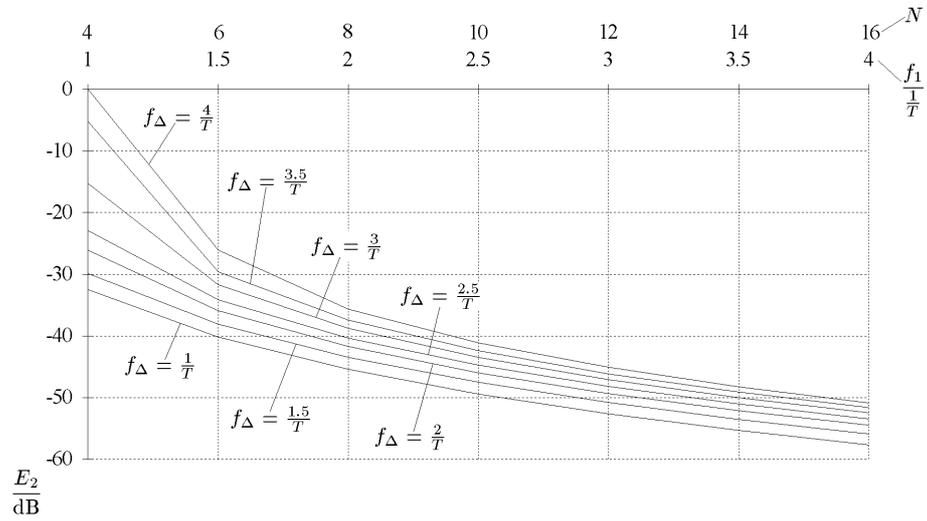


Abbildung 1.14: Fehler  $E_2$  innerhalb des Bandpasses, mit Aliasing

Abbildung 1.15: Fehler  $E_2$ , FortsetzungAbbildung 1.16: Fehler  $E_2$  für gegebene  $f_\Delta$  und  $f_1 = \frac{N}{4T}$

$N$	$f_1$ in $1/T$	$E_2$ in dB		
		$f_\Delta = 1.5/T$	$f_\Delta = 2/T$	$f_\Delta = 2.5/T$
4	1.0	-29.85	-26.12	-22.92
6	1.5	-38.09	-35.94	-34.15
8	2.0	-43.48	-41.75	-40.35
10	2.5	-47.53	-45.99	-44.74
12	3.0	-50.79	-49.34	-48.18
14	3.5	-53.53	-52.13	-51.02
16	4.0	-55.88	-54.52	-53.45

Tabelle 1.2: Fehler  $E_2$  für gegebene  $f_\Delta$  und  $f_1 = \frac{N}{4T}$ 

Tiefpaßfilterung im Zeitbereich untersucht und mit den vorherigen Ergebnissen verglichen.

Die Untersuchung findet im Tiefpaßbereich statt, da der vorherige Abschnitt gezeigt hat, daß unter bestimmten Näherungen Band- und Tiefpaßbereich zum gleichen Ergebnis führen. Nun wird allerdings der Einfluß des Überlappens der Spektren außer Acht gelassen, da nur der Zusammenhang zwischen Fehler im Spektrum und Fehler in der Phase betrachtet werden soll.

Abbildung 1.17 zeigt das komplexe MSK-Signal im Tiefpaßbereich, und Abbildung 1.18 das Argument. Von diesem Signal werden nun beispielsweise  $N = 4$  Abtastwerte pro Symbol genommen und durch einen Tiefpaß der Grenzfrequenz  $f_g = \frac{N}{2T}$  interpoliert. Dies entspricht dem Signal, das der Empfänger ohne Überlappen und Rauschen empfangen würde. Der Empfänger nimmt nun seinerseits Proben aus diesem Signal. Liegen diese Proben genau auf einem Abtastwert des Senders, ist der Fehler Null. Für alle anderen Bereiche entsteht ein Fehler. Dieser Fehler ist besonders groß, wenn die Steigung der Phase ihr Vorzeichen wechselt. Dies geschieht immer dann, wenn das Datum von 0 nach 1 oder umgekehrt wechselt. In den Quadraturkomponenten entstehen Ecken, die durch den Tiefpaß verschliffen werden.

Da Sender und Empfänger nicht exakt taksynchron arbeiten, ergibt sich eine permanente Drift des Abtastzeitpunktes. Daher werden im Empfänger alle Zeitpunkte gleichwahrscheinlich abgetastet. Dies wurde auf dem Rechner simuliert und ergibt die in Tabelle 1.3.5 stehenden Varianzen für den Phasenfehler

$$\varphi_E(t) = \varphi(t) - \varphi_{TP}(t) \quad (1.51)$$

Zum qualitativen Vergleich zwischen Fehler im Spektrum und Fehler in der Phase zeigt Abbildung 1.23 diese Tabelle grafisch. Der Vergleich mit Abbildung 1.16 zeigt eine gute Übereinstimmung.

]

$N$	$\text{var}(\varphi_E(t))$
4	$7.3 \cdot 10^{-4}$
6	$2.4 \cdot 10^{-4}$
8	$1.1 \cdot 10^{-4}$
10	$5.6 \cdot 10^{-5}$
12	$3.5 \cdot 10^{-5}$
14	$2.3 \cdot 10^{-5}$
16	$1.5 \cdot 10^{-5}$
24	$4.7 \cdot 10^{-6}$
32	$2.1 \cdot 10^{-6}$

Tabelle 1.3: Varianz des Phasenfehlers  $\varphi_E$ 

## 1.4 Eigenschaften des Interpolators

In Abbildung 1.1 ist die DA-Wandlung und die Interpolation der Abtastwerte getrennt dargestellt. Bisher wurde als Interpolator ein idealer Tiefpaß angenommen. Diese Annahme soll nun fallengelassen werden. Ein realer DA-Wandler liefert an seinem Ausgang keine Dirac-Stöße sondern eine Treppenkurve. Dabei wird ein Ausgabewert solange gehalten, bis der nächste in das Ausgaberegister geschrieben wird. Mathematisch wird dies durch eine Faltung der Dirac-Stoßfolge mit einer rect-Funktion beschrieben. Dieses Verhalten soll in zwei Stufen modelliert werden: Dem idealen DA-Wandler, der an seinem Ausgang Diracstöße liefert und dem ein rect-Interpolator. Es gelte daher weiterhin  $\tilde{s}_3 = \tilde{s}_1$ , das heißt, es werden noch keine Effekte berücksichtigt, die durch die Wertdiskretheit entstehen.

$$\tilde{s}_4(t) = \tilde{s}_3(t) * \text{rect}\left(\frac{t - \frac{T_A}{2}}{T_A}\right) \quad (1.52)$$

$$\tilde{S}_4(f) = \tilde{S}_3(f) \cdot T_A \text{si}(\pi f T_A) e^{-j2\pi f \frac{T_A}{2}} \quad (1.53)$$

bzw.

$$\Phi_{\tilde{s}_4\tilde{s}_4}(f) = \Phi_{\tilde{s}_3\tilde{s}_3}(f) \cdot T_A^2 \text{si}^2(\pi f T_A) \quad (1.54)$$

Abbildung 1.19 zeigt die beiden Funktionen rechts vom Gleichheitszeichen in (1.53); Abbildung 1.20 das Produkt. Um die Struktur des Spektrums besser erkennen zu können, wurde  $S_1$  und  $S_4$  linear und ohne Quadrierung aufgetragen. Die Nullstellen der si-Funktion liegen an der gleichen Stelle wie die periodischen Wiederholungen des Spektrums. Zwei Dinge sind erkennbar:

1. die  $\text{si}^2$ -Funktion besitzt Tiefpaß-Eigenschaften und
2. das Spektrum  $\Phi_{\tilde{s}_2\tilde{s}_2}(f)$  wird durch die  $\text{si}^2$ -Funktion verzerrt.

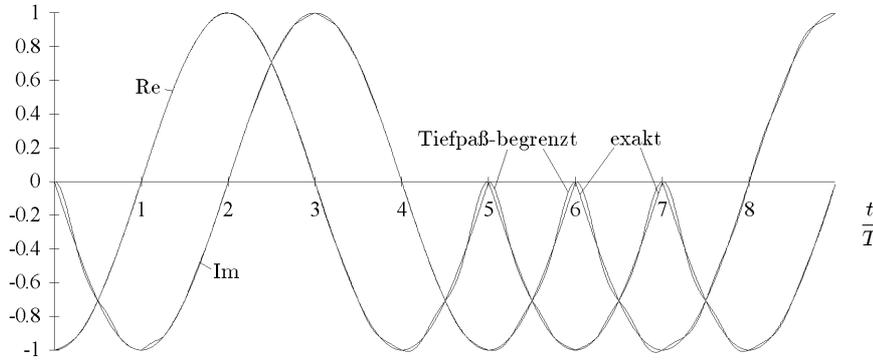


Abbildung 1.17: komplexes MSK-Signal im Tiefpaßbereich, exakt und TP-interpoliert

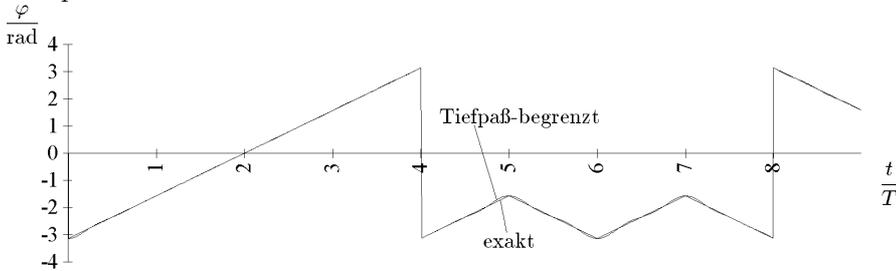


Abbildung 1.18: Argument des MSK-Signals, exakt und TP-interpoliert

### 1.4.1 Tiefpaßcharakter des rect-Interpolators

Der Tiefpaßcharakter der  $\text{si}$ -Funktion ist an die Abtastrate  $\frac{1}{T_A}$  gekoppelt. Für größere Abtastraten wird auch die Dämpfung durch die  $\text{rect}$ -Interpolation zu hohen Frequenzen geringer. Wird im besten Fall mit  $N = 4$  gearbeitet und  $f_2$  wieder zu 70MHz angenommen ergibt sich eine Dämpfung des linken Teils des Spektrums durch Mischung mit  $f_2$  von 40dB. Bei  $N = 8$  sind es noch 34dB und bei  $N = 16$  noch 28dB. Für höhere Abtastraten erscheint deshalb der Einbau eines zusätzlichen analogen Tiefpasses sinnvoll. Wird nun jedoch beachtet, daß mit höheren Abtastraten der Abstand zwischen den Wiederholungen der Spektren zunimmt, läßt sich durch geschickte Wahl von  $f_2$  erreichen, daß der Bandpaß nach der Mischung genau den Teil herausfiltert, der in der Mitte zwischen zwei wiederholten Spektren des linken Spektralteils liegt. Dazu muß folgende Bedingung erfüllt sein:

$$f_2 + f_1 = \frac{2n + 1}{2T_A} - f_2 \quad \text{mit } n \in \mathbb{N} \quad (1.55)$$

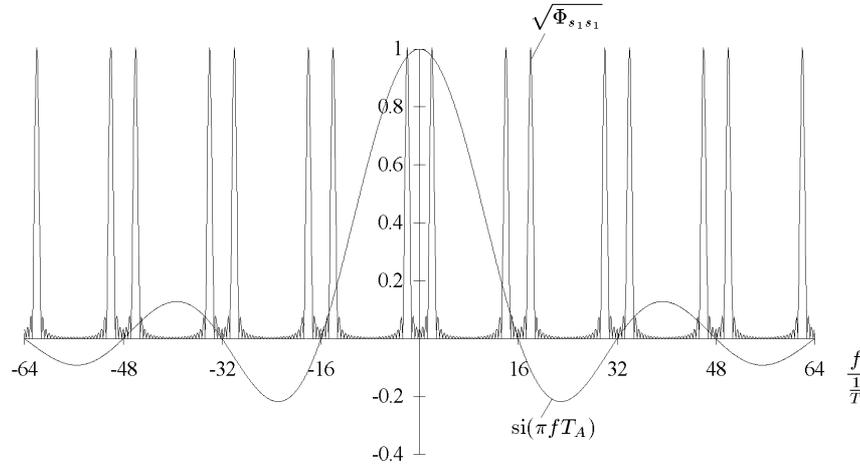


Abbildung 1.19: MSK-Spektrum für  $f_1 = \frac{2}{T}$  nach Abtastung mit  $N = 16$  und si-Funktion

$$\Rightarrow f_2 = \frac{1}{2} \left( \frac{2n+1}{2T_A} - f_1 \right) \quad (1.56)$$

### 1.4.2 si-Verzerrungen

Wie bereits erwähnt, verzerrt die si-Funktion das Spektrum des MSK-Signals. Es soll daher zunächst das Fehlerkriterium des vorigen Kapitels angewandt werden. Es sei noch ein zusätzlicher Tiefpaß nach der rect-Interpolation angenommen. Es ergibt sich nach der Bandpaßfilterung analog zu Abschnitt 1.3.2 nach den entsprechenden Vereinfachungen:

$$\begin{aligned} \tilde{\Phi}_{ss} = & \frac{1}{4T_A} \left[ \sum_{n=-\infty}^{\infty} \left( \Phi_{vv} \left( f - f_3 - \frac{n}{T_A} \right) + \Phi_{vv} \left( f - (f_3 - 2f_1) - \frac{n}{T_A} \right) \right) \right. \\ & \left. \cdot \text{si} \left( \pi \left( f - (f_3 - f_1) - \frac{n}{T_A} \right) T_A \right) \right] \cdot \text{rect} \left( \frac{f - f_{M'}}{f_{\Delta}} \right) \quad (1.57) \end{aligned}$$

mit  $f_{M'} = f_3$

Der Fehler  $E_1$  bleibt wieder unverändert. Der Fehler  $E_2$  ist analog wie in Abschnitt 1.3.2 definiert, werde hier jedoch nicht mehr explizit ausgeschrieben. Abbildung 1.21 zeigt den Fehler für  $N=16$ .

Es ist eine deutliche Verschlechterung gegenüber der Interpolation mittels idealem Tiefpaß in Abbildung 1.16 zu erkennen. Dies liegt daran, daß das Signal im Bandpaßbereich auf  $f_1$  abgetastet wird, einem Bereich, in dem sich der gekrümmte Verlauf der si-Funktion bereits stark bemerkbar macht.

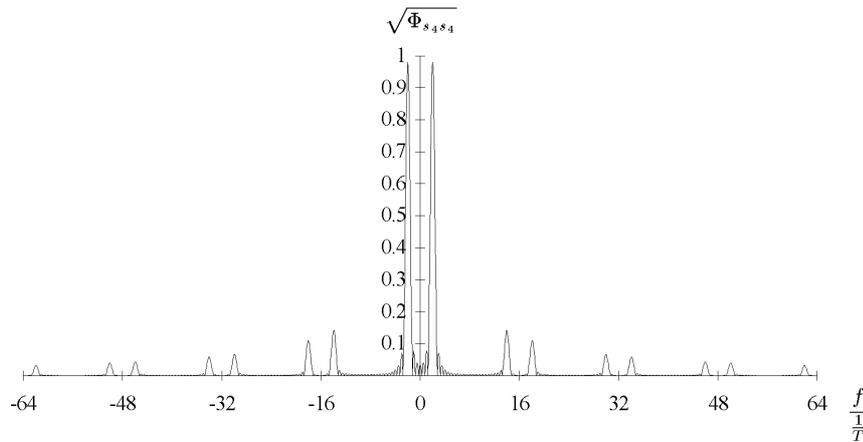


Abbildung 1.20: MSK-Spektrum für  $f_1 = \frac{2}{T}$  nach Abtastung mit  $N = 16$  multipliziert mit si-Funktion

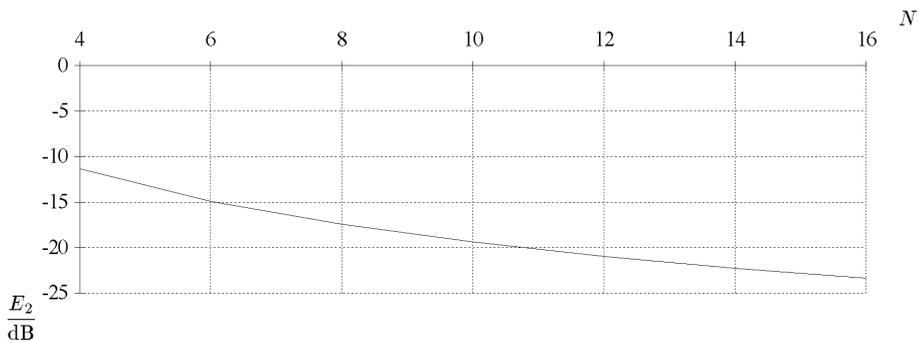


Abbildung 1.21: Fehler  $E_2$  für  $f_\Delta = \frac{1}{T}$  und  $f_1 = \frac{1}{T}$ , mit si-Verzerrungen

### 1.4.3 1/si-Filter

Nun wird die Wirksamkeit einer Vorverzerrung untersucht. Diese soll mit Hilfe eines FIR-Filters im digitalen Teil des Senders erfolgen. Aufgrund der Zeitdiskretheit ergibt sich ein Filterspektrum, das sich mit der Periode  $\frac{1}{T_A}$  wiederholt.

$$H_{\text{si}}(f) = \left( \frac{1}{\text{si}(\pi f T_A)} \cdot \text{rect}(f T_A) \right) * \sum_{n=-\infty}^{+\infty} \delta \left( f - \frac{n}{T_A} \right) \quad (1.58)$$

Die Transformation von (1.58) in den Zeitbereich liefert die Stoßantwort des FIR-Filters. Tabelle 1.4.3 zeigt die Koeffizienten.

An dieser Stelle muß nun zum ersten Mal die Wertdiskretheit der digitalen Verarbeitung berücksichtigt werden. Da die Filterkoeffizienten nicht beliebig genau gewählt werden können, ergibt sich ein etwas anderer Frequenzgang des

$k$	$h_{\text{si}}(k)$	$\frac{1}{h_{\text{si}}(k)}$
$\vdots$	$\vdots$	$\vdots$
-5	-0.005	-188
-4	0.008	122
-3	-0.014	-70.8
-2	0.030	33.8
-1	-0.092	-10.9
0	1	1
-1	-0.092	-10.9
-2	0.030	33.8
-3	-0.014	-70.8
-4	0.008	122
-5	-0.005	-188
$\vdots$	$\vdots$	$\vdots$

Tabelle 1.4: ideale FIR-Filterkoeffizienten

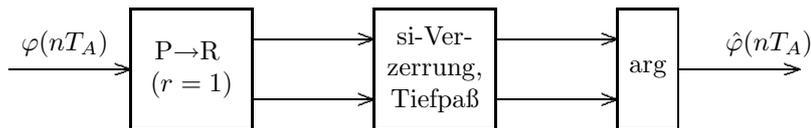


Abbildung 1.22: MSK-Signalweg, vereinfacht

Entzerrfilter. Besonders einfach sind Implementierungen, bei denen sich die Filterkoeffizienten als Binärbrüche darstellen lassen. Aus diesem Grund enthält die Tabelle 1.4.3 in der dritten Spalte die Kehrwerte der Koeffizienten, um eine leichtere Abschätzung des Abstands zur nächsten Zweierpotenz zu ermöglichen. Eine zusätzliche Veränderung des idealen  $\frac{1}{\text{si}(\pi f T_A)}$ -Frequenzgangs ergibt sich durch das Abschneiden der Stoßantwort.

Es zeigt sich, daß selbst bei einer sehr genauen Wahl der Koeffizienten die Werte für  $E_2$  ca. -30dB nicht unterschreiten. Der Gewinn beträgt im Vergleich zu dem Sender ohne  $\frac{1}{\text{si}(\pi f T_A)}$ -Filter maximal ca. 10dB. Bei zusätzlicher Rundung der Koeffizienten fällt der Gewinn noch geringer aus. Dies liegt daran, daß das gewählte Fehlerkriterium (1.26) willkürlich gewählt wurde und auf Ungenauigkeiten im Spektrum sehr empfindlich reagiert.

#### 1.4.4 Auswirkungen der si-Verzerrungen

Zur Vereinfachung werde die Zwischenfrequenzebene  $f_2$  sowie die Trägerfrequenz außer Acht gelassen. Es sei ein Transceiver angenommen, bei dem der Sender ein MSK-Signal der Mittenfrequenz  $f_1$  erzeugt, das im Empfänger wieder zurück in den Tiefpaßbereich geschoben wird.

Da der Empfänger das Eingangssignal mit  $e^{-j2\pi f_1 t}$  multipliziert, wird das Spek-

trum um  $f_1$  nach links verschoben. Dadurch gelangt das MSK-Spektrum wieder in den Tiefpaßbereich. Es erscheint jedoch auch noch einmal bei  $2f_1$ . Diese Wiederholung wird im Empfänger durch ein spezielles Filter weitgehend beseitigt, so daß sie hier zum Verständnis der Wirkung der si-Verzerrungen vorerst keine Berücksichtigung mehr finden soll. Übrig bleibt das durch die si-Funktion verzerrte MSK-Signal im Tiefpaßbereich, für dessen Spektrum gilt:

$$\Phi_{ww}(f) = \Phi_{vv}(f) \cdot \overset{2}{\text{si}}(\pi(f + f_1)T_A) \cdot \text{rect}\left(\frac{f}{f_\Delta}\right) \quad (1.59)$$

Abbildung 1.22 zeigt dieses System.

Die Modulation und Demodulation kann daher durch die gemachten Vereinfachungen als ein LTI-System aufgefaßt werden, dessen Übertragungsfunktion durch

$$H(f) = \text{si}(\pi(f + f_1)T_A) \cdot \text{rect}\left(\frac{f}{f_\Delta}\right) \quad (1.60)$$

beschrieben wird. Diese Übertragungsfunktion erfüllt nicht die Bedingung  $H(f) = H^*(-f)$ . Die Stoßantwort ist daher komplex. Durch Aufspalten von (1.60) in einen geraden und einen ungeraden Anteil läßt sich die Stoßantwort des Systems nach Real- und Imaginärteil gewinnen. Dazu werde die si-Funktion in eine Taylorreihe  $\tilde{H}(f)$  um  $f = 0$  entwickelt.

$$\tilde{H}(f) = \text{rect}\left(\frac{f}{f_\Delta}\right) \cdot (a_0 + a_1f + a_2f^2 + \dots) \quad (1.61)$$

mit

$$\begin{aligned} a_0 &= \text{si}(\pi f_1 T_A) \\ a_1 &= \frac{\pi T_A f_1 \cos(\pi f_1 T_A) - \sin(\pi f_1 T_A)}{\pi T_A f_1^2} \\ a_2 &= \dots \end{aligned}$$

Der Faktor  $f_1 T_A$  ist unter Berücksichtigung von (1.40) konstant 4. Damit ergibt sich:  $a_0 \approx 0.9$ ,  $a_1 \approx -0.19$  und  $a_2 \approx 0.08$ . Die lineare Näherung ist in dem hier betrachteten Bereich ausreichend genau. Damit ergibt sich sofort die Aufteilung in geraden und ungeraden Anteil. Der gerade Anteil, der später als Realteil der Stoßantwort erscheint, ist ein idealer Tiefpaß mit einer von der Abtastrate und der Zwischenfrequenz abhängigen Durchlaßdämpfung. Der ungerade Anteil hat eine Übertragungsfunktion, die durch den Ursprung geht und deshalb Hochpaßcharakter hat. Im Zeitbereich bedeutet dies, daß dem ideal tiefpaßgefilterten Signal ein um  $\frac{\pi}{2}$  gedrehter hochpaßähnlich gefilterter Anteil überlagert wird. Da sich der Hauptteil des MSK-Spektrums um  $f = 0$  befindet, macht sich der Fehler, der durch die Aufaddition des Imaginärteils entsteht, nur wenig bemerkbar.

[ Abbildung 1.23 zeigt die Varianz des Phasenfehlers bei Abtastung und Tiefpaßinterpolation mit und ohne si-Verzerrungen. Dabei wurde für die Kurve mit si-Verzerrungen  $f_1 = \frac{N}{4T}$  angenommen.

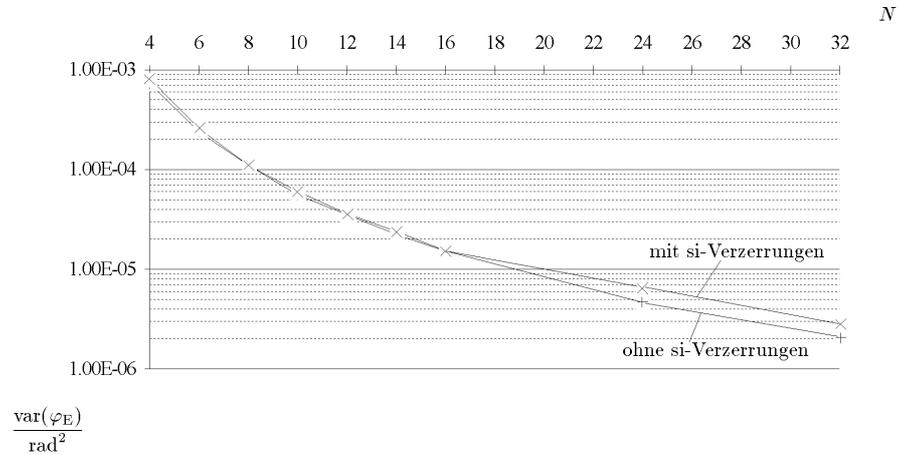


Abbildung 1.23: Varianz des Phasenfehlers

Die Varianz des Phasenfehlers hängt fast ausschließlich von der gewählten Abtastrate ab. Erst für große  $N$  und damit auch kleine Varianzen macht sich der Einfluß der si-Verzerrung bemerkbar. Daher ist das  $\frac{1}{\text{si}}$ -Filter von Seiten des Empfängers entbehrlich. ]

#### 1.4.5 Varianz des Phasenfehlers

Die Varianz des Phasenfehlers wird geprägt von einzelnen Ausreißern. Diese entstehen an den Stellen der Änderung der Phasensteigung. Abbildung 1.24

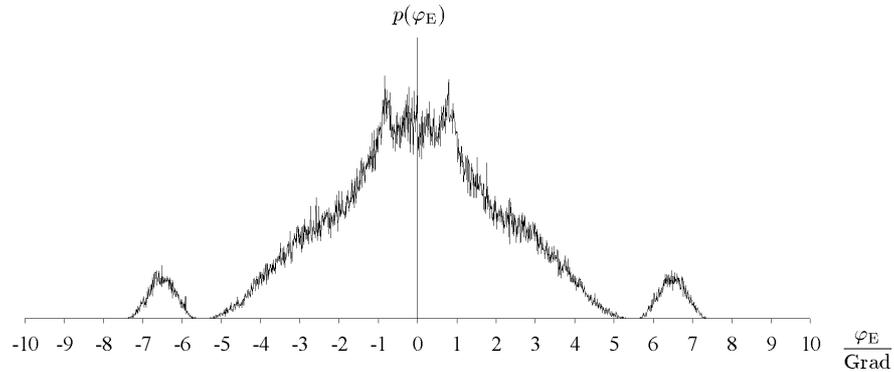


Abbildung 1.24: Verteilungsdichtefunktion des Phasenfehlers

zeigt die Verteilungsdichtefunktion des Phasenfehlers für den Fall  $N = 16$  und  $f_\Delta = \frac{1,5}{T}$ . Die Ausreißer sind deutlich im Bereich  $\pm 6^\circ \dots \pm 7^\circ$  zu erkennen. Eine

Beschreibung dieser Verteilung durch einfache statistische Basisverteilungen ist daher nicht möglich.

## 1.5 Bitfehlerraten ohne Quantisierung

Eine ausreichende Beschreibung des Fehlers über die Varianz ist wegen der speziellen Struktur der Verteilungsdichtefunktion nicht möglich. Da das entscheidende Optimalitätskriterium für die Entwicklung des MSK-Transceiver die Bitfehlerrate ist, wird im folgenden diese als Bewertungsgrundlage herangezogen. Dazu wird das System in Abbildung 1.25 als Ausgangspunkt dienen. Als Fehlerquellen sind enthalten:

1. Aliasing-Effekte durch zu niedrige Abtastrate
2. Bandbegrenzung

Das System enthält den MSK-Modulator dessen Signal im Tiefpaßbereich mit weißem Rauschen gestört wird. Im Empfänger wird dieses Signal tiefpaßgefiltert. Das Filter ist ein Rolloff-Tiefpaß dessen Übertragungsfunktion durch

$$H(f) = \sqrt{\frac{1}{2} \cdot \left( 1 - \sin \frac{\pi \cdot (f - f_{3dB})}{2\beta f_{3dB}} \right)} \quad (1.62)$$

gegeben ist.  $\beta$  ist der Rolloff-Faktor und ist hier zu 0.1 gewählt. An die Argumentbestimmung schließt sich die Phasendifferenzbildung mit nachfolgender Dezimation auf die Symbolrate an. Die folgende Schwellenwertentscheidung liefert die Daten wieder zurück.

Abbildung 1.26 enthält zusätzlich folgende Fehlerquellen:

1. Überlappen der Spektren durch das Mischen
2. si-Verzerrungen durch den rect-Interpolator
3. Beseitigung der Misch-Spiegelfrequenzen sowie des DC-Offsets durch  $H(z) = z^4 + 2z^3 + 2z^2 + 2z + 1$  (zur näheren Erläuterung von  $H(z)$  werde hier auf das nächste Kapitel verwiesen) statt durch idealen Tiefpaß

In diesem System wird das MSK-Signal auf  $f_1$  gemischt, bandpaßgefiltert und anschließend auf die Leistung  $P = 1$  normiert. Die Normierung ist aufgrund der Bandpaßfilterung notwendig. Zusätzlich kann der Bandpaßübertragungsfunktion an dieser Stelle die si-Verzerrung überlagert werden. Auf diese Weise läßt sich der Einfluß der si-Verzerrungen auf die Bitfehlerrate untersuchen. Nach der Normierung wird das Rauschen im Bandpaßbereich addiert. Der Empfänger mischt das Signal wieder in den Tiefpaßbereich. Die Mischung erfordert Tiefpaßfilter um die Spiegelfrequenzen zu beseitigen. Dieses Filter ist zum Ersten wieder als Rolloff-Tiefpaß ausgeführt. Zum Zweiten ist ein spezielles FIR-Filter mit der Übertragungsfunktion  $H(z)$  an dieser Stelle eingesetzt worden. Es läßt

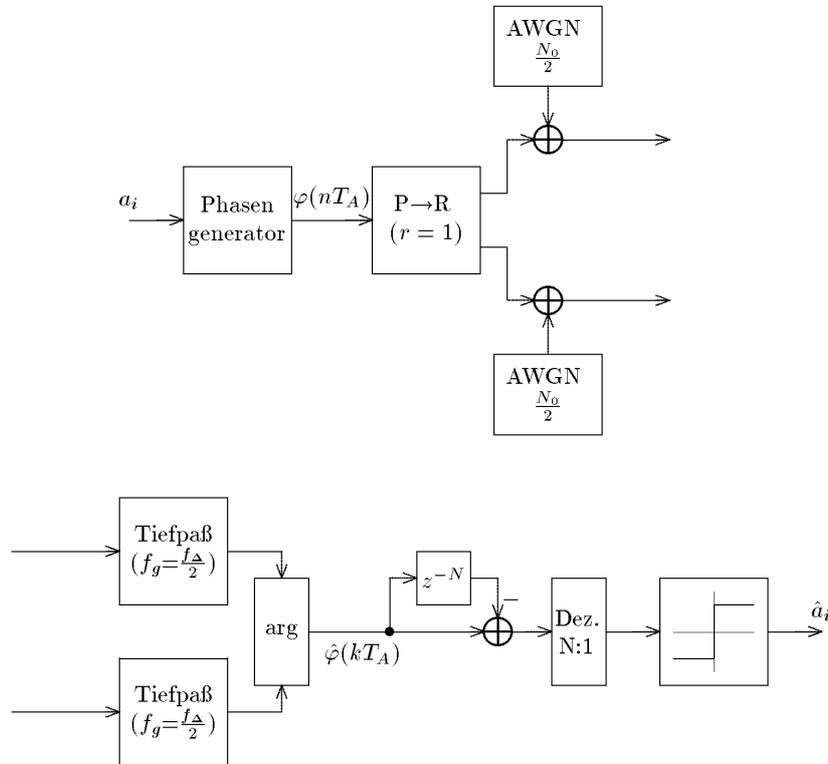


Abbildung 1.25: MSK-Transceiver-System ohne Mischung

sich günstiger als ein idealer Tiefpaß implementieren und beseitigt gleichzeitig einen vom DA-Wandler stammenden eventuellen DC-Offset.

Diese beiden Systeme wurden auf dem Rechner simuliert. Dabei wurden als weitere Parameter  $f_{\Delta} = \frac{1,5}{T}$ ,  $N = 4$  gewählt. Sender und Empfänger laufen ideal synchron, der Frequenzoffset ist zu Null eingestellt. Die letzten beiden Vereinfachungen werden gemacht, da an dieser Stelle nur die Modellparameter des Senders bestimmt werden sollen, und daher noch kein Augenmerk auf spezielle Empfangsprobleme gerichtet wird. Es wird daher kein Schätzer zur Takt- und Frequenzoffsetkorrektur benötigt.

[ ]

Abbildung 1.28 zeigt die Simulationsergebnisse für ein wertkontinuierlich bei idealer Synchronisation arbeitendes Transceiver-System. Es wurden Simulationen mit und ohne si-Verzerrungen durchgeführt. Bei diesen war jedoch im Rahmen der Simulationengenauigkeit kein Unterschied zu erkennen. Abbildung 1.28 zeigt, daß für geringe SNRs der Empfänger mit  $H(z)$  günstiger arbeitet, da er mehr Rauschen herausfiltert. Bei guten SNRs ist es günstiger mit einem steilflankigen Tiefpaß zu arbeiten.

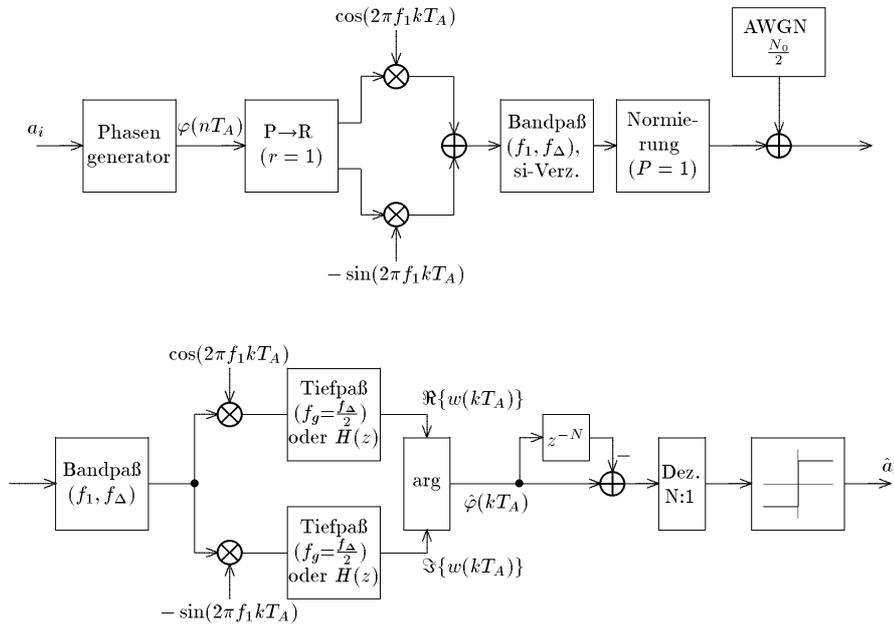


Abbildung 1.26: MSK-Transceiver-System mit Mischung

## 1.6 digitale Erzeugung (wertdiskret)

Eine weitere Fehlerquelle des nichtidealen MSK-Modulators von Abbildung 1.1 ist die wertdiskrete Verarbeitung.

### 1.6.1 effektive (DA-) Auflösung

In diesem Abschnitt wird schrittweise die notwendige Auflösung der DA-Wandlung und der wertdiskreten Verarbeitung bestimmt. Dabei wird vom DA-Wandler ausgehend rückwärts vorgegangen, da beim DA-Wandler nur eine Auflösung Sinn macht, die die vorhergehende Verarbeitung liefert. Der Empfänger arbeitet vorerst noch ohne Quantisierung.

Die Bitfehlerraten zeigen in Simulationen eine bemerkenswerte Abhängigkeit von der Quantisierungsauflösung. Für bereits in Hinblick auf diesen Effekt optimierte Kennlinien zeigt Abbildung 1.32, daß sich die Fehlerraten keineswegs mit steigender Auflösung monoton fallend den Fehlerraten ohne Quantisierung nähern. Die Fehlerraten werden im Vergleich zu denen ohne Quantisierung sogar für bestimmte Auflösungen unterschritten. Dieser Effekt resultiert aus der Übertragung des MSK-Signals über einen bandbegrenzten Kanal und wird in den folgenden Abschnitten näher untersucht.

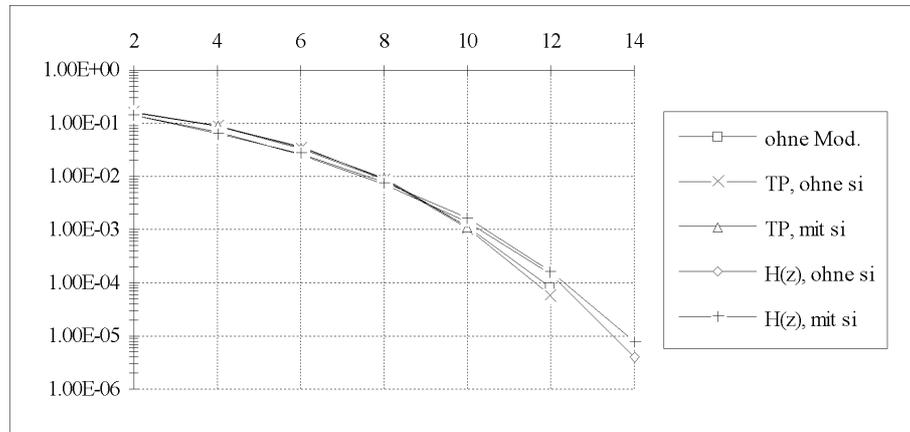


Abbildung 1.27: Bitfehlerraten für verschiedene Übertragungssysteme

### 1.6.1.1 Auswirkung der Quantisierung auf die Phase

Das komplexe MSK-Signal werde zunächst im Tiefpaßbereich quantisiert. Später wird gezeigt, daß dies für den Fall  $f_1 = \frac{N}{4T}$  direkt auf die Quantisierung im BP-Bereich übertragbar ist. Das komplexe Tiefpaß-MSK-Signal ist ein sich auf dem Einheitskreis drehender Zeiger, der für das Datenbit 0 bzw. 1 rechts- bzw. links-herum läuft. Wird die komplexe Ebene quantisiert, kann der Zeiger nur auf den Kreuzungen der Quantisierungsgitternetzlinien verlaufen. Mit der optimierten Quantisierungskennlinie

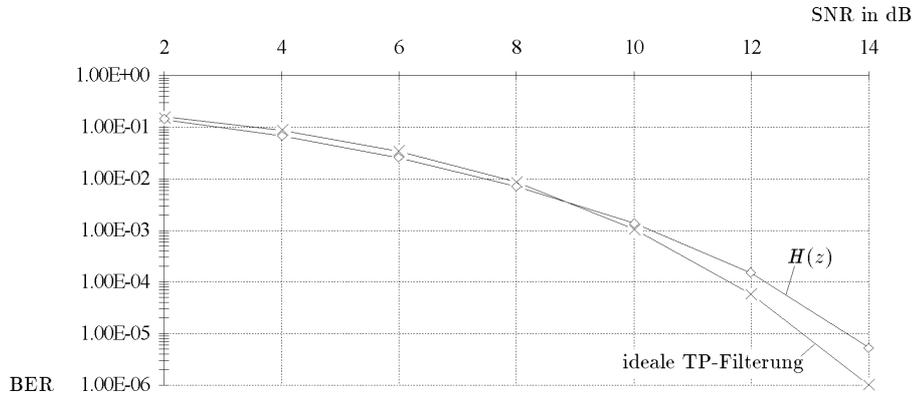
$$f_{DA}(x) = \text{integer}(c * x) / c \quad (1.63)$$

$$c = 2^{n-1} - \frac{1}{2}, n: \text{Zahl der Bitstellen} \quad (1.64)$$

$$\text{integer}(x) = \text{nächste betragsmäßig kleinere ganze Zahl} \quad (1.65)$$

zeigt Abbildung 1.29 in den Reihen von oben nach unten:

1. Die Quantisierungskennlinien, die folgende Merkmale aufweisen:
  - (a) Sie sind symmetrisch zum Ursprung.
  - (b) Der Nullpunkt ist enthalten.
  - (c) Der letzte Stufenwert wird nicht erst bei 1 sondern schon vorher erreicht. Dadurch wird der Einzugsbereich unterhalb von 1 vergrößert, ähnlich wie dies um die 0 durch Verdoppelung der Stufenbreite geschieht. Der Bereich, der zu 1 bzw.  $-1$  fällt, beträgt dabei nur ein Viertel des Bereichs für die 0. Dies ist deswegen sinnvoll, da die Steigung der trigonometrischen Funktionen um ihre Extremwerte gering ist.

Abbildung 1.28: Bitfehlerraten für ideale Tiefpaß-Filterung und  $H(z)$ 

Die Punkte (a) und (b) bedingen eine ungerade Anzahl von Stufen, so daß bei Ausgabe der quantisierten Werte mit realen DA-Wandlern ein Wert nicht genutzt werden kann. Ist  $n$  die Zahl der Bitstellen, so berechnet sich die Zahl der Stufen  $W$  zu

$$W = 2^n - 1 \quad (1.66)$$

- Die Ortskurve des komplexen MSK-Zeigers, die von einem Kreis abweicht. Für  $W = 7$  werden die Punkte  $1, j, -1, -j$  mehrfach getroffen, da der Einzugsbereich für diese Punkte mit größerer Quantisierung wächst. Dies ist nach der Bandbegrenzung für eine spätere Detektion von entscheidender Bedeutung.
- Die Phase läuft nicht mehr linear hoch, sondern weist eine nicht äquidistante Treppenstruktur auf. Die Wahl der Quantisierungskennlinie beeinflußt daher in hohem Maße den Verlauf der Phase.

### 1.6.1.2 Tiefpaßfilterung der verzerrten Phase

Abbildung 1.30 ist eine Fortsetzung von Abbildung 1.29 nach einer Begrenzung der Bandbreite auf  $f_{\Delta} = \frac{1.5}{T}$ . Die vier Reihen enthalten von oben nach unten:

- Die Ortskurven des komplexen MSK-Zeigers. Es ist deutlich zu erkennen, wie sich mit abnehmender Quantisierungsauflösung die Verformung des Einheitskreises zu einer Raute vollzieht. Schon im quasikontinuierlichen Fall mit 255 Stufen Auflösung weicht die Ortskurve vom Einheitskreis ab. Die Abweichung vom Einheitskreis zieht Amplitudenschwankungen im späteren Bandpaßsignal nach sich, die durch den ILO wieder ausgeglichen werden und spielen daher keine Rolle. Entscheidend ist der Effekt, daß der Abstand der Punkte für geringere Quantisierungsaufösungen nicht mehr

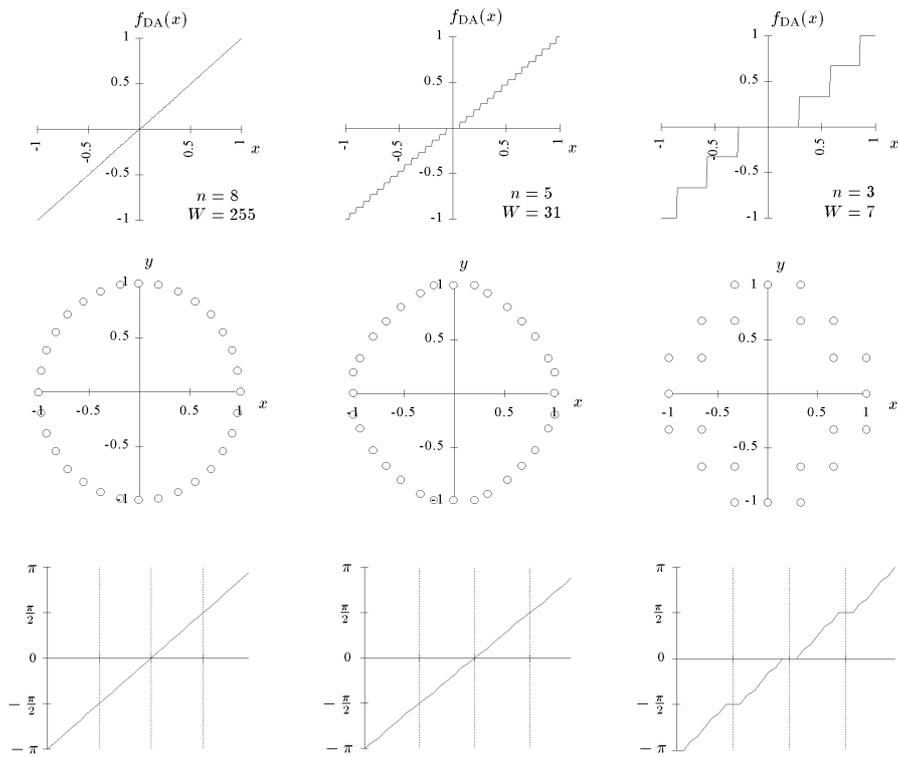


Abbildung 1.29: Quantisierungskennlinie, komplexe Zahlenebene und Phase für verschiedene Quantisierungen

äquidistant ist. Im Bereich der Winkel  $0, \frac{\pi}{2}, \pi$  und  $\frac{3\pi}{2}$  erfolgt eine Verdichtung der Abtastwerte. Dies ist darauf zurückzuführen, daß Punkte um diese Winkel in der quantisierten komplexen Zahlenebene auf einen Punkt  $(1, j, -1, j)$  abgebildet werden. Dies resultiert nach der Tiefpaßfilterung in einer Häufung der Abtastwerte um diesen Bereich.

2. Die sich aus den Ortskurven ergebenden Phasenverläufe. Die unter Punkt 1 beschriebene Häufung der Abtastwert drückt sich in der Phase durch eine Welligkeit aus. Der Vergleich mit Abbildung 1.29 zeigt, daß die dort vorhandenen Treppenstufen hier verschliffen werden. Dies geschieht jedoch so, daß die Differenzphase im Falle einer Phasenumkehr den Phasenendwert  $\pm \frac{\pi}{2}$  besser erreicht.
3. Die Differenzphase zwischen zwei benachbarten Symbolen fällt für den Fall der Phasenumkehr bei geringerer Quantisierung größer aus und läßt sich somit sicherer detektieren. Nachteilig weist die Phasendifferenz eine größere Welligkeit für den Fall ohne Phasenumkehr auf. Richtige Synchronisation vorausgesetzt ist diese Differenzphase mindestens genauso groß wie ohne Quantisierung.
4. Das Spektrum der tiefpaßbegrenzten Signale zeigt für gröbere Quantisierung eine Anhebung der Leistungsdichte bei Entfernung von der Mittenfrequenz  $f = 0$ . Das Signal nutzt daher die vom Kanal bereitgestellt Übertragungsbandbreite besser aus.

Eine Tiefpaßfilterung der komplex quantisierten MSK-Zeiger ergibt Phasenverläufe, die im Falle der Phasenumkehr eine größere Differenzphase aufweisen. Dies wird durch eine leichte Welligkeit der Differenzphase bei gleichen aufeinanderfolgenden Datenbits erkauft. Die Minima des Betrages der Differenzphase liegen dabei, eine geeignete Wahl der Quantisierungskennlinie vorausgesetzt, noch oberhalb der Werte ohne Quantisierung.

Die MSK-Phase wird daher durch die Quantisierung so verzerrt, daß sie bei differentieller Dekodierung nach der Tiefpaßfilterung besser zu detektieren ist. Welche Auswirkungen dieser von der normalen MSK-Phase abweichende Phasenverlauf für den Schätzalgorithmus im Empfänger hat, wird in Abschnitt 1.7 näher untersucht.

### 1.6.1.3 Quantisierung im Bandpaßbereich

$f_{DA}(x)$  repräsentiere die Kennlinie der Quantisierung,  $x(t)$  und  $y(t)$  die Quadraturkomponenten im Tiefpaßbereich und  $s(t)$  das MSK-Signal im Bandpaßbereich auf  $f_1$ . Dann gilt bei Quantisierung im Tiefpaßbereich:

$$s(kT_A) = f_{DA}(x(kT_A)) \cdot \cos(2\pi f_1 kT_A) - f_{DA}(y(kT_A)) \cdot \sin(\pi f_1 kT_A) \quad (1.67)$$

Ist die Bedingung  $f_1 = \frac{N}{4T}$  erfüllt, gilt:

$$\cos(2\pi f_1 kT_A) \neq 0 \iff \sin(2\pi f_1 kT_A) = 0, \text{ und}$$

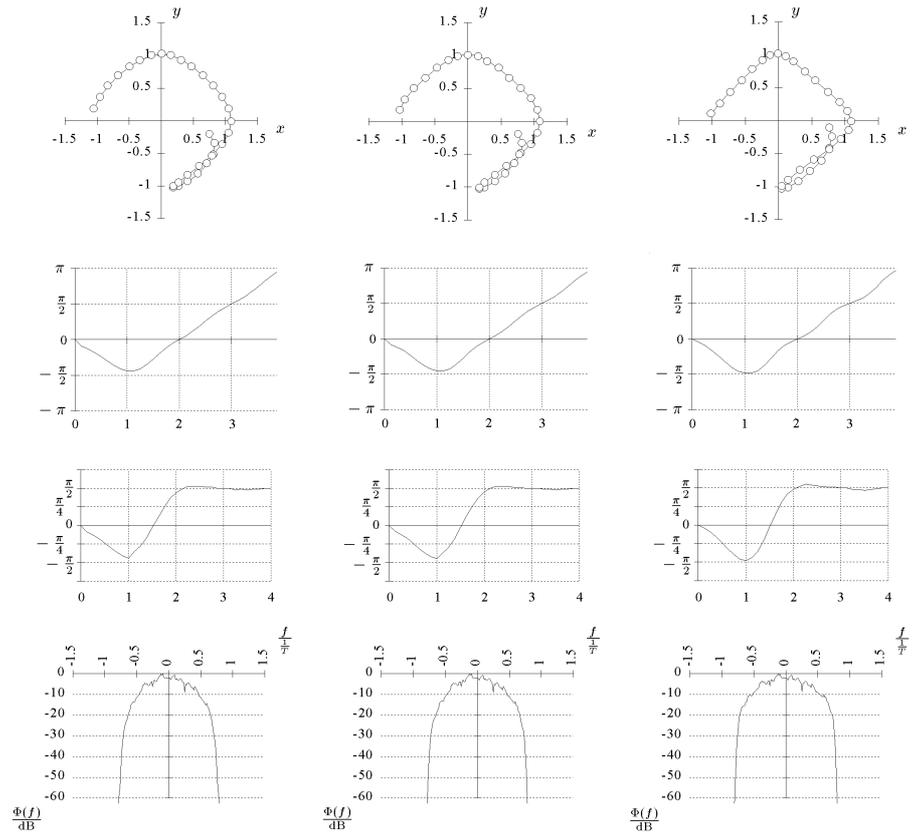


Abbildung 1.30: komplexe Zahlenebene, Phase, Differenzphase und Spektrum nach Bandbegrenzung für verschiedene Quantisierungen

$$\cos(2\pi f_1 k T_A) = 0 \iff \sin(2\pi f_1 k T_A) \neq 0 \quad (1.68)$$

sowie

$$\begin{aligned} \cos(2\pi f_1 k T_A) &\in \{-1, 0, 1\}, \text{ und} \\ \sin(2\pi f_1 k T_A) &\in \{-1, 0, 1\} \end{aligned} \quad (1.69)$$

Falls die Quantisierung zusätzlich symmetrisch zum Ursprung ist:

$$f_{\text{DA}}(x) = -f_{\text{DA}}(-x) \quad (1.70)$$

gilt:

$$s(kT_A) = f_{\text{DA}}\left(x(kT_A) \cdot \cos(2\pi f_1 k T_A) - y(kT_A) \cdot \sin(\pi f_1 k T_A)\right) \quad (1.71)$$

(1.71) entspricht einer Quantisierung im Bandpaßbereich, so daß die oben gemachten Überlegungen hierhin übertragbar sind. Die für diese Äquivalenz nötige Voraussetzung, daß die Quantisierungskennlinie symmetrisch zum Ursprung ist, stellt keine entscheidende Einschränkung dar, da die Kennlinie schon wegen der günstigeren Phasenverläufe den Nullpunkt symmetrisch umschließen sollte. Dies läßt sich bei der Implementierung mit in die Tabelle zur P→R-Wandlung integrieren. Siehe 1.6.4.

#### 1.6.1.4 Verwandtschaft mit Spezialform von OQPSK

Der hier durch die Quantisierung und Tiefpaßfilterung erreichte Phasenverlauf hat Ähnlichkeit mit einer besonders bandbreiteneffizienten Form des OQPSK (Offset Quadrature Phase Shift Keying). MSK selbst zählt zu OQPSK, besitzt jedoch einen rechteckigen Pulse-Shape (vgl.  $\frac{d}{dt}$  von (1.4)), was zu einem relativ weit ausgedehnten Spektrum führt. Die bandbreiteneffiziente Form des OQPSK besitzt stattdessen einen Raised Cosine Impulse, der zu wesentlich weicheren Flanken führt [9]. Genau diesen Effekt erzeugt auch die Quantisierung und anschließende Tiefpaßfilterung. Eine direkte Implementierung eines anderen Pulse-Shapes wäre relativ aufwendig, da der Phasengenerator aufwendiger würde. Zusätzlich müßte die Auflösung der sin- und cos-Tabellen in der polar→rechtwinklig-Wandlung stark erhöht werden. Durch die geschickte Quantisierung bekommt man diesen Aufwand geschenkt.

#### 1.6.2 Phasengenerator

An erster Stelle in Abbildung 1.12 ist der Phasengenerator zu erkennen. Da sich (1.4) nicht während eines Symbols ändert, läßt sich die Summation in (1.2) rekursiv schreiben.

$$\begin{aligned} \Delta\varphi(t) &= \varphi(t + \Delta t) - \varphi(t) \\ &= 2\pi h \sum_i b_i (q(t + \Delta t - iT) - q(t - iT)) \end{aligned} \quad (1.72)$$

$$\varphi(t + \Delta t) = \varphi(t) + 2\pi h \sum_i b_i (q(t + \Delta t - iT) - q(t - iT)) \quad (1.73)$$

Wird  $t = kT_A$  und  $\Delta t = T_A$  gesetzt, dann folgt:

$$\varphi((k+1) \cdot T_A) = \varphi(kT_A) + 2\pi h \sum_i b_i \underbrace{(q((k+1) \cdot T_A - iT) - q(kT_A - iT))}_M \quad (1.74)$$

mit (1.4) folgt:

$$M = \begin{cases} \frac{1}{2N} & \text{für } 0 \leq \frac{k}{N} - i < 1 \\ 0 & \text{sonst} \end{cases} \quad (1.75)$$

Es ergibt sich:

$$\varphi((k+1) \cdot T_A) = \varphi(kT_A) + 2\pi h b_i \frac{1}{2N} \quad (1.76)$$

Damit folgt wie gefordert (1.38). Für die Rekursionsbasis gelte  $\varphi(0) = 0$ .

Es sind  $n = \frac{2N}{h} = 4N$  Takte nötig, damit  $\varphi(kT_A)$  von 0 bis  $2\pi$  läuft.

Umnormieren von (1.76) durch Multiplikation mit  $\frac{N}{\pi h}$  liefert:

$$\hat{\varphi}((k+1) \cdot T_A) = \hat{\varphi}(kT_A) + b_i \quad (1.77)$$

(1.77) läßt sich sehr leicht implementieren, da der Faktor  $\frac{N}{\pi h}$  in der Tabelle berücksichtigt werden kann. Siehe Abbildung 1.31.

### 1.6.3 Sinustabelle

Während die Generierung von  $\tilde{\varphi}(kT_A) = \frac{\pi h}{N} \cdot \hat{\varphi}(kT_A)$  noch exakt erfolgen kann, ist die Bereitstellung der trigonometrischen Funktionen nicht mehr ohne Fehler möglich.

#### 1.6.3.1 polar $\rightarrow$ rechtwinklig

Der Phasengenerator liefert den Winkel für die Umwandlung polar  $\rightarrow$  rechtwinklig mit einer Auflösung von  $4N$  Stufen pro Periode. Damit steht die Abszissenauflösung für den Sinus bereits fest. Es bleibt die Wortlänge für die kartesischen Koordinaten zu bestimmen.

#### 1.6.3.2 Mischoszillator

Wird die Bedingung (1.40) erfüllt, läßt sich die Sinustabelle für den Mischoszillator mit anschließender Multiplikation stark vereinfachen. Da immer entweder der Sinus oder der Cosinus einen Wert ungleich Null haben, liegt die Idee nahe, die Addition durch eine entsprechende alternierende Tabelle bei der P  $\rightarrow$  R-Wandlung zu ersetzen, die auch schon die Multiplikationen mit  $-1$  enthält. Zur näheren Untersuchung werde folgende Tabelle 1.6.3.2 aufgestellt. Um nicht zuviel Platz zu beanspruchen, wurde  $N = 4$  gewählt.

$k$	$a_i$	$\hat{\varphi}(kT_A)$	$\Re$	$\Im$	$\Re \cdot \cos \dots + \Im \cdot \sin \dots$
0	1	0	$x_0$	$y_0$	$x_0$
1	1	1	$x_1$	$y_1$	$-y_1$
2	1	2	$x_2$	$y_2$	$-x_2$
3	1	3	$x_3$	$y_3$	$y_3$
4	0	4	$x_4$	$y_4$	$x_4$
5	0	3	$x_3$	$y_3$	$-y_3$
6	0	2	$x_2$	$y_2$	$-x_2$
7	0	1	$x_1$	$y_1$	$y_1$
8	?	0	$x_0$	$y_0$	$x_0$

Tabelle 1.5: Ablauf des Mischprozesses

Es ist zu erkennen, daß bei gleichem Phasenwert immer die gleichen Werte aus der Tabelle gelesen werden mit der Ausnahme, daß der Ausgabewert bei

$$a_i = 0 \wedge k \text{ ungerade} \quad (1.78)$$

negiert wird. Dieser Sachverhalt läßt sich leicht für andere Phasenbereiche als  $0 \dots \frac{\pi}{2}$  überprüfen. Es ist daher möglich, die Tabelle als Abbildung  $X^1 \mapsto Y^1$  zu gestalten und zusätzlich den Addierer einzusparen. Es wird einzig noch eine zusätzliche Negation nötig, die jedoch im Falle einer Implementierung mit Addierer gleich zweimal benötigt worden wäre. Damit entspricht die Auflösung dieser kombinierten Tabelle direkt der DA-Wanderauflösung.

#### 1.6.4 Implementierungsvorschlag

Abbildung 1.31 zeigt eine mögliche Implementierung für den digitalen Teil des MSK-Senders. Dabei wurde bis auf die Bitebene optimiert. Die Wortbreite des Vor- / Rückwärtszählers ist  $Z = ld(4N) = 2 + ld(N)$ . Für  $N = 8$  folgt  $Z = 5$ . Da es keine käuflichen DA-Wandler gibt, die eine Kennlinie wie (1.63) haben, wird die Quantisierung entsprechend (1.63) einfach durch die Tabelleneinträge realisiert.

### 1.7 Bitfehlerraten nach DA-Quantisierung

In diesem Abschnitt werden die Bitfehlerraten, die sich durch die erläuterte Sender- und Empfängerstruktur unter Berücksichtigung der DA-Quantisierung ergeben, diskutiert. Um die Auswirkung der verschiedenen Fehlerquellen des Senders auf den Empfänger untersuchen zu können, muß hier bereits ein recht komplexes Modell des Transceivers eingeführt werden. Insbesondere muß die Performance des Frequenz- und Taktoffsetschätzers im Empfänger berücksichtigt werden. Für eine genauere Erläuterung dieser Einheiten muß jedoch auf das nächste Kapitel, das sich ausschließlich mit dem Empfänger befaßt, verwiesen werden.

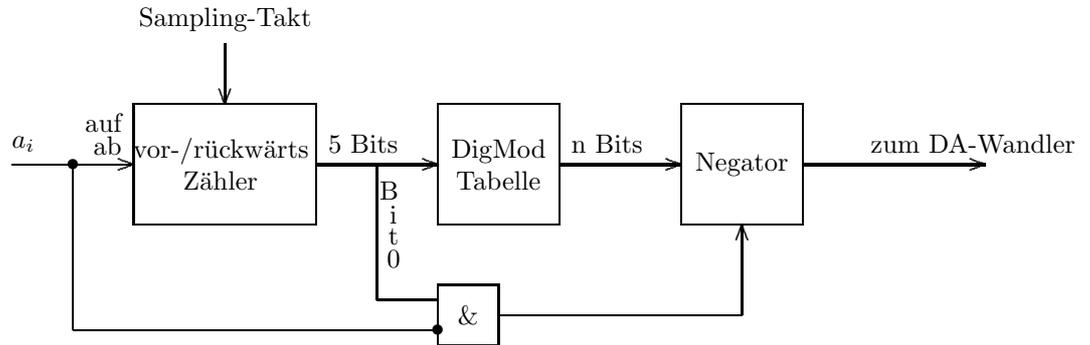


Abbildung 1.31: Implementierungsvorschlag für den digitalen Teil des MSK-Senders

Sofern nichts anderes beschrieben, werden in diesem Abschnitt die Simulationsparameter wie folgt eingestellt:

1. SNR = 10dB
2.  $f_{\Delta} = \frac{1.5}{T}$
3.  $\Delta f T = 18.75\%$

Es werden drei verschiedene Modelle untersucht. Sie besitzen alle die gleiche Struktur, unterscheiden sich jedoch in der Zahl der Abtastwerte pro Symbol an den einzelnen Verarbeitungsstufen. In einer späteren Implementation soll voraussichtlich ein System eingesetzt werden, das das Bandpaßsignal im Empfänger mit der Rate  $8/T$  abtastet und mischt. Datendetektion und Schätzer sollen jedoch aus Aufwandsgründen nur mit der Rate  $4/T$  arbeiten. Dies sei Modell 1. Um eine Abschätzung über den Performance-Verlust des Empfängers durch diese Parameterwahl zu gewinnen, werden als Referenz zwei weitere Transceivermodelle untersucht. Sie arbeiten im Sender und Empfänger durchweg mit der gleichen Abtastrate. Für Modell 2 beträgt sie  $4/T$ , für Modell 3  $8/T$ . Diese beiden Modelle sind so gewählt, daß Modell 1 bezüglich des Implementationsaufwandes dazwischen liegt. Dadurch kann beurteilt werden, ob es möglich ist, im Sender und Empfänger zugleich mit einer Abtastrate von nur  $4/T$  zu arbeiten, oder ob im Gegenteil durch die Dezimation auf die Rate  $4/T$  vor Schätzer und Datendetektion zuviel Information verschenkt wird.

Der Taktoffset werde im folgenden mit  $\varepsilon$  bezeichnet und ist so definiert, daß  $t$  auf der rechten Seite von (1.1) durch  $t - \varepsilon \frac{T}{8}$  substituiert wird:

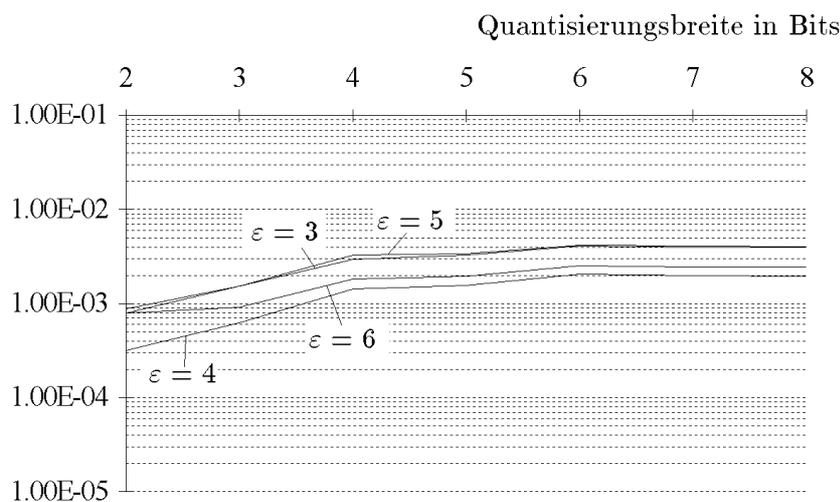
$$s(t) = \Re\{e^{j(2\pi f_c(t - \varepsilon \frac{T}{8}) + \varphi(t - \varepsilon \frac{T}{8}))}\} \quad (1.79)$$

$\varepsilon$  bezieht sich daher immer auf eine Abtastrate von  $8/T$ .

### 1.7.1 Modell 1

Abbildung 1.32 zeigt die Bitfehlerraten für verschiedene Taktoffsets. Man erkennt den im Abschnitt 1.6 diskutierten Effekt, daß die Bitfehlerraten mit sinkender DA-Auflösung geringer werden. Die Bitfehlerraten sind für  $\varepsilon = 3$  und  $\varepsilon = 5$  schlechter, da hier aufgrund der Überabtastung von nur 4 ein Taktfehler von  $T/8$  übrig bleibt. Der Verlust, der durch diesen Restdezimationsfehler verursacht wird, wird für kleinere Auflösungen größer. Dies liegt an der Zunahme der Welligkeit der Differenzphase (vgl. Abb 1.30), so daß ein genaues Treffen des idealen Abtastpunktes an Bedeutung gewinnt.

Es ist zu erkennen, daß für  $\varepsilon = 4$  Bitfehlerraten im Bereich von  $3 \cdot 10^{-4}$  zu erreichen sind. Ein Wert der fast um einen Faktor 10 besser ist als derjenige für reines MSK. Für  $\varepsilon = 6$  verschlechtern sich die Fehlerraten stark, obwohl mit dem richtigen Schätzwert für den Taktoffset gleichermaßen wie für  $\varepsilon = 4$  der Restdezimationsfehler 0 beträgt. An dieser Stelle macht sich der Taktoffset-Schätzfehler bemerkbar. Der Schätzer trifft nicht immer den richtigen Taktoffset. Der richtige Taktoffsetschätzwert wäre für  $\varepsilon = 6$  wegen der Überabtastung von nur 4 gleich 3. Liegt der Schätzer eine Stufe unterhalb dieses richtigen Wertes so erreicht er 2. Liegt er oberhalb, so müßte er 4 schätzen, was jedoch nicht möglich ist. Er würde daher den Taktoffset auf 0 schätzen. Durch diesen Effekt kann es passieren, daß Symbole doppelt detektiert werden [3].



BER

Abbildung 1.32: Bitfehlerraten bei Modell 1 für verschiedene Taktoffsets, abhängig von der Quantisierungsbreite

Abbildung 1.33 zeigt die Fehlerraten der Taktoffsetschätzung. Es wird zwischen Taktoffset-Schätzfehlern von 1 und größer unterschieden. Für einen Taktoffset

von  $\varepsilon = 4$  ist der einfache Taktfehler dominierend, besonders für kleine Quantisierungsbreiten. Zweifache Taktfehler sind relativ selten und treten für Quantisierungsbreiten  $\leq 4$  in der Simulation nicht mehr auf. Dieses Bild ändert sich für  $\varepsilon = 6$ . Der oben beschriebene Effekt der Doppeldetektion macht sich in der mehrfach-Fehlerrate bemerkbar. Es ist davon auszugehen, daß bei einer Fehlde-zimierung  $> 1$  kein vernünftiges Datum mehr zugewinnen ist. Daher steigt die Bitfehlerrate in Abbildung 1.32 für  $\varepsilon = 6$  für kleine Quantisierungsbreiten stark und für größere Quantisierungsbreiten ein wenig an.

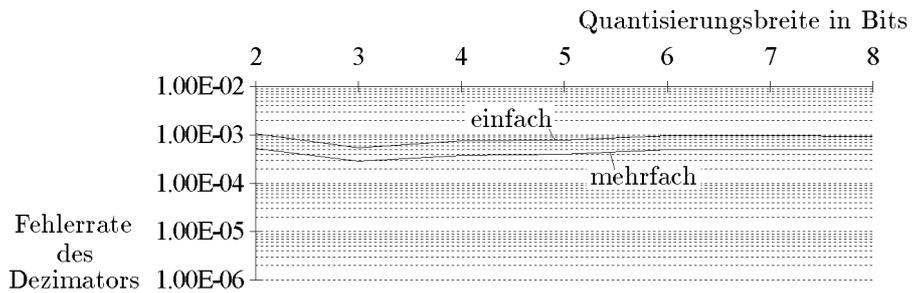
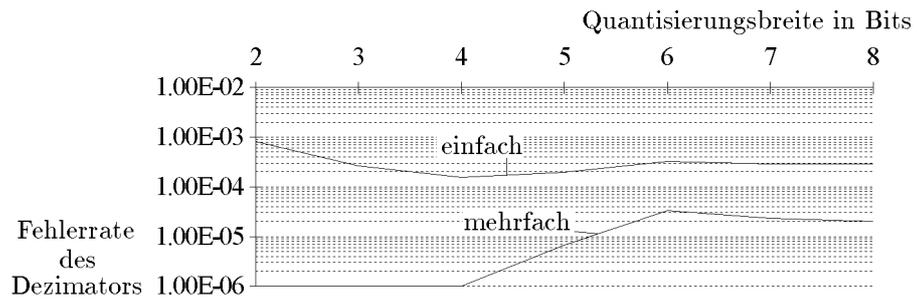


Abbildung 1.33: Fehlerraten des Taktoffsetschätzers bei Modell 1, einfach und mehrfach, oben:  $\varepsilon = 4$ , unten:  $\varepsilon = 6$

Die Varianz der Frequenzoffsetschätzung ist in Abbildung 1.34 dargestellt. Hierbei zeigt sich nur noch eine leichte Abhängigkeit von dem gewählten Taktoffset. Daß hier sogar die Schätzwerte für den Frequenzoffset mit geringerer Quantisierungsbreite besser werden, liegt an dem gleichen Effekt, der auch für die Abnahme der Bitfehlerraten verantwortlich ist. Dieser Zusammenhang wird in Abschnitt 2.3.1.2 näher erläutert.

### 1.7.2 Modell 2 und 3

Abbildung 1.35 zeigt Bitfehlerraten, Varianz der Frequenzoffsetschätzung und Fehlerrate der Taktoffsetschätzung für die Referenzmodelle. Der eingestellte

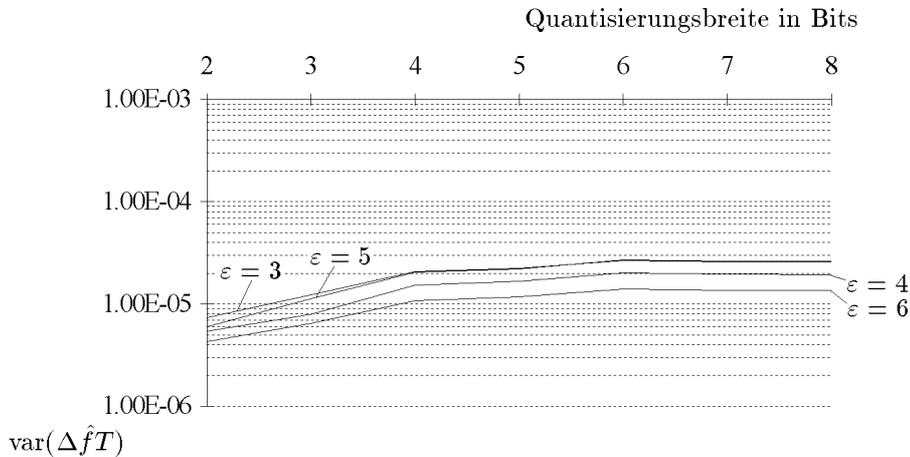


Abbildung 1.34: Varianz der Frequenzoffsetschätzung bei Modell 1 für verschiedene Taktoffsets, abhängig von der Quantisierungsbreite

Taktfehler bei Modell 2 und 3 beträgt  $T/2$ . Es fällt der große Anstieg der Bitfehlerraten bei Modell 2 auf. Modell 3 weist geringfügig schlechtere Bitfehlerraten als Modell 2 auf. Dies erscheint zunächst etwas merkwürdig, kann jedoch leicht durch gestiegene Fehlerrate bei Taktoffsetschätzung erklärt werden. Das Kanalrauschen macht sich bei gesteigerter Genauigkeit für die Taktoffsetschätzung stärker bemerkbar. Um eine Schätzung mit gleicher Sicherheit wie für Modell 1 zu erreichen, ist ein längerer Schätzer vorzusehen. Bei der Beurteilung der Bitfehlerraten von Modell 3 ist natürlich auch zu beachten, daß die maximal auftretende Fehldezimation nur noch  $T/16$  beträgt. Das heißt, daß der Unterschied zwischen dem Best- und Worst-Case bei der Dezimation wesentlich geringer ausfällt. Für einen Vergleich müßte daher über die Best- und Worst-Case Fehleraten von Modell 1 gemittelt werden. Nach dieser Anpassung gewinnt Modell 3 einen leichten Vorsprung gegenüber Modell 1. Ob dieser leichte Performancegewinn den erhöhten Aufwand rechtfertigt, muß in Zusammenhang mit den Fertigungskosten gesehen werden.

## 1.8 Dämpfungsverluste durch den ILO

Die Bandpaßfilterung in Abbildung 1.1 dient zur Begrenzung des Spektrums, das über die Antenne abgestrahlt wird. Diese Begrenzung wird durch die amplitudenstabilisierende Funktion des ILOs teilweise wieder beseitigt. Das Antennensignal enthält daher Oberwellen, die nicht zum bandbegrenzten MSK-Spektrum gehören. Um entscheiden zu können, wie groß der Abstand der einzelnen MSK-Kanäle in einem FDMA-Konzept später sein muß, wird die Verbreiterung des Spektrums durch den ILO hier näher untersucht. Abbildung 1.36 zeigt in je-

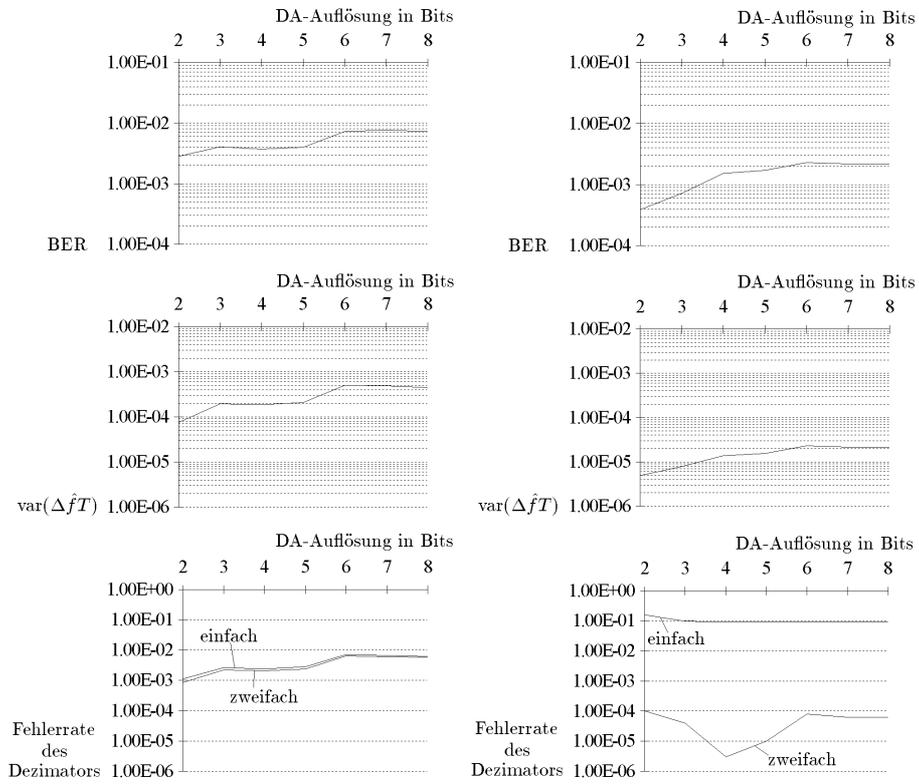


Abbildung 1.35: Simulationsergebnisse, links:  $N = 4$  (Modell 2), rechts:  $N = 8$  (Modell 3)

weils einem Diagramm das Spektrum vor und hinter dem ILO für verschiedene Filterbandbreiten, falls ein MSK-Signal übertragen wird. Man erkennt das Abschneiden des Spektrums durch den idealen Bandpaß, sowie die Störungen, die durch den ILO verursacht werden. Abbildung 1.37 zeigt die gleichen Diagramme für den Fall, daß ein OQPSK-Signal übertragen wird, das durch eine Quantisierungsbreite von 2 Bit erzeugt wird. Es ist deutlich eine Aufweitung des Spektrums zu beobachten. Dies rührt daher, daß das bandbegrenzte OQPSK-Spektrum zu den Rändern des Durchlaßbereiches hin eine größere Leistungsdichte hat. Der Spektralbereich außerhalb des durch  $f_{\Delta}$  gegebenen Bereichs trägt jedoch nicht zur Übermittlung der Daten bei und kann daher, falls es die Implementation erlaubt, bei Bedarf durch ein zusätzliches Filter entfernt werden, zumal im Empfänger die Bandbreite des genutzten Signals wegen der Begrenzung der Rauschleistung ebenfalls auf  $f_{\Delta}$  begrenzt wird.

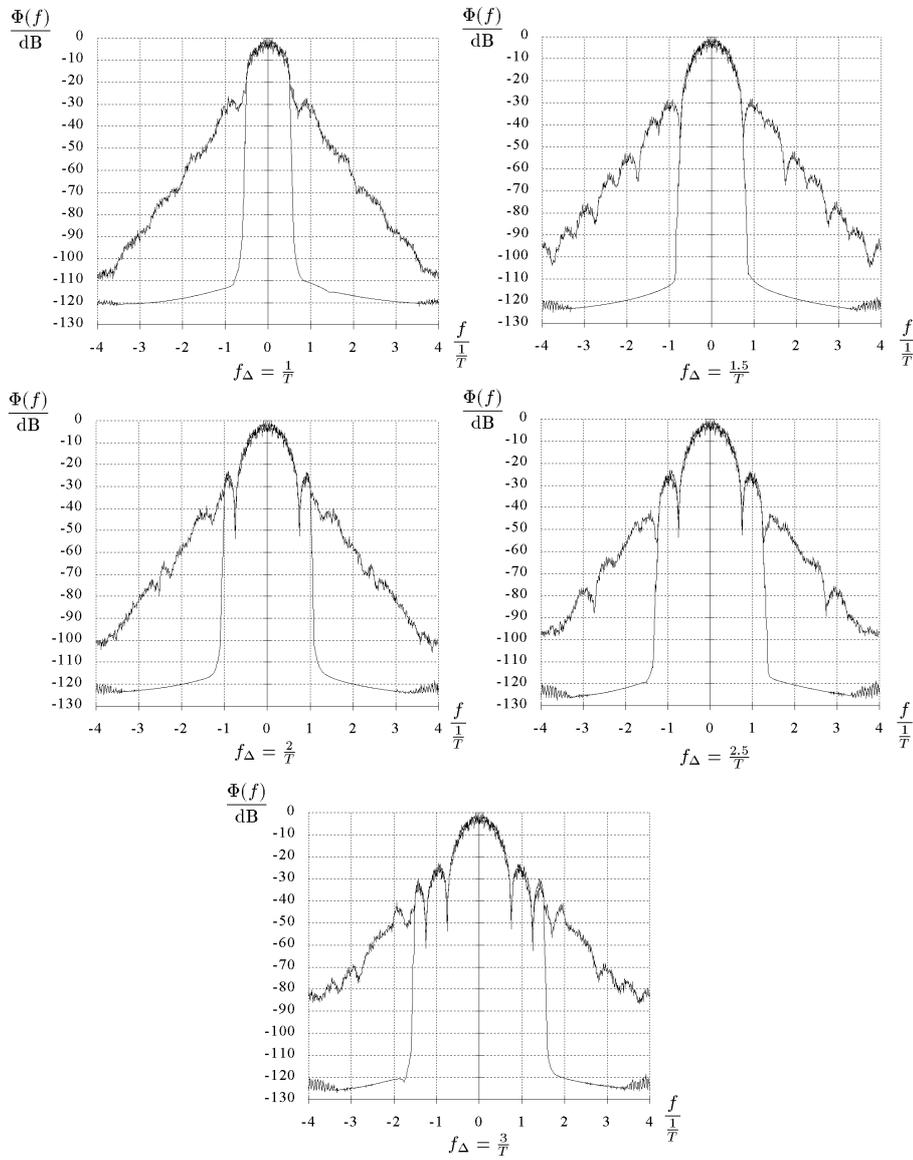


Abbildung 1.36: "ideal" bandbegrenztes Spektrum vor und nach ILO für MSK

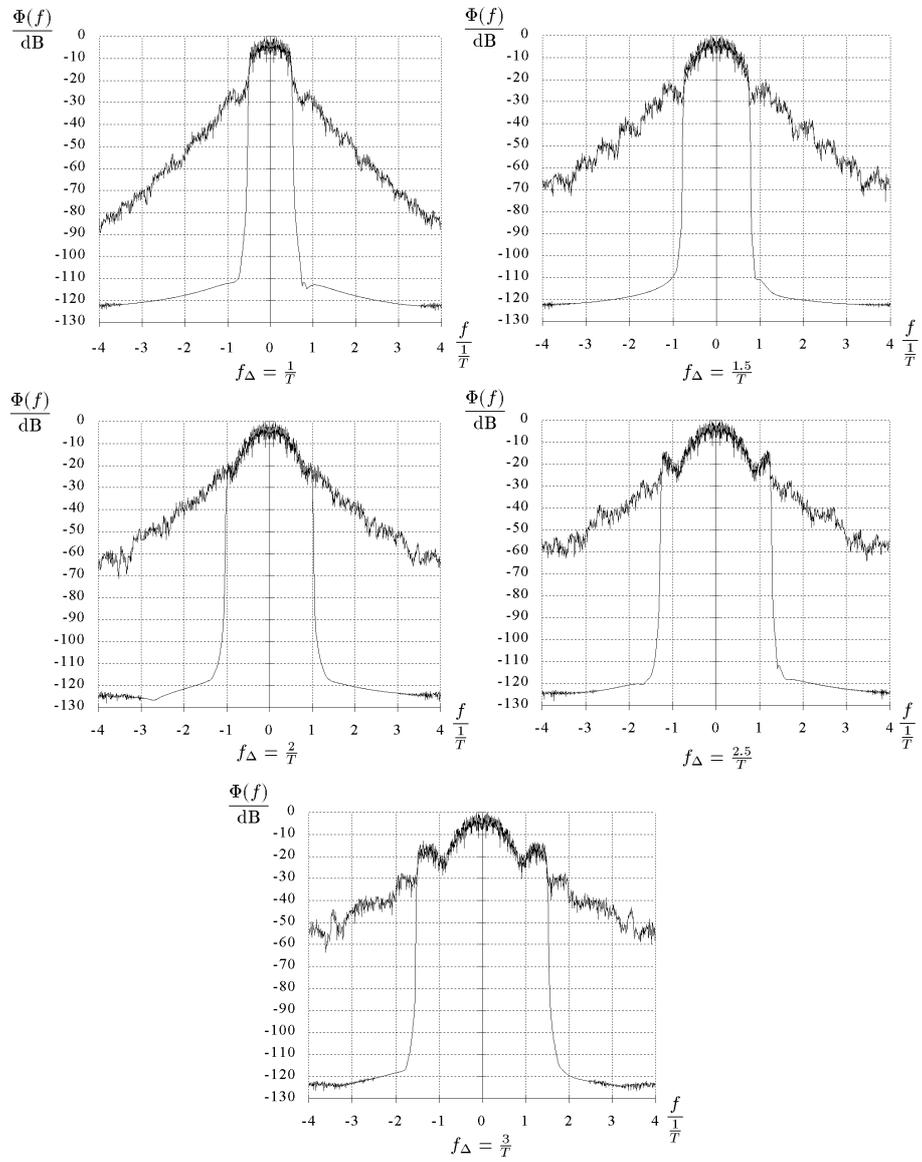


Abbildung 1.37: “ideal“ bandbegrenzttes Spektrum vor und nach ILO für OQPSK durch Quantisierung

### 1.8.1 Austausch: Spektraldämpfung gegen BER

In diesem Abschnitt wird ein Vergleich zwischen der benötigten Übertragungsbandbreite an der Antenne und den erreichbaren Fehlerraten beschrieben. Die Abbildung 1.38 zeigt die Spektren hinter dem ILO bei einer Bandbreite von  $f_{\Delta} = \frac{1.5}{T}$  für verschiedene Quantisierungsbreiten. Die oberste Kurve zeigt das ILO-Spektrum für eine Quantisierungsbandbreite von 2 Bit, die unterste dasjenige für 8 Bit. Von 3 nach 4 Bit Auflösung ist deutlich ein Sprung in der benötigten Bandbreite zu erkennen. Ein Vergleich mit Abbildung 1.32, die die Fehlerraten für die unterschiedlichen Quantisierungsbreiten zeigt, ermöglicht das Abwägen zwischen erforderlicher Bandbreite des Spektrums an der Antenne und der sich ergebenden Bitfehlerrate. Es bietet sich eine mit 4 Bit DA-Wandlungsauflösung arbeitende Implementation an, da einerseits die Dämpfung des ILO-Spektrums dann bereits tolerierbare Werte erreicht, und sich andererseits noch bessere Bitfehlerraten ergeben als für bessere Quantisierungsaufösungen.

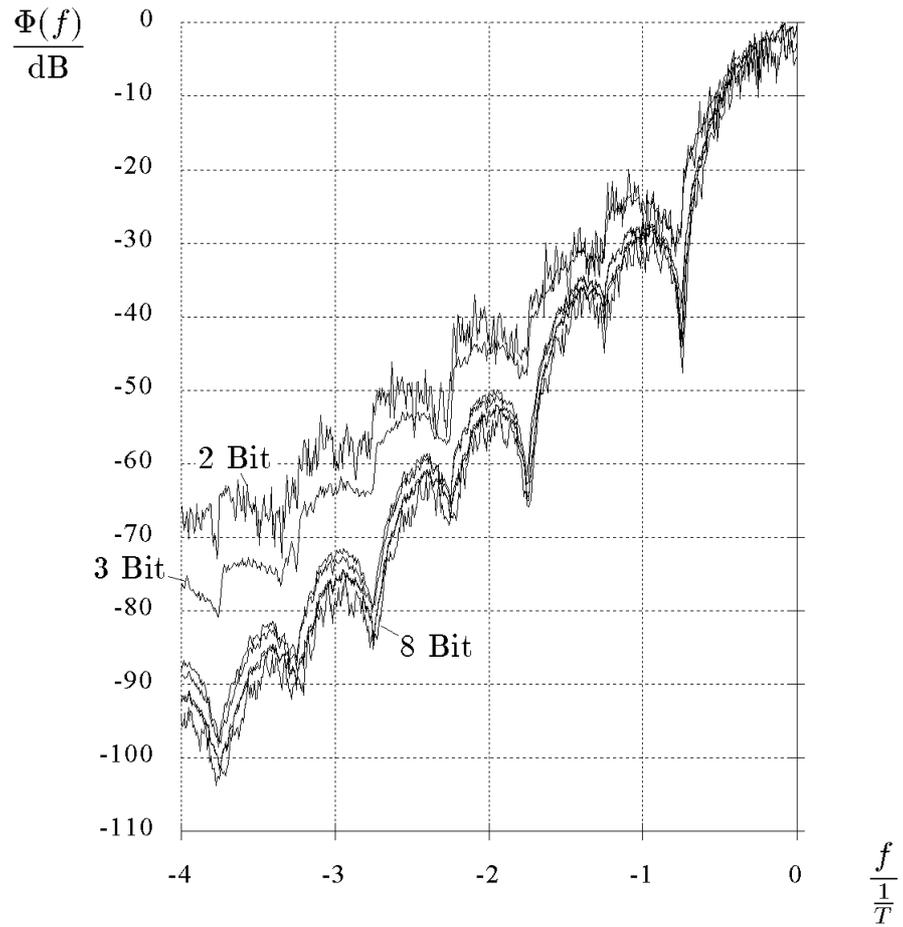


Abbildung 1.38: Spektrum nach ILO

## Kapitel 2

# Der Receiver

Dieses Kapitel beschäftigt sich mit dem Empfänger des in dieser Arbeit behandelten MSK-Transceiver-Konzepts. Abbildung 2.1 zeigt die Struktur des Empfängers. Die Betrachtung beginnt hier mit dem Abtasten des Empfangssignals in der Zwischenfrequenzebene auf  $f_1$ . Für diese Zwischenfrequenz  $f_1$  gelten die gleichen Anforderungen wie im vorigen Kapitel, um wieder mit der sehr einfachen Modulation bzw. Demodulation durch Multiplikation mit den Folgen  $1\ 0\ -1\ 0\ \dots$  arbeiten zu können. Nach der AD-Wandlung schließt sich die Mischung und Filterung mit dem FIR-Filter  $H(z)$  an. Dem Filter  $H(z)$  kommt dabei Aufgabe zu, Mischspiegelfrequenzen und etwaigen DC-Offset des AD-Wandlers zu beseitigen. Die folgende R→P-Wandlung und Dezimation wird in einer späteren Implementierung in umgekehrter Reihenfolge erfolgen. Hier wurde der Übersichtlichkeit halber die R→P-Wandlung mit in den ersten Verarbeitungsblock einbezogen. Nach der Differenzphasenbildung spaltet sich das Signal in zwei Zweige auf. Der erste Zweig liefert das Basissignal für die Takt- und Frequenzoffsetschätzung. Im zweiten Zweig erfolgt später die eigentliche Daten- und Präambeldetektion. Die vom Schätzer gelieferten Werte werden zur Kompensation und Dezimation der Differenzphasenwerte benutzt. Das Latch ist, falls kein Datenblock detektiert wird, für die Schätzwerte durchlässig. Wird die Präambel erkannt, werden Frequenz- und Taktoffset zum Zwecke der optimalen Datendetektion eingefroren, und die detektierten Datenbits werden zur Weiterverarbeitung bereitgestellt.

### 2.1 Das Empfangsfilter $H(z)$

Das vom AD-Wandler gelieferte Signal  $s_1(kT_A)$  wird durch Multiplikation  $e^{j2\pi k f_c T_A}$  in den Tiefpaßbereich gemischt.

$$s_2 = s_1 \cdot e^{j2\pi k f_c T_A} \quad (2.1)$$

Eine Filterung muß linearphasig sein, um die spätere Datendetektion zu ermöglichen. Damit scheiden IIR-Filter aus. Sie hätten zumal den Nachteil, daß sie, um

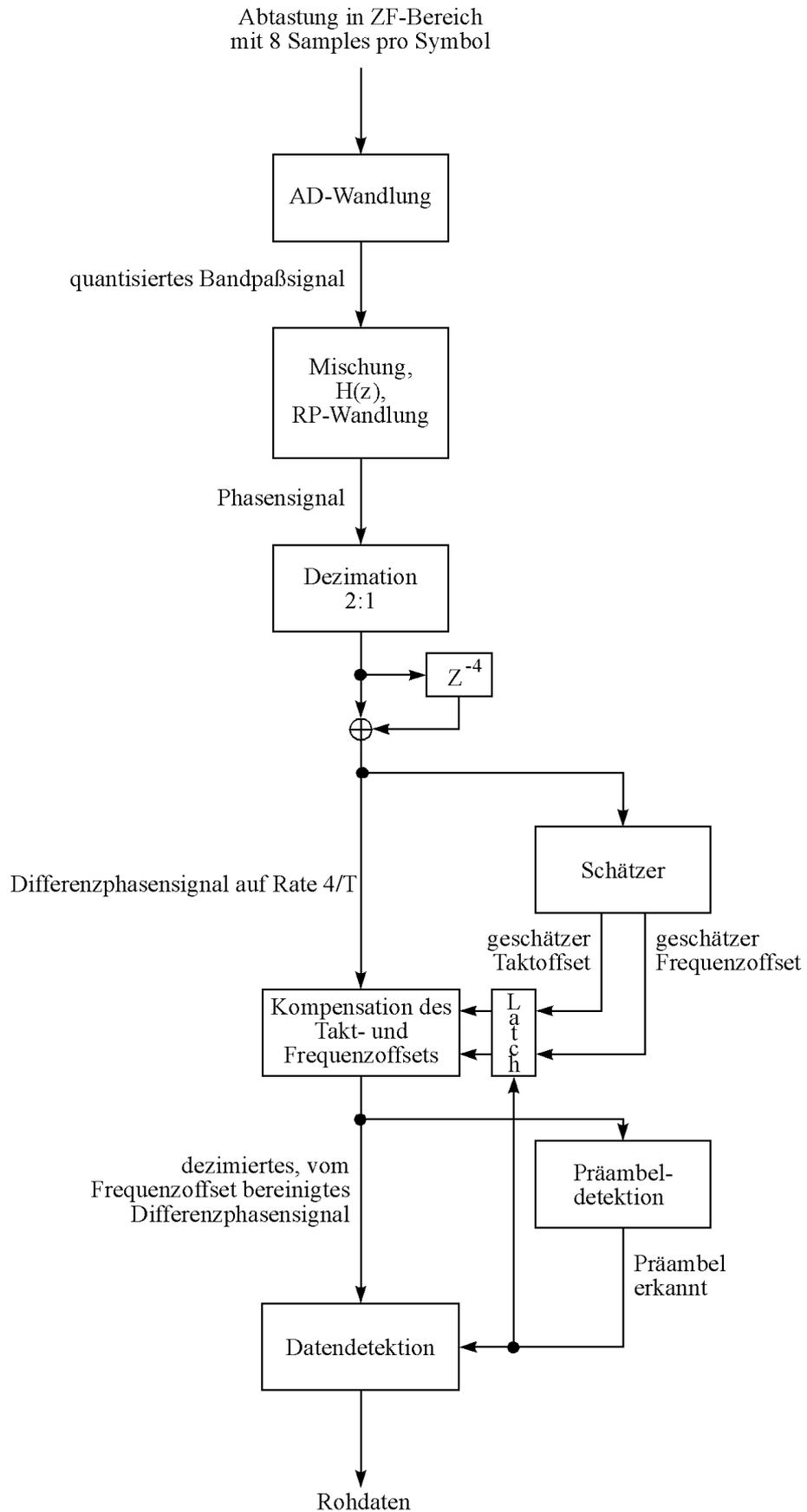


Abbildung 2.1: Struktur des Receivers

eine ausreichende Stabilität zu erreichen, eine große Rechengenauigkeit erforderten. Daher wird hier ein FIR-Filter eingesetzt, das einen Kompromiß aus gewünschtem Frequenzgang und Aufwand darstellt. Es sind zwei Anforderungen zu erfüllen: Die Beseitigung eines eventuell vorhandenen dc-Offsets, sowie die Entfernung von durch die Mischung entstandenen Spiegelfrequenzen.  $H(z)$  wird daher gebildet durch das Produkt von  $H_{\text{Misch}}$  und  $H_{\text{dc}}$ .

### 2.1.1 Beseitigung des dc-Offsets

Enthält das vom AD-Wandler gelieferte Signal einen dc-Offset, so macht sich dieser im komplexen Tiefpaßbereich als eine mit  $f_1$  oszillierende Schwingung bemerkbar. Zur Entfernung dieser Schwingung wird im Frequenzgang von  $H_{\text{dc}}(z)$  bei der Frequenz  $f_1$  eine Nullstelle eingefügt. Mit  $f_1 = \frac{N}{4T}$  gilt:

$$z = e^{sT_A} \quad (2.2)$$

$$H_{\text{dc}}(z = e^{\pm j\omega_0 T_A}) \stackrel{!}{=} 0 \quad (2.3)$$

$$\omega_0 = 2\pi f_1 = 2\pi \frac{1}{4T_A} = \frac{\pi}{2T_A} \implies \quad (2.4)$$

$$H_{\text{dc}}(z = e^{\pm j\frac{\pi}{2}}) = 0 \quad (2.5)$$

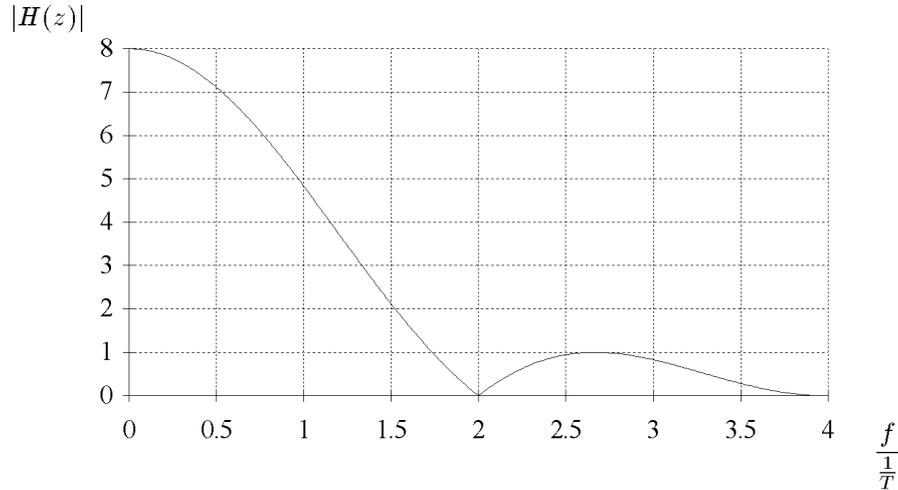
$$H_{\text{dc}}(z) = (z + j) \cdot (z - j) = z^2 + 1 \quad (2.6)$$

### 2.1.2 Herausfiltern der Misch-Spiegelfrequenzen

Die Mischung vom Bandpaßsignal in den komplexen Tiefpaßbereich erzeugt Spiegelfrequenzen die herausgefiltert werden müssen. Für den hier wieder interessierenden Fall  $f_1 = \frac{N}{4T}$  sitzt das MSK-Spektrum im Bandpaßbereich genau in der Mitte zwischen  $f = 0$  und  $f = \frac{N}{2T}$ , der Frequenz ab der das Spektrum periodisch wiederholt wird. Aus diesem Grund wird das Spektrum nach der Mischung achsensymmetrisch zu  $f = \pm f_1$  sein. Dadurch entsteht ähnlich wie in Kapitel 1 ein Überlappungsfehler. Bei  $f = f_1$  ist wegen der Symmetrie genau der Punkt erreicht, an dem das wiederholte MSK-Spektrum größer wird als dasjenige um  $f = 0$ . Durch einen Tiefpaß mit der Übertragungsfunktion

$$H(f) = \begin{cases} 1 & \text{für } |f| < f_1 \\ 0 & \text{für } |f| \geq f_1 \end{cases} \quad (2.7)$$

würde der störende Teil weggefiltert. Mit vertretbarem Aufwand läßt sich ein solches Filter mit hoher Genauigkeit jedoch nicht verwirklichen. Daher wird hier so vorgegangen, daß Nullstellen in dem Bereich  $f_1 \leq f \leq 2f_1$  eingefügt werden. Da  $H(z)$  später das Produkt von  $H_{\text{Misch}}$  und  $H_{\text{dc}}$  sein wird, können an dieser Stelle bereits die Nullstellen von  $H_{\text{dc}}$  bei  $f = f_1$  mit genutzt werden. Wünschenswert wäre eine Nullstelle bei  $f = 2f_1$  sowie eine zwischen  $f_1$  und  $2f_1$ . Dabei ist jedoch zu beachten, daß die Nullstellen für  $f \neq 0$  und  $f \neq 2f_1$  immer konjugiert komplex auftreten, so daß in diesem Bereich gleich zwei Nullstellen liegen müßten. Zusätzlich sollte  $H(z)$  möglichst einfach zu implementierende

Abbildung 2.2: Übertragungsfunktion des Empfangsfilters  $H(z)$ 

Polynomkoeffizienten haben. Wird die Nullstelle bei  $f = 2f_1$  doppelt ausgeführt, wird die Durchlaßdämpfung zwischen  $f_1$  und  $2f_1$  ausreichend klein, so daß auf Nullstellen in diesem Bereich vollständig verzichtet werden kann. Das heißt:

$$H_{\text{Misch}}(z) = (z + 1)^2 \quad (2.8)$$

Damit ergibt sich für das Gesamtfilter  $H(z)$ :

$$H(z) = (z^2 + 1) \cdot (z + 1)^2 \quad (2.9)$$

$$= z^4 + 2z^3 + 2z^2 + 2z + 1 \quad (2.10)$$

Dieses Filter hat zudem entscheidende Implementationsvorteile, da es zu Koeffizienten führt, die sich vollständig durch wenige versetzte Additionen ausführen lassen. Abbildung 2.2 zeigt den Verlauf der Übertragungsfunktion.

## 2.2 Wortbreiten der Verarbeitungsstufen

In dem folgenden Abschnitt wird die notwendige Auflösung in den einzelnen Verarbeitungsstufen des Empfängers so ermittelt, daß keine wesentlichen Performance-Einbußen auftreten. Dazu werden die Stufen zunächst einzeln quantisiert betrachtet, die benötigte Wortbreite ermittelt und anschließend die Stufen mit der jeweils ermittelten Wortbreite zusammengefügt, um den resultierenden Performance-Verlust zu bestimmen. Sofern nichts anderes beschrieben, werden in diesem Abschnitt die Simulationsparameter wie folgt eingestellt:

1. SNR = 10dB

2.  $f_{\Delta} = \frac{1.5}{T}$
3.  $\Delta fT = 18.75\%$

### 2.2.1 AD-Wandler

#### 2.2.1.1 Bitfehlerraten mit Frequenzoffset und Taktfehler

Abbildung 2.3 zeigt Simulationsergebnisse für unterschiedliche AD-Auflösungen mit dem Taktfehler  $\varepsilon$  als Parameter. Bereits für 5 Bit Eingangswortbreite erreichen die Fehlerraten annähernd ihren Endwert. Als konservative Wahl kann eine Auflösung von 6 bzw. 7 Bit gelten. Da der Aufwand für die sich anschließende Signalverarbeitung bis einschließlich  $H(z)$  nur linear mit der Auflösung wächst, kann an dieser Stelle etwas Sicherheit eingeplant werden. Der bei der DA-Wandlung auftretende Effekt einer Performance-Verbesserung mit abnehmender Auflösung tritt hier nicht auf.

[ Um eine Abschätzung über den Performance-Verlust des Empfängers durch die Abtastung zu gewinnen, werden als Referenz zwei weitere Empfängerstrukturen untersucht. Die einzelnen Abbildungen enthalten die Simulationsergebnisse für:

- Abbildung ??: einen Transceiver, der im Sender und Empfänger mit einer Abtastrate von  $N = 4$  arbeitet (Modell 1). Die Länge des Schätzers beträgt  $L = 64$ . Der Taktfehler beträgt  $\varepsilon = 2T_A$ . Bei der Kurve, die bessere Bitfehlerraten ergibt, ist der eingestellte Taktfehler und Frequenzoffset ideal kompensiert. Das heißt, daß der Schätzer deaktiviert ist. Auf diese Weise läßt sich der Performance-Verlust durch den Schätzfehler ablesen.
- Abbildung ??: einen Transceiver, der im Sender und Empfänger mit einer Abtastrate von  $N = 8$  arbeitet (Modell 2). Die Länge des Schätzers beträgt wieder  $L = 64$ . Dies bedeutet einen um den Faktor 2 erhöhten Aufwand für den Schätzer. Der Taktfehler beträgt nun  $\varepsilon = 4T_A$  und entspricht damit wieder dem gleichen zeitlichen Fehler von einer halben Symboldauer  $\frac{T}{2}$ . Bei der unteren Kurve ist der Schätzer wieder deaktiviert. Man erkennt, daß der Performance-Verlust durch den Schätzer hier sehr gering ausfällt.
- Abbildung 2.3: das System, das im Sender mit  $N = 8$ , bei der Demodulation mit  $N = 8$  und bei der Schätzung und Detektion mit  $N = 4$  arbeitet (Modell 3). Die Länge des Schätzers beträgt wieder  $L = 64$ . Hier sind Taktfehler von  $\varepsilon = 3T_A, 4T_A$  eingestellt, so daß sich vier Kurven ergeben. Der Performance-Verlust, der sich durch einen permanenten Abtastzeitpunktfehler von  $1T_A$  bei  $\varepsilon = 3T_A$  einstellt, läßt sich dadurch leicht ablesen.

Der Vergleich zwischen den Simulationswerten von Modell 3 und Modell 2 jeweils für  $\varepsilon = 4T_A$  zeigt, daß Modell 2 für höhere AD-Auflösungen besser wird als Modell 3. Dies ist dadurch zu erklären, daß im Falle eines Schätzfehlers im Modell 3 direkt ein Abtastzeitpunktfehler von  $2T_A$  auftritt, während im Modell 2 der Fehler nur  $1T_A$  beträgt. ]

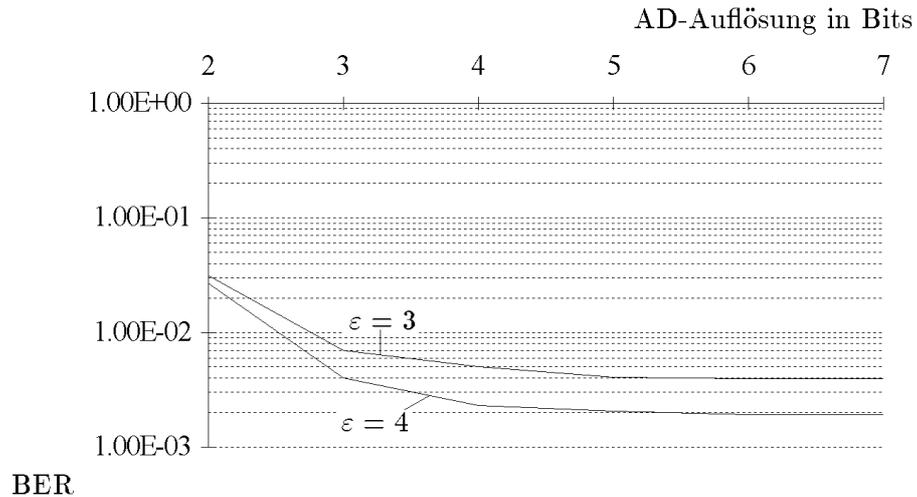


Abbildung 2.3: Bitfehlerraten in Abhängigkeit von der Auflösung der AD-Wandlung

### 2.2.1.2 Eingangswertebereich und dc-Offset

Im Empfänger befindet sich vor dem AD-Wandler ein Bandpaß-Limiter, der die Amplitude des empfangenen Signals fest auf einen Wert setzt. Dies ist möglich, da die Amplitude des MSK-Signals keine Information enthält. Der Eingangswertebereich des AD-Wandlers ist dadurch festgesetzt. Tritt nun jedoch ein dc-Offset auf, so würde bei einem auf das Maximum des Bandpaß-Limiters ausgerichteten AD-Eingangswertebereich ein Abschneiden der Signalkuppen erfolgen. Dies würde die spätere Phasengewinnung mit einem Fehler behaften. Daher sollte bei der Auslegung des Eingangswertebereichs etwas ‐Luft‐ für einen eventuell vorhandenen dc-Offset vorhanden sein. Dabei ist zu beachten, daß diese ‐Luft‐ die zur Verfügung stehende Auflösung des AD-Wandlers verringert. Je nach Größe des zu erwartenden dc-Offsets muß daher eventuell ein Bit mehr für den AD-Wandler vorgesehen werden. Der dc-Offset wird, sofern er den Wertebereich des AD-Wandlers nicht überschreitet, im Eingangsfiler  $H(z)$  des Empfängers vollständig kompensiert. Dies kann an Abbildung 2.4 verifiziert werden. Der normierte Eingangswertebereich des AD-Wandlers ist bei der Simulation auf das Intervall  $[-0.5 \dots 0.5]$  eingestellt. Das heißt, bis zu einem dc-Offset von 0.5 findet kein Abschneiden der Signalkuppen statt. Da der dc-Offset von  $H(z)$  vollständig beseitigt wird, ändert sich die Bitfehlerrate bis zu diesem Wert nicht. Erst für dc-Offsets größer 0.5 tritt ein deutlicher Performance-Verlust auf.

### 2.2.2 Mischung

Wenn die Bedingung (1.40) erfüllt ist, beschränkt sich das Mischen auf die Multiplikation mit  $-1, 0, 1$ , so daß der Wertebereich nach dem Mischen gleich bleibt.

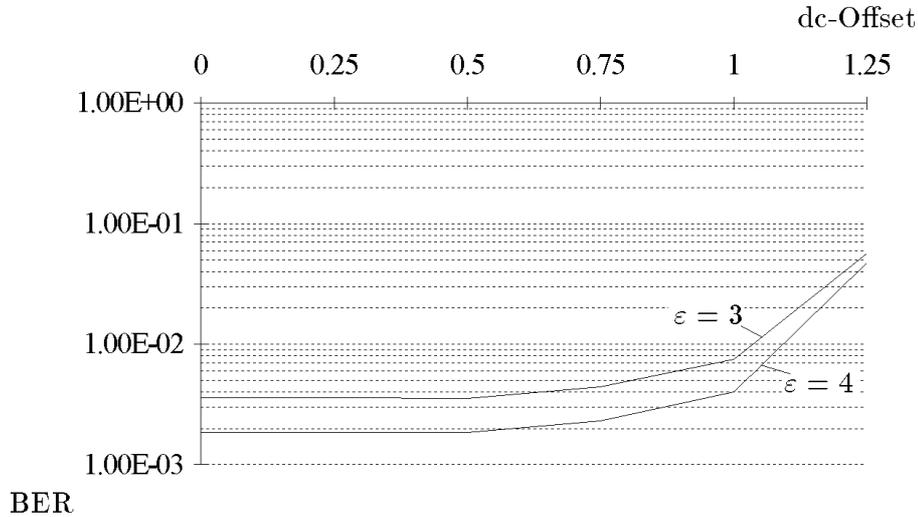


Abbildung 2.4: Einfluß des dc-Offsets auf die Bitfehlerraten

Hierbei ist zu beachten, daß die Quadraturkomponenten in der Zweierkomplementdarstellung einen Wertebereich von  $-2^{n-1}$  bis  $+2^{n-1} - 1$  haben. Das heißt, hier kann ein Überlauf auftreten, der die Performance drastisch verschlechtert. Angenommen der AD-Wandler quantisiere mit  $n$  Bits, dann liegt sein Wertebereich auch im Intervall  $[-2^{n-1}, +2^{n-1} - 1]$ . Liefert der AD-Wandler nun den Wert  $-2^{n-1}$  so kann dieser durch die Mischung invertiert und dadurch positiv werden. Der Wert  $2^{n-1}$  liegt jedoch außerhalb des Wertebereichs der im Zweierkomplement mit  $n$  Bitstellen darstellbaren ganzen Zahlen. Es wird daher ein Überlaufeffekt auftreten. Eine ganze Zahl  $a$  wird im Zweierkomplement mit  $n$  Bitstellen  $b_i$  dargestellt durch:

$$a = \sum_{i=0}^{n-2} b_i \cdot 2^i - b_{n-1} \cdot 2^{n-1} \quad (2.11)$$

Negation von  $a$  bedeutet in der Zweierkomplementdarstellung, daß 1 subtrahiert und anschließend sämtliche Bits invertiert werden. Die Darstellung von  $-2^{n-1}$  ist im Zweierkomplement:

$$b_{n-1} = 1, b_{n-2...0} = 0 \quad (2.12)$$

Wird diese Zahl negiert, so ergibt sich wieder dieselbe Zahl, also wieder  $-2^{n-1}$ . Dies bedeutet für die Weiterverarbeitung einen katastrophalen Fehler und wirkt sich auch dementsprechend stark in der Bitfehlerrate aus. Um diesen Effekt zu verhindern bieten sich mehrere Möglichkeiten an.

1. Reservierung von weiterer "Luft" im Eingangswertebereich des AD-Wandlers

2. Erhöhung der Bitstellen, so daß der Wert  $-2^{n-1}$  sicher nicht erreicht werden kann.
3. Begrenzung des AD-Wandlers auf den Zahlenbereich  $-2^{n-1}+1 \dots 2^{n-1}-1$ .
4. Einführung eines speziellen Negators, der bei der Negation von  $-2^{n-1}$   $2^{n-1} - 1$  liefern würde.

Methode 1 erscheint unpraktikabel, da sie die zur Verfügung stehende Auflösung weiter verringert. Die Reserve müßte auch ausreichend groß ausfallen, da sich die falsche Negation wesentlich verheerender auf die Bitfehlerraten auswirkt als das Clippen der Signalkurven. Methode 2 erforderte einen größeren Aufwand in der Schaltung, der die meiste Zeit ungenutzt wäre. Denkbar wären die Methoden 3 und 4 von denen die mit weniger Aufwand zu implementierende den Vorzug bekäme. Methode 3 wäre durch eine einfache Gatterlogik zu verwirklichen. Bit 0 müßte, falls der AD-Wandler den Wert  $-2^{n-1}$  liefert, zwangsweise auf 1 gesetzt werden.

$$\tilde{b}_0 = b_0 \vee \left[ b_{n-1} \wedge \bigwedge_{i=1}^{n-2} \bar{b}_i \right] \quad (2.13)$$

Der Aufwand für Methode 4 hängt von der Implementation des primitiven Multiplikators ab, dessen Bestandteil der Negator ist.

### 2.2.3 FIR-Empfangsfilter

Das FIR-Filter  $H(z) = z^4 + 2z^3 + 2z^2 + 2z + 1$  schließt sich nach der Mischung in den komplexen Tiefpaßbereich an den AD-Wandler an. Unter Berücksichtigung der im vorigen Abschnitt gemachten Einschränkung ist die Auflösung nach dem Mischen gleich der Auflösung nach dem AD-Wandler. Dabei ist im Real- bzw. Imaginärteilzweig jeder zweite Wert Null. Daher kommen von  $H(z)$  entweder die Faktoren  $z^4$ ,  $2z^2$ , 1 oder  $2z^3$ ,  $2z$  zum Tragen. In beiden Fällen erhöht sich der Wertebereich um den Faktor 4, was einer Erhöhung der Bitstellen um 2 entspricht. Die zur Verfügung stehende Auflösung ist nach  $H(z)$  daher um zwei Bits größer als die des AD-Wandlers. Die Simulation in Abbildung 2.5 zeigt, daß etwa 6 Bits hinter  $H(z)$  ausreichend sind. Dies bedeutet nicht, daß die Auflösung des AD-Wandlers nun nur 4 Bit betragen muß. Wie bereits oben gesehen, werden auch für den AD-Wandler etwa 6 Bits benötigt. Das heißt, daß die niederwertigsten Bits hinter  $H(z)$  abgeschnitten werden können.

### 2.2.4 R→P-Umwandlung nach Empfangsfilter

Abbildung 2.6 zeigt die Bitfehlerraten in Abhängigkeit von Auflösung der Phase. Es ist zu erkennen, daß mit etwa 7 Bits Phasenbreite kein nennenswerter Performance-Verlust auftritt. Dadurch kann die Breite des informationstragenden Signalpfades um den Faktor  $\frac{2.6}{7} = 1.7$  gesenkt werden. Dies wird auch für die Implementation des Schätzers eine Rolle spielen.

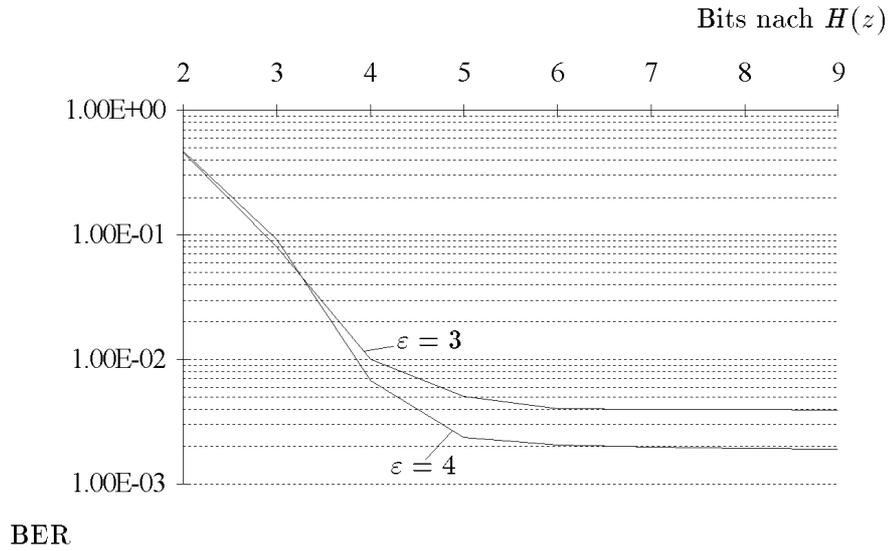


Abbildung 2.5: Bitfehlerraten in Abhängigkeit von der Auflösung hinter Empfangsfilter (Quadraturkomponenten)

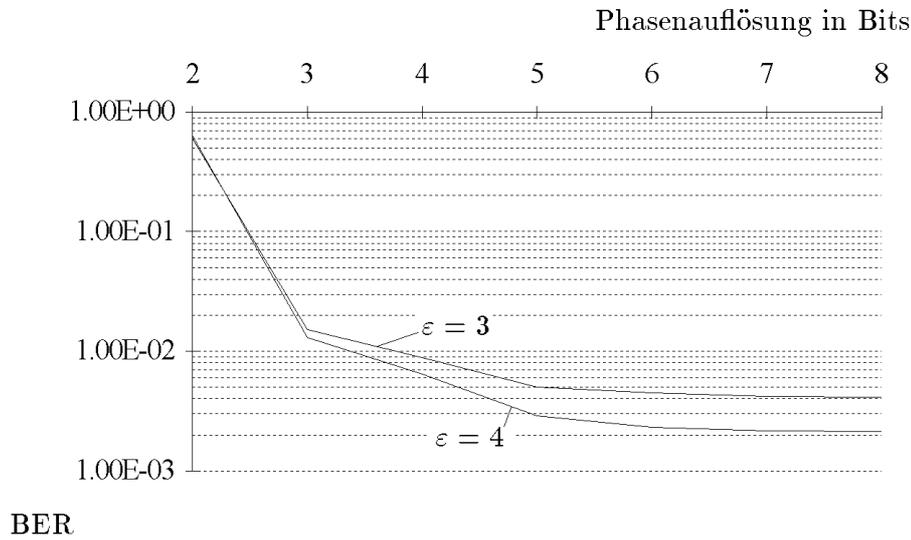


Abbildung 2.6: Bitfehlerraten in Abhängigkeit der Phasenauflösung hinter Empfangsfilter

## 2.3 Der Schätzer

### 2.3.1 Theorie

Die Theorie zu dem hier eingesetzten Schätzverfahren liefert wichtige Ergebnisse. Sie sind für eine effiziente Implementierung von großer Bedeutung. Zum einen liefert die Theorie eine direkte Konstruktionsvorschrift für einen realen Schätzer, zum anderen beschreibt sie die Bedingungen, unter denen die Schätzung besonders genau ist.

#### 2.3.1.1 Taktoffsetschätzung

Das gestörte MSK-Signal im Tiefpaßbereich ist beschrieben durch:

$$z(t) = e^{j\varphi(t-\tau)} + n(t) \quad (2.14)$$

$\tau$  ergibt sich daher bei vorgegebenen  $\varepsilon$  durch  $\tau = \varepsilon \frac{T}{N}$ . Bei der Differenzphasenbildung und anschließender Multiplikation mit 2 entsteht folgender Ausdruck:

$$c(t) = z^2(t) \cdot z^{*2}(t-T) = (z(t) \cdot z^*(t-T))^2 \quad (2.15)$$

Es läßt sich zeigen[8], [6], daß für den Erwartungswert von  $c(t)$  gilt:

$$E\{c(t)\} = \left| p + (1-p) \cdot \cos \left( (t-\tau) \frac{2\pi}{T} \right) + \tilde{n}(t) \right| \quad (2.16)$$

Darin ist  $p$  die Wahrscheinlichkeit, daß zwei benachbarte Bits gleich sind.  $\tilde{n}(t)$  ist ein Rauschterm, der vom Kanalrauschen, der Schätzfilterlänge und der Bandbreite abhängt. Durch die Abtastung wird  $t$  durch  $k \frac{T}{N}$  substituiert. Für ein festes  $k_0$  erhält man daher mit (2.16) einen Erwartungswert, dessen Größe zu dem Taktoffset  $\tau$  korrespondiert. Damit ergibt sich aus der Theorie direkt eine implementierbare Struktur eines Taktoffsetschätzers. Man berechnet für die  $N$  verschiedenen  $k = [0 \dots N-1]$  den Erwartungswert von  $c(t)$  und kann durch eine einfache Maximumsuche auf den Taktoffset  $\tau$  zurückschließen.

Die Güte des Zusammenhangs zwischen  $\tau$  und  $E\{c(t)\}$  wird jedoch von  $p$  beeinflusst. Für  $p = 1$ , einer konstanten Bitfolge, liefert  $E\{c(t)\}$  keine Information mehr. Für  $p = 0$ , einer alternierende Bitsequenz, stört die Betragsbildung in (2.16) die Erkennung des richtigen Maximums. Damit ist die Genauigkeit des Schätzers stark datenabhängig.

#### 2.3.1.2 Frequenzoffsetschätzung

Die Frequenzoffsetschätzung basiert auf folgendem Ausdruck:

$$\Delta\hat{\omega}T = \frac{1}{2} \arg \left( -E\{e^{j2\varphi_{\text{diff},i}}\} \right) \quad (2.17)$$

Im nicht bandbegrenzten Fall erreicht die Differenzphase  $\varphi_{\text{diff},i}$  in den Abtastzeitpunkten an den Symbolgrenzen die beiden Endwerte

$$\varphi_1 = \frac{\pi}{2} + \Delta\omega T, \text{ für } a_i = 1 \quad (2.18)$$

$$\varphi_0 = -\frac{\pi}{2} + \Delta\omega T, \text{ für } a_i = 0. \quad (2.19)$$

Darin ist  $\Delta\omega$  der Frequenzoffset, den es zu schätzen gilt. Führt man die Wahrscheinlichkeiten  $p_1$  und  $p_0$  dafür ein, daß eine 1 bzw. eine 0 gesendet wurde, berechnet sich der Erwartungswert in Gleichung (2.17) wie folgt:

$$\begin{aligned} \Delta\hat{\omega}T &= \frac{1}{2} \arg(-p_1 \cdot (-1) \cdot e^{j2\Delta\omega T} - p_0 \cdot (-1) \cdot e^{j2\Delta\omega T}) \\ &= \frac{1}{2} \arg((p_1 + p_0) \cdot e^{j2\Delta\omega T}) \\ &= \Delta\omega T \end{aligned} \quad (2.20)$$

Der Schätzwert  $\Delta\hat{\omega}T$  ist damit unverzerrt und datenunabhängig.

Dieses ändert sich jedoch, wenn eine Bandbegrenzung eingeführt wird. Abbildung 1.30 zeigt in der dritten Zeile den Verlauf der Differenzphase. Bei einem Datenwechsel von 0 auf 1 oder umgekehrt wird der Wert  $\frac{\pi}{2}$  nicht ganz erreicht. Es müssen daher zwei weitere mögliche Phasenwerte eingeführt werden, die jeweils bei einem Bitwechsel auftreten:

$$\varphi_{01} = \frac{\pi}{2} - \delta + \Delta\omega T, \text{ für } a_{i-1} = 0 \text{ und } a_i = 1 \quad (2.21)$$

$$\varphi_{10} = -\frac{\pi}{2} + \delta + \Delta\omega T, \text{ für } a_{i-1} = 1 \text{ und } a_i = 0. \quad (2.22)$$

$\delta$  hängt dabei von der Bandbreite ab. Mit den Wahrscheinlichkeiten für einen Bitwechsel von 0 auf 1 und von 1 auf 0  $p_{01}$  bzw.  $p_{10}$  ergibt sich für den Schätzwert in Gleichung (2.17):

$$\begin{aligned} \Delta\hat{\omega}T &= \frac{1}{2} \arg\left((p_1 - p_{01}) \cdot e^{j2\Delta\omega T} + (p_0 - p_{10}) \cdot e^{j2\Delta\omega T} \right. \\ &\quad \left. + p_{01} \cdot e^{j(2\Delta\omega T - \delta)} + p_{10} \cdot e^{j(2\Delta\omega T + \delta)}\right) \\ &= \frac{1}{2} \arg\left(e^{j2\Delta\omega T} (1 - p_{01} - p_{10} + p_{01}e^{-j\delta} + p_{10}e^{j\delta})\right) \\ &= \Delta\omega T + \underbrace{\frac{1}{2} \arg\left(1 + p_{01}(e^{-j\delta} - 1) + p_{10}(e^{j\delta} - 1)\right)}_E \end{aligned} \quad (2.23)$$

Es existieren Fälle, in denen der Fehlerterm  $E$  in Gleichung (2.23) zu Null wird. Für  $\delta = 0$  führt Gleichung (2.23) wieder zum Fall ohne Bandbegrenzung. Von größerer Bedeutung ist der Fall  $p = p_{01} = p_{10}$ . Dann gilt:

$$\begin{aligned} E &= \frac{1}{2} \arg(1 + 2p(\cos \delta - 1)) \\ &= 0 \end{aligned} \quad (2.24)$$

Es mag zunächst etwas merkwürdig erscheinen, wenn zuerst von zwei verschiedenen Wahrscheinlichkeiten  $p_{01}$  und  $p_{10}$  ausgegangen wird, da ein Bit das einmal auf 1 gegangen ist, erst wieder auf 0 gehen muß, bevor es wieder auf 1 zurückgehen kann. Bei einer Erwartungswertbildung über unendlich viele Symbole ist dies sicher richtig. Eine spätere Implementation mittelt jedoch immer nur über ein begrenztes Intervall, so daß ein Wechsel mehr in eine Richtung auftreten kann. Da die Wahrscheinlichkeiten  $p_{01}$  und  $p_{10}$  über den Beobachtungszeitraum mit der Länge  $L$  Symbole ermittelt werden müssen, ergibt sich hier durchaus ein Unterschied. Für den Fall, daß  $n$  0-1 Wechsel ( $2n + 1 < L$ ) und  $n + 1$  1-0 Wechsel stattgefunden haben, folgt:

$$p_{01} = \frac{n}{L} \quad (2.25)$$

$$p_{10} = \frac{n+1}{L} \quad (2.26)$$

$$(2.27)$$

Es folgt weiter:

$$\begin{aligned} E &= \frac{1}{2} \arctan \left( \frac{\frac{n}{L} \sin(-\delta) + \frac{n+1}{L} \sin \delta}{1 + \frac{2n+1}{L} (\cos \delta - 1)} \right) \\ &= \frac{1}{2} \arctan \left( \frac{\sin \delta}{L + (2n+1) \cdot (\cos \delta - 1)} \right) \end{aligned} \quad (2.28)$$

Gleichung (2.28) wird bei festem  $L$  und  $\delta$  maximal für  $L = -(2n+1) \cdot (\cos \delta - 1) \Rightarrow$

$$n = \frac{L/2}{1 - \cos \delta} - \frac{1}{2}. \quad (2.29)$$

Für den Fall  $L = 64$  und ein bei der Bandbreite  $f_{\Delta} = 1.5/T$  typisches  $\delta = 0.2$  ergibt sich  $n \approx 1605$ . Das heißt, das Maximum liegt weit außerhalb des zulässigen Bereichs für  $n$ . Gleichung (2.28) ist in dem interessierenden Bereich für  $n \in [0 \dots 30]$  monoton steigend, so daß der maximale Fehler  $E$  abgeschätzt werden kann durch Einsetzen von  $n = 30$ . Damit folgt  $E_{\max} = 1.58 \cdot 10^{-3}$ . Dieser Fehler ist so gering, daß er vernachlässigt werden kann. Damit wird die Frequenzoffsetschätzung in Bezug zu der hier betrachteten Genauigkeit dateunabhängig. Dieses Ergebnis ist für eine Implementierung sehr wichtig, da es dadurch erlaubt wird, die Frequenzoffsetschätzung für jedes Präambelsymbol getrennt auszuführen.

### 2.3.1.3 modifizierte Frequenzoffsetschätzung für kleine $\Delta f$

Falls bei dem Einsatz auf der Stecke nur mit kleineren Frequenzoffsets (Größenordnung 10%) gerechnet werden muß, oder falls der Restfehler der groben Frequenzoffsetkorrektur ausreichend klein ist, läßt sich eine sehr einfache Methode zur Frequenzoffsetschätzung gewinnen. Sie ist eng an Zahlendarstellung und Wertebereich im Zweierkomplement angelehnt.

Die Werte der doppelten Differenzphase schwanken im Bereich zwischen  $-\pi$  und  $\pi$ . Dies entspricht im Zweierkomplement den Werten  $-2^{n-1}$  und  $2^{n-1}-1$ . Durch einfache Uminterpretation der Zweierkomplementdarstellung als vorzeichenlose Binärdarstellung wird dieses Intervall auf  $[0 \dots 2^n - 1]$  abgebildet. Dies entspricht dem Bereich  $0 \dots 2\pi$ . Negative Differenzphasenwerte werden daher auf  $\pi + \varepsilon$ , positive auf  $\pi - \varepsilon$  abgebildet.  $\varepsilon$  wird durch das Rauschen und die Bandbreite bestimmt. Mit Frequenzoffset verschieben sich diese Werte vom Mittelwert  $\pi$  zu  $\pi + 2\Delta\omega T$ . Das heißt, durch einfache Subtraktion von  $\pi$  erhält man nach Mittelung einen Schätzwert für  $\Delta\omega$ . Die Varianz dieses Verfahrens steht dem oben beschriebenen in nichts nach. Nachteilig wirkt sich jedoch aus, daß die eindimensionale Struktur der Phase nicht gleichzeitig zyklisch und ohne Sprung sein kann. Für Frequenzoffsets, deren betragsmäßige Größe daher zusammen mit dem Fehler  $\varepsilon$  den Wert  $\pi$  überschreiten, erfolgt eine Verschiebung um  $2\pi$ . Ist beispielsweise  $2\Delta\omega T = \frac{\pi}{2}$  und  $\varepsilon = \frac{\pi}{2} + \varepsilon_\varepsilon$  so ergibt sich in der vorzeichenlosen Darstellung der Wert  $2\pi - \varepsilon_\varepsilon$  bzw. der Wert  $0 + \varepsilon_\varepsilon$ , was für die Mittelung den größtmöglichen Fehler darstellt. Bei der Mittelung in kartesischen Koordinaten entsprechend dem oben beschriebenen Schätzer erfolgt hier kein plötzlicher Sprung. Die Filterung im kartesischen Bereich ist daher fehlertoleranter und erlaubt auch die Schätzung von größeren Frequenzoffsets. Falls jedoch die zu Beginn dieses Abschnitts beschriebenen Bedingungen erfüllt sind, ist das in diesem Abschnitt beschriebene Verfahren eine einfache Alternative zu der Filterung im kartesischen Bereich. Dies gewinnt dann besonders an Gewicht, wenn auf die Taktoffsetschätzung mittels Filterung im kartesischen Bereich verzichtet werden kann, da sie auf andere Weise erfolgt. Ein mögliches anderes Verfahren zur Taktoffsetschätzung stellt die Ausnutzung der Korrelationswerte über das Synchronisationswort der Präambel dar. Es muß beachtet werden, daß selbst für die Funktion der Frequenzoffsetschätzung die Kenntnis des Taktoffsets notwendig ist.

## 2.3.2 Implementationsaspekte

### 2.3.2.1 Wortbreiten im Schätzer

Der Schätzer läßt sich in drei Funktionsblöcke teilen:

1. P→R-Wandlung
2. Mittelwertbildung
3. R→P-Wandlung und Betragsmaximumsuche

Die P→R-Wandlung bekommt als Eingangswert die Differenzphase, deren Auflösung bei der Simulation in Abbildung 2.7 mit  $L = 64$  im Bereich 5-7 Bit variiert wird. Von weiterem Interesse ist die Auflösung, die in kartesischen Koordinaten für den Schätzer notwendig ist. Sie wird im Bereich von 2 bis 6 Bits variiert. Abbildung 2.7 zeigt, daß eine Phasenauflösung von mindestens 6 Bit und eine kartesische Auflösung von ca. 4 Bit notwendig ist, um keine starken Performanceeinbußen zu erhalten. Die 6 Bits Phasenauflösung für P→R-Wandlung

mögen etwas knapp erscheinen. Man muß sich jedoch vor Augen halten, daß eine Erweiterung des Wertebereichs um ein Bit eine doppelt so große Wandlungstabelle erfordert.

Die Mittelwertbildung verarbeitet die Werte von der P→R-Wandlung und reicht sie an die R→P-Wandlung und die Betragsmaximumsuche weiter. Bei der Simulation mit  $L = 64$ , die in Abbildung 2.8 zu sehen ist, wurden die LSBs nach der Mittelwertbildung geclippt. Dabei war die Auflösung nach der P→R-Wandlung auf 6 Bit eingestellt. Es ergibt sich daher bei einer Schätzerlänge von  $L = 64$  eine theoretische Auflösung von 12 Bits. Abbildung 2.8 zeigt, daß für eine gute Taktoffsetschätzung eine Auflösung von mindestens 7 Bits erforderlich ist. Diese Auflösung wird jedoch in beiden Quadraturkomponenten benötigt, so daß eine Tabelle mit  $2^{14} = 16384$  Einträgen realisiert werden müßte. Die Frequenzoffsetschätzung ist etwas weniger anspruchsvoll, was die Auflösung angeht, und kommt mit 6 Bits aus. Die BER zeigt bis zu 5 Bits keine wesentliche Verschlechterung. Dies liegt daran, daß der Schätzer bei hoher Auflösung und moderate SNRs so präzise Taktoffset-Werte liefert, daß für die BER die eigentliche Datendetektion als Fehlerquelle überwiegt. Die Fehlerrate des Dezimators für  $\varepsilon = 3$  ist zunächst 0.5, da der Taktoffsetschätzwert zwischen 1 und 2 wechselt und mit  $\lceil \varepsilon/2 \rceil$  verglichen wird. Für geringere Auflösungen nach der Summation nimmt die Wahrscheinlichkeit, daß bei der Maximumssuche zwei gleiche Werte auftauchen zu. Die hier benutzte Maximumssuche wählt in diesem Fall den ersten Wert. Dadurch kommt es zu einer scheinbaren Verbesserung der Taktoffset-Schätzfehlerrate bei  $\varepsilon = 3$  für kleinere Auflösungen nach der Summation.

[ Bei 5 Bit Auflösung beträgt die Anzahl der Einträge in der Tabelle nur noch  $2^{10} = 1024$  Werte. Im späteren Einsatz ist Blockbetrieb vorgesehen. Dabei werden die Schätzwerte für Takt- und Frequenzoffset für die Dauer eines Datenblocks konstant gehalten. ]

Eine Untersuchung des tatsächlich genutzten Wertebereichs nach der Summation zeigte, daß trotz eines starken Rauschens auf der Phase mit den daraus im kartesischen Bereich folgenden Schwankungen der Wertebereich zu ungefähr 90% genutzt wird. Das heißt, daß ein Clipping des MSBs nicht möglich ist.

Es bleibt die notwendige Auflösung nach der R→P-Wandlung zu bestimmen. Hier muß eine getrennte Untersuchung für Betrag und Phase erfolgen. Die Phase liefert den Ausgangswert für die Frequenzoffsetkompensation. Es ist daher nur eine Auflösung sinnvoll, die auch die Differenzphase selbst besitzt. Der Betrag kann einerseits über die Tabelle, aus der auch die Phase entnommen wird, oder andererseits durch explizites Quadrieren und Summieren der kartesischen Komponenten bestimmt werden. In letzterem Fall kann bei wertkontinuierlicher Betrachtung das Wurzelziehen aufgrund der Monotonie der Wurzelfunktion entfallen. Hier sind jedoch die Werte quantisiert, so daß das Wurzelziehen einer Verzerrung der Quantisierungsabstände gleichkommt. Es wurde daher mit Hilfe einer Simulation, deren Ergebnis in Abbildung 2.9 mit  $L = 42$  dargestellt ist, überprüft, welche Darstellung mit weniger Quantisierungsstufen auskommt. Es ist zu erkennen, daß die Maximumssuche mit Hilfe des Betrags nur 8 Bits benötigt. Im Gegensatz dazu sind bei Maximumssuche mit Hilfe des Betragsquadrats mindestens 9, besser 10 Bits erforderlich. Da das Ziehen der Wurzel

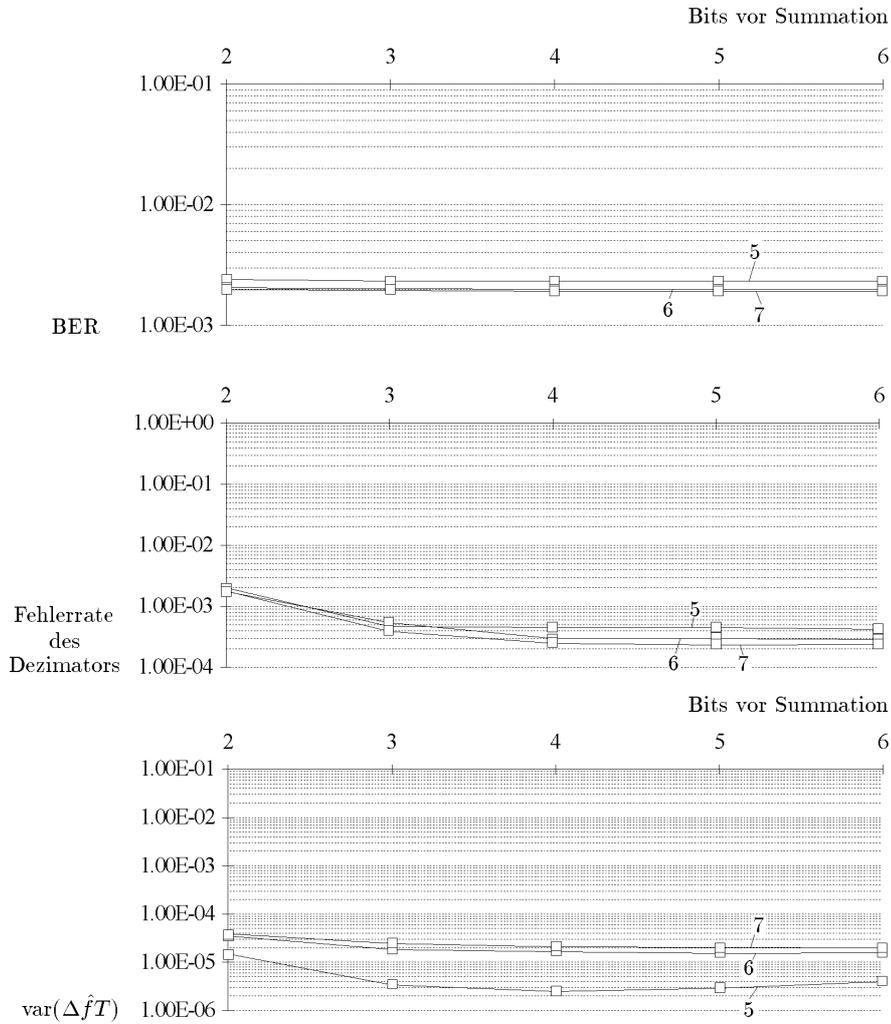


Abbildung 2.7: Bitfehleraten, Dezimationsfehleraten und Varianz der Frequenzoffsetschätzung in Abhängigkeit von der Auflösung vor der Summationsmittelung; Parameter Phasenbreite (5-7 Bit)

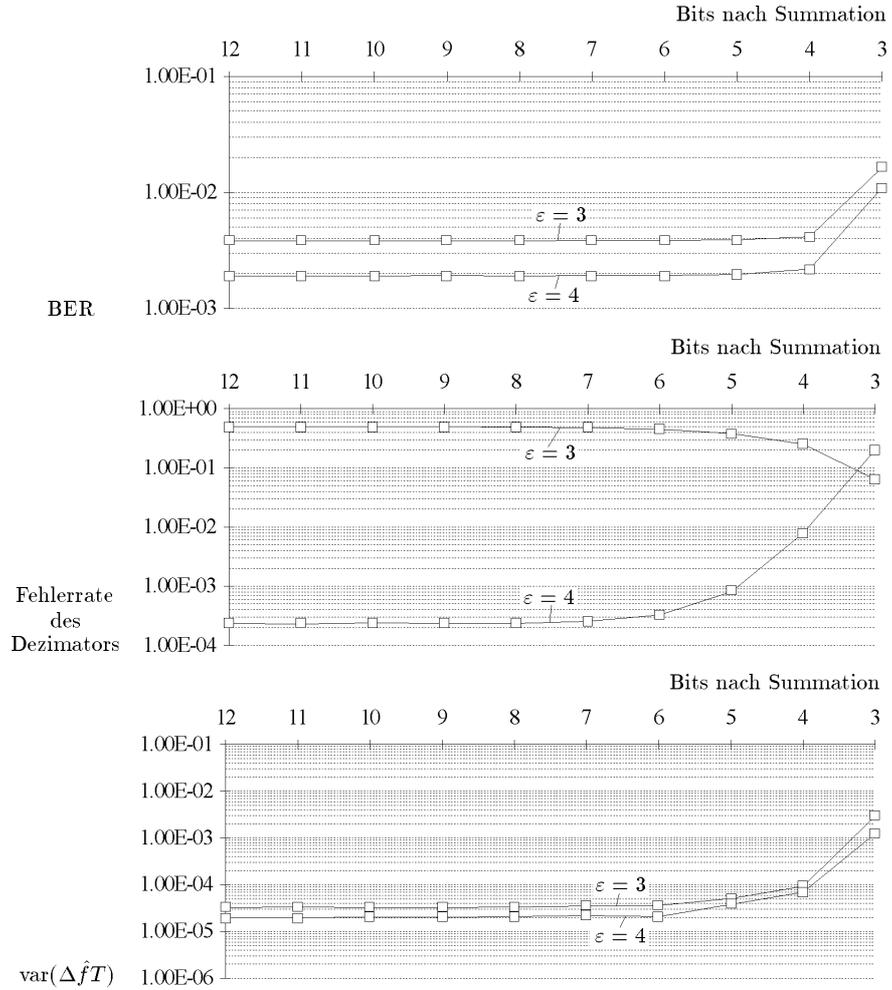


Abbildung 2.8: Bitfehlerraten, Dezimatorsfehlerraten und Varianz der Frequenzoffsetschätzung in Abhängigkeit von der Auflösung nach der Summationsmittelung, vor R→P-Wandlung

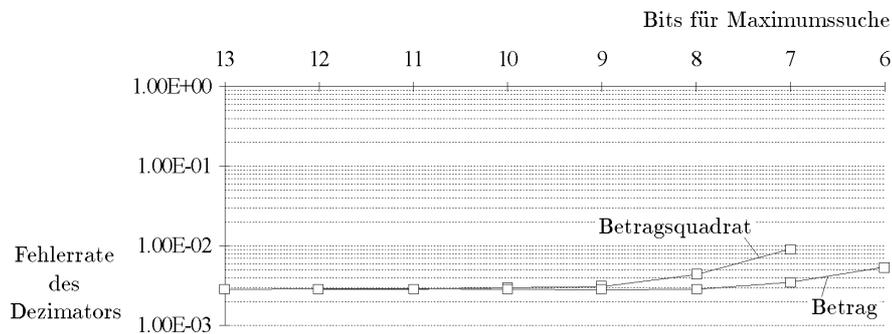


Abbildung 2.9: Dezimationsfehllerraten in Abhängigkeit von der Auflösung vor der Maximumssuche

auf dem Chip unpraktikabel ist, muß der Lösung über eine Tabelle der Vorzug gegeben werden.

### 2.3.2.2 Architekturen

Der Frequenzoffset- und Taktfehlerschätzer wird einen Großteil der späteren Chipfläche einnehmen und bedarf deswegen einer sorgfältigen Optimierung. Der Schätzer beinhaltet verschiedene Funktionsgruppen, die je nach gewählter Architektur in verschiedenen Häufigkeiten und Zusammenstellungen auftreten. Im einzelnen sind dies:

1. P→R-Wandlung (PRW)
2. R→P-Wandlung (RPW)
3. Speicherbausteine (RAM)
4. Seriell-Parallel-Wandler (SPW)
5. Parallel-Seriell-Wandler (PSW)
6. Vergleichseinheit zur Maximumssuche
7. Multiplexer (MUX)
8. Demultiplexer (DeMUX)
9. Addierer / Subtrahierer (ADD, SUB)

Eine direkte Implementierung der Erwartungswertbildung ist unökonomisch, da zum großen Teil immer wieder über dieselben Werte summiert wird. Es ist geschickter, die jeweils neuen Abtastwerte in einer Speicherzelle aufzusummieren und Abtastwerte, die außerhalb des Schätzerfensters liegen, zu subtrahieren.

Dadurch ist nur noch eine Addition und Subtraktion pro Sample-Takt notwendig. Eine daraus direkt folgende Implementierungsmöglichkeit zeigt Abbildung 2.10 (Modell 1). Sämtliche Speicherzellen werden beim Reset mit 0 initialisiert. Die Implementation von Modell 1 benötigt:

- 1 P→R-Wandlung
- 8 R→P-Wandlungen
- $4 \cdot 2 \cdot L + 4 \cdot 2$  Speicherzellen
- 2 Seriell-Parallel-Wandler
- 1 Vergleichseinheit zur Maximumsuche
- 2 4 zu 1 Multiplexer
- $2 \cdot 4 + 1$  Addierer und  $2 \cdot 4$  Subtrahierer

Modell 1 läßt die prinzipielle Funktion des Schätzers noch gut erkennen, besitzt jedoch den Nachteil, daß es sehr viele R→P-Wandlungen benötigt und einen sehr hohen Speicheraufwand besitzt.

Die benötigte Wortbreite bei der Darstellung in rechtwinkligen Koordinaten beträgt je Quadraturkomponente ungefähr 6 Bit. Für die Darstellung der Phase reichen dagegen ca. 7 Bit. Dies reduziert den Aufwand für die Speicher fast um den Faktor 2. Durch die Speicherung der Phase werden auch die P→R-Wandlung zu Beginn und die vier R→P-Wandlungen vor dem Detektionsmultiplexer überflüssig. Stattdessen werden nun für die Summation der Quadraturkomponenten P→R-Wandlungen notwendig. Einen weiteren Optimierungsansatz bieten die R→P-Wandler in Abbildung 2.10, da sie nur mit der Symbolrate betrieben werden. Durch geschickte Platzierung von Seriell-Parallel- und Parallel-Seriell-Wandlern lassen sie sich durch einen ersetzen. Abbildung 2.11 zeigt das so umgeformte System.

Die Implementation von Modell 2 benötigt:

- 2 P→R-Wandlungen
- 1 R→P-Wandlung
- $4 \cdot L + 4 \cdot 2$  Speicherzellen
- 7 Seriell-Parallel-Wandler
- 3 Parallel-Seriell-Wandler
- 1 Vergleichseinheit zur Maximumsuche
- 2 4 zu 1 Multiplexer
- $2 \cdot 4 + 1$  Addierer und  $2 \cdot 4$  Subtrahierer

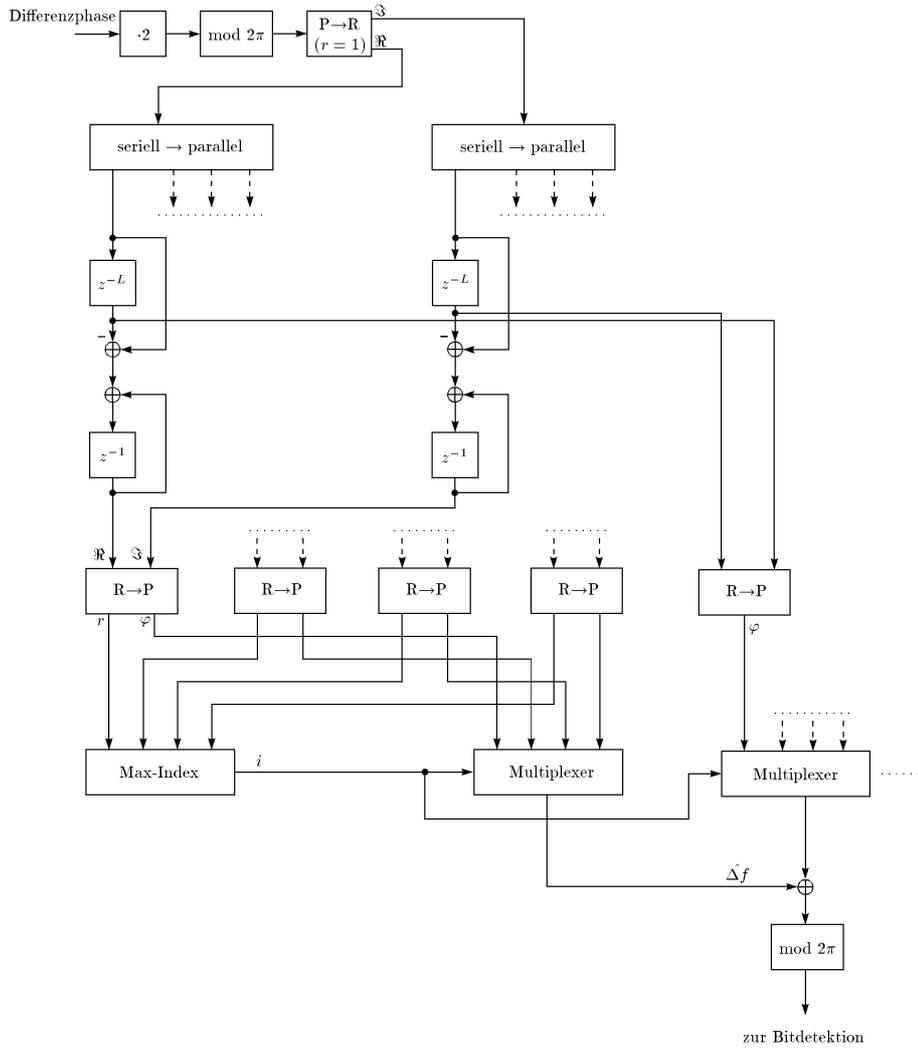


Abbildung 2.10: Schätzerimplementierung, Modell 1



Die Anzahl der Koordinatentransformationen kann durch diese Architektur stark gesenkt werden. Dies geht allerdings zu Lasten der Anzahl der notwendigen Seriell-Parallel- und Parallel-Seriell-Wandlungen, die jedoch nicht alle unabhängig voneinander laufen müssen. Sämtliche Umwandlungen können durch ein und den selben Takt der vierfachen Symbolrate durchgeführt werden.

## 2.4 Rahmensynchronisation und Präambel-detektion

Die Übertragung der späteren Nutzdaten erfolgt in Blöcken. Zur Kennzeichnung eines Blockanfangs wird eine spezielle Präambel gesendet, die dem Empfänger bekannt ist. Sie besteht aus drei Teilen:

1. Ein Block von mehreren Datenbits, die 1 sind. Sie dienen zur groben Frequenzoffsetkorrektur.
2. Es schließt sich ein Block mit optimierten Korrelationseigenschaften an. Die Autokorrelationsfunktion dieses Blocks besitzt ein ausgeprägtes Maximum und hebt sich deutlich von dem vorangegangenen Block von Einsen ab.
3. Abschließend folgen einige Bits, die speziell so gewählt sind, daß das Rauschen des Schätzers möglichst klein wird.

Ist die Präambel einmal erkannt, werden die Schätzwerte, mit denen die Präambeldetektion erfolgt ist, eingefroren, um den sich anschließenden Datenblock mit den optimalen Werten zu detektieren.

### 2.4.1 Architekturen

#### 2.4.1.1 komplette Präambeldetektion mit festen Schätzwerten

Da der Schätzer während der Präambeldetektion bei ungünstigen Taktoffsets zu einer doppelt-Detektion einzelner Datenbits führt, muß der Taktoffsetschätzwert für die gesamte Präambel konstant sein. In den Untersuchungen vor dieser Arbeit wurde zusätzlich noch der Frequenzoffset mit eingefroren. Die Detektion der gesamten Präambel mit ein und denselben Schätzwerten für Frequenz- und Taktoffset erfordert einen hohen implementatorischen Aufwand, da mit jedem Symboltakt der gesamte Phasenspeicher ausgelesen, um den Frequenzoffset bereinigt und mit dem Synchronisationswort korreliert werden muß. Dieses würde neben den Koordinatentransformationen einen Großteil der Chipfläche beanspruchen. Zugleich würden außerordentlich hohe Anforderungen an die Verarbeitungsgeschwindigkeit gestellt. Es läßt sich jedoch eine starke Vereinfachung des Algorithmus vornehmen.

### 2.4.1.2 stark vereinfachtes Verfahren

Im vorigen Abschnitt wurde davon ausgegangen, daß die Präambeldetektion mit einem einzigen Frequenz- und Taktoffsetschätzwert durchgeführt werden muß. Der Abschnitt 2.3.1.2 hat gezeigt, daß die Frequenzoffsetschätzung unabhängig von den Daten arbeitet. Daher kann die Korrektur des Frequenzoffsets für jedes Datenbit der Präambel getrennt erfolgen. Dieses hat weitreichende architekturvereinfachende Folgen. Es bleibt der Taktoffset als Schätzwert übrig, der für sämtliche Bits bei der Präambeldetektion gleich sein muß. Da dieser Schätzwert jedoch nur 4 Zustände hat, kann eine parallele Detektion der Bits für die Präambeldetektion zu den 4 Abtastzeitpunkten erfolgen. Es kann daher auf ein Auslesen des gesamten Phasenspeicherblocks innerhalb eines Symboltaktes verzichtet werden. Stattdessen sind nun 4 getrennte Bitkorrelationen notwendig. Der Aufwand hierfür ist jedoch, verglichen mit dem Auslesen des kompletten Phasenspeichers, um ein Vielfaches geringer. Abbildung 2.12 zeigt das Zusammenspiel von Schätzer und Präambeldetektion bei diesem Verfahren.

[

## 2.4.2 Präambeldetektion ohne Taktoffsetschätzwert

Der Aufwand für den Schätzer mit R→P- und P→R-Wandlung und Quadratur mit Maximumsuche ist beträchtlich. Es liegt daher nahe einen Versuch zu unternehmen, den Schätzer zu ersetzen. Ausgangspunkt dafür ist die Präambeldetektion. Wenn zu den vier möglichen Abtastzeitpunkten parallel eine Datendetektion mit anschließender Korrelation über das Korrelationswort ausgeführt wird, sollte sich für den richtigen Abtastzeitpunkt ein Korrelationsmaximum ergeben.

### 2.4.2.1 Hard-Decision vor Korrelation

Abbildung 2.13 zeigt die Bitfehlerraten für feste Dezimationszeitpunkte bei einem Taktoffset von  $\varepsilon = 4$  bzw.  $\varepsilon = 3$  und verschiedenen SNR's. Für  $\varepsilon = 4$  hebt sich deutlich das Bitfehlerratenminimum ab. Für  $\varepsilon = 3$  treten zwei deutliche Minima auf. Die Nebenminima bei  $\varepsilon = 4$  sind jedoch so niedrig, daß in einem Korrelationswort mit einer angestrebten Länge von 32 noch kein Fehler auftritt. Es werden daher in den Korrelationswerten 3 gleich hohe Peaks auftreten. Eine Selektion des mittleren scheidet als Lösung aus, da die Fehlerraten noch so groß sind, daß ab und zu nur 2 Peaks auftreten und dadurch der falsche Dezimationspunkt gewählt würde. Es gibt jedoch eine sehr einfache andere Möglichkeit. Man sucht nach dem Korrelationsminimum in einem Block und schließt von da aus auf den am weitesten entfernten Dezimationspunkt, der zwei Takte entfernt liegt. Dieses Verfahren ist sehr sicher und funktioniert für gerade und ungerade  $\varepsilon$  gleichermaßen. Es bleibt jedoch eine Schwierigkeit: Die Bestimmung des richtigen 4er-Blocks, innerhalb dessen das Minimum gesucht werden soll.



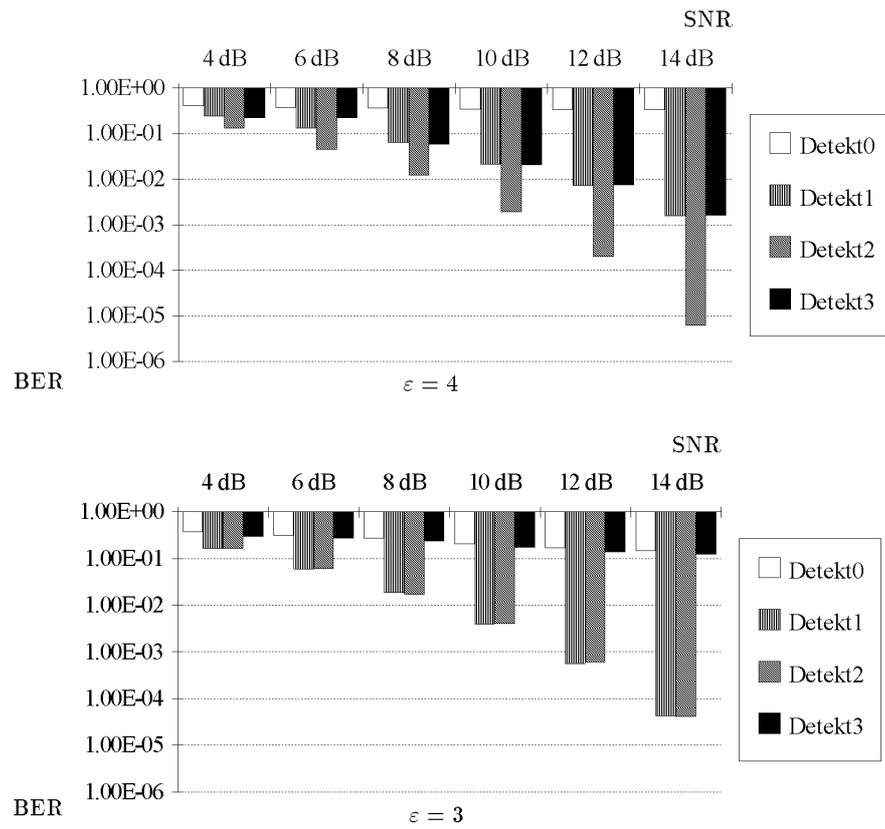


Abbildung 2.13: Bitfehlerraten bei festen Dezimationspunkten

$\varepsilon$	0	1	2	3	0	1	2	3
0	0	0	0	1	1	1	0	0
1	0	0	0	0	1	1	0	0
2	0	0	0	0	1	1	1	0
3	0	0	0	0	0	1	1	0
4	0	0	0	0	0	1	1	1
5	0	0	0	0	0	0	1	1
6	0	0	1	1	1	0	0	0
7	0	0	0	1	1	0	0	0

Tabelle 2.1:

$\varepsilon$	0	1	2	3	0	1	2	3
0	0	0	0	1	1	1	0	0
1	0	0	0	1	1	1	1	0
2	0	0	0	0	1	1	1	0
3	0	0	0	0	1	1	1	1
4	0	0	0	0	0	1	1	1
5	0	1	1	1	1	0	0	0
6	0	0	1	1	1	0	0	0
7	0	0	1	1	1	1	0	0

Tabelle 2.2:

#### 2.4.2.2 Soft-Decision vor Korrelation

#### 2.4.2.3 Korrelation mit Differenzphasenverlauf

]

### 2.4.3 Optimierung der Präambel

Die Präambel dient der Erkennung des Beginns eines Datenblocks. Zusätzlich werden die Schätzwerte für die Datendetektion benutzt, die der Schätzer geliefert hat, während das Schätzfenster über der Präambel lag. Die Performance des Schätzers ist, wie in Abschnitt 2.3.1.1 beschrieben, datenabhängig. Es liegt daher nahe, die Bitfolge der Präambel so zu optimieren, daß der Schätzer besonders gute Werte liefert [4].

#### 2.4.3.1 Korrelationseigenschaften

Die Bitfolge der Präambel ist dem Empfänger bekannt. Er kann daher in den empfangenen Bits durch Korrelation nach der Präambel suchen. Um eine sichere Erkennung der Präambel zu gewährleisten, müssen einzelne Bitfehler zugelas-

sen werden. Die Präambel gilt daher als erkannt, wenn  $32 - n_{\text{Err}}$ -Bits richtig übertragen wurden. Die genaue Bitfolge der hier gewählten Präambel lautet: 0011 0000 1100 1001 1110 1000 1010 1101 0101 0101 010. Die Präambel ist insgesamt 43 Bit lang. Davon dienen jedoch nur die ersten 32 der Korrelation. Die restlichen 11 dienen der Optimierung der Schätzerperformance. Eine Ausdehnung der für die Korrelation relevanten Bits auf die vollen 43 Symbole ist unterblieben, da das alternierende Bitmuster der letzten 11 Bits eine besonders hohe Fehlerrate besitzt. Es wäre daher eine Tolerierung zusätzlicher falscher Bits in der Präambel erforderlich geworden, die den Gewinn durch eine Verlängerung der Korrelation überkompensieren würde.

### 2.4.3.2 Schätzerperformance

(2.16) beschreibt den Verlauf des Schätzerfilterausgangs. Nach Abtastung gilt für das Signal ohne Bandbegrenzung und Rauschen mit  $\varepsilon = 0$ :

$$A_k = \left| p + (1 - p) \cdot \cos \frac{2\pi k}{N} \right| \quad (2.30)$$

Das Maximum der 4 Abtastwerte gibt den Taktoffset an. Entscheidend für die Sicherheit der Maximumerkennung ist der Abstand zwischen dem richtigen Maximum und dem zweitgrößten Wert. Dieser Abstand sei mit  $\Delta A$  bezeichnet. Es ergeben sich für die 4 Abtastwerte von (2.30) die Werte in Tabelle 2.3.

$k$	$A_k$
0	1
1	$p$
2	$1 - 2p$
3	$p$

Tabelle 2.3: Erwartungswert des Schätzfilters bei  $\varepsilon = 0$

$A_k$  muß für  $k \in [1 \dots 3]$  möglichst klein sein. Da die Minimierung für nur ein  $k$  keinen Sinn macht, muß gelten:

$$A_1 = A_2 = A_3 \quad (2.31)$$

Die Optimierung beschränkt sich also darauf, den Abstand unabhängig von  $k$  zu machen. Damit folgt sofort:

$$\begin{aligned} p &= 2p - 1 \\ \implies \\ p_{\text{opt}} &= \frac{1}{3} \end{aligned} \quad (2.32)$$

Ein Vergleich zu einem  $p$  von zum Beispiel 0.5 zeigt, daß sich der Abstand zwischen Maximum und zweitgrößten Wert für  $k = 1$  und  $k = 3$  vergrößert

hat. Im Gegenzug ist jedoch der Abstand für  $k = 2$  geringer geworden. Die Wahrscheinlichkeit für einen Taktschätzfehler der Größe 2 nimmt daher durch die Wahl von  $p = p_{\text{opt}}$  zu. Nimmt man an, daß alle  $A_k$  durch das Rauschen gleich gestört werden, so ergibt sich eine mittlere Abstandserhöhung. Sie bewirkt eine größere Störsicherheit der Taktoffsetschätzung.

Die Wirksamkeit der Erhöhung des Störabstandes läßt sich an Simulationen überprüfen. Dazu wurde die aus dem vorigen Abschnitt bekannte 43 Bit lange Präambel mit nachfolgenden 57 weißen Datenbits übertragen. Für die Datenbits gilt  $p = 0.5$ . Die Präambel erfüllt dagegen die Bedingung  $p = p_{\text{opt}}$ . Die letzten 11 Bits der Präambel dienen dazu, diese Bedingung zu gewährleisten und stellen daher eine alternierende Bitfolge dar. Simuliert wurde zunächst mit einem sehr einfachen Modell. Es enthält den MSK-Modulator, die Taktverzögerung, den Frequenzoffset, additives gaußsches weißes Rauschen, einen Roll-Off-Tiefpaß mit  $\beta = 0.1$  und schließlich den Schätzer. Als Parameter wurden eingestellt:

1. SNR = 8dB
2.  $\Delta fT = 0$ , bzw. 0.1875
3.  $\varepsilon = 4$
4. Phasenoffset = 0.8
5. L = 42

Das Fenster, innerhalb dessen der Schätzer arbeitet, gleitet kontinuierlich über die Symbole. Man kann daher sehr anschaulich die Wirkung einer Veränderung von  $p$  verfolgen, wenn, nach ein- und zweifachen Dezimationsfehlern getrennt, die Fehlerrate über die 100 Symbole aufgetragen wird. Dies ist in Abbildung 2.14 geschehen. Die unteren, schwarz gefüllten Balken stehen für die einfach-Taktschätzfehler. Auf diese sind die nicht gefüllten Balken aufgesetzt, deren Höhe die zweifach-Taktschätzfehlerrate wiedergibt. Auf diese Weise gibt die Gesamthöhe eines Balkens die Gesamtdezimationsfehlerrate an. Abbildung 2.15 enthält den Verlauf von  $1 - p$  über die Symbole verteilt. Der Zusammenhang zwischen  $1 - p$  und den ein- bzw. zweifach-Taktschätzfehlern ist gut zu erkennen. Im Nullpunkt liegt der Schätzer genau über den 43 Präambelbits. Daher weist  $1 - p$  den Wert  $\frac{2}{3}$  auf und die Fehlerrate ist minimal. Man erkennt jedoch auch, daß in diesem Punkt einfach- und zweifach-Fehler nicht im Verhältnis 2:1 vorkommen, wie es nach der vorhergehenden Betrachtung anzunehmen wäre. Der Grund hierfür ist in der Bandbegrenzung zu sehen. Im Bereich 43–57 auf der Abszisse liegt der Schätzer über weißen Daten. Es ein deutlicher Anstieg der einzel-Fehlerrate zu beobachten, die auch die Gesamtfehlerrate deutlich erhöht. Im Bereich um 20 treten dagegen sehr hohe zweifach-Fehlerraten auf, da das Schätzerfenster in diesem Bereich gerade über den letzten 11 alternierenden Bits der Präambel liegt. Ein vorhandener Frequenzoffset bewirkt eine Verstärkung des letzten Punktes.

Wird die Simulationsstrecke durch Einbau des Empfangsfilters  $H(z)$  im Empfänger erweitert, treten in der Fehlerverteilung Änderungen auf. Abbildung

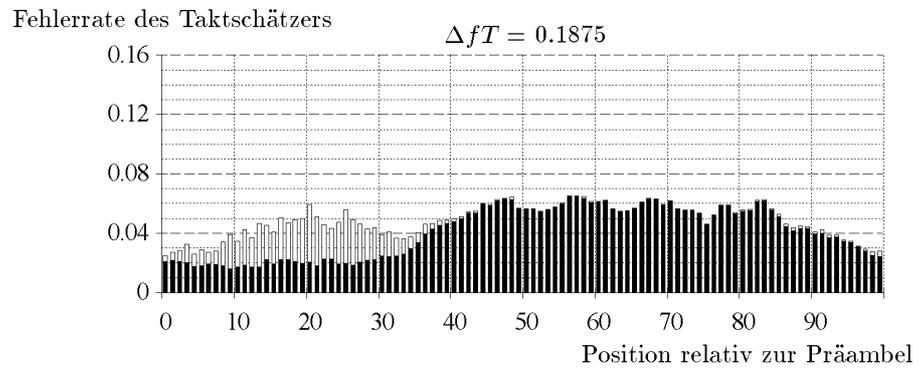
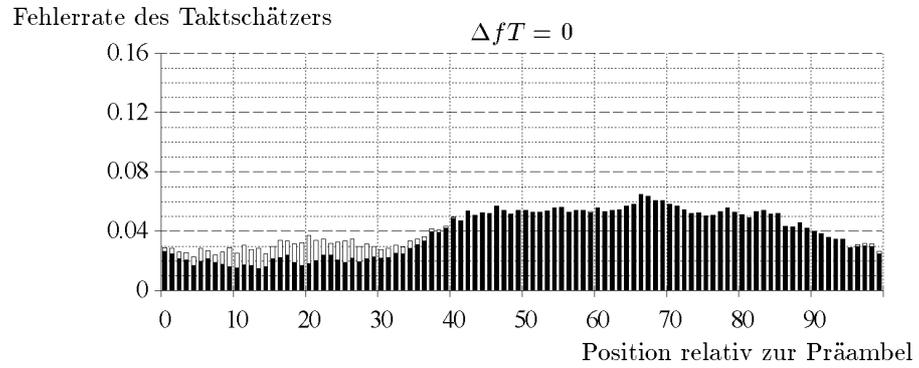


Abbildung 2.14: Taktoffset-Schätzfehlerraten nach einfach- und zweifach-Fehlern getrennt aufgetragen über den Versatz zur Präambel, einfaches Simulationsmodell

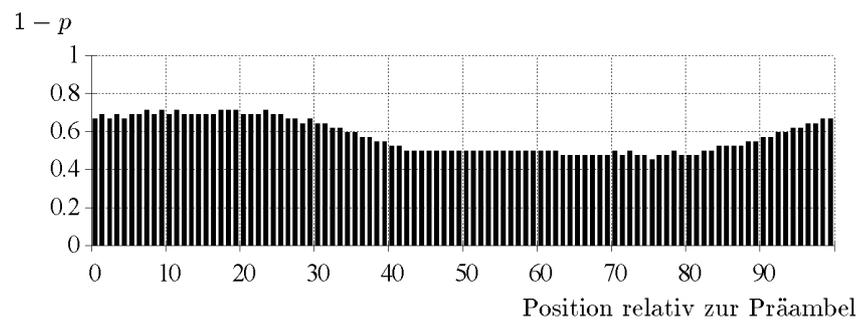


Abbildung 2.15:  $1 - p$  aufgetragen über den Versatz zur Präambel

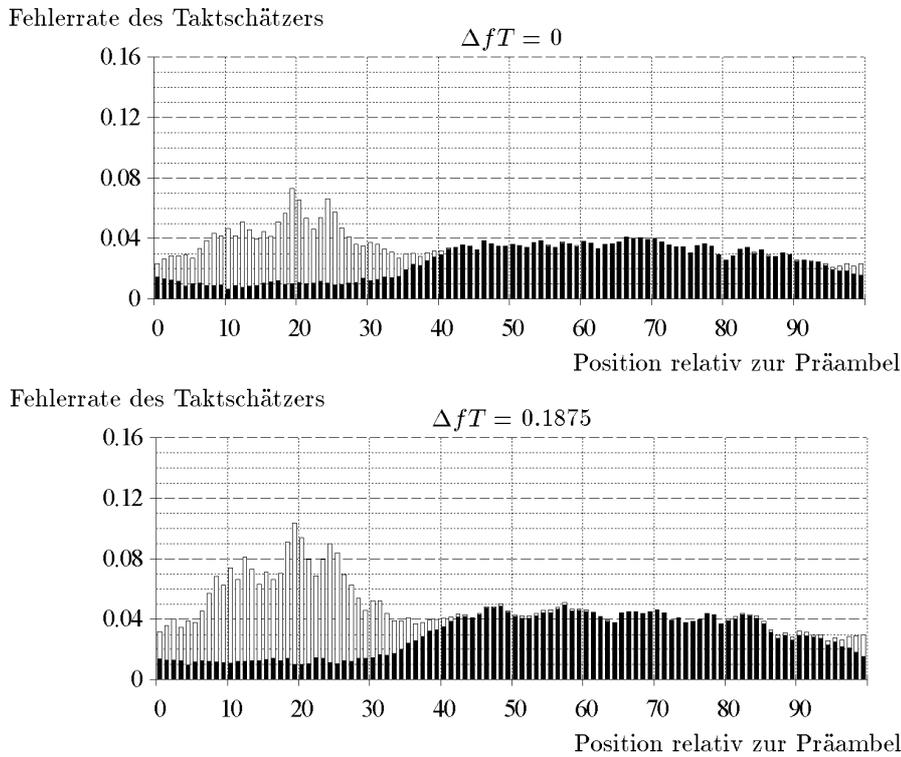


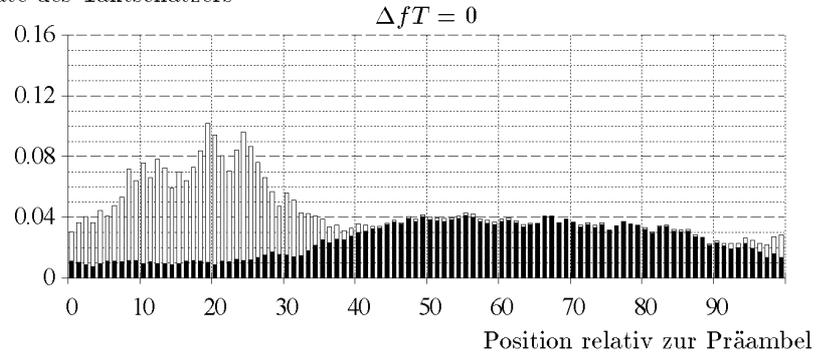
Abbildung 2.16: Taktoffset-Schätzfehlerraten nach einfach- und zweifach-Fehlern getrennt aufgetragen über den Versatz zur Präambel, Simulationsmodell mit  $H(z)$

2.16 zeigt diese Simulationsergebnisse, die ansonsten mit gleichen Parametern gewonnen wurden.

Drei Merkmale fallen besonders auf:

1. Die Fehlerrate im Bereich der weißen Daten von 43–57 auf der Abszisse hat abgenommen.
2. Es treten im Bereich um 20 deutlich höhere zweifach-Taktschätzfehler auf. Dafür ist die einfach-Fehlerrate in diesem Bereich gesunken.
3. Auch um den Nullpunkt ist eine Zunahme der zweifach- und eine Abnahme der einfach-Fehlerraten zu verzeichnen. Dieses legt den Schluß nahe, daß das Filter  $H(z)$  von sich aus die Wirkung hat, als ob  $p$  verkleinert würde. Durch den starken Anstieg der zweifach-Fehlerraten liegt das Minimum nun nicht mehr bei 0, sondern davor, da  $p$  zu diesem Zeitpunkt noch kleinere Werte besitzt.

Fehlerrate des Taktschätzers



Fehlerrate des Taktschätzers

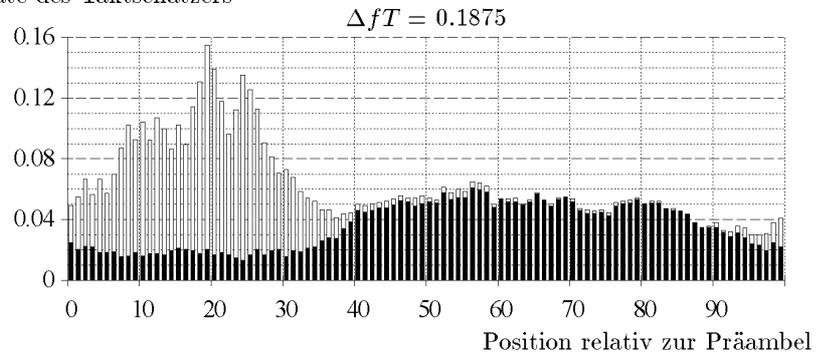


Abbildung 2.17: Taktoffset-Schätzfehlerraten nach einfach- und zweifach-Fehlern getrennt aufgetragen über den Versatz zur Präambel, vollständiges Simulationsmodell

Bei den Punkten 2. und 3. bewirkt der Frequenzoffset wieder eine Zunahme der zweifach-Fehler.

Als nächstes wurde eine Simulation mit der kompletten Strecke durchgeführt. Das heißt mit Mischung und ILO. Wieder mit den gleichen Parametern wie zuvor zeigt Abbildung 2.17 das Ergebnis. Die Punkte 2. und 3. der zweiten Simulation mit  $H(z)$  werden hier noch leicht verstärkt und verschlechtern die Performance des Schätzers in einem Maße, daß kein Gewinn durch die optimierte Präambel mehr zu verzeichnen ist.

Ein naheliegender Ansatz zur Verbesserung der Schätzerperformance unter diesen Umständen ist die Vergrößerung von  $p$ . Dadurch werden die Überhand genommenen zweifach-Fehler reduziert, sodaß zum Zeitpunkt 0, wenn der Schätzer genau über der Präambel liegt, geringere Fehlerraten erzielt werden. An dieser Stelle sei betont, daß die Abbildungen 2.14, 2.16 und 2.17 nur dem Verständnis dienen. Der für das Einfrieren des Taktoffsets entscheidende Moment ist der Nullpunkt.

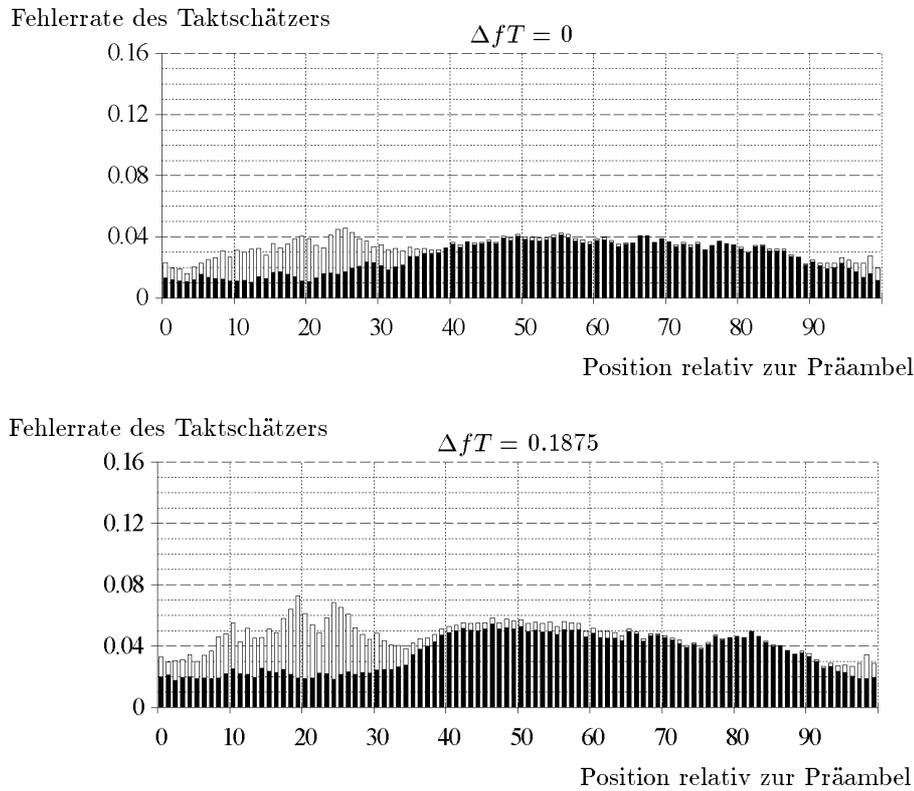


Abbildung 2.18: Taktoffset-Schätzfehlerraten nach einfach- und zweifach-Fehlern getrennt aufgetragen über den Versatz zur Präambel, vollständiges Simulationsmodell, modifiziertes Präambelwort

Als neues Präambelwort werde nun

0011 0000 1100 1001 1110 1000 1010 1101 0101 0101 100

gewählt. Dabei wurden nur in den letzten drei Bits Veränderungen durchgeführt. Sie bewirken, daß  $p$ , wenn der Schätzer genau über der Präambel liegt, jetzt 0.381 ist.

Abbildung 2.18 zeigt die Simulationsergebnisse für die komplette Strecke und Abbildung 2.19 den dazugehörigen Verlauf von  $1 - p$ .

Es ist eine deutliche die Verringerung der Gesamtfehlerraten zum Zeitpunkt 0 zu verzeichnen. Insgesamt ist die Wirkung der Präambeloptimierung bezüglich des Schätzers zwar nicht mehr so effektiv wie für das einfache Simulationsmodell, es ergibt sich jedoch immer noch eine Verbesserung der Taktoffset-Schätzfehlerraten. Eine weitere Anhebung von  $p$  durch Manipulation der Präambel führt bereits wieder zu einem Anstieg der Gesamtfehlerrate zum Zeitpunkt 0.

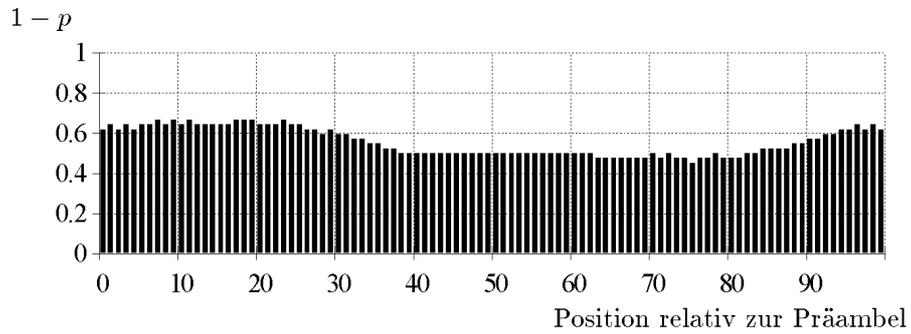


Abbildung 2.19:  $1-p$  aufgetragen über den Versatz zur Präambel für modifiziertes Präambelwort

## 2.4.4 Präambeldetektion mit Taktoffsetschätzwert

### 2.4.4.1 Dezimationsverfahren

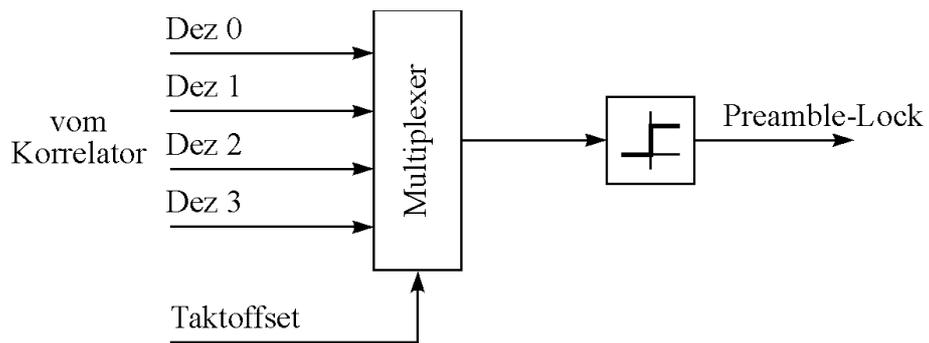


Abbildung 2.20: Einfache Präambeldetektion mittels Dezimation und Schwellwertentscheidung

Die einfachste Möglichkeit zur Präambeldetektion besteht darin, den vom Taktoffsetschätzer gelieferten Wert zur Auswahl des Dezimationspfades zu benutzen. Abbildung 2.20 zeigt diese Struktur. Ist der Korrelationswert größer als eine vorgegebene Schwelle, gilt die Präambel als erkannt. Daraufhin werden die Schätzwerte eingefroren und die Detektion des Datenblocks vorgenommen. Dieses Verfahren ist sehr abhängig von der Güte des Taktoffsetschätzwerts. Liegt dieser im entscheidenden Moment falsch, so muß mit einer erheblich höheren Bitfehlerrate gerechnet werden. Die Abhängigkeit zwischen Dezimationszeitpunkt und Bitfehlerrate zeigt Abbildung 2.13. Für den Fall eines Dezimationsfehlers sinkt daher die Wahrscheinlichkeit der Präambelerkennung.

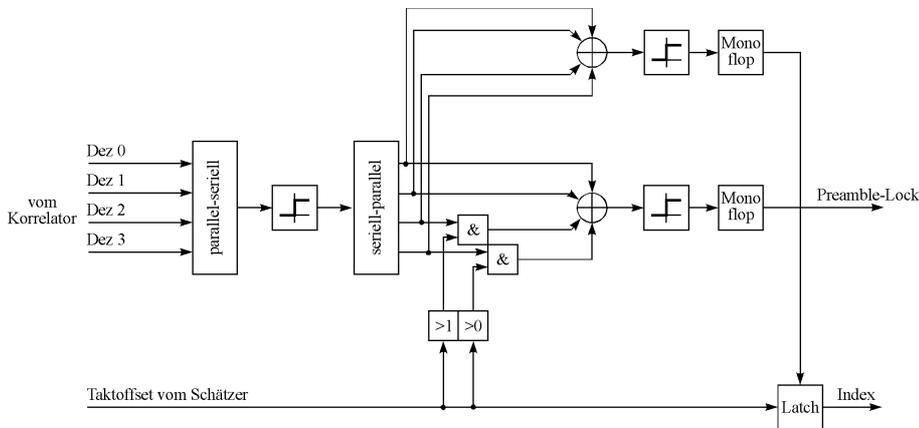


Abbildung 2.21: Einfache Präambeldetektion mittels Summation und Schwellwertentscheidung

#### 2.4.4.2 Summationsverfahren

Die starke Abhängigkeit der Präambelerkennung vom Schätzer ist der Nachteil des Dezimationsverfahrens. Das Summationsverfahren, in Abbildung 2.21 dargestellt, ist ein Versuch von dieser Abhängigkeit loszukommen. Der Taktoffsetschätzwert wird immer noch benötigt, besitzt jedoch nicht die gleiche Bedeutung. Abbildung 2.21 zeigt das Prinzip dieses Verfahrens. Es reicht bereits aus, in einem der Dezimationspfade eine Überschreitung der Korrelationsschwelle zu erkennen. Der Taktoffsetschätzwert wird nur noch dazu benötigt, die Synchronisation auf SymbolEbene zu gewährleisten. Er dient dazu, eventuell zu früh auftretende Korrelationsschwellenüberschreitungen auszublenden. Der zweite Summationspfad zum Einfrieren des Taktoffsets wird nötig, damit, falls ein Korrelationswert durch die Und-Gatter ausgeblendet wurde, der Taktoffset zum Preamble-Lock-Signal paßt. Ansonsten ergäben sich bei auf den Symbolgrenzen zufällig wechselnden Taktoffsets Timing-Fehler. Es muß jedoch beachtet werden, in wie weit eine Präambeldetektion mit anschließendem Empfang des Datenblocks sinnvoll ist, falls ein Taktoffsetschätzfehler vorliegt. In diesem Fall erfolgt die gesamte Datendetektion zum falschen Dezimationszeitpunkt, was einen Anstieg der Bitfehlerraten zur Folge hat. Ob sich das Summationsverfahren lohnt, hängt davon ab, mit welchem SNR standardmäßig zu rechnen ist, und welche Bitfehlerzahl bezüglich einer Fehlerkorrektur tolerierbar ist.

Abbildung 2.22 zeigt die Performance der beiden Verfahren für unterschiedliche SNRs. Ein Vergleich mit Abbildung 2.23 zeigt die starke Bindung des Dezimationsverfahrens an die Taktoffset-Schätzfehlerrate. Aus Abbildung 2.23 läßt sich auch der Gewinn durch die optimierte Präambel ablesen.

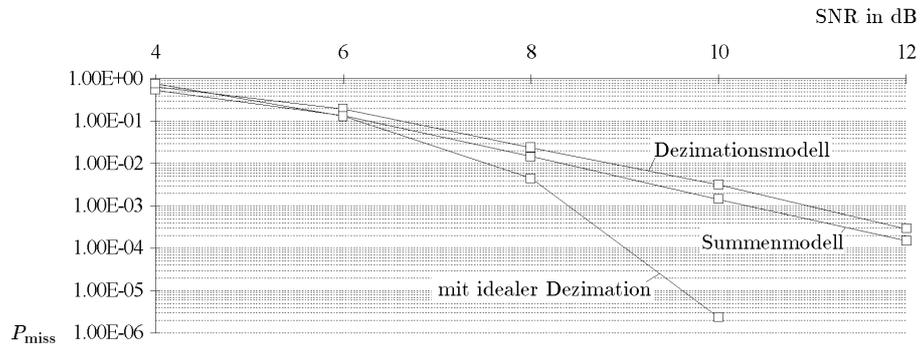


Abbildung 2.22: Dezimations- und Summationsverfahren im Vergleich

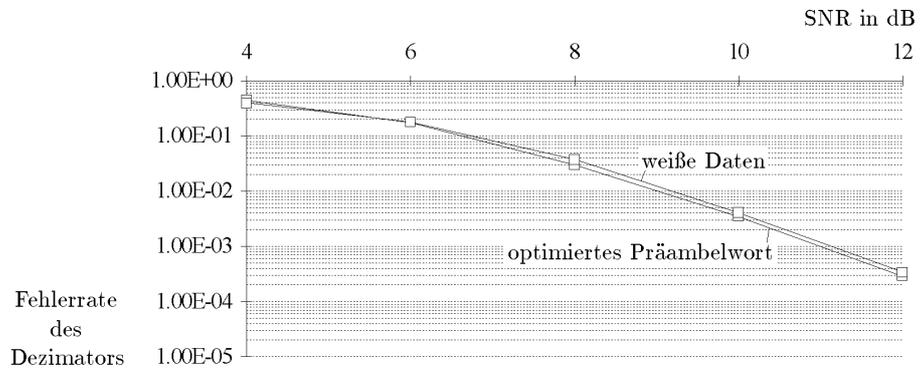


Abbildung 2.23: Vergleich: Taktoffsetschätzer-Fehlerrate bei weißen Daten und mit optimierter Präambel

## 2.4.5 Präambeldetektionswahrscheinlichkeiten

### 2.4.5.1 Missed Hit

Dieser Abschnitt berechnet die Missed-Hit WSK  $P_{miss}$  bei der Präambeldetektion. Die Präambel wird nicht erkannt, wenn von den  $m$ -Korrelationsbits mehr als die  $n$  Bits falsch sind.  $n$  ist die Zahl der tolerierten Bitfehler. Die Berechnung der Missed-Hit WSK wird dadurch etwas verkompliziert, daß die Einzelbitfehlerwahrscheinlichkeit innerhalb des Korrelationswortes unterschiedlich ist, je nachdem ob ein Bitwechsel vorliegt oder nicht. Für den Fall des Bitwechsels ergebe sich die Einzelbitfehlerwahrscheinlichkeit  $p_{err,o}$ , andernfalls  $p_{err,e}$ . Mit den oben beschriebenen Zusammenhängen folgt aufgrund der Bandbegrenzung:

$$p_{err,o} > p_{err,e} \quad (2.33)$$

Man stelle sich die Präambel als Folge  $m_e$  gleicher und  $m_o$  alternierender Bits vor. Dies ändert nichts an der Bitfehlerwahrscheinlichkeit. Die Korrelationsei-

genschaften würden dabei stark verschlechtert, was hier jedoch nicht von Interesse ist. Man definiere folgende Ereignisse:

- Das Ereignis  $A_i$  sei, wenn im even-Block genau  $i$ -Fehler aufgetreten sind.
- Das Ereignis  $B_i$  sei, wenn im odd-Block genau  $i$ -Fehler aufgetreten sind.

Damit ergibt sich für die Verbundwahrscheinlichkeit, daß im even-Block genau  $i$  Fehler und im odd-Block genau  $j$  Fehler aufgetreten sind:

$$p(A_i, B_j) = p(A_i|B_j) \cdot p(B_j) = p(A_i) \cdot p(B_j) \quad (2.34)$$

Das letzte Gleichheitszeichen gilt, da die Ereignisse  $A_i$  und  $B_j$  voneinander unabhängig sind. Die Gesamtheit aller Verbundereignisse ist das sichere Ereignis:

$$\begin{aligned} \sum_{i=0}^{m_e} \sum_{j=0}^{m_o} p(A_i, B_j) &= \sum_{i=0}^{m_e} p(A_i) \cdot \sum_{j=0}^{m_o} p(B_j) \\ &= 1 \end{aligned} \quad (2.35)$$

Es gilt nun aus der Gesamtheit aller einzelnen Verbundereignisse diejenigen auszuwählen, für die die Präambel nicht erkannt wird. Dies ist der Fall, wenn

$$i + j > n \text{ mit } i > 0 \text{ und } j > 0 \quad (2.36)$$

gilt. Damit folgt sofort für  $P_{miss}$

$$P_{miss} = \sum_{i=0}^{m_e} \sum_{j=\max(n-i+1, 0)}^{m_o} p(A_i, B_j) \quad (2.37)$$

$$= \sum_{i=0}^{m_e} \sum_{j=n-i+1}^{m_o} p(A_i, B_j) + \sum_{i=n+1}^{m_e} p(A_i) \quad (2.38)$$

Es bleibt die Wahrscheinlichkeiten  $p(A)$  und  $p(B)$  zu bestimmen. Die Wahrscheinlichkeit  $p_W$ , daß in einer Gruppe von  $m$  Bits genau  $i$  falsch sind, ist für eine Einzelbitfehlerwahrscheinlichkeit von  $p_B$  gegeben durch:

$$p_W(m, i, p_B) = \binom{m}{i} p_B^i (1 - p_B)^{m-i} \quad (2.39)$$

Damit gilt:

$$p(A_i) = p_W(m_e, i, p_{err,e}) \quad (2.40)$$

bzw.

$$p(B_i) = p_W(m_o, i, p_{err,o}) \quad (2.41)$$

Abbildung 2.22 enthält die in diesem Abschnitt auf Grund der Fehlerraten  $p_{err,e}$  und  $p_{err,o}$  berechnete Wahrscheinlichkeit  $P_{miss}$ . Es ist deutlich zu erkennen, daß die Fehlerrate bei der Taktoffsetschätzung für  $P_{miss}$  bei Summations- und Dezimationsmodell dominierend ist.

tolerierte Fehler $n$	Falsch-Alarm-WSK $p_E$	Abstand $t_E$
0	$2.32 \cdot 10^{-10}$	71 m
1	$7.68 \cdot 10^{-9}$	130 s
2	$1.23 \cdot 10^{-7}$	8.12 s
3	$1.28 \cdot 10^{-6}$	780 ms
4	$9.65 \cdot 10^{-6}$	103 ms

Tabelle 2.4: Falsch-Alarm-Wahrscheinlichkeiten für  $m = 32$ 

tolerierte Fehler $n$	False Alarm-WSK $p_E$	Abstand $t_E$
0	$1.13 \cdot 10^{-13}$	101 d
1	$5.00 \cdot 10^{-12}$	2.3 d
2	$1.07 \cdot 10^{-10}$	2.6 h
3	$1.51 \cdot 10^{-9}$	11 m
4	$1.55 \cdot 10^{-8}$	64 s

Tabelle 2.5: Falsch-Alarm-Wahrscheinlichkeiten für  $m = 43$ 

#### 2.4.5.2 False-Alarm

Werden  $n$  Bitfehler für eine Präambeldetektion toleriert, so wird eine Präambel fälschlicherweise erkannt, falls mindestens  $m - n$  Bits durch Zufall dem Korrelationswort entsprechen. Die Wahrscheinlichkeit  $p_E$  dafür ist die Summe über die Wahrscheinlichkeiten, daß durch Zufall  $i$ ,  $i \in [m - n \dots m]$  Bits einem bestimmten Muster entsprechen:

$$p_E = \sum_{i=m-n}^m \binom{m}{i} p_B^i (1 - p_B)^{m-i} \quad (2.42)$$

Dabei ist  $p_B = 0.5$ .

Bei einem freilaufenden Empfänger, der im Rauschen des Kanals nach Präambeln sucht, ist der mittlere Abstand zwischen zwei fälschlicherweise erkannten Präambeln  $t_E = \frac{T}{p_e}$ .

Damit ergibt sich für eine Korrelationslänge von  $m = 32$  und eine Symbolrate von  $10^6$  pro Sekunde Tabelle 2.4:

Wird die Schwelle für die Erkennung eines Korrelationswortes in der Präambel zu niedrig gesetzt, wird der mittlere Abstand zwischen zwei fehlerhaft detektierten Präambeln sehr gering. Falls dieser Abstand für die Anwendung zu klein wird, bestehen prinzipiell drei Möglichkeiten, um dies zu kompensieren.

1. Eine Verlängerung der für die Korrelation genutzten Daten reduziert die Fehlerwahrscheinlichkeit stark. Wird die Korrelationswortlänge auf die gesamte Präambellänge inklusive den Bits für die Anhebung der Schätzerperformance ausgedehnt, so ergibt sich für  $m = 43$  Tabelle 2.5.

2. Es besteht die Möglichkeit, durch zusätzliche Redundanz in den Nutzdaten die Plausibilität derselben zu überprüfen. Für den Fall, daß eine Präambel fälschlicherweise erkannt wurde, kann eine Fehlererkennung bei den Daten diesen Irrtum aufdecken. Dieses Verfahren läßt sich leicht in eine eventuell vorhandene Fehlerkorrektur integrieren.
3. Eine Auswertung eines Empfangsfeldstärkesignals kann die Suche des Empfängers nach Präambeln im Kanalrauschen ohne Sendesignal unterbinden. Ein derartiges Empfangsfeldstärkesignal läßt sich auch für eine verbesserte Fehlerkorrektur benutzen.

## 2.5 Abschließende Simulation auf bandbegrenztem AWGN-Kanal

Die folgenden Simulationen wurden mit den in den vorigen Abschnitten ermittelten Parametern durchgeführt. Getrennt nach äußeren Bedingungen und Schaltungs-Parametern zeigen die Tabellen 2.6 und 2.7 die gewählten Einstellungen:

Eigenschaft	Einstellung
Frequenzoffset	0.1875 / T
Taktoffset	3, 4
dc-Offset	0
Bandbreite $f_{\Delta}$	1.5 / T
SNR	8, 10, 12 dB

Tabelle 2.6: äußere Bedingungen

Zur Präambeldetektion diente das Dezimationsverfahren. In Simulationen zeigte sich außerdem, daß es prinzipiell nicht notwendig ist, den Frequenzoffsetschätzwert für die Dauer der Datendetektion einzufrieren. Da zwischen den Übertragungsblöcken jedoch Pausen bestehen, liegt das Schätzfenster gegen Ende des Datenblocks über dem Kanalrauschen. Aus diesem Grunde wird es doch erforderlich werden, den Frequenzoffsetschätzwert für die Dauer der Datendetektion einzufrieren. Die Darstellung der Fehlerraten in Abbildung 2.24 erfolgt hier als Histogramm. Dabei wurde ermittelt, wieviele Fehler innerhalb eines Datenblocks von 600 Bits auftreten. Das Histogramm zeigt einen Verlauf, der einer Binomial-Verteilung, wie sie für einen gedächtnislosen AWGN-Kanal zu erwarten ist, gut entspricht.

Abbildung 2.25 zeigt den Vergleich des Systems, das ohne Präambel und Blockorientierung wertkontinuierlich über weißen Daten arbeitet (Tabelle 2.13), mit dem blockorientierten, diskreten System. Dabei sind beim wertdiskreten System die Fälle der nicht erkannten Präambeln herausgenommen. Es ist im Rahmen der Simulationsgenauigkeit kein Performance-Verlust zu verzeichnen.

Eigenschaft	Einstellung
Eingangsabtastung	8 Werte pro Symbol
Abtastung der Differenzphase	4 Werte pro Symbol
Schätzerlänge	42
AD-Wandlersauflösung	6 Bits
Auflösung nach $H(z)$	6 Bits
Phasenauflösung	7 Bits
kartesische Auflösung vor Schätzersummation	4 Bits
Auflösung nach Summation	7 Bits
Auflösung für Frequenzoffset	7 Bits
Auflösung für Maximumsuche auf Betrag	8 Bits
Präambellänge	43 Symbole
Korrelationslänge	32 Symbole
tolerierte Bitfehler bei Präambel	3

Tabelle 2.7: Schaltungs-Parameter

Falls nicht erkannte Präambeln zu den Fehlerereignissen gezählt werden, schneidet die blockorientierte Verarbeitung für niedrige SNRs deutlich schlechter ab.  
[

## 2.6 Zusammenfassung wichtiger Ergebnisse

Hier werden noch einmal wichtige Ergebnisse dieses Abschnitts, sofern sie nicht aus den Tabellen 2.6 und 2.7 hervorgehen, zusammengefaßt:

1. Die Frequenzoffsetschätzung ist datenunabhängig. Damit kann die Präambeldetektion stark vereinfacht werden.
2. Ein Clipping des MSBs nach der Schätzersummation ist nicht möglich.
3. Die Maximumsuche auf dem Betrag benötigt eine geringere Auflösung als auf dem Betragsquadrat.

]

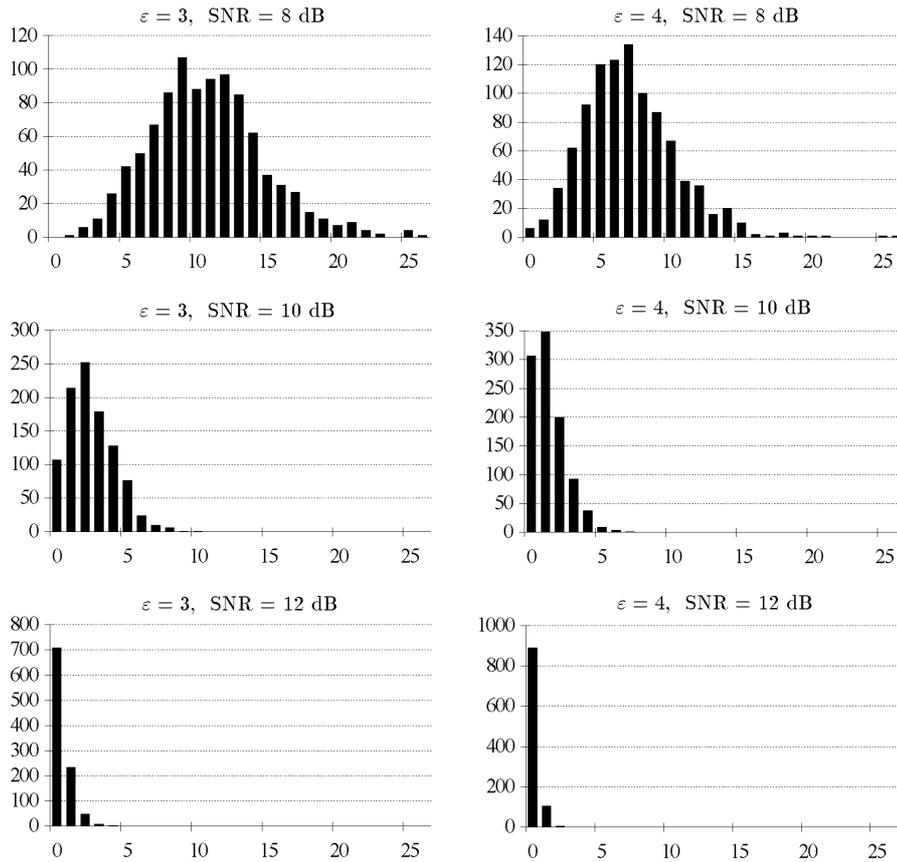


Abbildung 2.24: Fehlerverteilungen für Datenblöcke von 600 Bits über 1000 Blöcke, auf der Abszisse: Anzahl Fehler pro Block, auf der Ordinate: Häufigkeit des Auftretens

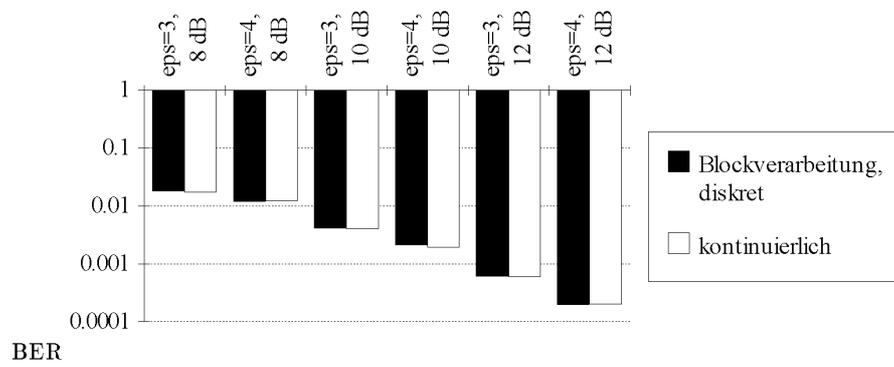


Abbildung 2.25: Vergleich des freilaufenden, kontinuierlichen Systems mit dem blockorientierten, diskreten System

## Kapitel 3

# Kanaleigenschaften und Fehlerkorrektur

In den vorigen Kapiteln wurde ein einfacher bandbegrenzter AWGN-Kanal in die Simulationsstrecken eingesetzt. Der reale Mobilkanal besitzt jedoch ein etwas anderes Verhalten als der AWGN-Kanal. Insbesondere kann sich das SNR während eines Datenpaketes ändern. Die Länge und Häufigkeiten dieser Fading-Störungen beeinflusst in starkem Maße die Performance des Transceivers, da die Datenpakete fehlerfrei übertragen werden müssen. Es nützt daher nichts, wenn der Kanal im Normalfall ein sehr gutes SNR besitzt, für kurze Zeit jedoch starke Einbrüche der Empfangsleistung auftauchen. Dieses Kapitel untersucht daher die Eigenschaften des Kanals und betrachtet verschiedene Fehlerkorrekturverfahren im Hinblick auf ihre Eignung für die Kanaleigenschaften.

### 3.1 Kanaleigenschaften

#### 3.1.1 mittleres SNR

Der Kanal stellt zunächst eine gewisse Empfangsleistung  $P_R$  an der Antenne zur Verfügung. Diese Leistung wird in nachfolgenden Verstärkerstufen angehoben und letztlich dem AD-Wandler zugeführt. Für eine vorgegebene Empfangsleistung berechnet sich das SNR des Kanal damit wie folgt:

$$\frac{E_S}{N_0} = \frac{E_b}{N_0} = \frac{P_R}{F \cdot k \cdot \Theta \cdot B} \quad (3.1)$$

Darin sind  $F = 10$  dB die Rauschzahl der Verstärkerstufen,  $k = 1.38 \cdot 10^{-23} \frac{\text{Ws}}{\text{K}}$  die Boltzmannkonstante,  $\Theta = 293$  K die Betriebstemperatur und  $B = 1.5/T$  die Filterbandbreite.

Empfangsfeldstärke

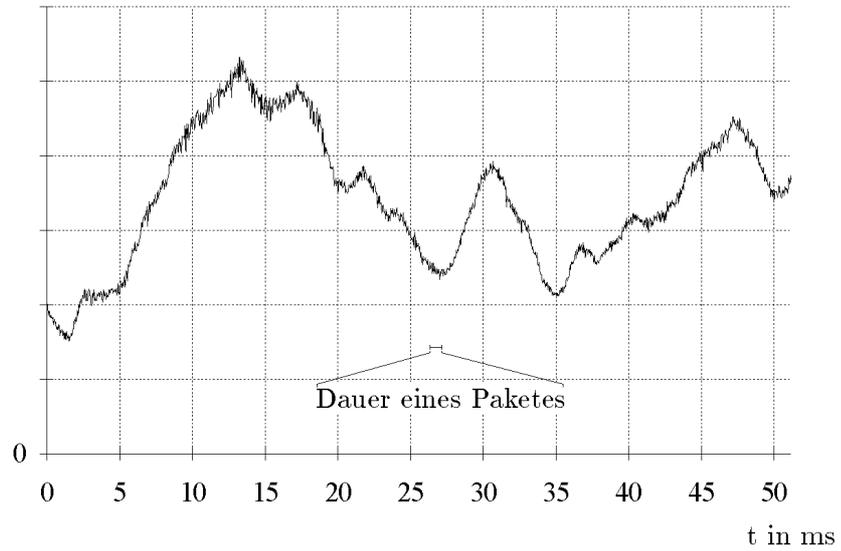


Abbildung 3.1: typischer Empfangsfeldstärkeverlauf des Kanals

rel. Häufigkeit

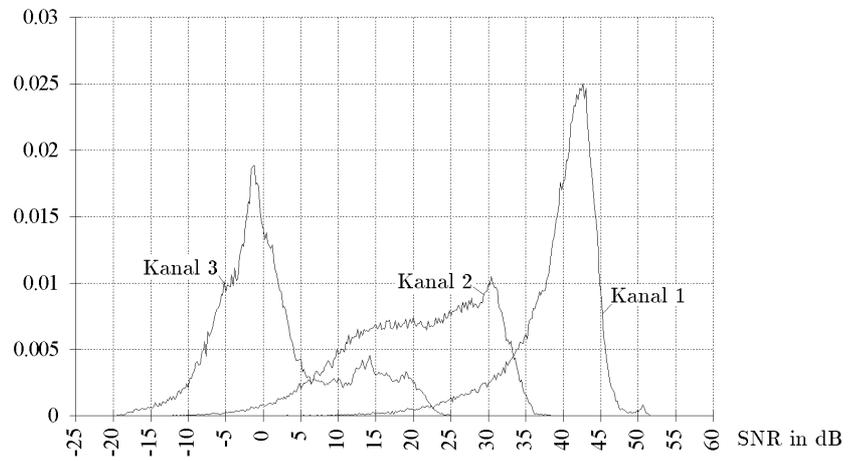


Abbildung 3.2: Histogramm der mittleren Empfangsfeldstärke pro Übertragungsblock, Langzeitanalyse (70 s, 70 s, 49 s)

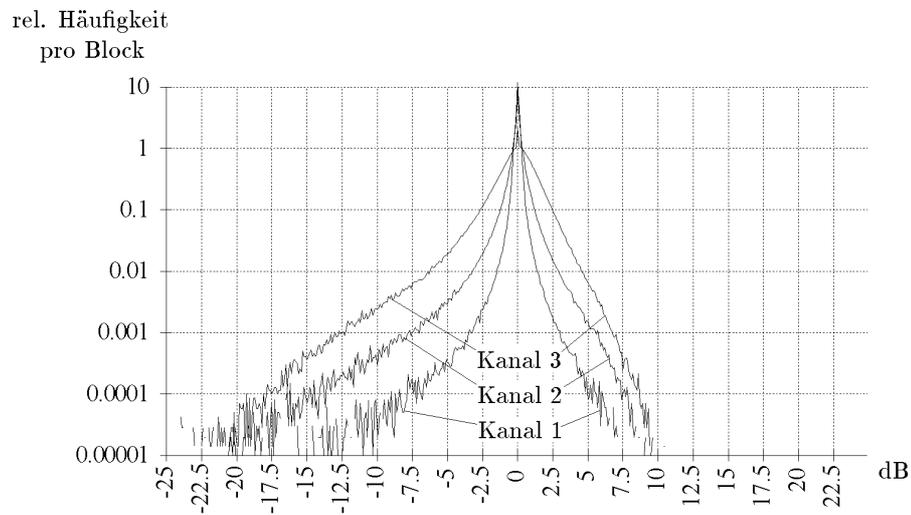


Abbildung 3.3: Histogramm der Abweichungen von mittlerer Empfangsfeldstärke

Es wurden drei Kanäle näher untersucht:

- Kanal 1:
  - $v_1 = 100$  km/h
  - $v_2 = 100$  km/h
  - $d = 300$  m
- Kanal 2:
  - $v_1 = 100$  km/h
  - $v_2 = 100$  km/h
  - $d = 500$  m
- Kanal 3:
  - $v_1 = 113$  km/h
  - $v_2 = 54$  km/h
  - $d(t = 0s) = 900$  m,  $d(t = 50s) = 80$  m

Darin sind  $v_1$  und  $v_2$  die Geschwindigkeiten der Fahrzeuge.  $d$  ist der Abstand zwischen beiden Fahrzeugen.

Abbildung 3.2 zeigt das mittlere SNR der Kanäle für die Dauer einer Präambel und Paketübertragung als Histogramm. Das Histogramm wurde für die Kanäle 1 und 2 über 70 Sekunden und für Kanal 3 über 49 Sekunden erstellt. Um 600 Datenblockbits, die wiederum aus 500 Nutzdatenbits und ca. 100 Prüfbits bestehen, Präambel (43 Bits) und Bits für eine grobe Frequenzkorrektur aufzunehmen, war die Blocklänge auf 700 Symbole eingestellt. Die Breite der Histogrammintervalle beträgt 0.2 dB.

Kanal 1 weist die meiste Zeit ein sehr gutes SNR auf. Das SNR von ca. 42 dB läßt problemlos eine fehlerfreie Datentübertragung zu. Die Wahrscheinlichkeit, ein geringeres SNR bei diesem Kanal anzutreffen, nimmt mit der Verschlechterung des SNRs schnell ab.

Kanal 2 hat aufgrund des größeren Abstandes zwischen den Fahrzeugen ein schlechteres mittleres SNR. Zusätzlich tritt hier eine breitere Streuung der Empfangleistung auf. Dies läßt sich dadurch erklären, daß auf der längeren Distanz mehr Störungsursachen auftreten können.

Kanal 3 weist die problematischsten Übertragungseigenschaften auf. Er besitzt eine sehr weite Streuung der Empfangsleistung. Dies ist mit dem Aufeinanderzubewegen der beiden Fahrzeuge zu erklären.

Das mittlere SNR des Kanals bei einer Blockübertragung beschreibt, ob eine Datenübertragung prinzipiell möglich ist. Dies ist bei den Kanälen 1 und 2 sicher der Fall. Bei Kanal 3 sind ausreichende Übertragungseigenschaften erst ab einem gewissen Abstand gegeben.

### 3.1.2 Fading während der Paketübertragung

Für die Auslegung der Fehlerkorrektur ist die Anzahl der Bitfehler je Paket von Bedeutung. Das Histogramm der Anzahl Fehler pro Block besitzt für den AWGN-Kanal die Struktur einer Binomialverteilung. Je nach Häufigkeit und Stärke der Einbrüche auf dem realen Mobilkanal ergeben sich daher unterschiedliche Fehlerraten und Bündelfehlerlängen. An dieser Stelle ist anzumerken, daß die vorliegenden Kanaldaten von der DLR mit einer Abtastrate von 40 kHz aufgenommen wurden. Die Abtastwerte müssen daher Tiefpaß-interpoliert werden, um auf die nötige Empfängerabtastrate  $8/T$  zu kommen. Dies bedeutet, daß pro DLR-Sample 25 Symbole übertragen werden. Störungen, die kürzer als  $25 \mu\text{s}$  dauern, werden daher von den Kanalmessungen nicht erfaßt.

Abbildung 3.3 zeigt die Ergebnisse dieser Kanalanalyse. Es ist die Anzahl der Samples aufgetragen, die von dem Empfangsleistungsmittelwert des betrachteten Datenblocks um  $x$  dB abweichen.  $x$  ist auf der Abszisse mit einer Auflösung von 0.1 dB aufgetragen. Die Datenblocklänge beträgt wieder 700 Bits. Dies entspricht 28 Samples aus dem Kanal. Maximal können daher pro Block 28 Samples eine bestimmte Abweichung vom Mittelwert haben. Dadurch erklären sich die relativen Häufigkeiten größer 1 in Abbildung 3.3.

Generell ist eine Unsymmetrie der Verteilung der Abweichungen in Abbildung 3.3 zu erkennen. Es treten speziell bei Kanal 3 teilweise sehr starke Empfangsleistungseinbrüche auf, die bis an 25 dB heranreichen. Je nachdem in welchem Zustand sich der Kanal vorher befunden hat, müssen in diesem Fall mindestens die 28 Bits, die während dieses Kanal-Samples übertragen wurden, als fehlerhaft eingestuft werden.

### 3.1.3 Simulationsergebnisse mit DLR-Kanälen

Abbildung 3.4 zeigt Simulationsergebnisse auf Kanal 2 und 3. Es sind vier Simulationen dargestellt. Die jeweils linke Grafik enthält das mittlere SNR des betrachteten Kanalausschnitts.

Die rechte Grafik zeigt die Fehlerzahl pro Block als Histogramm. Es wurde über 1000 Blöcke mit 600 Datenbits simuliert. Die Summation der einzelnen Säulen im Histogramm ergibt wegen der endlichen Wahrscheinlichkeit, daß eine Präambel nicht erkannt wird, nicht 1000. Über den fehlenden Betrag läßt sich  $P_{\text{miss}}$  ausrechnen.  $P_{\text{miss}}$  ist rechts neben dem Histogramm angetragen. Das Histogramm reicht theoretisch bis zu 600 Fehlern pro Block. Fehlerzahlen größer 29 sind jedoch einer Darstellung mit einem vernünftigen Maßstab gewichen. Es traten in diesem Bereich keine signifikanten Merkmale auf.

Ausgangspunkt für die Wahl des Fehlerkorrekturverfahrens sind die Kanalabschnitte mit schlechteren SNRs. Daher sind Kanalabschnitte gewählt worden, die ein starkes Einbruchverhalten und teilweise schlechte SNRs aufweisen.

Diskussion der einzelnen Simulationen:

1. Kanal 2 ab 47.5 s: Dieses Kanalteilstück weist im Mittel ein recht gutes SNR auf. Es wird jedoch von kurzen Einbrüchen gestört. Das Histogramm zeigt deutlich, daß, für den Fall, daß die Präambel erkannt worden ist, gute SNRs zur Verfügung stehen. Dies ist aus dem Quotienten aus der 0- und der 1-Fehlerhäufigkeit, der hier sehr groß ist, ersichtlich.
2. Kanal 3 ab 42.5 s und ab 45 s: Diese beiden Kanalteilstücke weisen schlechte Übertragungseigenschaften auf. Die Präambel wird nur selten erkannt. Der Quotient aus der 0- und der 1-Fehlerhäufigkeit ist sehr viel kleiner geworden. Daher besitzt der Kanal auch für den Fall, daß die Präambel erkannt wurde, schlechte SNRs. Untersuchungen, bei denen die Präam-

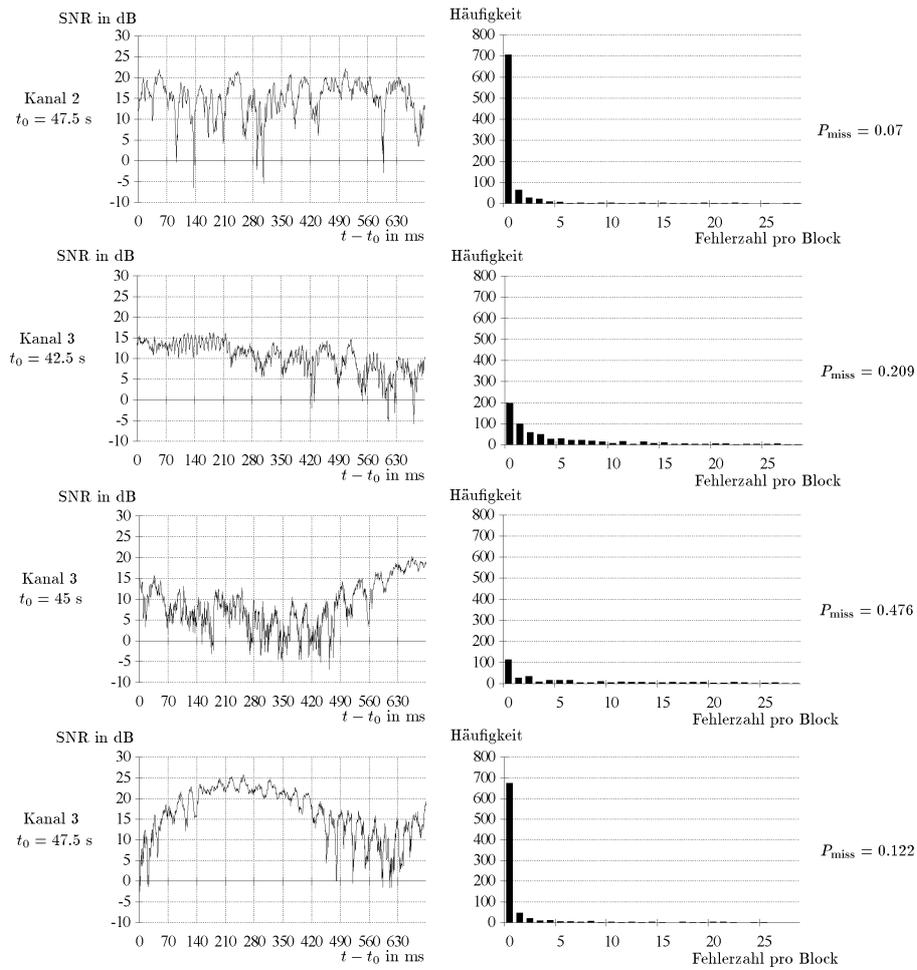


Abbildung 3.4: Simulationsergebnisse über 1000 Datenblöcke auf DLR-Kanälen

beldetektion ideal detektierend eingestellt wurde, haben dies bestätigt. Es treten dann große Häufigkeiten bei vielen Fehlern auf, die auch durch sehr leistungsfähige Fehlerkorrekturverfahren nicht mehr zu bewältigen sind. Eine Absenkung der Präambeldetektionsschwelle erscheint daher, auch wenn die Erkennungsraten der Präambel gering sind, nicht sinnvoll.

3. Kanal 3 ab 47.5 s: Dieses Kanalteilstück weist wieder recht gute Kanaleigenschaften auf. Der Quotient aus der 0- und der 1-Fehlerhäufigkeit ist recht hoch.

Der Quotient aus der 0- und der 1-Fehlerhäufigkeit bzw. der 1- und 2-Fehlerhäufigkeit kann als Auswahlkriterium für das Fehlerkorrekturverfahren dienen. Sind diese Quotienten groß, so bieten sich Korrekturverfahren an, die nur wenige Fehler korrigieren können. Durch Anwendung eines Hamming-Codes tritt bei Kanal 2 und 3 ab 47.5 s in ca. 75 % der Übertragungspakete kein Fehler mehr auf. Sollte dies nicht ausreichen, oder der Kanal im Mittel schlechter sein, lassen sich immer noch kleinere Blöcke innerhalb des großen Datenblocks durch Hamming-Codes schützen. Wird der Hamming-Code zur Fehlerkorrektur eingesetzt, so lassen sich bei einem Code der Ordnung  $K$   $n = 2^{2^K - 1} - 2^{2^K - K - 1}$  Fehlermuster innerhalb eines geschützten Blocks erkennen. Für einen Kanal, der zwei Zustände besitzt, einen guten und einen schlechten, scheinen die einfachen Hamming-Codes eine gute Wahl zu sein. Ist die Anwendung auf die Ausnutzung jedes dBs SNR angewiesen, so werden die aufwendigeren BCH-Code enger in Betracht gezogen werden müssen.

## 3.2 Fehlerkorrekturverfahren

Es gilt nun ein Fehlerkorrekturverfahren auszuwählen, das dem Kanal möglichst gut angepaßt ist und einfache Implementierungsmöglichkeiten bietet. Es werden zunächst 4 Verfahren zur Fehlererkennung bzw. -korrektur vorgestellt, von denen dann 2 näher diskutiert werden.

Als prinzipiell mögliche Kanalcodierungsverfahren werden folgende ausgewählt:

1. Hamming-Codierung
2. BCH-Codierung
3. Reed-Solomon-Codes
4. Faltungscodes

Verfahren	Fehlerkorrekturkraft	Komplexität
Hamming	1 Fehler pro Block	gering
BCH	$e$ Fehler pro Block	mittel
Reed-Solomon	$e$ Fehler pro Block, zusätzlich Bündelfehler	mittel-hoch
Faltungscodes	hoch speziell für Kanäle mit ISI	sehr hoch (Viterbi-Dekoder)

Tabelle 3.1: Fehlerkorrekturcodes mit ihren Eigenschaften

Diese 4 Verfahren werden mit ihren Eigenschaften in Tabelle 3.1 aufgelistet.

Der Hamming-Code bietet sich an, wenn innerhalb der äußeren Blockstruktur, die durch die Präambel eingeleitet wird, eine innere Blockstruktur eingeführt wird. Diese inneren Blöcke können dann über einen Hamming-Code geschützt werden. Da die minimale Hamming-Distanz beim Hamming-Code jedoch nur  $d_{\min} = 3$  beträgt, muß man sich bei diesem Code entscheiden, ob man entweder einfach- bis zweifach-Fehler erkennen will, oder ob ein Fehler korrigierbar ist. Da es wichtig ist, zu erkennen, daß ein Teilblock falsch übertragen wurde, ist die Fehlerkorrektur, falls es keine andere Information über die Zuverlässigkeit eines Blocks gibt, nicht nutzbar. Denkbar wäre auch ein Konzept, bei dem Hamming-Codes innerhalb von kleinen Blöcken Fehler korrigieren, und die Information über die Zuverlässigkeit von außen zum Beispiel in Form einer Empfangsfeldstärkesignals zur Verfügung steht. Die Anforderungen an die Bandbreite eines solchen Signals wären dann von der Blocklänge abhängig.

BCH-Codes besitzen eine höhere Leistungsfähigkeit als die Hamming-Codes. Sie können bis zu  $e$ -fache Fehler innerhalb eines Blocks erkennen. Von  $e$  hängt die Größe der hinzugefügten Redundanz ab.

Die Reed-Solomon-Codes sind eine Verallgemeinerung der binären BCH-Codes. Die RS-Codes spalten einen Block wiederum in Unterblöcke auf. Sie sind dadurch in der Lage, zusätzlich zu den auf  $e$ -Unterblöcke verteilten Fehlern auch höhere Fehlerzahlen innerhalb eines Unterblocks zu korrigieren. Sie eignen sich daher besonders für Kanäle mit Bündelfehlerstruktur.

Die Faltungscodes besitzen zwar sehr gute Fehlerkorrektureigenschaften und eignen sich besonders für Kanäle mit Inter-Symbol-Interference, der erforderliche Dekodieraufwand in Form eines Viterbi-Dekoders übersteigt jedoch das hier vertretbare Maß. Zusätzlich wird die besondere Stärke der Faltungscodes, nämlich die Minderung des Einflusses von ISI, nicht benötigt.

Im folgenden werden daher die Hamming- und BCH-Codes näher untersucht.

Um ein besseres Verständnis für die Codierungs- und Decodierungsalgorithmen zu erhalten, wird zunächst kurz auf den theoretischen Hintergrund der beiden Codierungsverfahren eingegangen. Für eine detaillierte Herleitung werde hier auf [5] verwiesen.

### 3.2.1 Allgemeines zur Codierung

Eine Kanalcodierung, die Redundanz hinzufügt, bildet  $2^n$  mögliche Informationsvektoren  $\vec{x}$  auf  $2^n$  Codevektoren  $\vec{y}$  der Dimension  $m$  mit  $m > n$  ab. Die relative Redundanz ist dabei definiert als

$$r = \frac{m - n}{n} \quad (3.2)$$

Da  $m > n$  gilt, gibt es im Bildraum der Dimension  $m$  Vektoren, die nicht zu den möglichen Codevektoren gehören. Dadurch werden Fehler erkenn- bzw. korrigierbar.

Die Hamming-Distanz  $d(\vec{y}_1, \vec{y}_2)$  ist der Abstand zweier Codevektoren  $\vec{y}_1$  und  $\vec{y}_2$ . Bei binärwertigen Codes ist dies die Zahl unterschiedlicher Nullen und Einsen in den beiden Codewörtern  $y_1$  und  $y_2$ . Ein Code ist um so sicherer, desto größer der Abstand zwischen zwei benachbarten Codevektoren ist, so daß eine Störung von einzelnen Stellen nicht zu einem neuen gültigen Codewort führt. Es gilt daher die minimale Hamming-Distanz  $d_{\min}$  zu maximieren.

Es läßt sich leicht einsehen, daß ein Code in der Lage ist

$$e \leq \frac{d_{\min} - 1}{2} \quad (3.3)$$

Fehler zu korrigieren. Es kann jedoch noch Fehlermuster mit mehr Einzelfehlern geben, die auch noch korrigiert werden können.

Es gilt zusätzlich folgende Ungleichung für  $n$ ,  $m$  und  $e$ :

$$2^n \sum_{i=0}^e \binom{m}{i} \leq 2^m \quad (3.4)$$

Ein Code heißt dicht gepackt, wenn in (3.4) das Gleichheitszeichen gilt. In diesem Fall gibt es keine Fehlermuster mit mehr Einzelfehlern als  $e$ , die noch korrigiert werden können.

### 3.2.2 Hamming-Codes

#### 3.2.2.1 Theorie

Hamming-Codes gehören zu den linearen Blockcodes und besitzen folgende Eigenschaften:

1.  $m = 2^k - 1$
2.  $n = 2^k - k - 1$
3.  $d_{\min} = 3$

Hamming-Codes sind daher dicht gepackt. Der Codier- und Decodiervorgang läßt sich mit

$$\vec{x} = (x_0, x_1, \dots, x_{n-1}) \quad (3.5)$$

$$\vec{y} = (y_0, y_1, \dots, y_{m-1}) \quad (3.6)$$

$$\text{mit } x_i, y_i \in \mathbb{GF}_z \quad (3.7)$$

besonders einfach in Matrizenform schreiben.

Die Codierung erfolgt folgendermaßen:

$$\vec{y} = \vec{x}G \quad (3.8)$$

Dabei sind die Multiplikationen wie bei den reellen Zahlen, die Additionen modulo 2 durchzuführen. Wird die Generatormatrix systematisch gewählt, daß heißt, daß das Codewort das Informationswort direkt enthält, so ist

$$G = [E_n | P] \quad (3.9)$$

$E_n$  ist die  $n$ -dimensionale Einheitsmatrix, und  $P$  eine  $n \times (m - n)$  Prüfmatrix. Durch die systematische Form lassen sich die Informationsbits aus dem Codewort direkt wiedergewinnen.

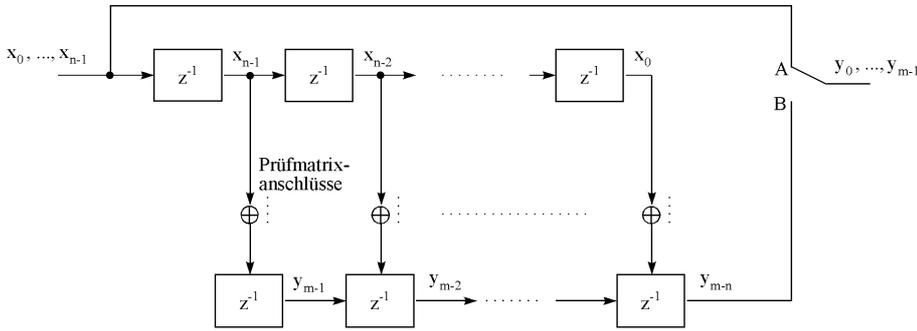


Abbildung 3.5: Hamming-Codierung

Der empfangene Vektor  $\vec{z}$  ist die Überlagerung des Codevektors  $\vec{y}$  mit einem Fehlervektor  $\vec{e}$ .

$$\vec{z} = \vec{y} + \vec{e} \tag{3.10}$$

Der Empfänger bildet nun

$$\vec{s} = \vec{z}H^T = \vec{y}H^T + \vec{e}H^T \tag{3.11}$$

Die Decodiermatrix  $H$  wird im systematischen Fall gegeben durch:

$$H^T = \begin{bmatrix} P \\ - \\ E \end{bmatrix} \tag{3.12}$$

und ist damit orthogonal zu  $G$ . Es folgt:

$$\vec{y}H^T = \vec{0} \tag{3.13}$$

Damit bleibt von (3.11) nur der Fehlerterm  $\vec{e}H^T$  übrig. Dieser Term wird Syndrom genannt und hat für die  $m$  möglichen Fehlerpositionen unterschiedliche Werte. Der Empfänger benutzt das Syndrom, um in einer Tabelle nachzuschlagen, welches Informationsbit fehlerhaft ist und invertiert es.

### 3.2.2.2 Implementation

**Codierung:** Hamming-Codes lassen sich mit wenig Aufwand implementieren. Abbildung 3.5 zeigt das Prinzip der Hamming-Codierung mit Hilfe eines Schieberegisters. Der Schalter steht dabei zunächst für  $m - n$  Takte  $i$  in Position A. Die Ergebnisse der Summationspunkte werden in das untere Schieberegister geladen. Für  $i > m - n$  werden die Ergebnisse der Summationspunkte nun über Schalter, der jetzt in Stellung B steht, nach außen geführt. In der Zeichnung ist die feste Verdrahtung der Prüfmatrix  $P$  nur angedeutet, während die Verdrahtung der Einheitsmatrix bereits eingezeichnet ist.

**Decodierung:** Die Decodierung der Hamming-Codes erfolgt ebenfalls mit Hilfe eines Schieberegisters. Abbildung 3.6 zeigt die Vorgehensweise. Der empfangene Bitvektor wird zunächst in das Register geschoben. Die Registerinhalte werden dann gemäß der Prüfmatrix auf die Summationspunkte geführt. Die Verdrahtung der Einheitsmatrix ist bereits eingezeichnet. Das an den Summationspunkten berechnete Syndrom wird zur Adressierung der Syndromtabelle genutzt. Diese liefert binär codiert die Fehlerpositionen. Durch eine sich anschließende  $m - n$  zu  $2^{m-n}$  Decodierung können die fehlerhaften Bits im Empfangsvektor direkt invertiert werden.

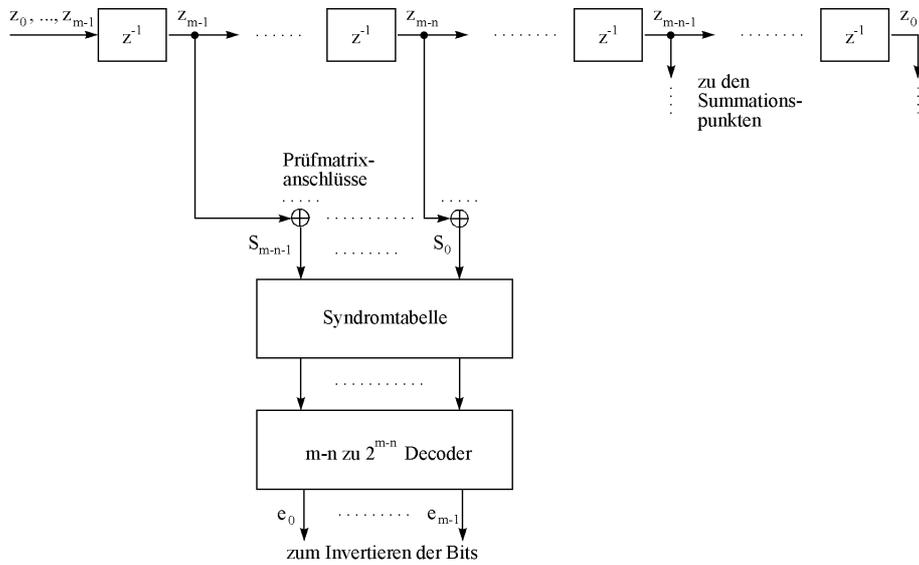


Abbildung 3.6: Hamming-Decodierung

### 3.2.3 BCH-Codes

#### 3.2.3.1 Theorie

BCH-Codes gehören zu den zyklischen Codes. Ein linearer Blockcode heißt zyklisch, wenn durch zyklische Verschiebung eines Codewortes wieder ein neues zulässiges Codewort entsteht. Für die zyklischen Codes ist eine mathematische Beschreibung in Polynomschreibweise von Vorteil:

$$Y(z) = \sum_{i=0}^{m-1} y_i z^i \tag{3.14}$$

Mit  $y_i \in \mathbb{GF}_z$ .

$z$  entstammt dabei nicht den reellen Zahlen sondern einer speziell definierten Menge, die jedoch auch einen Körper darstellen muß. Es gilt  $z \in \mathbb{GF}_z K$ , mit

$$z = \sum_{i=0}^{K-1} z_i q^i \tag{3.15}$$

Darin ist  $z_i \in \mathbb{GF}_z$ . Damit werden auch die Elemente des Körpers selbst in Polynomschreibweise dargestellt.

Für BCH-Codes werden primitive zyklische Codes betrachtet. Sie besitzen eine Codelänge  $m$  von

$$m = 2^K - 1 \quad (3.16)$$

Das Codepolynom ergibt sich durch Multiplikation des Informationspolynoms mit einem Generatorpolynom.

$$Y(z) = X(z) \cdot G(z) \quad (3.17)$$

In dieser Form enthält  $Y(z)$  nicht mehr direkt die Koeffizienten des Informationspolynoms. Der Code ist daher zunächst nicht systematisch. Er kann jedoch durch folgende Manipulationen in eine systematische Form umgewandelt werden. Das Informationspolynom werde zunächst mit  $z^{m-n}$  multipliziert. Das so entstandene Polynom läßt sich in einen durch  $G(z)$  teilbaren Teil  $A(z)$  und einen Rest  $B(z)$  aufspalten.

$$z^{m-n}X(z) = A(z)G(z) + B(z) \quad (3.18)$$

Dies läßt sich auch schreiben als:

$$B(z) + z^{m-n}X(z) = A(z)G(z) \quad (3.19)$$

Da die rechte Seite von (3.19) ein gültiges Codewort darstellt, ist die linke Seite ebenfalls eins. Damit ist die systematische Codeform gefunden: Man schiebe die Koeffizienten des Informationspolynoms um  $m - n$  Bits nach rechts und füge in die leergewordenen Stellen den Rest der Division von  $z^{m-n}X(z) : G(z)$  ein. Diese Operationen lassen sich elegant mittels Schieberegistern ausführen.

Für zyklische Codes ist  $G(z)$  Teiler von  $z^m - 1$  und besitzt die Nullstellen  $\alpha_i$ . Es gilt daher:

$$G(z) = \prod_{i=1}^{m-n} (z - \alpha_i) \quad (3.20)$$

$\alpha_i$  entstammt in dieser Gleichung nicht notwendigerweise aus dem  $\mathbb{GF}_z$ . Dies kann in einer naheliegenden Analogie zu den reellen und komplexen Zahlen gesehen werden, da ein Polynom mit reellen Koeffizienten komplexe Nullstellen haben kann. Es ist daher zur Beschreibung der Nullstellen eine Zahlenbereichserweiterung notwendig. Dies geschieht hier in Form der Erweiterung vom  $\mathbb{GF}_z$  auf das Erweiterungsfeld  $\mathbb{GF}_zK$ . In dieser neuen Menge sind analog zu den komplexen Zahlen neue Rechenoperationen zu definieren, so daß die erweiterte Menge mit ihren Operatoren  $+$  und  $\cdot$  wieder einen Körper darstellt.

Für zwei Elemente  $\alpha, \beta \in \mathbb{GF}_zK$  sind die beiden Rechenoperatoren  $+$  und  $\cdot$  wie folgt definiert:

$$\alpha(q) + \beta(q) = \sum_{i=0}^{m-1} (a_i + b_i)z^i \quad (3.21)$$

$$\alpha(q) \cdot \beta(q) = \left[ \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} (a_i \cdot b_j)z^{i+j} \right]_{\text{mod } Q(q)} \quad (3.22)$$

$$a_i, b_i \in \mathbb{GF}_z \quad (3.23)$$

$$(3.24)$$

$Q(q)$  ist dabei ein primitives Polynom. Sämtliche Elemente  $\alpha_i \in \mathbb{GF}_z K$  bis auf  $\alpha = 0$  können durch Potenzieren des primitiven Elementes  $\alpha = q$  gewonnen werden. Daher:

$$\alpha_i = \alpha^i = q^i \text{ mod } Q(q), \quad i \geq 0 \quad (3.25)$$

Es gilt  $\alpha_0 = \alpha^0 = 1$  und  $\alpha^{2^K-1} = \alpha^0 = 1$ .

Für die primitiven zyklischen Codes gilt, daß sämtliche von Null verschiedenen Elemente des Erweiterungsfeldes  $\mathbb{GF}_z K$  Nullstellen des Polynoms  $P(z) = z^m + 1$  sind. Damit gilt  $P(z = \alpha_i) = 0$ . Woraus seinerseits folgt:

$$P(z) = \prod_{i=1}^m (z - \alpha_i) \quad (3.26)$$

Wiederum analog zu den komplexen Zahlen lassen sich bestimmte Nullstellen  $\alpha_i$ , die dem Erweiterungsfeld  $\mathbb{GF}_z K$  entstammen, zu Polynomen zusammenfassen, deren Koeffizienten Elemente des  $\mathbb{GF}_z$  sind.  $P(z)$  kann daher alternativ geschrieben werden als:

$$P(z) = \prod_{l=1}^L M_l(z) \quad (3.27)$$

$M_l$  wird Minimalpolynom bezüglich  $\alpha_i$  genannt. Es ist das Polynom kleinsten Grades mit Koeffizienten aus dem Grundkörper  $\mathbb{GF}_z$  für das gilt:

$$M_l(z = \alpha_i) = 0 \quad (3.28)$$

Die Indizes sind unterschiedlich, weil zu einem  $M_l(z)$  mehrere  $\alpha_i$  gehören können. Zu einem  $\alpha_i$  gehöre das Minimalpolynom  $M_{\alpha_i}(z)$ .

Mit Hilfe dieser Minimalpolynome läßt sich das Generatorpolynom  $G(z)$  derartig entwickeln, daß es die Nullstellen  $\alpha^1 \dots \alpha^{2e}$  besitzt und gleichzeitig Koeffizienten aus dem  $\mathbb{GF}_z$  enthält. Da mehrere  $\alpha_i$  zu einem  $M_l(z)$  gehören, wird das Generatorpolynom als kleinstes gemeinsames Vielfaches der Minimalpolynome gebildet.

$$G(z) = \text{KGV} \{M_{\alpha_1}, \dots, M_{\alpha_{2e}}\} \quad (3.29)$$

Dadurch besitzt auch das Codepolynom  $Y(z)$  mindestens die Nullstellen  $\alpha^1 \dots \alpha^{2e}$ . Mindestens deswegen, weil ein Minimalpolynom  $M_{\alpha_i}(z)$  bereits eine Nullstelle  $\alpha_j, j > 2e$  enthalten kann. Die Anzahl der tatsächlichen Nullstellen von  $G(z)$  bestimmt den Grad von  $G(z)$  und damit die Zahl der Informationsbits  $n$ , nachdem  $m$  und  $e$  vorgegeben wurde. Es gilt:

$$n = m - \text{Grad} \{G(z)\} \quad (3.30)$$

Wird das Codepolynom gestört, so besitzt das gestörte Polynom nicht mehr

die gleichen Nullstellen wie das richtige Codepolynom. Das heißt, für ein oder mehrere  $\alpha^i, i \leq 2e$  ist es nicht mehr 0.

$$Z(z) = Y(z) + E(z) = E(z) \quad (3.31)$$

$$Z(\alpha_i) = E(\alpha_i) \quad (3.32)$$

Durch Einsetzen der Nullstellen  $\alpha^i$  in das Fehlerpolynom  $E(z) = e_0 + e_1z + \dots + e_{m-1}z^{m-1}$  ergeben sich folgende Ausdrücke:

$$S_i = \sum_{j=0}^{m-1} e_j \alpha^{(i+1)j} \quad 0 \leq i \leq d-2 \quad (3.33)$$

Nimmt man an, daß  $e$  Fehler an den Positionen  $n_1, \dots, n_e$  aufgetreten sind, läßt sich (3.33) schreiben als:

$$S_i = \sum_{j=1}^e e_{n_j} \alpha^{(i+1)n_j} \quad 0 \leq i \leq d-2 \quad (3.34)$$

Dies ist das sogenannte Syndrom-Gleichungssystem. Es besteht aus  $d-1$  Gleichungen für die  $e$  Unbekannten  $e_i$ . Dabei ist  $e$  selbst auch eine Unbekannte. Je nachdem wie die Wahl bei den erkenn- bzw. korrigierbaren Fehlermustern ausfällt, wird für  $e$  eine Obergrenze gesetzt, um (3.34) zu lösen. Ist das Gleichungssystem nicht lösbar, liegt ein mindestens  $e+1$ -facher Fehler vor.

Hiermit ist bereits eine prinzipielle Implementierungsmöglichkeit für die Fehlerkorrektur bei BCH-Codes gegeben. Da die Lösung des Gleichungssystems in dieser Form unpraktikabel ist, ließe es sich auch über eine Tabelle mit den möglichen Syndromkomponenten im Fehlerfall lösen. Die Zahl der Einträge in dieser Tabelle beträgt  $(d-1) \cdot K$  und wird damit schnell sehr groß. Der nächste Abschnitt enthält die Beschreibung eines optimierten Verfahrens, das ohne Tabelle auskommt.

### 3.2.3.2 Implementation

**Codierung:** Im vorigen Abschnitt wurde die mathematische Umwandlung in die systematische Form beschrieben. Abbildung 3.7 zeigt eine effiziente Implementierungsmöglichkeit auf der Basis von rückgekoppelten Schieberegistern. Das Register wird mit Nullen initialisiert. Die beiden Schalter stehen zunächst in Position A.

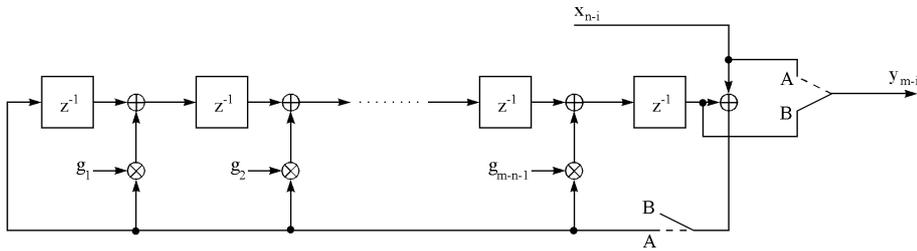


Abbildung 3.7: Generierung eines systematischen BCH-Codes mittels Schieberegister

Für  $n$  Takte  $i$  werden die Informationsbits  $x_{n-i}$  über den Schalter nach außen geführt. Zugleich werden sie in das Schieberegister geladen. Die Rückkopplungsstruktur des Schieberegisters ist durch die Koeffizienten des Generatorpolynoms gegeben. Für  $i > n$  werden beide Schalter auf Position B umgestellt. Dadurch wird nun einerseits das Schieberegister ausgelesen, und andererseits die Rückkopplungsschleife unterbrochen. Dieses Schieberegister ist eine in Hardware gegessene Form des schriftlichen Polynomdivisionsalgorithmus.

**Ausführen der Multiplikation im  $\mathbb{GF}_z K$ :** (3.22) zeigt die Definition der Multiplikation zweier Elemente des  $\mathbb{GF}_z K$ . Sie erfolgt modulo dem primitiven Polynom  $Q(q) = f_0 + f_1 q + \dots + f_K q^K$ . Es existiert ein sehr effizienter Algorithmus auf der Basis von rückgekoppelten Schieberegistern, die eine Implementation von (3.22) auf Hardware-Ebene erlaubt. Abbildung 3.8 zeigt das Verfahren.

Die Speicherzellen werden beim Start mit 0 geladen. Der Rückkopplungsweig zu einer Speicherzelle ist nur dann vorhanden, wenn der entsprechende Koeffizient des primitiven Polynoms ungleich Null ist. Mit jedem Takt werden nacheinander  $b_{K-1} \dots b_0$  an die Multiplikatoren angelegt. Diese Multiplikatoren sind für den Fall der BCH-Codes einfache Und-Gatter. Nach  $K$  Takten enthalten die Speicherzellen die Koeffizienten des gesuchten Produkts  $\alpha(q) \cdot \beta(q)$ .

Für sehr kleine  $K$  kann auch eine Implementation der Multiplikation mit Hilfe einer Tabelle in Betracht kommen.

**Decodierung:** Eine effiziente Implementierungsmöglichkeit ergibt sich, wenn (3.34) in eine andere Form gebracht wird. Diese Form erlaubt eine Berechnung der Fehlerpositionen über ein iteratives mit Schieberegistern operierendes Verfahren. Die Herleitung ist in [5] beschrieben. Das Verfahren benötigt benötigt eine Zwischengröße  $\sigma(z)$  über deren Nullstellen die Fehlerpositionen zu bestimmen sind.

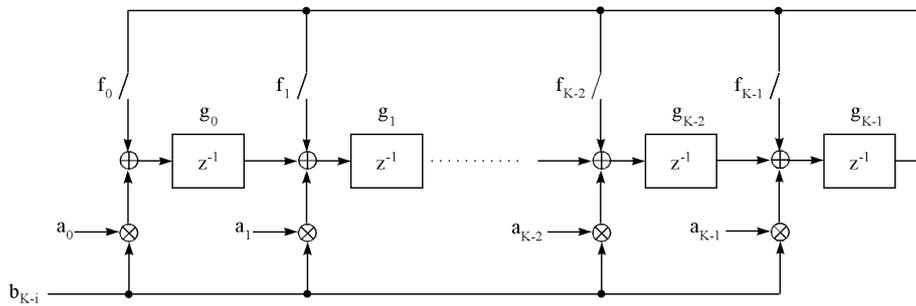


Abbildung 3.8: Multiplikation zweier Körperelemente  $\alpha, \beta \in \mathbb{GF}_z K$  mittels Schieberegister

Die Bestimmung der Fehlerpositionen läßt sich daher in 3 Teilaufgaben zerlegen:

1. Berechnung der Syndromkomponenten aus dem Empfangspolynom  $Z(z)$
2. Berechnung der Zwischengröße  $\sigma(z)$
3. Finden der Nullstellen von  $\sigma(z)$

**zu 1.:** Es gilt folgenden Ausdruck zu berechnen:

$$S_i = \sum_{j=0}^{m-1} z_j \alpha^{(i+1)j} \quad 0 \leq i \leq d-2 \quad (3.35)$$

$$= (\dots (z_{m-1} \alpha^{i+1} + z_{m-2}) \alpha^{i+1} + z_{m-3}) \alpha^{i+1} \dots + z_0) \quad (3.36)$$

Die zweite Zeile bietet einen direkten Implementierungsvorschlag mittels rückgekoppelten Schieberegistern an. In getrennten Pfaden werden die Syndromkomponenten  $S_i$  berechnet. Abbildung 3.9 zeigt die dazu gehörige Anordnung.

**zu 2.:** Die Berechnung der Zwischengröße  $\sigma(z)$  erfolgt iterativ mit Hilfe von 3 Schieberegistern.  $\sigma_0$  ist gleich 1 und braucht daher nicht berechnet zu werden.

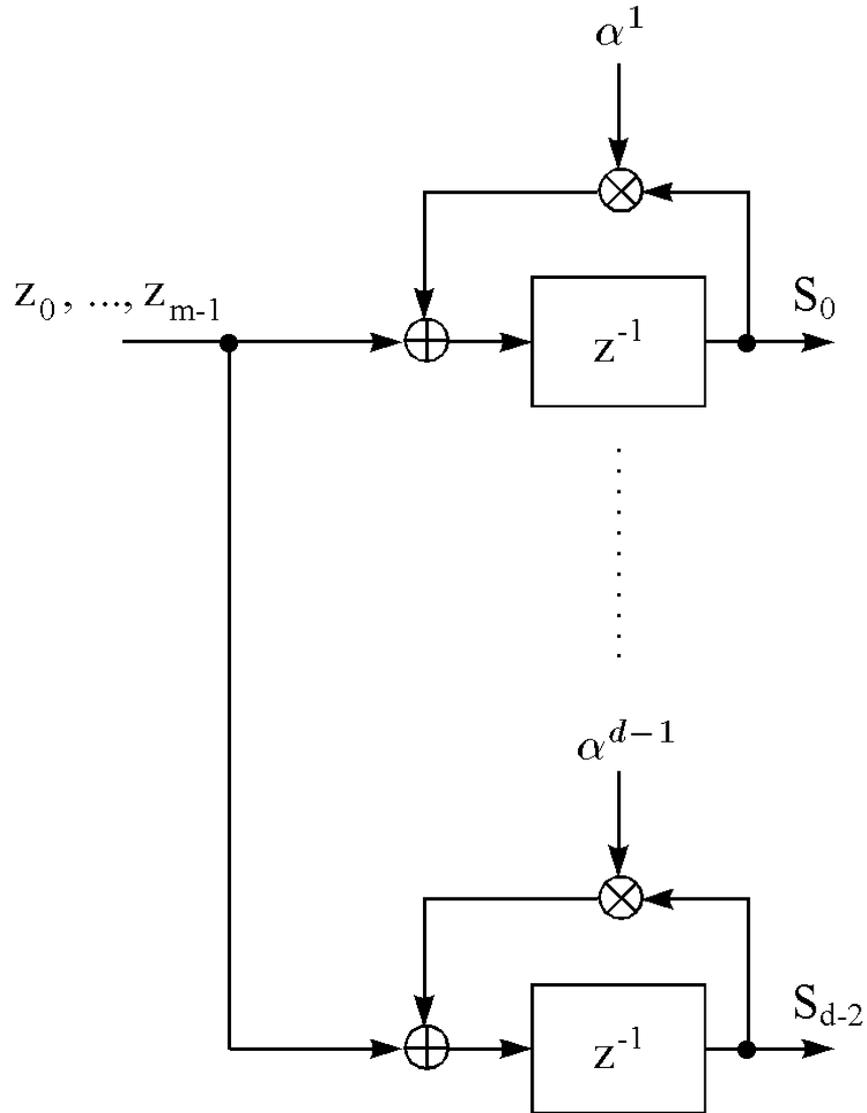


Abbildung 3.9: Syndromberechnung mittels rückgekoppelten Schieberegister

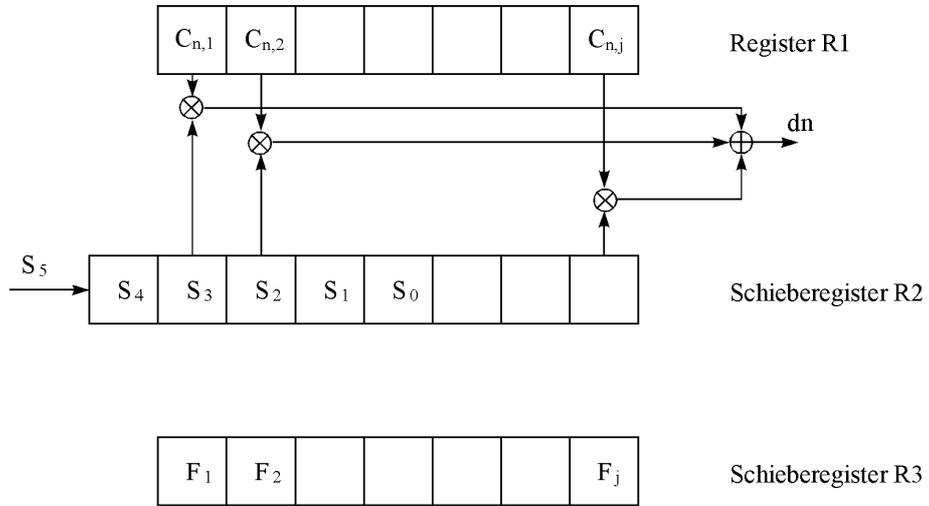


Abbildung 3.10: Registersatz für den Algorithmus zur iterativen Bestimmung von  $\sigma(z)$

Abbildung 3.10 zeigt die Anordnung der Register, Abbildung 3.11 zeigt das Flußdiagramm des Algorithmus zur Bestimmung von  $\sigma(z)$ . Für die Multiplikationen in Abbildung 3.10 ist das Verfahren von Abbildung 3.8 einzusetzen. Register R1 enthält nach dem Abschluß der  $L = d - 2$  Iterationen die Koeffizienten von  $\sigma(z)$  bis auf  $\sigma_0$ . Wird die Länge der Register  $j$  zu  $\lfloor (d-1)/2 \rfloor$  gewählt, können  $j$  Fehler korrigiert werden.

**zu 3.:**  $\sigma(z)$  ist gegeben durch

$$\sigma(z) = \prod_{j=1}^e (1 - \alpha^{n_j} z) \quad (3.37)$$

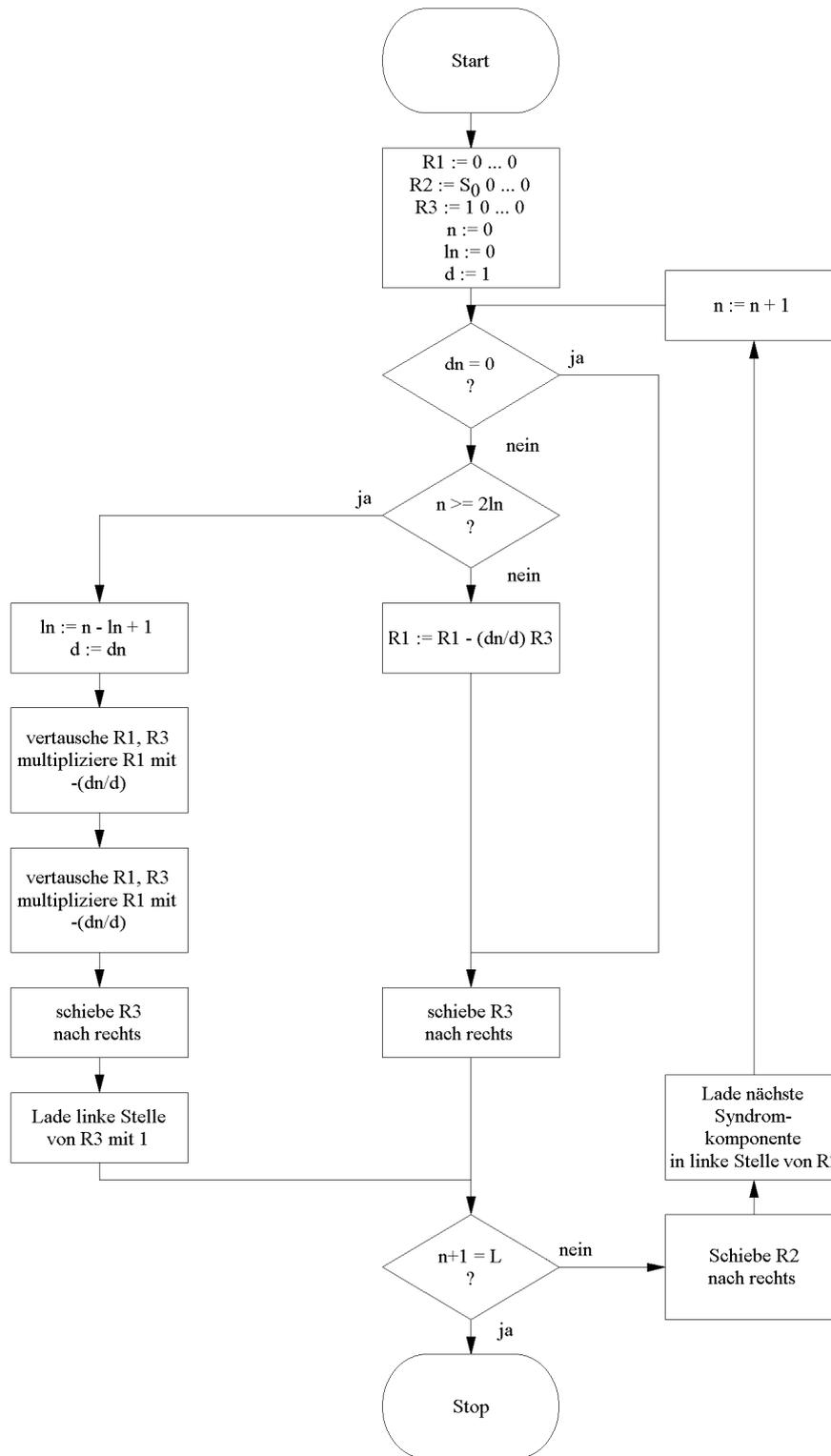
mit  $n_j$  der Fehlerposition des Fehlers  $j$ . Ist  $n$  eine Fehlerposition, so liefert das Einsetzen der Nullstelle  $\alpha^n$ :

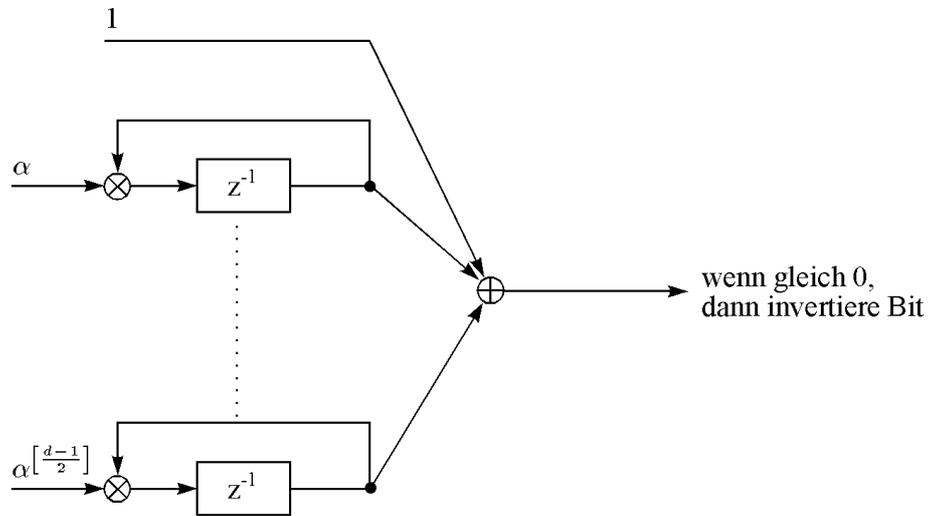
$$\sum_{i=0}^e \sigma_i \alpha^{-ni} = 0 \quad (3.38)$$

Die Berechnung der Summe in (3.38) läßt sich einfach implementieren, da ein  $\sigma_i$  immer nur mit ganz bestimmten Potenzen von  $\alpha$  multipliziert wird. Durch  $e$  parallele Zweige, die jeweils  $\sigma_i \alpha^{-ni}$  rekursiv berechnen, können daher nach Summation über die Zweige und anschließendem Vergleich mit 0 die Fehlerpositionen bestimmt werden. Jeder Takt in Abbildung 3.12 entspricht einer Erhöhung von  $n$  in (3.38) um eins, so daß nacheinander alle Bits des Empfangspolynoms  $Z(z)$  bei Bedarf korrigiert (invertiert) werden können.

### 3.2.4 Nutzung von Erasure-Informationen

Um die Suche nach der Präambel im Kanalrauschen zu unterbinden, muß dem digitalen Empfänger ein Empfangsfeldstärkesignal zur Verfügung gestellt werden. Dieses kann auch dazu dienen, einzelne Bits nach der Datendetektion als unzuverlässig zu kennzeichnen.

Abbildung 3.11: Algorithmus zur iterativen Bestimmung von  $\sigma(z)$

Abbildung 3.12: Nullstellensuche in  $\sigma(z)$ 

Es gilt folgende Ungleichung:

$$2e + l + 1 \leq d_{\min} \quad (3.39)$$

Darin ist  $e$  die Anzahl der Fehler, die an beliebigen unbekanntem Stellen aufgetreten sind, und  $l$  die Anzahl Fehler, die durch eine unter einen Schwellenwert gesunkene Empfangsfeldstärke als unzuverlässig gekennzeichnet sind. Man erkennt, daß für den Fall, daß die möglichen Fehlerpositionen bekannt sind, doppelt so viele Fehler korrigiert werden können.

Der Algorithmus der Erasure-Korrektur läuft dabei folgendermaßen ab:

1. Ersetze alle als unzuverlässig gekennzeichneten Bits durch 0
2. Führe normale Fehlerkorrektur für den so erhaltenen Bitvektor aus
3. Ersetze alle als unzuverlässig gekennzeichneten Bits durch 1
4. Führe normale Fehlerkorrektur für den so erhaltenen Bitvektor aus
5. Wähle den korrigierten Bitvektor, bei dem weniger Korrekturen durchgeführt wurden.

Dadurch wird die Fehlerkorrektur nur unwesentlich komplizierter. Die Syndromtabelle bei den Hamming-Codes muß hier für bis zu zweifachen Fehlern erstellt werden. Dadurch, daß ein Bit an einer festen Position als unzuverlässig gekennzeichnet ist, kann die Mehrdeutigkeit der Syndromtabelle umgangen werden. Der Korrekturalgorithmus muß jedoch für die Erasure-Korrektur mit der doppelten Geschwindigkeit arbeiten.

Für den Fall der sehr starken Störung des Empfangsbitvektors nützt natürlich auch die Erasure-Korrektur nur wenig. Die Erasure-Korrektur ist beispielsweise dann sinnvoll einzusetzen, wenn der Aufwand für eine BCH-Dekodierung gescheut wird, aber dennoch bessere Fehlerkorrektureigenschaften als die der Hamming-Codes erwünscht sind.  $d_{\min}$  beträgt für die Hamming-Codes 3, so daß durch die Erasure-Information 2 Fehler an festen Positionen korrigiert werden können. Das heißt, daß sich die korrigierbare Fehlerzahl pro Hamming-Codeblock verdoppelt hat. Man könnte daher als Grundkorrekturverfahren die Hamming-Codes benutzen und darauf die Erasure-Korrektur aufsetzen.

Ein Einbruch der Empfangsfeldstärke ließe sich auch dazu nutzen, einen Teilblock als unzuverlässig bzw. fehlerhaft zu kennzeichnen. Dieser Teilblock könnte dann über die Zeitdiversität bei einem späteren Empfangszyklus gewonnen werden.

# Literaturverzeichnis

- [1] John E. Proakis: "Digital Communications", McGraw-Hill, USA 1989, ISBN 0-07-050937-9
- [2] Ralf Mehlan "Zwischenbericht: MSK-Transceiver mit digitaler Mischung", Zwischenbericht VI, IS2, RWTH Aachen, Juni 1993
- [3] R. Mehlan, U. Keisinger "Untersuchungen zur Implementierung eines MSK-Transceivers", Zwischenbericht V, IS2, RWTH Aachen, Juni 1993
- [4] H. Meyr, R. Mehlan "Efficient Preamble Design for DMSK Modulation", IS2, RWTH Aachen
- [5] Robert G. Gallager "Information Theorie and Reliable Communication", J. Wiley, 1968
- [6] R. Mehlan "Studie eines 64 GHz Transceivers zur Fahrzeug-Fahrzeug Kommunikation", IS2, RWTH Aachen, Mai 1992
- [7] U. Lambrett "Ein integrales Systemkonzept für Fahrzeug-Fahrzeug Kommunikation", IS2, RWTH Aachen, Abschlußbericht Juni 1993
- [8] Aldo N. N'Andrea, U. Mengali and R. Reggiannini: "A digital approach to clock recovery in Generalized Minimum Shift Keying", *IEEE Transactions on Vehicular Technology*, Vol. 39, No. 3, pp. 227-234, August 1990.
- [9] John H. Lodge, Michael L. Moher, Stewart N. Crozier "A Comparison of Data Modulation Techniques for Land Mobile Satellite Channels" *IEEE Transactions on Vehicular Technology*, Vol. VT-36, No.1, February 1987