Prof. Dr.-Ing. Dr. h. c. J. Becker

Digitaltechnik

1. Tutorium - Lösung

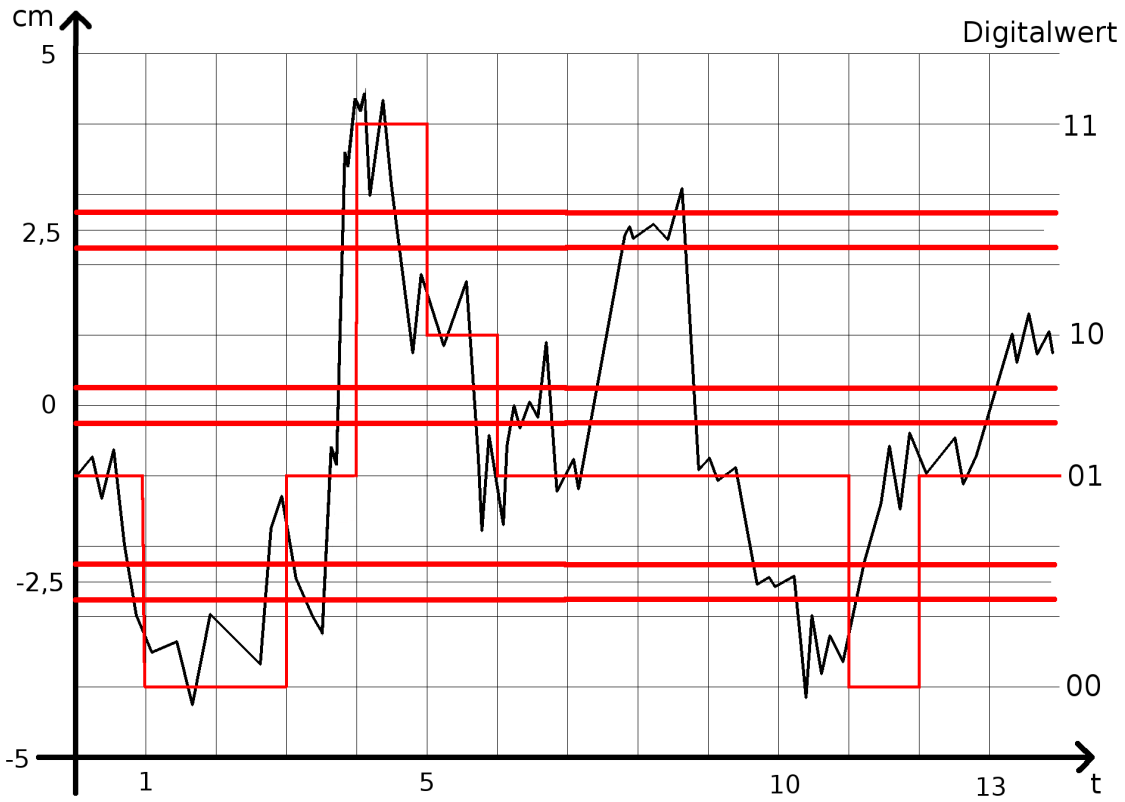
Institut für Technik der Informationsverarbeitung (ITIV), Karlsruher Institut für Technologie (KIT)

Aufgabe 1: Digitalisierung

Für verschiedene Studien soll der Verlauf von Erdbeben elektronisch zur Verfügung gestellt werden. Hierfür soll die Ausgabe eines Seismographen digitalisiert werden.
In der folgenden Tabelle wurde jedem Intervall ein Signalwert zugeordnet.

Intervall	Signalwert	definierter Bereich von - bis
-5 bis -2,5	00	-5 bis -2,75
-2,5 bis 0	01	-2,25 bis -0,25
0 bis 2,5	10	0,25 bis 2,25
2,5 bis 5	11	2,75 bis 5

1.1 Um eine „weiche“ Diskrimination zu realisieren, sollen zwischen den Intervallen undefinierte Bereiche eingefügt werden. Berechnen Sie die Größe dieser undefinierten Bereiche. Diese sollen 20% des Werteintervalls betragen. Zeichnen Sie die undefinierten Bereiche in den Verlauf des Seismographen ein.



© ITIV 2015

1.2 Digitalisieren Sie das Signal Wert- und Zeitdiskret.

1.3 Wie viel Bit werden benötigt um 100 Abtastpunkte zu speichern?
2 Bit pro Wert → 200 Bit

Aufgabe 2: Paritätsprüfung

2.1 Eine Nachricht im 7-Bit-ASCII-Code soll über eine fehleranfällige Leitung übertragen werden. Es wird eine Blocksicherung mit doppelter Paritätsprüfung vorgenommen. Sender und Empfänger haben sich auf ungerade Parität geeinigt. Tragen Sie die Zeichen in die Tabelle ein, die vom Empfänger gelesen werden!

Zeichen	Codewort							Parität
r	1	1	1	0	0	1	0	1
y	1	1	1	1	0	0	1	1
c	1	1	0	0	0	1	1	1
h	1	1	0	1	0	0	0	0
d	1	1	0	0	1	0	0	1
i	1	1	0	1	0	0	1	1
g	1	1	0	0	1	1	1	0
Prüfwort	0	0	1	0	1	0	1	0

Anmerkung: das ‚y' und ‚d' wurden fälschlicherweise empfangen. Auf Grund des Paritätsfehlers kann der Empfänger der Nachricht schlussfolgern, dass diese Buchstaben wohl nicht gesendet wurden.

2.2 Anstelle des ASCII-Codes wird nun ein (4-aus-7)-Code ohne Parität oder Prüfwort verwendet. Wie viele verschiedene Zeichen können nun codiert werden?
$N = \binom{7}{4} = \frac{7!}{4!(7-4)!} = \frac{7!}{4!\cdot 3!} = \frac{7\cdot 6\cdot 5}{3\cdot 2\cdot 1} = \frac{210}{6} = 35$

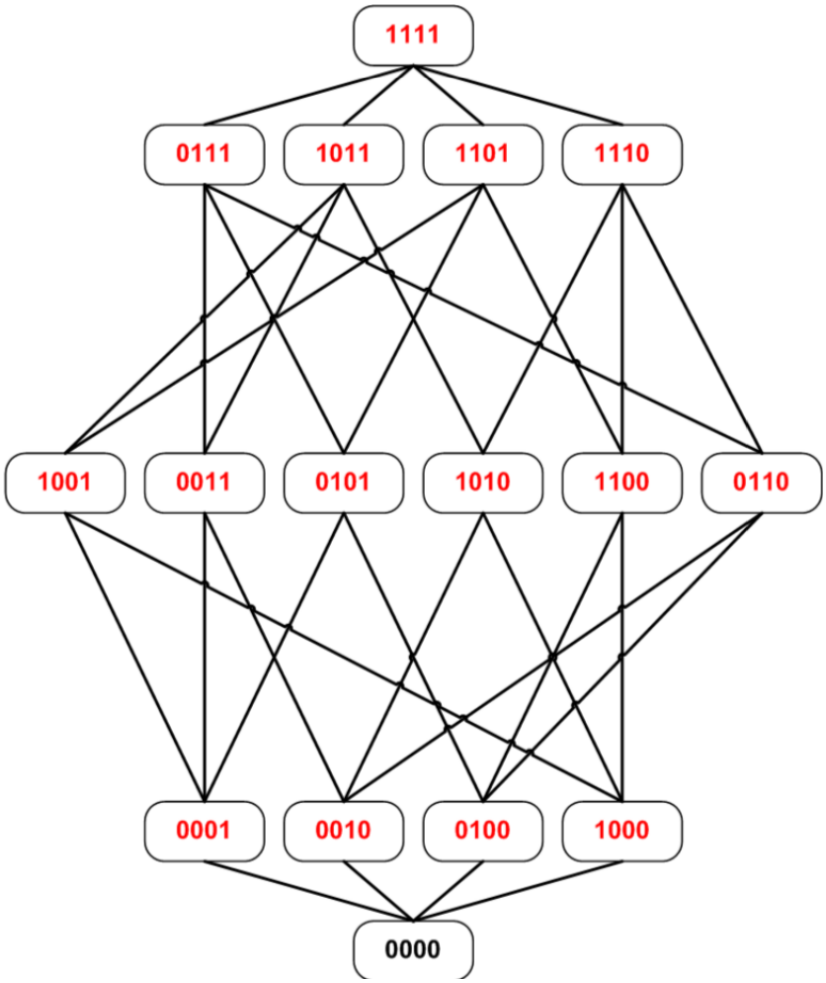
2.3 Es werden die gleichen Codewörter (Bitfolgen) wie in Aufgabe 2.1 gesendet und empfangen. Die Paritätsbits und das Prüfwort werden jedoch nicht gesendet. Können Einzelfehler erkannt werden, wenn ein (4-aus-7)-Code verwendet wird?

Ja. Mit 7 Bit können 128 Codeworte gebildet werden. Bei Verwendung des (4-aus-7)-Codes gibt es 35 gültige Codewörter (siehe 2.1). Diese haben jeweils eine minimale Hammingdistanz von mindestens $HD_{min} = 2$ zueinander (wird eine der vier '1' innerhalb des Codewortes umsortiert, ändern sich immer 2 Binärstellen des Codewortes!!).
Wenn ja: welche Codewörter aus Aufgabe 2.1 werden als fehlerhaft erkannt wenn man den (4-aus-*)-Code zugrunde legt? Wenn nein: was muss getan werden, um eine Fehlererkennung zu ermöglichen?
1. Zeichen: r – nicht fehlerhaft (4 Einsen – gültiges Codewort)
2. Zeichen: y – fehlerhaft (5 Einsen – ungültiges Codewort)
3. Zeichen: c – nicht fehlerhaft (4 Einsen – gültiges Codewort)
4. Zeichen: h – fehlerhaft (3 Einsen – ungültiges Codewort)
5. Zeichen: d – fehlerhaft (3 Einsen – ungültiges Codewort)
6. Zeichen: i – nicht fehlerhaft (4 Einsen – gültiges Codewort)
7. Zeichen: g – fehlerhaft (5 Einsen – ungültiges Codewort)

LSB	MSB							
Binär	000	001	010	011	100	101	110	111
	Steuerzeichen				Großbuchstaben		Kleinbuchstaben	
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	„	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	´	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

Abbildung 1: ASCII-Tabelle

Aufgabe 3: Hammingdistanz



Die vorstehende Abbildung soll die Nachbarschaftsbeziehungen für einen Code mit vier Binärstellen darstellen. Die Hammingdistanz zwischen zwei benachbarten Knoten soll HD=1 sein.
Das Thema Hammingdistanz wird in den Folien der 4. Vorlesung „Codierung“ sehr anschaulich erklärt.

3.1 Vervollständigen Sie das Diagramm.
Wegen der Nachbarschaftsbeziehungen ist die Anordnung NICHT beliebig.
Wie viele Binärstellen werden mindestens benötigt, um 8 Codewörter darstellen zu können? (Rechnung)
$\lceil ld\ anzahlcodewoerter \rceil = \lceil ld8 \rceil = \lceil \frac{log8}{log2} \rceil = 3$

3.2 Wie viele Codewörter können insgesamt mit 4 Binärstellen codiert werden?
$\#CW = 2^N = 2^4 = 16$, kann auch direkt aus dem ausgefüllten Diagramm abgelesen werden.

3.3 Wie viele Codewörter können maximal mit 4 Binärstellen codiert werden, wenn Einzelbitfehler erkannt werden sollen? Zeigen Sie dies am Diagramm.

Zur Fehlererkennung muss die Hammingdistanz mindestens $HD_{min} = 2$ sein.
Durch die Hammingdistanz von 2 kann nur jede 2. Ebene der Nachbarschaftsbeziehun-

gen benutzt werden. Damit können 8 Codewörter dargestellt werden.
1. Möglichkeit:
Ebene 1: 0000
Ebene 3: 1001, 0011, 0101, 1010, 1100, 0110
Ebene 5: 1111
2. Möglichkeit:
Ebene 2: 0001, 0010, 0100, 1000
Ebene 4: 0111, 1011, 1101, 1110

3.4 Wie viele Codewörter können mit 4 Binärstellen codiert werden, wenn die minimale Hammingdistanz 3 betragen soll? Was ermöglicht eine minimale Hammingdistanz von 3?

2 Mögliche Codewörter (Codewörter auf derselben Ebene haben eine Hammingdistanz von 2);
Ermöglicht die Korrektur von ein-Bit Fehlern oder Erkennung von zwei-Bit Fehlern.

3.5 Was kann in einer Codierung mit 4 Binärstellen mit einer minimalen Hammingdistanz von 4 erreicht werden? Welche möglichen gültigen Codewörter gibt es?

Variante A: Man kann Ein-Bit-Fehler korrigieren und Zwei-Bit-Fehler erkennen.
Variante B: Drei-Bit-Fehler können erkannt, aber nicht korrigiert werden.

0000, 1111	1000, 0111	1100, 0011	1110, 0001
0010, 1101	0100, 1011	0101, 1010	0110, 1001