Prof. Dr.-Ing. J. Becker

Digitaltechnik

1. Tutorium - Lösung

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

1. Aufgabe: Digitalisierung

Mit Hilfe der unten dargestellten Drehscheibe soll ein Drehwinkel digitalisiert werden. Die Scheibe ist in 16 Sektoren mit jeweils vier Feldern eingeteilt. Vier Schleifkontakte stellen fest, ob ein Feld metallisch beschichtet ist oder nicht. Entsprechend melden sie das Signal 1 oder 0 zurück.

In der folgenden Tabelle wurde jedem Sektor ein Signalwert zugeordnet. Das am weitesten rechts stehende Signalwert-Bit sei dem *inneren* Schleifkontakt zugeordnet.

Intervall	Signalwert	undef. Bereich	Intervall	Signalwert	undef. Bereich
0°-22,5°	0000	3,82°	180°-202,5°	1100	3,82°
22,5°-45°	0001	2,86°	202,5°-225°	1101	2,86°
45°-67,5°	0011	3,82°	225°-247,5°	1111	3,82°
67,5°-90°	0010	2,29°	247,5°-270°	1110	2,29°
90°-112,5°	0110	3,82°	270°-292,5°	1010	3,82°
112,5°-135°	0111	2,86°	292,5°-315°	1011	2,86°
135°-157,5°	0101	3,82°	315°-337,5°	1001	3,82°
157,5°-180°	0100	1,91°	337,5°-360°	1000	1,91°

1.1 Die Schleifkontakte liegen auf den Radien 3, 4, 5 und 6cm. An der Grenze zweier Sektoren mit gleichem Radius und unterschiedlichem Digitalwert soll jeweils ein undefinierter Bereich mit der Toleranz ±1mm in tangentialer Richtung eingerichtet werden. Ermitteln Sie zunächst für jeden Radius/ Schleifkontakt den Winkelgrad dieses undefinierten Bereichs. (*Hinweis*: Zur Umrechnung der gegebenen Toleranz in das Gradmaß kann ein Dreisatz hilfreich sein, der Bogenlänge und Winkelgrad in Relation zum Kreisumfang setzt.)
Tragen Sie anschließend für alle 16 Intervallgrenzen den daraus resultierenden undefinierten Winkelbereich, also den Bereich, in dem mindestens ein Schleifkontakt einen undefinierten Wert misst, im Gradmaß in obige Tabelle ein. (*Hinweis*: zur Veranschaulichung, welche der vier Schleifkontakte sich an welchen Sektorengrenzen in einem undefinierten Bereich befinden, ist es ratsam, die Signalwerte explizit in die Zeichnung einzutragen)

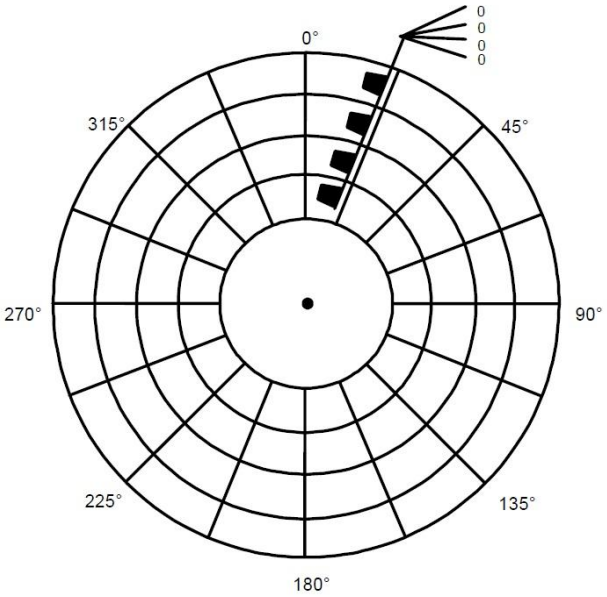
© ITIV

Die undefinierten Bereiche liegen an den Intervallgrenzen. Ihr Wirkungsbereich lässt sich annähern, indem man die Toleranzbreite zum Kreisumfang in Beziehung setzt:

$$\frac{Toleranz}{Umfang} \approx \frac{Winkelbereich}{360}$$

Radius	Umfang	Winkelbereich
3 cm	18,8 cm	3,82°
4 cm	25,1 cm	2,86°
5 cm	31,4 cm	2,29°
6 cm	37,7 cm	1,91°

1.2 Wie viele Schleifkontakte werden benötigt, wenn die Winkelauflösung auf 1° genau erfolgen soll? Welches Problem würde sich dabei zusätzlich ergeben?



Zur Auflösung auf 1° genau benötigt man 360 Sektoren und $\lceil ld360 \rceil = 9$ Schleifenkontakte. Die Codierung mit einem *zyklischen* Gray-Code, der mit 0 0000 0000 beginnt, ist nicht möglich. Abhilfe: Beginn der Codierung verschieben.

2. Aufgabe: Paritätsprüfung

2.1 Bei der Übertragung einer Nachricht im 7-Bit-ASCII-Code über eine fehleranfällige mit einer Blocksicherung mit doppelter Paritätsprüfung wurde nachfolgender Block empfangen. Sender und Empfänger haben sich auf ungerade Parität geeinigt. Geben Sie die Zeichen an, 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	0
c	1	1	0	0	0	1	1	1
h	1	1	0	1	0	0	0	1
d	1	1	0	0	1	0	0	0
i	1	1	0	1	0	0	1	1
g	1	1	0	0	1	1	1	1
Prüfwort	0	0	1	0	1	0	1	0

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

$$N = \binom{7}{4} = \frac{7!}{4!(7-4)!} = \frac{7!}{4! \cdot 3!} = \frac{7 \cdot 6 \cdot 5}{3 \cdot 2 \cdot 1} = \frac{210}{6} = 35$$

2.2 Anstelle des ASCII-Codes wird nun ein 4-aus-7-Code ohne Parität oder Prüfwort verwendet. Wie viele verschiedene Zeichen können nun codiert werden?

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 H=2 zueinander (wird eine der vier '1' innerhalb des Codewortes umsortiert, ändern sich immer 2 Binärstellen des Codewortes!!).

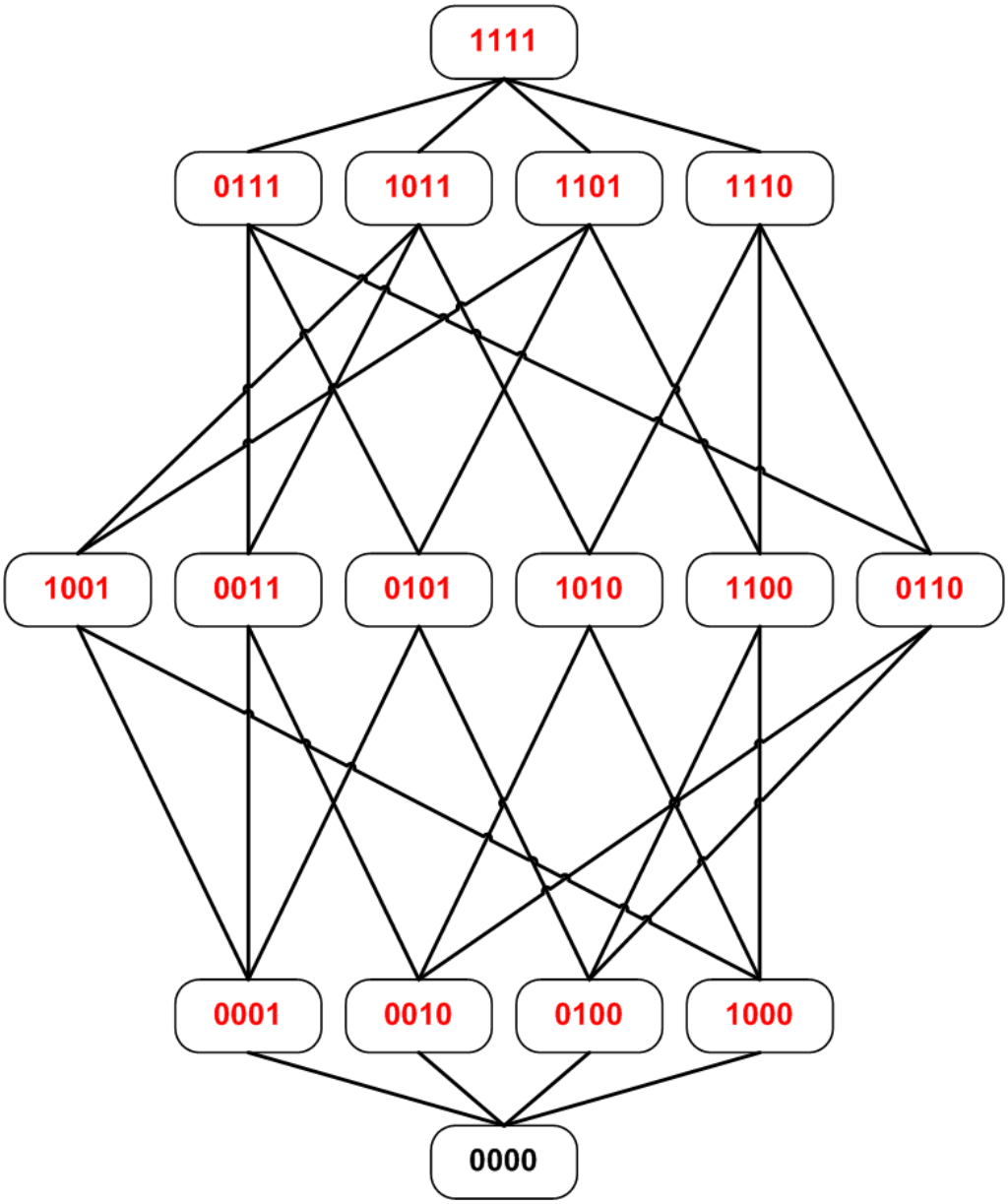
Wenn ja: welche Codewörter aus Aufgabe 2.1 werden als fehlerhaft erkannt wenn man den (4 aus 7) Code zugrunde legt? Wenn nein: was muss getan werden, um eine Fehlererkennung zu ermöglichen?

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)

3. Aufgabe:

Die folgende Abbildung soll die Nachbarschaftsbeziehungen für einen Code mit vier Binärstellen darstellen. Die Hammingdistanz zwischen zwei benachbarten Knoten soll HD=1 sein.



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)

$$N = \lceil ld\,\#codewörter \rceil = \lceil ld\,8 \rceil = \left\lceil \frac{\log 8}{\log 2} \right\rceil = 3$$

3.2 Wie viele Codewörter können insgesamt mit 4 Binärstellen kodiert 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 kodiert werden, wenn Einzelbitfehler erkannt werden sollen. Zeigen Sie dies am Diagramm.

Zur Fehlererkennung muss die Hammingdistanz mindestens H=2 sein.

Durch die Hammingdistanz von 2 kann nur jede 2. Ebene der Nachbarschaftsbeziehungen 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, 111