

Allgemeine Hinweise: Versuchen Sie beim Lösen der Beispiele *keine elektronischen Hilfsmittel* zu verwenden – beim Test werden Sie diese nicht zur Verfügung haben.

Damit ein Beispiel anerkannt wird, muss ein Lösungsweg erkennbar sein und es müssen alle enthaltenen Teilaufgaben gelöst sein. Ein korrektes Endergebnis ist nicht zwingend erforderlich!

Deadline für das Ankreuzen und Hochladen der Lösungen in TUWEL: Dienstag, 27.10.2015, 13:00 Uhr (Toleranzzeit ohne Gewähr, verspätete Abgaben per Email werden ausnahmslos nicht akzeptiert!)

Aufgabe 1: Binäre Gleitpunkt-Arithmetik – Addition & Subtraktion

Gegeben sind die Zahlen $A = (0.0014BB2A8)_{16}$ und $B = (-0.025764)_8$.

Es gilt folgendes Gleitpunktformat:

$\mathbb{F}(2, 11, -14, 15, true)$ mit Formatbreite 16 Bit und *impliziter* Darstellung des ersten Bits. Mit Ausnahme der kleineren Formatbreite ist dieses Gleitpunktformat analog zum IEEE 754 *Single Precision*-Format aufgebaut.

- a) Stellen Sie A und B in diesem Gleitpunktformat dar. Verwenden Sie Guard- und Round-Digit sowie das Sticky-Bit zur Vermeidung von numerischen Ungenauigkeiten (vgl. *Informatik Grundlagen*, 5. Auflage, Kapitel 8.6.4). Runden Sie mittels *round to nearest* zusammen mit *round away from zero*.

- b) Berechnen Sie anschließend $A + B$ sowie $B - A$ und stellen Sie das Ergebnis wieder als Gleitpunktzahl in dem angegebenen Format dar. Runden Sie die Ergebnisse wieder mittels *round to nearest* in Kombination mit *round away from zero*.

Aufgabe 2: Binäre Gleitpunkt-Arithmetik – Multiplikation & Division

Gegeben sind die folgenden, im 16 Bit-Gleitpunktformat aus Aufgabe 1 codierten Zahlen:

$$A = 1\ 10001\ 0010110111$$

$$B = 0\ 00101\ 0010011000$$

$$C = 1\ 10011\ 1110110000$$

Führen Sie die nachfolgenden Berechnungen durch. Verwenden Sie Guard- und Round-Digit sowie das Sticky-Bit zur Vermeidung von numerischen Ungenauigkeiten. Runden Sie mittels *round to nearest* zusammen mit *round to even*.

Hinweis: Beachten Sie das implizite erste Bit!

a) $A * B$

b) $\frac{A}{C}$

Aufgabe 3: Binäre Gleitpunkt-Arithmetik – Sonderfälle

Gegeben sind die folgenden, im 16 Bit-Gleitpunktformat aus Aufgabe 1 codierten Zahlen:

$$\begin{aligned}A &= 0\ 00011\ 0100000000 \\B &= 0\ 01010\ 1110011101 \\C &= 1\ 11110\ 0000000000 \\D &= 0\ 00110\ 1000101100\end{aligned}$$

Führen Sie mit den Zahlen folgende Berechnungen durch und codieren Sie das Ergebnis jeweils im angegebenen Gleitpunktformat. Runden Sie mittels *round toward plus infinity* (= gerichtetes Aufrunden)!

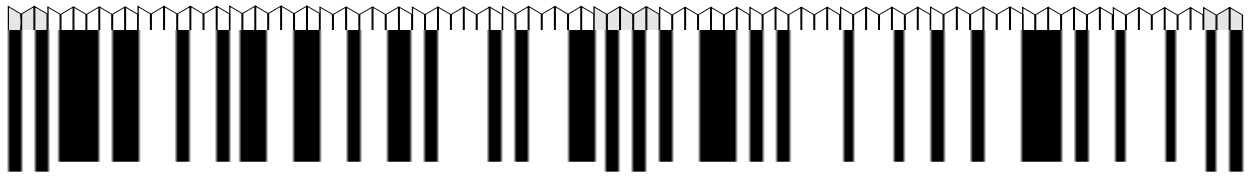
a) $A * B$

b) $B + D$

c) A/C

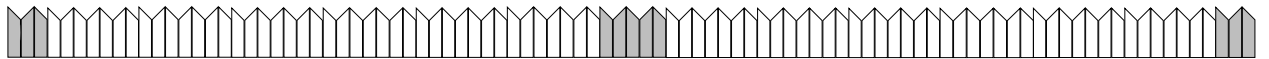
Aufgabe 4: EAN-13-Code

- a) Decodieren Sie die nachfolgende EAN!



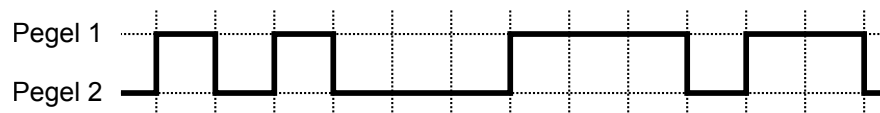
- b) Codieren Sie die EAN 7 235953 52525! Berechnen Sie hierzu die Prüfziffer und tragen Sie den resultierenden Code in den vorgedruckten Raster ein.

Hinweis: Rand- und Trennzeichen sind grau hinterlegt.

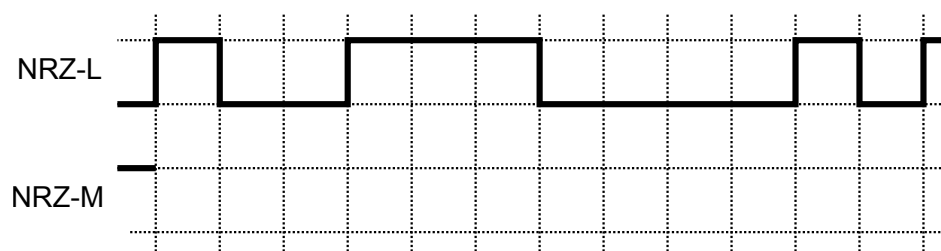


Aufgabe 5: NRZ-Code

Gegeben ist der nachfolgende Signalverlauf mit Pegel 1 (*high*) und Pegel 2 (*low*).



- a) Interpretieren Sie den Signalverlauf in NRZ-L-Codierung und geben Sie die decodierte 0/1-Folge an!
- b) Interpretieren Sie den Signalverlauf in NRZ-S-Codierung und geben Sie die decodierte 0/1-Folge an! Gehen Sie davon aus, dass die Folge bei Pegel *low* mit Wert 0 startet.
- c) Zeichnen Sie zum nachfolgend gegebenen Signalverlauf in NRZ-L-Codierung darunter den entsprechenden Signalverlauf in NRZ-M-Codierung! Gehen Sie davon aus, dass NRZ-M mit Pegel *high* startet.



Aufgabe 6: CRC-Code

a) Sie sollen das ASCII-Zeichen C mittels CRC-CCITT absichern. Welche Bitfolge ergibt das?

b) Sie haben ein mit CRC-12 codiertes Zeichen als Bitfolge '00 0011 1011 0001 1000' empfangen. Begründen Sie, ob ein Fehler vorliegt oder nicht!

Aufgabe 7: Hamming-Distanz

Gegeben ist ein Code mit fünf Codewörtern: 0011001, 0010101, 1101011, 0001111 und 1010100

a) Berechnen Sie die Hamming-Distanz zwischen den einzelnen Codewörtern und vervollständigen Sie die nachfolgende Distanz-Matrix!

	0011001	0010101	1101011	0001111	1010100
0011001					
0010101					
1101011					
0001111					
1010100					

b) Geben Sie den Hamming-Abstand D des Codes an!

c) Wie viele Bits braucht man mindestens, um einen Code für sechs Codewörter zu entwerfen, der einen Hamming-Abstand von $D = 2$ aufweist?

