Embedded Systeme 1 & 2

Klassen T-2/I-2 // 2018-2019

a.14 - Numerische Verarbeitung der Zahlen

Übung Nr. 1

Übungen zu den ganzen Zahlen ohne Vorzeichen.

a) Wandeln Sie die folgenden Zahlen in Binärzahlen um:

- 125 (Basis 10)
- 0377 (Basis 8)
- 0xADE1 (Basis 16)

b) Wandeln Sie die folgenden Zahlen in Dezimalzahlen um:

- 0b10011000 (Basis 2)
- 0177 (Basis 8)
- 0x25E1 (Basis 16)

c) Addieren Sie die folgenden Binärzahlen und geben Sie den Zustand der Flags C und Z an:

- 0b10011000 + 0b10011000
- 0b11111101 + 0b00000011
- 0b00011000 + 0b10011100

d) Subtrahieren Sie die folgenden Binärzahlen und geben Sie den Zustand der Flags C und Z an:

- 0b10011000 - 0b10011000
- 0b11111101 - 0b00000011
- 0b00011000 - 0b10011100

e) Geben Sie den Zustand der Flags C und Z für die folgenden Vergleichsoperationen an:

- cmp 125, 128
- cmp 77, 26
- cmp 254, 254
- cmp 255, 0

Übung Nr. 2

Übungen zu den ganzen Zahlen mit Vorzeichen.

a) Wandeln Sie die folgenden Zahlen in Binärzahlen um und geben Sie den Zustand des Flags N an:

- -125 (Basis 10)
- 0271 (Basis 8)
- 0x50F1 (Basis 16)

b) Wandeln Sie die folgenden Zahlen in Dezimalzahlen um:

- 0b10011000 (Basis 2)
- 0177 (Basis 8)
- 0x85E1 (Basis 16)

c) Addieren Sie die folgenden Binärzahlen und geben Sie den Zustand der Flags V und N und Z an:

- 0b10011000 + 0b10011000
- 0b11111101 + 0b00000011
- 0b00011000 + 0b10011100

d) Subtrahieren Sie die folgenden Binärzahlen und geben Sie den Zustand der Flags V und N und Z an:

- 0b10011000 - 0b10011000
- 0b11111101 - 0b00000011
- 0b00011000 - 0b10011100

e) Geben Sie den Zustand der Flags V und N und Z für die folgenden Vergleichsoperationen an:

- cmp 127, -125
- cmp 77, -26
- cmp -30, -34
- cmp 55, 66

Übung Nr. 3

Sagen Sie den Zustand der Flags Z, C, N und V und das Ergebnis voraus, das im Register R2 nach der Ausführung der folgenden Assemblerbefehle enthalten sein wird:
(Anmerkung: Wir gehen davon aus, dass der µP in der Lage ist, 8-Bit-Worte zu verarbeiten

a)
```
ldr         r2, =128
ldr         r1, =-128
cmp         r2, r1
```

Z=___ C=___ N=___ V=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

b)
```
ldr         r0, =64
ldr         r1, =-128
adds        r2, r0, r1
```

Z=___ C=___ N=___ V=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

c)
```
ldr         r0, =228
ldr         r1, =128
subs        r2, r0, r1
```

Z=___ C=___ N=___ V=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

d)
```
ldr         r0, =240
ldr         r1, =-16
subs        r2, r0, r1
```

Z=___ C=___ N=___ V=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

e)
```
ldr         r2, =0
ldr         r1, =0
cmp         r2, r1
```

Z=___ C=___ N=___ V=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

Übung Nr. 4

Betrachten Sie die nachstehenden 5 Assemblercodes. Definieren Sie für jeden von ihnen den Zustand der Flags N,Z,V,C und interpretieren Sie das Ergebnis des Wertes mit und ohne Vorzeichen.
(Anmerkung: Wir gehen davon aus, dass der µP in der Lage ist, 8-Bit-Worte zu verarbeiten

a)
```
ldr   r0, =-7
ldr   r1, =249
adds  r2, r0, r1
```

N=___ Z=___ V=___ C=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

b)
```
ldr   r0, =248
ldr   r1, =-128
adds  r2, r0, r1
```

N=___ Z=___ V=___ C=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

c)
```
ldr   r0, =128
ldr   r1, =0
adds  r2, r0, r1
```

N=___ Z=___ V=___ C=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

d)
```
ldr   r0, =62
ldr   r1, =200
subs  r2, r0, r1
```

N=___ Z=___ V=___ C=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

e)
```
ldr   r0, =-8
ldr   r1, =-96
subs  r2, r0, r1
```

N=___ Z=___ V=___ C=___ R2 (mit Vorzeichen)=_____R2(ohne Vorzeichen)=_____

Übung Nr. 5

Stellen Sie die folgenden reellen Werte als Hexadezimalzahlen mit 32 Bit (einfache Genauigkeit) dar :

- 1'048'576 : 0x _____ _____ _____ _____
- 2048 : 0x _____ _____ _____ _____
- 55.75 : 0x _____ _____ _____ _____
- 5 / 4096 : 0x _____ _____ _____ _____
- 25 / 2 : 0x _____ _____ _____ _____