



Embedded Systeme 1 & 2

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

a.11 - Befehlssatz des ARM- μ Prozessors

Übung 1

Transkribieren Sie die folgenden C-Algorithmen in Assembler:

a)

```
int i = 0;
int j = 0;
for (i=0; i<100; i++) j += i;
```

b)

```
int i = 100;
unsigned int j = 2;
while (i>0) {
    j *= j; i--;
}
```

c)

```
int i = 8;
unsigned int j = 0x7000;
do {
    j = j >> 1; i--;
} while (i > 0);
```



d)

```
int i;  
int j = 0;  
switch(i) {  
    case 0: j=11; break;  
    case 1: j=8; break;  
    case 2: j=25; break;  
    case 3: j=99; break;  
    case 4: j=33; break;  
    default: j=-1; break;  
}
```

e)

```
int i;  
int j=0;  
if (i>0) j = i % 16;  
else if (i==0) j = 0xaa;  
else j = -i / 16;
```

Übung 2

Codieren Sie den folgenden Algorithmus in ARM-Assembler

```
short  toto;    // 2 bytes  
char   i;       // 1 byte  
  
toto = 20;  
i = 0;  
while (i<7) {  
    toto += toto;  
    toto -= i;  
    i++;  
}
```



Übung 3

Codieren Sie den folgenden Algorithmus in ARM-Assembler

```
k := 0;
switch (i) {
    case 0 : k := -20; break;
    case 2 : k := -2; break;
    case 4 : k := 0; break;
    case 5 : k := 5; break;
    case 8 : k := 24; break;
}
```

Übung 4

Codieren Sie in ARM-Assembler die Paritäts-Behandlung, die die vertikale Parität einer bestimmten Anzahl von Bytes n berechnet.

C-Code der Funktion:

```
#define LENGTH 4
char chaine[LENGTH+1] = {0xC4,0xBF,0x1F,0xAD,0x00};
main() {
    char n = 0;
    while (n < LENGTH) {
        chaine[LENGTH] = chaine[LENGTH] ^ chaine[n] ;
        n++;
    }
}
```

Beispiel:

1	1	0	0	0	1	0	0
1	0	1	1	1	1	1	1
0	0	0	1	1	1	1	1
1	0	1	0	1	1	0	1
1	1	0	0	1	0	0	1



Übung 5

Für den Assembler-Code unten:

```
res:      .short    1
var1:     .byte     67

main:     ldr        r0, =var1
          ldrb       r4, [r0]
          lsl        r2, r4, #3
          lsl        r4, #1
          add        r4, r2
          ldr        r0, =res
          strh       r4, [r0]
```

- A. Welcher Wert wird nach dem Starten des Codes in der Variablen "res" gespeichert?
- B. Erklären Sie in einigen Zeilen die Funktion dieses Codes.
- C. Welche Werte befinden sich als Ergebnis in den Registern R2 (in dezimal, hexadezimal und binär) ?
- D. Welche Werte befinden sich als Ergebnis in den Registern R4 (in dezimal, hexadezimal und binär) ?
- E. Für welche Bandbreite (minimaler und maximaler Wert) des in der Variablen "var1" initialisierten Wertes ist dieser Algorithmus gültig?