Atmel AT90Sxxxx ASM Programmierung

16Bit Subtraktion & Division

von Thimo Eichstädt

16 Bit Subtraktion

Rechnung

- Addition: 3+12=15

```
  0011      (03)
+1100       (12)
  1111      (15)
```

- Subtraktion: 15-3=12
  - Problem: 15-3= ?

```
  01111
- 00011  (incl. Vorzeichenbit)
  ?????
```

  - Lösung, 2er Komplement: 15 + (-3) = ?

```
  01111
+ 11101
  01100
```

16 Bit Subtraktion

Beispiel: 8Bit mit Übertrag

- 8Bit mit Carry auf einer 4Bit Recheneinheit:
  (235-99)=136

	1110 | 1011	(235)
+	1001 | 1101	(- 99)
	1000 | 1000	(136)

- Merkmale
  - Gleiches Verfahren bei 16Bit (2x8Bit)
  - Vorzeichenbit kann ebenfalls benutzt werden

16Bit Subtraktion

ASM Programmierung

- 16Bit Zahlenbereiche
  - Unsigned: 0 bis 65535
  - Signed: 32767 bis -32768
- Beispiel: 65000-40000=15000
  - Binär:
    - 65000: 1111 1101 | 1110 1000
    - 40000: 1001 1100 | 0100 0000
  - ASM:

```
ldiw r28,65000    ; Lade Register mit 16Bit Wert
ldiw r30,40000    ;

sub r29,r31       ; High Register abziehen
sbc r28,r30;      ; Low Register abziehen (mit Carry)
```

16Bit Division

Einführung

- Probleme:
  - Div Befehl auf Atmel nicht vorhanden
  - 16Bit Zahl, also Berechnung von 2 Registerinhalten (Low und High Bytes)
- Lösung:
  - Algorithmus: Division and Restore
- Verfahren:
  - Iterative Berechnung des Ergebnisses
  - Wie handschriftliches Dividieren

16Bit Division

Handschriftlich

- 154 : 14 = Quotient und Rest

```
  10011010 : 1110 = 01011 (Dezimal: 11)
- 1110
  -0101          negativ → Restore
+ 1110
  10011
-  1110
   01010         positiv
-   1110
    -0100        negativ → Restore
+   1110
    10101
-    1110
     01110       positiv
-    1110
     0000        positiv
                 Rest: 0000
```

Verfahren:

1. Dividend – Divisor = D1
2. D1 ≥ 0 : Q_{m-1}=1
3. D1 < 0 : Q_{m-1}=0 und Restore
4. Divisor um 1 Stelle nach rechts
5. Gehe zu 1.

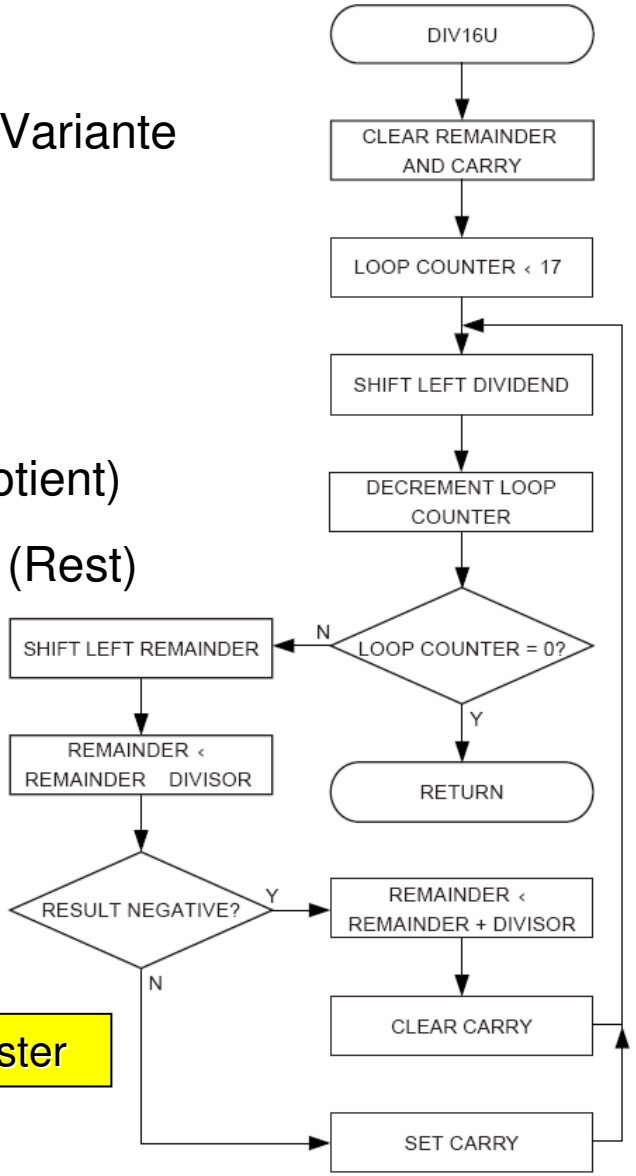
16Bit Division

Flussdiagramm

Code optimierte Variante

Register:

- Dividend
- Divisor
- Result (Quotient)
- Remainder (Rest)



Dividend Register = Result Register

16Bit Division

ASM Listing

```
.def   drem16uL=r14
.def   drem16uH=r15
.def   dres16uL=r16
.def   dres16uH=r17
.def   dd16uL   =r16
.def   dd16uH   =r17
.def   dv16uL   =r18
.def   dv16uH   =r19
.def   dcnt16u  =r20

;***** Code

div16u:         clr     drem16uL            ;clear remainder Low byte
                sub     drem16uH,drem16uH   ;clear remainder High byte and carry
                ldi     dcnt16u,17          ;init loop counter
d16u_1:         rol     dd16uL              ;shift left dividend
                rol     dd16uH
                dec     dcnt16u             ;decrement counter
                brne    d16u_2              ;if done
                ret                         ;    return
d16u_2:         rol     drem16uL            ;shift dividend into remainder
                rol     drem16uH
                sub     drem16uL,dv16uL     ;remainder = remainder - divisor
                sbc     drem16uH,dv16uH     ;
                brcc    d16u_3              ;if result negative
                add     drem16uL,dv16uL     ;    restore remainder
                adc     drem16uH,dv16uH
                clc                         ;    clear carry to be shifted into result
                rjmp    d16u_1              ;else
d16u_3:         sec                         ;    set carry to be shifted into result
                rjmp    d16u_1
```

16Bit Division

Vorzeichenbehaftet

- Prinzipielles Vorgehen:
  1. Vor Berechnung:
     Dividend XOR Divisor → Ergebnis wird positiv bzw. negativ
  2. Wenn Höchstes Bit in Dividend oder Divisor gesetzt, jeweils negieren
  3. **Nun Berechnung wie bei vorzeichenlosen Zahlen**
  4. Nach Berechnung das höchste Bit der Operation aus 1. prüfen; wenn höchstes Bit gesetzt ist, dann Ergebnis einmal negieren.

Rechenaufwand

Vergleich einiger Implementierungen

- Vorzeichenlos:
  - Vorgestellte Version (Code Optimiert):
    - Code Größe: 19 Words
    - Ausführungsdauer: 243 Zyklen
    - Benutzte Register: 5
  - Geschwindigkeitsoptimierte Version:
    - Code Größe: 196
    - Mittl. Ausführungsdauer: 173 Zyklen (Faktor 1,4 schneller)
    - Benutzte Register: 6
- Vorzeichenbehaftet:
  - Code Größe: 39 Words
  - Ausführungsdauer: 255 Zyklen
  - Benutzte Register: 7

Quellen

- Webseiten
  - Präsentation, ASM Listings, Links:

http://home.digithi.de/digithi/uni/uc-seminar/