Abgabe von Bruno Stendal, Martin Baer, Lukas Gewinner und Christian Schäfer
3. Aufgabenblatt zum Kurs

TI2: Rechnerarchitektur

von Bernadette Keßler

bis Freitag, den 25.11.2022, 10:15 Uhr.

Zahlendarstellung und Rechnen

Führen Sie die folgenden vier Berechnungen in B+V Darstellung, im Einerkomplement, Zweierkomplement und in einer Exzessdarstellung durch. Gehen Sie von 8-Bit breiten Registern aus. Wählen Sie für die Exzessdarstellung einen sinnvollen Offset und begründen Sie Ihre Wahl.

1. $23 + 81 = 104$

   (a) B+V Darstellung:

   $$\begin{array}{cr} & 00010111 \\ + & 01010001 \\ \text{Ü} & \text{—1-111-} \\ \hline & \underline{\underline{01101000}} \end{array}$$

   (b) Einerkomplement:

   $$\begin{array}{cr} & 00010111 \\ + & 01010001 \\ \text{Ü} & \text{—1-111-} \\ \hline & \underline{\underline{01101000}} \end{array}$$

   (c) Zweierkomplement:

   $$\begin{array}{cr} & 00010111 \\ + & 01010001 \\ \text{Ü} & \text{—1-111-} \\ \hline & \underline{\underline{01101000}} \end{array}$$

   (d) Exzessdarstellung: Begründung für den Offset in der Fußnote [1]. Offset wird auf die Zahlen addiert:
   $23 + 128 = 151 = 10010111$
   $81 + 128 = 209 = 11010001$

   $$\begin{array}{cr} & 10010111 \\ + & 11010001 \\ \text{Ü} & \text{—1-111-} \\ \hline & 01101000 \end{array}$$

   Vom Ergebnis den Offset subtrahieren um die richtige Darstellung zu erhalten.

[1] Wir haben den Offset 128 gewählt, da die Bitlänger der Binärzahl 8 Bit beträgt. Beim Offset rechnen kommt mit dieser Formel $2^{n-2} - 1 = 127$ raus bei der positiven Grenze und beim negativen $-2^{n-1} = -128$ raus als Grenze. Also ist der Offset $128 = 2^{n-1}$.

$$\begin{array}{cr} & 01101000 \\ - & 10000000 \\ \text{Ü} & 1\text{———————}- \\ \hline & \underline{\underline{11101000}} \end{array}$$

2. $36 - 14 = 22$

   (a) B+V Darstellung:

$$\begin{array}{cr} & 00100100 \\ - & 00001110 \\ \text{Ü} & \text{–}1111\text{–} \\ \hline & \underline{\underline{00010110}} \end{array}$$

   (b) Einerkomplement:

$$\begin{array}{cr} & 00100100 \\ + & 11110001 \\ \text{Ü} & 111\text{—————}- \\ \hline & 00010101 \end{array}$$

   Der zusätzliche Übertrag muss zum Zwischenergebnis dazu addiert werden:

$$\begin{array}{cr} & 00010101 \\ + & 00000001 \\ \text{Ü} & \text{—————}1- \\ \hline & \underline{\underline{00010110}} \end{array}$$

   (c) Zweierkomplement:

$$\begin{array}{cr} & 00100100 \\ + & 11110010 \\ \text{Ü} & 111\text{—————}- \\ \hline & \underline{\underline{00010110}} \end{array}$$

   (d) Exzessdarstellung: Begründung für den Offset in der Fußnote [2]. Offset wird auf die Zahlen addiert:
   $36 + 128 = 164 = 10100100$
   $14 + 128 = 142 = 10001110$

$$\begin{array}{cr} & 10100100 \\ - & 10001110 \\ \text{Ü} & \text{——}-111\text{——} \\ \hline & 00010110 \end{array}$$

   Vom Ergebniss den Offset addieren um die richtige Darstellung zu erhalten.

$$\begin{array}{cr} & 00010110 \\ + & 10000000 \\ \text{Ü} & \text{—————}- \\ \hline & \underline{\underline{10010110}} \end{array}$$

[2] Wir haben den Offset 128 gewählt, da die Bitlänger der Binärzahl 8 Bit beträgt. Beim Offset rechnen kommt mit dieser Formel $2^{n-2} - 1 = 127$ raus bei der positiven Grenze und beim negativen $-2^{n-1} = -128$ raus als Grenze. Also ist der Offset $128 = 2^{n-1}$.

3. $72 - 87 = -15$

   (a) B+V Darstellung:

$$\begin{array}{cc} & 01001000 \\ - & 10101000 \\ \text{Ü} & \text{1111–111—} \\ \hline & \underline{\underline{11110001}} \end{array}$$

   (b) Einerkomplement:

$$\begin{array}{cc} & 01001000 \\ + & 10101000 \\ \text{Ü} & \text{—-1——} \\ \hline & \underline{\underline{11110000}} \end{array}$$

   (c) Zweierkomplement:

$$\begin{array}{cc} & 01001000 \\ + & 10101001 \\ \text{Ü} & \text{—-1——} \\ \hline & \underline{\underline{11110001}} \end{array}$$

   (d) Exzessdarstellung: Begründung für den Offset in der Fußnote [3].Offset wird auf die Zahlen addiert:
   $72 + 128 = 200 = 11001000$
   $87 + 128 = 215 = 11010111$

$$\begin{array}{cc} & 11001000 \\ - & 11010111 \\ \text{Ü} & \text{1111–111—} \\ \hline & 11110001 \end{array}$$

   Vom Ergebniss den Offset addieren um die richtige Darstellung zu erhalten.

$$\begin{array}{cc} & 11110001 \\ + & 10000000 \\ \text{Ü} & \text{1———-} \\ \hline & \underline{\underline{01110001}} \end{array}$$

4. $-113 - 37 = -150$

   (a) B+V Darstellung:

$$\begin{array}{cc} & 10001110 \\ - & 00100101 \\ \text{Ü} & \text{111——1—} \\ \hline & \underline{\underline{01101001}} \end{array}$$

   (b) Einerkomplement:

[3]Wir haben den Offset 128 gewählt, da die Bitlänger der Binärzahl 8 Bit beträgt. Beim Offset rechnen kommt mit dieser Formel $2^{n-2} - 1 = 127$ raus bei der positiven Grenze und beim negativen $-2^{n-1} = -128$ raus als Grenze. Also ist der Offset $128 = 2^{n-1}$.

$$
\begin{array}{cr}
 & 10001110 \\
+ & 11011010 \\
\text{Ü} & 1\text{—}1111\text{——} \\
\hline
 & \underline{\underline{01101000}}
\end{array}
$$

Übertrag bei zwei Summanden mit identischem Vorzeichen führt zu einem Overflow!

(c) Zweierkomplement:

$$
\begin{array}{cr}
 & 10001111 \\
+ & 11011011 \\
\text{Ü} & 1\text{—}11111\text{—} \\
\hline
 & \underline{\underline{01101010}}
\end{array}
$$

Übertrag bei zwei Summanden mit identischem Vorzeichen führt zu einem Overflow!

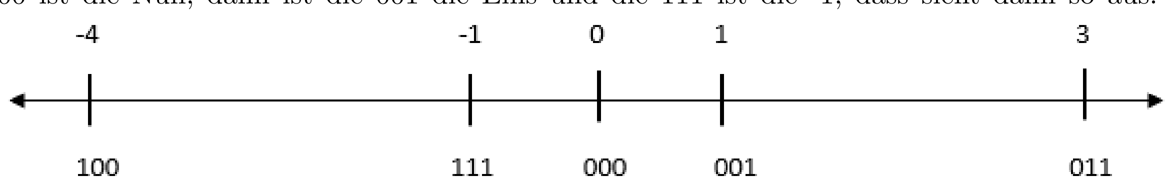
(d) Exzessdarstellung: Begründung für den Offset in der Fußnote [4].Offset wird auf die Zahlen addiert:
$-113 + 128 = 15 = 00001111$
$37 + 128 = 165 = 10100101$

$$
\begin{array}{cr}
 & 00001111 \\
- & 10100101 \\
\text{Ü} & 111\text{———-} \\
\hline
 & 01101010
\end{array}
$$

Vom Ergebniss den Offset addieren um die richtige Darstellung zu erhalten.

$$
\begin{array}{cr}
 & 01101010 \\
+ & 10000000 \\
\text{Ü} & \text{———-} \\
\hline
 & \underline{\underline{11101010}}
\end{array}
$$

Vorteil von Zweierkomplement:
Das Zweierkomplement ist besser, da es nur eine Darstellung für die Null hat und auch gleichzeitig, die richtige mathematische Reihenfolge der Zahlen zeigt. Als Beispiel 000 ist die Null, dann ist die 001 die Eins und die 111 ist die -1, dass sieht dann so aus:



Die negativen Zahlen werden durch das Vorzeichen gekennzeichnet, wie bei der B+V Darstellung, bloß ohne die Schreibweise von zwei Nullen.

[4]Wir haben den Offset 128 gewählt, da die Bitlänger der Binärzahl 8 Bit beträgt. Beim Offset rechnen kommt mit dieser Formel $2^{n-2} - 1 = 127$ raus bei der positiven Grenze und beim negativen $-2^{n-1} = -128$ raus als Grenze. Also ist der Offset $128 = 2^{n-1}$.

Typkonvertierung

Implementieren Sie eine „String to Integer“ *oder* - als deutlich forderndere Aufgabe - eine „Integer to String“ Funktion. Diese sollen zwischen einer ASCII Zeichenkette der Form „1234“ mit einer gegebenen Basis (bspw. 10) und ihrer entsprechenden Integer Zahlendarstellung 1234 konvertieren.

Es wurde sich für „String to Integer“ entschieden.

```
;Bruno Stendal, Martin Baer, Lukas Gewinner, Christian Schaefer
            global strToInt

strToInt:
    mov rax, 0; output ausnullen
    mov rcx, 0; counter ausnullen
  .loop:
    mov rdx, 0; data ausnullen
    mov dl, [rdi+rcx] ; erstes 8bit teilregister von rdi
                      ;nach 8-bit data register moven
    sub dl, 48; ascii versatz subtrahieren
    add al, dl; auf die letzten 8bit von rax addieren
    inc rcx; versatzcounter erh hen
    mov dl, [rdi+rcx]; n chstes 8bit teilregister einlesen
    cmp dl, 0;  mit 0 vergleichen, ob es der nullterimantor ist
    je .end;  falls ja springe zum ende
    jne .mult;  falls nein, multipliziere rax mit der basis
  .mult
    mul rsi;  rax mit der basis multiplizieren
    jmp .loop;
  .end
    ret
```

In der Abgabe ist der c-wrapper mit enthalten, da er angepasst wurde.