

Rechnerorganisation im WS 2017/18

Musterlösungen zum 2. Übungsblatt

Prof. Dr. Wolfgang Karl
Haid-und-Neu-Str. 7

Dr.-Ing. Ömer Terlemez
Adenauerring 2, Geb. 50.20

Email: ti@ira.uka.de

Web: <http://ti.ira.uka.de>

Lösung 1

(5 Punkte)

1. Was ist der Unterschied zwischen einer arithmetischen und einer logischen Verschiebung nach rechts (*shift right*)? 1 P.

Der Unterschied liegt darin, wie die freiwerdenden Bits auf der Seite des *most significant bits* aufgefüllt werden. Bei einer arithmetischen Verschiebung werden diese Stellen mit dem ursprünglichen Wert des *most significant bits* aufgefüllt, während bei einer logischen Verschiebung immer mit Null aufgefüllt wird.

Dies führt dazu, dass die arithmetische Verschiebung für beliebige Zahlen einer Division durch eine Zweierpotenz entspricht, wohingegen dies bei der logischen Verschiebung nur für positive Zahlen gilt.

2. `>>`-Operator angewendet auf `unsigned int`: 1 P.
Realisiert wird eine logische Verschiebung nach rechts. (Bei der Anwendung auf einem `signed int` wird die Art der Verschiebung durch den C-Standard nicht genau festgelegt, aber in den meisten Fällen eine arithmetische Verschiebung durchgeführt.)

3. Implementieren Sie `rotateLeft` unter Verwendung von bitweisen Operatoren in C: 3 P.

```
unsigned int rotateLeft(unsigned int number, unsigned int n) {  
    return (number << n) | (number >> (32 - n));  
}
```

Lösung 2

(4 Punkte)

1. Komponenten eines von-Neumann-Rechners:
Steuerwerk, Rechenwerk, Speicher, Verbindungseinrichtung (Bus) und Eingabe-/Ausgabe-Einheiten
2. i.) Befehlsregister
ii.) Statusregister
iii.) Programmzähler

Lösung 3

(4 Punkte)

1. Im Statusregister können je nach Architektur verschiedene Informationen ausgelesen werden. Gebräuchliche Flags sind: 2 P.

- Carry Flag (Übertrag aus höchstwertigen Bits)
- Zero Flag (Ergebnis war Null)
- Overflow Flag (Überlauf bei Operation)
- Sign Flag (Vorzeichen des Ergebnis)
- Even Flag (Ergebnis war gerade)
- Parity Flag (Ergebnis hat gerade Anzahl an Einsen)

2 P.

2. Beim arithmetischen Rechtsschieben findet eine sogenannte Vorzeichenerweiterung statt, d.h. führende Leerstellen werden mit dem vormalig höchstwertigen Bit aufgefüllt.

Dadurch entspricht das arithmetische Rechtsschieben bei beliebigen Zahlen im Zweierkomplement einer Division durch die entsprechende Zweierpotenz der geschobenen Bits.

Beim logischen Rechtsschieben wird immer mit Nullen aufgefüllt. Es gibt hier bei negativen Zahlen also keine anschauliche arithmetische Interpretation.

Lösung 4

(7 Punkte)

1. Die Lese-Phase (*fetch phase*) erledigt folgende Dinge: 2 P.

- i.) Das InstruktionsAdressRegister in das SpeicherAdressRegister laden und das Lese-Flag auf 1 setzen ($R = 1$).
- ii.) Das InstruktionsAdressRegister in das X Register der ALU laden; das Eins Register in das Y Register laden und beide Werte addieren.
- iii.) Anschließend das berechnete Ergebnis in das InstruktionsAdressRegister speichern.
- iv.) Zum Schluss das aktuelle Datum aus dem SpeicherAdressRegister in das InstruktionsRegister laden.

Damit wird der aktuelle Befehl aus dem Speicher geholt und in das InstruktionsRegister kopiert. Zusätzlich wird das InstruktionsAdressRegister um den Wert 1 erhöht um im nächsten Zyklus den nachfolgenden Befehl aus dem Speicher laden zu können.

2. Kodiertes Mikroprogramm für die Lese-Phase (*fetch phase*): 5 P.

0x00	2 1 0 8 8 0 1	$X = P_w = S = 1; R = 1$
0x01	1 4 0 0 8 0 2	$Y = E = 1; R = 1$
0x02	0 0 0 1 8 0 3	$C_2 - C_0 = 001; R = 1$
0x03	0 A 0 0 0 0 4	$Z = P_r = 1$
0x04	0 0 9 0 0 0 5	$I_r = D_w = 1$