

Rechnerorganisation im WS 2017/18

Frohe Weihnachten und ein erfolgreiches neues Jahr

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

(12 Punkte)

Siehe `blattWeihnachten_vier_gewinnt_musterloesung.s`.

Lösung 2

(11 Punkte)

1. MYSTIC CODE setzt die folgende Switch-Anweisung in MIPS-Assembler um:

2 P.

```
switch ( kobi_dir ) {  
    case 0: temp1 = kobi_pos_y - 1;  
            break;  
    case 1: temp0 = kobi_pos_x + 1;  
            break;  
    case 2: temp1 = kobi_pos_y + 1;  
            break;  
    case 3: temp0 = kobi_pos_x - 1;  
}
```

Ein default-Fall wurde nicht implementiert. Auf die Benutzung einer Lookup-Tabelle in Form von je einem Label pro Case wurde hier verzichtet. Stattdessen wird das Sprung-Offset direkt berechnet und die Ausführung an der Programmadresse `Case0 + kobi_dir * 8` fortgesetzt.

2. Siehe `blattWeihnachten_robot_musterloesung.s`.

2 P.

3. Siehe `blattWeihnachten_robot_musterloesung.s`.

3 P.

4. Siehe `blattWeihnachten_robot_musterloesung.s`.

4 P.

Lösung 3

(3 Punkte)

1. Taktfrequenz 100 Mhz $\Rightarrow$ Zykluszeit $\frac{1}{100000000Hz} = 10ns$

1 P.

Benötigte Zyklen: $100 + 7 = 107$ (7 zum Befüllen der Pipeline ohne dass ein Befehl abgeschlossen wird)

$\Rightarrow$ Benötigte Zeit: $107 \cdot 10ns = 1070ns$

2. Speedup: $S = \frac{8 \cdot 100 \cdot 10ns}{1070ns} = 7.4766$

2 P.

Durchsatz: $D = \frac{100}{1070ns} = 0.0936 \frac{1}{ns}$