

Name:

Punkte: /23P Note:

1) Praxis Assemblerprogrammierung

```
.include "m16def.inc"
    clr XH
    ldi XL, 0x66
    clr YH
    ldi YL, 0x60
    ldi R16, 0x02
    out SPH, R16
    ldi R16, 0x00
    out SPL, R16
    rcall strcat
end:
    rjmp end
strcat:
    ld R16, X+
    cpi R16, 0x00
    brne strcat
    dec XL
cp:
    ld R16, Y+
    st X+, R16
    cpi R16, 0x00
    brne cp
    ret
```

Speicherauszug:

0x5F	0x60	0x61	0x62	0x63	0x64	0x65	0x66	0x67	0x68	0x69	0x6A	0x6B	0x6C	0x6D	0x6E	0x6F	0x70	0x71	0x72
0x37	0x35	0x41	0x48	0x45	0x4C	0x00	0x48	0x20	0x69	0x00	0x00	0x4A	0x5A	0x42	0x6C	0x63	0x52	0x51	0x32

1.a) Wie groß ist das Programm in Bytes? ___/1P
 ___ Bytes

1.b) Wie viele Takte benötigt das Programm bis zum Erreichen des Labels end? ___/2P
 ___ Takte

1.c) Ausführung des Programms ___/3P

Was steht im Speicher nach der Ausführung?

0x5F	0x60	0x61	0x62	0x63	0x64	0x65	0x66	0x67	0x68	0x69	0x6A	0x6B	0x6C	0x6D	0x6E	0x6F	0x70	0x71	0x72

Welche Werte haben die Register nach der Ausführung?

R16	X	Y	SP

1.d) Fragen zum Programm ___/3P

- | | richtig | falsch |
|--|--------------------------|--------------------------|
| Das Programm zählt die Anzahl der 0x00 Bytes | <input type="checkbox"/> | <input type="checkbox"/> |
| Das Programm nutzt den Stack | <input type="checkbox"/> | <input type="checkbox"/> |
| Es wird nur das Register R16 verwendet | <input type="checkbox"/> | <input type="checkbox"/> |
| Es werden Daten kopiert | <input type="checkbox"/> | <input type="checkbox"/> |
| Die Laufzeit ist abhängig von den Daten im Speicher | <input type="checkbox"/> | <input type="checkbox"/> |
| Am Ende der Routine ist der X Zeiger wieder am ursprünglichen Wert | <input type="checkbox"/> | <input type="checkbox"/> |

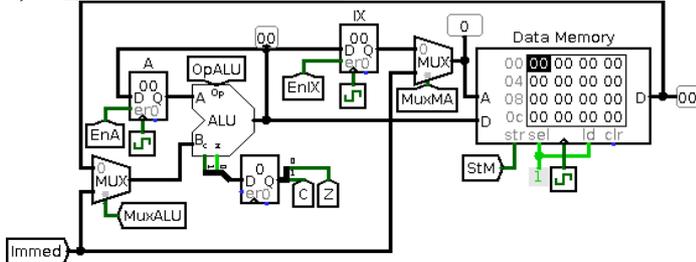
2) CISC vs. RISC

2.a) Multiple Choice ___/4P

Ordne die folgenden (typischen) Eigenschaften der jeweiligen Architektur zu

- | | CISC | RISC |
|---|--------------------------|--------------------------|
| Befehlsausführung meist in einem Takt | <input type="checkbox"/> | <input type="checkbox"/> |
| Steuerwerk wird oft mittels Mikrocode realisiert | <input type="checkbox"/> | <input type="checkbox"/> |
| Es gibt meist nur ein oder zwei allgemeine Register | <input type="checkbox"/> | <input type="checkbox"/> |
| Für Pipelining optimiert | <input type="checkbox"/> | <input type="checkbox"/> |
| Gilt als "modernere" Architektur | <input type="checkbox"/> | <input type="checkbox"/> |
| Datentransfer fast nur über Load-Store Befehle | <input type="checkbox"/> | <input type="checkbox"/> |
| Atmel AVR ist ein typischer... | <input type="checkbox"/> | <input type="checkbox"/> |
| Die Programme sind im allgemeinen kleiner | <input type="checkbox"/> | <input type="checkbox"/> |

3) Praxis Rechnerarchitektur



ALU Operation	Beschreibung
000	Result=A
001	Result=B
010	Result=A+B
011	Result=A-B
100	Result=A AND B
101	Result=A OR B
110	Result=A EOR B
111	Result=A>>1

3.a) Ordne die Beschreibung richtig zu

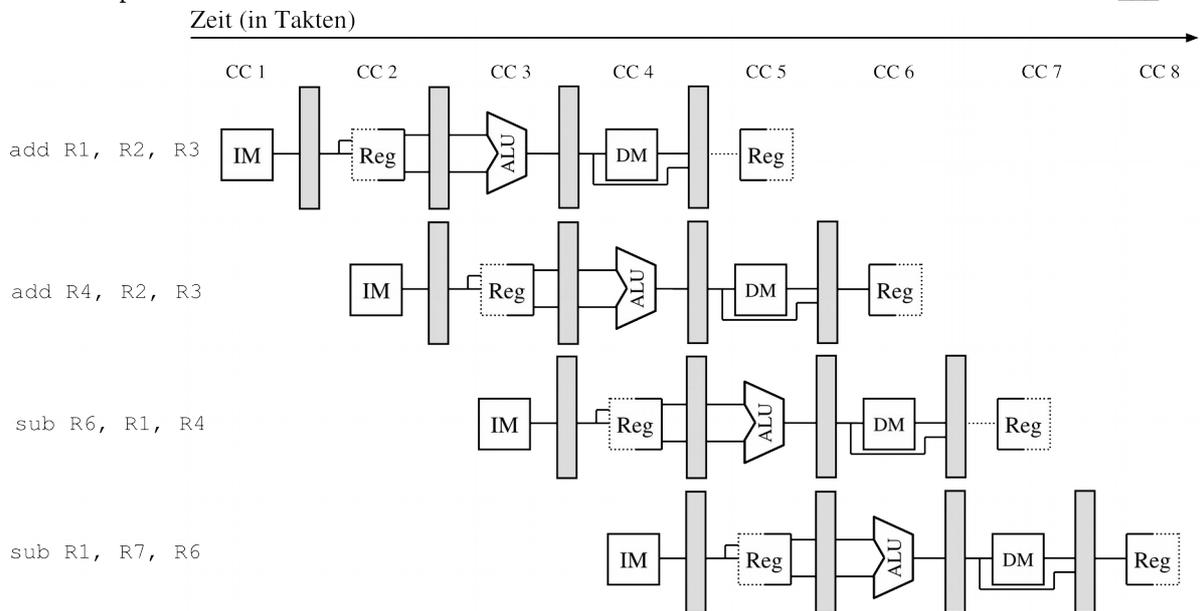
___/4P

- or A, Imm. - Ver-ODER-t Register A mit Konstante
- st (IX), Imm. - Speichert die Konstante Imm. an der Adresse IX
- ld IX,A – Lädt Register IX mit dem Wert aus Register A
- and A, (IX) – Ver-UND-et Register A mit dem Wert an Adresse IX
- ld A, (Imm.) - Lädt Register A mit dem Wert an der Adresse Imm.
- ld IX, (IX) – Lädt Register IX mit dem Wert an der Adresse IX
- nop – Führt keine Operation aus (No Operation)
- st (IX), A – Speichert den Wert von Register A an der Adresse IX

Nummer	EnA	EnIX	StM	MuxALU	MuxMA	OpALU
	0	1	0	x	x	000
	1	0	0	1	x	101
	1	0	0	0	1	001
	0	0	1	x	0	000
	0	0	0	x	x	x
	0	1	0	0	0	001
	1	0	0	0	0	100
	0	0	1	1	0	001

3.b) DLX Pipeline: Zeichne die Datenhazards ein

___/3P



3.c) 0-Adressarchitektur

___/3P

Eine gegebene Stackmaschine unterstützt die Befehle push, add und mult. Welche Befehle werden ausgeführt, um die Berechnung $(3+1)*(7+3)$ auszuführen?

