Lösungsvorschläge zur Hauptklausur 1705 vom 01.02.1997

Aufgabe 1)

a) 1) S-END = S-START + LEN -1, falls LEN > 0.

2) D_END = D-START + LEN -1, falls LEN > 0.

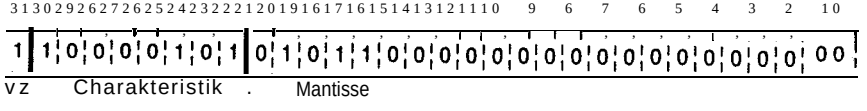
b)

```
START:  LI R16, 0          : Register R14 bis R16 mit den
        LI R15, 1          ; benötigten Konstanten initialisieren
        LI R14, 4
        SGR R13, R3, R16   ; LEN > O?
        BEQZ R13, 7        ; LEN=0 -> Sprung zum Ende
        LD R12, R1, 0      ; Speichewort kopieren
        ST R12, R2, 0
        ADD R1, R1, R14    ; S-START, D-START um 4 erhöhen
        ADD R2, R2, R14    ; (Anzahl der Bytes pro Speicherwort)
        SUB R3, R3, R15    ; LEN dekrementieren.
        JMP -7             ; Sprung zum Schleifenanfang
ENDE:   JMP 0              ; Ende
```

c) Das obige Programm kopiert die Speicherworte aufsteigend von S-START bis S-END. Ist D-START kleiner oder gleich S_END (D-START <= S-END) wird beim Kopieren ein Teil des Zielbereichs überschrieben.

Zu Aufgabe 2:

a) **i)**



ii)

Vorzeichen: VZ = 1 = '-'

Charakteristik: C = $85 = 133_{10}

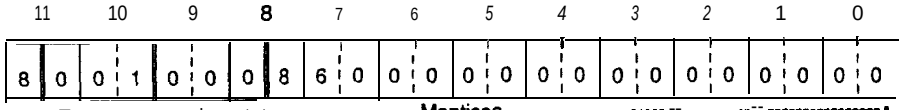
Exponent: E = $ 85 - $ 7F = $ 06 = $+6_{10}$

Mantisse: I.M = $(1.010110 \ldots 0)_2$ = 1.3437510

iii) $Z = -0.86 \cdot 10^{+2} = -86.0_{10}$

b) Z = $ 80 00 00 00 00 00 00 00 00 86

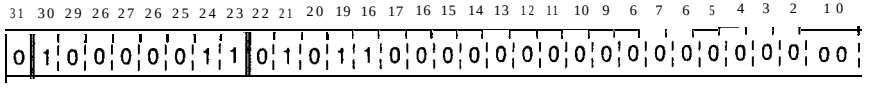
c)



Z = $ 80 01 00 08 60 00 00 00 00 00 00 00

d)

Z' = $\underline{+ 1.34375 * 2^{+4}}$ = 66.0 / 4 = -Z / 4 (vgl. a) ===> Charakteristik C' = $83



Z' = $ 41 AC0000

Zu Aufgabe 3:

a) realisierte Strategie: **Pseudo-LRU** (*Least recently used*, s. S. ???? in KE ????)

Begründung: Die Zustände der Bits B2, B1, BO nach dem Zugriff bzw. vor der Verdrängung eines Eintrags sind jeweils invertiert zueinander. Mit großer Wahrscheinlichkeit wird daher immer der Eintrag verdrängt, auf den am längsten nicht mehr zugegriffen wurde. Die Abweichung zum „echten" LRU-Verfahren kommt hauptsächlich daher, daß durch B2 eine Paarbildung vorgenommen wird und ein Verdrängungskandidat nur im jeweils anderen Paar gesucht wird. (Vgl. Zugriff 10 unter b), bei dem eigentlich Var6 verdrängt werden müßte.)

b) Die Cache-"Fehlgriffe" (*Miss*) sind grau unterlegt.

Zugriff	Variable	Weg 3	Weg 2	Weg 1	Weg 0	B2	B1	B0
0		(frei)	(frei)	(frei)	(frei)	0	0	0
1	Var0	(frei)	(frei)	(frei)	Var0	0	0 (u)	0
2	Var1	(frei)	(frei)	Var1	Var0	0	0 (u)	1
3	Var2	(frei)	Var2	Var1	Var0	1	0	1 (u)
4	Var3	Var3	Var2	Var1	Var0	1	1	1(u)
5	Var1	Var3	Var2	Var1	Var0	0	1 (u)	1
6	Var0	Var3	Var2	Var1	Var0	0	1 (u)	0
7	Var4	Var3	Var4	Var1	Var0	1	0	0 (u)
8	Var3	Var3	Var4	Var1	Var0	1	1	0 (u)
9	Var4	Var3	Var4	Var1	Var0	1	0	0 (u)
10	Var0	Var3	Var2	Var1	Var0	0	0 (u)	0
11	Var1	Var3	Var2	Var1	Var0	0	0 (u)	1

c)

Kapazität der einzelnen Cache-Wege: 128 / 4 = 32 kbyte

Anzahl der Einträge pro Weg: 32k / 64 = 512 Einträge pro Weg

31 30 29 26 27 26 25 24 23 22 21 20 19 16 17 16 15 14 13 12 11 10 9 6 7 6 5 4 3 2 1 0		
TAG	INDEX	Byte

Tag: Bits 31 - 15, Index: Bits 14 - 6, Byte-Adresse: Bits 5 - 0

d)

Speicheradresse	Tag	Index	Byte-Adresse
$FBC3 5AB0	1F786	16A	30
$0625 9A87	00C4B	06A	07
$CD2F DAA6	19A5F	16A	26

Nur die 1. und 3. Adresse belegen einen Eintrag mit derselben Nummer (Index), und zwar den Eintrag Nr. $16A = 362. (Dies ist der 363. Eintrag, da mit 0 begonnen wird.)

Aufgabe 4:

a) CS-DRAM = (DS1 + DSO) . !A23 . IA22 * !BOOT)

CS-EPROM = (DS1 + DSO) * ((A23 . !A22 . !BOOT) + (!A23 . !A22 * BOOT)

OE_EPROM = (DS1 + DSO) . ((A23 . IA22 . !BOOT) + (IA23 . IA22 . BOOT) . R/W

b) BYTE = !DS1 . DSO

HALF = DS1 . !DSO

WORD = DS1 . DSO

BE0 = (BYTE . !A1 * !AO) + (HALF . !Al * !AO) + (WORD . !Al . !AO)

BE1 = (BYTE . !Al . AO) + (HALF * !Al * !AO) + (WORD * !Al . !AO)

BE2 = (BYTE . Al . !AO) + (HALF . Al . IAO) + (WORD . !A1. !AO)

BE3 = (BYTE * Al * AO) + (HALF . Al * !AO) + (WORD . !Al . !AO)

c) Durch die 4 CAS-Signale ist eine gezielte Auswahl der einzelnen Speicherbytes möglich und gewährleistet. daß durch das Einschreiben eines Datums, das kürzer als 32 Bit ist, nicht die anderen Bytes des selktierten Speicherwortes zerstört werden.

Aufgabe 5:

a) Maskenregister: 00000000 = 0x00

Befehlsregister: 00000100 = 0x04

b)

IR7	IR6	IR5	IR4	IR3	IR2	IR1	IR0	Ereignis	Interrupt
								Masken- und Befehlsregister erhalten die Werte aus a)	-
				X					3
		X							5
								EOI-Bit wird gesetzt	3
								EOI-Bit wird gesetzt	-
								Maskiere IR7 und IR6	-
								AC-Bit wird gesetzt	-
X									-
		X							5
				X					3
			X						4

Aufgabe 6:

a) Viel Erfolg!

b) FireWire