Aufgabe 1: (10 Punkte)

Welche der folgenden Aussagen sind richtig ?

		ja	nein
a)	Die Begriffe **Speicherseite** und **Speicherbank** bedeuten dasselbe, werden also synonym gebraucht.	☐	☐
b)	Beim **LW-Verfahren** (Least *Recently Used*) kann u.U. auch das zuletzt eingelagerte Datum als erstes wieder verdrängt werden.	☐	☐
c)	Unter dem Begriff **DMA** (*Direct* Memory *Access)* versteht man den direkten Zugriff des Prozessors auf den Arbeitsspeicher ohne den Umweg über den Cache.	☐	☐
d)	Gleitpunktzahlen nach dem IEEE-754-Standard werden stets durch **Abschneiden** (*Truncation*) niederwertiger Bits gerundet.	☐	☐
e)	Die maximalen **Zählzykluslängen** eines 16-bit-Timers sind im 2x8-bit- und 16-bit-Zählmodus gleich.	☐	☐

Aufgabe 2: (10 Punkte)

Nennen Sie für die im folgenden genannten Paare von Verfahren bzw. Begriffen jeweils wenigstens einen Vorteil und einen Nachteil:

a) Der **Multiplexbus** hat gegenüber dem nicht **gemultiplexten Bus** den

Vorteil.....

.....

Nachteil.....

.....

b) Das **Hardware Interlocking** einer Pipeline hat gegenüber einem **Bypass** den

Vorteil*.....

.....

Nachteil.....

.....

c) Ein **vollassoziativer Cache** hat gegenüber einem **Direct-Mapped Cache** den

Vorteil.....

.....

Nachteil*.....

.....

d) Der Anschluß eines Druckers über die **V.24-Schnittstelle** hat gegenüber dem Anschluß über die **Centronics-Schnittstelle** den

Vorteil: .....

.....

Nachteil: .....

.....

e) Die Verwaltung des Registersatzes in **überlappenden Registerbänken** hat gegenüber einem **homogenen Registersatz** den

Vorteil: .....

.....

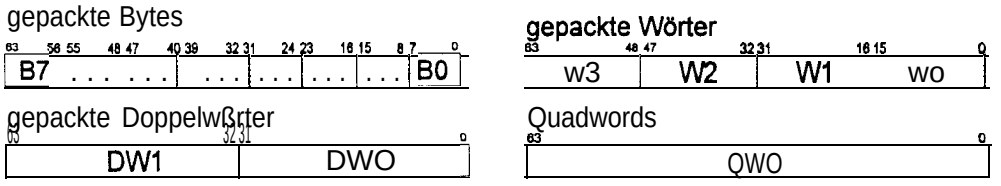
Nachteil: .....

.....

Aufgabe 3: MMX-Rechenwerk

(20 Punkte)

Ein MMX-Rechenwerk (Multimedia Extension) unterstützt gepackte Datenformate nach untenstehendem Bild, die in einem 64-bit-Register wahlweise 8 Bytes, 4 (16-bit-)Wörter, 2 (32-bit-)Doppelwörter oder ein 64-bit-Wort (Quadword) unterbringen. Alle Werte können vorzeichenlos oder vorzeichenbehaftet sein. Negative Werte werden dabei im 2er-Komplement dargestellt. Spezielle MMX-Befehle wirken parallel auf diese Datenformate, d.h. es können z.B. durch einen einzigen Addier-Befehl zweimal 8 Bytes addiert werden. Ein Übertrag zwischen den einzelnen Werten der gepackten Daten findet dabei nicht statt. Die Datenbreite wird im Assemblerbefehl spezifiziert. Der erwähnte Addier-Befehl hat z.B. die Form: PADDX, wobei X=B,W,D für 8 Bytes, 4 Wörter, 2 Doppelwörter steht. Alle Befehle werden im Zweiadreß-Format benutzt, d.h. das Ergebnis wird in einem der Eingaberegister abgelegt.

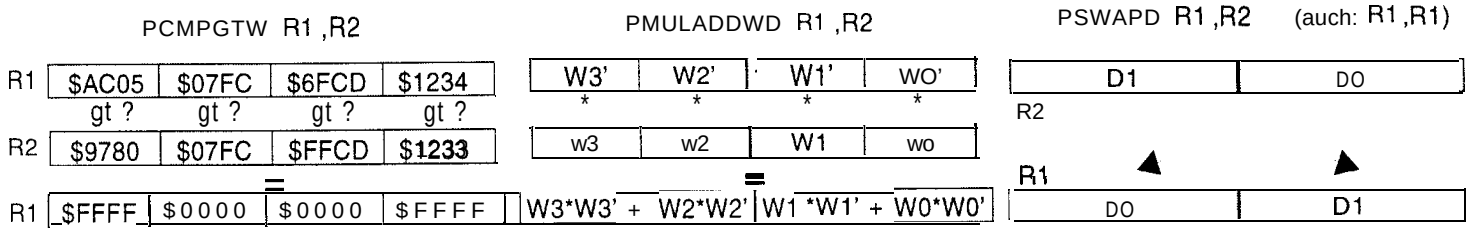


Der MMX Befehlssatz enthält u.a. die in folgender Tabelle angegebenen Befehle:

Mnemo		Bemerkung
	Logische Operationen (wirken auf alle 64 Bits)	
PAND	*Packed And*	bitweise Und-Verknüpfung
PANDN	*Packed And-Not*	bitweise Und-Verknüpfung mit negiertem 1. Operanden
POR	*Packed Or*	bitweise Oder-Verknüpfung
PXOR	*Packed Exclusive Or*	bitweise Antivalenz-Verknüpfung
	Vergleichsbefehle (für X = B, W, D, s. Skizze)	
PCMPEQX	*P. Compare Equal*	elementeweiser, vorzeichenloser Vergleich auf gleich
PCMPGTX	*P. Compare Greater*	elementeweiser, vorzeichenloser Vergleich auf größer
	Transferbefehle	
MOVQ	*Move Quadword*	Transfer zw. MMX-Reg. Und MMX-Reg. Bzw. Speicher
PSWAPD	*P. Swap Doubleword*	Vertauschen der Doppelwörter in Register, s. Skizze
	Multiplizier-Addierbefehl (W x W → D, s. nachfolgende Skizze)	
PMULADDWD	Multiply-Add	4fache Multiplikation mit Addition zu Doppelwörtern
PADDX	*Packed Add*	Parallele Addition mit X=B,W,D

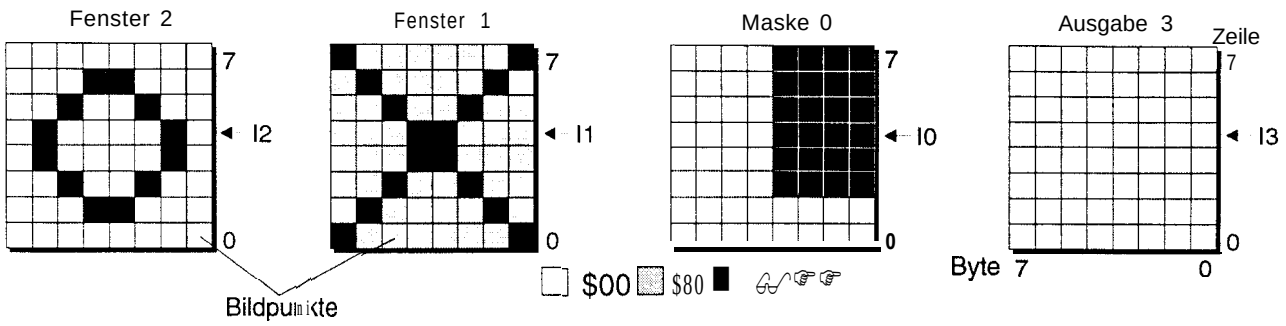
Als Ergebnis liefern die Vergleichsbefehle für jedes Element (B,W,D) den Wert $F..F (also eine Folge von ‚I'-Bits), wenn der Vergleich wahr ist, andernfalls den Wert $O..O (also eine Folge von ‚O'-Bits).

Skizze:



Assemblernotation: <Befehl> R1 ,R2 bedeutet: R1 <op> R2 → R1.

a) In vier Bereichen eines Graphikspeichers seien die im folgenden Bild dargestellten 8x8-byte-Ausschnitte zweier Bildschirmfenster (Bereiche 2, 1), eine gleich große Maske (Bereich 0) sowie ein Ausgabefeld (Bereich 3) abgelegt. Die Adressierung der korrespondierende Zeilen dieser Bereiche geschieht – wie im Bild gezeigt -jeweils Register-indirekt über ein Adreßregister I3,..,I0 mit der Assembler-Syntax: (Ij). Durch die benutzten Farben der Elemente weiß, grau, schwarz sollen die Hexadezimalwerte $00, $80, $FF repräsentiert werden.



Geben Sie eine Befehlsfolge an, die

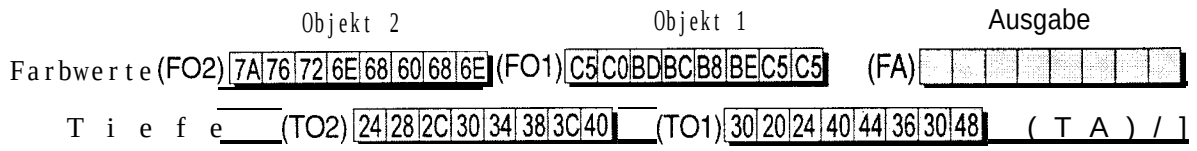
- die Zeilen der Bereiche i in die Register Ri (i=0,1,2) der MMX-Einheit einliest,
- die Register R2, R1 mit dem Register RO nach folgender Vorschrift parallel zum Ergebnisregister R3 verknüpft (j=0,..,7):

$$\text{Byte j von R3} = \begin{cases} \text{Byte j von R2,} & \text{falls Byte j von RO} = \$00 \\ \text{Byte j von R1,} & \text{falls Byte j von RO} = \$FF. \end{cases}$$

- das Ergebnisregister in den Ausgabebereich 3 überträgt.

Skizzieren Sie durch verschiedene Muster (weiß, schwarz, schraffiert) im Ausgabebereich 3 des Bildes das Ergebnis der oben stehenden Befehlsfolge für alle Zeilen.

b) Die Byte-Vektoren im folgenden Bild repräsentieren jeweils acht Punkte, die auf einer Linie zweier dreidimensionaler Objekte liegen und sich bei der Bildschirmdarstellung überlagern. Die oberen Zahlen geben dabei die Farbwerte der Punkte wieder, die unteren ihren Abstand von der Bildschirmoberfläche: je größer dieser Wert, desto ,,tiefer“ liegt der Punkt. Geben Sie eine Befehlsfolge an, die parallel für alle Paare der Punkt-Vektoren diejenigen Punkte ermittelt, die näher an der Bildschirmoberfläche liegen, und die Far- und Tiefenwerte dieser Punkte in die Ausgabe-Vektoren überträgt. Es werde dabei davon ausgegangen, daß alle Byte-Vektoren im Speicher abgelegt sind.



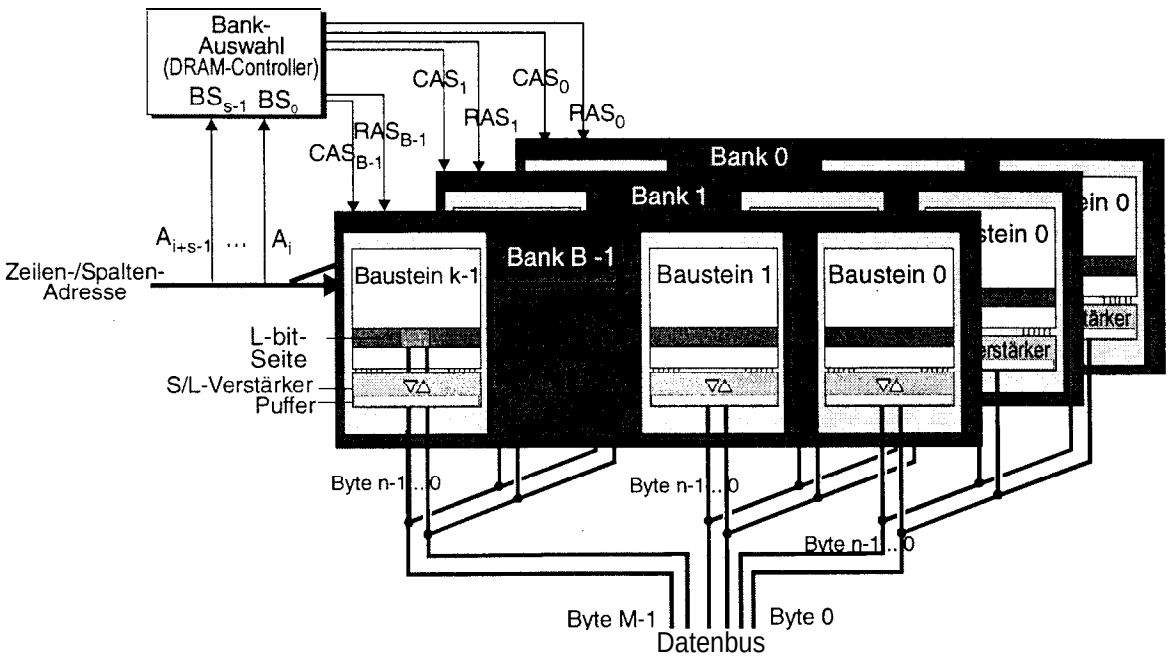
Benutzen Sie für Ihre Lösung bitte die im Bild in Klammern angegebenen Variablennamen.

Tragen Sie die Ergebnisse der Befehlsfolge in die Ausgabevektoren FA, TA ein.

Aufgabe 4: (20 Punkte)

Das Bild zeigt einen (byteweise adressierbaren) Arbeitsspeicher aus DRAM-Bausteinen, die in B Speicherbänken ($B = 2^s$, s=0,..,3) untergebracht sind. Die Breite eines Speicherworts sei M byte ($M = 2^m$; m=O,...,3). Die DRAM-Bausteine besitzen eine Organisation von N x n byte ($n = 2^q$; q=0,1,2; $n \leq M$, N beliebig). Eine Seite (Zeile) ihrer Speichermatrizen habe eine Länge von L bit ($L = 2^p$, p=8,..,12).

Es werde davon ausgegangen, daß die meisten Speicherzugriffe auf eine Folge von Speicherwörtern unter sequentiellen Adressen stattfinden.



a) Die benutzten Speicherbausteine sollen zunächst nicht über die Möglichkeit eines Seitenzugriffs verfügen, d.h. die Schreib/Leseverstärker (S/L, s. Bild) besitzen keine Pufferregister. In einer Zugriffsfolge unter sequentiellen Wortadressen finde ein Bankwechsel nach jedem Zugriff statt.

i. Bestimmen Sie allgemein (in s, m, p, q) die geeigneten Adreßsignale $A_{i+s-1},..,A_i$ für die Ansteuerung der (binär-codierten) Bank-Auswahlsignale BS,., ,..., BS_0.

ii. Geben Sie die Adreßsignale für den Fall: vier Speicherbänke, byteweise organisierte DRAMs, 64-bit-Speicherwörter, Seitengröße: 1024 bit an.

b) Nun bestehe der Arbeitsspeicher aus DRAMs mit Seitenzugriff (Pufferregister in den S/L-Verstärkern). Weiterhin sollen alle anderen oben gemachten Annahmen gelten. In einer Zugriffsfolge unter sequentiellen Wortadressen finde ein Bankwechsel bei jedem Wechsel der aktiven Speicherseite statt.

i. Bestimmen Sie wiederum allgemein (in s, m, p, q) die geeigneten Adreßsignale $A_{i+s-1},..,A_i$ für die Ansteuerung der Bank-Auswahlsignale $BS_{s-1},...,BS_0$.

ii. Geben Sie die Adreßsignale für den Fall: M = 4 byte, zwei Speicherbänke, Nx16-bit-DRAMs, Seitengröße: 4096 bit an.

Aufgabe 5: Cache (20 Punkte)

Ein Mikroprozessor mit 16-bit-Adreßbus und 32-bit-Datenbus besitze einen *Direct-Mapped* Cache mit acht 32-bit-Einträgen (Cache *Lines*), dessen Aufbau im Bild 1 dargestellt ist.

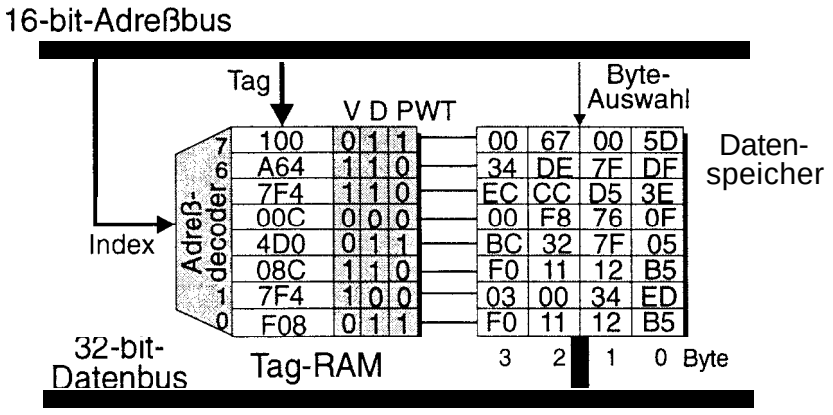


Bild 1: Der Direct-Mapped Cache

Zur Vereinfachung der hexadezimalen Darstellung wird das ‚$'-Zeichen stets weggelassen und die Tag-Einträge werden durch soviele niederwertige ‚0'-Bits ergänzt, daß ihre Länge ein Vielfaches von 4 ist'. Zu jedem Eintrag stehen im Tag-RAM drei Statusbits zur Verfügung:

- Das V-Bit (*Valid*) zeigt an, daß der Cache-Eintrag gültig ist,
- das D-Bit (*Dirty*) zeigt an, daß auf den Cache-Eintrag schreibend zugegriffen wurde,
- das PWT-Bit (Page *Write-Through*) zeigt (durch eine ‚l') an, daß der Cache-Eintrag zu einer Speicherseite gehört, für die das Rückschreibverfahren benutzt wird.

Bild 2 zeigt drei Speicherseiten (mit je acht 32-bit-Wörtern). Jede Seite besitzt zwei Steuerbits, die ihre Verarbeitung im Cache regeln:

- PCE (Page Cache *Enable*) zeigt an, ob die Wörter der Seite in den Cache eingelagert werden dürfen (PCE=1, *cacheable*) oder nicht (PCE=O, *non-cacheable*).
- PWT *(Page Write-Through*) legt für die Seite die Durchführung der Schreibzugriffe fest:
  PWT=1: Durchschreibverfahren; bei einem *Write Miss* wird das Datum nur in den Arbeitsspeicher geschrieben.
  PWT=O: Rückschreibverfahren (*Write-Back*); beim Rückschreiben werden nur die veränderten (D=l), gültigen (V=1) Cache-Einträge zurückgeschrieben.

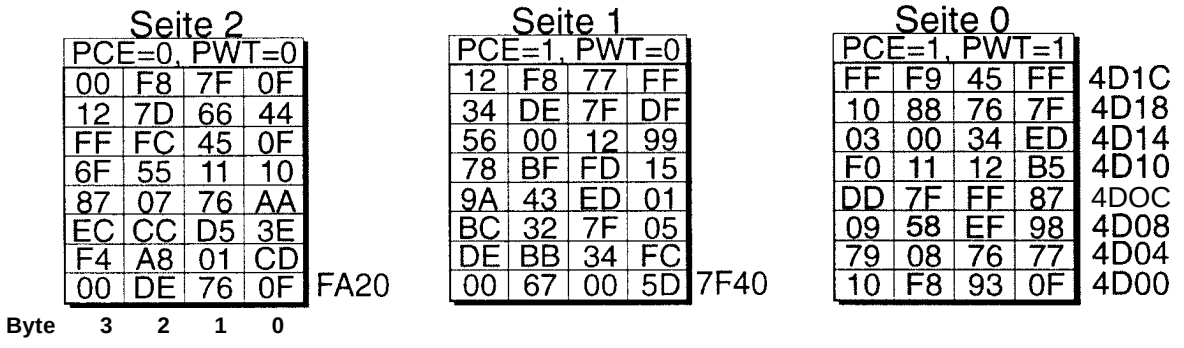
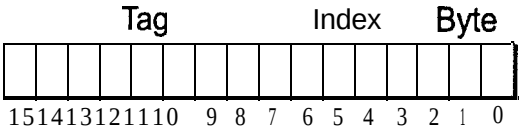


Bild 2: Belegung der Speicherseiten

a) Geben Sie für den Cache nach Bild 1 die Aufteilung einer Adresse in Tag, Index und Byteauswahl an:



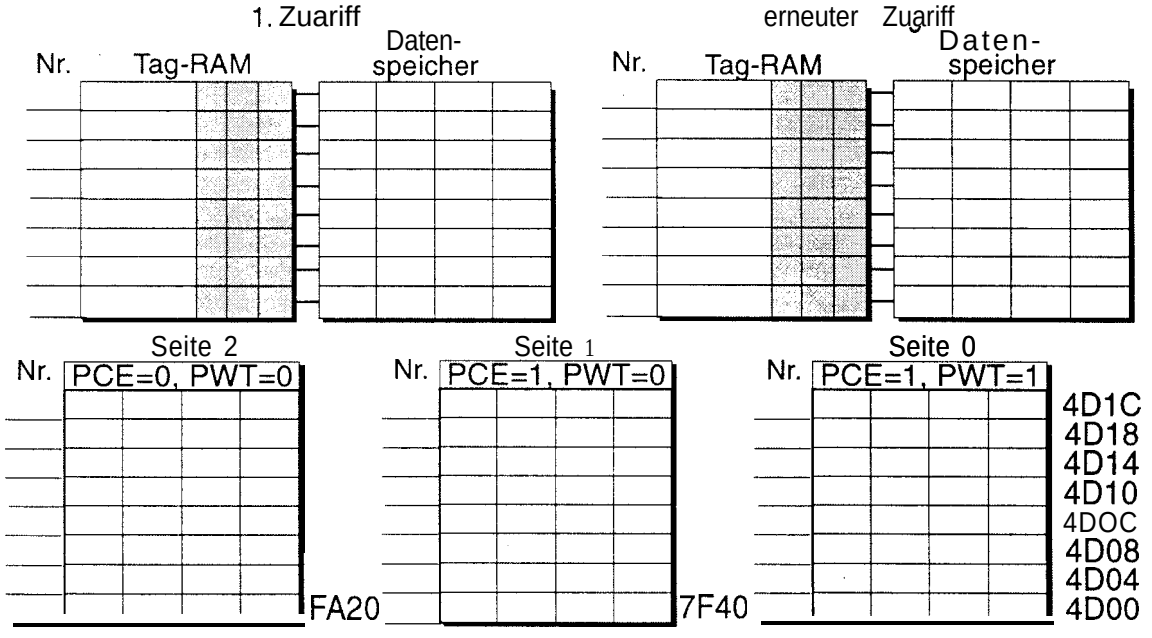
b) Tragen Sie in die Tabellen und Bilder die Auswirkungen der Folge von Lade- und Speicherbefehle auf den Cache und die Speicherseiten ein. Beschreiben Sie im Feld Bemerkung die Auswirkung der Befehle, insbesondere der Befehle, die nicht die betrachteten Speicherseiten oder den Cache betreffen. (Benutzen Sie die Abkürzungen: AS(n): Arbeitsspeicher, C(i): Cache-Eintrag i) Die Spalte ‚Ri' gibt für Schreibbefehle den Inhalt des Registers vor, bei Lesebefehlen müssen Sie dort das Ergebnis eintragen. Kennzeichnen Sie die Einträge in die Speicherbilder durch die Befehlsnummer.

LD Ri, n *,,Lade das Register Ri mit dem Inhalt des Speicherwortes mit der Adresse n. "*

ST *Rj,* n *,,Speichere den Inhalt des Registers ßj in das Speicherwort mit der Adresse n. "*

FLUSH *,,Schreibe alle gültigen, veränderten Cache-Einträge in die Speicherseiten mit Rückschreib-Verfahren zurück. Erkläre dann* <u>*alle*</u> *Cache-Einträge für ungültig. "*

Nr.	Befehl	Ri	Bemerkung
1	LD R0,$FA28		
2	ST R1,$7F44	FE5460DC	
3	LD R2,$4D0C		
4	LD R3,$7F58		
5	ST R4,$FA3C	01234567	
6	ST R5,$4D04	5F3C72AB	
7	LD R6,$7F4C		
8	FLUSH		



[1] Beispiel: Tag 1101010101 wird dargestellt als 1100 0101 0100 = $C54.

Aufgabe 6: Timer (20 Punkte)

Geben sei ein 16-bit-Zeitgeber/Zähler-Baustein (Timer) mit dem im folgenden Bild dargestellten Steuerregister CR. Die darin eingetragenen Bitfelder sollen im wesentlichen die im Kurs beschriebenen Funktionen haben.

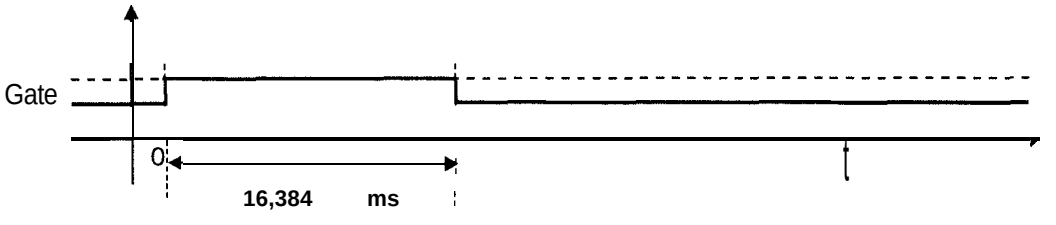
IE	OE	M2	M1	MO	CM	CL	PR

Es gelten die folgenden Abweichungen und Festlegungen:

- der Timer arbeitet nur mit einem einzigen Anfangswert, d.h. einem Auffangregister (*Latch*) pro Zähler; er ist re-triggerbar;
- das Rücksetz-Bit fehlt;
- durch IE=1 werden Interrupts zur CPU durchgelassen;
- durch OE=1 wird der Timer-Ausgang freigeschaltet;
- der Zähler kann nur dual im 16-bit-Modus (CM=1) oder 2x8-bit-Modus (CM=0) betrieben werden, nicht jedoch als Dezimalzähler;
- CL wählt zwischen einem externen 1 MHz-Takt (CL=1) oder dem internen Systemtakt (f >> 1 MHz, CL=O);
- durch PR=1 wird ein 1:8-Frequenzteiler aktiviert;
- durch die Bits M2, M1, MO werden die folgenden Arbeitsweisen selektiert:

M2	M1	MO	Mode
0	0	0	Periodischer Zählmodus (Taktgenerator), nur Hardware-Triggerung
0	0	1	Periodischer Zählmodus, Hardware- und Software-Triggerung
0	1	0	Monoflop-Betrieb, nur Hardware-Triggerung
0	1	1	Monoflop-Betrieb, Hardware- und Software-Triggerung
1	0	0	*Strobe*-Betrieb, nur Hardware-Triggerung
1	0	1	*Strobe*-Betrieb, Hardware- und Software-Triggerung
1	1	0	Impulsmessung; Interrupt, falls Impulslänge < Zählzyklus
1	1	1	‘Frequenzmessung; Interrupt, falls Schwingungsdauer < Zählzyklus

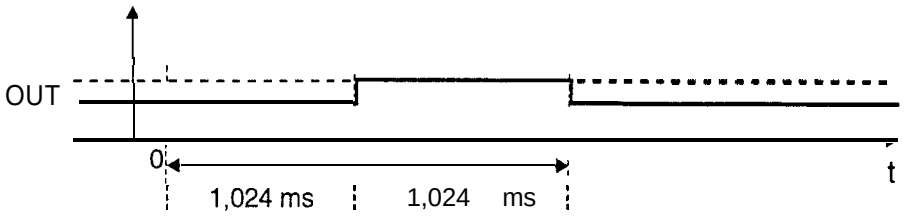
a) Das Auffangregister sei mit dem Wert $FFFF, das Steuerregister mit dem Wert $36 geladen. Am Gate-Eingang werde das folgende Zeitsignal angelegt:



Geben Sie an, in welchem Betriebsmodus der Timer arbeitet und welcher Zählerstand im Timer nach dem Ende des Impulses am Gate-Eingang vorliegt.

b) Geben Sie an, wie der Prozessor aus dem unter a) ermittelten Zählerstand die Länge des Impulses am Gate-Eingang berechnen kann. Wie lang ist er in Periodendauern des Zählertaktes ?

c) Am Timer-Ausgang soll nach einer Verzögerung, die mit der unter b) ermittelten Impulsdauer übereinstimmt, ein Strobe-Signal der Länge 1 µs ausgegeben werden. (Interrupts zum Prozessor seien deaktiviert.) Welche Werte müssen dazu in das Auffangregister und das Steuerregister geschrieben werden ?

d) Im periodischen Zählmodus (Taktgenerator) wechsle der Timer-Ausgang mit jeder Initialisierung des Zählers seinen Zustand.
Geben Sie für den 2x8-bit- sowie den 16-bit-Zählmodus die Programmierung des Auffangregisters und des Steuerregisters an, für die das Signal am Ausgang OUT die niedrigste Frequenz annimmt. (Software-Triggerung, Interrupts zum Prozessor seien deaktiviert.) Bestimmen Sie für beide Fälle die Periodendauer des Ausgangssignals.

e) Am Timer-Ausgang OUT soll im 2x8-bit-Zählmodus ein Signal nach folgender Skizze ausgegeben werden. Geben Sie an, mit welchen Werten dazu das Steuerregister und das Auffangregister geladen werden müssen. (Weitere Annahmen: keine Interrupts, Software-Triggerung, externer Takt.)