

# 7. Optimierung von PC-Speicherkonfigurationen

Im PC gibt es zwei leistungsentscheidende Speichersubsysteme:

1. den Arbeitsspeicher-Cache-Verbund,
2. den Bildspeicher (Frame Buffer) im Videoadapter.

(Es ist möglich, Arbeits- und Bildspeicher zu einer Speicheranordnung zu vereinigen (UMA = Unified Memory Architecture). Näheres in Abschnitt 7.2.6.)

Ein Speichersubsystem sollte (1) eine jeweils hinreichende Speicherkapazität bereitstellen und (2) jede Anforderung (d. h. alle Lese- und Schreibzugriffe) möglichst ohne Wartezustände bedienen können.

## 7.1. Der Cache-Arbeitsspeicher-Verbund

### 7.1.1. Leistungsvermögen

Das Leistungsvermögen von Speichersubsystemen wird durch Datenraten, Taktzyklen bzw. Taktfrequenzen, Wartezustände und Zugriffszeiten gekennzeichnet. Ein ideales Speichersubsystem muß nicht "unendlich schnell" sein, sollte aber alle Zugriffe in dem Zeitraster ausführen, das von der jeweils anfordernden Einrichtung vorgegeben wird (mit anderen Worten: das Speichersubsystem sollte die Zugriffe nicht durch Einfügen von Wartezuständen verlängern). In der PC-Praxis bedeutet dies: das Speichersubsystem ist an die Taktfrequenzen und Signalspiele der typischen Prozessor-Bussysteme anzupassen.

*Kennzeichen der gängigen Prozessor-Bussysteme:*

- Bus-Takt: in nicht allzu alten PCs sind folgende Taktfrequenzen üblich: 25, 33, 50, 60, 66 MHz (wobei 66 MHz dem gängigen Stand der Technik entsprechen). Weiterentwicklungen: 75, 83, 100, 125 MHz (und künftig auch noch darüber hinaus). Mit der Taktfrequenz steigen die technologischen Anforderungen (Tabelle 7.1).
- Zugriffsschema: leistungsentscheidend sind Burst-Zugriffe, in denen jeweils 4 aufeinanderfolgende Speicherworte (gemäß der jeweiligen Datenwegbreite; Stand der Technik: 64 Bits) gelesen oder geschrieben werden. Seitens des Prozessors erfordert der erste Zugriff 2 Taktzyklen, während die 3 folgenden Zugriffe jeweils nur einen Taktzyklus benötigen.
- Bezeichnungsweise: Burst-Zyklen werden genauer gekennzeichnet, indem nacheinander angegeben wird, wieviele Taktzyklen der einzelne Zugriff erfordert. So führen die Prozessoren ihre Burst-Zugriffe als 2-1-1-1-Zyklen aus, falls keine Wartezustände eingefügt werden. Als Beispiel für einen durch Wartezustände verlängerten Burstzyklus mag ein 5-3-3-3-Zyklus dienen (der erste Zugriff erfordert 5 Takte, die 3 folgenden Zugriffe benötigen jeweils 3 Takte).

Bus-Taktfrequenz	Beherrschbarkeit (in der Fertigung)	Typische Merkmale der Schaltungstechnik
25...33 MHz	nicht trivial, aber ohne weiteres beherrschbar (relativ unkritische Toleranzen)	5-V-Logik, asynchrone SRAMS und DRAMs
> 33...66 MHz	noch mit herkömmlichen Mitteln beherrschbar	äußerste Grenze für herkömmliche 5-V-Logik und asynchrone Speicherschaltkreise
> 66...100 MHz	enge Toleranzen <sup>*)</sup> , aber noch mit vergleichsweise kostengünstigen Mitteln beherrschbar	3,3-V-Logik, LVTTTL-Signalpegel, herkömmliche Taktverteilung, synchrone SRAMs und DRAMs
> 100...< 200 MHz	sehr enge Toleranzen <sup>*)</sup> (200 MHz $\triangleq$ 5 ns Taktzyklus - nicht etwa Takttoleranz!), mit herkömmlichen Mitteln gar nicht mehr beherrschbar, hohe Anforderungen an die Leiterplattenauslegung	SSTTL-Signalpegel, PLL-Taktverteilung, synchrone SRAMs und DRAMs mit doppelter Datenrate (DDR = Nutzung beider Taktflanken zur Signalübertragung)
200 MHz und mehr	extrem enge Toleranzen, mit herkömmlichen Mitteln gar nicht mehr beherrschbar, extreme Anforderungen an die Leiterplattenauslegung und konstruktive Durchbildung. Erfordert durchgängige "Kochbuch"-Systemlösungen (Schaltkreise - Interfaces - Leiterplattenauslegung)	Speisespannungen < 3,3 V, besondere Interface-Auslegungen und Signalpegel (SSTL, GTL, Rambus Signaling Logic (RSL), SLIO ( $\triangleq$ SSTL-2; früher: SyncLink), gesonderter Bus zwischen Prozessor und L2-Cache (DIB = Dual Independent Bus Architecture)), auf schnelle Blocktransporte optimierte Interfaceprotokolle (Rambus, SLDRAM)

\*): "enge Toleranzen" bedeutet auch: es ist mit Problemen zu rechnen, wenn Hardware verschiedenartiger Herkunft "gemischt" eingesetzt wird (betrifft z. B. Speichermoduln verschiedener Hersteller - manche nehmen es mit den Spezifikationen sehr genau, manche weniger)

**Tabelle 7.1** Zur Kennzeichnung von Bus-Taktfrequenzen

## 7.1.2. L2-Caches

Bei Bus-Taktfrequenzen von 33 MHz und mehr sind externe bzw. L2-Caches die einzige Möglichkeit, Wartezustände bei Speicherzugriffen zu vermeiden (dies kann allerdings nicht für *alle* Zugriffe erfüllt werden - Stichwort: Trefferrate). L2-Caches werden typischerweise mit SRAMs bestückt. Anhand der Tabellen 7.2 und 7.3 soll das Leistungsvermögen der verschiedenen SRAM-Typen diskutiert werden.

Bus-Takt	L2-Cache-Bestückung (SRAMs)		
	asynchron	Flow Thru Burst	Pipelined Burst
33 MHz	2-1-1-1	2-1-1-1	3-1-1-1
50 MHz	3-2-2-2	2-1-1-1	3-1-1-1
60 MHz	3-2-2-2	2-1-1-1	3-1-1-1
66 MHz	3-2-2-2	2-1-1-1	3-1-1-1
75 MHz	3-2-2-2	3-2-2-2	3-1-1-1
83 MHz	3-2-2-2	3-2-2-2	3-1-1-1
100 MHz	3-2-2-2	3-2-2-2	3-1-1-1
125 MHz	3-2-2-2	3-2-2-2	3-1-1-1

**Tabelle 7.2** Zum Leistungsvermögen verschiedener L2-Cache-Bestückungen (1). Typische Burst-Zyklen bei Lesezugriffen

*Erklärung:*

Wir betrachten hier nicht die Trefferrate, sondern lediglich das Vermögen des L2-Cache, Speicherzugriffe des Prozessors im Falle eines Treffers (Cache Hit) in einem bestimmten Taktraster zu bedienen. Ohne jegliche Wartezustände (d. h. mit 2-1-1-1-Zyklen) gelingt dies nur dann, wenn die ersten Lesedaten noch im 2. Taktzyklus des ersten Zugriffs bereitgestellt werden können.

*Zusammenhänge:*

- beim Pipelined-Burst-SRAM (PBSRAM) gelingt dies prinzipbedingt gar nicht, da der erste Lesezugriff infolge der Pipelining-Organisation grundsätzlich mit einer Verzögerung (Latenzzeit) von einem Taktzyklus ausgeführt wird (vgl. Kapitel 2, Abschnitt 2.1.2.).
- bei asynchronen und Flow-Thru-SRAMs gibt es keine derartige Latenzzeit. Deshalb ist es an sich möglich, Wartezustände vollkommen zu vermeiden.
- Caches mit asynchronen SRAMs können nur bis zu einer Taktfrequenz von ca. 33 MHz ohne Wartezustände betrieben werden,
- bei Flow-Thru-SRAMs ist diese Grenze bei ca. 66 MHz erreicht,
- höhere Taktfrequenzen erfordern grundsätzlich den Einsatz von PBSRAMs,
- entscheidend ist die Minimierung der gesamten Übertragungsdauer (für die 4 Worte des Burst-Zyklus),
- Übertragungsdauer  $t_t = \text{Gesamt-Zahl aller Taktzyklen} \cdot \text{Dauer des einzelnen Taktzyklus } (t_c)$ . Aus einem Taktschema A-B-B-B ergibt sich also  $t_t = (A + 3 B) \cdot t_c$ .
- der PBSRAM ist dann überlegen, wenn gilt  $6 t_{c\text{PBS}} < 5 t_{c\text{Flow-Thru}}$  bzw.  $t_{c\text{PBS}} < 5/6 t_{c\text{Flow-Thru}}$ . Da bei 66 MHz der Flow-Thru-Speicher seine Leistungsgrenze (hinsichtlich der 2-1-1-1-Zyklen) erreicht hat, setzen wir  $t_{c\text{Flow-Thru}} = 15 \text{ ns}$  (1 : 66 MHz) an. Damit ergibt sich  $t_{c\text{PBS}} < 12,5 \text{ ns}$ . Ergebnis: an einem 80-MHz-Bus (Industriestandard: 83 MHz) gewährleistet der PBSRAM trotz des zusätzlichen Wartezustandes dieselbe Übertragungsdauer wie ein ohne Wartezustand betriebener Flow-Thru-SRAM an einem 66-MHz-Bus (mit anderen Worten: von 80 MHz an aufwärts wird der PBSRAM eindeutig überlegen).

Bus-Takt	Takt- zyklus	L2-Cache-Bestückung (SRAMs)							
		asynchron				Flow Thru Burst		Pipelined Burst	
		$t_{AC}^{1)}$	Banks	Lesen	Schreiben	$t_{AC}^{1)}$	Zugriff <sup>2)</sup>	Lesen <sup>3)</sup>	Schreiben <sup>4)</sup>
50 MHz	20 ns	20	1	3-2-2-2	4-2-2-2	12	2-1-1-1	3-1-1-1	2-1-1-1
60 MHz	16,7 ns	15	1 2	3-3-3-3 3-2-2-2	4-3-3-3 4-2-2-2	10	2-1-1-1	3-1-1-1	2-1-1-1
66 MHz	15 ns	15	1 2	3-3-3-3 3-2-2-2	4-4-4-4 4-2-2-2	9	2-1-1-1	3-1-1-1	2-1-1-1
75 MHz	13,3 ns	15	2	3-2-2-2	4-2-2-2	9	3-2-2-2	3-1-1-1	2-1-1-1
83 MHz	12 ns	12	2	3-2-2-2	4-2-2-2	9	3-2-2-2	3-1-1-1	2-1-1-1
100 MHz	10 ns	10	2	3-2-2-2	4-2-2-2	9	3-2-2-2	3-1-1-1	2-1-1-1
125 MHz	8 ns	8	2	3-2-2-2	4-2-2-2	9	3-2-2-2	3-1-1-1	2-1-1-1

1)...4): siehe Erklärung im Text

**Tabelle 7.3** Zum Leistungsvermögen verschiedener L2-Cache-Bestückungen (2). Nach: Micron

*Erklärung:*

- 1)  $t_{AC}$  = Zugriffszeit in ns (Datenblattangabe),
- 2) Lese- und Schreibzugriffe können nach dem gleichen Zeitraster ausgeführt werden,
- 3) der erste Lesezugriff erfordert stets einen zusätzlichen Takt,
- 4) beim ersten Schreibzugriff gibt es keine Latenzzeit.

Tabelle 7.3 ist etwas detaillierter als Tabelle 7.2, allerdings werden auch hier gewisse Spitzfindigkeiten (wie der Wechsel zwischen Lese- und Schreibzugriffen) nicht beachtet.

*Hinweise:*

1. Asynchrone SRAMs sind bei Taktfrequenzen  $> 33$  MHz "gerade noch" sinnvoll einsetzbar. Bei höhere Taktfrequenzen sind aufwendigere Speicherstrukturen (2 Banks) und schnelle (also teure) Speicherschaltkreise erforderlich.
2. Pipelined-Burst-RAMs (PBSRAMs) können bei jeder beliebigen Taktfrequenz in den Taktrastern 3-1-1-1 (Lesen) bzw. 2-1-1-1 (Schreiben) betrieben werden.
3. Bis 66 MHz sind Flow-Thru-RAMs den PBSRAMs überlegen - aber nur dann, wenn tatsächlich ein Betrieb ohne Wartezustände möglich ist (2-1-1-1).
4. Speicherschaltkreise sind stets gemäß der jeweiligen Taktfrequenz- bzw. Zugriffszeitspezifikation auszuwählen (bzw. "1 Nummer besser").
5. Die Angaben in Tabelle 7.3 gelten nur dann, wenn das gesamte Speichersubsystem entsprechend ausgelegt ist (also Speichertyp und Zugriffsschema auch unterstützt). Das betrifft im besonderen die Angaben zu den asynchronen und Flow-Thru-RAMs. In der Praxis: die Dokumentation des Motherboards ansehen (ersatzweise: das Datenmaterial des Schaltkreissatzes - aber Vorsicht: Motherboard-Fabrikanten sparen gelegentlich gern an der "Restlogik", z. B. an zwischengeschalteten Bustreibern). Deshalb kann es sein, daß die Zugriffe länger dauern als in Tabelle 7.3 angegeben - und dann helfen auch die schnellsten Speicher-ICs nichts.

6. Flow-Thru-RAMs sind (s. oben) bis zu 66 MHz zwar überlegen, sind aber schwerer beschaffbar als PBSRAMs - und sie werden auch nicht von allen einschlägigen Motherboard-Schaltkreissätzen unterstützt.
7. Faustregel: wenn man das Zugriffsschema des L2-Caches nicht kennt: die Annahme 3-2-2-2 ist eine gute Näherung.

### 7.1.3. Arbeitsspeicher

DRAMs erfordern, infolge ihrer Funktionsweise (vgl. Kapitel 5, Abschnitt 5.2.1.), daß besondere Zeit-Anforderungen eingehalten werden. Sie sind deshalb typischerweise "langsamer" als die Prozessoren, und nur bei vergleichsweise niedrigen Bus-Taktfrequenzen (Richtwert: > 16 MHz) gelingt es, ohne das Einfügen von Wartezuständen auszukommen. Um zu ermitteln, nach welchem Zugriffsschema eine DRAM-Anordnung bei gegebener Bus-Taktfrequenz betrieben werden kann, muß man alle Minimum-Zeitspezifikationen (gemäß DRAM-Datenblatt) sozusagen auf ganze Taktperioden aufrunden und die einzelnen Zeiten gemäß den jeweils erforderlichen Signalfolgen (z. B. RAS - Adreßumschaltung - CAS - Datenübernahme - Precharge) aufaddieren. Hinzu kommen noch Zeitzuschläge für das Erkennen, ob der Zugriff im Page Mode ausführbar ist (Page Hit) oder nicht (Page Miss), für das Umsteuern der Datenwege zwischen Lese- und Schreibzugriffen (Bus Turnaround), für das Durchlaufen von Treiber- und Registerstufen (Datenweg-Pipeline) usw. In der Praxis wird vielfach nur ein einziges Zugriffsschema angegeben. Dieses bezeichnet den Ablauf eines Burst-Zyklus unter bestmöglichen Bedingungen (Page Hit). Tabelle 7.4 nennt entsprechende Richtwerte und ergänzt diese durch Angaben zu weiteren Zugriffsabläufen.

DRAM-Typ	Zugriffsablauf			
	Page Hit <sup>1)</sup>	Bank (Row) Miss <sup>2)</sup>	Page Miss <sup>3)</sup>	aufeinanderfolgende Page Hits <sup>4)</sup>
asynchron, FPM <sup>5)</sup>	5-3-3-3 6-3-3-3	8-3-3-3 9-3-3-3	11-3-3-3 12-3-3-3	x-3-3-3-3-3-3
asynchron, EDO <sup>6)</sup>	5-2-2-2 6-2-2-2	8-2-2-2 9-2-2-2	11-2-2-2 12-2-2-2	x-2-2-2-3-2-2-2
SDRAM <sup>7), 8)</sup>	5-1-1-1 6-1-1-1	8-1-1-1 9-1-1-1	11-1-1-1 12-1-1-1	x-1-1-1-2-1-1-1

1)...8): siehe Erklärung im Text

**Tabelle 7.4** Burstzyklen mit verschiedenen DRAMs bei 66 MHz Bustakt (Richtwerte)

#### Erklärung:

- 1) Seite ist geöffnet bzw. (SDRAM) aktiviert. Zugriff erfolgt nur mit Spaltenadresse (CAS-Impuls oder (SDRAM) Zugriffskommando). Aber auch bei einem Page Hit dauert der erste Zugriff eines Burst-Zyklus länger als die folgenden: es ist festzustellen, ob es sich überhaupt um einen Page Hit handelt, Datenwege sind umzuschalten, die Datenweg-Pipeline ist anfänglich zu füllen usw. Die Dauer des ersten Zugriffs heißt üblicherweise *Leadoff Time*, wobei gelegentlich zwischen Read Leadoff und Write Leadoff unterschieden wird (z. B. Read Leadoff = 6 Takte, Write Leadoff = 5 Takte).
- 2) "Row" ist hier nicht die Zeile im DRAM (gemäß Zeilenadresse), sondern eine Reihe oder Bank von DRAM-Schaltkreisen in der (typischerweise aus Speichermoduln gebildeten) Speicheranordnung (vgl. Kapitel 4, Abschnitt 4.6.3.). Betrifft der Zugriff eine andere Reihe, so muß diese aktiviert werden (/RAS → Low bzw. Kommando "Aktivieren"). Der maßgebliche Datenblatt-Kennwert (RAS to CAS Delay  $t_{\text{RCD}}$ ) bestimmt die zusätzliche Leadoff-Zeit (Row Miss Leadoff). Richtwert: 4 Takte.

- 3) "Page Miss" heißt: der Zugriff betrifft dieselbe (bereits aktive) Reihe (Bank) von DRAM-Schaltkreisen, es ist aber eine neue Zeilenadresse zu übernehmen. Hierzu ist es notwendig, die geöffnete Seite zu schließen (= die DRAMs zu deaktivieren) und mit der neuen Zeilenadresse wieder zu aktivieren. Die Leadoff-Zeit verlängert sich deshalb um RAS-Vorladezeit (Precharge Leadoff; Richtwert: 3 Takte) + RAS to CAS Delay (bzw. Row Miss Leadoff - s. Punkt 2). Es ergibt sich somit insgesamt ein zusätzlicher Page Miss Leadoff von typischerweise  $3 + 3 = 6$  Takten.
- 4) diese Spalte veranschaulicht schnelle Folgen von Zugriffen, die Page Hits sind.  $x$  = Leadoff-Zeit des ersten Zugriffs. Hinweis: Lesezugriffe, die Page Hits sind, können an sich stets lückenlos aneinandergereiht werden. Vielfach ist jedoch die Speichersteuerung hierzu nicht in der Lage.
- 5) Geschwindigkeitsklasse: 60 ns. Gelegentlich können 70-ns-Typen noch im gleichen Taktraster betrieben werden (Page-Mode-Zykluszeit (CAS-Periodendauer)  $t_{PC} = 40$  bzw. 45 ns).
- 6) Geschwindigkeitsklasse: 60 ns. Maßgeblich ist die gegenüber FPM kürzere Page-Mode-Zykluszeit (CAS-Periodendauer)  $t_{PC}$  (25 ns).
- 7) Geschwindigkeitsklasse: 100 MHz (10 ns). Gelegentlich können 83-MHz-Typen (12 ns) noch im gleichen Taktraster betrieben werden.
- 8) Burst-EDO- (BEDO-) DRAMs können bei 66 MHz nach dem gleichen Zugriffsschema betrieben werden. (Bei höheren Taktfrequenzen gelingt dies aber nicht mehr.) Typische Zugriffsfolgen von BEDO-RAMs bei Page Hit: 6-1-1-1 bzw. 6-2-2-2.

*Hinweise zur Speicherauswahl:*

1. FPM-DRAM: für 66 MHz Bustakt sind 60-ns-Typen zu empfehlen. Bei langsamerem Bustakt genügen meistens die 70-ns-Typen.
2. EDO-DRAM: bei 66 MHz Bustakt sollten wenigstens 60-ns-Typen eingesetzt werden (besser: 50-ns-Typen).
3. Zum Einfluß der DRAM-Zugriffszeiten (Datenblattwerte) siehe Tabelle 7.5.
4. SDRAM: je höher die Taktfrequenz, um so wahrscheinlicher ist mit Problemen zu rechnen, im besonderen beim Einsatz von Typen verschiedener Herkunft (Stichwort: "Lernphase" der Schaltkreishersteller - es dauert seine Zeit, bis alle Hersteller die Spezifikationen gleichartig auffassen und gleichermaßen verlässlich prüfen).
5. Das Zeitraster ("DRAM-Timing") eines Motherboards kann typischerweise über das BIOS-Setup beeinflußt werden (das einschlägige Auswahl-Menue heißt "Chipset Features" o. ä.). Hier sind u. a. die Leadoff-Zeiten einzustellen.

Kennwert	Beschleunigung eines Burst-Zyklus bei Verkürzung um 1 Takt
RAS-Zugriffszeit $t_{RAC}$	1 Taktperiode ( $\triangleq 7\%$ bei 5-3-3-3)
CAS-Zugriffszeit $t_{CAC}$	3 Taktperioden ( $\triangleq 21\%$ bei 5-3-3-3)

**Tabelle 7.5** Einfluß der Zugriffszeit-Kennwerte auf den Datendurchsatz

*Erklärung:*

Wir betrachten hier, um wieviele Taktperioden wir den Burstzugriff beschleunigen könnten, wenn wir schnellere DRAMs auswählen. "Schneller" heißt hier: der jeweilige Zugriffszeit-Kennwert erlaubt es, den Burst-Zugriff um die Dauer einer Taktperiode zu verkürzen. Die Tabelle bestätigt die Erkenntnis, daß - wenn es um Burst-Zugriffe geht - die CAS-Zugriffszeit (bzw. die CAS-Periodendauer) wichtiger ist als die RAS-Zugriffszeit (vgl. die Punkte 5 und 6 der Erklärung zu Tabelle 7.4).

## 7.1.4. L2-Cache und Arbeitsspeicher im Verbund

### 7.1.4.1. Speicherkonfigurationen und Zugriffszeiten

Wie lange ein Burst-Zyklus des Prozessors in einem Verbund aus L2-Cache und Arbeitsspeicher dauert, läßt sich lediglich als Erwartungswert angeben:

*Erwartungswert für Datenaustausch (Füllen/Zurückschreiben) zwischen L1-Cache (im Prozessor) und L2-Cache:*

$$L2\text{-Trefferrate}^{(1)} \cdot L2\text{-L1-Transportzeit}^{(2)} + L2\text{-Miss-Rate}^{(3)} \cdot DRAM\text{-L2-Transportzeit}^{(4)}.$$

- 1) Angabe als Wahrscheinlichkeit (Wert zwischen 0 und 1; = %-Wert : 100). Typische L2-Trefferraten: 0,8...0,98 (80...98%).
- 2) gemäß Taktschema (z. B. 2-1-1-1). Takte zusammenzählen.
- 3) = 1 - L2-Trefferrate.
- 4) gemäß dem nachfolgend beschriebenen Erwartungswert. Oft begnügt man sich aber mit dem Taktschema des Page Hit (z. B. 5-3-3-3). Takte zusammenzählen.

*Erwartungswert für Datenaustausch (Füllen/Zurückschreiben) zwischen L2-Cache und Arbeitsspeicher (DRAM-L2-Transportzeit):*

$$Page\text{-Trefferrate}^{(1)} \cdot Page\text{-Mode-Transportzeit}^{(2)} + Page\text{-Miss-Rate}^{(3)} \cdot Random\text{-Transportzeit}^{(4)}.$$

bzw. (genauer, falls Speicheranordnung aus mehreren Banks besteht):

$$Page\text{-Trefferrate} \cdot Page\text{-Mode-Transportzeit} + Row\text{-Miss-Rate} \cdot (Page\text{-Mode-Transportzeit} + Row\text{-Miss-Leadoff}) + Page\text{-Miss-Rate} \cdot (Page\text{-Mode-Transportzeit} + Page\text{-Miss-Leadoff}).$$

- 1) Angabe als Wahrscheinlichkeit (Wert zwischen 0 und 1; = %-Wert : 100). Typische Page-Mode-Trefferrate um 0,8 (80%).
- 2) gemäß Taktschema (z. B. 5-3-3-3). Takte zusammenzählen.
- 3) = 1 - Page-Mode-Trefferrate.
- 4) gemäß Taktschema (Page-Mode-Transportzeit + Page-Miss-Leadoff; z. B. 12-3-3-3).

Tabelle 7.6 veranschaulicht, welcher Zuwachs an Leistung sich durch verschiedene Verbundlösungen aus L2-Cache und Arbeitsspeicher erzielen läßt.

Speicher- konfiguration	Konfiguration 1 (66/100 MHz) <sup>1)</sup>		Konfiguration 2 50/75 MHz <sup>2)</sup>	
	Benchmark <sup>3)</sup>	Leistungsgewinn	Benchmark <sup>3)</sup>	Leistungsgewinn
FPM ohne Cache <sup>5)</sup>	179	_ <sup>4)</sup>	137	_ <sup>4)</sup>
EDO ohne Cache <sup>6)</sup>	189	5,6%	143	4,4%
FPM + 256 kBytes SRAM- Cache (asynchron) <sup>7)</sup>	208	16,2%	156	13,9%
FPM + 256 kBytes PBSRAM-Cache <sup>8)</sup>	219	22,3%	165	20,4%
EDO + 256 kBytes PBSRAM-Cache <sup>8)</sup>	220	22,9%	165	20,4%

1)...8): siehe Erklärung im Text

**Tabelle 7.6** Speicherkonfigurationen im Leistungsvergleich (nach: Cypress Semiconductor; Quelle: Electronic Design/February 5, 1996)

*Erklärung:*

Man hat 2 PC-Systeme mit verschiedenen Speicherkonfigurationen bestückt und jeweils die Systemleistung mit einem Benchmark-Program gemessen.

- 1) Pentium, 66 MHz Bustakt, 100 MHz interner Prozessortakt,
- 2) Pentium, 50 MHz Bustakt, 75 MHz interner Prozessortakt,
- 3) Ziff-Davis Labs PC Bench 9.0 CPUmark16. Je größer die Zahl, desto höher die Leistung.
- 4) Bezugspunkt ( $\Delta$  100%),
- 5) Page Hit: 5-3-3-3, Page Miss: 9-3-3-3,
- 6) Page Hit: 5-2-2-2, Page Miss: 9-2-2-2,
- 7) asynchroner SRAM-Cache: 3-2-2-2,
- 8) PBSRAM-Cache: 3-1-1-1.

*Zur Interpretation der Ergebnisse:*

- nur EDO anstelle von FPM-DRAM bringt lediglich 4...5% mehr Systemleistung,
- der Vorteil des PBSRAM-Caches ist offensichtlich (hat seine Ursache aber darin, daß auf den verwendeten Motherboards keine schnelle Ansteuerung asynchroner SRAMs vorgesehen war (vor allem bei 50 MHz Bustakt könnte man SRAM-Caches durchaus noch ohne Wartezustände (2-1-1-1) betreiben - der Aufwand dürfte aber höher liegen als bei Einsatz von PBSRAMs).
- wenn wir den FPM-DRAM beibehalten, aber einen Cache vorsehen, so steigt die Systemleistung um 13...> 20%,
- die Kombination FPM-DRAM/PBSRAM-Cache ist immerhin ca. 16% schneller als eine EDO-Konfiguration ohne L2-Cache,
- Folgerung: ein leistungsfähiger L2-Cache ermöglicht es, mit einem langsameren (aber preisgünstigeren) Arbeitsspeicher auszukommen. Die Gesamt-Kosten (L2-Cache + billigere DRAMs) sind hierbei oft geringer (im Vergleich zur Ausstattung des Arbeitsspeichers mit schnellen, aber teureren DRAMs). Diese Überlegungen gelten sinngemäß auch bei höheren Taktfrequenzen (sie sind nicht deshalb überholt, nur weil die als Beispiel verwendeten Konfigurationen vielleicht 2, 3 Jahre alt sind).

Tabelle 7.7 soll eine weitere Anforderung an das Speichersubsystem veranschaulichen.

Speicheranordnung	interner Cache (L1) <sup>6)</sup>	externer Cache (L2)	DRAM; $t_{RAC} = 70 \text{ ns}$	DRAM; $t_{RAC} = 50 \text{ ns}$
Zugriffshäufigkeit <sup>1)</sup>	80%	16%	4%	4%
<i>Prozessor: 486DX, 33 MHz (Prozessortakt = Bustakt)</i>				
Bus-Takte je Burst-Zyklus	5 (2-1-1-1)	5 (2-1-1-1)	11 (5-2-2-2)	11 (5-2-2-2)
Prozessor-Takte je Burst-Zyklus <sup>2)</sup>	5	5	11	10
Zeitanteil der Burstzugriffe bei $t_{RAC} = 70 \text{ ns}$ <sup>3)</sup>	76,3%	15,3%	8,4%	-
Zeitanteil der Burstzugriffe bei $t_{RAC} = 50 \text{ ns}$ <sup>3)</sup>	77%	15,3%	-	7,7%
<i>Prozessor: 486DX4, 33 MHz Bustakt, 100 MHz Prozessortakt (Taktvervielfachung 1:3)</i>				
Bus-Takte je Burst-Zyklus <sup>4)</sup>	5 (2-1-1-1)	5 (2-1-1-1)	11 (5-2-2-2)	11 (5-2-2-2)
Prozessor-Takte je Burst-Zyklus <sup>7)</sup>	5 <sup>6)</sup>	15	33	30
Zeitanteil der Burstzugriffe bei $t_{RAC} = 70 \text{ ns}$ <sup>3)</sup>	52%	31%	17%	-
Zeitanteil der Burstzugriffe bei $t_{RAC} = 50 \text{ ns}$ <sup>3)</sup>	52,5%	31,5%	-	16%

1)...7: siehe Erklärung im Text

**Tabelle 7.7** Speicherzugriffe im Vergleich. Oben: Prozessor ohne, unten: Prozessor mit interner Taktvervielfachung (nach: Micron)

*Erklärung:*

Es geht darum, zu untersuchen, welchen Einfluß die (heutzutage allgemein übliche) interne Taktvervielfachung in den Prozessoren auf das Speichersubsystem hat. Die Prozessoren sind hier noch altmodischer (als jene von Tabelle 7.6) - aber die Erkenntnisse sind nach wie vor aktuell.

- 1) Beispiel: Trefferrate beider Caches (L1, L2) jeweils 80%. Bei einer L1-Trefferrate von 80% werden nur 20% aller Prozessor-Zugriffe über den Prozessorbus geführt. Hiervon betreffen wiederum 80% den L2-Cache. 80% von 20% sind 16%, somit verbleiben 4% aller Zugriffe als "echte" DRAM-Zugriffe.
- 2) 1 Bustakt  $\triangleq$  1 Prozessortakt,
- 3) alle Zugriffe zusammen  $\triangleq$  100%,
- 4) der Bustakt ist der gleiche wie im darüberstehenden Beispiel,
- 5) 1 Bustakt  $\triangleq$  3 Prozessortakte,
- 6) dies ist eine vereinfachende Annahme: der Prozessor greift auf den L1-Cache auch intern in Burst-Zyklen zu (nach dem Schema 2-1-1-1). Tatsächlich sind viele interne Cache-Zugriffe wahlfreie Zugriffe, die jeweils nur einen Prozessortakt erfordern. Die Folgerung, um die es hier geht, bleibt aber dieselbe.
- 7) bei Treffer im L1-Cache ist nur der interne Takt des Prozessors maßgebend (es gibt dann keine Buszugriffe).

Aus der Tabelle ist ersichtlich, daß der prozentuale Anteil der DRAM-Zugriffe (an der Gesamt-Zeit, die alle Zugriffe erfordern) bei Prozessoren mit interner Taktvervielfachung deutlich höher ist (z. B. 16% gegenüber ca. 8%). (Das ist auch ohne weiteres einzusehen: je schneller es intern zugeht, um so mehr macht sich eben die "Zeitverschwendung" bei externen Zugriffen bemerkbar.)

*Folgerung:*

Prozessoren mit interner Taktvervielfachung erfordern schnellere Speicherzugriffe bzw. höhere Speicher-Datenraten (wobei es allerdings nicht nur auf MBytes/s (bzw. GBytes/s) ankommt, sondern auch auf eine angemessene Verringerung der Leadoff- bzw. Latenzzeiten).

### 7.1.4.2. Entwicklungsziele

Dem Stand der Technik gemäß ist das Speichersubsystem gleichsam das schwächere Glied der Kette: der Speicher bremst sozusagen den Prozessor. Deshalb strebt man höhere Bus-Taktfrequenzen (> 66 MHz) und Speicher-Datenraten an. Mit beherrschbaren (allerdings auch nicht gerade hausbackenen) Technologien sind bis zu 100 MHz realisierbar (was in 1-1-1-Zyklen mit 64-Bit-Worten bis zu 800 MBytes/s ermöglicht). Das derzeitige Endziel liegt bei 1,6...2 GBytes/s. Hierfür stehen an Grundsatzlösungen zur Wahl:

- a) als evolutionäre Weiterentwicklungen eingeführter Technologien:
  - SDRAMs mit doppelter Datenrate (durch Nutzung beider Taktflanken); SSTL-Interface, PLL-Taktverteilung,
  - Multibank-DRAMs (MDRAMs).
  
- b) als Lösungen auf Grundlage teils neuartiger Prinzipien:
  - Rambus-DRAMs,
  - SLDRAMs (früher: SyncLink-DRAMs).

### 7.1.4.3. Speicherkapazität

Tabelle 7.8 nennt Praxisempfehlungen für eine sinnvolle Speicherausstattung typischer PCs.

Betriebssystem	Anwendungssoftware	Sinnvolle Mindest-Ausstattung	
		L2-Cache	Arbeitsspeicher
Windows 3.x/DOS	16-Bit-Programme	128 kBytes	16 MBytes
Windows 95 bzw. OS/2	16-Bit- und 32-Bit-Programme	256 kBytes (512 kBytes bei Multimedia-Anwendungen)	32 MBytes
Windows NT	32-Bit-Programme, Multitasking	512 kBytes	64 MBytes

**Tabelle 7.8** Zur sinnvollen Speicherausstattung typischer PCs (nach: Samsung)

*Erklärung:*

Viele Aussagen zur Speicherausstattung (wie man sie u. a. in Zeitschriften, in Software-Handbüchern usw. findet) betreffen jeweils *gerade noch betriebsfähige* Minimalkonfigurationen. In Tabelle 7.8 sind hingegen *empfehlenswerte* Mindestausstattungen angegeben, also Konfigurationen, die es ermöglichen, mit der jeweiligen Betriebs- und Anwendungssoftware vernünftig zu arbeiten, die aber andererseits nicht übermäßig teuer sind.

#### *Hinweise zum L2-Cache:*

1. 256 kBytes sind für herkömmliche Anwendungen durchaus genug (das betrifft den Einzel-PC, an dem ein Nutzer vor dem Bildschirm sitzt und an dem typischerweise zu einer Zeit nur mit einem Anwendungsprogramm gearbeitet wird).
2. Bei Multimedia-Anwendungen sind hingegen 512 kBytes auch im Einzel-PC sinnvoll (weil hier das Bewegen größerer Datenblöcke leistungsentscheidend ist).
3. Beim "echten" Multitasking wird der Cache von mehreren Anwendungen gleichzeitig belegt, wodurch sich die Trefferrate aus Sicht der einzelnen Anwendung deutlich verringern kann. Hier können auch Caches mit 1...4 MBytes ihr Geld wert sein.
4. Die ..86-Prozessorarchitektur (Intel's IA32) hat es an sich, daß 32-Bit-Programme (Stichwort: Protected-Modus) auffallend mehr Speicherplatz benötigen als 16-Bit-Programme (Stichwort: Realmodus) - auch bei an sich gleichem Funktionsumfang. Infolge dessen weisen kleinere Caches im Protected-Modus schlechtere Trefferraten aus als im Realmodus. Somit ist bei intensiver Nutzung von Protected-Mode-Software (Windows NT) ein größerer L2-Cache (wenigstens 512 kBytes) auch im Einzel-PC zweckmäßig.
5. Assoziativität ist wichtiger als Speicherkapazität. Ein 2-fach blockassoziativer Cache hat praktisch die gleiche Trefferrate wie ein direktabbildender Cache mit doppelter Speicherkapazität (bei Multitasking-Betrieb wird die Überlegenheit noch deutlicher).

#### *Hinweise zum Arbeitsspeicher:*

1. Arbeitsspeicherkapazität ist oftmals wichtiger als Prozessor-Leistung (Abbildung 7.1).
2. Speicherkapazität ist wichtiger als Datenrate ( so sind z. B. 64 MB EDO besser als 32 MB SDRAM).
3. Grundsätzlich: soviel Speicherkapazität wie möglich (bzw. erschwinglich) installieren, hierzu aber die jeweils kostengünstigsten Speichertypen (bzw. Moduln) einsetzen - auch bei Verzicht auf extreme Datenraten.

**Abbildung 7.1** Speicherweiterung oder neuer Prozessor (Memory Upgrade vs. Processor Upgrade; Samsung)

#### *Erklärung:*

Es handelt sich um Benchmark-Meßergebnisse (je größer die Zahl, desto höher die Leistung). Test-Voraussetzungen: Windows NT, 256 kBytes L2-Cache, Benchmark Winstone 32.

- a) ein 100-MHz-Pentium mit 32 MBytes DRAM leistet praktisch dasselbe wie ein 200-MHz-PentiumPro mit 16 MBytes DRAM. Die Erweiterung von 16 auf 32 MBytes kostet im Beispiel 96\$, das PentiumPro-System ist hingegen 456\$ teurer als das Pentium-System (mit ebenfalls 16 MBytes).
- b) bei gleicher Prozessorarchitektur (Pentium) sind die 64-MBytes-Systeme wenigstens genauso leistungsfähig wie 32-MBytes-Systeme mit dem nächst-schnelleren Prozessor.
- c) das Leistungsvermögen des Prozessors kommt bei zu geringer Speicherausstattung gar nicht zur Wirkung.

### **7.1.4.4. Speicherausstattung und Betriebssystem**

Bei Nutzung moderner Betriebssysteme (OS/2, Windows) hängt die Systemleistung direkt von der installierten Speicherkapazität ab. Diese Systeme nutzen das Prinzip des seitenorientierten virtuellen Speichers und legen sog. Swap-Dateien (Auslagerungsdateien) an, um im Speicheradreßraum auf ausführbare Programme und auf temporäre Daten zugreifen zu können (das sind Daten, die nur zeitweilig benötigt werden, aber stets verfügbar sein müssen). Je mehr RAM zur Verfügung steht, um so weniger ist es notwendig, Seiten des virtuellen Speichers auf die Festplatte auszulagern (Page Swapping) - selbst der "langsamste" FPM-DRAM ist viel schneller als die schnellste Festplatte.

### *Praxistips:*

1. Anzeichen dafür, daß es Zeit ist, mehr DRAM zu installieren: merklich schlechte Systemleistung, System reagiert irgendwie "zähe", dabei werden beständig Zugriffe auf die Festplatte ausgeführt (an der Aktivitäts-LED erkennbar), ohne daß der Nutzer Dateizugriffe veranlaßt hat. Das Einbauen eines schnelleren Prozessors nützt gar nichts!
2. Es gibt Dienstsoftware, mit der man sich die Auslastung des Arbeitsspeichers und des Dateisystems ansehen kann (z. B. Windows 95 Systemmonitor).

### *Grenzen der Speichererweiterung:*

- die technische Ausstattung des Motherboards mit Steckfassungen für Speichermoduln,
- die Speicherkapazität des einzelnen Moduls,
- das Adressierungsvermögen der Motherboard-Schaltkreise (einschließlich der Anzahl der Adreßleitungen zu den DRAM-Moduln),
- das Abbildungsvermögen des L2-Caches.

Die ersten 3 Punkte sind bereits ausführlich erörtert worden (vgl. Kapitel 4, Abschnitt 4.6.). Zum letzten Punkt (vgl. Kapitel 3, Abschnitt 3.1.3.8.) folgende Anmerkung:

Der maßgebende Kennwert heißt "Cacheable Limit", "Cacheable Memory Size" oder ähnlich. Typische Werte hierfür sind u. a. 64 MBytes, 512 MBytes oder 1 GBytes. Es kann sein, daß das Adressierungsvermögen größer ist als das Abbildungsvermögen (beispielsweise können bis zu 128 MBytes DRAM bestückt werden, aber nur die ersten 64 MBytes des Speicheradreßraumes werden in den L2-Cache abgebildet).

*Die Folge:* eine installierte DRAM-Kapazität oberhalb des "Cacheable Limit" wird nicht die an sich erwartete Leistungssteigerung bewirken. Sie kann sogar zu Leistungseinbußen führen.

### *Die Zusammenhänge:*

- der vom L2-Cache abgebildete ("cacheable") Teil des Speicheradreßraumes beginnt typischerweise mit Adresse 0,
- typische Betriebssysteme belegen aber den Speicher vom oberen Ende her (und laden dabei zunächst "sich selbst"),
- werden Zugriffe auf dieses obere Ende aber (wegen Überschreitung des "Cacheable Limits") nicht mehr über den L2-Cache geführt, so kann es sein, daß ausgerechnet leistungsbestimmende Systemfunktionen nur noch "mit DRAM-Geschwindigkeit" ausgeführt werden. Beispiel: bei einem "Cacheable Limit" von 64 MBytes installieren wir 96 MBytes DRAM. Zugriffe auf die obersten 32 MBytes werden dann nicht mehr über den L2-Cache geführt. Dann kann die Leistung im Vergleich zu einer Ausstattung von 64 MBytes (wo aber *alle* Zugriffe über den L2-Cache laufen) merklich absinken.

## 7.1.5. RAS-Vorkehrungen

Speicherschaltkreise und Speichermoduln haben eine erstaunlich hohe Zuverlässigkeit (Abbildung 7.2 - die Werte stammen allerdings von einem Hersteller, der für seine hohe Qualität bekannt ist (bei ausgesprochenen Billigprodukten ist durchaus mit wesentlich höheren Fehlerraten zu rechnen)).

**Abbildung 7.2** Fehlerabstände (MTBF) von Speichermoduln (Micron). \*): gepufferte Moduln; die Pufferstufen verringern (zusätzliche Bauelemente und Lötstellen) die MTBF in gewissem Maße; \*\*): Erklärung weiter unten im Text

### *Soft Errors*

Die Häufigkeit gelegentlicher Datenverluste in DRAMs (Soft Error Rate SER) konnte im Verlauf der Entwicklung drastisch gesenkt werden (Abbildung 7.3). Hierdurch wurde es möglich, in vielen Systemen auf Fehlerkorrekturvorkehrungen (ECC) und auch auf Paritätskontrolle zu verzichten.

**Abbildung 7.3** Zur geschichtlichen Entwicklung der Soft Error Rate SER (Micron). Moderne DRAM-Schaltkreise haben eine Soft-Error-MTBF von einigen hundert Jahren

Mit Datenverlust (Soft Errors) ist um so häufiger zu rechnen, je öfter ein DRAM angesprochen wird. Ein DRAM, der gleichsam in Ruhe gelassen wird, ist kaum von Soft Errors betroffen. Die einschlägigen Fehlermechanismen werden vielmehr vorzugsweise dann wirksam, wenn die Inhalte der Speicherzellen ausgelesen und wieder zurückgeschrieben werden (vgl. Kapitel 5, Abschnitt 5.2.1.). Je häufiger auf den DRAM zugegriffen wird, um so höher ist die Fehlerwahrscheinlichkeit.

Das kommt auch in Abbildung 7.2 zum Ausdruck (vgl. die mit \*\*) gekennzeichneten Spalten):

- 100% Refresh heißt, daß alle 15,6  $\mu$ s ein Refreshzugriff erfolgt, ansonsten aber nicht auf den DRAM zugegriffen wird.
- 96% Refresh heißt, daß in 96% der Gesamtzeit lediglich die beschriebenen Refreshvorgänge ablaufen, in 4% der Gesamtzeit aber Lese- und Schreibzugriffe. Aber diese 4% genügen bereits, um die MTBF beachtlich absinken zu lassen (z. B. von 53 auf 35 Jahre).

### *Alphastrahlung*

Der Fehlermechanismus "Alphastrahlung" ist nach wie vor für Soft Errors verantwortlich (Abbildung 7.4).

**Abbildung 7.4** Zum Fehlermechanismus "Alphastrahlung" (Micron)

### *Erklärung:*

Die Kurve wurde mit sog. beschleunigten Testverfahren (Accelerated Tests) meßtechnisch ermittelt. (Die Überschrift "5 V Checkerboard Pattern" kennzeichnet ein bestimmtes Prüfmuster, mit dem 5-V-DRAMs getestet wurden.) Es ist die Rate der möglichen Einwirkung von Alpha-Teilchen (Alpha Hits) in Abhängigkeit vom Rhythmus der Zugriffe angegeben. Hierbei wird angenommen, daß die Zugriffe gemäß der Zykluszeit (Cycle Time) lückenlos aufeinander folgen. Die Werte der senkrechten Diagrammachse betreffen Soft Errors in einem bestimmten, nicht näher spezifizierten Betriebszeit-Intervall. Wesentlich ist, in welchem Verhältnis die Fehlerrate (SER) ansteigt, wenn anstelle der (unumgänglich notwendigen) Refresh-Zugriffe Nutz-Zugriffe in kürzeren Abständen ausgeführt werden.

- a) Zugriffe im Intervall von 200 ns (0,2  $\mu$ s). Hierbei ist mit 160 Soft Errors infolge von Alpha Hits zu rechnen.
- b) Zugriffe im Intervall von 15,6  $\mu$ s (Refresh-Zugriffe). Es ergeben sich 3,4 Soft Errors (Alpha Hits).

Dies ergibt das Verhältnis  $160 : 3,4 = 47$ . Um die SER einer im Rhythmus von 200 ns beanspruchten DRAM-Anordnung zu bestimmen, muß man also die Soft-Error-MTBF, die bei bloßem Refresh gilt, durch 47 dividieren (bzw. die Fehlerrate (z. B. in FIT) mit 47 multiplizieren). Hierdurch sinken beruhigend große

Fehlerabstände (von Jahrzehnten, vgl. Abbildung 7.2) doch auf besorgniserregend kurze Werte herab (eine MTBF von größenordnungsmäßig einem Jahr bedeutet praktisch, daß ein PC doch ab und zu wegen eines Soft Errors abstürzt).

*Hinweise:*

1. Die Erläuterungen zur Funktionsweise des DRAM legen folgende Auffassung nahe: “je öfter Refresh-Zugriffe erfolgen, desto besser (für den Datenerhalt)”. Die Abbildungen 7.4 und 7.5 zeigen, daß das Gegenteil der Fall ist (auch erklärlich: je öfter die Daten über die Spaltenleitungen, Leseverstärker usw. bewegt werden, um so öfter haben die Fehlermechanismen (Alpha-Teilchen, Störungen im System usw.) Gelegenheit zum Zuschlagen).
2. Sinngemäß wirkt sich ein L2-Cache vorteilhaft auf die Systemzuverlässigkeit aus (die weitaus meisten Zugriffe betreffen dann den L2-Cache (sind Cache Hits), so daß die DRAM längere Zeit “in Ruhe gelassen” werden).
3. Die Praxis zeigt, daß Stör-Mechanismen innerhalb des Systems wesentlich öfter an Soft Errors schuld sind als die Alpha-Teilchen. Zu diesen Stör-Mechanismen gehören Über- und Unterschwingen auf Signalleitungen, Übersprechen (Crosstalk), Schwankungen des Massepotentials (Ground Bounce), Speisespannungsschwankungen und auch Zeitabläufe, die außerhalb der Spezifikationen liegen (Toleranzprobleme bei “zu knappem Timing”, aber auch subtile Entwurfs- und Fertigungsfehler).
4. Das Überwachen der Speicherinhalte mittels Paritätsprüfung nützt bei derartigen Fehlermechanismen nicht viel (es könnte aber durchaus angebracht sein, statt dessen die Signalwege (Adressen, Daten) mit Paritätsbits zu überwachen - vgl. auch die einschlägigen Vorkehrungen z. B. an den Pentium-Prozessoren).
5. Stand der Technik (1): in PCs des Massen-Marktes kann man guten Gewissens (vorausgesetzt, die Hardware ist an sich sauber entworfen) auf die Paritätsprüfung der Speicherinhalte verzichten. *Praxistip:* das Speichersubsystem von Zeit zu Zeit mit einem wirklich scharfen Speichertestprogramm prüfen (kein Speichertest, der nach wenigen Sekunden schon eine o.k.- oder PASSED-Meldung bringt, kann wirklich scharf sein).
6. Stand der Technik (2): wenn es auf höhere Zuverlässigkeit ankommt, bevorzugt man ECC-Vorkehrungen. Deren Vorteile:
  - einzelne Soft Errors werden korrigiert, führen also nicht zum Abstürzen des Systems,
  - bei 8 ECC-Bits und 64 Datenbits wird der eigentliche Speicher nicht teurer als bei Paritätsprüfung (es ergeben sich sogar Vereinfachungsmöglichkeiten, z. B. das Zusammenfassen der 8 ECC-Bits in einem  $\cdot$  8-DRAM, da man auf byteweise Zugriffe keine Rücksicht mehr nehmen muß).
  - mit ECC werden mehr Fehlermechanismen erkannt als mit Paritätsprüfung.
  - man hat etwas, das man in der Werbung herausstellen kann.

Gute ECC-Lösungen sollten allerdings auch “rundum zufriedenstellend” ausgelegt sein: Fehlerprotokollierung (Error Logging), Memory Scrubbing, Testbarkeit der ECC-Hardware).

**Abbildung 7.5** Soft Errors in Abhängigkeit von der Zugriffshäufigkeit (Micron)

*Erklärung:*

- a) Soft-Error-MTBF (in Jahren; die Werte betreffen einen einzelnen DRAM-Schaltkreis),
- b) Anteil der Lese-Schreib-Zugriffe an der Gesamtzeit (0%: es finden ausschließlich Refresh-Zugriffe statt) .

## 7.2. Der Bildspeicher

### 7.2.1. Varianten der Auslegung

Der Bildspeicher (Frame Buffer) eines Videoadapters muß 2 Anforderungen praktisch gleichzeitig bedienen können:

1. Lieferung der zur Bilddarstellung notwendigen Videodaten (Screen Refresh). Dies sind reine Lesezugriffe mit regelmäßigen Adressierungsfolgen (die Pixel sind synchron zur rasterförmigen Ablenkbewegung des Elektronenstrahls oder zur seriellen Ansteuerung des Flachbildschirms zu liefern; das läuft typischerweise auf eine fortlaufende Adreßzählung mit Rückkehr zur Anfangsadresse (beim Bildrücklauf) hinaus. Die Anforderungen an die Datenrate ergeben sich aus der für das jeweilige Darstell-Raster erforderlichen Pixel-Frequenz und der Farbtiefe (Bits je Pixel).
2. Zugriffe zum Aufbauen und Ändern von Bildinhalten. Hierbei müssen wir wiederum verschiedene Auslegungen voneinander unterscheiden:
  - a) diese Funktionen werden ausschließlich vom Prozessor ausgeführt (es handelt sich also um einen "dummen" Video-Controller). An sich kommt man mit Schreibzugriffen aus, die einfache Blocktransporte (mit Adreßzählung) sind. Man könnte z. B. den Bildinhalt im Arbeitsspeicher aufbereiten und (während des Bildrücklaufs) mit einem einzigen Blocktransport in den Bildspeicher überführen. Aus Kostengründen betreibt man aber vielfach den Bildspeicher aus Sicht des Prozessors wie einen Ausschnitt des Arbeitsspeichers. Die Bilddarstellungssoftware führt also unmittelbar wahlfreie Zugriffe auf den Bildspeicher aus. Anspruchsvolle Bilddarstellungen (Bedienoberflächen mit Fenstern, 2D- und 3D-Graphik, Spiele, Bewegtbildsequenzen) können nur dann realisiert werden, wenn die Zugriffszeiten hinreichend kurz sind. Dabei muß aber der Vorrang der Bilddarstellung stets gewährleistet sein. Bei derartigen Anforderungen sind die Tricklösungen der "uralten", einfachsten Videoadapter nicht mehr anwendbar: weder kann man, um zum Bildspeicher zuzugreifen, die Rücklaufperioden abwarten, noch die Bilddarstellung einfach für kurze Zeit unterbrechen.
  - b) Prozessor und Video-Controller teilen sich in die Aufgabe. Hierbei übernimmt der Video-Controller bestimmte Grundfunktionen (wie z. B. das Einfärben von Flächen oder das "Umschauen" von Pixeln (BitBlit-Abläufe). Der Video-Controller heißt dann beispielsweise "Windows Accelerator" oder "Graphics Engine". In einer solchen Konfiguration gibt es also 3 verschiedene Anforderungen für Bildspeicher-Zugriffe: (1) zwecks Video-Darstellung (Screen Refresh), (2) seitens des Prozessors, (3) seitens der Beschleunigungsschaltungen des Video-Controllers).
  - c) es ist ein Video-Prozessor vorgesehen, der den Bildaufbau "in eigener Regie" durchführt und vom "eigentlichen" Prozessor nur Anweisungen ("Graphik-Kommandos") erhält.

### 7.2.2. Bildspeichergrößen

Dem Stand der Technik entsprechen - im Massen-Markt - Bildspeicher von 1, 2 und 4 MBytes.

### 7.2.3. Speichertypen

Grundsätzlich können alle Speichertypen, sofern sie die jeweiligen Anforderungen an Speicherkapazität und Zeitkennwerte erfüllen, als Bildspeicher verwendet werden (und man hat dies auch ausgiebig getan - es gibt Grafikkarten mit Multibank-DRAMs, mit Rambus-DRAMs, WRAMs usw.). In Tabelle 7.9 sind häufig eingesetzte Speichertypen angeführt.

Speichertyp	besondere Unterstützung für Graphik	Datenwegbreite (des einzelnen Speicherschaltkreises)	Anmerkungen
FPM-DRAM	-	1, 4, 8, 16 Bits	in älterer Hardware üblich
EDO-DRAM	-	4, 8, 16 Bits	üblich in preisgünstiger Hardware (betrifft im besonderen die Organisationsform 256k · 16 - 2 DRAMs bilden einen 1-MBytes-Bildspeicher). Taktfrequenzen bis 66 MHz (typisch sind max. 50 MHz)
SDRAM	-	4, 8, 16 Bits	Alternative zum SGRAM (Kosten). Taktfrequenzen von 83, 100 MHz (und mehr) möglich
SGRAM	bitselektives Schreiben, Block-Füll-Funktion	32 Bits	Taktfrequenzen von 83, 100 MHz (und mehr) möglich. Synchrone Betriebsweise mit 2 Banks. Praxisgerechte Zugriffsbreite (1 SGRAM mit 8 oder 16 MBits stellt einen Bildspeicher von 1 oder 2 MBytes dar)
Multibank-DRAM	-	16 Bits	Taktfrequenzen von über 100 MHz möglich. Besonders geringe Latenzzeiten bei konkurrierenden wahlfreien Zugriffen (sofern diese verschiedene Banks betreffen)
Speichertyp	besondere Unterstützung für Graphik	Datenwegbreite (des einzelnen Speicherschaltkreises)	Anmerkungen
Rambus-DRAM <sup>*)</sup>	bitselektives Schreiben	8 bzw. 9 Bits; maximal 2 Rambus-Kanäle	Datenrate 500 MBytes je Rambus-Kanal
Video-RAM (VRAM)	2. (serieller) Zugriffsweg (SAM), bitselektives Schreiben	parallel: 4, 8 oder 16 Bits, seriell: 4 oder 8 Bits	paralleler Zugriffsweg hat nur Zeitverhalten gemäß FPM bzw. EDO. Serieller Takt ca. um 40...50 MHz
Window-RAM (WRAM)	2. (serieller) Zugriffsweg (SAM), bitselektives Schreiben, Block-Füll-Funktion, Blocktransporte (BitBlt)	parallel: 32 Bits, seriell: 16 bits	Taktfrequenzen parallel und seriell bis zu 83 MHz

\*) vgl. auch die Weiter- und Konkurrenzentwicklungen Concurrent Rambus, Direct Rambus, SLDRAM (Kapitel 8)

**Tabelle 7.9** Speichertypen für Bildspeicher - eine Übersicht

## 7.2.4. Videoadapter-Strukturen und Speicherauswahl

Die Problematik der Bildspeicherkonfigurationen wollen wir anhand von 2 typischen Videoadapter-Auslegungen erläutern (Abbildungen 7.6, 7.7).

**Abbildung 7.6** Videoadapter-Strukturen (1)

*Erklärung:*

1- Steuerschaltkreis (Video Controller; heutzutage - weil's mehr hermacht: Graphics Engine), 2 - Pixel-Wandlung (ggf. weitere Serialisierung, Bildung der analogen Videosignale; andere Bezeichnung für RAMDAC: Farbpalette), 3, 4 - Bildspeicher (Beispiele). In beiden Fällen ist der Steuerschaltkreis über den jeweiligen Systembus (PCI, AGP o. ä.) mit dem Prozessor verbunden. Die Zahlenangaben an den Zugriffswegen bezeichnen lediglich die Anzahl der Datenbits.

- a) der Bildspeicher besteht aus 2 parallel angesteuerten Window-RAMs (WRAMs). Jeder serielle Zugriffsweg liefert Pixeldaten in 16 Bits Breite an den RAMDAC. Die maximale Taktfrequenz von 83 MHz ermöglicht folgende Video-Datenraten (am Video-Ausgang des RAMDAC): 83 MHz bei 32 Bits/Pixel, 166 MHz bei 16 Bits/Pixel, 332 MHz bei 8 Bits/Pixel. Auch der parallele Zugriffsweg kann im Rahmen eines 83-MHz-Taktrasters angesteuert werden. Es werden mindestens 8 Pixel (bei 32 Bits/Pixel) parallel vom DRAM in die SAM-Anordnung übernommen. Diese Übernahme kostet nur einen Taktzyklus (12 ns). Demzufolge stehen von 8 Takten 7 für prozessorseitige Zugriffe zu r Verfügung. Ähnliche Konfigurationen kann man mit "gewöhnlichen" Video-RAMs (VRAMs) aufbauen.
- b) der Bildspeicher besteht aus DRAMs, die nur einen einzigen Zugriffsweg haben (ohne grundsätzliche Typenbeschränkung; das Blockschaltbild gilt also für FPM, EDO, SDRAM, SGRAM usw.). Um konkurrierende Zugriffe (Prozessor + Bilddarstellung) zu ermöglichen, aber eine lückenlose Bilddarstellung zu gewährleisten, werden die videoseitigen Lesezugriffe über eine FIFO-Anordnung im Steuerschaltkreis geführt. Die Steuerung versucht, diese stets gefüllt zu halten. Greift der Prozessor auf den Bildspeicher zu, so wird die Bilddarstellung aus dem FIFO heraus aufrecht erhalten. Droht die FIFO-Anordnung leer zu laufen, werden weitere Prozessor-Anforderungen zurückgestellt. Die Datenrate des Speichers muß größer sein als für die Bilddarstellung an sich notwendig, um die FIFO-Anordnung rechtzeitig nachfüllen zu können.

**Abbildung 7.7** Videoadapter-Strukturen (2); (Rambus, Inc.)

*Erklärung:*

Die Anzahl der Anschlüsse am Controllerschaltkreis ist ein bedeutsamer Kostenfaktor.

- a) Video-Subsystem mit 2 Rambus-Kanälen und je einem Rambus-DRAM. Ermöglicht extreme Datenraten und kommt mit vergleichsweise wenigen Anschlüssen/Signalwegen aus (hier: 20 Signale - s. auch den folgenden Hinweis).
- b) Video-Subsystem mit 2 SGRAMs. Ersichtlich werden wesentlich mehr Anschlüsse/Signalwege benötigt (hier: 84 Signale)

*Hinweis:*

Man hat hier nur jene Signale zusammengezählt, die im Normalbetrieb zur Adressierung und Datenübertragung dienen (so wurden beispielsweise Taktsignale nicht berücksichtigt). Ein Rambus-Kanal an sich umfaßt insgesamt (bei 9 Datenbits) 15 Signale und belegt 32 Schaltkreis-Anschlüsse bzw. Leiterzüge (zu Einzelheiten siehe Kapitel 8, Abschnitt 8.3.). Der Anschluß- und Leitungsbedarf einer SGRAM-Anordnung richtet sich nach der Zugriffsbreite und den Einzelheiten der Auswahlorganisation. (Vgl. hierzu die Auslegung der verschiedenen Speichermoduln (Kapitel 5, Abschnitt 5.7.). In Video-Hardware werden vorzugsweise 144-polige SODIMMs eingesetzt.

## 7.2.5. Moderne Video-Subsysteme

Die Videohardware muß mehr leisten als nur Bytes bzw. Pixel weiterzureichen. Typische leistungsentscheidende Graphikfunktionen (wie sie z. B. für 3-dimensionale Darstellungen, Bewegtbildsequenzen usw. erforderlich sind) erfordern schnelle wahlfreie Zugriffe auf kleine Ausschnitte des Bildinhalts. Der Bildspeicher sollte deshalb eine möglichst hohe überschüssige Datenrate (über die Erfordernisse der Video-Darstellung hinaus) aufweisen, und zwar bei möglichst geringen Latenz- bzw. Leadoff-Zeiten. Abbildung 7.8 ist das Ergebnis eines diesbezüglichen Vergleichs.

**Abbildung 7.8** Bildspeicher im Leistungsvergleich (Rambus, Inc.)

### *Erklärung:*

Es wurde das Leistungsvermögen hinsichtlich der Beschleunigung typischer Grundoperationen der 3D-Darstellung verglichen (im Sinne eines Benchmarks).

- Bezugspunkt ist ein Video-Subsystem mit SGRAMs, das mit 100 MHz betrieben wird (Leistung = 100% bzw. 1,0),
- ein Video-Subsystem mit Rambus-DRAM (RDRAM) erbringt demgegenüber eine 2,34-fache Leistung (Logik-Takt 76 MHz, Rambus-Takt 600 MHz (technisch: 300 MHz + Nutzung beider Taktflanken),
- die Leistung eines mit EDO-DRAMs bestückten Video-Subsystems ist deutlich geringer (das 0,6-fache der SGRAM-Lösung), da man die Taktfrequenz kaum über 50 MHz hinaus erhöhen kann (66 MHz bilden die endgültige Obergrenze).

Ein Video-Subsystem mit WRAMs (vgl. Abbildung 7.6a) würde leistungsmäßig zwischen dem Rambus- und dem SGRAM-System liegen.

## 7.2.6. Der Bildspeicher im Arbeitsspeicher: UMA

UMA = Unified Memory Architecture. Der Bildspeicher ist Teil des Arbeitsspeichers, und der Video-Controller wirkt mit dem Motherboard-Schaltkreissatz zusammen (Prinzip des "Cycle Stealing" - immer dann, wenn es notwendig ist, den Pixel-FIFO (vgl. Abbildung 7.6b) nachzufüllen, fordert der Video-Controller den Arbeitsspeicher an - und erhält dann Vorrang vor ggf. gleichzeitig abhängigen Anforderungen des Prozessors). Diese Lösung wurde von einigen Herstellern im Sinne der Kostenminimierung propagiert (es gibt auch einen einschlägigen VESA-Standard: VUMA).

Größenordnung der Kostensenkung (je PC): 100...200 DM. Kostenminimierung führt aber auch zu Leistungsverlust:

- die Videozugriffe behindern die Zugriffe des Prozessors: hierdurch ergeben sich auf Seiten des Prozessors bis zu 25% Einbuße an Datendurchsatz (über den Prozessorbus). Ein L2-Cache vermindert diesen Wert auf etwa 10%.
- die Prozessorzugriffe behindern die Zugriffe des Video-Controllers: das bremst im besonderen Funktionen der Bildaufbau-Beschleunigung. Richtwert: Leistungsverlust bei "anspruchsvollen" Videodarstellungen (bewegte Bilder, 3D) um ca. 30% (ruckartige Bewegungen, spürbar langsamer Bildaufbau).
- die für den Bildspeicher "abgezwigte" DRAM-Kapazität steht nicht mehr als Arbeitsspeicher zur Verfügung: dies veranlaßt das Betriebssystem, mehr Dateien und Programme auf die Festplatte auszulagern (Swapping; vgl. Abschnitt 7.1.4.4.). Richtwert: das Verkleinern des Arbeitsspeichers um 1 MBytes kann die Systemleistung durchaus um ca. 15% verringern (merklich bei an sich geringer Speicherausstattung (z. B. 8 MBytes und Windows 95)).

*Hinweis:*

Die Leistungsschwäche ist durch Wirkprinzip und Entwicklungsziel (Cycle Stealing, Kostenminimierung) bedingt, nicht aber grundsätzlich dadurch, daß man den Bildspeicher als Teil des Arbeitsspeichers vorsieht (also auf eine "Grafikkarte" vollkommen verzichtet). Ist Leistungsvermögen wichtiger als Kostensenkung, kann man auf dieser Grundlage sogar Systeme bauen, deren Video-Leitungsvermögen allen anderen Lösungen überlegen ist, auch Systemen mit AGP-Bus. (Allein schon deshalb, weil Transporte zwischen Arbeits- und Bildspeicher vermieden werden. Auch geht es meist schneller, wenn der Prozessor, sofern er dafür ausgerüstet ist (Stichwort: MMX-Befehlssatz), die Arbeit selbst tut, anstatt eine andere Einrichtung (3D-Beschleuniger, "Graphics Accelerator") damit zu beauftragen.) Wenn von "Unified Memory" die Rede ist, sollten wir also genauer hinsehen, was man jeweils darunter versteht: entweder das beschriebene UMA-Konzept (bzw. den VUMA-Standard) oder allgemein eine Systemstruktur, in der der Bildspeicher einen Teil des Arbeitsspeicher-Adreßraums belegt und über den Arbeitsspeicher-Datenweg zugänglich ist.

### 7.2.7. Tendenzen

Video-Subsystem mit asynchronen DRAMs wird wohl kaum noch jemand entwickeln. Der bevorzugte Speichertyp des Massen-Marktes dürfte wohl zunächst der SGRAM werden (ob man "gewöhnliche" SDRAMs auch als Bildspeicher einsetzt, ist eine reine Kostenfrage). Im oberen Leistungsbereich werden derzeit Rambus-DRAMs eingesetzt (z. B. in Workstations, aber auch in Spiele-Konsolen<sup>\*)</sup>). Künftig wird man die Speicherarchitektur mitnutzen, die sich allgemein durchgesetzt hat (Direct Rambus oder SLDRAM; vgl. Kapitel 8). Der Einsatz von "Exoten" (WRAMs, Multibank-DRAMs) hängt von den Kosten und der Verfügbarkeit ab. Die Systemauslegung ("echte" Videokarten, womöglich mit Beschleunigungs-Hardware bis hin zum autonomen Video-Prozessor, "Mischlösungen" (wie AGP) und UMA-Strukturen) ist aus technischer Sicht zunächst eine Geschwindigkeitsfrage: kann die Hardware beide Anforderungen (Prozessor- und Video-Zugriffe) schnell genug bedienen oder nicht? - Die einschlägigen Anforderungen lassen sich näherungsweise durch "Bandbreiten"-Angaben (in MBytes/s) charakterisieren (Tabelle 7.10).

\*) : so hat die Spiel-Konsole Nintendo 64 (ausgerüstet mit einem 64-Bit-Prozessor MIPS RS4300i) ein "vereinigtes" (unified) Rambus-Speichersubsystem, das lediglich zwei 16-MBit-RDRAMs enthält (es kann durch Stecken zusätzlicher Speichermoduln erweitert werden).

Zugriffe	durchschnittliche Bandbreite (Sustained Bandwidth)	Spitzenbedarf (Peak Bandwidth)
vom Prozessor <sup>1)</sup>	133 MBytes/s <sup>2)</sup>	533 MBytes/s <sup>3)</sup>
Bilddarstellung (Screen Refresh) <sup>4), 6)</sup>	160 MBytes/s	160 MBytes/s
2D-Graphikoperationen <sup>5), 6)</sup>	50 MBytes/s	50 MBytes/s
3D-Graphikoperationen <sup>5), 6)</sup>	300 MBytes/s	300 MBytes/s
bewegte Videobilder (MPEG) <sup>5), 6)</sup>	150 MBytes/s	150 MBytes/s
Gesamt-Bandbreite eines "vereinigten" Speichersubsystems (UMA) <sup>7)</sup>	593 MBytes/s <sup>8)</sup>	993 MBytes/s <sup>9)</sup>

1)...9): siehe Erklärung im Text

**Tabelle 7.10** Anforderungen an die Speicherbandbreite (anhand eines typischen PCs des Massen-Marktes; nach: Rambus, Inc.)

*Erklärung:*

1) den Zahlenwerten liegt ein Pentium mit 133 MHz interner Taktfrequenz und 66 MHz Bustakt zugrunde. Ein L2-Cache wird vorausgesetzt.

- 2) Richtwert für die durchschnittliche Speicherbandbreite, die ein Prozessor benötigt, um “vernünftig” arbeiten zu können: 1 MBytes/s je MHz Taktfrequenz. (Nach: Rambus, Inc. Mit Vorsicht zu genießen. Dürfte nur bei Vorhandensein eines brauchbaren L2-Cache zutreffen.)
- 3) Spitzenbedarf ergibt sich bei Cache Miss im L2-Cache (das 4-fache der durchschnittlichen Bandbreite, um den Cache-Eintrag ohne zusätzliche Wartezustände transportieren zu können),
- 4) betrifft eine Anzeige von 1024 · 768 Pixeln bei 16 Bits Farbtiefe und einer Bildfrequenz  $\geq 70$  Hz (wobei die Bandbreite nur in den Darstellperioden gefordert wird - während der Zeilen- und Bildrücklaufperioden gibt es keine derartigen Zugriffe),
- 5) betrifft Zugriffe der Beschleunigungshardware des Videocontrollers + Prozessor-Zugriffe zwecks Bildaufbau,
- 6) bei Bildspeicherzugriffen kann man typischerweise nicht von einer Spitzenbelastung sprechen, weil der Bildinhalt immer wieder regeneriert bzw. in Bewegung gehalten werden muß. Verbleibt eine Bilddarstellung in Ruhe, so entfallen die Zugriffe gemäß Punkt 5,
- 7) ein “vereinigtes” (unified) Speichersubsystem müßte diese Datenraten gewährleisten. Wir haben hier angenommen, daß das Video-Subsystem bewegte 3D-Bilder darstellt (woraus sich die höchste Anforderung an die Speicher-Bandbreite ergibt).
- 8) mit SDRAM/SGRAM “gerade so” zu schaffen (64 Bits bei 66 MHz  $\triangleq$  533 MBytes/s),
- 9) nur mit DDR, Rambus oder SDRAM zu schaffen.

Abbildung 7.9 soll die Entwicklungstendenzen anhand der Auffassung eines Anbieters veranschaulichen (in der Richtung - hin zu höheren Bandbreiten - sind sich alle Hersteller einig, nicht aber in den technischen Lösungen hierfür).

**Abbildung 7.9** Tendenzen der Bildspeicherentwicklung (Rambus, Inc.)

*Erklärung:*

- a) Rambus und SGRAM im Vergleich,
- b) Video-Subsysteme auf Rambus-Grundlage. Die geringe Zahl der Anschlüsse bzw. Leiterzüge ermöglicht es, ohne weiteres 2 und - wenn nötig - auch 4 Rambus-Kanäle vorzusehen, die jeweils parallel betrieben werden. Siehe auch Tabelle 7.11.

Rambus-Kanäle	Bildauflösung (Pixel)	Bandbreite für Bildaufbau <sup>*)</sup>	Bildspeicher
1	800 · 600 · 8	250...300 MBytes/s	1 8-MBit-RDRAM
	800 · 600 · 16		
	1024 · 768 · 8		
	800 · 600 · 24	200...250 MBytes/s	1 16-MBit-RDRAM
	1024 · 768 · 16		
	1280 · 1024 · 8		
2	1024 · 768 · 24	600...700 MBytes/s	2 16-MBit-RDRAMs
	1280 · 1024 · 16/24		

\*) : diese verbleibt nach Abzug jener Bandbreite, die zwecks Bilddarstellung (Screen Refresh) erforderlich ist

**Tabelle 7.11** Video-Subsysteme auf Rambus-Grundlage (Beispiele; nach: Rambus, Inc.)

# 8. Entwicklungstendenzen

## 8.1. Extreme Speicherkapazitäten

Die DRAM-Massenfertigung ist derzeit durch 16-MBit- und 64-MBit-Typen gekennzeichnet; die Einführung von 256-MBit-Typen ist eine Zeitfrage - und unterliegt Nützlichkeits-Erwägungen:

- wenn man mit dem Alten noch genügend Geld verdienen kann (oder wenn sich gar die Kosten noch nicht amortisiert haben), hat man es mit dem Neuen nicht so eilig,
- die modernen Technologien können mehr leisten, als der Markt abzunehmen (und zu bezahlen) gewillt ist. Der einzelne 256-MBit-Schaltkreis entspricht 32 MBytes - und man wird in einer Umgebung mit 64 Bits breiten Datenwegen wenigstens 2 Stück (in  $\cdot$  32-Organisation) einsetzen müssen. Das ergibt eine Mindest-Speichergröße von 64 MBytes, und die Speicherkapazität kann auch nur in dieser "Stückelung" (Granularity) erweitert werden. Da die Speicher keineswegs Pfennigartikel sind, ergeben sich Kosten und Preis-Abstufungen, die für die weitaus meisten Anwender viel zu hoch sind. Deshalb ziehen dies die Schaltkreishersteller vor, zunächst die 256-MBit-Technologie zu nutzen, um damit 16- und 64-MBit-Typen herzustellen (Verbilligung, da je Schaltkreis weniger Silizium benötigt wird).

Im Labor hat man bereits Speicherschaltkreise mit 1 GBits und 4 GBits zum Funktionieren gebracht (mit wenigen solcher Schaltkreise könnte man - was die Speicherkapazität angeht - ohne weiteres eine CD ersetzen!). Man verwendet hierfür Halbleitertechnologien, mit denen Strukturbreiten von 0,18  $\mu\text{m}$  (und weniger) verwirklicht werden können. Die Betriebsspannung liegt zwischen 1,8 und 2,5 V. Die 1-GBit-Typen haben noch zweiwertige Speicherzellen (jede Speicherzelle nimmt ein Bit auf). Demgegenüber hat man z. B. bei der Firma NEC, um 4 GBits speichern zu können, dieses Prinzip verlassen: eine Speicherzelle kann 2 Bits aufnehmen (der 4G-Schaltkreis beruht auf einer 2G-Speichermatrix). Um 2 Bits zu speichern, brauchen wir eine 4-wertige Informationsdarstellung; jede Zelle muß also in der Lage sein, 4 verschiedene Ladungswerte (einschließlich "keine Ladung") zu speichern, und die Leseverstärker müssen diese 4 Werte zuverlässig voneinander unterscheiden können. Angaben zur Organisationsform sind mit Vorsicht zu betrachten (da es sich um Muster handelt und die Entwicklung im Fluß ist, gibt es verständlicherweise noch keine Standards). Man bevorzugt aber grundsätzlich die synchrone Betriebsweise. Als Beispiel seien die veröffentlichten Angaben zu einem 4-GBit-Typ genannt: Organisationsform  $64\text{M} \cdot 64$ , 2 Banks, maximale Taktfrequenz 125 MHz (nach: NEC).

## 8.2. Hochleistungs-Speichersubsysteme (Übersicht)

Von künftigen Speichersubsystemen werden Datenraten von über 1 GBytes/s bei geringstmöglichen Latenzzeiten (und Kosten) gefordert. Die naheliegende Lösung besteht darin, entsprechend viele herkömmliche, kostengünstiger Speicherschaltkreise und extrem breite Datenwege (128, 256 und mehr Bits) vorzusehen. Hiermit lassen sich die gewünschten Speicherkapazitäten und Datenraten zwar verwirklichen (wie dies bei Mainframes und Supercomputern seit langem der Fall ist), infolge der Nachteile (Kosten, Platzbedarf, Störstrahlung) kommen derartige Lösungen aber nicht für den Massen-Markt in Frage.

Um die Nachteile zu vermeiden, bevorzugt man vergleichsweise schmale, aber mit extremen Taktfrequenzen betriebene Signalwege. Die zugrunde liegende "Systemphilosophie" kann folgendermaßen gekennzeichnet werden:

- Grundlage der Datenspeicherung sind nach wie vor herkömmliche DRAM-Strukturen (die zu speichernden Bits werden in DRAM-Matrizen gehalten, die so aufgebaut sind, wie in Kapitel 5, Abschnitt 5.2.1. beschrieben - der Vorteil: Kostensenkung durch Nutzung bewährter Technologien),
- Anordnung mehrerer DRAM-Matrizen (Memory Banks) in einem Speicherschaltkreis,
- Nutzung der Leseverstärker (Sense Amplifiers) als eine Art Cache für den Zeilen-Inhalt,
- synchrone Signalübertragung unter Nutzung beider Taktflanken,
- geringer Signalhub,
- Unterscheidung zwischen Low und High auf Grundlage einer eigens zugeführten Referenzspannung,

- Takt und Daten haben jeweils gleiche Richtung,
- Taktsynchronisation über PLL (in jedem Speicherschaltkreis),
- “schmales” Interface, über das Daten, Adressen, Kommandos usw. abschnittsweise (in Worten, Bytes oder bitseriell) nacheinander übertragen werden,
- genaue Vorgaben hinsichtlich der elektrischen Auslegung (definierter Wellenwiderstand Leitungsabschluß),
- genaue Vorgaben zur Speichersteuerung (Memory Controller),
- genaue Vorgaben (“Kochbuch-Lösungen”) zur Leiterplatten- und Systemgestaltung.

Wir erkennen eine evolutionäre Weiterentwicklung gemäß den Schritten *herkömmliche DRAMs (FPM, EDO)* → *synchrone DRAMs (SDRAMs)* → *SDRAMs mit doppelter Datenrate (DDR)* → *Multibank-DRAMs (MDRAMs)* → *moderne Systemlösungen: Rambus* → *Concurrent Rambus* → *Direct Rambus/SLDRAM*. Und dies sind die wesentlichen Entwicklungsschritte:

- Taktsteuerung statt Strobe-Steuerung (synchrone Arbeitsweise),
- Kommando-Steuerung statt Steuerung über Einzelsignale,
- paketweise Übertragung statt Übertragung einzelner Adreß- und Datenworte (man spricht von “paketorientierten” - Packed Based - DRAMs),
- Schaltkreisauswahl über Adreßvergleich statt über besondere Auswahlssignale,
- Betrieb mit extremen Taktfrequenzen (200 MHz und mehr); Nutzung beider Taktflanken.

Die extreme Taktfrequenz ist das Entscheidende - hierdurch allein ist es möglich, kostengünstige, aus nur wenigen Schaltkreisen bestehende Speichersubsysteme für hohe Datenraten und geringe Latenzzeiten auszulegen.

Ein gern herausgestellter Kennwert heißt “Datenrate je Anschluß”: MBits/s/p bzw. GBits/s/p (Bits je Sekunde und Pin).

Viele der technischen Detaillösungen sind darauf gerichtet, höchste Datenraten je Anschluß zu realisieren. Hierzu müssen die einschlägigen Schwierigkeiten (Übersprechen, Zeittoleranzen, Skew usw.) gelöst werden. Im Grundsatz zwingt dies zur Nutzung bestimmter Prinzipien (die allen paketorientierten Speicherarchitekturen gemeinsam sind), in den Einzelheiten gibt es aber beachtliche Unterschiede.

#### *Latenzzeiten*

Betrachtet man die Latenzzeiten (und sonstigen “Zeitverluste”), die sich aus den DRAM-Wirkprinzipien (vgl. Kapitel 5, Abschnitt 5.2.1.) ergeben, so sind paketorientierte DRAMs an sich nicht besser als die anderen DRAM-Architekturen. Die - für höchste Datenraten notwendige - synchrone Arbeitsweise des Interfaces verlängert sogar noch diese prinzipbedingten Zeiten, da jeweils auf ganze Taktzyklen aufzurunden ist. Bei einer extrem hohen Taktfrequenz ist aber dieser “Verschnitt” naturgemäß geringer als bei einer niedrigeren (mit anderen Worten: die Auflösung des Zeitrasters ist nicht so grob). Beispiel: 66 MHz entsprechen einem Zeitraster von 15 ns, 250 MHz einem Zeitraster von 4 ns. Demzufolge kann ein paketorientiertes DRAM-Subsystem trotz der schmalen Datenwege (8/9 oder 16/18 Bits) und der umständlich scheinenden Übertragungsabläufe (Paketübertragung) kürzere Latenzzeiten haben als ein technologisch vergleichbares Speichersubsystem auf Grundlage herkömmlicher SDRAMs.

Im folgenden wollen wir jene Speicherarchitekturen kurz beschreiben, die bereits eingeführt sind bzw. die mit dem Ziel entwickelt werden, demnächst die vorherrschende Systemlösung am Massen-Markt darzustellen.

*Hinweise:*

1. Die folgenden Erläuterungen wurden auf Grundlage vorläufigen Datenmaterials ausgearbeitet. Die Speicher-Architekturen sind teils bisher eher in Nischen-Märkten wirksam geworden (Rambus), teils befinden sie sich noch in der Phase der Markteinführung (Concurrent Rambus) oder der Entwicklung (Direct Rambus, SLDRAM). Dementsprechend sind nicht alle Spezifikationen zugänglich, und die zugänglichen sind nicht immer vollkommen zutreffend. Das heißt: im Bedarfsfall an Ort und Stelle nachsehen (Internet).
2. Um sich in die Wirkungsweise dieser Schaltkreise und Speichersubsysteme einzuarbeiten, ist es nützlich, sich zunächst an Gemeinsamkeiten zu orientieren und die Entwicklungsgeschichte im Auge zu behalten. Es baut tatsächlich eines auf dem anderen auf, die Grundprobleme (Takttoleranzen, Ground Bounce, Signalreflexionen, Busumschaltung (Bus Turnaround), Latenzzeiten usw.) sind stets die gleichen, und viele Lösungen ähneln einander. (Dies erleichtert das Lernen beträchtlich, wird aber im Datenmaterial der Anbieter kaum herausgestellt.)
3. Wir werden im folgenden versuchen, typischer Merkmale des Aufbaus und der Wirkungsweise auch mit deutschsprachigen Fachbegriffen zu beschreiben. Dies sind aber keine "offiziellen" Begriffsbildungen (der Hersteller oder Standardisierungsgremien). Uns geht es auch nicht um eine wörtliche Übersetzung, sondern um möglichst zutreffende und einleuchtende Bezeichnungen. Sollten sich Hersteller oder Standardisierungsgremien der Mühe unterziehen, ihr Datenmaterial in's Deutsche zu übersetzen (womit allerdings kaum zu rechnen ist), so haben selbstverständlich die dort gewählten Begriffe Vorrang.

## 8.3. Rambus-DRAMs (RDRAMs)

### 8.3.1. Rambus-Schaltkreise

Rambus-Speicherschaltkreise (RDRAMs) werden mit Speicherkapazitäten von 8, 16 und 64 MBits angeboten. Sie werden in speziellen Gehäusen geliefert (Abbildung 8.1).

**Abbildung 8.10** RDRAM-Gehäuse (Rambus, Inc.)

*Erklärung:*

Beide Gehäusetypen sind zur Oberflächenmontage vorgesehen. Der eigentliche Rambus-Kanal (Rambus Channel) umfaßt 32 Leitungen, denen 32 Anschlüsse des RDRAM-Gehäuses entsprechen. Anschlußabstand: 0,65 mm. Die einseitige Anordnung ermöglicht eine geradlinige Leiterzugführung (ohne Knicke usw.). Damit lassen sich die elektrischen Anforderungen selbst auf Zweiebenen-Leiterplatten gut einhalten.\*)

- a) SVP = Surface Mount Vertical Package (steht senkrecht auf der Leiterplatte),
- b) SHP = Surface Mount Horizontal Package (gelegentlich auch als SSOP bezeichnet; liegender Einbau),
- c) "Mechanical Pins" haben keine elektrische Verbindung (Not Connected) und dienen lediglich zur Befestigung des Schaltkreises auf der Leiterplatte.

\*): bei extremen Frequenzen kommt es darauf an, daß es sich wirklich um "homogene" Signalleitungen handelt, die einen konstanten Wellenwiderstand über die gesamte Leitungslänge aufweisen. Knicke, Lötstellen, Bohrungen usw. bewirken eine örtliche Veränderung des Wellenwiderstandes (man spricht von "Inhomogenitäten"). Die hierdurch bedingten Reflexionen können bei 200 und mehr MHz schon zuviel sein!

*Hinweis:*

Während sich herkömmlicherweise der Schaltkreishersteller nur um seinen Speicherschaltkreis kümmert, bietet Rambus ein Subsystem "aus einem Guß" an (Speicherschaltkreise + Leiterplattengestaltung + Controller-Strukturen; siehe die Abbildungen 8.4 und 8.5). Der Nutzer (z. B. ein Motherboard- oder Videokarten-Hersteller) muß zwar dafür bezahlen, erspart aber eigene Entwicklungsarbeit.

Die Abbildungen 8.2 und 8.3 veranschaulichen die mechanische Auslegung von Rambus-Speichersubsystemen.

**Abbildung 8.11** Rambus-Speichersubsysteme (Rambus, Inc.)

*Erklärung:*

1 - Steuerschaltkreis, 2 - Speicher-Grundausrüstung, 3 - Steckfassung (RSocket™), 4 - Steckmodul mit zusätzlichen RDRAMs, 5 - in die Steckfassungen passende RDAMs in Einzelgehäusen.

- a) waagerechter Aufbau (SHP-Gehäuse). Der Vorteil: die geringe Bauhöhe. Es ist aber je Rambus-Kanal nur ein Erweiterungs-Modul steckbar.
- b) Ausführungsbeispiel dazu,
- c) senkrechter Aufbau (SVP-Gehäuse). SVP-Gehäuse ermöglichen eine größere Packungsdichte auf der Leiterplatte, zudem können mehrere Erweiterungs-Moduln gesteckt werden.

*Hinweis:*

Es ist möglich, Leiterplatten so auszulegen, daß sie wahlweise mit SHP- oder mit SVP-Gehäusen bestückt werden können.

**Abbildung 8.12** Videokarte mit Rambus-Bildspeicher (Rambus, Inc.). Zur Erklärung der Bezugszeichen siehe Abbildung 8.2

**Abbildung 8.13** Rambus-Strukturen. a) einfachstes Blockschaltbild, b) Ausführungsbeispiel (Rambus, Inc.)

**Abbildung 8.14** RDRAMs mit Controller (Rambus, Inc.)

*Erklärung:*

Die Schnittstellen zum Rambus-Kanal (RAC Interface Unit) und zur Anwendungsschaltung (Application Unit - Prozessor, Motherboard-Steuerlogik, Video-Steuerung o. dergl.) sind fertig ausgearbeitet (die gesamte Schaltung wird als ASIC-Standardzelle angeboten). Der Rambus-Kanal hat eine Zugriffsbreite von 9 (bzw. 8) Bits, die Schnittstelle zur Anwendungsschaltung hingegen hat 72 (bzw. 64) Bits.

*Hinweis:*

Die Abbildung zeigt eine Anordnung mit "Concurrent RDRAMs". Es handelt sich um weiterentwickelte Typen (die in Abschnitt 8.4. vorgestellt werden). Sie haben aber dieselbe Schnittstelle (Rambus-Kanal) wie die "gewöhnlichen" RDRAMs.

### 8.3.2. Die Signale des Rambus-Kanals

Abbildung 8.6 und Tabelle 8.1 geben einen Überblick über die Rambus-Signale.

**Abbildung 8.15** Anschlußbelegung eines RDRAM (NEC)

Signal(e)	Signalpegel, Übertragungsrichtung	Anwendung
BusData8...0	RSL, bidirektional	Übertragung von Daten, Adressen und Kommandos
BusCtrl	RSL, bidirektional	Steuerung der paketweisen Übertragung (10. Bitposition in den Anforderungspaketen), Übertragung serieller Steuerpakete, Bestätigen von Anforderungen
BusEnable	RSL, Eingang	zeigt Betriebsbereitschaft des Rambus-Kanals an. Längere Deaktivierung bewirkt Rücksetzen. Übertragung serieller Adreß- und Moduspakete
RxCclk	RSL, Eingang	Takt für Busbelegungen, die zum RDRAM geschickt werden
TxCclk	RSL, Eingang	Takt für Busbelegungen, die der RDRAM liefert
S <sub>In</sub>	TTL, Eingang	Daisy-Chain-Eingang zum Steuern der Initialisierung. Aktiv High
S <sub>Out</sub>	TTL, Ausgang	Daisy-Chain-Ausgang. Aktiv High
Signal(e)	Signalpegel, Übertragungsrichtung	Anwendung
V <sub>REF</sub>	Eingang	RSL-Referenzspannung
V <sub>DD</sub> , V <sub>DDA</sub> *)	Speisespannung (3,3 V)	
GND, GNDA*)	Masse	
NC	nicht beschaltet	
IC	intern verbunden	reserviert; außen nichts anschließen

\*) : V<sub>DDA</sub>, GNDA sind "analoge" Speisespannungs- und Masseanschlüsse (zur Versorgung der Takt-PLL)

**Tabelle 8.12** Die Anschlüsse eines RDRAMs (nach: NEC)

### 8.3.3. Signalpegel (Rambus Signaling Logic RSL)

RSL ist eine elektrische Schnittstellenspezifikation mit folgenden Merkmalen (Abbildung 8.7):

- Open-Drain-Treiberstufen,
- Pull-up-Widerstände, die als Abschlußwiderstände dimensioniert sind,
- Stromsteuerung der Treiberstufen (die Treiberströme stellen sich so ein, daß die spezifizierten Low- und High-Pegel gehalten werden),
- die logischen Werte 0 und 1 werden über den Stromfluß definiert: 0 = kein Stromfluß, 1 = Stromfluß. Demzufolge haben wir es aus Sicht der Signalpegel mit einer *negativen Logik* zu tun: in der Open-Drain-Schaltung fließt dann ein Strom über die Signalleitung, wenn der Transistor durchgesteuert ist, also bei Low-Pegel. Folglich gilt: Low (= Stromfluß) = 1, High (= kein Stromfluß, sondern lediglich Halten des Pegels über den Pull-up-Widerstand) = 0.

**Abbildung 8.16** Rambus Signaling Logic (RSL). Übersicht

*Erklärung:*

- a) zum Prinzip des Rambus-Kanals: Signalleitung mit Pull-up-Widerstand (= Abschlußwiderstand) und Referenzspannungszuführung ( $V_{REF}$ ),
- b) Bildung der Referenzspannung mittels Spannungsteiler,
- c) die Rambus-Signalpegel.

*Leitungsführung* (Abbildung 8.8)

Die Rambus-Signale sind so geführt, daß der Takt immer die gleiche Richtung hat wie die jeweils zu übertragenden Datensignale (Lesen: vom RDRAM zur Speichersteuerung, Schreiben: von der Speichersteuerung zum RDRAM).

*Hinweise:*

1. Die Open-Drain-Schaltung entspricht dem Wirkprinzip nach der Open-Collector-Schaltung.
2. Für die TTL-Signale (Tabelle 8.1) gilt die herkömmliche positive Logik.
3. Auf den ersten Blick (Abbildungen 8.7, 8.8) sieht RSL wie ein einfaches Open-Drain-Interface aus. Tatsächlich haben wir es hier mit einer recht trickreichen Ausnutzung des Verhaltens von Signalleitungen zu tun.

**Abbildung 8.17** Leitungsführung der Taktsignale (Rambus, Inc.)

*Erklärung:*

- 1) Speichersteuerung (Rambus Memory Controller RMC). RAC Interface Unit = Anschlußeinheit für Rambus-Kanal.
- 2) Taktgenerator,
- 3) Lesetak (TxClk). Wird an alle RDRAMs geführt und gelangt schließlich zur Speichersteuerung.
- 4) Schreibtakt (RxClk). Gelangt von der Speichersteuerung nacheinander an alle RDRAMs.
- 5) Abschlußwiderstände,
- 6) RDRAM-Speicherschaltkreise.

### 8.3.4. Struktur eines RDRAM

Abbildung 8.9 gibt einen Überblick über den Aufbau "herkömmlicher" RDRAMs.

**Abbildung 8.18** Blockschaltbilder herkömmlicher RDRAMs (NEC; Rambus, Inc.)

*Erklärung:*

- a) 16/18-MBit-RDRAM. Der Schaltkreis enthält 2 Speichermatrizen (Bank 0, Bank 1). Jede Speichermatrix hat 16k oder 18k Spalten ( $2k \cdot 8$  oder  $2k \cdot 9$ ) und 512 Zeilen. Die den Spaltenleitungen nachgeschalteten Leseverstärker-Anordnungen wirken als eine Art Cache für den gesamten Zeileninhalt, in dem jeweils eine Seite (Page) zwischengespeichert wird (typischerweise je Bank  $k$  Bytes (zu 8 oder 9 Bits)).
- b) 64/72-MBit-RDRAM. Der Schaltkreis enthält 4 Banks mit jeweils 2 MBytes (Verdoppelung der Bank- und der Zeilenzahl gegenüber den  $\cdot$  16/18-MBit-Typen). Des weiteren ist dargestellt, daß zwischen Speichersteuerung (Memory Controller) und RDRAM die Daten zwecks Übertragung über den nur bytebreiten Rambus-Kanal serialisiert bzw. deserialisiert werden.

### 8.3.5. Steuerregister

Jeder RDRAM enthält einen Registersatz. Jedes Register umfaßt 36 Bits (die in 4 Abschnitten zu 9 Bits übertragen werden). Zur Registerauswahl wird eine 8-Bit-Adresse verwendet, so daß man theoretisch bis zu 256 Register vorsehen könnte. Tabelle 8.2 gibt einen Überblick über die Registerbelegung.

Registeradresse <sup>*)</sup>	Bezeichnung	Inhalt
0	Device Type	Konfigurations- und Speicherkapazitätsangaben
1	Device Ident	Basisadresse des Schaltkreises (am Rambus-Kanal)
2	Delay	CAS-Verzögerungszeiten (Register enthält mehrere Felder)
3	Mode	bestimmt den Treiberstrom der RSL-Ausgänge
5	RefRow	Bank- und Zeilenadresse für Refresh
6	RasInterval	RAS-Verzögerungszeiten
7	MinInterval	Zeitangaben der Refresh-Steuerung
8	Address Select	steuert die Adreßumsetzung im RDRAM
9	DeviceManufacturer	Herstellerangaben
128	Row	die jeweils aktuelle Zeile in jeder Bank

\*) : enthalten in den Adreßbits 9...2

**Tabelle 8.13** Der Registersatz eines RDRAM (Übersicht; nach: NEC)

### 8.3.6. Prinzip der Datenübertragung

Die Signale BusEnable, BusCtrl und BusData8...0 werden mit jeder Taktflanke ausgewertet. Gemäß der jeweiligen Taktflanke (High-Low oder Low-High) unterscheidet man zwischen "geraden" (Even) und "ungeraden" (Odd) Busbelegungen (Abbildung 8.10).

**Abbildung 8.19** Taktbezogene Auswertung der Busbelegung

### 8.3.7. Pakete

Daten, Adressen und Kommandos werden in Form von "Paketen" (Packets) übertragen (Tabellen 8.3 bis 8.5).

Paket-Typ	Übertragungsrichtung	Kurzbeschreibung
Anforderungspaket (Request Packet)	wird vom RDRAM empfangen	besteht aus 6 10-Bit-Abschnitten <sup>*)</sup> , die in 3 Taktzyklen übertragen werden
Bestätigungspaket (Acknowledge Packet)	wird vom RDRAM gesendet	es wird nur die Leitung BusCtrl ausgewertet, und zwar in einem Taktzyklus ( $\triangleq$ 2 Bits)
Lesedatenpaket (Read Data Packet)	wird vom RDRAM gesendet	<ul style="list-style-type: none"> <li>■ bei Registerzugriffen: stets 4 Abschnitte zu 9 Bits = 1 "Quadbyte"</li> <li>■ bei Datenzugriffen: stets Vielfache von 8 Bytes (8 oder 9 Bits) = "Octbytes". Mindestlänge = 1 Octbyte (<math>\triangleq</math> 4 Taktzyklen)</li> </ul>
Schreibdatenpaket (Write Data Packet)	wird vom RDRAM empfangen	
Seriellles Adreßpaket (Serial Address Packet)	wird vom RDRAM empfangen	Länge: 8 Bits (in 4 Taktzyklen über Leitung BusEnable übertragen). Damit wird ein Octbyte (64/72 Bits) im "Leseverstärker-Cache" der ausgewählten Bank adressiert
Seriellles Steuerpaket (Serial Control Packet)	wird vom RDRAM empfangen	Länge: 8 Bits (in 4 Taktzyklen über Leitung BusCtrl übertragen). Hierüber kann u. a. der aktuelle Zugriff beendet werden
Seriellles Moduspaket (Serial Mode Packet)	wird vom RDRAM empfangen	Länge: 2 Bits (in einem Taktzyklus über Leitung BusEnable übertragen). Steuerung (Zählen, Rücksetzen) der Adreßzähler im RDRAM

\*) BusCtrl + BusData8...0

**Tabelle 8.14** Pakete (Übersicht)

Takt <sup>*)</sup>	Bus Ctrl	BusData									
		8	7	6	5	4	3	2	1	0	
1. g	Start	Op0	Adresse 9...2								
1. u	Op1	Op3	Adresse 17...10								
2. g	OpX1	Adresse 26...18									
2. u	OpX2	Adresse 35...27									
3. g	OpX0	-	Anzahl (Count) 6, 4, 2				-				
3. u	-	Anzahl (Count) 7, 5, 3			Anzahl 1, 0			Adresse 1, 0			

\*): g = gerade (Even; High-Low-Taktflanke), u = ungerade (Odd; Low-High-Taktflanke). “-“ = reserviert (werden mit Nullen aufgefüllt)

**Tabelle 8.15** Anforderungspaket (Request Packet)

*Erklärung:*

Das Anforderungspaket enthält alles, was man zum Starten eines Speicherzugriffs braucht: eine Funktionsbestimmung (Operationscode Op3...0 + Erweiterung OpX2...0), eine Anfangsadresse und eine Anzahlangabe. Der Inhalt der Op- und OpX-Bits bestimmt das jeweils auszuführende Kommando (Tabelle 8.5).

Op3...0	OpX2...0	Kommando-Kurzbeschreibung	Mnemonic
0H	0H	Daten sequentiell aus Speicher lesen.	Rseq
	1H	Daten wahlfrei aus Speicher lesen	Rnsq
4H	0H	Daten sequentiell in Speicher schreiben. Keine Bitmaskierung	WseqNpb
	1H	Schreibdatenpaket sequentiell in Speicher schreiben. Bitmaske in MD-Register	WseqDpb
	2H	Daten sequentiell in Speicher schreiben. Daten und Bitmaske kommen im selben Schreibdatenpaket	WseqBpb
	3H	Inhalt des MD-Registers sequentiell in Speicher schreiben. Schreibdatenpaket ist Bitmaske	WseqMpb
6H	0H	Register lesen (sequentiell)	Rreg
7H	0H	Register schreiben (sequentiell)	Wreg
8H	0H	Daten wahlfrei in Speicher schreiben. Keine Bitmaskierung	WnsqNpb
	1H	Schreibdatenpaket wahlfrei in Speicher schreiben. Bitmaske in MD-Register	WnsqDpb
	2H	Daten wahlfrei in Speicher schreiben. Daten und Bitmaske kommen im selben Schreibdatenpaket	WnsqBpb
	3H	Inhalt des MD-Registers wahlfrei in Speicher schreiben. Schreibdatenpaket ist Bitmaske	WnsqMpb
CH	0H	Daten wahlfrei in Speicher schreiben. Bytemaske im Schreibdatenpaket. Keine Bitmaskierung	WbnsNpb
	1H	Schreibdatenpaket wahlfrei in Speicher schreiben. Bytemaske im Schreibdatenpaket. Bitmaske in MD-Register	WbnsDpb
	3H	Inhalt des MD-Registers wahlfrei in Speicher schreiben. Schreibdatenpaket enthält Byte- und Bitmaske	WbnsMpb
FH	0H	in Register aller RDRAMs gleichzeitig schreiben (globales Schreiben, Broadcast Write)	WregB

**Table 8.16** RDRAM-Kommandoübersicht

## 8.3.8. Einzelheiten der Kommandoausführung

### 8.3.8.1. Registerzugriffe

Es werden stets Blöcke von 4 9-Bit-Bytes (Quadbytes) transportiert (ein solcher Block wird in 2 Taktzyklen übertragen).

### 8.3.8.2. Speicherzugriffe

Es werden stets Blöcke von 8 Bytes (Octbytes) zu 8 oder 9 Bits transportiert (ein solcher Block wird in 4 Taktzyklen übertragen). Die Übertragung eines jeden Paketes beginnt mit einer "geraden" (Even) Taktflanke (High → Low).

### 8.3.8.3. Adresse

Es werden 36-Bit-Adressen übertragen. Höherwertige Adreßbits werden im RDRAM zu Vergleichszwecken verwendet (bei Gleichheit mit entsprechenden Teilen des Device-Ident-Registers "erkennt sich der Schaltkreis als angesprochen" (Slave-Auswahl)). Die Adreßbits 35...2 wählen ein Quadbyte aus, die Adreßbits 35...3 ein Octbyte. Ungenutzte Adreßbits werden typischerweise mit Nullen belegt. Register werden über die Adreßbits A9...A2 adressiert. Zur Aufteilung der Speicheradressen siehe Tabelle 8.6.

Nutzung	Speichertyp		
	16/18 MBits; 2 Banks mit 1 kBytes Seitengröße	16/18 MBits; 2 Banks mit 2 kBytes Seitengröße	64/72 MBits; 4 Banks mit 2 kBytes Seitengröße
Spaltenadresse (Octbyte in Seite)	A9...A3 (8 Bits $\triangle$ 256 Octbytes/Seite)	A10...A3 (9 Bits $\triangle$ 512 Octbytes/Seite)	A10...A3 (9 Bits $\triangle$ 512 Octbytes/Seite)
Zeilenadresse	A19...A10 (10 Bits $\triangle$ 1024 Zeilen)	A19...A11 (9 Bits $\triangle$ 512 Zeilen)	A20...A11 (10 Bits $\triangle$ 1024 Zeilen)
Bankauswahl	A20	A20	A22, A21
Schaltkreisauswahl (Adreßvergleich)*	A35...A21 (15 Bits)	A35...A21 (15 Bits)	A35...A23 (13 Bits)

\*) auch bei Registeradressierung

**Tabelle 8.17** Adreßaufteilung (nach: Rambus, Inc.)

### 8.3.8.4. Längenzählung

Im Anforderungspaket wird eine 8-Bit-Anzahlangabe übertragen (Tabelle 8.7).

Zugriff	Bereich der Längenzählung	Anzahlangabe
Speicherzugriffe	wenigstens 1 Octbyte... maximal 32 Octbytes (= 256 Bytes)	Count 7...3
Registerzugriffe	stets 1 Quadbyte = 1 Register (unabhängig von Anzahlangabe)	beliebig

**Tabelle 8.18** Längenzählung

### 8.3.8.5. Byteversatz beim sequentiellen Schreiben (Contiguous Byte Masking)

Beim sequentiellen Schreiben ist es möglich, innerhalb eines Octbytes an einer beliebigen Byteadresse mit dem Schreiben zu beginnen und eine beliebige Zahl zusammenhängender Bytes zu schreiben (die Speicherinhalte am Anfang und am Ende der Octbyte-Blöcke werden nicht überschrieben). Der Inhalt der Adreßbits A2...0 wird hierbei als Byte-Anfangsadresse im ersten Octbyte und der Inhalt der gesamten Längenangabe (Count 7...0) als Byte-Anzahl interpretiert.

### 8.3.8.6. Wahlfreie Zugriffe

Es kann nur auf die jeweilige Seite (von beispielsweise 2 kBytes) wahlfrei zugegriffen werden. Die Octbyte-Adressen werden hierzu als serielle Adreßpakete über die Leitung BusEnable übertragen.

### 8.3.8.7. Das Bestätigungspaket (Acknowledge Packet)

Bestätigungspakete werden von den RDAMs in jeweils einem Taktzyklus über die Leitung BusCtrl zur Speichersteuerung übertragen. Sie sind somit nur 2 Bits lang (Tabelle 8.8).

Bestätigung		Bezeichnung	Reaktion auf	Kurzbeschreibung
Ack1 <sup>1)</sup>	Ack0 <sup>2)</sup>			
0	0	Nonexistent	alle Kommandos	<ul style="list-style-type: none"> <li>■ Kommando WregB: Kommando akzeptiert,</li> <li>■ sonst: RDRAM antwortet nicht</li> </ul>
0	1	Okay	alle Kommandos außer WregB	Kommando akzeptiert
1	0	Nack	alle Kommandos	Kommando kann derzeit nicht akzeptiert werden (Zurückweisung)
1	1	Ack3	alle Kommandos außer WregB	reserviert; sollte nicht gesendet werden

1): als 2. (ungerades) Bit übertragen, 2): als erstes (gerades) Bit übertragen

**Tabelle 8.19** Codierung der Bestätigungssignale (nach: NEC)

### 8.3.9. Elementare Zugriffsabläufe (Transaktionen)

Die Abbildungen 8.11 bis 8.14 veranschaulichen wichtige Zugriffsabläufe (zu näheren Einzelheiten muß auf die Originalliteratur der Fa. Rambus und der Schaltkreishersteller verwiesen werden (Internet)).

Zum Aufbau der Abbildungen 8.11 bis 8.14:

- es sind die aufeinanderfolgenden Belegungen der Informationsleitungen des Rambus-Kanals dargestellt (BusEnable, BusCtrl, BusData8...0),
- weiße Flächen: Belegung mit einem gültigen Paket für den betreffenden Zugriffsablauf,
- graue Flächen: im betreffenden Zugriffsablauf nicht belegt (entweder geschieht auf dem Rambus-Kanal gar nichts (Warten), oder es können anderweitige Zugriffsfolgen eingeschoben werden). Solange "nichts" geschieht, werden die betreffenden Signale auf High (= logisch Null) gehalten.

**Abbildung 8.20** Lesezugriff (Read Transaction; NEC)

*Erklärung:*

- 1) Anforderungspaket mit Anfangsadresse und Längenangabe (von der Speichersteuerung),
- 2) adressierter SDRAM bestätigt Kommando (Okay Acknowledge),
- 3) Lesedatenpaket vom RDRAM. Dies folgt mit einer festen Verzögerung  $t_{\text{ReadDelay}}$  (dieser Wert - gemessen in Taktzyklen - muß zuvor (beim Initialisieren) in das ReadDelay-Feld des Delay-Registers geschrieben worden sein (Beispiel-Wert: 7 Takte).

Ein "Okay Acknowledge" besagt, daß nach Ablauf des  $t_{\text{ReadDelay}}$  tatsächlich ein Lesedatenpaket zurückkommt. Bleibt diese Bestätigung aus (kommt also innerhalb des Zeitrahmens von  $t_{\text{AckDelay}}$  keine solche Bestätigung), so folgt auch kein Lesedatenpaket nach.

### **Abbildung 8.21** Schreibzugriff (Write Transaction; NEC)

*Erklärung:*

- 1) Anforderungspaket mit Anfangsadresse und Längenangabe (von der Speichersteuerung),
- 2) Speichersteuerung schickt Schreibdatenpaket nach, und zwar mit einer festen Verzögerung  $t_{\text{WriteDelay}}$  (dieser Wert - gemessen in Taktzyklen - muß zuvor (beim Initialisieren) in das WriteDelay-Feld des Delay-Registers geschrieben worden sein (Beispiel-Wert: 1 Takt).
- 3) RDRAM bestätigt Empfang des Datenpakets mit einer "Okay-Acknowledge"-Meldung. Gibt es keine solche Bestätigung, war das Schreiben erfolglos.
- 4) im folgenden Takt kann sofort auf einen anderen RDRAM zugegriffen werden. Soll der nächste Zugriff den selben RDRAM betreffen, muß eine - von der Art des eben abgeschlossenen Zugriffs abhängende - Anzahl von Takten abgewartet werden.

### **Abbildung 8.22** Lesezugriff (wahlfrei) mit seriellem Adreßpaket (NEC)

*Erklärung:*

- 1) Anforderungspaket mit Anfangsadresse und Längenangabe (von der Speichersteuerung). Dieses Anforderungspaket adressiert das erste zu lesende Octbyte.
  - 2) adressierter SDRAM bestätigt Kommando (Okay Acknowledge),
  - 3) Lesedatenpaket mit dem ersten Octbyte vom RDRAM. Der Beginn eines solchen Zugriffs entspricht dem des sequentiellen Lesens.
  - 4) serielle Übertragung der ersten 8-Bit-Octbyte-Adresse,
  - 5) Lesedatenpaket mit dem zweiten Octbyte (dieses wurde mit der ersten seriell übertragenen Adresse (4) ausgewählt),
  - 6) serielle Übertragung weiterer 8-Bit-Octbyte-Adressen,
  - 7) auf jede seriell übertragenen Octbyte-Adresse antwortet der RDRAM mit dem Senden eines Lesedatenpakets, das das adressierte Octbyte enthält. Zwischen der seriellen Übertragung und dem Senden des jeweils adressierten Octbytes liegt ein vorgegebenes festes Zeitintervall  $t_{\text{SerialReadOffset}}$  (z. B. von 12 Takten). *Hinweis:* Auch zwischen der ersten seriellen Octbyte-Adresse (4) und dem zugehörigen Lesedatenpaket (5) liegen  $t_{\text{SerialReadOffset}}$  Takte. Die Speichersteuerung darf die Adresse vor dem Eintreffen der Bestätigung (2) senden, muß aber  $t_{\text{SerialReadOffset}}$  beachten. Trifft keine "Okay"-Bestätigung ein, war die gesamte Zugriffsfolge erfolglos (sie ist dann abzubrechen und nach einer gewissen Zeit zu wiederholen\*).
- \*) ein Zugriff ist an sich nur dann erfolglos, wenn im RDRAM gerade ein Refresh- oder Precharge-Vorgang stattfindet. Bis zur Wiederholung des Zugriffs (Wiederanlauf (Retry)) ist jeweils ein bestimmtes Zeitintervall abzuwarten (Abschnitt. 8.3.12.).

### **Abbildung 8.23** Schreibzugriff (wahlfrei) mit seriellem Adreßpaket (NEC)

*Erklärung:*

- 1) Anforderungspaket mit Anfangsadresse und Längenangabe (von der Speichersteuerung). Dieses Anforderungspaket adressiert das erste zu schreibende Octbyte.
- 2) Speichersteuerung schickt das erste Schreibdatenpaket (mit dem ersten Octbyte) nach,
- 3) RDRAM bestätigt Empfang des Datenpakets mit einer "Okay-Acknowledge"-Meldung. Der Beginn eines solchen Zugriffs entspricht dem des sequentiellen Schreibens.
- 4) serielle Übertragung der ersten 8-Bit-Octbyte-Adresse,
- 5) Speichersteuerung schickt das zweite Schreibdatenpaket (dessen Zieladresse wurde mit dem ersten seriellen Adreßpakete (4) übertragen),

- 6) serielle Übertragung weiterer 8-Bit-Octbyte-Adressen,
- 7) jeder seriell übertragenen Octbyte-Adresse schickt die Speichersteuerung ein Schreibdatenpaket nach, das das jeweils zu schreibende Octbyte enthält. Zwischen der seriellen Übertragung und dem Senden des jeweiligen Octbytes liegt ein vorgegebenes festes Zeitintervall (in Takten)  $t_{\text{SerialWriteOffset}}$  (z. B. von 8 Takten). *Hinweis:* Auch zwischen der ersten seriellen Octbyte-Adresse (4) und dem zugehörigen Schreibdatenpaket (5) liegen  $t_{\text{SerialWriteOffset}}$  Takte. Die Speichersteuerung darf die Adresse (4) sogar schon gleichzeitig mit dem Anforderungspaket (1) senden, muß aber  $t_{\text{SerialWriteOffset}}$  beachten. Trifft keine "Okay"-Bestätigung ein, war die gesamte Zugriffsfolge erfolglos (das Schreiben ging "ins Leere" und ist nach einer gewissen Zeit zu wiederholen - siehe hierzu den Punkt \*) in der Erklärung zu Abbildung 8.13).
- 8) Vgl. Punkt 4 der Erklärung zu Abbildung 8.12.

### 8.3.10. Rücksetzen

Alle RDRAMs eines Rambus-Kanals gelangen in den inaktiven (Rücksetz-) Zustand, wenn BusEnable während einer gewissen Anzahl von Taktzyklen (Richtwert: 40) inaktiv (High) bleibt. Ein inaktiv gewordener RDRAM schaltet sein Daisy-Chain-Ausgangssignal SO inaktiv.

### 8.3.11. Initialisierung

Im Rücksetzzustand befindliche RDRAMs müssen vor der Nutzung durch Laden ihrer Steuerregister initialisiert werden. Da im Rücksetzzustand die Slave-Adressen noch nicht geladen worden sind, erfolgt die Auswahl über die Daisy-Chain-Signale SI, SO: die Kommandos (mit Ausnahme des globalen Schreibers WregB, das alle RDRAMs gleichzeitig betrifft) wirken nur auf den RDRAM, der sich noch im Rücksetzzustand befindet, dessen Daisy-Chain-Eingang SO aber aktiv ist. (Der SI-Eingang des ersten RDRAMs ist fest aktiviert.) Ist dieser RDRAM initialisiert worden, so aktiviert er seinen SO-Ausgang. Dies bewirkt, daß der nächste RDRAM zur Initialisierung ausgewählt wird.

### 8.3.12. Refresh und Precharge

Beides erfolgt im RDRAM sozusagen automatisch. Die Speichersteuerung muß sich zunächst nicht darum kümmern. Ist bei einem Zugriff in einer Bank eine andere Zeile zu aktivieren (Row Miss) oder trifft ein Zugriff auf einen Refresh-Ablauf, so antwortet der RDRAM mit einer Zurückweisung im Bestätigungspaket (Negative Acknowledge, Nack).

#### *Wiederanlaufzeiten*

Nach einer Zurückweisung kann der Zugriff wiederholt werden. Welche Zeiten bis zum erneuten Start des Zugriffs (Retry) abzuwarten sind (damit dann der Zugriff garantiert erfolgreich ablaufen wird), ist vorgegeben (es ist also nicht erforderlich, daß die Speichersteuerung versucht, den Zugriff zyklisch zu wiederholen). Beispiele für solche Wiederanlauf-Zeitintervalle:

- von Row Miss (Nack-Bestätigung) bis zum nächsten Anforderungspaket (und zwar dann, wenn in die "alte" Zeile nichts mehr geschrieben werden muß):  $t_{\text{RetrySensedClean}} = 22$  Takte,
- von Row Miss (Nack-Bestätigung) bis zum nächsten Anforderungspaket (und zwar dann, wenn für die "alte" Zeile noch Schreibdaten im RDRAM vorliegen):  $t_{\text{RetrySensedDirty}} = 22$  Takte,
- von Refresh (Nack-Bestätigung) bis zum nächsten Anforderungspaket:  $t_{\text{RetryRefresh}} = 221$  oder  $213$  Takte (je nachdem, ob in die "alte" Zeile noch geschrieben werden muß (Dirty) oder nicht (Clean)).

## 8.4. Concurrent Rambus DRAMs

### 8.4.1. Aufbau

Concurrent Rambus RDRAMs (wir wollen sie im folgenden CRDRAMs nennen) sind RDRAMs mit einem neuen Signalprotokoll. Die Neuerungen haben den Zweck, bessere Gelegenheiten zu bieten, mehrere Zugriffe nahezu gleichzeitig, also zeitlich ineinander verschachtelt (interleaved, überlappt) auszuführen (Concurrency).

*Gemeinsamkeiten* mit den herkömmlichen RDRAMs:

- der Rambus-Kanal an sich,
- Schaltkreisgehäuse und Anschlußbelegungen (Abweichung: bei 64/72-MBit-Typen ist Pin 6 mit der Referenzspannung ( $V_{REF}$ ) belegt),
- Versorgungsspannung: 3,3 V,
- die grundsätzliche Struktur des RDRAM (Abbildung 8.15),
- die Ausnutzung beider Taktflanken zur Signalübertragung,
- die Übertragung von Paketen (Packets), im besonderen als Vielfache von 8 Bytes (Octbytes).

*Neu* ist das Signalprotokoll, das man als “mehr SDRAM-mäßig” bezeichnen könnte. Man hat hierfür 2 Rambus-Signale um-interpretiert:

- BusEnable = ADDRESS (kennzeichnet Adressenübertragung),
- BusCtrl = COMMAND (kennzeichnet Kommandoübertragung).

Eine weitere Änderung: Ersatz der TTL-Signalpegel durch CMOS-Signalpegel (unter Beibehaltung der positiven Logik. Dies betrifft die Signale  $S_{in}$  und  $S_{out}$ ).

*Entfallen sind:*

- der Bestätigungspakete (Acknowledge Packets),
- die “vollautomatische” Erledigung der Precharge- und Refreshabläufe (als zwangsläufige Folge des Verzichts auf die Bestätigungspakete),
- Vorkehrungen zur Längenzählung. (Das Weiterzählen der Spaltenadresse in Burst-Zugriffen obliegt hier der Speichersteuerung. Solche Zugriffe lassen sich aber lückenlos aneinanderreihen - somit sind Bursts beliebiger Länge möglich).

Herkömmliche RDRAMs und CRDRAMs können gemeinsam an einem Rambus-Kanal betrieben werden (die Speichersteuerung muß dies unterstützen, und die Steuerregister der CRDRAMs sind dementsprechend zu laden).

**Abbildung 8.24** Blockschaltbild eines Concurrent RDRAM (Rambus, Inc)

### 8.4.2. Pakete

Adressen, Kommandos und Daten werden in Form von Paketen übertragen. Aus Tabelle 8.9 und Abbildung 8.16 ist ersichtlich, welche Pakete es gibt und wie die Signalübertragung grundsätzlich abläuft.

Paket-Typ	Kurzbeschreibung
Anforderungspaket (REQ Packet)	besteht aus einem Octbyte (Abbildung 8.17)
Lesedatenpaket (DIN Packet)	bestehen aus wenigstens einem Octbyte
Schreibdatenpaket (DOUT Packet)	
Spaltenadreßpaket (COL Packet)	Länge: 8 Bits (in 4 Taktzyklen über Leitung BusEnable übertragen). Damit wird ein Octbyte (64/72 Bits) im "Leseverstärker-Cache" der ausgewählten Bank adressiert
Takterlaubnispaket (CKE Packet)	Jedes dieser "Pakete" entspricht bestimmten Bitposition in einem Kommandobyte, das seriell in 4 Taktzyklen über die Leitung BusCtrl (= COMMAND) übertragen wird.
Lese-Startpaket (RSTRB Packet)	
Lese-Endepaket (RTERM Packet)	
Schreib-Startpaket (WSTRB Packet)	
Schreib-Endepaket (WTERM Packet)	
Rücksetzpaket (RESET Packet)	
Einschaltpaket (PWRUP Packet)	

**Tabelle 8.20** Pakete (Übersicht)

*Erklärung* zu Abbildung 8.16:

Ein Paket wird grundsätzlich in 4 Taktzyklen (d. h. mit 8 Taktflanken) übertragen (Octbyte-Prinzip).

- a) Datenübertragung. Es werden stets Octbytes übertragen (8 Bytes zu 8 oder 9 Bits in 4 Taktzyklen = 8 Taktflanken). Auch Anforderungspakete und Registerinhalte haben dieses Format. In der Bezeichnungsweise unterscheidet man 64 "eigentliche" Datenbits D63...D0 und 8 "ECC-Bits" (E7...E0). Die Bits E7...E0 werden in Anforderungspaketen sowie zur Bytemaskierung verwendet. Gespeichert werden sie nur in Speichern der 72-Organisation. In Register-Formaten werden sie (im Gegensatz zum herkömmlichen RDRAM) nicht verwendet.
- b) Spaltenadreßpaket (COL Packet). Die 8 Bits der Spaltenadresse (A10...A3) werden im Rahmen einer Octbyte-Übertragung seriell über die Leitung BusEnable (= ADDRESS) übertragen. Dies kann parallel zu einer Daten- und Kommandoübertragung ablaufen.
- c) Kommandopakete (CKE...PWRUP; vgl. Tabelle 8.9). Kommandos werden im Rahmen einer Octbyte-Übertragung seriell über die Leitung BusCtrl (= COMMAND) übertragen. Dies kann parallel zu einer Daten- und Spaltenadreßübertragung ablaufen. Die Kommandos CKE, RSTRB, RTERM,WSTRB, WTERM belegen jeweils eine Bitposition. Somit können unterschiedliche Kommandos in einer einzigen Octbyte-Übertragung signalisiert werden. RESET und PWRUP sind durch die Belegung aller 8 Bitpositionen mit Einsen gekennzeichnet (welches der beiden Kommandos ausgeführt wird, hängt vom Zustand des CRDRAMs ab).

**Abbildung 8.25** Grundabläufe der Signalübertragung (Rambus, Inc.)

*Anforderungspakete*

Jeder Zugriffsablauf beginnt mit einem Anforderungspaket (REQ Packet; Abbildung 8.17).

**Abbildung 8.26** Das Anforderungspaket (REQ Packet; Rambus, Inc.)

*Erklärung:*

Die Abbildung veranschaulicht in ihrem oberen Teil die "gedrängte" Symbolik, in der die einzelnen Zugriffsfolgen dokumentiert sind (vgl. weiter unten Abbildung 8.18 und folgende). Darunter ist dargestellt, welche Felder bzw. Einzelbits nacheinander übertragen werden (Tabelle 8.10). *Hinweise:* (1) vgl. Abbildung 8.17 mit Tabelle 8.4, (2) grau hinterlegte Bitpositionen gelten als reserviert. Sie sind mit Nullen aufzufüllen.

Feld/Bit im Anforderungspaket	Bedeutung
START	stets auf 1 gesetzt. Kennzeichnet Beginn einer Zugriffs-Anforderung
OP5...0	Operationscode (Kommandocode). Siehe dazu Tabelle 8.11
REGSEL	Unterscheidung zwischen Speicher- und Registerzugriffen
A35...A3	Adresse (Schaltkreis, Bank, Zeile, Spalte, Octbyte). Vgl. Tabelle 8.6
ACTV	Aktivieren. Aktiviert vor dem eigentlichen Zugriff die ausgewählte Zeile in der ausgewählten Bank ( $\triangle$ RAS-Erregung bzw. dem Aktivieren beim SDRAM). War zuvor in derselben Bank eine Zeile aktiv gewesen, so wird diese zunächst deaktiviert (Precharge-Ablauf)
AUTO	Auto Precharge. Bewirkt Precharge-Ablauf (Deaktivieren der jeweiligen Zeile) in der ausgewählten Bank nach Ende des Zugriffs (wie beim SDRAM)
PEND2...0	bei Zugriffen ohne Interleaving sind die Bits mit Nullen belegt. Werden Zugriffe überlappt (interleaved) ausgelöst, so ist in diesem Feld jeweils anzugeben, wieviele (andere Zugriffe betreffende) STRB- und TERM-Pakete zu übergehen sind
M7...M0	Byte-Maskierung im ersten Octbyte (bei Schreibzugriffen). Bei Registerzugriffen mit Nullen füllen

**Tabelle 8.21** Zur Belegung des Anforderungspaketes

### 8.4.3. Kommandos

Tabelle 8.11 gibt einen Überblick über die Kommandos.

OP5...0 <sup>1)</sup>	ACTV, AUTO	Mnemonic	Kurzbeschreibung
00H oder 04H <sup>2)</sup>	0, 0	READ	Daten aus Speicher lesen
b1, b0, D, B, 0, 1	0, 0	WRITE	Daten in Speicher schreiben
06H	0, 0	RREG	Register lesen
0, 0, D, 1, 1, 1	0, 0	WREG	Register schreiben
00H oder 04H <sup>2)</sup>	0, 1	READA	Speicher-Lesen mit anschließendem Auto Precharge
b1, b0, D, B, 0, 1	0, 1	WRITEA	Speicher-Schreiben mit anschließendem Auto Precharge
00H oder 04H <sup>2)</sup>	1, 0	ACTV/READ, PRE/ACTV/READ <sup>3)</sup>	Zeile in Bank aktivieren, dann Speicher lesen. Wenn erforderlich, zuvor Zeile deaktivieren (Precharge)
b1, b0, D, B, 0, 1	1, 1	ACTV/WRITE, PRE/ACTV/WRITE <sup>3)</sup>	Zeile in Bank aktivieren, dann Speicher schreiben. Wenn erforderlich, zuvor Zeile deaktivieren (Precharge)
00H oder 04H <sup>2)</sup>	1, 1	ACTV/READA, PRE/ACTV/READA <sup>3)</sup>	Zeile in Bank aktivieren, dann Speicher lesen. Wenn erforderlich, zuvor Zeile deaktivieren (Precharge). Nach Zugriffsende deaktivieren (Auto Precharge)
b1, b0, D, B, 0, 1	1, 1	ACTV/WRITEA, PRE/ACTV/WRITEA <sup>3)</sup>	Zeile in Bank aktivieren, dann Speicher schreiben. Wenn erforderlich, zuvor Zeile deaktivieren (Precharge). Nach Zugriffsende deaktivieren (Auto Precharge)

1): Angabe hexadezimal oder als Einzelbits (dazu Erklärung im Text); 2): Belegung von Bit OP2 ist beliebig, 3) der CRDRAM erkennt selbständig, ob zunächst ein Precharge-Ablauf erforderlich ist oder nicht

**Tabelle 8.22** CRDRAM-Kommando-Übersicht

*Erklärung zu den Schreibkommandos:*

- B: Steuerung der Bytemaskierung (vom 2. zu schreibenden Octbyte an). B = 0 Schreiben mit Bytemaske in den Bits E7...0 der Datenpakete, B = 1: Schreiben ohne Bytemaskierung. Siehe weiter unten Abbildung 8.21.

- D: Steuerung selektives oder globales Schreiben. D = 0: selektives Schreiben (betrifft nur den jeweils durch Adreßvergleich ausgewählten CRDRAM); D = 1: globales Schreiben (Broadcast; betrifft alle CRDRAMs des Rambus-Kanals gleichzeitig),
- b1, b0: Steuerung der Bitmaskierung (Tabelle 8.12). Siehe weiter unten Abbildung 8.22.

Bits b1, b0	Funktion
0, 0	keine Bitmaskierung
0, 1	bitselektives Schreiben (Write per Bit) mit statischer Bitmaskierung. Die 72 oder 64 Bits des Maskenregisters (MASK) dienen als Bitmaske und bestimmen für jedes Daten-Octbyte, welche Bits geschrieben werden und welche nicht (Maskenbit = 0: nicht schreiben, Maskenbit = 1: schreiben)
1, 0	dynamische Bitmaskierung. Die zum CRDRAM übertragenen Octbytes dienen abwechselnd als Bitmaske und als Daten, wobei die Bitmasken-Octbytes jeweils in das Maskenregister geladen werden
1, 1	Bitselektive Maskierung (Mask per Bit) mit statischer Datenbelegung. Der Inhalt des Maskenregisters wird als Daten-Octbyte interpretiert. Die zum CRDRAM geschickten Octbytes wirken als Bitmasken und steuern, ob die Maskenregister-Bits geschrieben werden oder nicht (Octbyte-Bit = 0: nicht schreiben, Octbyte-Bit = 1: schreiben)*

\*) Anwendung z. B. zum selektiven Einfärben von Flächen (Bildspeicher)

**Tabelle 8.23** Steuerung der Bitmaskierung beim Schreiben

*Hinweise:*

4. Laden des Maskenregisters: hierzu ist ein Schreibkommando mit b1, b0 = 1, 0 zu geben (dynamische Bitmaskierung). Das zuletzt in's Maskenregister geladene Octbyte bleibt dort erhalten. (Als Maskenregister-Ladekommando eignet sich ein Schreibkommando mit b1, b0 = 1, 0, in dem nur ein Octbyte zum CRDRAM übertragen wird.)
5. Vgl. das bitweise Schreiben bei den SGRAMs und WRAMs (Kapitel 5, Abschnitt 5.4.; Kapitel 6).

#### 8.4.4. Steuerregister

Ein CRDRAM enthält 6 Steuerregister (Tabelle 8.13). Jedes Register kann bis zu 64 Bits umfassen (entsprechend den Datenbits D63...0 eines Octbytes). Es sind aber stets nur ausgewählte Bitpositionen belegt. Zu Einzelheiten sei auf das einschlägige Datenmaterial verwiesen (Internet).

Registeradresse <sup>*)</sup>	Bezeichnung	Inhalt
000H	DEVICETYPE	Architektur-Version (Unterscheidung RDRAM/CRDRAM), Speichertyp, Adreßbit-Anzahl für Bank-, Zeilen- und Spaltenadresse, Zugriffsbreite ( $\cdot 8/9$ )
001H	DEVICEID	Basisadresse des Schaltkreises (am Rambus-Kanal). Belegt sind 15 Bitpositionen ID35...21 (vorgesehen zum Adreßvergleich mit den Adreßbits in den Anforderungspaketen: 16/18 MBits: A35...A21, 64/72 MBits: A35...A23)
003H	MODE	Treiberstrom der RSL-Ausgänge, Ein- und Ausschalten der internen Takt-PLL, Anwesenheit herkömmlicher RDRAMs (die Bestätigungspakete senden), weitere Angaben zur Betriebssteuerung
005H	REFROW	Refresh-Adreßangaben, Steuerbit für Übergang in Power-Down-Zustand
006H	RASINTERVAL	RAS-Zeitintervallangaben
009H	DEVICEMFGR	Herstellerangaben

\*) : HEX-Adresse gemäß Belegung der Bits A10...A3, REGSEL (im Anforderungspaket)

**Tabelle 8.24** Die Steuerregister im CRDRAM

### 8.4.5. Elementare Zugriffsabläufe (Transaktionen)

Die Abbildungen 8.18 bis 8.25 veranschaulichen wichtige Zugriffsabläufe (zu näheren Einzelheiten muß auf die Originalliteratur verwiesen werden (Internet)).

**Abbildung 8.27** Lesezugriff (Read Transaction; Rambus, Inc.)

*Erklärung:*

- 1) Senden des Anforderungspaketes (hier: mit Kommando "Aktivieren/Speicher lesen"). Im Anforderungspaket wird die vollständige Adresse (Speicherschaltkreis, Bank, Zeile, Octbyte in Spalte) übertragen.
- 2) die Zeile in der ausgewählten Bank wird aktiviert ( $\Delta$  RAS-Erregung; Zeileninhalt wird in die Leseverstärker-Anordnung übernommen). Hierfür wird die Zeit  $t_{\text{RCD}}$  benötigt.
- 3) in der aktiven Zeile (d. h. im "Leseverstärker-Cache") wird das adressierte Octbyte ausgewählt ( $\Delta$  CAS-Erregung). Nach Ablauf der hierfür erforderlichen CAS-Zugriffszeit  $t_{\text{CAC}}$  wird dieses Octbyte über den Rambus-Kanal übertragen.
- 4) ein RSTRB-Kommando wird parallel mit einer weiteren Spaltenadresse (COL b) übertragen (in 4 Taktzyklen = 8 Taktflanken über die Leitungen BusCtrl und BusEnable),
- 5) es folgen weitere Spaltenadressen nach,
- 6) das RTERM-Kommando beendet den Lesezugriff,
- 7) die gemäß den weiteren Spaltenadressen ausgewählten Octbytes werden lückenlos nacheinander über den Rambus-Kanal geliefert,
- 8) zu diesem Zeitpunkt kann ein neues Anforderungspaket gesendet werden.

**Abbildung 8.28** Schreibzugriff (Write Transaction; Rambus, Inc.)

*Erklärung:*

- 1) Senden des Anforderungspaketes (hier: mit Kommando "Aktivieren/Speicher schreiben"). Im Anforderungspaket wird die vollständige Adresse (Speicherschaltkreis, Bank, Zeile, Octbyte in Spalte) übertragen.
- 2) die Zeile in der ausgewählten Bank wird aktiviert ( $\triangle$  RAS-Erregung; Zeileninhalt wird in die Leseverstärker-Anordnung übernommen). Nach Ablauf der Aktivierungszeit  $t_{\text{RCD}}$  können die ersten Schreibdaten zum CRDRAM geschickt werden.
- 3) ein WSTRB-Kommando wird parallel mit einer weiteren Spaltenadresse (COL b) übertragen (in 4 Taktzyklen = 8 Taktflanken über die Leitungen BusCtrl und BusEnable),
- 4) es folgen weitere Spaltenadressen nach,
- 5) das WTERM-Kommando beendet den Schreibzugriff,
- 6) die gemäß den weiteren Spaltenadressen zu schreibenden Octbytes werden lückenlos nacheinander über den Rambus-Kanal geliefert (Reihenfolge: erst Spaltenadresse, darauf das zugehörige Octbyte),
- 7) zu diesem Zeitpunkt kann ein neues Anforderungspaket gesendet werden.

**Abbildung 8.29** Lese- und Schreibabläufe (Beispiele)

*Erklärung:*

- a) Lesezugriff auf eine aktive Bank (= geöffnete Seite). Die ersten Daten werden nach dem Ablauf der CAS-Zugriffszeit  $t_{\text{CAC}}$  geliefert.
- b) Schreibzugriff auf eine aktive Bank (= geöffnete Seite). Die ersten Schreibdaten dürfen dem Anforderungspaket sofort nachfolgen.
- c) Lesezugriff auf eine aktive Bank mit Auswahl einer neuen Zeile (Row Miss). Die bisher aktive Zeile (geöffnete Seite) muß zunächst deaktiviert werden. Das Lesekommando läuft also in der Form "Precharge-Aktivieren-Lesen" ab. Das Deaktivieren benötigt zusätzlich (im Vergleich zu Abbildung 8.18) die Precharge-Zeit  $t_{\text{RP}}$ .

*Hinweis:*

Aus den Abbildungen 8.18 bis 8.20 geht hervor, daß man Datenübertragungen lückenlos aneinanderreihen kann. Somit lassen sich auch schnelle Burst-Zugriffe verwirklichen. Die Schaltkreise haben aber keine Burst-Adreßzähler. Sind Burst-Zugriffe auszuführen, muß die Speichersteuerung die entsprechenden Spaltenadressen errechnen (z. B. durch +1-Zählung) und für jedes Octbyte ein Spaltenadreibpaket übertragen. (Der Vorteil des CRDRAMs: sofern es sich um Row Hits handelt, sind Burst- und wahlfreie Zugriffe gleich schnell.)

**Abbildung 8.30** Schreiben mit Bytemaskierung (Rambus, Inc.)

*Erklärung:*

In einem Speicherschreibkommando veranlaßt  $OP2 = 0$ , daß vom zweiten Octbyte an mit Bytemaskierung geschrieben wird. Das erste Octbyte hat stets eine Bytemaske: die Bits  $M7...0$  des Anforderungspaketes. Beim Schreiben der folgenden Octbytes dienen die Belegungen der Bits  $E7...0$  des jeweils zuvor übertragenen Octbytes als Schreibmaske.

Interpretation der Maskenbits: 0 = zugehöriges Byte nicht schreiben (der bisherige Speicherinhalt bleibt erhalten), 1 = Byte schreiben.

- a) Schreibablauf. Diagramm zeigt (1) Wirkung der Bits  $M7...0$  beim Schreiben des ersten Octbytes  $DIN a$  sowie (2) Nutzung der 9. Datenbits (Leitung DQ8) beim Schreiben der folgenden Octbytes ( $M b$  ist Maske für  $DIN b$  usw.),
- b) Zuordnung der Maskenbit- zu den Datenbytepositionen in den Octbytes.

*Hinweis:*

Die Bytemaskierung funktioniert nur in Speichern der Organisationsform  $\cdot 8/\cdot 64$  (also bei 16-MBit- oder 64-MBit-CRDRAMs) - und dort ist sie auch anwendungsseitig von Bedeutung (vgl. das byteselektive Schreiben bei  $\cdot 32/\cdot 64$  organisierten Speichermoduln). Soll in Speicher der Organisationsform  $\cdot 9/\cdot 72$  (18/72 MBits) byteselektiv geschrieben werden, so ist eine der beiden folgenden Möglichkeiten anzuwenden: (1) die Speichersteuerung führt explizit Read-Modify-Write-Zugriffe aus, (2) solche Zugriffe sind stets Einzelzugriffe (es wird jeweils nur ein Octbyte geschrieben; dann wirken die Bits M7...0 als Bytemaske).

**Abbildung 8.31** Schreiben mit Bitmaskierung (Rambus, Inc.)

*Erklärung:*

- a) dynamische Bitmaskierung. Schreibreihenfolge:
  - 1. Bitmaske  $m a$ ,
  - 2. Daten-Octbyte  $DIN b$  (wird geschrieben unter Steuerung von Bitmaske  $m a$ ),
  - 3. Bitmaske  $m c$ ,
  - 4. Daten-Octbyte  $DIN d$  (wird geschrieben unter Steuerung von Bitmaske  $m c$ ).
- \*) Es sind nur jene Spaltenadressen von Bedeutung, die den Daten-Octbytes vorangehen.
- b) Schreiben mit bitselektiver Maskierung.

Das Schreiben mit *statischer* Bytemaske (im Maskenregister) unterscheidet sich im Ablauf nicht vom "gewöhnlichen" Schreiben, z. B. gemäß Abbildung 8.19 oder 8.20b). Das Maskenregister kann geladen werden, indem man einen Ablauf gemäß Abbildung 8.22a nach dem Übertragen des ersten Octbytes ( $m a$ ) beendet.

**Abbildung 8.32** Registerzugriffe. a) Lesen, b) Schreiben (Rambus, Inc.)

*Erklärung:*

Registerzugriffe sind Lese- oder Schreibzugriffe mit jeweils einem Daten-Octbyte. Hierbei sind besondere Zeitvorgaben (z. B. die Mindest-Schreibzeit  $t_{\text{WREG}}$ ) einzuhalten.

**Abbildung 8.33** Refresh (Rambus, Inc.)

*Erklärung:*

Den Refresh-Ablauf im Normalbetrieb kann man als "RAS-Only-Refresh" ansehen. Ein Refresh-Zugriff ist ein globales Schreiben (Broadcast - betrifft alle CRDRAMs gleichzeitig). Hierzu muß die Speichersteuerung die Bank- und Zeilenadresse liefern (im Anforderungspaket). Damit die Schreibdaten ( $DIN a$ ) in den CRDRAMs nicht wirksam werden, müssen die Bytemasken-Bits M7...0 des Anforderungspakets mit Nullen belegt werden. Bereits aktive Banks werden vor dem Refresh automatisch deaktiviert (da alle CRDRAMs gleichzeitig angesprochen werden, kann es ist damit zu rechnen, daß es aktive Zeilen gibt, so daß die Precharge-Zeit  $t_{\text{RP}}$  stets zu berücksichtigen ist). Im Stromsparmodus mit Datenerhalt (Power Down) ist Self Refresh wirksam.

**Abbildung 8.34** Zeitverschachtelte Zugriffe (Interleaving; Rambus, Inc.)

*Erklärung:*

Die Abbildung zeigt anhand eines Beispiels, wie Zugriffe überlappt und wie Lücken eines Zugriffs mit Übertragungen ausgefüllt werden können, die zu anderen Zugriffen gehören. Überlappte Zugriffe kennen wir bereits von anderen Speicher-Architekturen her. Die Besonderheit beim CRDRAM: Beginn und Ende der eigentlichen Datentransporte wird durch STRB- und TERM-Kommandos gekennzeichnet. Hierbei handelt es sich aber lediglich um Bitpositionen auf der Leitung Busctrl, so daß die CRDRAMs nicht ohne weiteres "wissen" können, zu welchem der überlappten Zugriffe das gerade empfangene STRB- oder TERM-Kommando gehört. Der Ausweg: die CRDRAMs müssen die Kommandos abzählen. Hierzu stellt die Speichersteuerung das

PEND-Feld in den Anforderungspaketen entsprechend ein. Sind beispielsweise bereits zwei Zugriffe anhängig und startet die Speichersteuerung einen dritten, so muß das PEND-Feld in dessen Anforderungspaket auf 2 gesetzt werden. Damit "wissen" die Speicherschaltkreise, daß zwischen der Anforderung und den zugehörigen STRB- und TERM-Kommandos noch zwei weitere derartige Kommandofolgen liegen.

## 8.5. Direct Rambus DRAMs (Direct RDRAMs)

Direct Rambus ist eine (von den Firmen Rambus und Intel) weiterentwickelte Speichersubsystem-Architektur (Speicherschaltkreise, Speichersteuerung, Interface). Das Rambus-Interface und die Speicherschaltkreise (die wir im folgenden als DRDRAMs bezeichnen werden) sind durch folgende Besonderheiten (im Vergleich zu den bisher beschriebenen Rambus-Lösungen) gekennzeichnet:

- Datenwegbreite 16 bzw. 18 Bits,
  - zusätzliche Signalwege für die direkte (daher Name) Kommando- und Adreßübertragung (insgesamt 8 Leitungen),
  - differentielle Taktsignale,
  - Taktfrequenz 400 MHz<sup>\*</sup>,
  - Speisespannung der DRDRAMs: 2,5 V,
  - Speicherschaltkreise in neuartigen Gehäusen (Chip Scale Package CSP - einer Art Pin-Ball-Array mit flächenhaft angeordneten Anschlüssen),
  - verringerte Rambus-Pegel (unter Beibehaltung des Prinzips der Rambus Signaling Logic RSL),
  - die DRDRAMs enthalten 8 oder 16 Banks,
  - an eine Speichersteuerung (Memory Controller) können mehrere Direct-Rambus-Kanäle angeschlossen sein (typisch sind 1, 2 oder 4),
  - an einen Direct-Rambus-Kanal lassen sich bis zu 32 DRDRAMs anschließen (weitere 32 lassen sich über Erweiterungs-Pufferstufen nachsetzen).
- \*): bei Ausnutzung beider Taktflanken entsprechen 400 MHz 800 M Datenübertragungen/s, wobei jeweils 2 Bytes übertragen werden. Dies ergibt die gern herausgestellte DRDRAM-Datenrate von 1,6 GBytes/s.

### 8.5.1. Direct-Rambus-Speichersubsysteme

Die Struktur derartiger Speichersubsysteme soll anhand der Abbildungen 8.26 bis 8.28 veranschaulicht werden.

**Abbildung 8.35** Speichersubsysteme im Vergleich (Rambus, Inc.)

*Erklärung:*

- a) herkömmliches Speichersubsystem: Speichersteuerung (Memory Controller) + 8 Bytes breite Datenwege + herkömmliche DRAMs (asynchron oder synchron),
- b) Direct-Rambus-Speichersubsystem mit Speichersteuerung (Memory Controller) und 4 Direct-Rambus-Kanälen (Datenwegbreite jeweils 2 Bytes),
- c) Direct-Rambus-Speichersubsystem mit 2 Kanälen,
- d) Direct-Rambus-Speichersubsystem mit 1 Kanal.

Die erzielbaren maximalen Datenraten sind in Tabelle 8.14 angegeben.

Speichersubsystem gemäß Abb. 8.26	Gesamt-Datenwegbreite	maximale (Burst-) Datenrate
a)	8 Bytes	528 <sup>1)</sup> ...800 <sup>2)</sup> MBytes/s
b)	8 Bytes	6,4 GBytes/s
c)	4 Bytes	3,2 GBytes/s
d)	2 Bytes	1,6 GBytes/s <sup>3)</sup>

1): bei 66 MHz; 2): bei 100 MHz; 3): 400 MHz Takt + Ausnutzung beider Flanken  $\triangleq$  4 Bytes in 2,5 ns)

**Tabelle 8.25** Datenraten von Speichersubsystemen gemäß Abbildung 8.26

**Abbildung 8.36** Struktur eines Direct-Rambus-Speichersubsystems (Rambus, Inc.)

*Erklärung:*

Die Abbildung zeigt einen Direct-Rambus-Kanal mit Speichersteuerung und 2 angeschlossenen Speicherschaltkreisen.

- 1) Speichersteuerung (Memory Controller),
- 2) Datenwege (2 Bytes),
- 3), 4) zusätzliche Signalwege zur Kommando- und Adreßübertragung (3 bzw. 5 Bits breit),
- 5) Speicherschaltkreis mit 16 DRAM-Speicherbanks.

**Abbildung 8.37** Direct-Rambus-Speichersubsystem (Rambus, Inc.). Die Bezugszeichen entsprechen Abbildung 8.27

*Erklärung:*

Die Darstellung ergänzt Abbildung 8.27. Beide Abbildungen zusammen veranschaulichen wesentliche Merkmale von Direct-Rambus-Speichersubsystemen:

- jede Bank im Speicherschaltkreis (5) ist eine DRAM-Speicheranordnung (Storage array) mit Zeilendecoder (Row decode), Spaltendecoder (Col decode) und Leseverstärkern (Sense amp),
- die Zeileadresse wird über die zusätzlichen Signalwege ROW2...0 übertragen, die Spaltenadresse über die zusätzlichen Signalwege COLUMN4...0.

## 8.5.2. Der Direct-Rambus-Kanal

Der Direct-Rambus-Kanal (Abbildung 8.29, Tabelle 8.15) ist eine evolutionäre Weiterentwicklung des herkömmlichen Rambus-Kanals (vgl. die Abschnitte 8.3.2. und 8.3.3.):

- Taktfrequenz: 400 MHz,
- differentielle Taktsignale,
- Rambus Signaling Logic RSL mit verringerten Signalpegeln (Abbildung 8.30; vgl. auch Abschnitt 8.3.3.),
- weniger zeitkritische Signale haben CMOS-Pegel.

*Hinweis:*

Auch für die CMOS-Signale gilt, wie für die RSL-Signale, die "negative" Logik.

**Abbildung 8.38** Der Direct-Rambus-Kanal (Übersicht; Rambus, Inc.)

*Erklärung:*

- 1) Speichersteuerung (Memory Controller),
- 2) DRDRAM-Speicherschaltkreise,
- 3) Datenweg für serielle Zugriffe auf die Steuerregister (Daisy-Chain-Leitungsführung),
- 4) DRDRAM-Datenwege (2 Bytes zu je 8 oder 9 Bits),
- 5) RSL-Abschlußwiderstände (Pull-up's; vgl. Abbildung 8.7),
- 6) Taktgenerator (400 MHz),
- 7) Spannungsteiler für RSL-Referenzspannung (vgl. Abbildung 8.7),
- 8) Taktsignale. Leitungsführung wie beim herkömmlichen Rambus-Kanal: der Taktgenerator (6) sitzt hinter dem (physisch) letzten DRDRAM. Das Taktsignal führt vom Taktgenerator an die DRDRAMs (2) zur Speichersteuerung<sup>\*)</sup> (1) und von dort wieder zu den DRDRAMs zurück<sup>\*\*)</sup> Es endet an einem Leitungsabschluß. Da es sich um ein differentielles Signal handelt, sind jeweils 2 Leitungen erforderlich (x2).
- 9) zusätzliche Signalwege für Adressen und Steuersignale.

\*) : Bezeichnung "Clock to Master", \*\*) : Bezeichnung "Clock form Master".

Signal(e)	Signalpegel, Übertragungsrichtung <sup>*)</sup>	Anwendung
DQA8...0	RSL, bidirektional	Datenübertragung (Datenweg A)
DQB8...0	RSL, bidirektional	Datenübertragung (Datenweg B)
ROW2...0 (auch: RQ7...5)	RSL, Eingang	Übertragung von Zeilenadressen und Steuerangaben
COL4...0 (auch: RQ4...0)	RSL, Eingang	Übertragung von Spaltenadressen und Steuerangaben
CFM, CFMN	RSL, Eingänge (differentiell)	Takt für Busbelegungen, die zum DRDRAM geschickt werden (Clock from Master). CFM ist das "wahre", CFMN das invertierte Taktsignal
CTM, CTMN	RSL, Eingänge (differentiell)	Takt für Busbelegungen, die der DRDRAM liefert (Clock to Master). CTM ist das "wahre", CTMN das invertierte Taktsignal
SIO1, SIO0	CMOS, bidirektional	Datenweg für serielle Zugriffe auf die Steuerregister
CMD	CMOS, Eingang	Steuersignal für serielle Zugriffe auf die Steuerregister
SCK	CMOS, Eingang	Takt für serielle Zugriffe auf die Steuerregister
V <sub>REF</sub>	Eingang	RSL-Referenzspannung
V <sub>TERM</sub>	Eingang	Spannung für RSL-Leitungsabschluß
V <sub>DD</sub>	Speisespannung (2,5 V)	
GND	Masse	

\*) : aus Sicht des Speicherschaltkreises

**Tabelle 8.26** Die Signale eines Direct-Rambus-Kanals

### 8.5.3. Direct-Rambus-Speicherschaltkreise

Die Abbildungen 8.31 und 8.32 veranschaulichen den Aufbau eines DRDRAM-Schaltkreises anhand eines Blockschaltbildes und der Anschlußbelegung.

**Abbildung 8.40** Blockschaltbild eines Direct-Rambus-DRAMs (Rambus, Inc.)

*Erklärung:*

Die Abbildung zeigt einen 72-MBit-Schaltkreis der Organisationsform  $256k \cdot 18 \cdot 16d$  (256k Worte zu 18 Bits in 16 "doppelten" Banks). Zu den Banks: DRDRAMs haben entweder 8 "einfache" (single, s) oder 16 "doppelte" (double, d) Banks. "Doppelte" Banks unterscheiden sich von den "einfachen" dadurch, daß sich je 2 Banks eine Leseverstärker-Anordnung teilen. Dies hat zur Folge, daß jeweils nur eine Bank einer solchen Doppelanordnung aktiviert sein darf. Zum Rambus-Kanal hin sind 2 Datenwege (DQA, DQB) von je einem Byte (8 oder - wie hier - 9 Bits) vorgesehen. Die Übertragung eines jeden Paketes erfordert hier - wie beim CRDRAM - stets 4 Taktzyklen = 8 Taktflanken (mit anderen Worten: über jeden der beiden Datenwege DQA, DQB) werden Octbytes übertragen). Die schaltkreis-internen Datenwege (Internal DQA/DQB Data Paths) sind demgemäß 64 oder - wie hier - 72 Bits breit (beim Lesen erfolgt eine 8:1-Serialisierung, beim Schreiben eine 1:8-Deserialisierung). Im Beispiel hat die einzelne Bank 512 Zeilen und 9216 ( $64 \cdot 144$ ) Spalten. An jeden der beiden internen Datenwege sind jeweils 4608 ( $64 \cdot 72$ ) Leseverstärker angeschlossen. Demgemäß gilt: 1 Zeile = offene Seite = 9216 Bits = 128 Worte zu 72 Bits = 1k Bytes (zu 9 Bits).

**Abbildung 8.41** Anschlußbelegung eines DRDRAM-Schaltkreises (Rambus, Inc.)

### 8.5.4. Kommando- und Adreßübertragung

*Zeilen- und Spaltenpakete*

Für Zeilen- und Spaltenadressen sind unabhängige Signalwege vorgesehen, über die Zeilen- und Spaltenpakete übertragen werden. Jedes Paket entspricht 8 Taktflanken. Demgemäß enthält ein Zeilenpaket  $8 \cdot 3 = 24$  Bits (Signale ROW2...0) und ein Spaltenpaket  $8 \cdot 5 = 40$  Bits (Signale COL4...0).

Abbildung 8.33 veranschaulicht das Prinzip der Paketübertragung, die Paketstrukturen sind in den Abbildungen 8.34 und 8.35 dargestellt.

**Abbildung 8.42** Prinzip der Paketübertragung (Rambus, Inc.)

*Erklärung:*

Beide Flanken des Taktsignals (Clock) dienen zur Informationsübertragung. Die Low-High-Flanke ist die "ungerade" (odd) Flanke, die High-Low-Flanke die gerade (even; vgl. auch Abschnitt 8.3.6.). Die Übertragung eines Paketes beginnt stets mit einer "geraden" Taktflanke (G). Jedes dieser Übertragungs-Intervalle dauert 1,25 ns, und die jeweilige Taktflanke erscheint in der Mitte des betreffenden Daten-Intervalls.

**Abbildung 8.43** Zeilenpakete (Rambus, Inc.)

*Erklärung:*

- a) Aktivierungspaket (ROWA Packet), gekennzeichnet durch Bit AV = 1. Wählt Schaltkreis, Bank und Zeile aus und aktiviert die ausgewählte Zeile. Schaltkreisauswahl: DR4T/F...DR0, Bankauswahl: BR3...0, Zeilenauswahl: R8...0. RsvB = reserviert für weitere Bankauswahl (erlaubt Ausbau auf 64 Banks), RsvR = reserviert für weitere Zeilenauswahl (erlaubt Ausbau auf 2048 Zeilen),
- b) Kommandopakete (ROWR Packet), gekennzeichnet durch Bit AV = 0. Wählt Schaltkreis und Bank aus (s. hierzu Punkt a) und überträgt das auszuführende Kommando in den Bitpositionen ROP10...0 (Tabelle 8.17).

### Schaltkreisauswahl

An einen Direct-Rambus-Kanal können bis zu 32 DRDRAMs angeschlossen werden. Demgemäß ist eine 5-Bit-Auswahladresse vorgesehen, die in den DRDRAMs mit einer fest eingestellten Schaltkreisadresse (DEVID4...0) verglichen wird. Die Auswahladresse wird folgendermaßen gebildet:

- die Bits DR3...0 enthalten die Adreßbits 3...0,
- die Bits DR4T, DR4F werden gemäß Tabelle 8.16 interpretiert.

Bits DR4T, DR4F	Auswahl
0, 0	kein Schaltkreis ausgewählt ( $\triangle$ NOP)
0, 1	bei Adreßgleichheit ein Schaltkreis ausgewählt (DEVID4 = 0, DEVID3...0 = DR3...0) - d. h. Schaltkreisadresse von 0 bis 15
1, 0	bei Adreßgleichheit ein Schaltkreis ausgewählt (DEVID4 = 1, DEVID3...0 = DR3...0) - d. h. Schaltkreisadresse von 16 bis 31
1, 1	alle Schaltkreise gleichzeitig ausgewählt (Broadcast)

**Tabelle 8.27** Schaltkreisauswahl

Bits ROP10...3 <sup>1)</sup>	Mnemonic	Kommando
-	ACT	Aktivieren <sup>2)</sup>
1, 1, 0, 0, 0, x, x, x	PRER	Deaktivieren der ausgewählten Bank (Precharge)
0, 0, 0, 1, 1, 0, 0, x	REFA	Refresh-Ablauf (1). Aktivieren der ausgewählten Zeile in der ausgewählten Bank
1, 0, 1, 0, 1, 0, 0, x	REFP	Refresh-Ablauf (2). Deaktivieren (Precharge-Ablauf)
x, x, 0, 0, 0, 0, 1, x	PNDR	Überführen des ausgewählten Schaltkreises in den "Power-Down"-Stromsparszustand
x, x, 0, 0, 0, 1, 0, x	NAPR	Überführen des ausgewählten Schaltkreises in den "NAP"-Stromsparszustand <sup>3)</sup>
x, x, 0, 0, 0, 1, 1, x	NAPRC	bedingtes Überführen des ausgewählten Schaltkreises in den "NAP"-Stromsparszustand <sup>3)</sup>
x, x, x, x, x, x, x, 1	RLXR	Überführen des Schaltkreises in den jeweiligen Stromsparszustand <sup>4)</sup> (Relax)
0, 0, 0, 0, 0, 0, 0, 0	NOROP	keine Operation (NOP)

1): die Bits ROP2...0 müssen stets mit Nullen belegt sein; 2): siehe Hinweis; x = beliebige Belegung; 3) NAP  $\triangle$  Stand by (Allgemeinbezeichnung); 4) von Normalbetrieb nach NAP, von NAP nach Power Down

**Tabelle 8.28** Kommandocodes in ROWR-Paketen

### Hinweis:

Das Kommando "Aktivieren" steht nur der Vollständigkeit halber in der Tabelle. Es wird in Form des ROWA-Pakets übertragen. Die weiteren Kommandos werden hingegen in ROWR-Paketen zu den RDRAMs transportiert.

**Abbildung 8.44** Spaltenpakete (Rambus, Inc.). Weiß gelassene Bitpositionen sind mit Nullen belegt

*Erklärung:*

- a) Zugriffspaket (COLC Packet), gekennzeichnet durch Startbit S = 1. Veranlaßt die Ausführung von Lese- oder Schreibzugriffen. Schaltkreiswahl: DC4...0, Bankauswahl: BC3...0, Spaltenadresse (des 16-Byte-Wortes ( $\triangleq$  2 Octbytes)): C5...0, RsvB = reserviert für weitere Bankauswahl (erlaubt Ausbau auf 64 Banks), RsvC = reserviert für Spaltenadresse (erlaubt Ausbau auf 128 16-Byte-Worte je Spalte), COP3...0: Kommandocode (Tabelle 8.18).
- b) Maskenpaket (COLM Packet), gekennzeichnet durch Bit M = 1 (und Startbit S = 0). Enthält eine Bytemaske. Die Maskenbits MA7...0 betreffen den Datenweg DQA8...0, die Maskenbits MB7...0 den Datenweg DQB8...0. Maskenbit = 0: betreffendes Byte (im jeweiligen Octbyte) nicht schreiben, Maskenbit = 1: betreffendes Byte schreiben.
- c) Sonderfunktionspaket (COLX Packet), gekennzeichnet durch Bit M = 0. Schaltkreiswahl: DX4...0 Bankauswahl: BX3...0, Spaltenadresse (der 16-Byte-Struktur): C5...0, RsvB = reserviert für weitere Bankauswahl (erlaubt Ausbau auf 64 Banks), XOP4...0: Kommandocode (Tabelle 8.19).

*Schaltkreiswahl*

Spaltenpakete betreffen stets nur einen einzigen DRDRAM (es gibt hier keine Broadcast-Zugriffe). Demgemäß ist eine 5-Bit-Auswahladresse DC4...0 bzw. DX 4...0 vorgesehen, die in den DRDRAMs mit einer fest eingestellten Schaltkreisadresse (DEVID4...0) verglichen wird.

Bits COP3...0	Mnemonic	Kommando
x, 0, 0, 0	NOCOP	Schreibpuffer freimachen <sup>1)</sup> ; sonst keine Wirkung (NOP)
x, 0, 0, 1	WR	Schreibpuffer freimachen, dann Schreibdatenpaket in Schreibpuffer eintragen ( $\triangleq$ Schreiben in die adressierte Spaltenposition der ausgewählten Bank des ausgewählten Schaltkreises) <sup>2)</sup>
x, 0, 1, 0	RSRV	reserviert; keine Wirkung
x, 0, 1, 1	RD	Datenpaket gemäß Adressierung (Schaltkreis, Bank, Spaltenposition) lesen <sup>2)</sup>
x, 1, 0, 0	PREC	Schreibpuffer freimachen, dann ausgewählte Bank deaktivieren (Precharge-Ablauf)
1, 1, 0, 1	WRA	Schreiben mit Auto Precharge. Ablauf zunächst wie WR, dann Freimachen des Schreibpuffers (= Überführen der eben geschriebenen Daten in die DRAM-Matrix), anschließend Deaktivieren der ausgewählten Bank (Precharge-Ablauf)
x, 1, 1, 0	RSRV	reserviert; keine Wirkung
x, 1, 1, 1	RDA	Lesen mit Auto Precharge. Ablauf zunächst wie RDR, anschließend Deaktivieren der ausgewählten Bank (Precharge-Ablauf)
1, x, x, x	RLXC	Überführen des ausgewählten Schaltkreises in den jeweiligen Stromsparmmodus (Relax; vgl. Tabelle 8.17)

1), 2): siehe Hinweise im Text

**Tabelle 8.29** Kommandocodes in COLC-Paketen

*Hinweise:*

1. DRDRAMs enthalten Schreibpuffer (vgl. Abbildung 8.31). Schreibzugriffe betreffen zunächst den Schreibpuffer (sowohl die Schreibdaten als auch die Bank- und Spaltenadressen werden gepuffert). "Freimachen" des Schreibpuffers (retire) bedeutet, die Daten in die DRAM-Anordnung zu überführen.
2. Die gesamte Zugriffsadresse ergibt sich folgendermaßen: Schaltkreisauswahl + Bankauswahl (im COLC-Paket) + Zeilenadresse (vorausgegangenes Aktivierungskommando = ROWA-Paket) + Adresse des 16-Byte-Wortes (2 Octbytes) in der Spalte (im COLC-Paket)

Bits XOP4...0	Mnemonic	Kommando
0, 0, 0, 0, 0	NOXOP	keine Wirkung (NOP)
1, x, x, x, 0	PREX	Deaktivieren der ausgewählten Bank im ausgewählten Schaltkreis (Precharge-Ablauf)
x, 1, x, x, 0	RSRV	reserviert
x, x, 1, x, 0	RSRV	reserviert
x, x, x, 1, 0	RLXX	Überführen des ausgewählten Schaltkreises in den jeweiligen Stromsparmmodus (Relax; vgl. Tabelle 8.17)
x, x, x, x, 1	RSRV	reserviert; keine Wirkung

**Tabelle 8.30** Kommandocodes in COLX-Paketen

## 8.5.5. Datenübertragungsabläufe

Daten werden stets in Blöcken zu 16 Bytes (mit jeweils 8 oder 9 Bits) übertragen (2 Octbytes = 1 "Dualoct"). Lese- und Schreibzugriffe erfordern an sich dieselben Adressierungs-Vorkehrungen wie jeder andere DRAM; hier werden lediglich die Adreßangaben in Paketen über gesonderte Signalwege übertragen (Abbildungen 8.36, 8.37).

**Abbildung 8.45** Lesezugriff (Rambus, Inc.)

*Erklärung:*

- 1) Übertragen der Zeilenadresse und Aktivieren der Zeile (ROWA-Paket). Dies entfällt (in der Abbildung: "can be skipped"), wenn die Zeile schon aktiv ist (geöffnete Seite).
- 2) Übertragen der Spaltenadresse und des Zugriffskommandos (COLC-Paket),
- 3) der DRDRAM liefert die Lesedaten,
- 4) Lesedaten können lückenlos aufeinanderfolgen.

Aus den Zeitangaben in der Abbildung ist ersichtlich, daß die "eigentlichen" Zugriffszeiten (jene der DRAM-Anordnung) in derselben Größenordnung liegen wie bei den anderen DRAM-Architekturen.

**Abbildung 8.46** Schreibzugriff (Rambus, Inc.)

*Erklärung:*

- 1) Übertragen der Zeilenadresse und Aktivieren der Zeile (ROWA-Paket). Dies entfällt, wenn die Zeile schon aktiv ist (geöffnete Seite).

- 2) Übertragen der Spaltenadresse und des Zugriffskommandos (COLC-Paket),
- 3) die Speichersteuerung liefert die Schreibdaten,
- 4) Schreibdaten können lückenlos aufeinanderfolgen. (Die Spitzfindigkeiten, die mit dem Freimachen der Schreibpuffer zusammenhängen, wollen wir hier nicht näher betrachten.)
- 5) bei herkömmlichen DRAM-Architekturen werden die Schreibdaten typischerweise zusammen mit der Spaltenadresse und dem Schreibkommando geliefert, beim DRDRAM gibt es hingegen einen zeitlichen Abstand zwischen Schreibkommando und Schreibdaten. Dies bringt Vorteile beim Umsteuern der Datenwege zwischen Lese- und Schreibzugriffen (Bus Turnaround; vgl. auch Kapitel 2, Abschnitt 2.4.3.).

#### *Zugriffsüberlappung (Interleaving)*

In jedem DRDRAM gibt es 8 Banks (bzw. 8 Anordnungen aus 2 Banks mit gemeinsamen Leseverstärkern). Alle diese Banks können gleichzeitig aktiv sein. Sind mehrere DRDRAMs vorgesehen, so vervielfacht sich die Anzahl der Banks, die gleichzeitig aktiv sein können (bei nur 4 DRDRAMs wären dies bereits  $4 \cdot 8 = 32$  Banks). Somit können entsprechend viele Zugriffe lückenlos ineinander verschachtelt werden. Da jedes der hierfür benötigten Zeilen- und Spaltenpakete die Schaltkreis- und Bankadresse mitbringt, sind in den Speicherschaltkreisen keinen Sondervorkehrungen zur Steuerung des Interleaving erforderlich (vgl. demgegenüber das PEND-Feld im Anforderungspaket der CRDRAMs; vgl. die Erklärung zu Abbildung 8.25). Wichtig ist lediglich, daß die Speichersteuerung über alle Zugriffe genau Buch führt und zu den richtigen Zeitpunkten (hierfür sind Takte abzuzählen) Lesedaten liefert bzw. Schreibdaten abholt. (Dies gehört zur "Philosophie" der Direct-RDRAM-Architektur: die Speicherschaltkreise selbst zwar schnell, aber funktionell einfach auszulegen und die erforderliche "Intelligenz" in der Speichersteuerung zentral vorzusehen.)

### 8.5.6. Refresh

Refresh-Zugriffe (Abbildung 3.38) werden typischerweise als Broadcast-Zugriffe ausgeführt, d. h. sie betreffen alle Speicherschaltkreise gleichzeitig. Die Abläufe entsprechen dem RAS-Only-Refresh der "gewöhnlichen" DRAMs - die Speichersteuerung muß jeweils die Zeilen- und Bankadresse liefern. Ein Refresh-Ablauf besteht aus einem ROWR-Paket mit REFA-Kommando (Zeilen werden aktiviert) und einem (in einem Abstand gemäß Zeitvorgabe  $t_{RAS}$  folgendem) ROWR-Paket mit REFP-Kommando (Zeilen werden wieder deaktiviert (Precharge)). In den Stromspar-Betriebsarten ist ein Self Refresh wirksam.

**Abbildung 8.47** Refresh-Ablauf (Rambus, Inc.). Es ist weiterhin dargestellt, daß noch vor dem Deaktivieren (REFP-Kommando) ein ROWA-Paket gesendet werden kann, um eine bestimmte Zeile (für nachfolgende Zugriffe) zu aktivieren

### 8.5.7. Steuerregister

Für Steuerregisterzugriffe ist ein besonderer serieller Datenweg vorgesehen. Die Steuerregister selbst sind 16 Bits lang und haben ähnliche Funktionen wie in den bisher beschriebenen Rambus-Speichern. Wir verweisen deshalb für die Einzelbeschreibung der Register auf das einschlägige Datenmaterial (Internet) und beschränken uns auf eine Kurzbeschreibung der Zugriffsabläufe (Abbildung 8.39).

**Abbildung 8.48** Steuerregisterzugriffe (Rambus, Inc.). a) Schreiben, b) Lesen. \*): s. Erläuterung zum Datenweg

#### *Datenweg*

Die Signale SIO<sub>1, 0</sub> bilden den bitseriellen Datenweg. Sie sind typischerweise als Daisy-Chain-Leitungen verschaltet (vgl. die Leitungen S<sub>In</sub>, S<sub>Out</sub> der "gewöhnlichen" RDRAMs). Das eine Signal wirkt als SIO<sub>IN</sub>, das andere als SIO<sub>OUT</sub>. Bei den in Abbildung 8.39 gezeigten Abläufen wirken die SIO<sub>IN</sub> - SIO<sub>OUT</sub>-Verkettungen wie eine einzige bidirektionale Datenleitung (Bezeichnungsweise in Abbildung 8.39: beim Schreiben wird die SIO<sub>IN</sub>-Belegung nach SIO<sub>OUT</sub> weitergeleitet (repeated), beim Abholen der Lesedaten (umgekehrte Signalfußrichtung!) die SIO<sub>OUT</sub>-Belegung nach SIO<sub>IN</sub>.)

### Signalübertragung

Die Signale des seriellen Datenwegs (SIO1, 0, CMD, SCLK) sind für CMOS-Pegel spezifiziert. Richtwerte der SCLK-Taktfrequenz: Registerzugriffe max. 100 MHz, Zugriffe zum Umschalten der Stromsparbetriebsarten (hier nicht weiter beschrieben) max. 1 MHz.

Die Belegung der Datenleitung (SIO1, 0 = SIO<sub>IN</sub> - SIO<sub>OUT</sub>-Kette) wird lediglich mit der High-Low-Flanke von SCLK abgetastet, die Belegung der Steuerleitung (CMD) hingegen mit beiden Flanken.

### Paketstruktur

Jedes Pakt ist 16 Bits lang, jeder Schreib- oder Lesezugriff umfaßt 4 Pakete:

1. das serielle Anforderungspaket (Serial Request Packet SRQ). Es enthält einen 4-Bit-Kommandocode (Serial Opcode SOP4...0), eine 5-Bit-Schaltkreisadresse (SDEV4...0) und ein Steuerbit SBC (Serial Broadcast), das die gleichzeitige Auswahl aller Schaltkreise bewirkt.
2. das serielle Adreßpaket (SA Packet). Es enthält eine 12-Bit-Adresse zur Steuerregisterauswahl.
3. das serielle Datenpaket (SD Packet). Es enthält 16 Datenbits (SD15...0).
4. serielle Intervallpaket (SINT Packet). Dies dient lediglich als "Lückenfüller" (der Inhalt ist bedeutungslos).

*Schreibzugriff (Abbildung 8.39a):*

- 1) Start mit Belegung FOH auf CMD-Leitung (Ankündigung des Anforderungspaketes SRQ),
- 2) Übertragung des Anforderungspaketes SRQ - dieses enthält Schaltkreisadresse und Kommandocode,
- 3) Übertragung des Adreßpaketes SA. Hiermit wird das jeweilige Register ausgewählt.
- 4) Übertragen des Datenpaketes SD. Die Daten werden in das ausgewählte Register geschrieben.
- 5) ein Intervallpaket SINT ist erforderlich, damit im DRDRAM die Schreib- und Umschaltvorgänge (als Folge des Register-Ladens) ablaufen können.

*Lesezugriff (Abbildung 8.39b):*

- 1) Start mit Belegung FOH auf CMD-Leitung (Ankündigung des Anforderungspaketes SRQ),
- 2) Übertragung des Anforderungspaketes SRQ - dieses enthält Schaltkreisadresse und Kommandocode,
- 3) Übertragung des Adreßpaketes SA. Hiermit wird das jeweilige Register ausgewählt.
- 4) ein Intervallpaket SINT ist erforderlich, um dem DRDRAM zu ermöglichen, das adressierte Register auszulesen (währenddessen kann der serielle Datenweg von Schreiben auf Lesen umgesteuert werden),
- 5) Übertragen des Datenpaketes SD mit den Lesedaten zur Speichersteuerung. Anschließend kann der serielle Datenweg wieder auf Schreiben umgesteuert werden.

## 8.5.8. Speichermoduln (RIMM-Moduln)

Die Abbildungen 8.40 bis 8.43 veranschaulichen den Aufbau von Speichermoduln, die mit Direct-Rambus-DRAMs bestückt sind. Sie haben ähnliche Abmessungen wie die 168-poligen SDRAM-Moduln (Kapitel 5, Abschnitt 5.7.1.). Kennzeichnende Merkmale:

- 168 Anschlüsse,
- Anschlußabstand: 1 mm,
- Außenabmessungen (Leiterplattenfläche) 145 · 36 mm,
- Speisespannung: 2,5 V,
- Bestückung mit maximal 16 DRDRAMs (ergibt bis zu 128 MBytes bei Einsatz von 64/72-MBit-Typen),
- die Leiterplatte enthält lediglich die RDRAMs sowie einen EEPROM zur Konfigurationsspeicherung (Serial Presence Detect), also keine Takt-PLL und keine Pufferstufen.

**Abbildung 8.49** Abmessungen (Vorderansicht; Rambus, Inc.)

**Abbildung 8.50** RIMM-Modul: Blockschaltbild (Rambus, Inc.)

*Erklärung:*

Der Direct-Rambus-Kanal wird auf dem Modul gleichsam durchgereicht. Die mit L als erstem Buchstaben bezeichneten Signale treten in das Modul ein, die sinngemäß mit R bezeichneten verlassen es wieder.

Solche Moduln können selektiv bestückt werden (mit 1, 2, 3 usw. bis zu 16 DRDRAMs). Dies ergibt die (am Markt erwünschte) "Feinstufigkeit" (Fine Granularity) der Speicherbestückung und -Aufrüstung (z. B. könnte man auf Grundlage von 64/72-MBit-DRDRAMs Moduln mit 8, 16, 24, 32 usw. MBytes anbieten).

**Abbildung 8.51** RIMM-Modul: Anschlußbelegung (Rambus, Inc.)

**Abbildung 8.52** Modul-Steckverbinder (AMP)

## 8.6. SLDRAMs

SLDRAM-Schaltkreise und Speichersubsysteme beruhen auf Spezifikationen, die von einer herstellerunabhängigen Vereinigung ausgearbeitet werden (anfänglich: SLD RAM Consortium, seit 1998: SLD RAM Inc.) Es handelt sich um eine nicht auf Gewinn ausgerichtete ("non profit") Organisation, in der eine Vielzahl von Schaltungsherstellern zusammenarbeiten. Das Ziel besteht darin, Architekturen für moderne hochleistungsfähige Speicherschaltkreise und Subsysteme zu entwickeln. Die einschlägigen Standards sollen "offen" sein, also allen unentgeltlich zur Verfügung stehen und nicht von einem einzigen Hersteller kontrolliert werden. Die Standardisierung im Rahmen des JEDEC und des IEEE ist vorgesehen (der IEEE-Vorschlag (Draft) heißt IEEE P15696.7-199X). Diese Aktivitäten stehen in direktem Wettbewerb mit den Rambus-Entwicklungen. Die Rambus-Spezifikationen sind aber "proprietäre" Standards (die von Rambus, Inc. - und (Direct Rambus) von Intel - kontrolliert werden). Um sie (in Produkten) zu nutzen, sind Lizenzgebühren zu zahlen. (Allerdings: ist ein Standard "offen", so bedeutet dies lediglich, daß man ihn, ohne etwas zahlen zu müssen, verwenden kann, um eigene Produkte von Grund auf zu entwickeln (z. B. einen Controller, der die SLD RAM-Signalspiele unterstützt). Die Nutzung von Patenten und Dienstleistungen wird in der Regel auch nicht umsonst sein.)

Zur Bezeichnung: SLD RAM hieß früher SyncLink. Diese Bezeichnung wurde aber aus Gründen des Warenzeichen-Rechts aufgegeben. Kurzdarstellung des Entwicklungsweges: Scalable Coherent Interface (SCI) → RamLink (ein aus SCI abgeleitetes Speicherinterface) → SyncLink → SLD RAM.

### 8.6.1. SLD RAM-Speichersubsysteme

Die grundlegende "Philosophie" des SLD RAM entspricht jener des Rambus:

- herkömmliche DRAM-Anordnungen in den Speicherschaltkreisen,
- Serialisierung der Zugriffe über vergleichsweise schmale Datenwege,
- synchrone Übertragung mit extremen Taktfrequenzen unter Ausnutzung beider Taktflanken,
- Open-Drain-Interfaces mit Stromsteuerung und geringem Signalhub,
- paketweise Übertragung von Kommandos, Adressen und Daten,
- Anordnung mehrerer DRAM-Banks (8, 16 oder mehr) im Speicherschaltkreis,
- Takt und Daten fließen jeweils in die gleiche Richtung,
- Schaltungsauswahl über Adreßvergleich,
- Steuerung der Initialisierung über Daisy-Chain-Leitungen.

*Die Gemeinsamkeiten mit Direct RDRAM:*

- 18 Bits breite bidirektionale Datenwege,
- gesonderte taktsynchrone Signalwege zur Adreß- und Kommandoübertragung,
- vergleichsweise einfache Strukturen in den Speicherschaltkreisen, Konzentration der "Intelligenz" in der Speichersteuerung.

Die *Unterschiede* liegen in den Einzelheiten (Abbildungen 8.44 bis 8.50).

**Abbildung 8.53** Einfaches SLDRAM-Speichersubsystem (SLDRAM, Inc.)

*Erklärung:*

- 1) Speichersteuerung (Memory Controller),
- 2) Speicherschaltkreise. Einem Memory Controller können bis zu 8 Einrichtungen nachgeschaltet werden (entweder einzelne Speicherschaltkreise oder - über Pufferstufen - Speichermoduln).
- 3) Signalweg für Kommandos und Adressen (CommandLink),
- 4) Datenweg (DataLink; 18 Bits),
- 5) Daisy-Chain-Signale.

Die Kommando- und Datenwege haben eigene Taktsignale.

**Abbildung 8.54** SLDRAM-Speichersubsysteme mit gepufferten Speichermoduln (SLDRAM, Inc.)

*Erklärung:*

- a) nur der Kommandosignalweg ist gepuffert,
- b) Kommandosignal- und Datenweg sind gepuffert.

Wo Pufferstufen anzuordnen sind, richtet sich nach der jeweiligen kapazitiven Belastung. Taktsignale werden mitgepuffert, es sind aber keine besonderen PLL-Schaltkreise erforderlich.

*Hinweis:*

Zu den Besonderheiten der SLDRAMs gehört die Möglichkeit, Latenzzeiten sowie den Versatz (Offset) zwischen Takt und Daten feinstufig einstellen zu können. Dies ermöglicht es, die durch Pufferstufen bedingten zusätzlichen Verzögerungszeiten automatisch zu berücksichtigen.

**Abbildung 8.55** Speichersubsystemstrukturen (IEEE P15696.7-199X)

*Erklärung:*

- a) einfaches Speichersubsystem. Ein Kommandosignal- und ein Datenweg.
- b) Speichersubsystem mit gemeinsamem Kommandosignalweg und mehreren Datenwegen (größere Zugriffsbreite),
- c) Speichersubsystem mit gepufferten Speichermoduln und mehreren Datenwegen.

Darüber hinaus kann man Konfigurationen mit mehreren vollständigen SLDRAM-Interfaces an einer Speichersteuerung, mit mehreren Speichersteuerungen an einem Systembus usw. aufbauen.

## 8.6.2. SDRAM-Speicherschaltkreise

Weitere Einzelheiten wollen wir anhand eines Ausführungsbeispiels überblicksweise erläutern (Abbildungen 8.47 bis 8.56, Tabellen 8.20 bis 8.30). Es handelt sich um einen 72-MBit-Speicherschaltkreis der Organisationsform 4M · 18. Weitere Merkmale:

- 200 MHz Taktfrequenz,
- 400 MHz Datenübertragungsfrequenz (da beide Taktflanken ausgenutzt werden),
- 800 MBytes/s Datenrate (da mit jeder Taktflanke 2 Bytes übertragen werden),
- 8 DRAM-Banks,
- jede Bank hat 1024 Zeilen und 128 Spalten zu 72 Bits,
- bis zu 8 Zugriffe gleichzeitig (zeitmultiplex) ausführbar,
- Burst-Längen (einstellbar): 4 oder 8 (jeweils · 2 Bytes, also 8 oder 16 Bytes),
- Speisespannung: 2,5 V.

**Abbildung 8.56** Anschlußbelegung eines SDRAMs (SDRAM, Inc.). Ansicht von oben

### Erklärung:

Die Speicherschaltkreise sind in senkrecht zu montierenden Plastikgehäusen mit 64 Anschlüssen untergebracht (VSMP = Vertical Surface Mount Package; Breite: 10,16 mm (400 mil), Anschlußabstände: 0,8 mm (auf jeder Seite gemessen) bzw. 0,4 mm (zwischen benachbarten Anschlüssen auf beiden Seiten).

**Abbildung 8.57** Blockschaltbild eines SDRAMs (SDRAM, Inc.)

Signal(e)	Signalpegel, Übertragungsrichtung	Anwendung
DQ17...0	SLIO <sup>*</sup> , bidirektional	Datenübertragung (Speicherinhalte, Register-Lesedaten)
DCLK0, DCLK0#	SLIO, differentiell, bidirektional	1. Takt für Datenübertragung
DCLK1, DCLK1#	SLIO, differentiell, bidirektional	2. Takt für Datenübertragung
FLAG	SLIO, Eingang	kennzeichnet Start eines Anforderungspakets
CA9...0	SLIO, Eingang	Kommandosignalweg: Kommandocodes, Adressen, Register-Schreibdaten
CCLK, CCLK#	SLIO, differentiell	Takt für Kommandoübertragung
LISTEN	LVC MOS, Eingang	Einleiten bzw. Verlassen der Stromsparbetriebsart "Standby"
LINKON	LVC MOS, Eingang	Einleiten bzw. Verlassen der Stromsparbetriebsart "Shutdown"
RESET#	LVC MOS, Eingang	Hardware-Rücksetzen
SI	LVC MOS, Eingang	Daisy-Chain-Eingang zu Initialisierungszwecken
SO	LVC MOS, Ausgang	Daisy-Chain-Ausgang zu Initialisierungszwecken

TEST	Eingang	mit Masse verbinden
V <sub>REF</sub>	Eingang	SLIO-Referenzspannung
V <sub>DD</sub>	Speisespannung (2,5 V)	
V <sub>SS</sub>	Masse	
V <sub>DDQ</sub>	Speisespannung (2,5 V) für Treiberstufen	
V <sub>SSQ</sub>	Masse für Treiberstufen	

\*) : siehe nachfolgenden Text

**Tabelle 8.31** Anschlußbelegung eines SLDRAM-Schaltkreises

### 8.6.3. Das SDRAM-Interface (SLIO)

Die "elektrische" Interface-Spezifikation für die Hochgeschwindigkeits-Signalwege (SLIO) entspricht - mit gewissen Vereinfachungen - SSTL-2. Merkmale:

- Referenzspannung  $V_{REF} = 1,25 \text{ V}$  ( $\triangleq 0,5 V_{DD}$ ),
- einseitiger Leitungsabschluß durch Pull-up-Widerstände (am Leitungsende),
- Serienwiderstände (Richtwert:  $20 \Omega$ ) in den Stickleitungen (Stubs) zu den Speicherschaltkreisen bzw. Pufferstufen,
- Low-Pegel:  $0,9 \text{ V}$  ( $\leq V_{REF} - 0,3 \text{ V}$ ), High-Pegel:  $1,6 \text{ V}$  ( $\geq V_{REF} + 0,2 \text{ V}$ ), Signalhub  $0,7 \text{ V}$ .
- Stromsteuerung. Nach dem Einschalten werden die Treiberströme von der Speichersteuerung aus  $\infty$  eingestellt, daß sich die genannten Logikpegel ( $0,9 \text{ V}$  bzw.  $1,6 \text{ V}$ ) ergeben.
- im Gegensatz zum Rambus verwendet SLIO *positive* Logik (High = logisch 1).

### 8.6.4. Taktsignale

Es sind 3 differentielle Taktsignale vorgesehen:

- CCLK: dies ist ein von der Speichersteuerung gelieferter durchlaufender Takt, der sowohl zur Kommandoübertragung als auch als Referenztakt zur Bildung des Lesetaktes im Speicherschaltkreis dient.
- DCLK1, 0 sind 2 alternativ zur Datenübertragung nutzbare Takte. Es wird jeweils nur einer dieser Takte verwendet (Auswahl durch Speichersteuerung).

Wichtig ist, daß Takt und Daten jeweils in die gleiche Richtung fließen. Während dies beim Rambus auf sozusagen mechanische Weise erreicht wird (vgl. die Abbildungen 8.8 und 8.29), hat man hier "echte" bidirektionale Taktsignale vorgesehen. Beim Schreiben wird DCLK1 oder DCLK0 vom Controller erregt, beim Lesen hingegen von einem Taktgenerator im Speicherschaltkreis (eine PLL, die mit CCLK synchronisiert wird; vgl. Abbildung 8.48). Das Umschalten und Synchronisieren dauert jeweils eine gewisse Zeit: dies ist der Grund dafür, daß man 2 Takte vorgesehen hat. Hat der Controller einen Zugriff, in dem solche Pausen auftreten können, z. B. mit DCLK0 gestartet, so kann er die Pause mit einem weiteren Zugriff füllen und hierfür DCLK1 verwenden.

### 8.6.5. Paketstrukturen

Es gibt Datenpakete und Anforderungspakete (Abbildungen 8.49 und 8.50, Tabelle 8.21). Datenpakete (Data Packets) werden über den Datenweg (DQ17...0) übertragen (Takt = DCLK0 oder DCLK1), Anforderungspakete (Request Packets) über den Kommandosignalweg (CA9...0; Takt = CCLK).

**Abbildung 8.58** Datenpakete (SLDRAM, Inc.). a) Burst-Länge =4, b) Burst-Länge = 8

**Abbildung 8.59** Anforderungspaket (Beispiel; SLDRAM, Inc.). 1 - Beginn des Pakets, 2 - letzte vorausgegangene Signalbelegung, 3 - die 4 Signalbelegungen des Pakets

*Erklärung:*

Die Abbildung soll die Symbolik der Original-Dokumentation veranschaulichen. In den folgenden Tabellen geben wir die Strukturen der einzelnen Pakete auf vereinfachte Weise an: die Flag-Belegungen werden weggelassen, und es werden nur die 4 jeweils "gültigen" Belegungen der CA-Leitungen dargestellt.

- 1) FLAG = 1 kennzeichnet den Beginn eines Paketes,
- 2) die vorausgegangene Belegung ist für das aktuelle Paket ungültig,
- 3) ein Paket besteht aus 4 aufeinanderfolgenden Belegungen der Leitungen CA9...0. Es wird in 2 Takten (= 4 Taktflanken) übertragen. Pakete dürfen lückenlos aufeinanderfolgen.

Kommandosignalleitungen										
CA9	CA8	CA7	CA6	CA5	CA4	CA3	CA2	CA1	CA0	
ID8 - Schaltkreisauswahl -								ID0	CMD5	
CMD4 - Kommandocode -				CMD0		BNK2 - Bank -		BNK0	ROW9	ROW8
ROW7 - Zeilenadresse -						ROW0		0	0	
0	0	0	COL6 - Spaltenadresse -				COL0			

**Tabelle 8.32** Lese-, Schreib- oder Zeilenkommando-Anforderungspaket

*Erklärung:*

Dies ist gleichsam das "Grundformat". Die weitere Interpretation bestimmt der Kommandocode (CMD5...0). Einzelheiten dazu in den nachfolgenden Tabellen und Erklärungen.

*"Vorsortierung":*

- CMD5 = 0: Lese- und Schreibkommandos,
- CMD5 = 1: Zeilen- und Sonderkommandos.

## 8.6.6. Kommando-Übersicht

### 8.6.6.1. Lese- und Schreibkommandos

Mit CMD2...0 wird der jeweilige Zugriff bestimmt (Tabelle 8.22). CMD4 steuert die Art des Zugriffs, CMD3 die Burst-Länge:

- CMD4 = 0: Zugriff auf geöffnete Seite (= aktive Zeile; Page Access). Zeilenadresse im Paket wird ignoriert.
- CMD4 = 1: Zugriff auf Bank mit Aktivieren der Zeile (Bank Access). Zeilenadresse im Paket wählt zu aktivierende Zeile aus.
- CMD3 = 0: Burst-Länge = 4 (8 Bytes),
- CMD3 = 1: Burst-Länge = 8 (16 Bytes).

*Hinweis:*

Die im Paket mit Nullen belegten Bitpositionen stehen zur Verfügung, um die Adreßangaben zu verlängern. Beispielsweise könnte mit den 2 bzw. 3 freien Bits die Anzahl der adressierbaren Zeilen vervierfacht und die Anzahl der adressierbaren Spalten verachtacht werden (wahrscheinlich wird man aber wohl zunächst die Anzahl der Banks vergrößern).

CMD2...0	Wirkung
0H	Lesen mit Takt DCLK0, Zeile aktiv lassen
1H	Lesen mit Takt DCLK1, Zeile aktiv lassen
2H	Lesen mit Takt DCLK0, anschließend Zeile deaktivieren (Precharge)
3H	Lesen mit Takt DCLK1, anschließend Zeile deaktivieren (Precharge)
4H	Schreiben mit Takt DCLK0, Zeile aktiv lassen
5H	Schreiben mit Takt DCLK1, Zeile aktiv lassen
6H	Schreiben mit Takt DCLK0, anschließend Zeile deaktivieren (Precharge)
7H	Schreiben mit Takt DCLK1, anschließend Zeile deaktivieren (Precharge)

**Tabelle 8.33** Lese- und Schreibkommandos

*Hinweis:*

Offensichtlich steuert CMD2 den Zugriff (0: Lesen, 1: Schreiben), CMD1 die Deaktivierung (0: nicht deaktivieren, 1: deaktivieren) und CMD0 die Taktauswahl (zwischen DCLK0 und DCLK1).

### 8.6.6.2. Zeilen- und Sonderkommandos

Diese Kommandos sind vorgesehen, um die "eigentlichen" Zugriffe vorzubereiten und um die Speicherschaltkreise zu initialisieren, zu konfigurieren und zu verwalten (Tabelle 8.23).

CMD5...0	Wirkung
20H	nichts tun (NOP)
21H	Zeile öffnen (Aktivieren)
22H	Zeile schließen (deaktivieren, Precharge)
23H	Steuerregister schreiben - Paket-Format gemäß Tabelle 8.24
24H	Zustandsregister lesen mit Takt DCLK0 - Paket-Format gemäß Tabelle 8.25
25H	Zustandsregister lesen mit Takt DCLK1 - Paket-Format gemäß Tabelle 8.25
27H	Ereignis auslösen (Event) - Paket-Format gemäß Tabelle 8.26
28H	Lesen zwecks Synchronisation (Read Sync); beide DCLK-Takte werden angesteuert. Der Schaltkreis liefert ein besonderes Synchronisations-Bitmuster
29H	Lesen zwecks Synchronisation anhalten (Stop Read Sync)
2AH	DCLK-Takte (beide gleichzeitig) nach Low treiben
2BH	DCLK-Takte (beide gleichzeitig) nach High treiben
2CH	Schreiben zwecks Synchronisation (Write Sync); beide DCLK-Takte werden angesteuert. Bewirkt, daß der Schaltkreis die ankommenden Daten als Synchronisations-Bitmuster interpretiert
2EH	DCLK-Takte abschalten (Disable). Bewirkt, daß der Speicherschaltkreis die Takte nicht mehr treibt (Treiberausgänge werden hochohmig)
2FH	DCLK-Takte zyklisch umschalten (Toggling) gemäß Referenztakt (CCLK)

**Tabelle 8.34** Zeilen- und Sonderkommandos (ohne reservierte Codes)

*Hinweis:*

Die Kommandos 28H...2FH dienen zum Initialisieren und Kalibrieren des Speicherschaltkreises (Taktsynchronisation, Einstellen der Latenzzeiten, Einstellen der Verzögerung (Offset) zwischen Takt und Lesendaten).

Kommandosignalleitungen									
CA9	CA8	CA7	CA6	CA5	CA4	CA3	CA2	CA1	CA0
ID8	- Schaltkreisauswahl -							ID0	CMD5
CMD4	- Kommandocode <sup>-1)</sup>			CMD0	SID4			- Unterauswahl <sup>-2)</sup> SID0	
REG6	- Registeradresse (Steuerregister) -					REG0	0	0	0
RD9	- Registerdaten <sup>-3)</sup>								RD0

1)...3): siehe Erklärung im Text

**Tabelle 8.35** Register-Schreibpaket

*Erklärung:*

Mit diesem Paket werden Daten in das ausgewählte Steuerregister (Control Register) eingeschrieben.

- 1) Kommandocode = 23H,
- 2) Originalbezeichnung: Device SUB-ID Value. Wird in den Adreßvergleich einbezogen. Zu Einzelheiten siehe Abschnitt 8.6.6.4.

Länge eines Steuererregisters: 20 Bits (Architektur-Definition mit Erweiterungs-Reserven). In den meisten Registern sind nur 8 Bits (oder weniger) belegt; im Kommando werden nur 10 Bits geliefert. Nicht belegte Bitpositionen sollten mit Nullen gefüllt werden. Siehe weiterhin Abschnitt 8.6.6.3.

Kommandosignalleitungen												
CA9	CA8	CA7	CA6	CA5	CA4	CA3	CA2	CA1	CA0			
ID8								- Schaltkreisauswahl -		ID0	CMD5	
CMD4				- Kommandocode - <sup>*)</sup>		CMD0		REG6			- Registeradresse -	REG2
REG1	REG0	0	0	0	0	0	0	0	0	0		
0	0	0	0	0	0	0	0	0	0	0		

\*) : 24H oder 25H

**Tabelle 8.36** Register-Lesepaket

**Erklärung:**

Mit diesem Paket wird ein bestimmtes Zustandsregister (Status Register) in einem bestimmten Schaltkreis zum Lesen ausgewählt. Die gelesenen Daten werden später über den Datenweg (DQ17...0) zur Speichersteuerung geliefert.

Länge eines Zustandsregisters: 72 Bits ( $\Delta$  4 Taktflanken = 2 Taktzyklen). Typischerweise sind nicht alle Bits belegt. Siehe weiterhin Abschnitt 8.6.6.5.

Kommandosignalleitungen												
CA9	CA8	CA7	CA6	CA5	CA4	CA3	CA2	CA1	CA0			
ID8								- Schaltkreisauswahl -		ID0	CMD5	
CMD4				- Kommandocode - <sup>1)</sup>		CMD0		SID4			- ID-Untercode -	SID0
E6						- Ereigniscode - <sup>2)</sup>		E0	0	0	0	
div. Einstellangaben (Adjust Settings) <sup>3)</sup>												

1)...3): siehe Erklärung im Text

**Tabelle 8.37** Ereignispaket

**Erklärung:**

- 1) Kommandocode = 27H,
- 2) siehe Tabelle 8.27,
- 3) siehe Original-Dokumentation (Internet). Beispiele: (1) Feineinstellung der Lese- und Schreiblatenzzeiten bei geöffneter Seite (Funktion "Increment/Decrement Fine Read Vernier"), (2) Einstellung des Versatzes (Offset) zwischen Takt und Lesedaten.

Ereignisse (Events) sind Kommandos, die keine besonderen Adreßangaben (Bank-, Zeilen- oder Spaltenadressen) benötigen.

E6...0	Wirkung
0H	Schaltkreis zurücksetzen, ID-Register auf 255 und SUB-ID-Register auf 15 stellen (Hard Reset)
1H	Schaltkreis zurücksetzen, ID- und SUB-ID-Register aber nicht verändern (Soft Reset)
2H	Auto Refresh
3H	alle Zeilen deaktivieren (in allen Banks des Schaltkreises)
4H	Self Refresh einleiten
5H	Self Refresh beenden
6H	Einstellungen ausführen (Adjust Settings)
7H	reserviert

**Tabelle 8.38** Ereigniscodes

### 8.6.6.3. Steuerregister

Die Steuerregister dienen zum Halten der Schaltkreisadresse sowie von Frequenz- und Latenzzeitangaben (Tabelle 58.28).

Registeradresse	Belegung
0H	ID: Schaltkreis-Adresse (8 Bits ID7...0). Wert wird beim Initialisieren eingestellt und mit der ID-Angabe im jeweiligen Kommandopaket verglichen
1H	SUB-ID: ID-Untercod (4 Bits) als "Verlängerung" der Schaltkreis-Adresse. Wert wird beim Initialisieren eingestellt und mit der ID-Angabe im jeweiligen Kommandopaket verglichen (betrifft "Register schreiben" und "Ereignis auslösen")
2H	Betriebsfrequenz (Operating Frequency, 8 Bits). 01H: 100 MHz, 02H: 200 MHz, 04H: 300 MHz, 08H: 400 MHz, 10H: 500 MHz, 20H: 600 MHz
3H	Test. Für herstellerepezifische Testzwecke vorgesehen
4H	Leselatenzzeit bei geöffneter (aktiver) Seite (Page Read Delay, 8 Bits). Enthält Verzögerungszeit zwischen Empfang des Lesekommandos und dem Senden der ersten Lesedaten (in Taktflanken)*
5H	Schreiblatenzzeit bei geöffneter (aktiver) Seite (Page Write Delay, 8 Bits). Enthält Verzögerungszeit zwischen Empfang des Lesekommandos und dem Senden der ersten Lesedaten (in Taktflanken)
6H	Leselatenzzeit bei inaktiver Seite (Bank Read Delay, 8 Bits). Enthält Verzögerungszeit zwischen Empfang des Lesekommandos und dem Senden der ersten Lesedaten (in Taktflanken)*
7H	Schreiblatenzzeit bei inaktiver Seite (Bank Write Delay, 8 Bits). Enthält Verzögerungszeit zwischen Empfang des Lesekommandos und dem Senden der ersten Lesedaten (in Taktflanken)

\*) kann durch Ereignisauslösung verändert werden (Feineinstellung). Der aktuelle Wert ist über das Verzögerungs-Zustandsregister zurücklesbar

**Tabelle 8.39** Steuerregister

### 8.6.6.4. Adreßerkennung

Zur Adreßerkennung werden in den Speicherschaltkreisen die Inhalte der ID- und der SUB-ID-Register (d. h. - vgl. Tabelle 8.28 - der Steuerregister 0 und 1) mit den entsprechenden Adreßangaben in den Anforderungspaketen verglichen. Jeder Schaltkreis hat somit eine 8-Bit-ID-Adresse (Bits ID7...0) und eine 4-Bit-”Unteradresse” (SUB-ID3...9). Die höchstwertigen Bits in den Anforderungspaketen (ID8, SUB-ID4) dienen nicht dem Adreßvergleich, sondern der Adressierungssteuerung. Wir unterscheiden folgende Adressierungsweisen:

#### 1. Einzelauswahl

Es wird jeweils nur ein Schaltkreis ausgewählt. Je nach Kommando werden entweder nur die Bits ID7...0 allein oder der Verbund von ID7...0 und SUB-ID3...0\*) in den Vergleich einbezogen. Die Bitposition ID8 im Anforderungspaket muß fest mit Null belegt sein, ebenso SUB-ID4, falls das SUB-ID-Feld in den Adreßvergleich einbezogen ist.

\*) : dies betrifft nur die Kommandos “Register schreiben” und “Ereignis auslösen”.

#### 2. Gruppenauswahl (Multicast) und globale Auswahl (Broadcast)

Es gibt 2 Arten der Gruppenauswahl:

1. die ID-gesteuerte Gruppenauswahl. Ist ID8 = 1, so kann die Belegung des gesamten ID-Feldes (ID8...0 als Binärzahl interpretiert) Werte zwischen 256 und 511 annehmen. Jeder dieser Belegungen ist ein Bereich von Schaltkreisen mit bestimmten Belegungen der ID-Register zugewiesen, der hiermit ausgewählt wird<sup>1)</sup>. Beispiel 1: ID-Feld = 282 → Auswahl der Schaltkreise mit den ID-Belegungen 26 und 27, Beispiel 2: ID-Feld = 511 → Auswahl der Schaltkreise mit den ID-Belegungen 255...0<sup>3)</sup>.

1. die SUB-ID-gesteuerte Gruppenauswahl. Ist SUB-ID4 = 1, so kann die Belegung des gesamten SUB-ID-Feldes (SUB-ID4...0 als Binärzahl interpretiert) Werte zwischen 16 und 31 annehmen. Jeder dieser Belegungen ist ein Bereich von Schaltkreisen zugewiesen<sup>2)</sup>, der hiermit ausgewählt wird. Beispiel 1: SUB-ID-Feld = 17 → Auswahl der Schaltkreise mit den SUB-IDs 0...3, Beispiel 2: SUB-ID-Feld = 31 → Auswahl der Schaltkreise mit den SUB-IDs 15...0<sup>3)</sup>.

- 1) die Original-Dokumentation enthält Tabellen, die die Auswahl im einzelnen beschreiben,
- 2) es handelt sich um eine “Unterauswahl” aus einer Gruppe von Schaltkreisen, die bereits über ID8...0 ausgewählt wurde,
- 3) das heißt: es werden *alle* betreffenden Schaltkreise ausgewählt (globale Auswahl (Broadcast)).

### 8.6.6.5. Zustandsregister

Die Zustandsregister enthalten feste Konfigurationsangaben sowie die aktuell eingestellten Zeitkennwerte (Tabelle 8.29)

Registeradresse	Belegung
0H	Konfiguration (Configuration, 32 Bits). Hersteller-Code und Speicherorganisation (Abbildung 8.51a, Tabelle 8.30)
1H	aktuelle Latenzzeiten (Actual Delays). Dient zum Zurücklesen der tatsächlich eingestellten Werte
2H	minimale Latenzzeiten (Minimum Delays)
3H	maximale Latenzzeiten (Maximum Delays)
4H	Testregister. Herstellerspezifisch

5H	$t_{RAS}/t_{RP}$	Zeitkennwerte (Timing Parameters). Über diese Register können die Datenblatt-Kennwerte des Schaltkreises ausgelesen werden. Jedes Register enthält 2 Kennwerte, die wiederum aus einer digitalen und einer analogen Angabe bestehen (Abbildung 8.52)
6H	$t_{RC1}/t_{RC2}$	
7H	$t_{RCD}/t_{XSR}$	
8H	$t_{WR}/t_{WR}$	
9H	$t_{PR}/t_{BR}$	
AH	$t_{PW}/t_{BW}$	

**Tabelle 8.40** Zustandsregister

**Abbildung 8.60** Zustandsregister (SLDRAM, Inc.)

*Erklärung:*

a) *Konfigurationsregister:*

- 1) Herstellercode. Siehe Tabelle 8.30.
- 2) Data Offset. DO0 = 0: nur Wort-Offset, DO0 = 1: Wort- und Bit-Offset. Zweck: Einstellung des Zeitversatzes zwischen Daten und Taktflanken (siehe die einschlägige Erklärung zu Abbildung 8.53).
- 3) Betriebsfrequenz (im 1-aus-n-Code),
- 4) Anzahl der Banks. Binärzahl  $n \triangleq 2^n$  Banks (1H: 2, 2H: 4, 3H: 8, 4H: 16, 5H: 32, 6H: 64),
- 5) Anzahl der Zeilen (je Bank). Binärzahl  $n \triangleq 2^{n+8}$  Zeilen (1H: 512, 2H: 1k, 3H: 2k, 4H: 4k, 5H: 8k),
- 6) Anzahl der Datenausgänge des Schaltkreises (0H: 4, 1H: 8, 2H: 9, 3H: 16, 4H: 18, 5H: 32, 6H: 36, 7H: 64, 8H: 72),
- 7) Anzahl der Spalten (je Zeile). Binärzahl  $n \triangleq 2^{n+3}$  Spalten (1H: 16, 2H: 32, 3H: 64, 4H: 128).

b) *Latenzzeiten (Read/Write Delays).* Angabe in Taktflanken (Clock Ticks). *Page*  $\triangleq$  Zugriff bei aktiver (geöffneter) Seite, *Bank*  $\triangleq$  Zugriff bei inaktiver Seite (dem eigentlichen Zugriff geht ein Aktivieren voran). Es gibt 3 derartige Register, die alle gemäß Abbildung 8.51b aufgeteilt sind:

- aktuelle Latenzzeiten. Die einzelnen Felder des Registers enthalten die aktuellen digitalen Werte.
- minimale Latenzzeiten. Die einzelnen Angaben sind Summen aus den jeweiligen digitalen und (in's Digitale gewandelten) analogen Werten, aufgerundet auf die nächsthöhere Taktflanken-Anzahl (zu den digitalen und analogen Werten vgl. auch die Erklärung zu Abbildung 8.52). Die Angaben sind auf die für den Schaltkreis spezifizierte minimale Taktzykluszeit bezogen.
- maximale Latenzzeiten. Die einzelnen Angaben werden genauso gebildet wie die minimalen Latenzzeiten.

M6...0	Hersteller	M6...0	Hersteller
00H	Fujitsu	09H	National/Panasonic
01H	Hitachi	0AH	NEC
02H	Hyundai	0BH	Nippon Steel
03H	IBM	0CH	OKI

04H	LG Semicon	0DH	Samsung
05H	Micron	0EH	Siemens
06H	Mitsubishi	0FH	Texas Instruments
07H	Mosel Vitelic	10H	Toshiba
08H	Motorola	11H	Vanguard

**Tabelle 8.41** Hersteller-Kennungen (nach: SLDRAM, Inc.)

**Abbildung 8.61** Digitale und analoge Kennwertangaben (SLDRAM, Inc.)

*Erklärung:*

- Übertragung der Kennwerte beim Register-Lesen. Es werden 2 Kennwerte übertragen, wobei jeder aus einem "digitalen" und aus einem "analogen" Byte besteht.
- Struktur einer analogen Kennwertangabe. Sie besteht aus einem Schrittweiten-Code  $s$  (Bits A7...5) und einem Wert  $v$  (Bits A4...0), der die Anzahl der Schritte angibt.
- Codierung der Schrittweite (Step Size, im folgenden mit  $ss$  bezeichnet).

*Digitale Kennwert-Anteile*

Die digitalen Kennwertangaben sind ganze Binärzahlen zwischen 0 und 255. Sie kennzeichnen die jeweil erforderliche Anzahl an Taktflanken (Clock Ticks), die sich aus Struktur und Wirkungsweise der taktgesteuerten (synchronen) Hardware im Speicherschaltkreis ergibt. Die Werte sind demgemäß von der Taktfrequenz unabhängig.

*Analoge Kennwert-Anteile*

Die analogen Kennwertangaben entsprechen den Datenblattwerten der "eigentlichen" DRAM-Anordnung (vgl. Kapitel 5, Abschnitt 5.2.2.).

*Berechnung des Analogwertes  $a$ :*

$a = ss \cdot v$ , wobei sich  $ss$  gemäß Abbildung 8.56c aus  $s$  ergibt.

*Berechnung des Gesamtwertes (in Taktflanken (Clock Ticks)):*

- Analogwert  $a$  durch die aktuelle Taktzykluszeit teilen,
- das Ergebnis auf die nächste ganze Zahl aufrunden,
- diesen Wert zum jeweiligen digitalen Kennwert-Anteil addieren.

### 8.6.7. Zugriffsabläufe

Dem Prinzip nach laufen die Zugriffe ähnlich ab wie bei den Rambus-Speichertypen: Kommandocode senden, Adreßangaben senden, gemäß der jeweiligen Latenzzeit-Vorgaben warten, Daten entgegennehmen (Lesen) oder Daten senden (Schreiben). Zu den Besonderheiten siehe die Abbildungen 8.53 bis 8.55.

**Abbildung 8.62** Lesezugriffe (SLDRAM, Inc.). a) bei aktiver (geöffneter) Seite, b) bei inaktiver Seite. Zu 1, 2, 3 siehe den Punkt "Taktumschaltung" im Text

### *Erklärung:*

Alle zum Starten des Lesezugriffs benötigten Angaben sind im Anforderungspaket enthalten. Um die Daten zur Speichersteuerung zu liefern, aktiviert der Speicherschaltkreis den jeweiligen Daten-Takt (DCLK1 oder DCLK0).

### *Taktumschaltung*

Wird ein Daten-Takt gar nicht benötigt, so wird das betreffende differentielle Leitungspaar auch gar nicht angesteuert; die Leitungen sind hochohmig. Daß ein Daten-Takt genutzt werden soll, wird besonders angekündigt<sup>\*)</sup>, und zwar durch einen Vorspann (Preamble), der 2 ½ Taktperioden = Taktflanken (Clock Ticks; bezogen auf CCLK) dauert. Schema (Belegung der halben Taktperioden): 00010. Der Takt wird zunächst für die Dauer von 3 Clock Ticks auf Low gezogen (1). Die erste "richtige" Taktperiode (2 Clock Ticks, (2)) dient als Kennung dafür, daß der Takt nunmehr aktiv ist. Die Datenübertragung selbst beginnt dann mit der Low-High-Flanke des folgenden Taktimpulses (3)

\*) : beim Lesen muß dies der Speicherschaltkreis tun, beim Schreiben die Speichersteuerung (vgl. Abbildung 8.54).

### *Latenzzeiteinstellung*

Die Lese-Latenzzeiten können feinstufig eingestellt werden:

- Grobeinstellung: durch Laden der entsprechenden Steuerregister (Bank Read Delay, Page Read Delay). Auflösung: 1 Clock Tick = ½ Taktperiode.
- Feineinstellung: über einen Vorwärts/Rückwärts-Zähler (Read Data Vernier) im Speicherschaltkreis. Zählen (Increment/Decrement) über Ereignis-Kommandos. Auflösung: 1/32 Taktperiode.

### *Einstellung des Zeitversatzes zwischen Daten und Taktflanken (Data Offset)*

Der Zeitversatz kann über einen weiteren Vorwärts-Rückwärts-Zähler (Data Offset Vernier) mittels Ereignis-Kommandos eingestellt werden. Auflösung: 1/32 Taktperiode. Justierbereich: ± 1/4 Taktperiode. Anfangszustand: Data Offset Vernier hat Inhalt Null (kein Zeitversatz zwischen Taktflanken und Daten). Die Einstellung wirkt typischerweise auf alle Datenleitungen gemeinsam (Wort-Offset). Es ist aber vorgesehen, die Bitpositionen einzeln einstellen zu können (Bit Offset, vgl. die Belegung des Konfigurationsregisters gemäß Abbildung 8.51a).

**Abbildung 8.63** Schreibzugriffe (SLDRAM, Inc.). a) bei aktiver (geöffneter) Seite, b) bei inaktiver Seite

### *Latenzzeiteinstellung*

Die Schreib-Latenzzeiten können nur "grob" eingestellt werden (durch Laden der entsprechenden Steuerregister (Bank Read Delay, Page Read Delay)). Auflösung: 1 Clock Tick = ½ Taktperiode.

**Abbildung 8.64** Überlappte (interleaved) Zugriffe (SLDRAM, Inc.)

### *Erklärung:*

- 1) Senden eines ersten Lesekommandos,
- 2) ein zweites Lesekommando für dieselbe Bank folgt unmittelbar nach,
- 3) für einen anderen Schaltkreis wird ein weiteres Lesekommando gesendet,
- 4) die zum ersten Lesekommando gehörenden Lesedaten treffen ein,
- 5) die zum zweiten Lesekommando gehörenden Lesedaten schließen sich lückenlos an (da es sich um denselben Schaltkreis handelt und da derselbe Daten-Takt verwendet wird, ist kein Takt-Vorspann erforderlich),
  - a) während die Lesedaten geliefert werden, startet die Speichersteuerung weitere Kommandos,
  - b) die zum dritten Lesekommando gehörenden Lesedaten werden geliefert. Hierfür ist der andere Daten-Takt spezifiziert worden. Dessen Vorspann kann sich mit der vom anderen Takt gesteuerten Datenübertragung überlappen, so daß sich - aus Sicht der Speichersteuerung - die Lesedaten lückenlos anschließen.

- 7) die Schreibdaten zu dem unter a) gestarteten Schreibkommando (Write2) werden geliefert. Hier wird wiederum der andere Daten-Takt (gegenüber 6) verwendet, dessen Vorspann sich mit der laufenden Datenübertragung überlappt.
- b) die Speichersteuerung startet weitere Kommandos,
- 8) die zum Lesekommando Read3 gehörenden Lesedaten werden geliefert,
- 9) die Speichersteuerung liefert die zu den Schreibkommandos Write4, 5 gehörenden Schreibdaten.

### 8.6.8. Refresh

Es sind die Betriebsarten "Auto Refresh" (im Normalbetrieb) und "Self Refresh" (in den Stromspar-Zuständen) vorgesehen. Zum Auslösen des Auto Refresh genügt ein Ereignis-Kommando. Die Refresh-Adresse wird von einem Adreßzähler im Speicherschaltkreis geliefert. Ein Auto-Refresh-Ereignis darf nur dann ausgelöst werden, wenn alle Banks inaktiv sind. Es hängt von ID8 und von der SUB-ID-Angabe im Ereignis-Paket ab, ob der Refresh-Ablauf nur in einem SLD RAM oder in mehreren oder in allen gleichzeitig stattfindet (Multicast/Broadcast-Übertragung).

Zum Ein- und Ausschalten des Self Refresh gibt es besondere Ereignis-Kommandos. Die typische Folge: Self Refresh einschalten → in Stromspar-Zustand übergehen → Stromspar-Zustand verlassen → Self Refresh ausschalten (vgl. Abbildung 8.56).

### 8.6.9. Stromspar-Betriebsarten

Das Einleiten und Verlassen der Stromspar-Betriebsarten wird nicht über Kommandos, sondern über besondere Signalleitungen gesteuert (Abbildung 8.56).

#### **Abbildung 8.65** Stromspar-Betriebsartensteuerung (SLDRAM, Inc.)

*Erklärung:*

- a) Übergang in den Standby-Modus durch Deaktivieren von LISTEN,
- b) Übergang in den Powerdown-Modus durch Deaktivieren von LINKON. Hierzu muß sich der Schaltkreis bereits im Standby-Modus befinden (1). Wird eine bestimmte Zeitgrenze überschritten, so ist nach dem Verlassen des Shutdown-Modus eine erneute Synchronisation und Ausgangsspannungskalibrierung erforderlich (wie nach dem Rücksetzen).

Die Abbildung zeigt weiterhin das Ein- und Ausschalten des Self Refresh. Schraffierte Flächen kennzeichnen, welche Signale zu welchen Zeiten inaktiv (hochohmig) geschaltet werden dürfen.

### 8.6.10. Rücksetzen und Initialisierung

Ein Impuls auf der RESET-Leitung bewirkt, daß alle Speicherschaltkreise gleichzeitig zurückgesetzt werden (hinsichtlich der Feinheiten verweisen wir auf das Original-Datenmaterial). Dabei werden die ID- und SUB-ID-Register der Speicherschaltkreise auf FFH bzw. FH gestellt. Nach dem Hardware-Rücksetzen sind folgende Aktivitäten erforderlich:

- Synchronisation der Kommando- und Schreibsignalwege. Hierzu liefert die Speichersteuerung spezielle Signalmuster an die Speicherschaltkreise. Der Daisy-Chain-Signalweg SI → SO wird verwendet, um der Speichersteuerung mitzuteilen, daß die Synchronisation abgeschlossen ist. Die Speichersteuerung aktiviert SI des ersten SLD RAMs. Alle Speicherschaltkreise werten die zyklisch wiederholten Signalmuster aus. Jene SLD RAMs, die den Synchronisationsvorgang abgeschlossen haben (dies wird anhand des korrekten Empfangs der Bitmuster erkannt), leiten ihre SI-Belegung nach SO weiter. Die Speichersteuerung wertet den SO-Ausgang des letzten SLD RAMs aus. Wird dieser aktiv (High), so kann das Synchronisieren beendet werden. Die Speichersteuerung schaltet dann den SI-Eingang des ersten SLD RAMs wieder inaktiv.

- Zuordnung der Schaltkreisadressen (ID-Assignment). Die ID- und SUB-ID-Werte werden mit entsprechenden Register-Schreibkommandos eingetragen. Die Schaltkreisauswahl wird hierbei über den Daisy-Chain-Signalweg gesteuert. Ein Schaltkreis ist dann ausgewählt, wenn seine ID- und SUB-ID-Belegung = FFH/ FH und wenn sein SI-Eingang aktiv ist. Die Speichersteuerung aktiviert hierzu den SI-Eingang des ersten SLDRAMs. Damit trägt das erste Register-Schreibkommando zunächst dessen Adresse ein. Anschließend reicht der Schaltkreis seine (aktive) SI-Belegung über seinen SO-Ausgang zum SI-Eingang des folgenden SLDRAMs durch, so daß das nächste Register-Schreibkommando die Schaltkreisadresse in den 2. SLDRAM einträgt usw.
- Einstellung (Kalibrierung) der Treiberströme. Die Treiberströme werden in jedem Schaltkreis einzeln so eingestellt, daß sich die spezifizierten Low- und High-Pegel ergeben. Als Bezugssignale dienen hierbei die Daten-Taktsignale. Ablauf:
  - DCLK-Takte nach High treiben (Kommando 2BH). Der Controller bewertet den High-Pegel  $V_{OH}$  und sendet solange Einstellkommandos<sup>\*)</sup> (als Ereignis-Pakete), bis der spezifizierte Pegel erreicht ist.
  - DCLK-Takte nach Low treiben (Kommando 2AH). Der Controller bewertet den Low-Pegel  $V_{OL}$  und sendet solange Einstellkommandos<sup>\*)</sup> (als Ereignis-Pakete), bis der spezifizierte Pegel erreicht ist.
  - Die Daten-Takte werden wieder freigegeben (Kommando 2EH).

\*) : die Speicherschaltkreise haben für beide Pegel je einen Vorwärts-Rückwärts-Zähler, und es gibt entsprechende Einstellkommandos zum Auf- und Abwärtszählen (Increment/Decrement  $V_{OL}/V_{OH}$ ).
- Lese-Synchronisation und Einstellen der Latenzzeiten. Latenzzeiten werden zunächst grob voreingestellt. Zwecks Lese-Synchronisation und Feineinstellung der Lese-Latenzzeiten fordert die Speichersteuerung vom SLDRAM an, daß dieser ein besonderes Synchronisations-Bitmuster an die Speichersteuerung liefert (Kommando "Lesen zwecks Synchronisation"). Die Einstellwerte werden solange verändert, bis das Bitmuster einwandfrei empfangen werden kann. Sinngemäß (d. h. praktisch: durch Ausprobieren) können dann die anderen Einstellungen optimiert werden.

## 8.7. Alles braucht seine Zeit

64-MBit-DRAMs, SDRAMs, Rambus-DRAMs usw. standen schon vor Jahren in den Datenbüchern (als Preliminary oder Advance Information<sup>\*)</sup>) - aber erst jetzt werden diese Technologien nach und nach am Massen-Markt wirksam. Gerade auf dem Gebiet der Speicherschaltkreise sind die Entwicklungszeiten keineswegs kurz. Schaltkreistechnologien und Speicherarchitekturen sind nicht von heute auf morgen zusammengeschustert. Sowohl Entwicklung als auch Fertigungsvorbereitung kosten mehrstellige Millionen-Beträge. Geht es um eine Massenfertigung mit nennenswertem Weltmarktanteil, so kann dies Investitionen von mehreren Milliarden DM erfordern. Tabelle 8.31 veranschaulicht, mit welchen Zeiträumen zu rechnen ist. Die Konsequenzen für den Praktiker:

- wir haben durchaus genügend Zeit, uns sorgfältig einzuarbeiten,
  - die mühsam erarbeiteten Kenntnisse und Erfahrungen zu älteren (aus Sicht mancher Zeitschriften: total veralteten) Technologien, Schaltkreisen usw. veralten nicht so schnell, wie es den Anschein hat.
- \*) : und in dieser Zeit wurde tatsächlich intensiv weitergearbeitet: zwischen den Vorankündigungen und den tatsächlich lieferbaren Serienprodukten gibt es doch - besonders in den tieferen Einzelheiten - etliche Unterschiede.

Technologie	Ankündigung, erste Muster, erste (vorläufige) Datenblätter	Wirksamkeit am (Massen-)Markt
DRAM, 64 MBits	1992	1996/97
DRAM, 256 MBits	1994	1998/99
DRAM, 1 GBits	1996/97	am Markt ab 2000, am Massen-Markt ab 2005
DRAM, 4 GBits	1996/97	am Markt ab > 2000
Synchrone DRAMs	1992	1997/98
Rambus	1992	1995/96
Concurrent Rambus	1996	1998
Direct Rambus <sup>*)</sup>	1997	1999
SLDRAM <sup>*)</sup>	1997 (Interface-Entwicklung - von SCI an - ab 1988)	1999

\*) : siehe Hinweis im Text

**Tabelle 8.42** Entwicklungs- und Markteinführungszeiten (Übersicht). Grundlage: Datenmaterial und Pressemitteilungen verschiedener Hersteller

*Hinweis:*

Direct Rambus und SLDRAM stehen in direkter Konkurrenz zueinander. Die technischen Unterschiede sind eher gering; beide Architekturen weisen im Grunde gleiches Leistungsvermögen (Datenraten) und vergleichbare Funktionalität (Kommando-Steuerung usw.) auf. Die Fachwelt neigt zu der Auffassung, daß sich wohl nur ein System am Markt durchsetzen wird. Hinter Direct Rambus steht Intel. Andererseits hat man es in der Industrie nicht gern, von einem einzigen Lizenzgeber abhängig zu sein. Besonders vorsichtige Schaltkreishersteller orientieren sich allseitig ("man weiß nie, was kommt - und doppelt hält besser"). Sie sind deshalb in den Referenzlisten (der Lizenznehmer bzw. Entwicklungs-Partner) beider Systemanbieter (Rambus/Intel, SLDRAM Inc.) zu finden.

## 8.8. Online-Informationsquellen (Auswahl)

- <http://developer.intel.com> (Schaltkreissätze, Direct RDRAM) <sup>1)</sup>
- <http://www.chips.ibm.com>
- <http://www.corsairmicro.com>
- <http://www.cypress.com>
- <http://www.eia.org/jedec/download> (JEDEC-Standards)
- <http://www.fujitsumicro.com>
- <http://www.halsp.hitachi.com>
- <http://www.hea.com> (Hyundai)
- <http://www.idt.com>
- <http://www.micron.com> <sup>3)</sup>
- <http://www.mitsubishichips.com>
- <http://www.mosysinc.com>
- <http://www.nec.com> <sup>2)</sup>
- <http://www.okisemi.com>
- <http://www.opti.com> (Schaltkreissätze)
- <http://www.rambus.com> <sup>1), 5)</sup>
- <http://www.samsungsemi.com> <sup>4)</sup>
- <http://www.sldram.com> <sup>1), 5)</sup>

<http://www.smartm.com> (SMART Modular Technologies)

<http://www.toshiba.com>

<http://www.viatech.com> (Schaltkreissätze)

<http://www.vlsi.com> (Schaltkreissätze)

- 1) gute Quelle für Spezifikationen, Standards, Standard-Entwürfe usw.,
- 2) u. a. Datenmaterial zu Rambus-Schaltkreisen,
- 3) u. a. viele Applikationsschriften,
- 4) Site enthält u. a. Prinzipdarstellungen zur Arbeitsweise von SDRAMs und zur seriellen Konfigurationsspeicherung (Serial Presence Detect SPD),
- 5) die Sites enthalten neben technischen Beschreibungen, Datenblättern, Anwendungshinweisen usw. auch Links zu den einschlägigen Schaltkreisherstellern. (Konkrete Datenblätter sind gelegentlich aussagekräftiger als Standard-Entwürfe o. dergl.)

### **Der JEDEC-Standard JESD 21-C**

Quelle: <http://www.eia.org/jedec/download> .

Der Standard beschreibt Gehäuse- bzw. Bauformen, Anschlußbelegungen, Konfigurationsangaben (Presence Detect) usw. von Speicherschaltkreisen und Speichermoduln. Er ist recht umfangreich und liegt in Form mehrerer PDF-Dateien vor (*Praxistip*: zunächst das Inhaltsverzeichnis (Table of Contents) holen).

#### *Hinweise:*

1. Nicht alle Hersteller halten sich bei allen Schaltkreis- und Modultypen genau an den Standard (es ist also mit Abweichungen zu rechnen).
2. Der Standard spezifiziert oftmals viel mehr Varianten bzw. Auslegungsmöglichkeiten, als in der Praxis tatsächlich genutzt werden.
3. Die Anschluß- und Signalbezeichnungen weichen mitunter von jenen ab, die die Hersteller bevorzugen (zum Standard gehört aber ein Kapitel "Terms and Definitions", in dem die verwendeten Bezeichnungen erklärt werden).
4. Es gibt verschiedene Ausgaben (Releases). Es werden aber nicht alle im Internet vorgehalten (und nur zahlende JEDEC-Mitglieder werden in den Änderungsdienst einbezogen).