

Folie 1

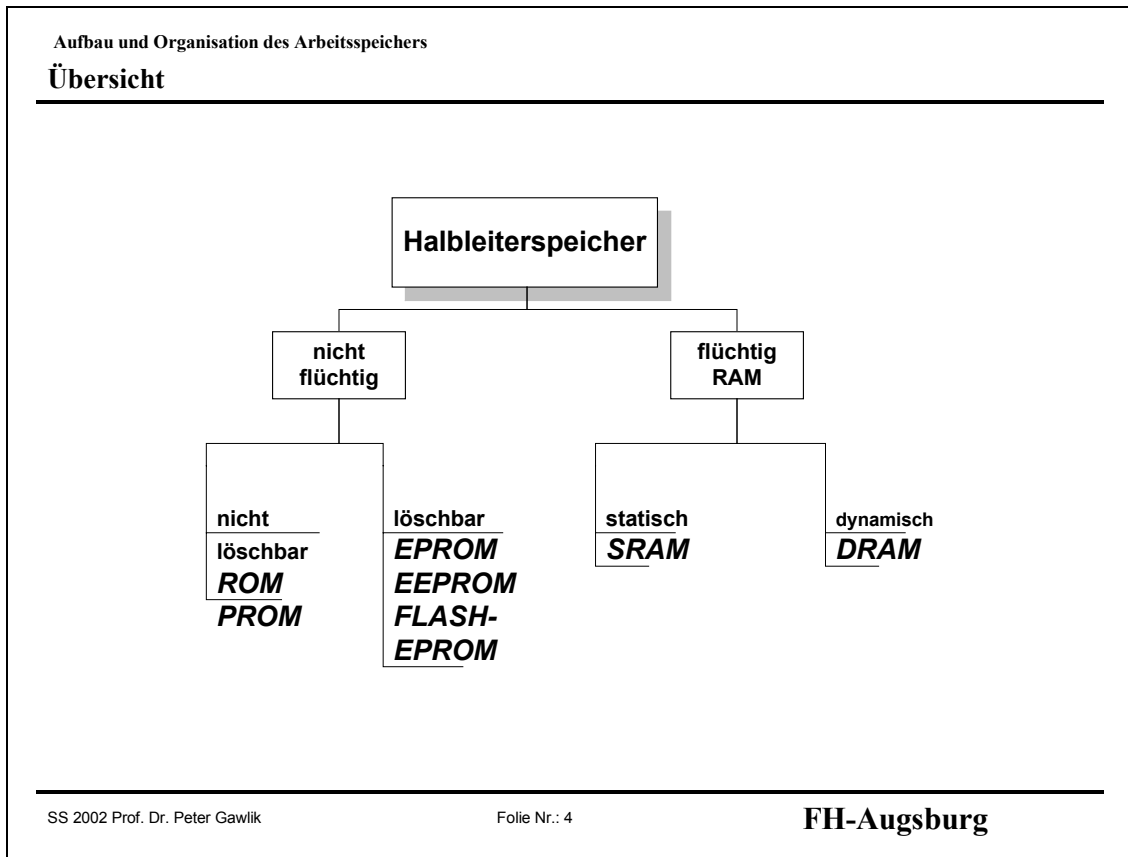
Aufbau und Organisation des Arbeitsspeichers	
Inhaltsverzeichnis 1	
<hr/>	
1.	Inhaltsverzeichnis 1-3
4.	Übersicht
5.	Prinzipieller Aufbau eines Speichers
6.	ROM
7.	ROM Aufbau
8.	ROM Ansicht eines 64MB-ROMs
9.	PROM
10.	EPROM 1
11.	EPROM 2
12.	EPROM 3
13.	EEPROM 1
<hr/>	
SS 2002 Prof. Dr. Peter Gawlik	Folie Nr.: 1
FH-Augsburg	

Folie 2

Aufbau und Organisation des Arbeitsspeichers	
Inhaltsverzeichnis 2	
<hr/>	
14.	EEPROM 2
15.	EEPROM 3
16.	EEPROM4
17.	Flash EPROM
18.	SRAM
19.	DRAM 1
20.	DRAM 2
21.	DRAM 3
22.	DRAM-Organisation eines 256x256x1Bit DRAM
23.	DRAM-Innerer Aufbau eines DRAM-Speichers 1
24.	DRAM-Innerer Aufbau eines DRAM-Speichers 2
<hr/>	
SS 2002 Prof. Dr. Peter Gawlik	Folie Nr.: 2
FH-Augsburg	

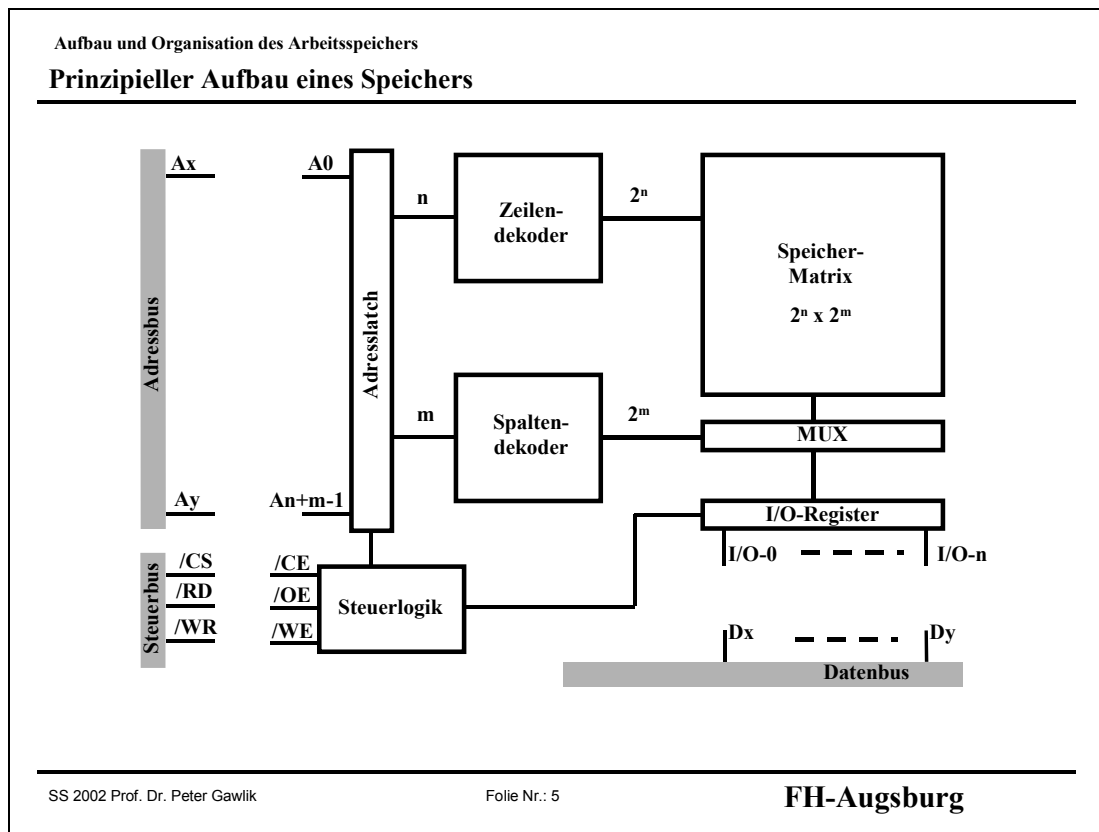
Inhaltsverzeichnis 3

- 25. DRAM-Innerer Aufbau eines DRAM-Speichers 3**
- 26. DRAM-Innerer Aufbau eines DRAM-Speichers 4**
- 27. DRAM-Verkürzung der Zugriffszeit beim Lesen Page-Modus**
- 28. DRAM-Zugriff im Burst-Page-Modus**
- 29. DRAM-Interleave-Modus**
- 30. SDRAM 1**
- 31. Refresh**
- 32. Refresh RAS Only**
- 33. Refresh CAS before RAS**
- 34. Übungen 1-5**



Dabei bedeuten die Abkürzungen:

ROM	Read Only Memory
RAM	Random Access Memory
PROM	Programmable ROM
EPROM	Erasable PROM
EEPROM	Electrical EPROM
FLASH-EPROM	FLASH (Blitz) EPROM mit spezieller Löschtechnik
SRAM	Static RAM
DRAM	Dynamic RAM



Speicher sind in der Regel matrixförmig organisiert. Die zur Verfügung stehenden Adressleitungen werden in Zeilen- und Spaltenadressen aufgeteilt. Mit einer bestimmten Adresse wird somit ein Kreuzungspunkt in der Speichermatrix adressiert. An den Kreuzungspunkten werden Speicherelemente (Bits) nach verschiedenen Technologien angeordnet.

Die Adressanschlüsse des Speichers werden mit dem Adressbus des μP s verbunden. Welche Adressleitungen des μP s an die Adressanschlüsse des Speichers angeschlossen werden, hängt u. a. von der Datenbreite des Speichers ab. Näheres dazu wird im Abschnitt „Anschluss von Speicher und Peripherie“ besprochen.

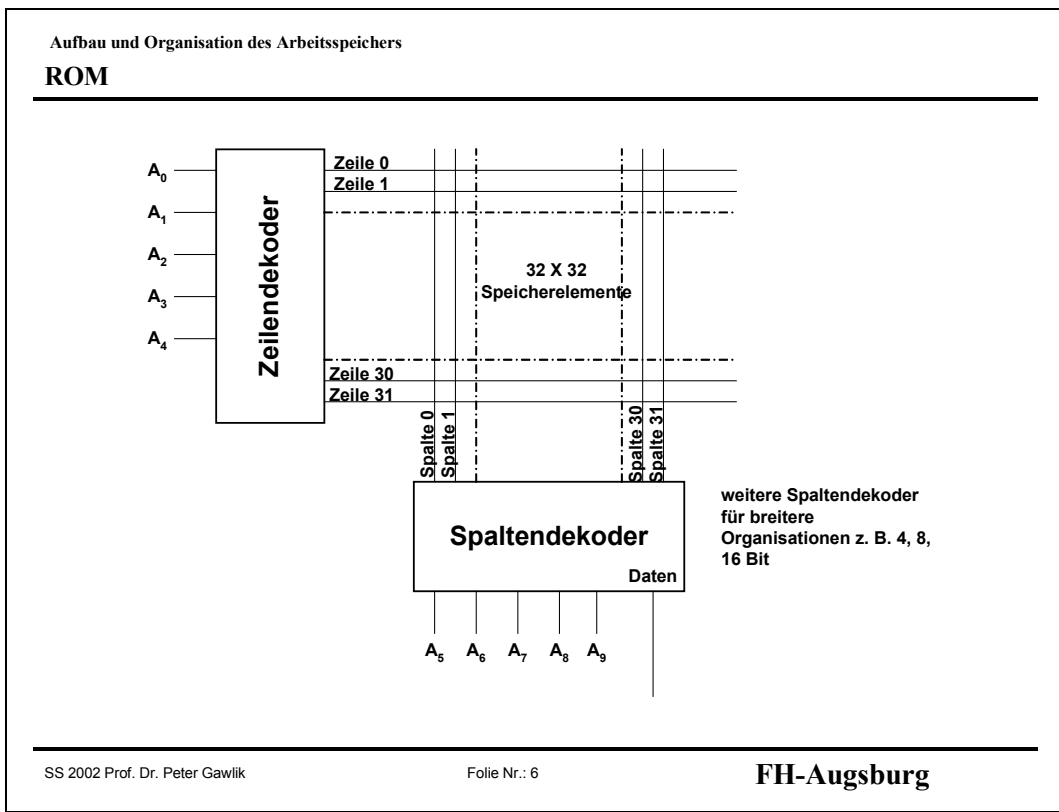
Die I/O-Anschlüsse des Speichers werden mit dem Datenbus des μP s verbunden. Welche Datenleitungen des μP s mit den I/O-Anschlüssen verbunden werden, hängt von der Datenbusstruktur des μP ab. Näheres dazu wird im Abschnitt Anschluss von Speicher und Peripherie besprochen.

Ein zeitgenaues Lesen oder Schreiben wird mit den Steuersignalen ermöglicht. Mit $/CE$ wird der Speicher aktiviert. Mit $/OE$ kann der Benutzer ein Datenwort, das im I/O-Register steht, auf den Datenbus schalten (Tri State) schalten. Mit $/WE$ wird Lesen oder Schreiben eingestellt.

Welche μP -Steuersignale an die Steuereingänge des Speichers angeschlossen werden, hängt stark von der speziellen Ausführung des μP s ab.

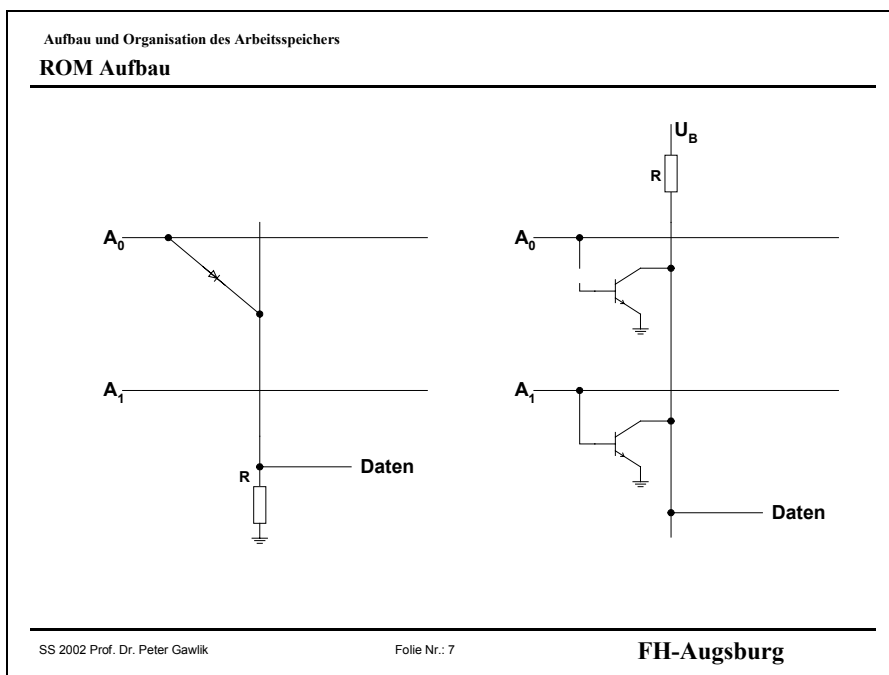
Z.B. liefert ein μP nicht immer ein $/CS$ -Signal zur Aktivierung eines Speichers. In diesen Fällen muss der Hersteller eines μP -Systems diese Signale mit externer Logik selbst entwerfen.

Folie 6



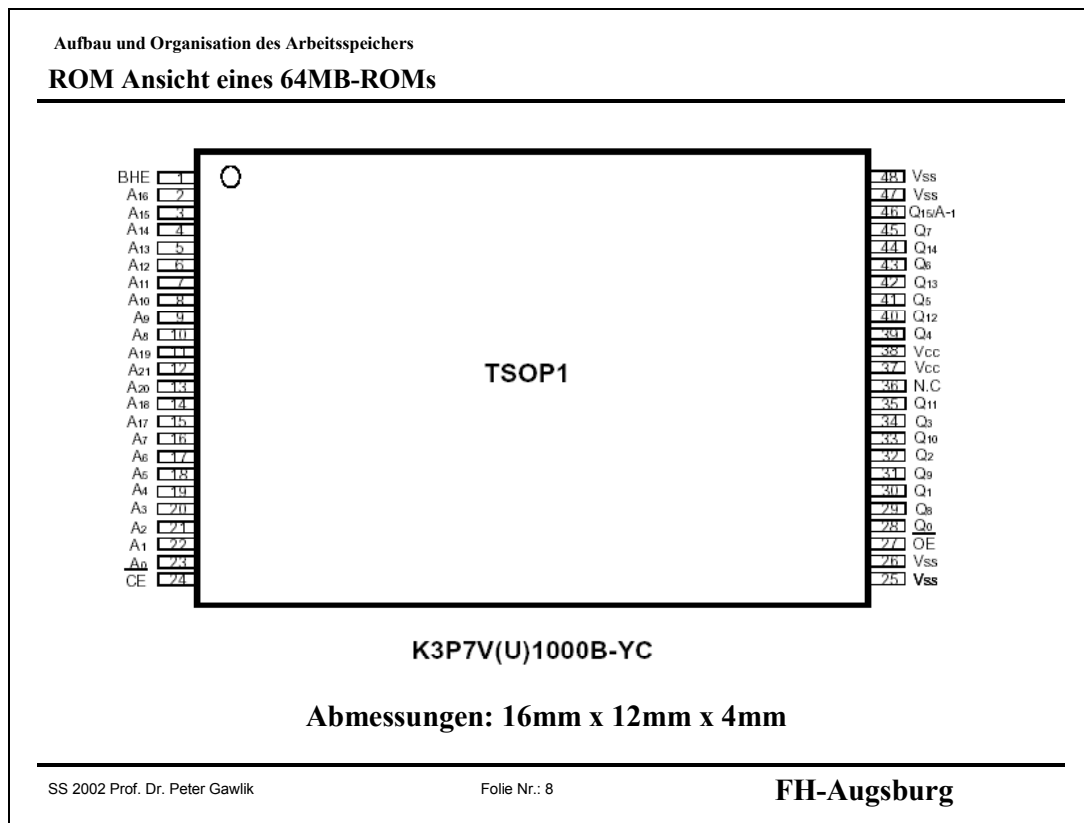
ROMs werden während des Herstellungsprozesses programmiert und sind deshalb nur bei großen Stückzahlen kostengünstig. Nachträgliche Änderungen sind nicht möglich. Einsatzgebiete für ROMs sind beispielsweise Programmspeicher, Datenspeicher und Charakter-Generatoren. Beim ROM werden feste Verbindungen als binäre Speicher verwendet. Diese Verbindungen können z. B. Dioden oder Transistoren sein.

Folie 7



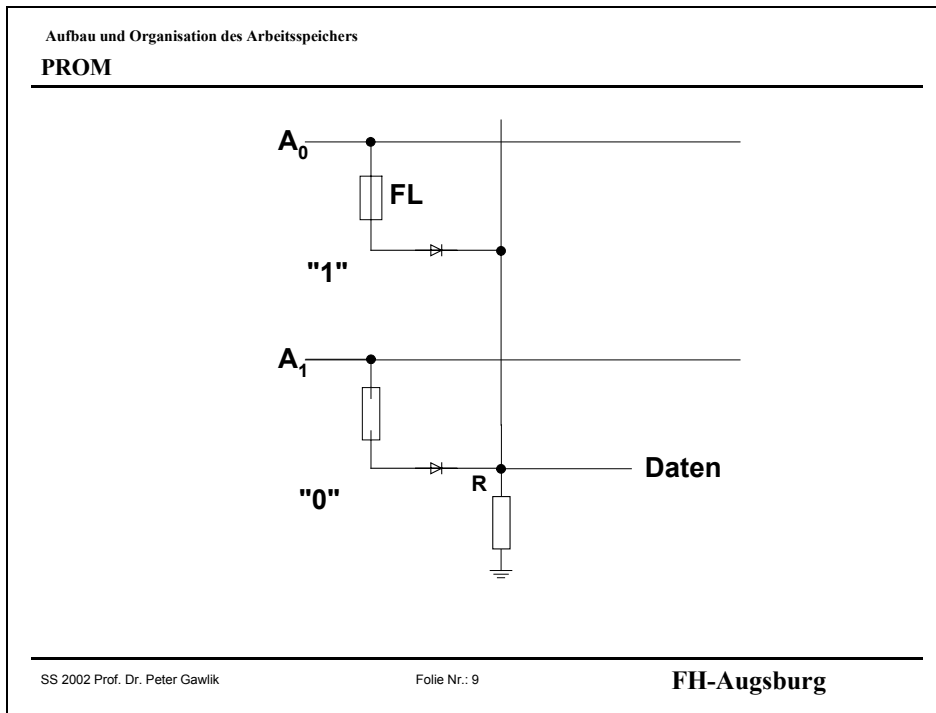
Wird eine Diode als Verbindungselement eingesetzt, fließt beim Aktivieren der Adressleitung A_0 ein Strom über die Diode. Dieser verursacht über den Widerstand R einen Spannungsabfall, der als logische „1“ an der Datenleitung abgegriffen werden kann. Bei der Herstellung wird nur an den zu programmierenden Kreuzungspunkten eine Diode diffundiert und sonst nicht. Wenn Transistoren als Koppellemente eingesetzt werden, befindet sich an jedem Kreuzungspunkt ein Transistor. Die Basis wird im letzten Metallisierungsschritt mit der Zeilenleitung verbunden, wenn dieser Kreuzungspunkt programmiert werden soll. In diesem Fall wird eine logische „0“ programmiert. Das Aktivieren der Adressleitung A_1 schaltet den Transistor ein, so dass dadurch das Potential der Spaltenleitung auf Massepotential liegt. Die typischen Eigenschaften eines ROMs sind anhand des Beispiels KM23c8105b, ein 8M-Bit ROM von SAMSUNG, im folgenden aufgelistet.

Organisation: Byte-Mode ; Word-Mode
 Random Access Time: 100ns
 Page Access Time: 50ns
 Versorgungsspannung: 5V
 Stromaufnahme in Betrieb: 80mA
 im Standby: 50 μ A
 Eingänge/Ausgänge TTL kompatibel; Tristate Ausgänge
 Polaritäten der Enable-Leitungen: programmierbar
 Folie 8



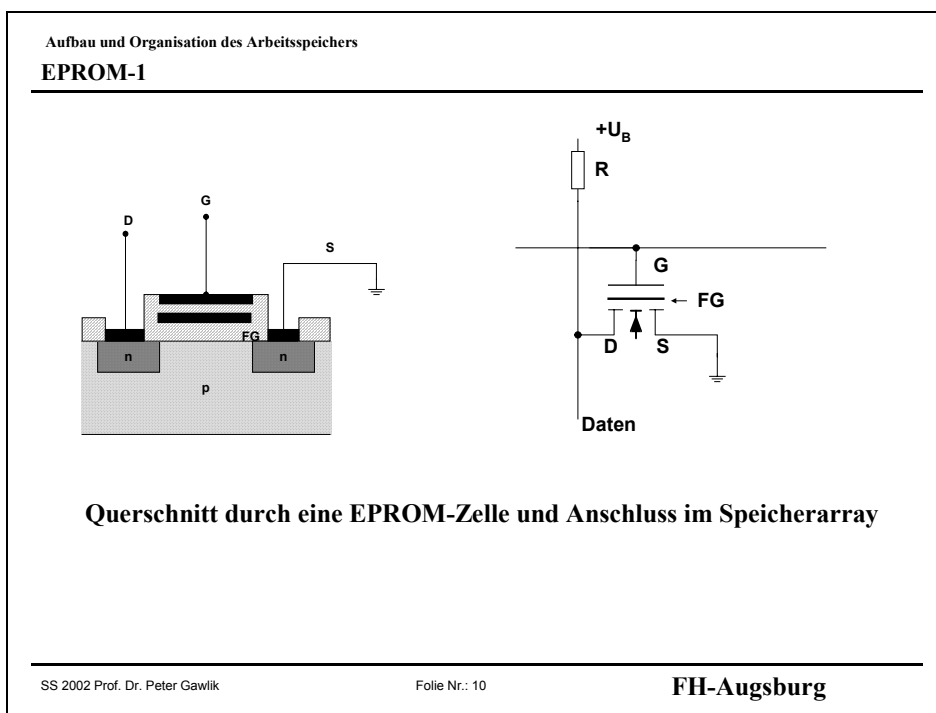
TSOP1 ist die Bezeichnung für einen bestimmten, normierten Gehäusetyp. K3P7V(U)1000B-YC ist die Herstellerbezeichnung.

Folie 9



Um die Festwertspeicher einen größeren Anwenderkreis zugänglich zu machen, wäre es vorteilhaft, wenn der Anwender den Speicher programmieren könnte. Dies gelingt indem man das Verbindungselement mit einer Verbindung versieht, die vom Anwender unterbrochen werden kann. Vom Hersteller wird im Normalfall eine „1“ programmiert. Die Verbindung der Diode zur Zeilenleitung wird sehr dünn ausgeführt, so dass sie wie eine Sicherung wirkt. Der Anwender kann durch das „Durchbrennen“ dieser Sicherung mit einem geeigneten Stromstoß eine „0“ programmieren. Als weitere Varianten des PROMs gibt es eine Transistorvariante und eine Variante bei der die Sicherung eine Diode darstellt. Genaueres ist z.B. /Bähring Mikrorechner-Systeme/ zu entnehmen.

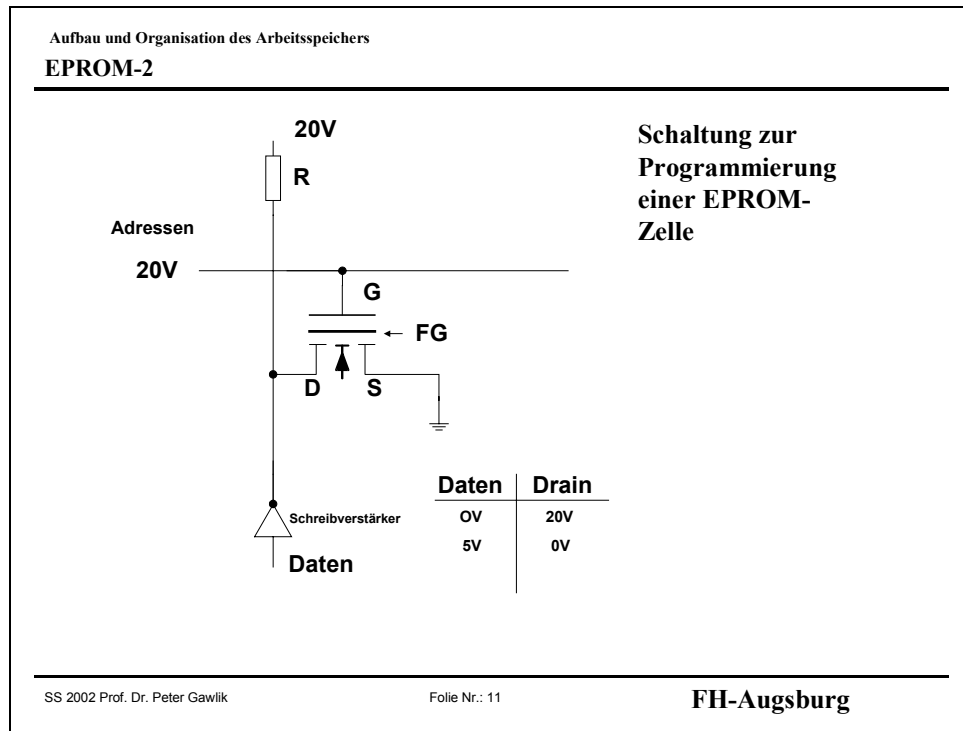
Folie 10



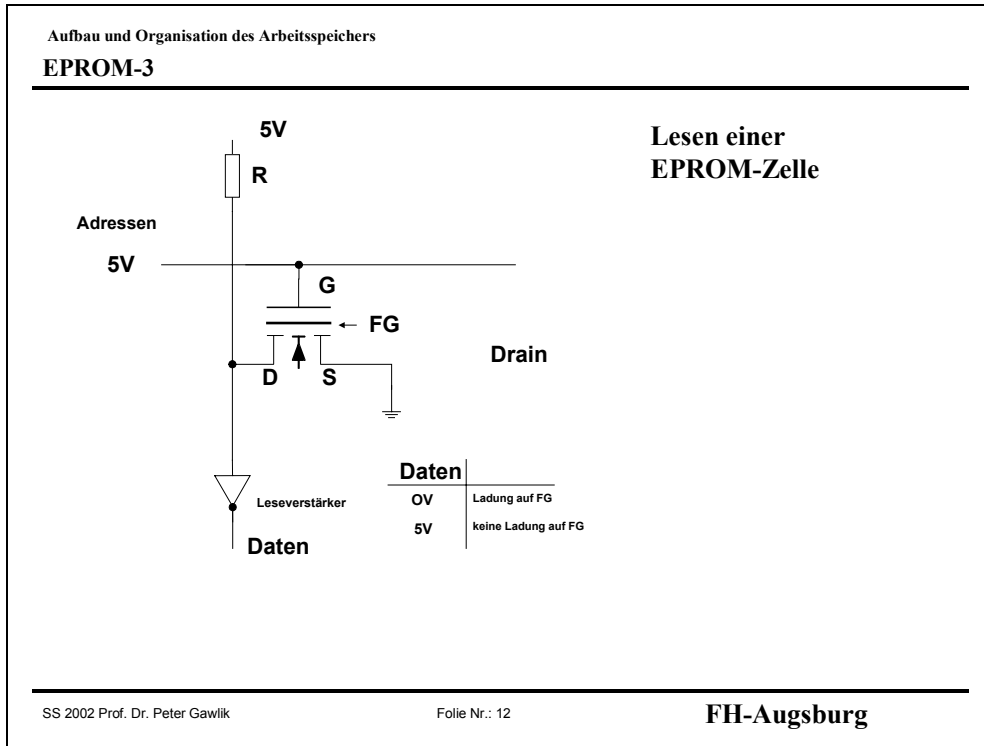
Oft tritt der Fall auf, dass der Inhalt des Speichers geändert werden muss. Änderungswünsche können z.B. durch geänderte Anforderungen oder durch Entwicklungsfehler entstehen. In solchen Fällen sind Speicher ideal, die man bei Bedarf löschen und wieder neu programmieren kann.

Ein zweites Gate liegt zwischen Steuergate und Substrat. Dieses Gate (Floating Gate - FG) ist vollständig isoliert. Mit Hilfe geeigneter Spannungen können Elektronen auf das FG gelangen.

Folie 11



Zum Programmieren muss ein verhältnismäßig großer Kanalstrom I_D fließen. Mit einer hohen Feldstärke zwischen Gate und Source werden Elektronen, die eine ausreichend hohe Energie besitzen, durch das dünne SiO_2 (Dicke ca. 30 nm) auf das **FG** transportiert. Diese sammeln sich dort an und verschieben mit ihrer Ladung die Schwellspannung des Transistors auf ca. 6V. Mit der üblichen Betriebsspannung von 5 V am Gate wird der Transistor nicht mehr leitend. Jede Zelle muss zur Programmierung adressiert werden. Ist das EPROM breiter organisiert 4-Bit, 8-Bit usw. können diese Zellen gleichzeitig programmiert werden. Ein EPROM wird mit einem Programmiergerät programmiert. Die benötigte Programmierspannung wird heute auf dem Chip erzeugt, so dass 5V Betriebsspannung ausreichen. Die Isolation des **FG** ist so gut, dass die Programmierung für mehr als 10 Jahre erhalten bleibt.



An Drain und Gate werden 5V (Betriebsspannung) angelegt. Ist die Zelle programmiert, wird der Transistor nicht leitend. Das Drainpotential bleibt auf 5V und das invertierte Datum „0“ wird gelesen. Ist die Zelle nicht programmiert, wird der Transistor leitend, so dass das Drainpotential auf 0V liegt. Über den invertierenden Leseverstärker wird ein High (5V) gelesen.

Löschen: Eine EPROM-Zelle wird durch energiereiches UV-Licht gelöscht. Hierzu hat der EPROM-Speicher ein lichtdurchlässiges Fenster im Gehäuse. Durch die Photonen erhalten die FG-Elektronen soviel Energie, dass sie durch den Isolator zum Gate bzw. zum Substrat gelangen und dorthin abfließen können. Durch den Photonenbeschuss werden allerdings auch Fehlstellen im Kristallgitter erzeugt, die dazu führen, dass die Schwellspannung sich mit jeder Programmierung ein wenig verschiebt. Aus diesem Grund ist die Anzahl der Lösch- und Programmierzyklen auf einige Hundert beschränkt.

Allg. Daten

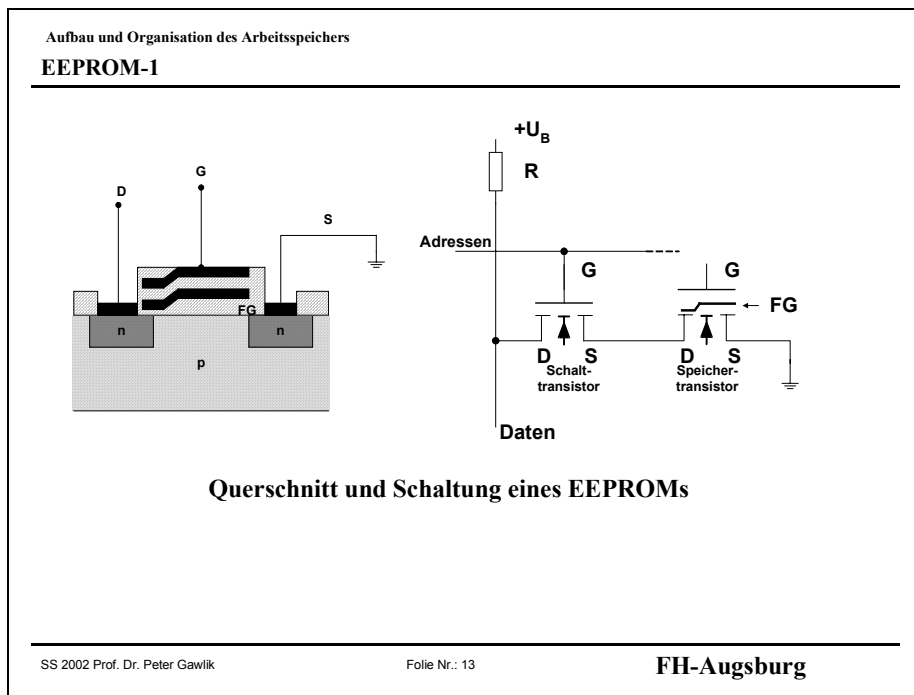
Organisation:	Byte
Access Time:	35-200ns
Versorgungsspannung:	5V
Leistungsbedarf aktiv:	200 mW
Leistungsbedarf Power down:	50 μ W

Programmierung

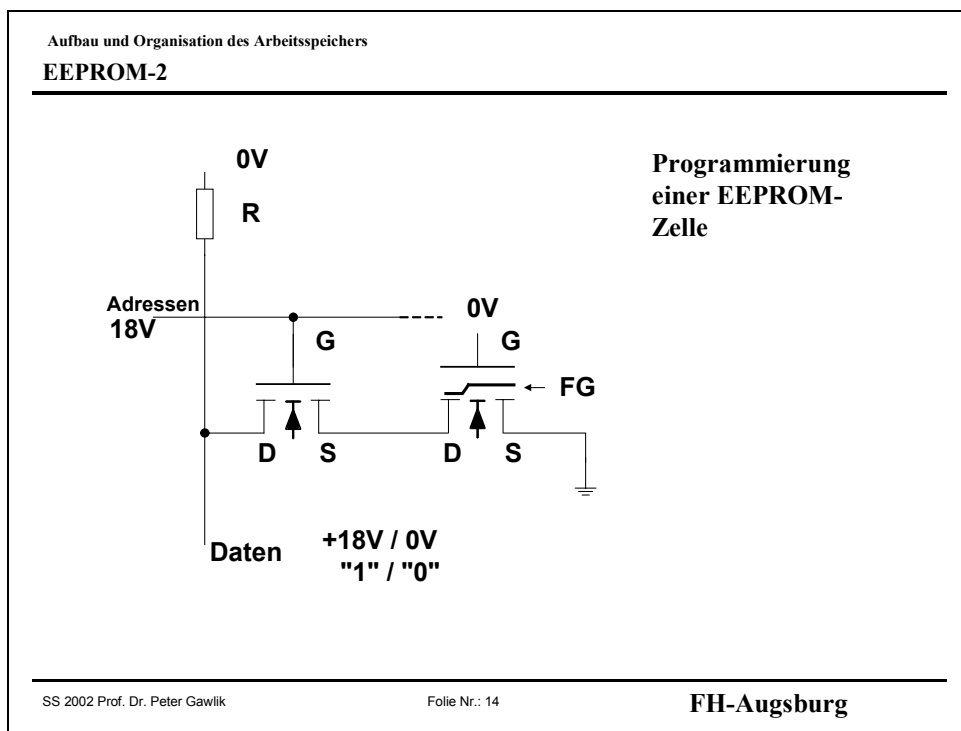
Spannung:	12,5 V
Auflösung:	Byte
Zeit:	< 100 μ s

Löschen:

Spannung:	12,5 V
Auflösung:	Chip
Zeit:	15 min



Electrical EPROMs sind, wie der Name sagt elektrisch löscher. Sie brauchen nicht zum Löschen aus der Schaltung genommen werden.
 Der Aufbau einer EEPROM-Zelle ähnelt dem EPROM-Aufbau. Das FG ist beim EEPROM bis über das n - Gebiet des Drain geführt. Der Abstand zwischen FG und Drain ist nur einige Nanometer groß. Dadurch tritt ein Effekt auf, der die elektrische Programmierung ermöglicht. Zur Selektion der Speicherzelle muss der Speicherzelle ein Schalttransistor vorgeschaltet werden.



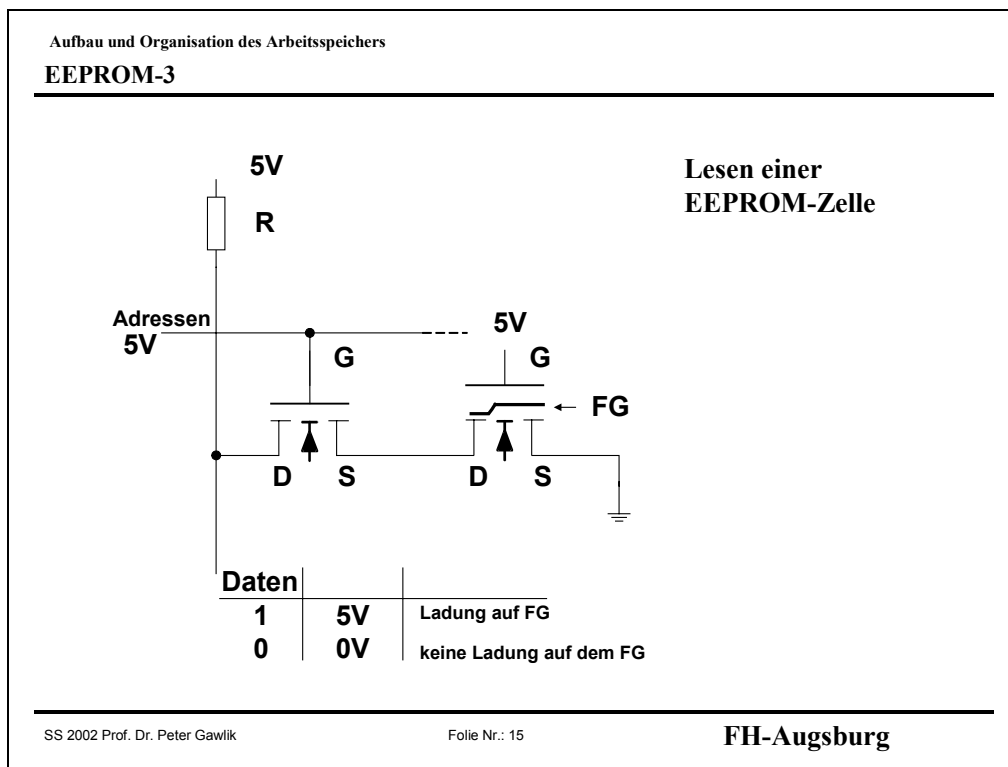
Der Grundzustand zum Programmieren ist eine geladenes **FG**. D. h. der Speichertransistor sperrt (Es ist eine „0“ programmiert). Wie die Spannungen zur Programmierung angelegt sein müssen, zeigt die Folie.

Mit 18V auf der Adressleitung wird der Schalttransistor leitend. Das Gate des Speichertransistors wird auf 0V gelegt.

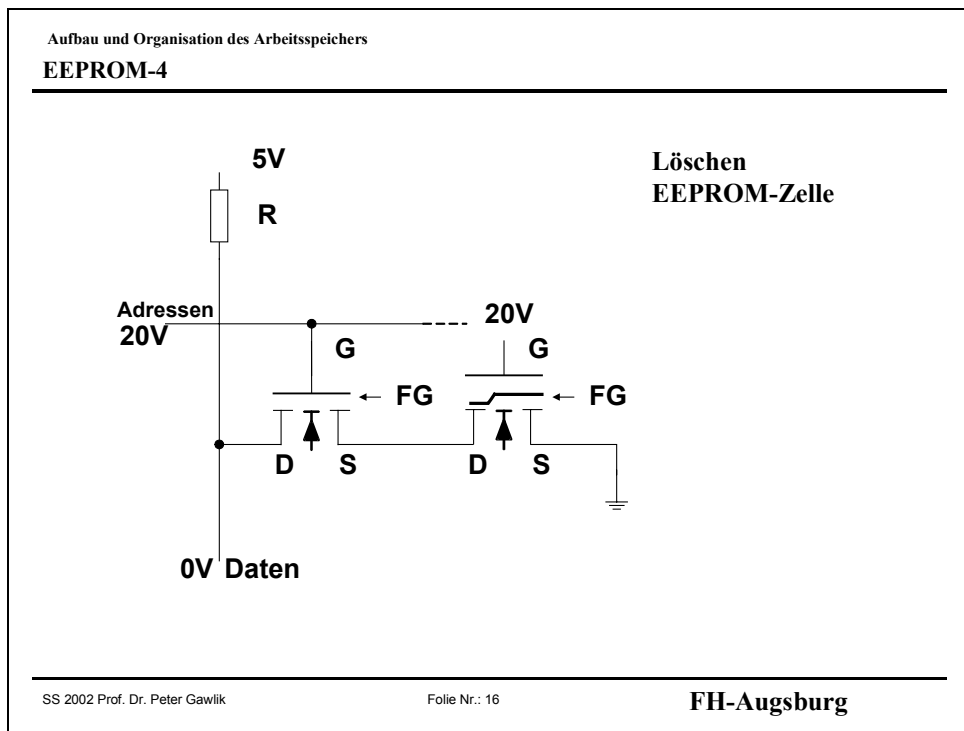
Soll eine „0“ programmiert werden, wird 0V an die Datenleitung angelegt. Das Potential gelangt zum Drain des Speichertransistors. Durch diese Spannungsverhältnisse wird die Ladung des **FG** nicht verändert.

Legt man hingegen 18V an die Datenleitung, entsteht ein großes Feld zwischen Gate und Drain des Speichertransistors. Infolge des kleinen Abstandes zwischen Gate und Drain kommt es zum sogenannten Tunneleffekt d.h. die Elektronen durchtunneln die dünne SiO₂-Isolation und fließen über den Drainbereich ab. Es wird eine „1“ programmiert. Die Schwellspannung des Speichertransistors wird negativ, d.h. er wird ein selbstleitender Transistor. Aus diesem Grund muss dem Speichertransistor eine Schalttransistor vorgeschaltet werden.

Folie 15



Zum Lesen wird die Zeilenleitung und das Gate des Speichertransistors auf 5V gelegt. Hierdurch wird der Schalttransistor leitend. Der Speichertransistor wird nur leitend, wenn sich auf dem FG keine Ladungen befinden. In diesem Fall wird die Datenleitung auf Massepotential „0V“ gezogen. Befindet sich Ladung auf dem FG schaltet dieser nicht, somit bleibt das Potential an der Datenleitung auf 5V.



Löschen bedeutet beim EEPROM das FG mit Elektronen zu laden, so dass der Speichertransistor sperrt. Zum Löschen muss ein großes elektrisches Feld zwischen Gate und Drain gelegt werden. Die Folie zeigt wie in diesem Fall die Spannungen angelegt werden müssen. Der Schalttransistor wird leitend und legt das Drain des Speichertransistors auf Masse-potential. Durch das hohe Feld und die sehr dünne Isolationsschicht können Elektronen die Isolation durchwandern (Tunneleffekt) und gelangen auf das FG. Dadurch vergrößert sich die Schwellspannung des Speichertransistors. Der Transistor sperrt im Normalbetrieb (5V am Gate). Im folgenden sind einige Daten von heute verfügbaren EEPROM aufgelistet:

Allg. Daten

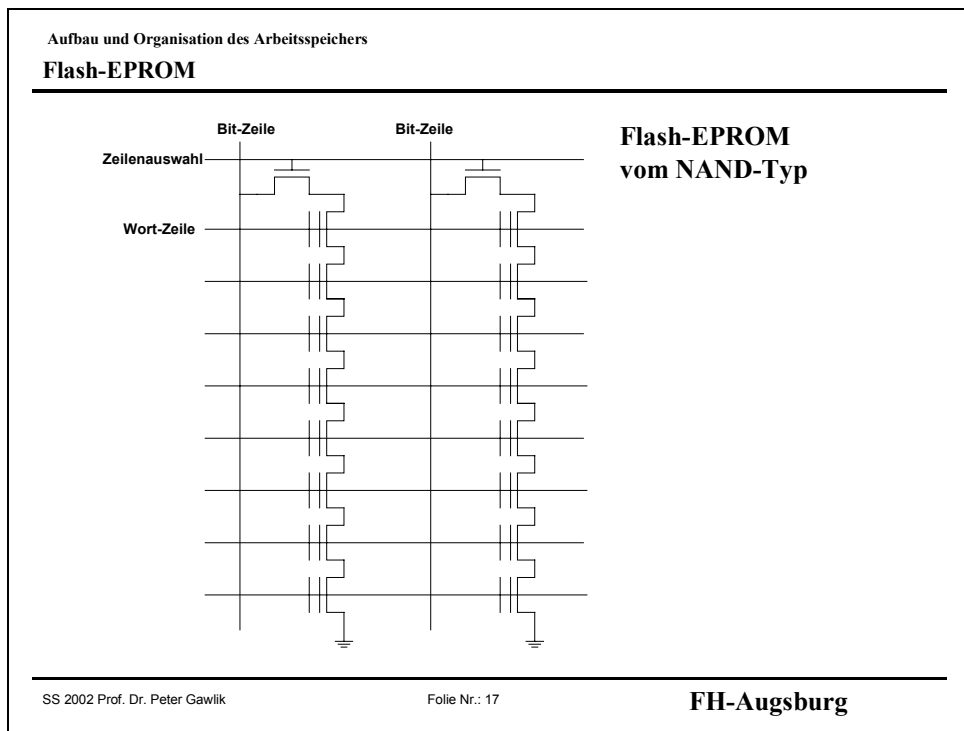
Organisation:	Byte
Access Time:	70-200ns
Versorgungsspannung:	5V
Leistungsbedarf aktiv:	200 mW
Leistungsbedarf power down:	50 μ W

Programmierung

Spannung:	12,5 V intern erzeugt
Auflösung:	Byte
Zeit:	< 5 ms

Löschen:

Spannung:	20 V intern erzeugt
Auflösung:	Byte
Zeit:	5 ms
Anzahl der Programmier - Löschzyklen:	1000 - 10 000



Flash-EPROM sind spezielle Formen der EEPROMs. Bei ihnen ist das Löschen des ganzen Chips oder Teile davon möglich. Dies ist besonders für Anwendungen interessant, bei denen z. B. im Boot-EPROM häufig bestimmte Parameter geändert werden.

Flash-EPROM Speicherzellen sind in NAND-Form organisiert, wie die Folie zeigt. Der Speicher wird dadurch programmiert, dass an der Wortzeile 22 V angelegt wird und die Bitzeile der zu programmierenden Zelle auf Massepotential liegt. Diese Spannung reicht zum programmieren aus. Alle anderen Zellen der Spalte müssen eingeschaltet werden, d.h. das Gate muss auf 10 V gelegt (über die Schwellspannung eines programmierten Transistors) werden. Beim Lesevorgang wird die Lesespannung an die Wort-Zeile gelegt. Alle anderen Transistoren der Bit-Zeile werden eingeschaltet (10V). Somit rückt das Massepotential an die Source des zu lesenden Transistors. Das Drain ist dann mit dem Selektionstransistor verbunden. Flash-EPROMs werden heute bis zu einer Größe von ca. 64 MB hergestellt. Die Zugriffsform ist bei einigen Typen seriell organisiert, was den Vorteil bietet, dass Updates zu größeren Einheiten leichter möglich sind, weil die Bauelementanschlüsse gleich bleiben können. Typische Daten sind im folgenden aufgelistet.

Allg. Daten

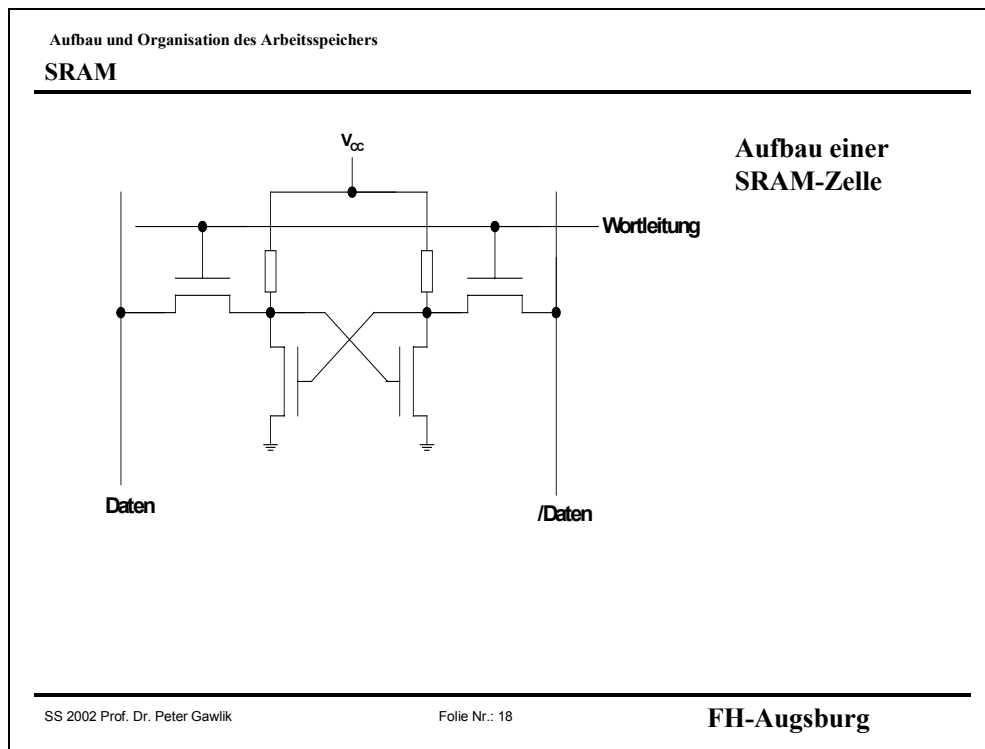
Organisation: Seiten, Blöcke
 Access Time: 45-200ns
 Versorgungsspannung: 5V

Programmierung

Spannung: 22 V intern erzeugt
 Auflösung: Byte
 Zeit: < 10 μ s

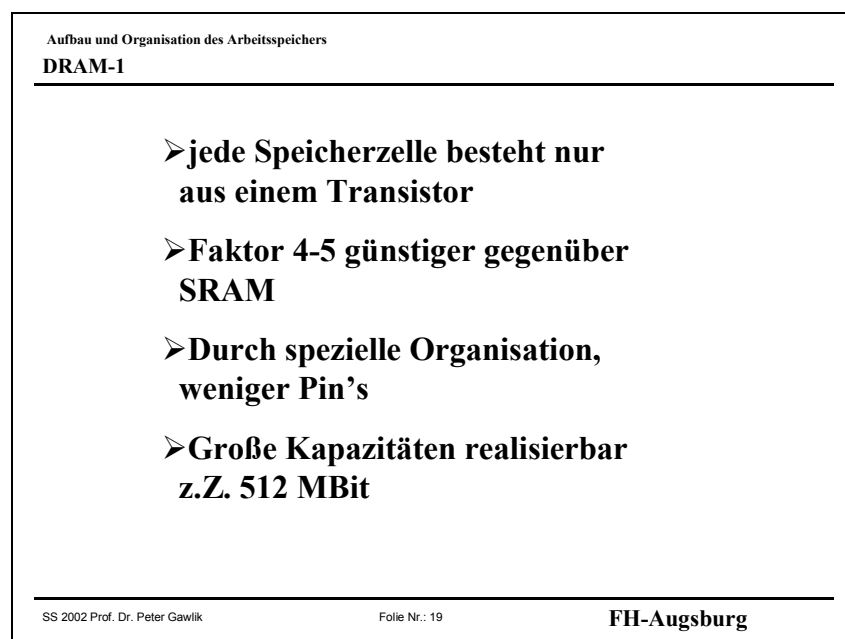
Löschen:

Auflösung: Chip, Block, Blöcke
 Zeit: 1 s
 Anzahl der Programmier - Löschzyklen: > 100 000



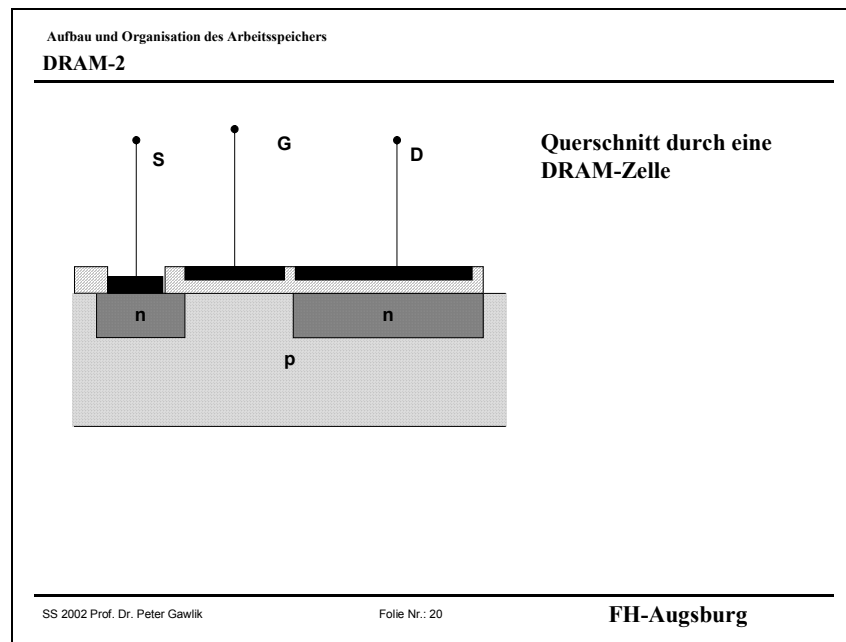
Es gibt auch Versionen bei denen die Widerstände als Transistoren ausgeführt sind. Auch gibt es Realisierungen in C-MOS-Technik. Die Funktionsweise wird als bekannt vorausgesetzt.
Typische Daten sind:

Organisation: 1,4,8,32 Bit
Kapazität: bis 32 MB
Access Time: 2-50ns
Versorgungsspannung: 5V, 3.3V
Leistungsaufnahme: 200 mW; 30 mW (μ W)



Der Einsatz von DRAM's ist allerdings nicht ganz unproblematisch. So sind z.B. Sondersignale, und ein spezielles Timing erforderlich. Außerdem müssen Maßnahmen getroffen werden, damit der Datendurchsatz im μP trotz der relativ langsamen DRAM's erhalten bleibt.

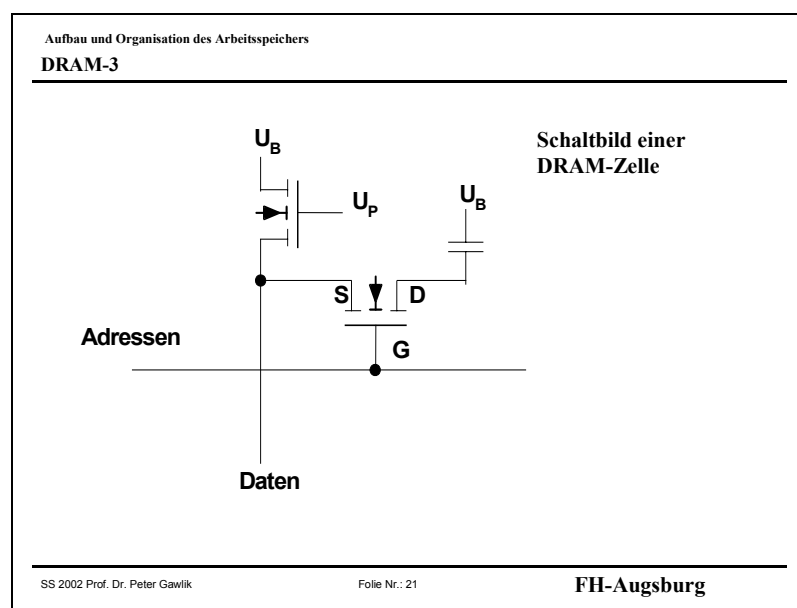
Folie 20



Die Information wird beim DRAM in einem kleinen Kondensator gespeichert, der durch das vergrößerte Draingebiet gebildet wird. Die Kapazität beträgt ca. 0,1 pF - 0,5 pF.

Die Ladung bleibt infolge von Leckströmen auf dem Speicher nur für ca. 4-64 ms. Es müssen also Maßnahmen getroffen werden, um die Information auf Dauer zu speichern

Folie 21

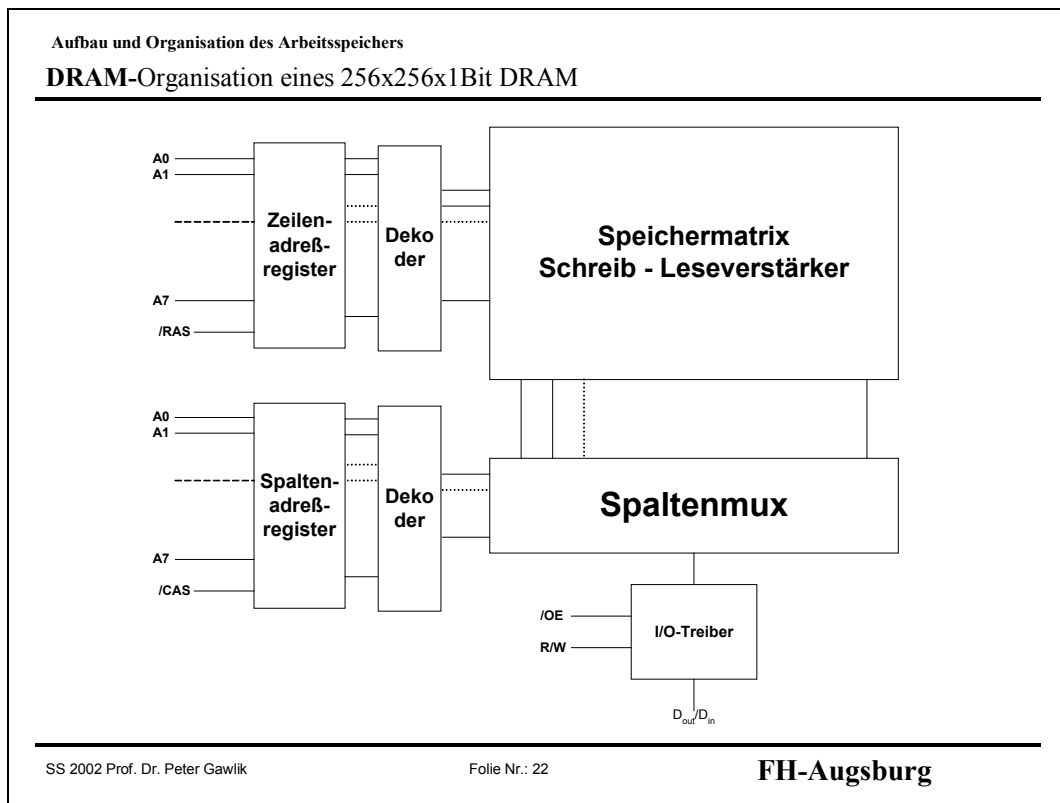


Lesen einer DRAM-Zelle: Die Ladungsmengen auf dem Speicherkondensator und der Bitleitung liegen in der gleichen Größenordnung. Zur Feststellung ob sich Ladung auf dem Speicherkondensator befindet oder nicht,

wird die Bitleitung zunächst an $+U_B$ gelegt. Wird nun das Gate an positives Potential gelegt, fließt ein Ausgleichsstrom dessen Größe vom Ladungszustand des Speicherkondensators abhängt. Mit einem speziellen Leseverstärker wird dieser Strom gemessen und in eine 0/1 Information umgesetzt. Durch das Lesen wird die DRAM-Zelle entladen. Die Information muss also zurückgeschrieben werden.

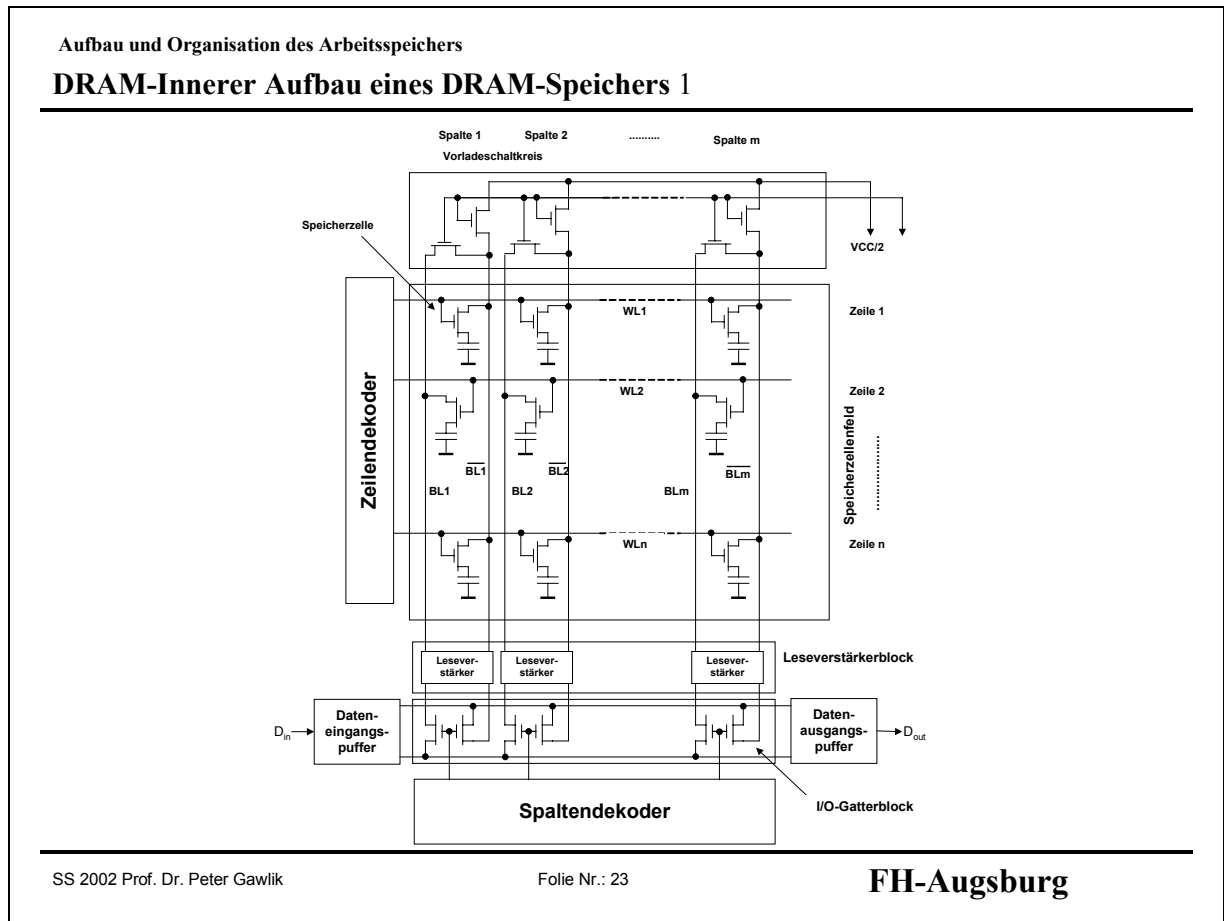
Schreiben einer DRAM-Zelle: Zunächst wird eine positive Spannung U_{GS} zwischen Gate und Source angelegt, wodurch der Transistor leitend wird. Legt man nun die Bitleitung auf Massepotential, wandern Elektronen zum n-Gebiet des Drains, denen positive Ladungen gegenüberstehen. Der Speicherkondensator ist geladen. Wird die Bitleitung auf positives Potential gelegt, werden Elektronen abgesaugt. Der Kondensator wird entladen (entspricht dem Lesevorgang).

Folie 22



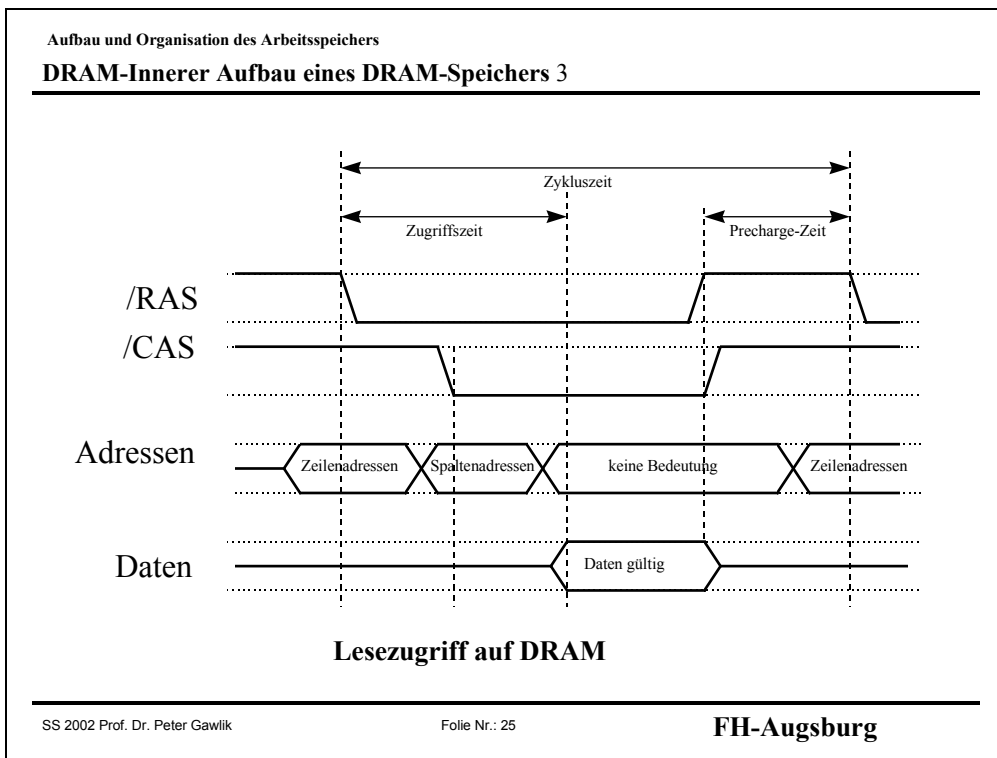
Die Zeilen und Spalten werden im Multiplexbetrieb übergeben. Erst die Zeilenadresse, dann die Spaltenadresse. Steuerung durch:

/RASRow-Address-Signal
/CASColumn-Adress-Signal



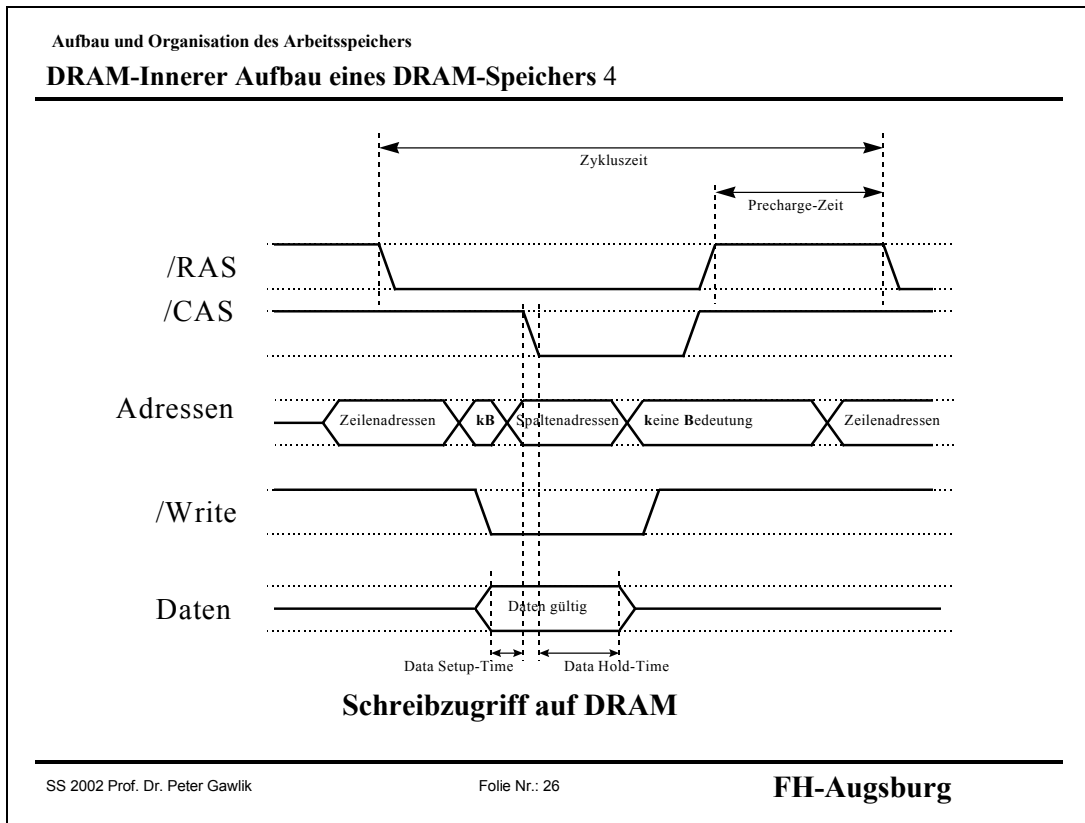
Den internen Ablauf beim Zugriff auf ein DRAM-Speicher soll im folgenden erläutert werden.

1. Bitleitungen vorladen
2. Zeilendekoder aktiviert Wortleitung
3. Alle Transistoren auf der Wortleitung schalten durch
4. Ladungen fließen auf Bitleitung
- 4 MB à 2048 Bitltg.-Paare, 2048 Spalten
5. Spannung ändert sich um ca. 100 mV Kondensator leer → Potential fällt
Kondensator voll → Potential steigt
6. Leseverstärker wird aktiviert Potentiale werden verstärkt
Bitleitungen werden auf Masse (Fall fällt)
auf Vcc (Fall steigt) gezogen.
7. Leseverstärker (2048) geben Ausgangssignal ab
8. Spaltendekoder aktiviert entsprechendes Gatter
9. Ausgangspuffer verstärkt Signal
10. Da Zeile weiter aktiviert → Daten werden zurückgeschrieben
11. Leseverstärker werden deaktiviert
12. Vorladeschaltkreis wird aktiviert
13. Bereit für eine neue Zeile



Eine Zeile wird mit der Zeilenadresse angewählt. Alle Spalten werden gelesen und gespeichert. Mit Spaltenadresse wird eine Zelle ausgewählt. Eine bestimmte Zeit nach der negativen Flanke von /CAS erscheint das Datum am Ausgang. Die Zugriffszeit wird von der Low-Flanke /RAS bis zum stabilen Anstehen des Datums am Ausgang gemessen.

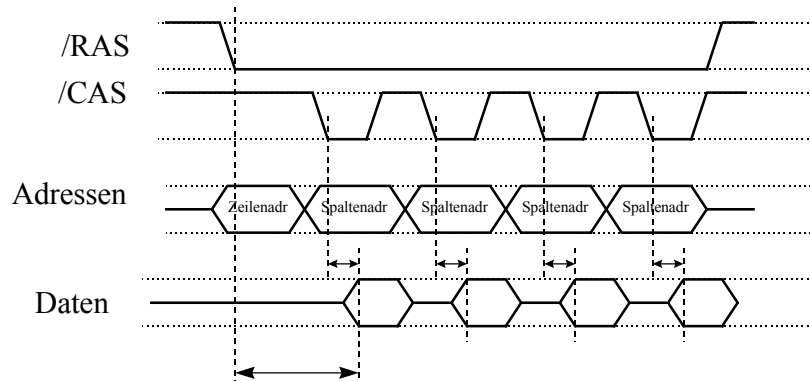
Solange /CAS auf Low gehalten wird, liegt das Datum am Ausgang an. Nach der High-Flanke von /CAS wird der Ausgang kurze Zeit später hochohmig. Ein neuer Lesezugriff kann aber erst eine bestimmte Zeit (Precharge) später erfolgen, weil die Daten zurückgeschrieben werden müssen und die Bitleitungen neu geladen werden müssen. Die Zykluszeit wird zwischen zwei Low-Flanken von /RAS gemessen. Die kürzeste mögliche Zugriffszeit ist aus der Summe von Precharge und Zugriffszeit zu bestimmen.



Der Schreibzugriff wird dem DRAM mit $\text{/WRITE}=\text{Low}$ signalisiert. Beim schreibenden Zugriff muss das Datum mit Low-Flanke /CAS stabil anstehen. Hierzu ist die Data-Setup-Time einzuhalten. Das Datum wird mit dieser Flanke in den Zwischenspeicher übernommen. Nach Low-Flanke von /CAS muss das Datum noch für die „Data-Hold-Time“ stabil anstehen. Auch hier ist für einen erneuten Zugriff auf die Zelle die Precharge-Zeit zu beachten.

Typische Zugriffszeiten liegen heute zwischen **50** und **120 ns**. Die Precharge-Zeit beträgt ca. 40 bis 90 ns. Zwei aufeinander folgende Zugriffe müssen also **90** bis **210 ns** auseinander liegen.

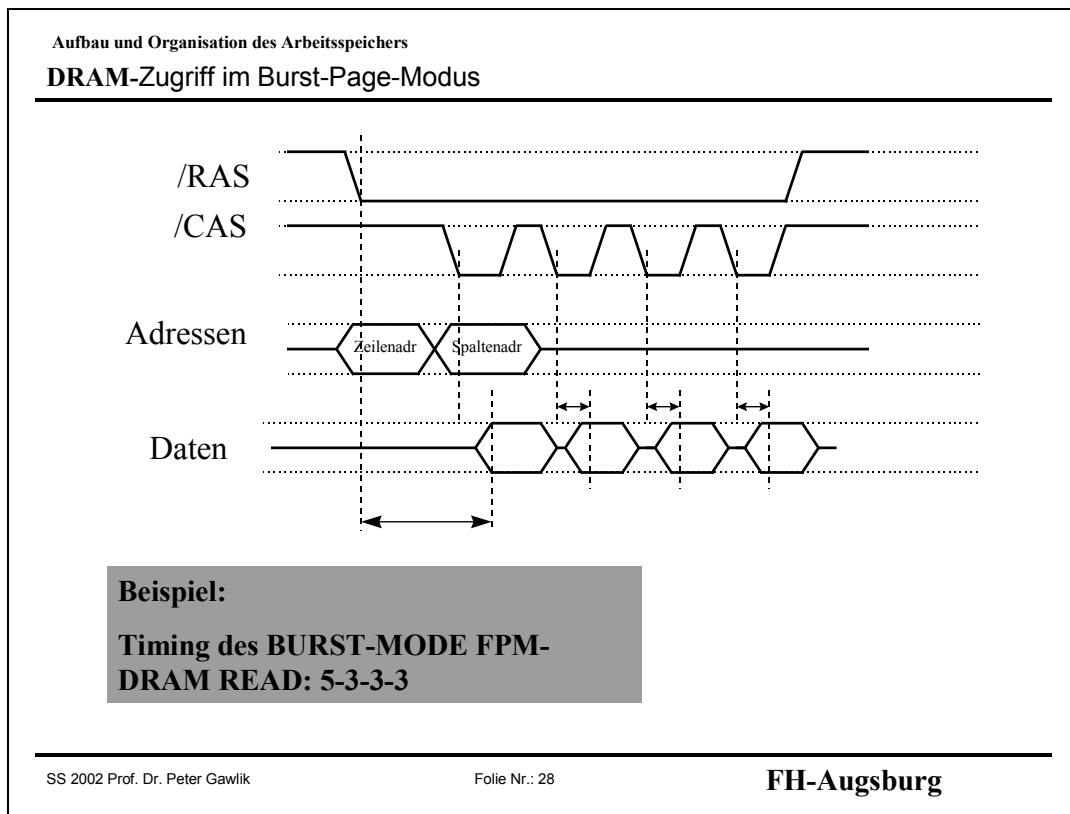
Aufbau und Organisation des Arbeitsspeichers

DRAM-Verkürzung der Zugriffszeit beim Lesen Page-Modus**Voraussetzung**

- **Daten müssen nahe beieinander liegen**
- **DRAM muss entsprechend organisiert sein**

Beim Page-Modus (Seitenzugriff) bleibt eine Zeile adressiert. Nur beim ersten Zugriff auf die Zeile wird die volle Zugriffszeit benötigt. Da bei den Folgezugriffen innerhalb der Zeile nur noch die Spaltenadressen angelegt werden müssen, erfolgen diese Zugriffe schneller.

Wie in der Folie angedeutet, erfolgt der Zugriff innerhalb einer Seite (Page) ca. 2-3 mal schneller. Muss auf eine andere Zeile zugegriffen werden, wird der Page-Modus abgebrochen. Beim Refresh (darauf gehen wir später ein) wird ebenfalls der Page-Modus verlassen.



Beim Burst-Page-Modus (Nibble-Modus) werden 4 aufeinander folgende Bits gelesen. Adressiert wird nur das erste Bit. Die folgenden Bits werden vom Baustein selbständig adressiert. Die Adressierung erfolgt durch einen internen 2 Bit-Zähler, der durch /CAS erhöht wird. Die Bits werden dabei immer in der Sequenz: **00 01 10 11 00 01** ausgegeben, wobei der Startpunkt beliebig ist. Die Zugriffe laufen ca. 4 mal schneller als Normalzugriffe ab. Heutige Prozessoren unterstützen diesen Modus direkt. D.h. sie besitzen Speicherzugriffszyklen, bei denen auf 4 aufeinander folgenden Adressen zugegriffen wird. Die Adressierung der einzelnen Spalten einer Zeile erfolgt außerdem unter Verwendung einer Pipeline. Das bedeutet, dass die Ausgabe der Daten mit der Adressierung der nächsten Speicherzelle zeitlich überlagert wird. Dadurch ergibt sich eine weitere Steigerung der Zugriffszeit im HyperPage-Mode (HPM).

Beispiel:

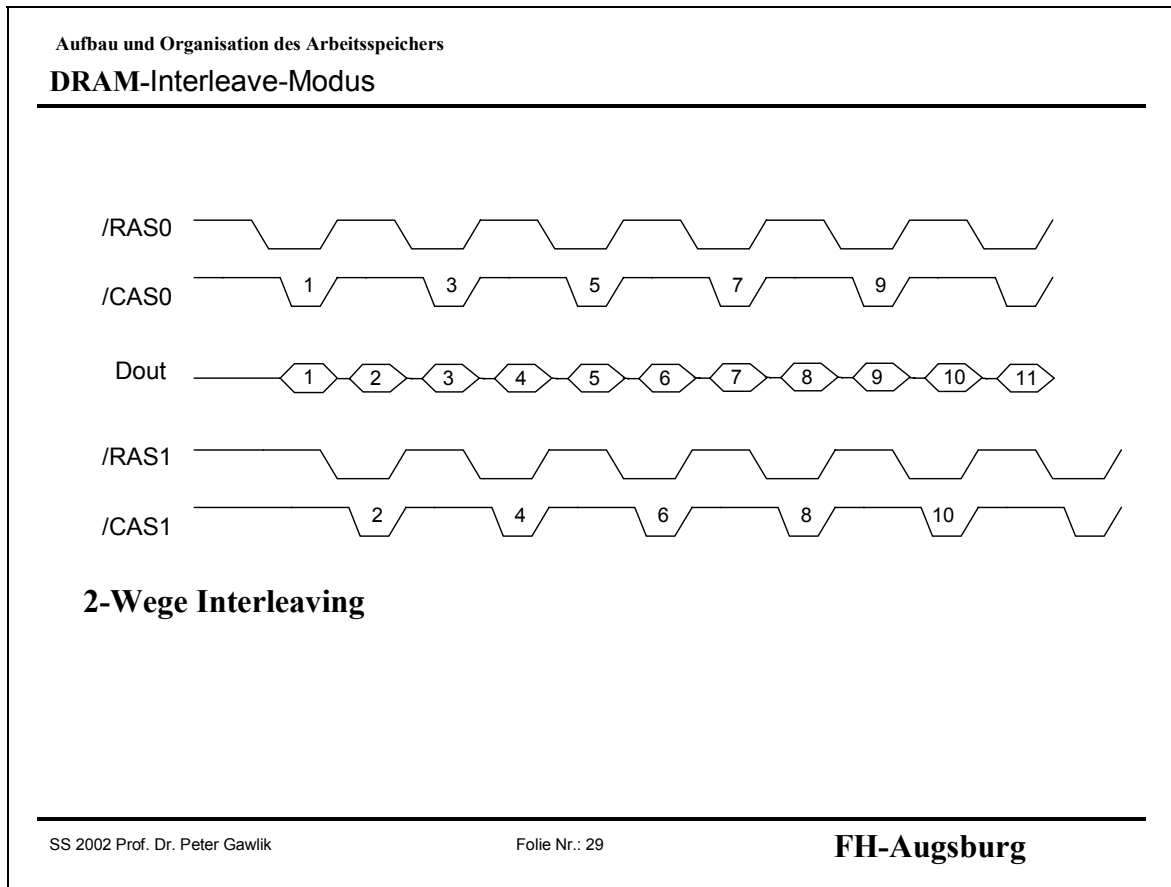
Timing des BURST-MODE FPM-DRAM READ: 5-3-3-3

Prozessor: Pentium, 66 MHz externe Taktfrequenz (1 Taktzyklus ist etwa 15 ns lang)

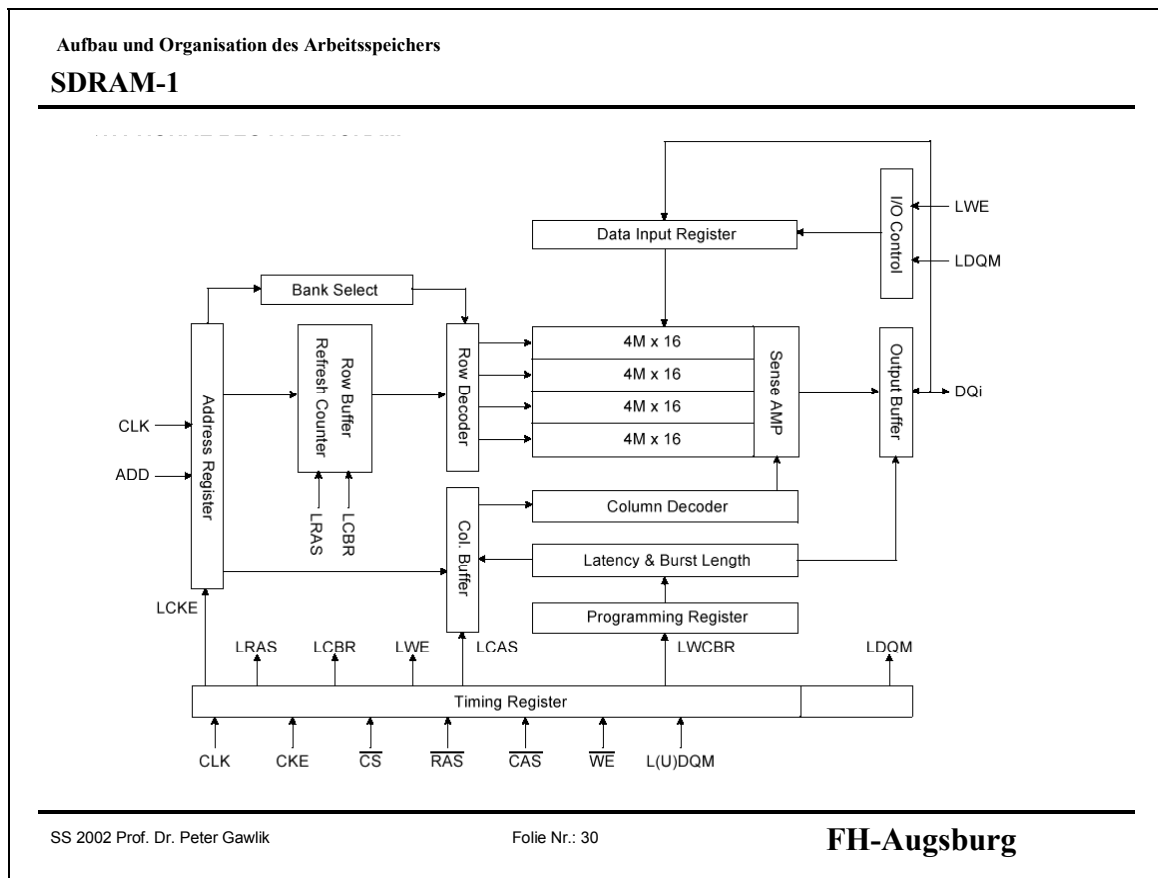
DRAM: FPM-DRAM mit 60 ns Zugriffszeit

Der erste Zugriff auf die Zeile erfordert 5 Taktzyklen (die adressierte Zeile muss erst durch Einschreiben der Zeilenadresse (RAS geht auf LOW) aktiviert werden)

Die folgenden Zugriffe im FPM erfordern nur 3 Taktzyklen (2 Waitstates für die Lesezyklen)



Eine weitere Möglichkeit die Zykluszeit beim Seitenwechsel zu umgehen, ist das Interleaving. Hierzu wird der Speicher in sogenannte Banks aufgeteilt, die dann separat mit zeitlich versetzten RAS- und CAS-Signalen angesprochen werden. Am Beispiel eines 2-Wege-Interleaving soll das Prinzip verdeutlicht werden. Bei einem 32-Bit Speicher bilden die geraden Doppelwort-Adressen die erste Bank und die ungeraden Doppelwort-Adressen die zweite. Bei einem sequentiellen Zugriff (z.B. Cache-Line auffüllen) werden die beiden Banks nacheinander angesprochen, so dass sich, wie oben gezeigt, effektiv eine Verdoppelung der Zugriffsrates ergibt. Während auf Bank 1 zugegriffen wird, wird Bank 2 vorgeladen. Für die Zugriffsrates ist also nicht die Zykluszeit, sondern nur die Zugriffszeit maßgebend.



Synchrone DRAMs (Synchronous DRAMs) sind eine neue Entwicklung und haben sich inzwischen als Standardspeicher durchgesetzt. Die oben beschriebenen DRAM-Typen arbeiten asynchron zum CPU-Takt. Es ist daher ein Handshaking zwischen CPU und Memory-Controller notwendig, um Daten zu übertragen. Beim Lesevorgang wird die Adresse dem Memory-Controller durch das CPU-Signal ADS (Address Strobe) übergeben. Die CPU liest die Daten vom Speicher, wenn der Memory-Controller das BRDY-Signal aktiviert (active LOW). Dazwischen führt die CPU Wartezyklen aus.

Die oben beschriebenen BEDO-DRAMs können zwar im Burst-Mode mit jedem Taktzyklus (ohne Waitstates) Daten an die CPU übergeben, das ist allerdings nur möglich, solange die externe Taktfrequenz nicht höher als 66 MHz ist. Heute sind externe CPU-Taktfrequenzen von 100-133 MHz und mehr üblich. Heute angebotene SDRAMs arbeiten meist bis zu 133 MHz und können im Burst-Mode Daten mit jedem Takt liefern.

SDRAMs basieren, wie alle hier beschriebenen DRAM-Typen, auf der DRAM-Speicherzelle mit einem CMOS-FET-Transistor und einem Kondensator. Auch bei SDRAMs sind diese Speicherzellen in einer Matrix (Zeilen und Spalten) angeordnet. SDRAMs haben jedoch zwei Speicherbänke (zwei Speichermatrizen), je nach Bitbreite auch noch mit mehreren Ebenen. Die Verwendung von zwei Speicherbänken erlaubt das Vorladen der jeweils anderen Bank, während auf die eine Bank zugegriffen wird. Damit werden die Precharge-Zeiten versteckt. Der Zugriff auf die Daten erfolgt also meist abwechselnd durch Zugriff auf die verschiedenen Bänke (INTERLEAVE-Verfahren). Durch verschiedene Kombinationen der Eingangssignale (RAS, CAS, Chip Select, Write Enable, Bank Select) werden die verschiedenen Befehle (ACTIVATE = Aktivieren einer Zeile, READ, WRITE, REFRESH) an ein SDRAM synchron zum externen Takt übergeben und intern gespeichert.

SDRAMs haben intern eine Pipeline-Architektur. Sie erlaubt die überlappende Ausführung von aufeinander folgenden Lese- bzw. Schreibzyklen. So kann also beispielsweise ein zweiter Lesebefehl bereits einen Takt nach dem vorherigen an den SDRAM-Chip gesendet werden. Synchrone DRAMs funktionieren aber prinzipiell wie herkömmliche DRAMs - das Vorladen und Aktivieren einer Zeile (Row-Address mit RAS übergeben) ist vor dem Auslesen von Daten erforderlich und benötigt ebensoviel Zeit wie bei diesen Typen. Erst im Burst-Mode spielt der SDRAM seine Vorteile aus, denn die Burst-Adressen werden wie beim BEDO-DRAM intern erzeugt und durch alternierenden Zugriff auf die beiden Speicherbänke liefert das SDRAM die Daten im 8-12 ns-

Abstand (je nach SDRAM-Typ). SDRAMs werden nach dieser Burstmode-Zugriffszeit bezeichnet. Ein SDRAM, das im Burstmode Daten mit 10 ns Abstand liefert, wird demnach als 10ns- bzw. 100 MHz-Typ gekennzeichnet.

Synchrone DRAM-Bausteine müssen konfiguriert werden. Sie enthalten ein Mode-Register, in das Informationen zum Burstmode (Länge, Typ) und zur CAS-Latency (wie viele Takte vergehen zwischen der Übergabe der Spaltenadresse und dem Gültigwerden der Daten).

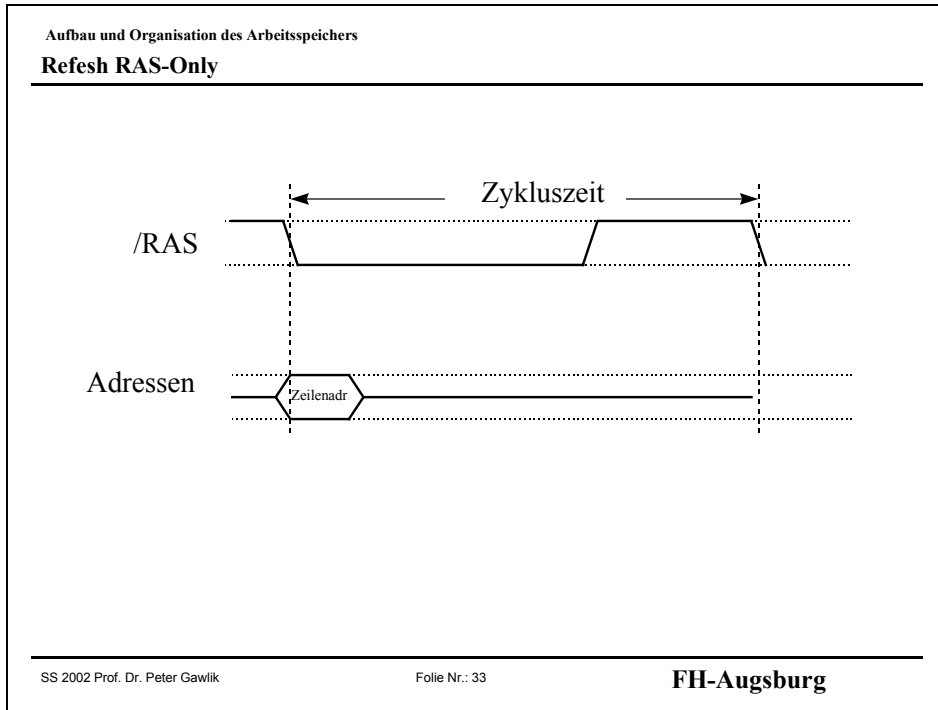
Timing des SDRAM Burstmode-READ: 5-1-1-1

Dieses Timing - und das ist der wesentliche Unterschied zum BEDO-DRAM - ist bis zu externen Taktfrequenzen von 100 MHz möglich.

Folie 32

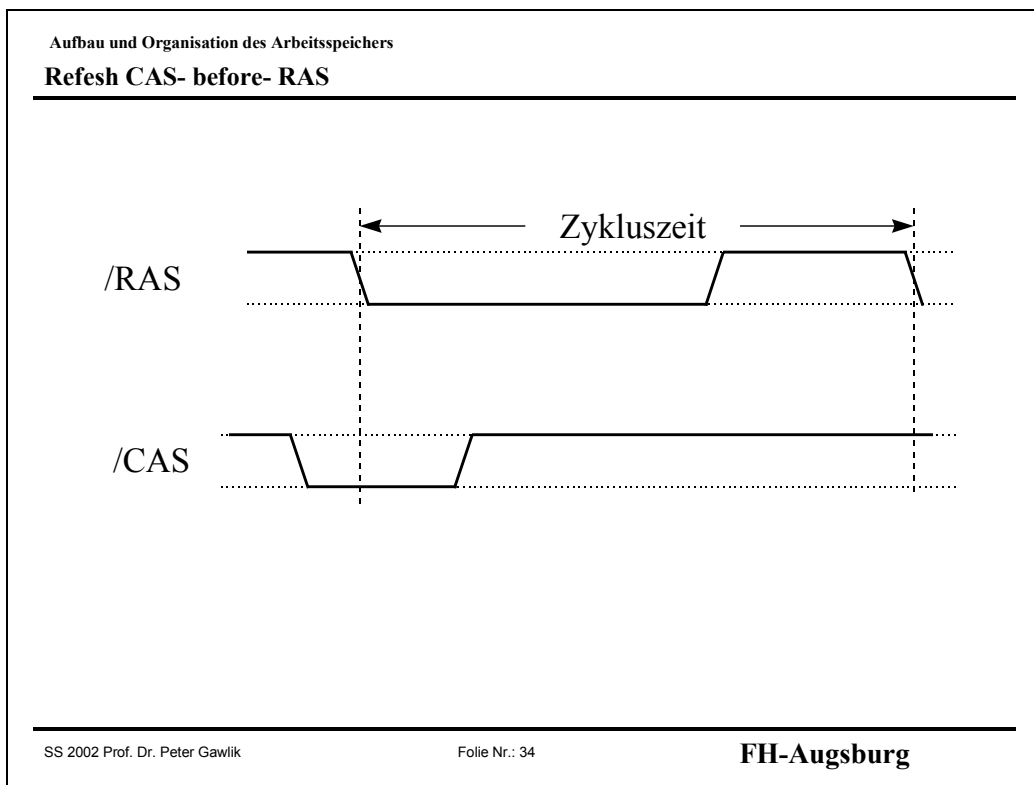
- **Ähnlich dem Lesezugriff**
- **Zeilenadressen werden intern oder extern erzeugt**
- **Eine Zeile wird gleichzeitig aufgefrischt**
- **3-5% der Zugriffszeit geht im Mittel durch Refresh verloren**
- **Es gibt mehrere Refresh-Modi**

Folie 33



Beim RAS-Only-Refresh wird von außen die aufzufrischende Zeilenadresse angelegt. Und nur /RAS aktiviert. Innerhalb einer spezifizierten Zeit muss dies für jede Zeile einmal gemacht werden. Extern muss ein Zähler vorhanden sein, der bei jedem Refresh erhöht wird. Diesen Modus beherrschen alle DRAM-Typen.

Folie 34



In diesem Fall besitzt das DRAM einen internen Adresszähler. Eine Refresh-Adresse braucht nicht angelegt zu werden. Durch die umgekehrte Signalfolge wird der Refresh-Modus aktiviert und der interne Refreshzähler inkrementiert. Dieser Modus ist bei den heutigen Bausteinen Standard.

DRAM - Controller

Wie oben gezeigt, sind die erforderlichen Timings für DRAMs komplex. Dementsprechend sind für die Generierung komplexe Schaltungen erforderlich. Diese werden heute mit den DRAM-Controllern angeboten oder sie sind schon auf dem μ P-Chip oder in Chipsätzen integriert. Sie stellen die Schnittstelle zwischen μ P und DRAM dar. Eine detaillierte Beschreibung eines DRAM-Controllers befindet sich z.B. in /Bähring Mikrorechner-Systeme/.

Folie 35

Aufbau und Organisation des Arbeitsspeichers

Übungen 1

- 1. Welche Speicherarten kennen Sie?**
- 2. Skizzieren Sie mögliche Koppellemente eines ROMs**
- 3. Wie wird beim PROM die Programmierbarkeit erzeugt?**
- 4. Skizzieren Sie den Querschnitt einer EPROM-Zelle.**
- 5. Wie wird eine EPROM-Zelle programmiert?**
- 6. Wie wird eine EPROM-Zelle gelöscht?**
- 7. Was ist ein Floating Gate?**
- 8. Was ist der Unterschied zwischen einem EPROM und einem EEPROM?**
- 9. Was ist der Vorteil von Flash-EPROMs?**
- 10. Skizzieren Sie den Querschnitt einer DRAM-Zelle.**

Aufbau und Organisation des Arbeitsspeichers

Übungen 2

1. Skizzieren Sie den prinzipiellen Aufbau eines DRAM-Speichers
2. Wozu werden die Signale RAS und CAS verwendet?
3. Was ist die Vorladezeit?
4. Erklären Sie den Unterschied zwischen Zykluszeit und Zugriffszeit?
5. Nennen Sie Methoden, um den Zugriff auf DRAMs zu beschleunigen.
6. Warum müssen DRAM refreshed werden?
7. Nennen Sie mögliche Refresh-Methoden.
8. Was passiert beim Refreshen von DRAMs?
9. Kann man DRAMs direkt am MC68000 anschließen? - Begründung

SS 2002 Prof. Dr. Peter Gawlik Folie Nr.: 36 **FH-Augsburg**

Aufbau und Organisation des Arbeitsspeichers

Übungen 3

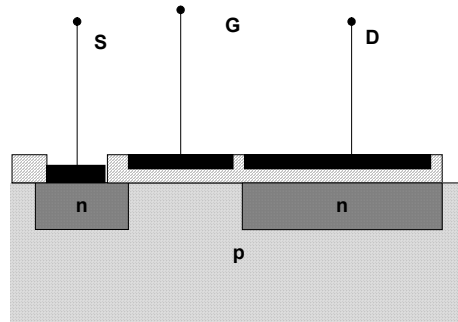
Versuchen Sie folgend aufgeführte Kürzel zu entziffern

SRAM
DRAM
EDO-RAM
BEDO-RAM
VRAM
WRAM
SDRAM
SGRAM
MDRAM
RDRAM
SLDRAM
EDRAM
CDRAM
3DRAM
FCRAM
ESDRAM

SS 2002 Prof. Dr. Peter Gawlik Folie Nr.: 37 **FH-Augsburg**

Übungen 4

Das folgende Bild zeigt ein DRAM im Querschnitt.



a) Tragen Sie U_{GS} und U_{DS} so an, dass der Speicherkondensator geladen wird.

b) Markieren Sie mit einigen + für positive Ladungsträger und einigen - für negative Ladungsträger den geladenen Kondensator im obigen Bild.

Folie 39

Übungen 5

Unten ist das Schaltbild einer DRAM Speichereinheit gezeigt. Erklären Sie

- wozu dient T1?
- den Lesevorgang
- den Schreibvorgang

