

Vorlesung 11

Themen dieser Vorlesung sind die Speichertechnologien

- Statischer RAM
- Dynamischer RAM
- EEPROM
- FLASH

Abbildung 1 zeigt die Klassifizierung von Speicherelementen.

Wir können die Halbleiterspeicherelemente in Festwertspeicher (Nur-Lesespeicher) und in Schreib-Lesespeicher unterteilen. Statischer- und dynamischer read only memory (RAM) Speicher verlieren die gespeicherte Information, wenn die Spannungsversorgung ausgeschaltet wird. Sie sind flüchtige Speicher (volatile memories). Andere Speicherarten sind permanent. ROM steht für read only memory, PROM für programmable ROM. PROM wird mit speziellen Geräten irreversibel programmiert. Erasable PROM (EPROM) kann mit UV Licht gelöscht werden um es wieder zu programmieren. Electrically erasable PROM (EEPROM) kann elektrisch programmiert und gelöscht werden. Flash ist eine Weiterentwicklung von EEPROM mit mehr Speicherkapazität. Ferroelectric-, Magnetoresistive- und Phase-Change RAM sind nichtflüchtige RAM Varianten.

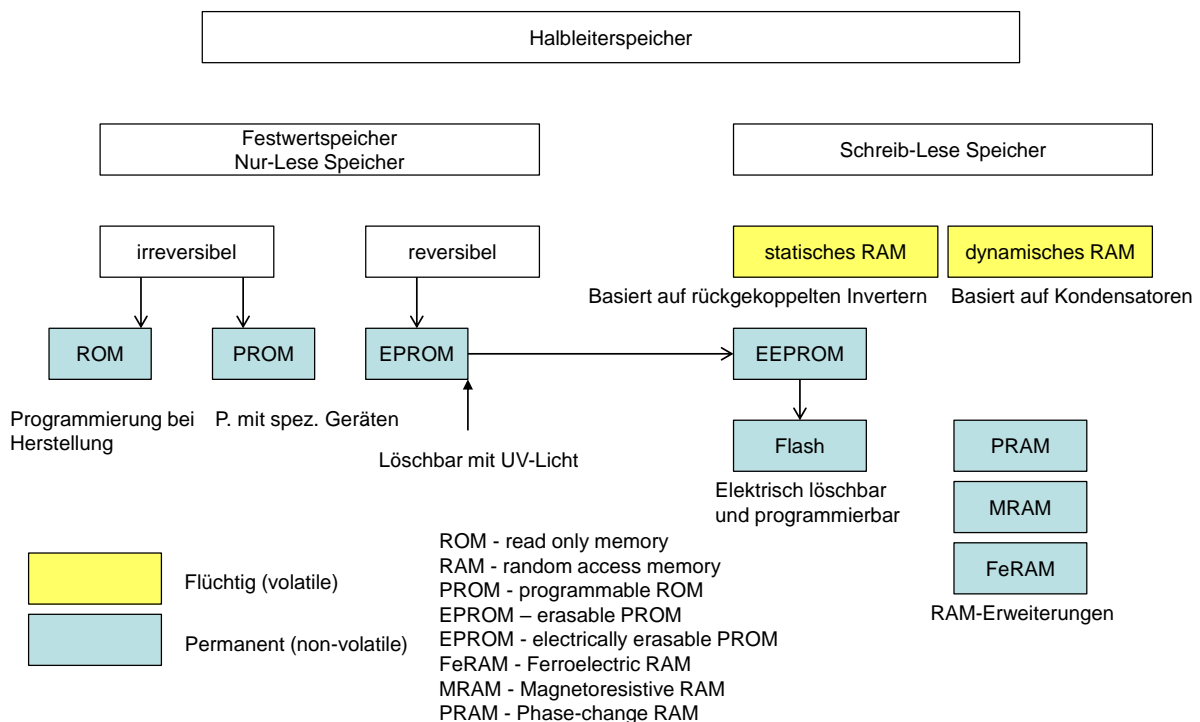


Abbildung 1: Klassifizierung von Speicherelementen

Abbildung 2 zeigt die Standardstruktur des Speicherblocks. Eine Speichermatrix besteht aus Speicherzellen. Die Zeilen der Matrix stellen „Worte“ dar. Die Speicherzellen eines Wortes (memory word) sind an eine gemeinsame Schreib/Lese-Linie (read write line – RW line)

angeschlossen. Wenn diese Linie logisch Eins wird, werden die Ausgänge der Speicherzellen mit den Bit-Linien verbunden. Ein Adressendekoder setzt die Read-Linien. Z.B. Wenn die Adresse = 8 ist, wird die RW-Linie 8 aktiviert. Die Bit-Linien werden sowohl fürs Schreiben als auch fürs Lesen benutzt. Beim Schreiben werden die Bit-Linien auf die gewünschten Potentiale gelegt. Das erreicht man mithilfe von Treibern (driver) in der Peripherie. Fürs Lesen werden die Bit-Linien von den Treibern getrennt, sie sind dann im hochohmigen Zustand. Die Speicherzellen (mit RW-Linie = 1) können dann ihre Zustände an die Bit-Linien übertragen. Manchmal werde auch Verstärker benutzt (sense amplifier). Ein Problem der einfachen Struktur ist es, dass die Höhe der Matrix oft viel größer als ihre Breite ist.

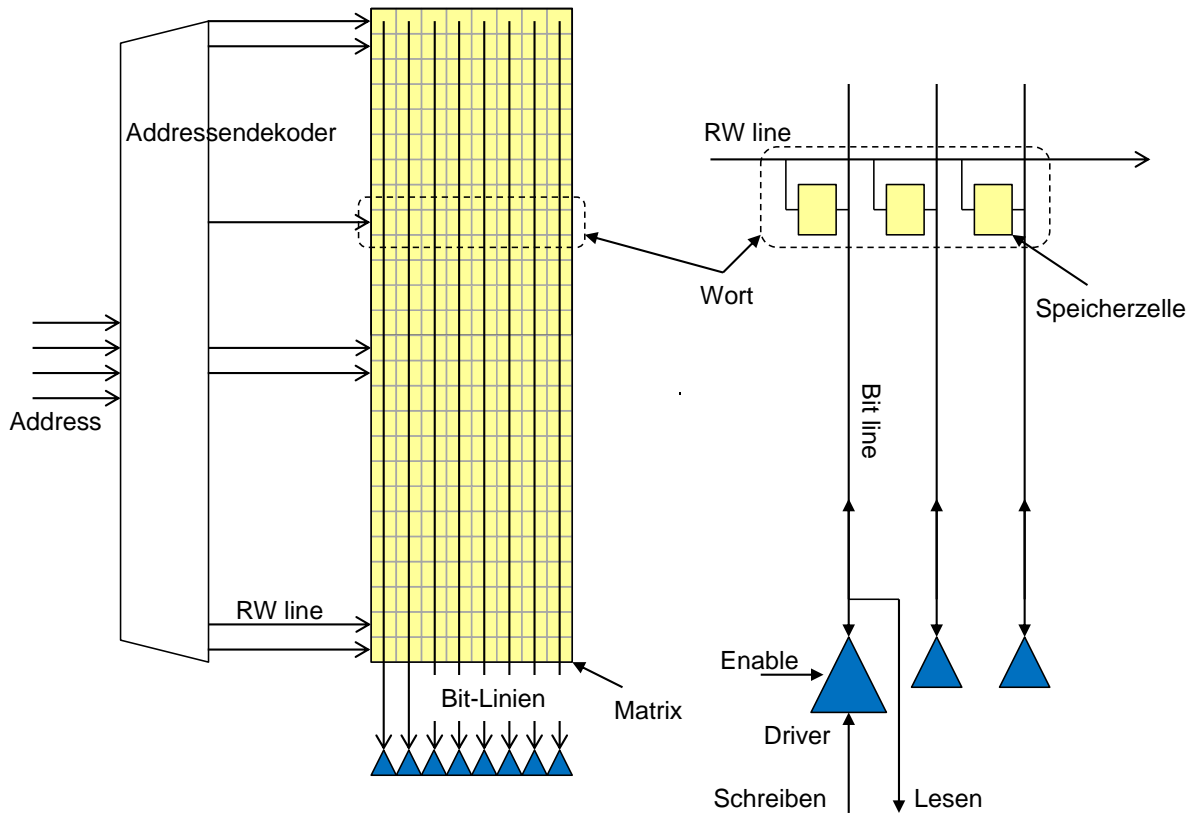


Abbildung 2: Struktur eines Speicherblocks

Dieses Problem kann mit der Struktur von Abbildung 3 behoben werden.

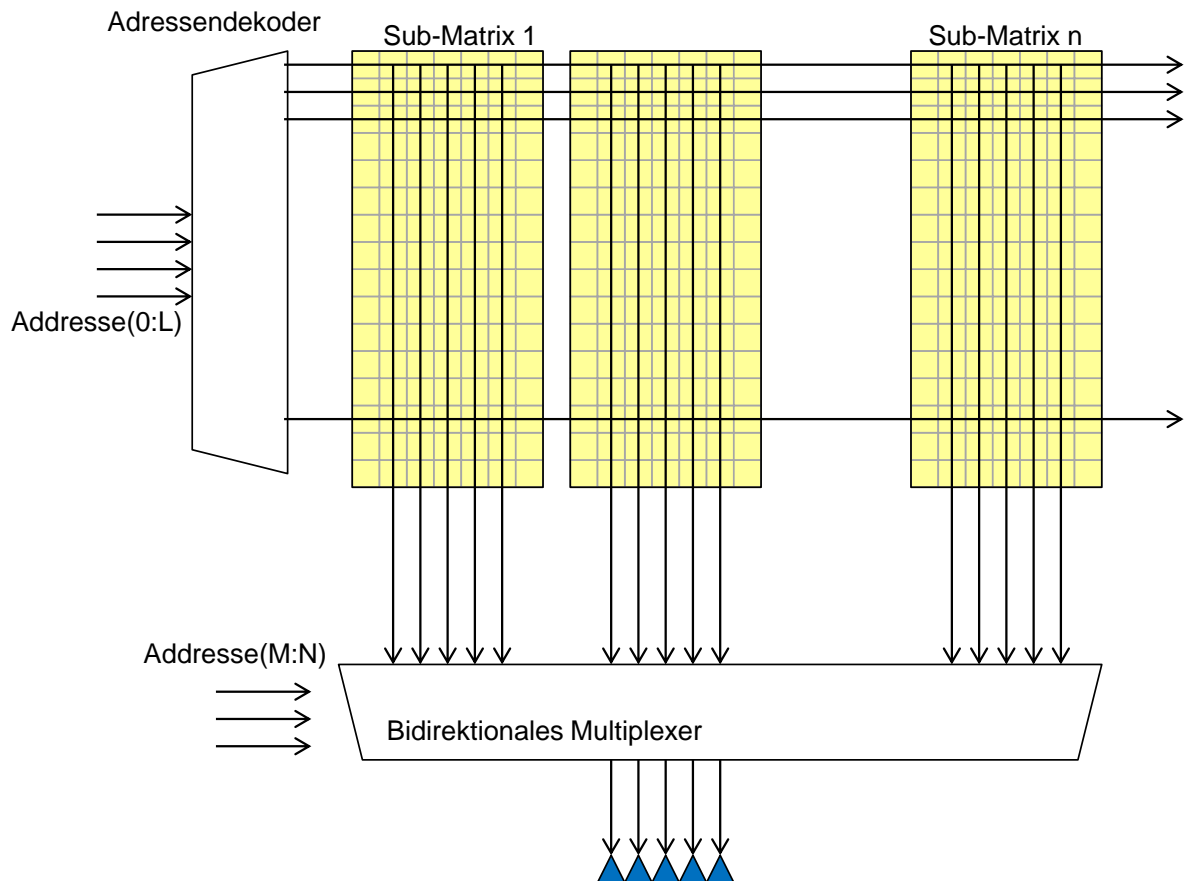


Abbildung 3: Speicher mit mehreren Blöcken

Statisches RAM

[Random-Access Memory – Wikipedia](#)

Random-Access Memory (zu Deutsch: „Speicher mit wahlfreiem/direktem Zugriff“ = Direktzugriffsspeicher)

Die Bezeichnung des Speichertyps als „wahlfrei“ bedeutet in diesem Zusammenhang, dass jede Speicherzelle (Speicherwort) über ihre Speicheradresse direkt angesprochen werden kann. Der Speicher muss also nicht sequenziell oder in Blöcken ausgelesen werden.

Der Begriff Random-Access Memory wird heute immer im Sinne von „Schreib-lese-RAM“ (read-write random-access memory – RWRAM) verwendet.

(Kein ROM)

Die Entstehung des Begriffs geht in die Anfangszeit der modernen Computer zurück, bei denen alle Daten auf sequenziell zu lesenden Speicherformen wie Lochkarten oder Magnetbändern vorlagen, die zur Verarbeitung in schnelle Rechenregister geladen wurden.

Die wesentliche Neuerung bestand im wahlfreien Zugriff der magnetischen Kernspeicher und der nachfolgend auf Halbleiterspeichern aufsetzenden RAM-Bausteine.

Statische RAM oder SRAM Zellen basieren auf einem rückgekoppelten Buffer. Wir haben diese Struktur bereits in Vorlesung 4 vorgestellt. Es handelt sich um eine Kippschaltung. Sie ist ebenfalls die Basis für Flipflops.

Wenn wir auf gleichem Graph die Kennlinie des Buffers und die Kennlinie der Rückkopplung zeichnen, erkennen wir, dass es drei Schnittpunkte gibt. Zwei davon entsprechen den stabilen Arbeitspunkten, logisch 1 und 0. SRAM Zelle bleibt in diesen Arbeitspunkten solange die Spannungsversorgung besteht.

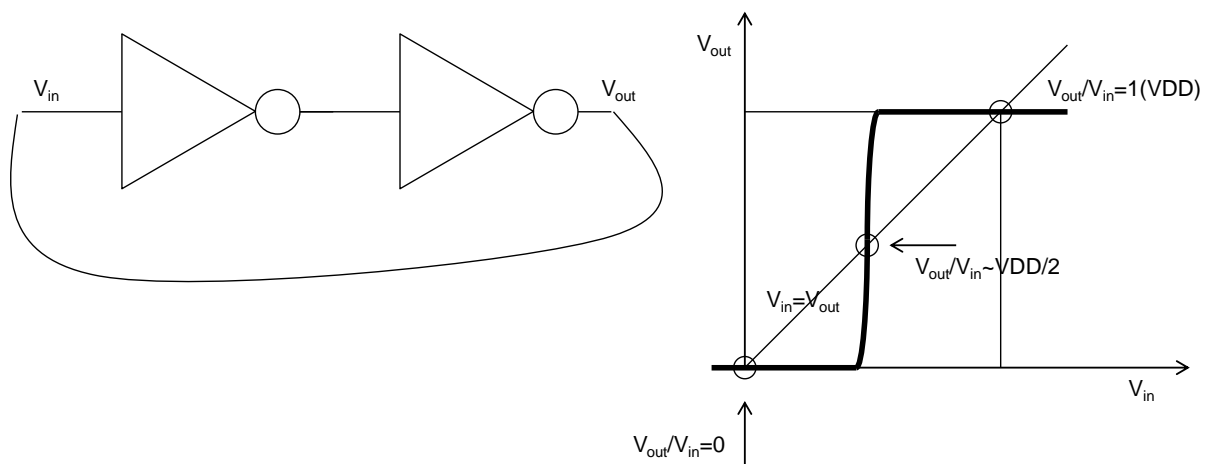


Abbildung 4: Bistabile Kippschaltung (Kippstufe)

Abbildung 5 zeigt den Schaltplan einer SRAM Zelle, oben ist die vereinfachte Schaltung und unten die Schaltung auf Transistorebene.

Neben dem rückgekoppelten Buffer, der aus zwei Inverter besteht, gibt es noch zwei Transistoren T1 und T2. Diese Transistoren verbinden die Knoten QN und Q mit den Bit-Linien. Der Wert wird in der SRAM Zelle „differenziell“ gespeichert, negiert (QN) und nicht-negiert (Q). Es gibt zwei Bit-Linien, eine für das negierte Signal (bit line N) und eine für das normale (nicht-negierte) Signal (bit line P).

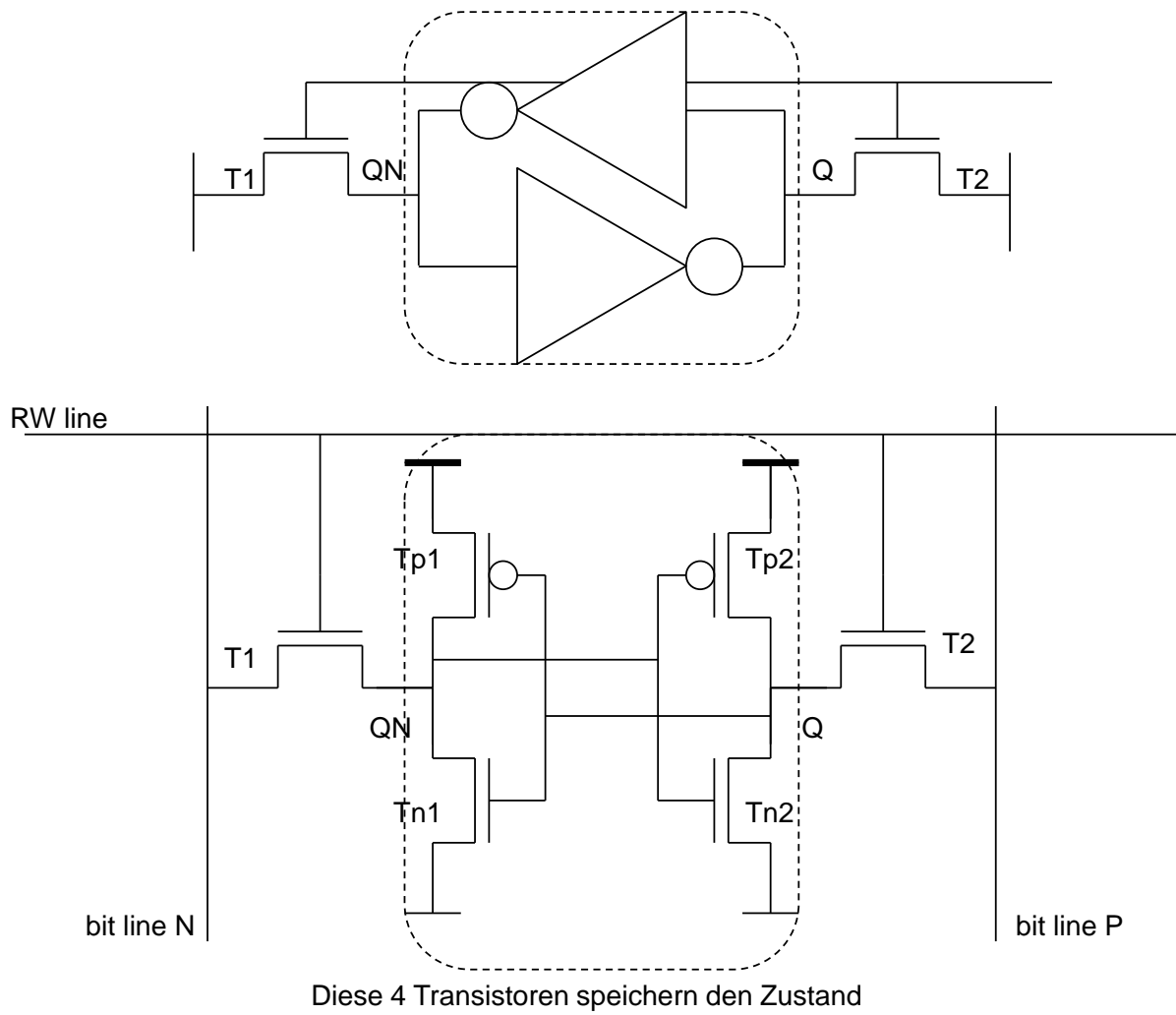


Abbildung 5: Statische RAM Zelle mit 6 Transistoren

Speicherzustand

Abbildung 6 zeigt die RAM Zelle im Zustand $Q = 0$.

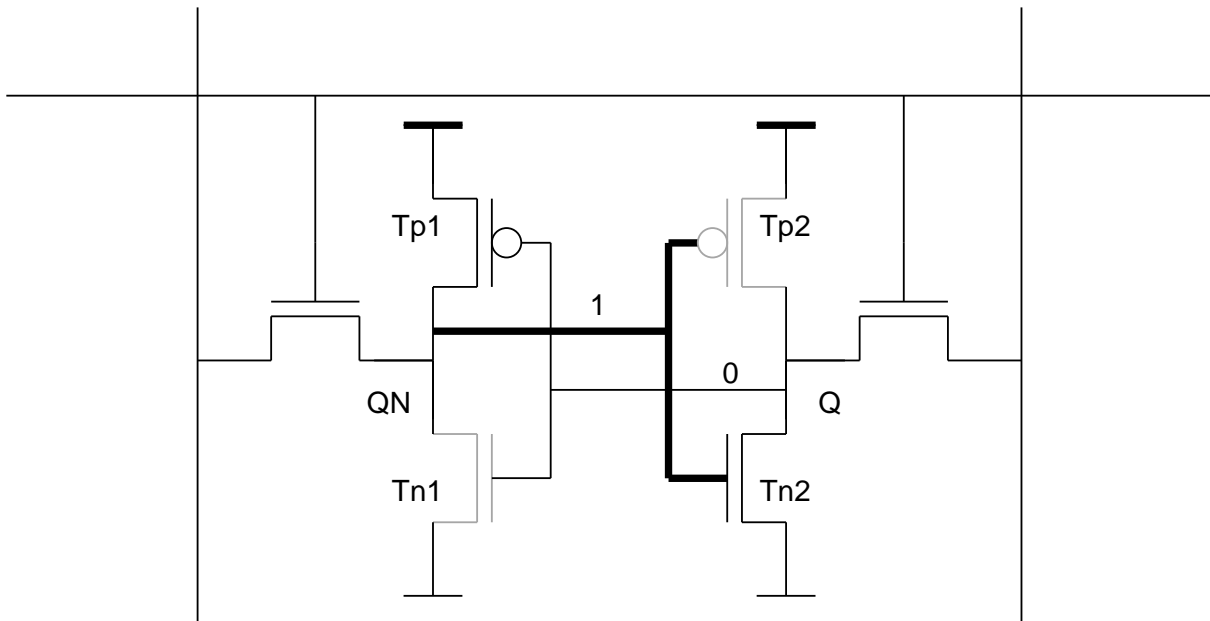


Abbildung 6. Speicherzustand

Die dicke Linien haben logisch 1 Potential, sie befindet im „Zustand 1“, die dünnen Linie sind im Zustand 0. Logisch 1 am PMOS Tp2 Gate schaltet den Transistor aus. Logisch 0 am NMOS Tn1 Gate schaltet den NMOS aus. Die ausgeschalteten Transistoren sind in der Abbildung ausgeblendet. Die verbliebenen (eingeschalteten) Transistoren Tn2 und Tp1 halten die Potentiale der Linien fest und verhindern, dass sich die Linien entladen.

SRAM Programmierung: Funktionsweise

Wie wird SRAM Zelle beschrieben? Die Annahmen sind, dass ein NMOS Transistor stärker leitet als ein PMOS und dass sich die RAM Zelle vor dem Schreibvorgang im Zustand $Q = 0$ befindet.

Die Bit-Linien werden, mittels Treiber in der Peripherie, auf die gewünschten Potentiale gelegt. Die Bit-Linie P wird auf 1 gelegt und die Bit-Linie N auf 0. Read-Linie wird auf 1 geschaltet.

Sind zwei Bit-Linien und zwei zum Schreiben notwendig?

Ja!

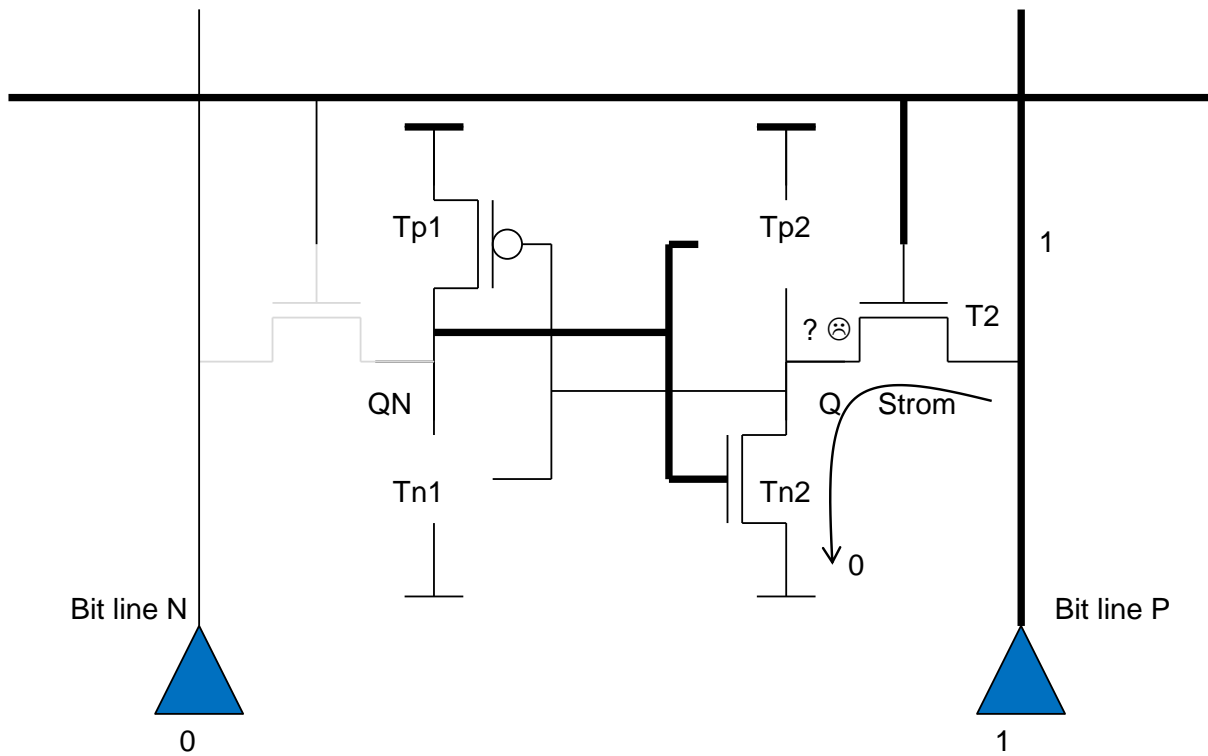


Abbildung 7: Kein Schreiben möglich

Es wäre unmöglich nur mit der positiven Bit-Linie, und nur mit T_2 , logisch Eins in die RAM Zelle zu schreiben, wenn sie sich in $Q = 0$ Zustand befindet.

Das ist in Abbildung 7 illustriert. Transistor T_2 leitet wegen der kleinen V_{gs} schwächer als T_{n1} und kann das Potential Q nicht über die Schwelle des Inverters T_{n1}/T_{p2} (etwa $V_{DD}/2$) erhöhen.

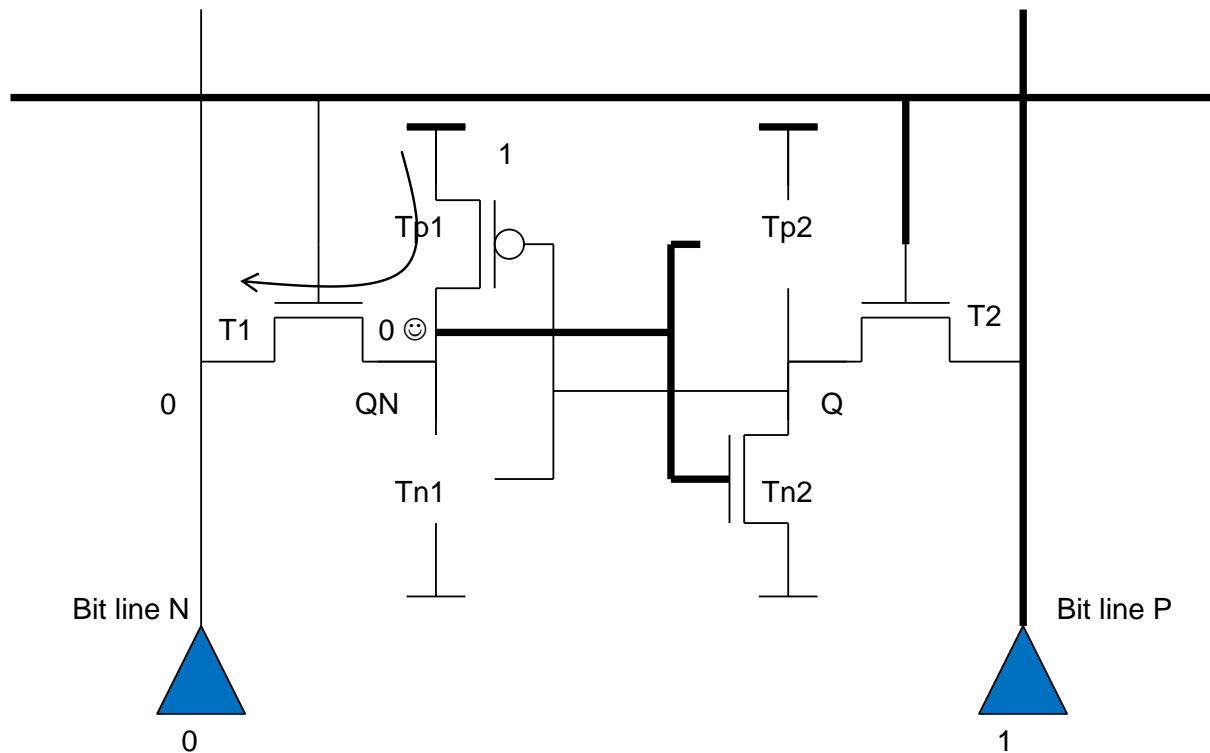


Abbildung 8: Schreiben ist möglich

Die andere (negative) Bit-Linie ermöglicht das Schreiben. NMOS T1 leitet stärker als PMOS Tp2 und zieht das Potential QN unter die Schwelle. Das führt zum “Umkippen” von RAM Zelle. Sie kommt in den Zustand $Q = 1/QN = 0$ (Abbildung 8).

SRAM Lesen: Funktionsweise

Beim Lesen müssen die Bit-Linien in hochohmigen Zustand versetzt werden, die Trieber werden deaktiviert. Nehmen wir an, dass die Bitlinien Kapazitäten haben (C_a und C_2) und dass die Kapazitäten anders aufgeladen sind als die RAM Zelle selbst. Kapazität C_1 von Bit-Linie N befindet sich auf hohem Potential und QN ist logisch 0, Abbildung 9

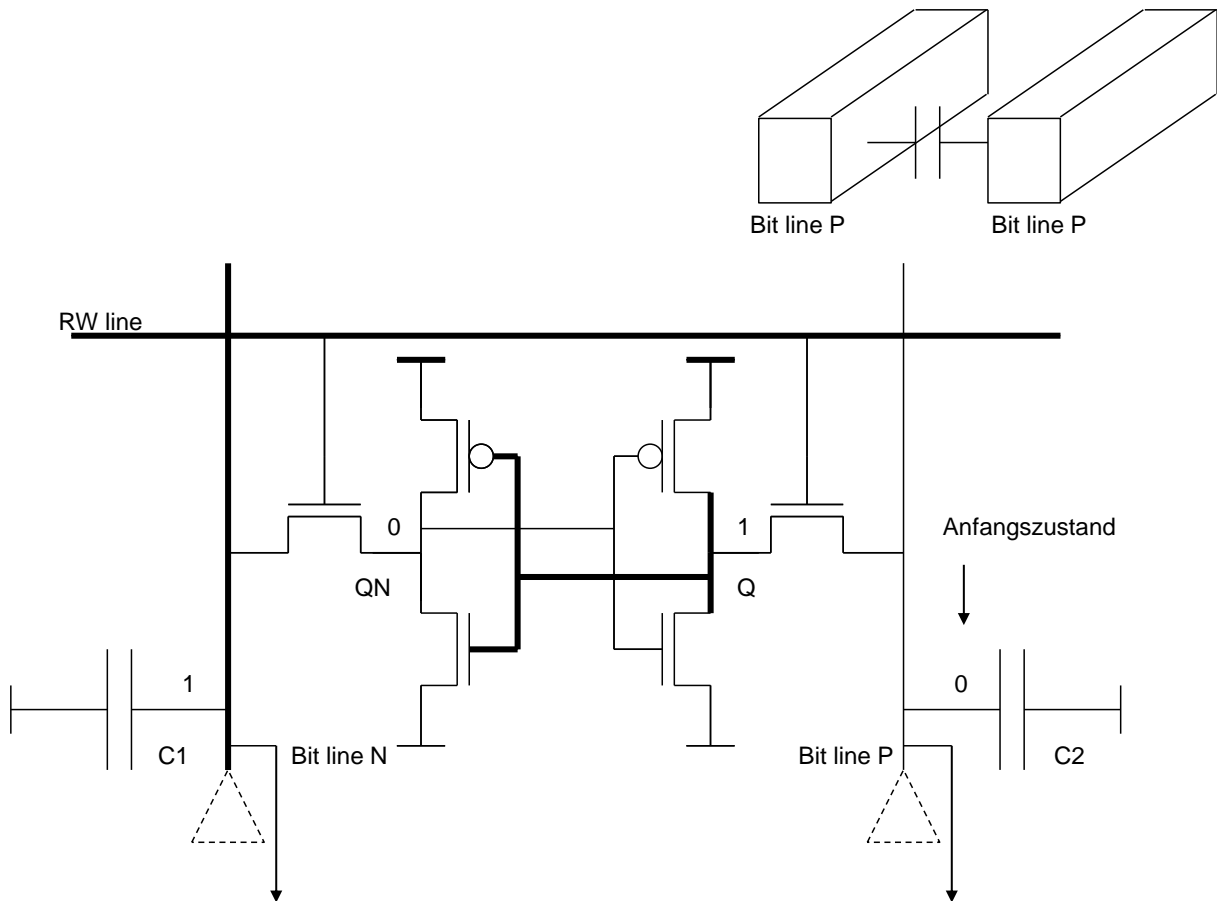


Abbildung 9: Lesevorgang, Anfangszustand

Beim Lesen wird RW-Linie wird auf 1 gelegt. Diskutieren wir nun ein mögliches Problem.

Es kann passieren, dass die RAM Zelle beim Lesevorgang überschrieben wird. Abbildung 10 zeigt das.

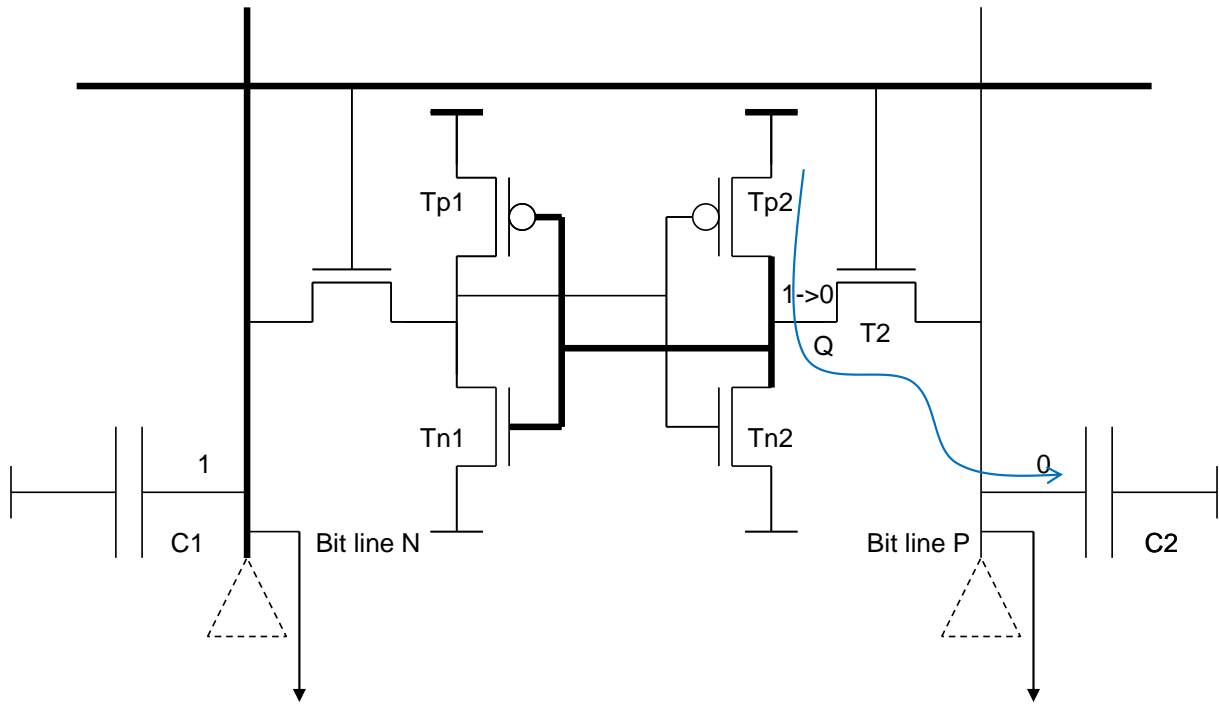


Abbildung 10: Fehler beim Lesen, SRAM Zelle wird versehentlich überschrieben

Kapazität C2 muss aufgeladen werden. Deswegen fließt ein Strom durch Tp2 und T2. Da der R_{on} Widerstand vom T2 kleiner ist als R_{on} vom Tp2, fällt das Potential von Q gegen 0V. Der Inverter Tn1/Tp1 erzeugt logisch 1 am QN und die RAM Zelle kippt um.

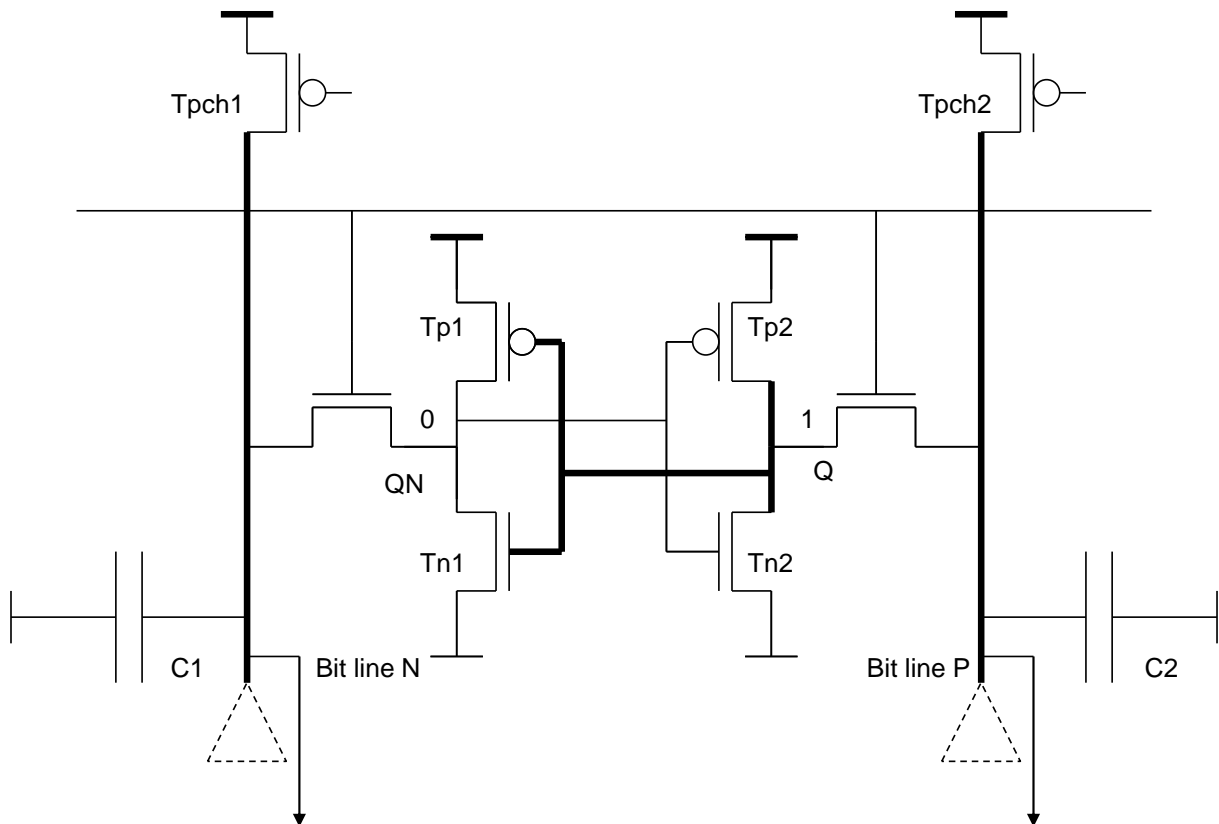


Abbildung 11: Precharge

Eine Lösung des Problems ist es, die Bitlinien vorm Lesen auf logisch 1 aufzuladen – precharge.

Dafür verwendet man die Transistoren T_{pch1} und T_{pch2} in Abbildung 11.

Wenn wir nun die RAM Zelle im Zustand $Q = 1$ lesen möchten, müssen T_{n1} und T_1 die Kapazität der Bit-Linie 1 auf Null entladen. Das ist jetzt einfacher als vorher. Wenn T_{n1} einen kleineren R_{on} wie T_1 hat (was leicht zu erreichen ist), steigt beim Entladen vom C_1 das Potential am Q_N nicht viel und die Zelle schafft es C_1 zu entladen ohne dass sie umkippt.

Mithilfe eines Verstärkers (sense amp) kann das Lesezyklus beschleunigt werden.

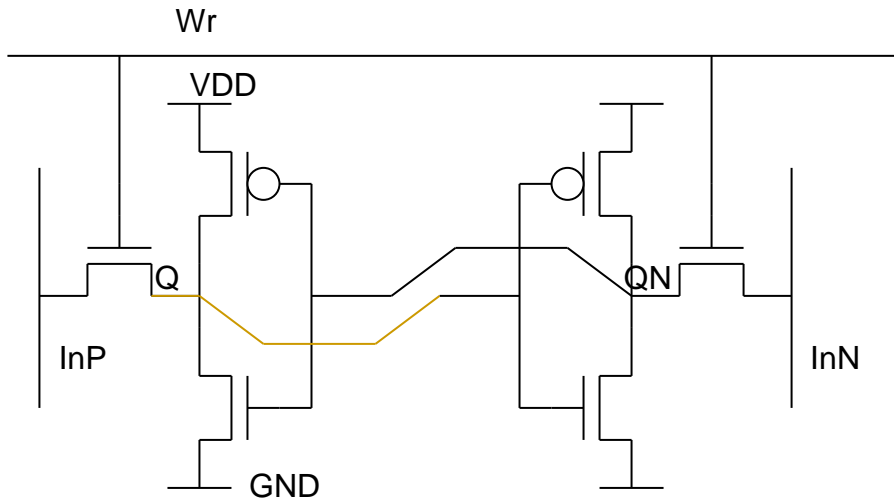


Abbildung 12: Schaltplan einer SRAM Zelle

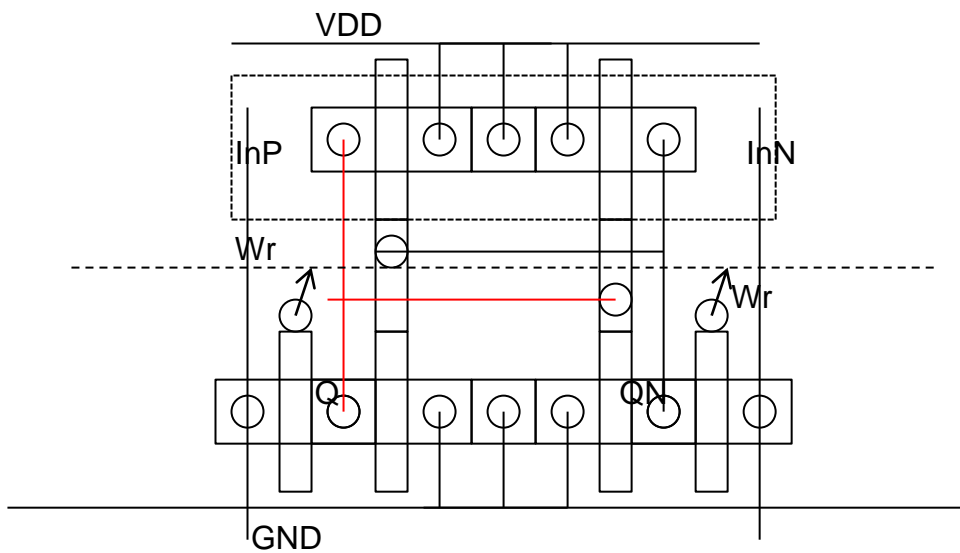


Abbildung 13: Layout der SRAM Zelle

Abbildungen 12 und 13 illustrieren das Layout einer RAM Zelle. Das Layout wird auf kleine Fläche optimiert, indem einige Kontakte durch Überlappen kurzgeschlossen werden.

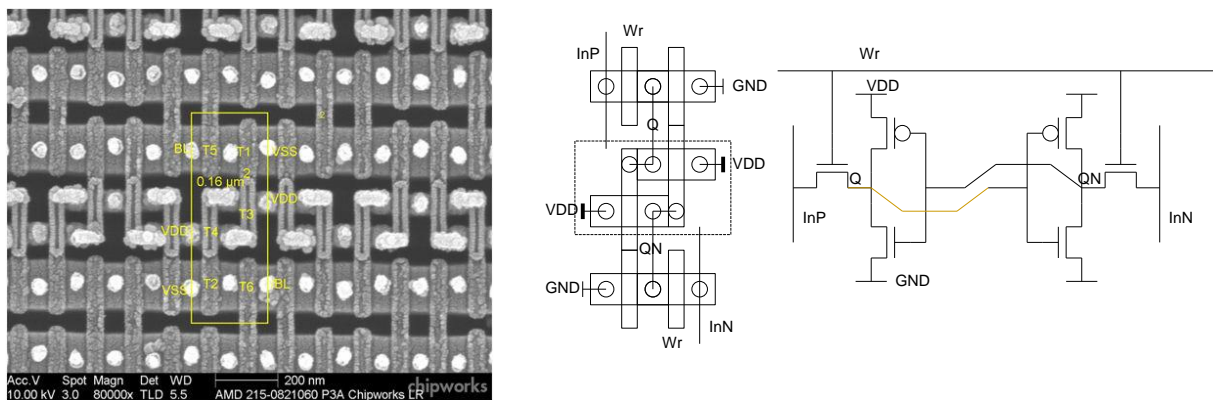


Abbildung 14: AMD 215-0821060 28 nm HP 6T-SRAM at Poly - Plan View SEM

Abb. 14 zeigt das Mikroskop-Foto der SRAM Zellen in einem Prozessor implementiert in einem 28 nm Prozess. Rechts sehen wir das schematische Layout, das dem Foto entspricht.

Dynamisches RAM

[Dynamic Random Access Memory – Wikipedia](#)

Dynamic Random Access Memory (DRAM) oder der halb eingedeutschte Begriff dynamisches RAM bezeichnet eine Technologie für einen elektronischen Speicherbaustein mit wahlfreiem Zugriff (Random-Access Memory, RAM), der hauptsächlich in Computern eingesetzt wird, jedoch auch in anderen elektronischen Geräten wie zum Beispiel Druckern zur Anwendung kommt. Das speichernde Element ist dabei ein Kondensator, der entweder geladen oder entladen ist. Über einen Schalttransistor wird er zugänglich und entweder ausgelesen oder mit neuem Inhalt beschrieben.

Der Speicherinhalt ist flüchtig, das heißt die gespeicherte Information geht bei fehlender Betriebsspannung oder zu später Wiederauffrischung verloren.

Ein Kennzeichen des DRAM ist die Kombination aus einer sehr hohen Datendichte verbunden mit sehr preiswerten Herstellungskosten. Es findet deswegen vor allem dort Verwendung, wo große Speichermengen bei mittleren Zugriffszeiten (verglichen mit statischem RAM, SRAM) zur Verfügung gestellt werden müssen.

Der Speicherinhalt muss bei DRAMs im Gegensatz zu SRAMs zyklisch aufgefrischt werden (Refresh). Dies ist normalerweise in Abständen von einigen zig Millisekunden erforderlich.

Dynamische RAM-Zellen (DRAM) basieren auf Speicherung von logischen Niveaus auf einem Kondensator. Abbildung 15 zeigt eine DRAM Variante mit 3 Transistoren und einem Kondensator. Trd wird zum Lesen, Twr zum Schreiben und T1 als invertierender Verstärker verwendet. Die Zelle hat getrennte Read- und Write-Linien.

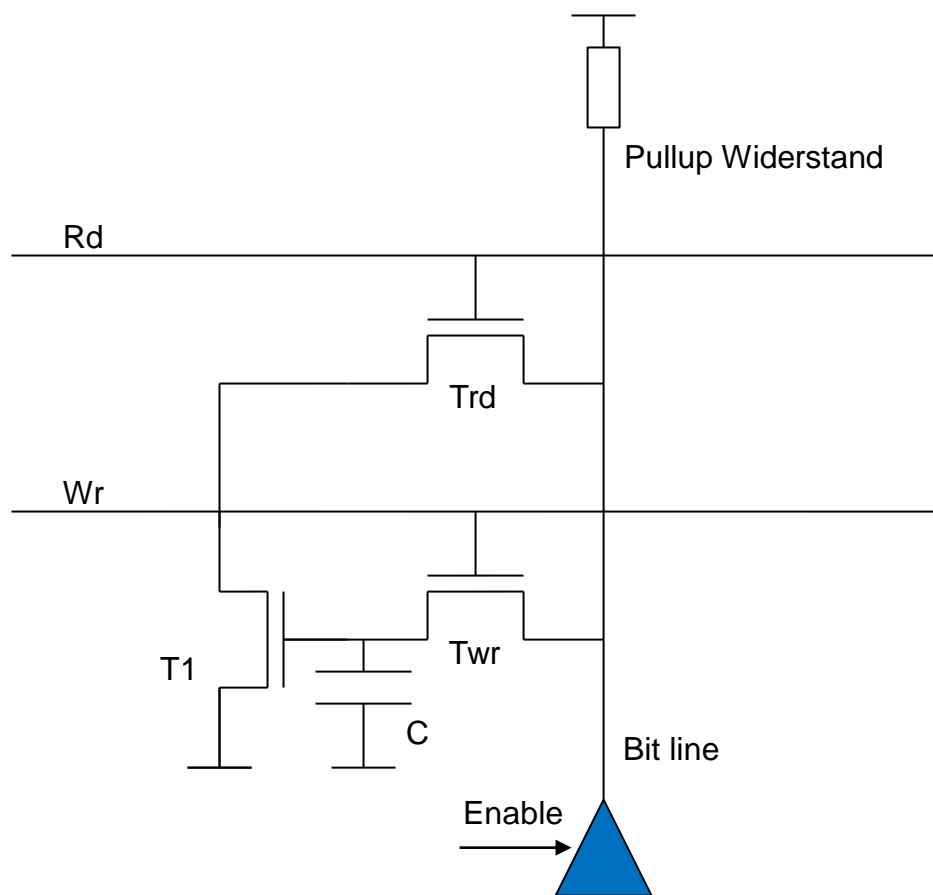


Abbildung 15: DRAM

Beim Schreiben wird Write-Linie (Wr) auf 1 geschaltet, Bit-Linie auf das gewünschte Potential gelegt und der Kondensator aufgeladen.

Eine dynamische RAM Zelle kann nicht lange ihren Zustand halten. Transistor Twr ist nicht perfekt „dicht“, wenn er ausgeschaltet ist, es entsteht ein Leckstrom und der Kondensator wird innerhalb typischerweise 1 ms entladen (Abbildung 16).

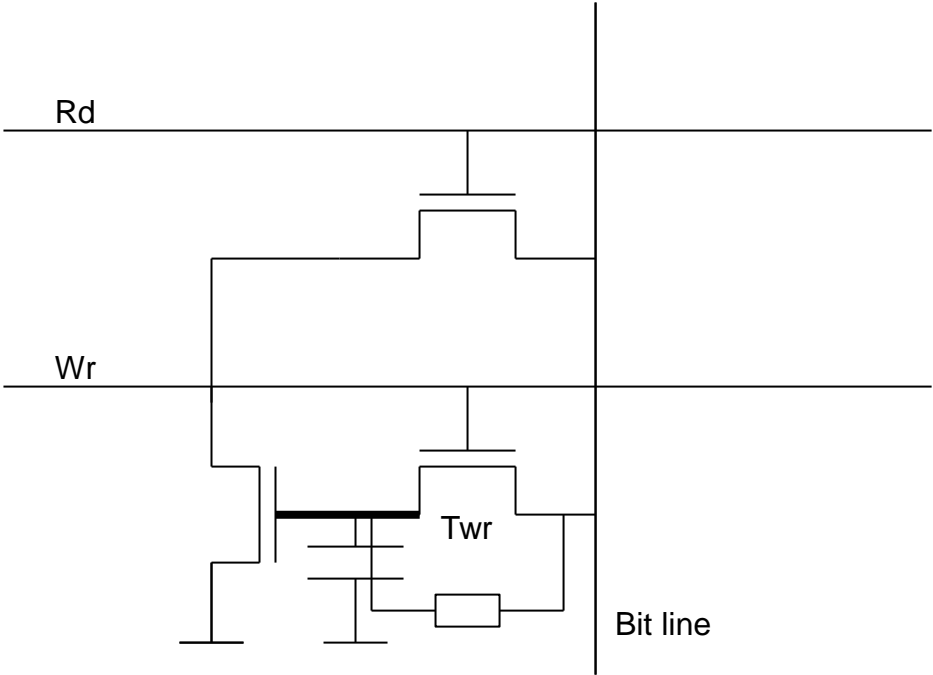


Abbildung 16: Kondensator wird entladen

Beim Lesen (Abbildung 17) wird Linie Rd auf logisch Eins gelegt und die Bit-Linie vom Treiber getrennt.

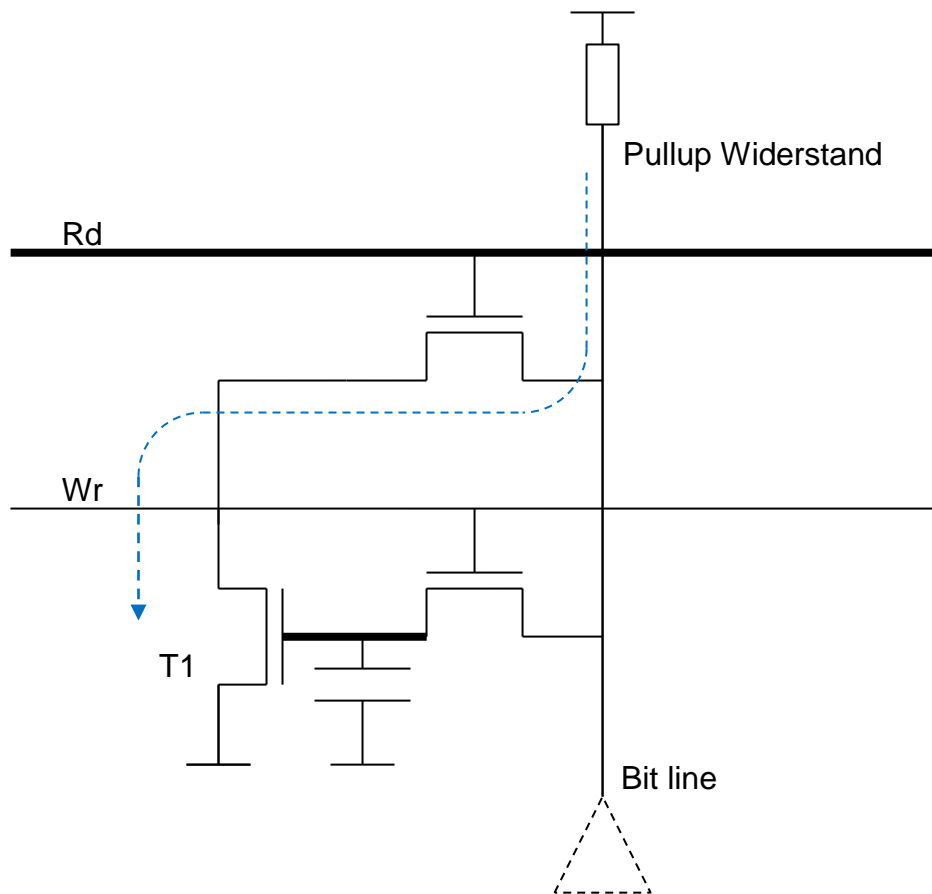


Abbildung 17: Lesen vom DRAM

Transistor T_1 dient als Verstärker. Seine Gate-Spannung ist hoch, weil der Kondensator aufgeladen ist (die Zelle ist im Eins-Zustand). Deswegen erzeugt der Transistor T_1 Strom, der in die Bit-Linie fließt. Zustand der DRAM Zelle kann als Spannung am Pullup Widerstand gemessen werden.

Wir zeigen nun eine noch kleinere Realisierung der DRAM Zelle. Die Zelle enthält nur einen Transistor und einen Kondensator (Abbildung 18).

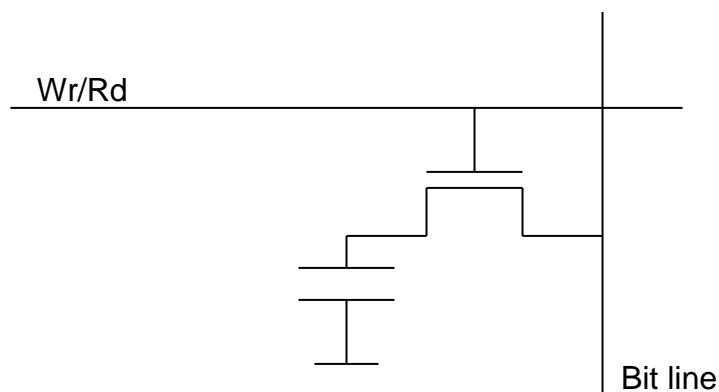


Abbildung 18: DRAM Zelle mit einem Transistor

Beim Schreiben von DRAM Zelle (Abbildung 19) wird W_r/R_d Linie auf das logische Niveau gelegt und der Kondensator aufgeladen (Bit-Linie ist mit dem Treiber verbunden).

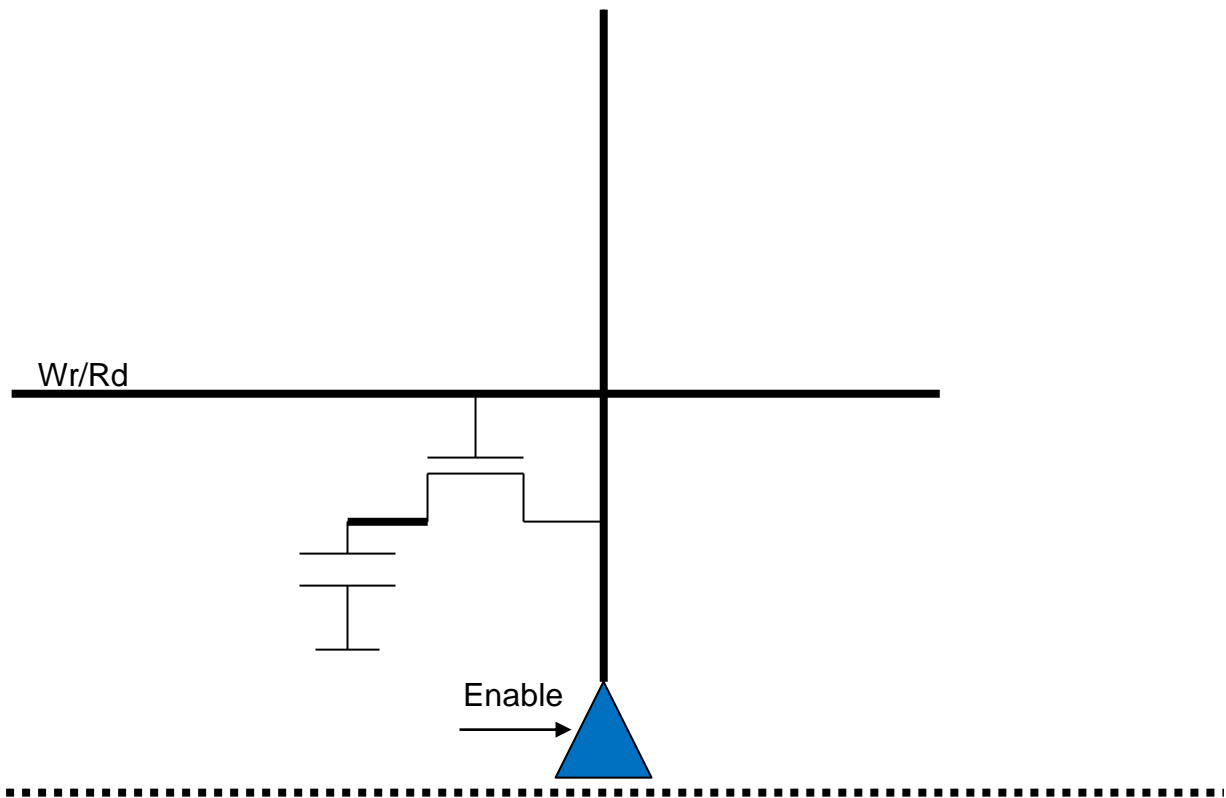


Abbildung 19: Schreiben in eine DRAM Zelle

Beim Lesen der DRAM Zelle (Abbildung 20) wird die Bit-Linie vom Buffer getrennt, mit einem precharge Schalter entladen (oder auf logisch 1 aufgeladen) und an den Kondensator der DRAM Zelle angeschlossen. Wir nehmen an, dass auf der DRAM Kapazität C logisch 1 gespeichert wurde. Beim Lesevorgang wird C teilweise entladen, da sich die gespeicherte Ladung auf Gesamtkapazität der DRAM Zelle und der Bit-Linie verteilt.

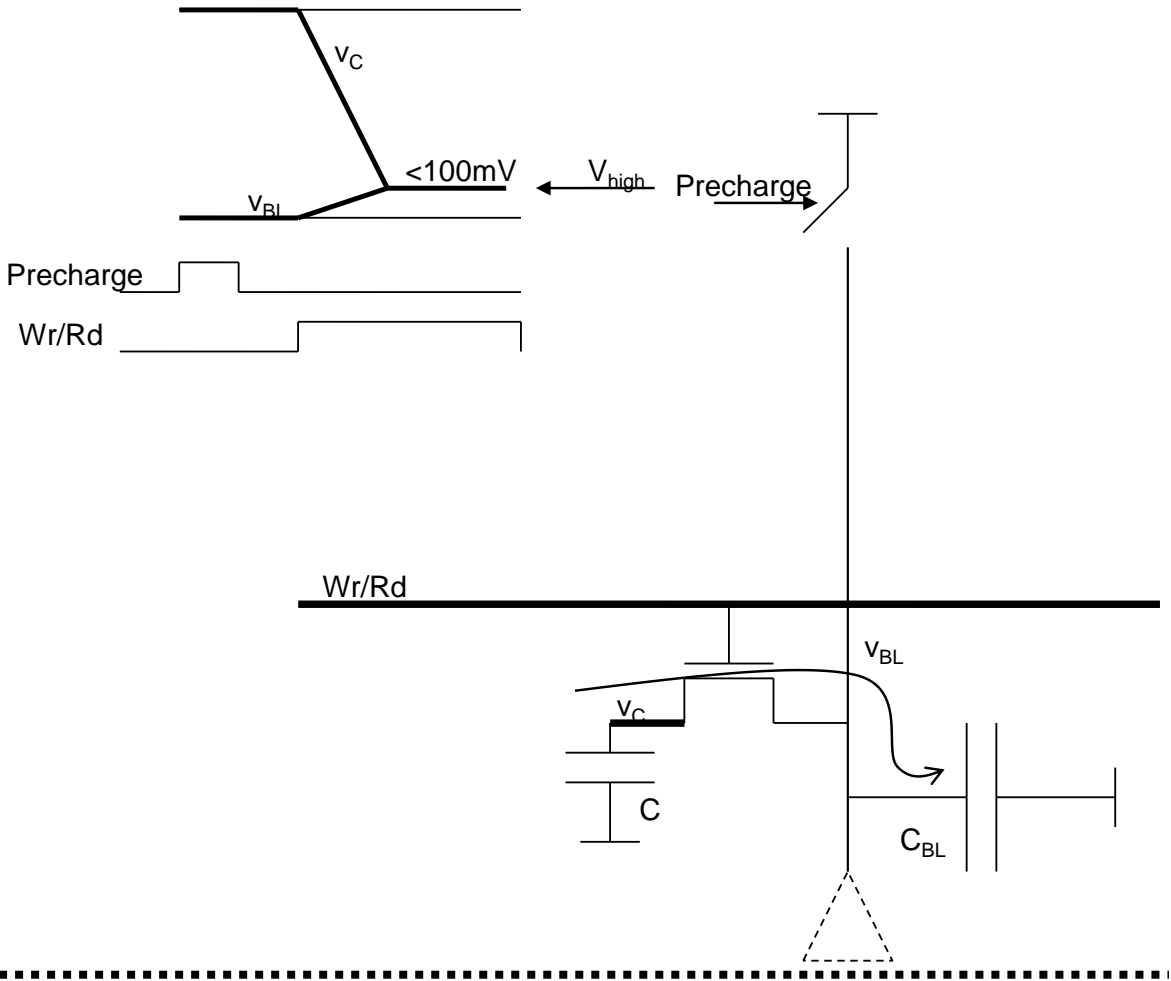


Abbildung 20: Lesen der DRAM Zelle

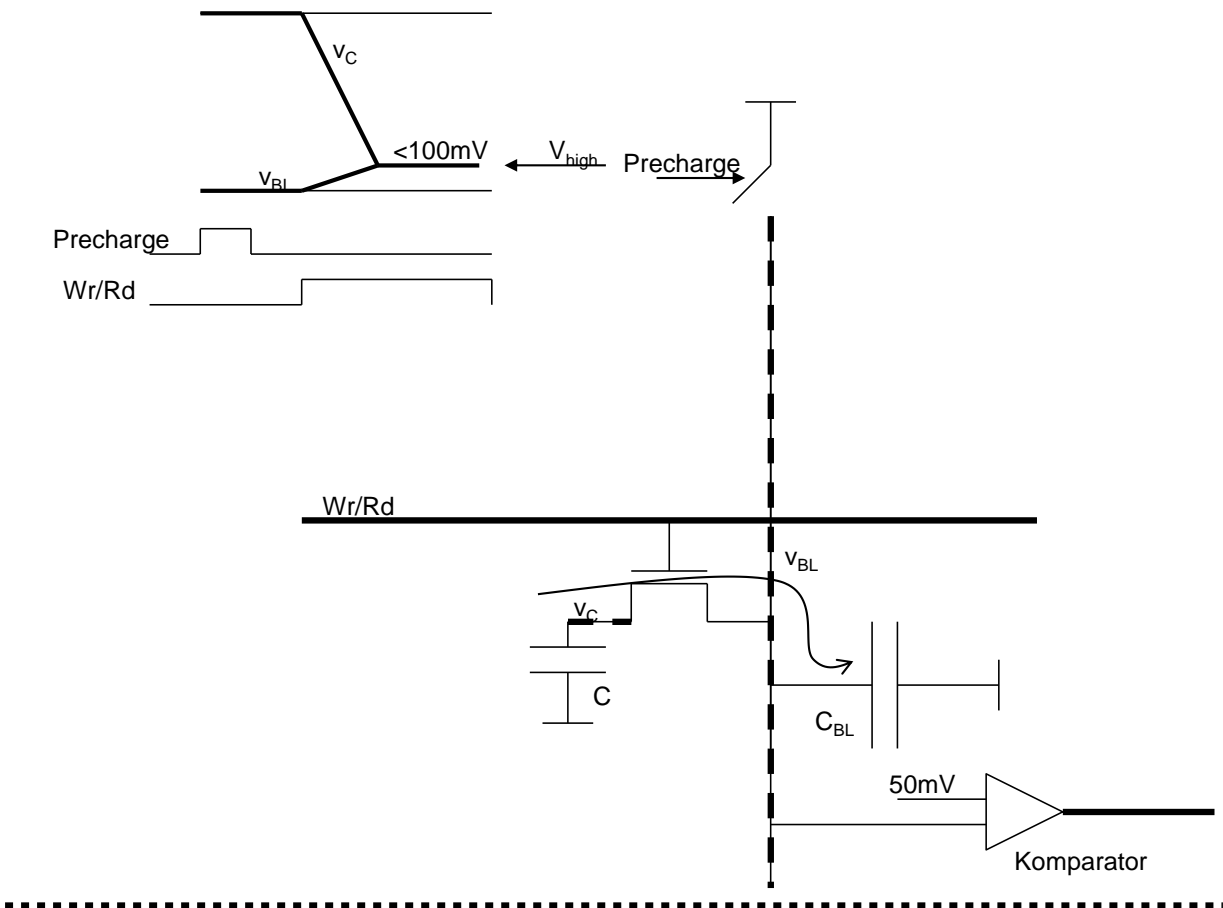


Abbildung 21: DRAM und Komparator

Abbildung 20 zeigt den Zeitverlauf von Signalen $v_C(t)$ und $v_{BL}(t)$ nach dem Einschalten von W_r/R_d -Linie. Spannung am Kondensator v_C sinkt vom V_{DD} (logisch 1) auf ein Wert V_{high} , gegeben durch das Verhältnis von Kapazitäten C und C_{BL} . V_{high} kann unter 100mV liegen. Potential v_{BL} steigt von 0 auf V_{high} .

Ein Verstärker oder ein analoger Komparator wird benutzt um V_{high} als logisch 1 zu erkennen (Abbildung 21).

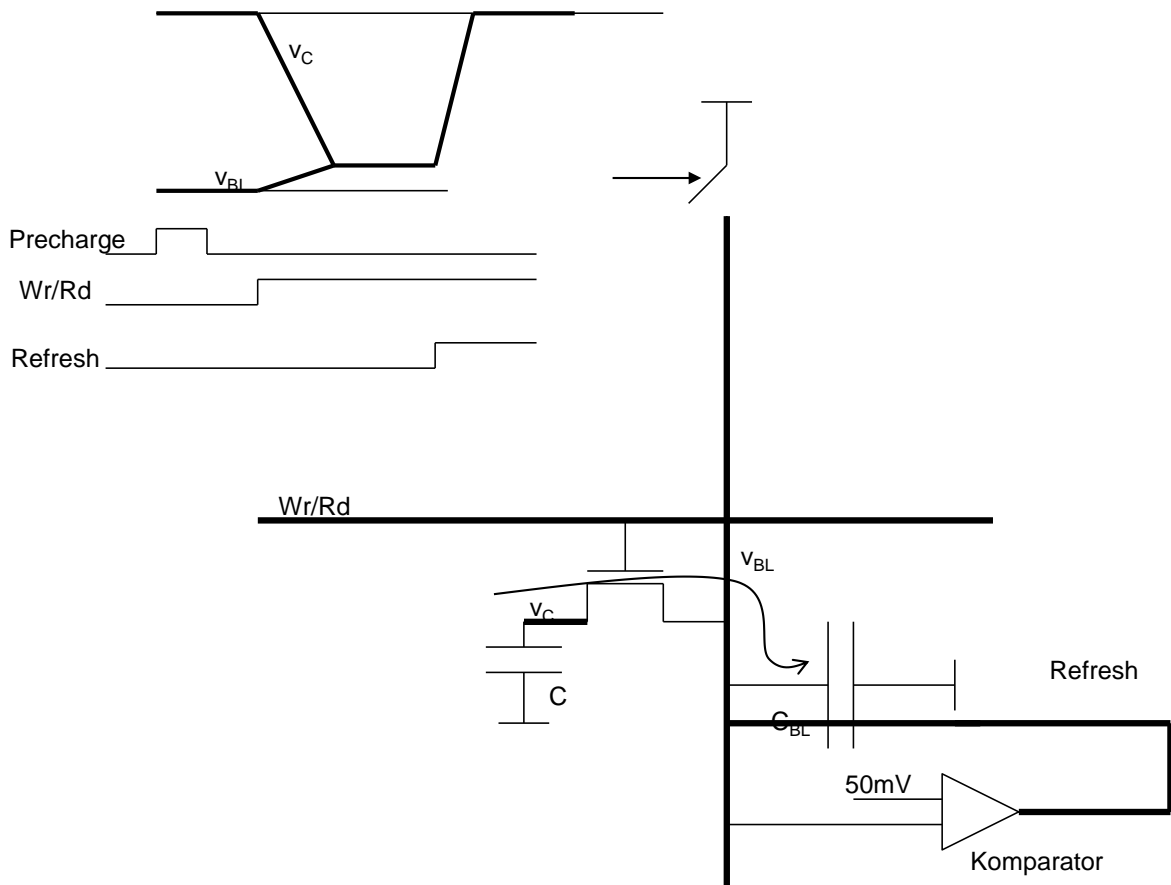


Abbildung 22: Refresh

Komparator-Ausgang wird an die Bit-Linie zurückgekoppelt, um den richtigen Spannungsniveau wieder herzustellen. Man nennt das Refresh (Abb. 22).

Abbildung 23 und Abbildung 24 zeigen wie eine DRAM Zelle typischerweise im Silizium implementiert wird. Es wird eine tiefe Elektrode (through silicon via) als Kondensator verwendet. Auf diese Weise nimmt die DRAM Zelle wenig Fläche. Ihr Kondensator hat große Kapazität und wird langsamer entladen und einfacher gelesen (da V_{high} größer ist).

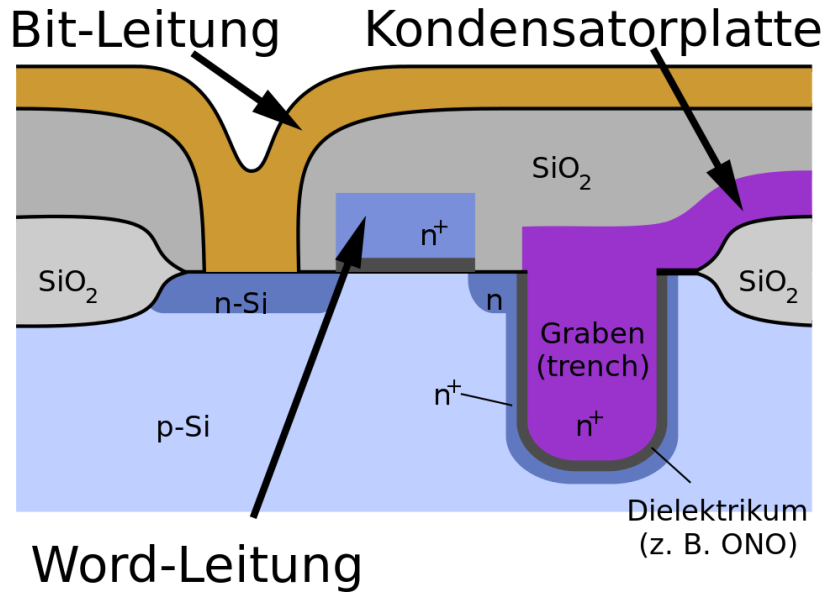


Abbildung 23: Vertikale Implementierung des Kondensators

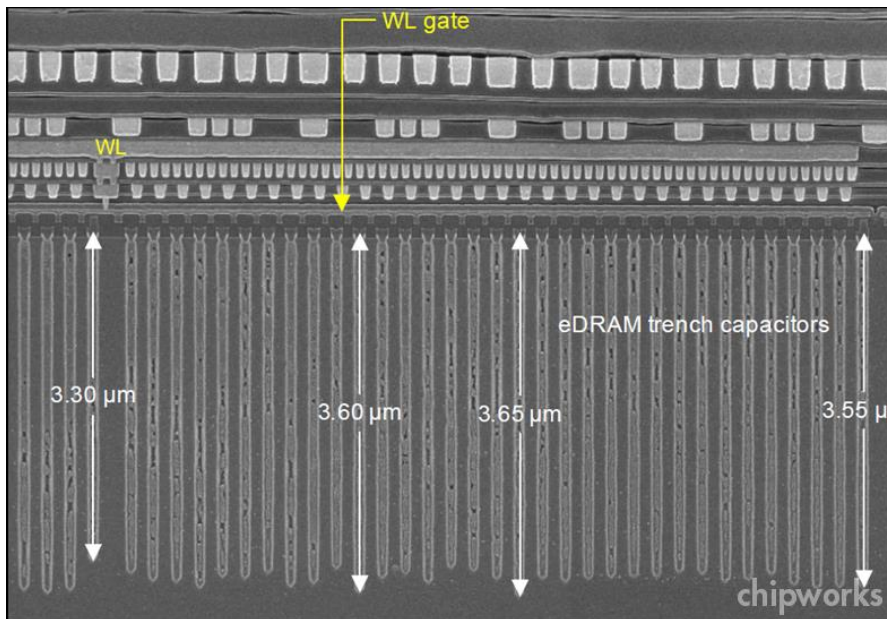


Abbildung 24: Embedded DRAM in IBM Power 7+ microprocessor (32-nm), <http://chipworksrealchips.blogspot.com/>

Permanentspeicher

SRAM und DRAM Zellen behalten die gespeicherte Information nur solange Spannungsversorgung erhalten bleibt. Jetzt werden wir auch einige Speicherkomponenten die Information permanent speichern beschreiben. Beispiele sind Speicherkarten, Festplatten, usw.

Programmable Read Only Memory

Die einfachste Variante von einem Permanentpeicher ist PROM – programmable read only memory (Abbildung 25). Dieser Speicher kann nur einmal geschrieben, bzw. programmiert werden.

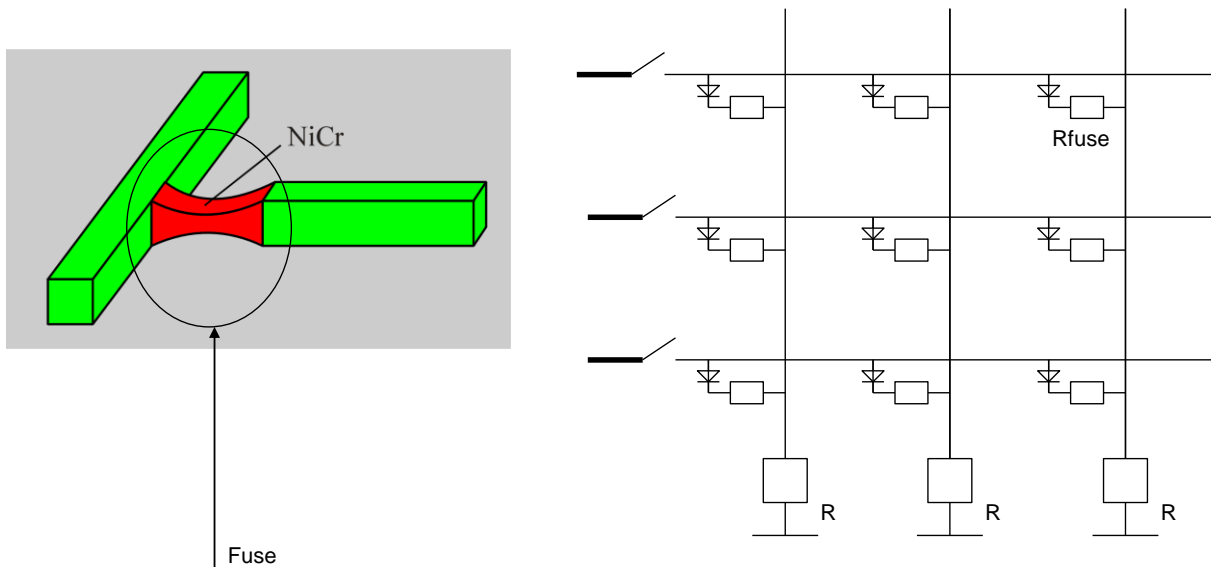


Abbildung 25: Programmable ROM

PROM basiert auf Metallverbindungen (fuses) die durch gezielte Anwendung von hoher Spannung verdampft werden. Solche Programmierung ist auf Abbildung 26 zu sehen.

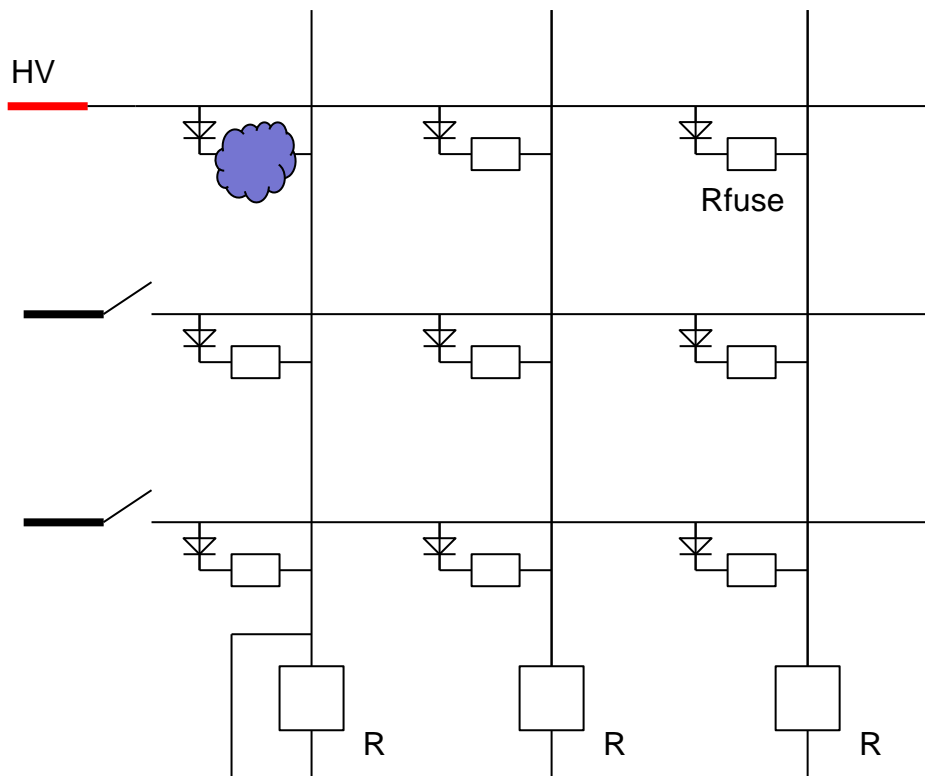


Abbildung 26: Programmierung vom PROM

Widerstände R (pull down Widerstände) sind deutlich höher als die fuse-Widerstände R_{fuse} .

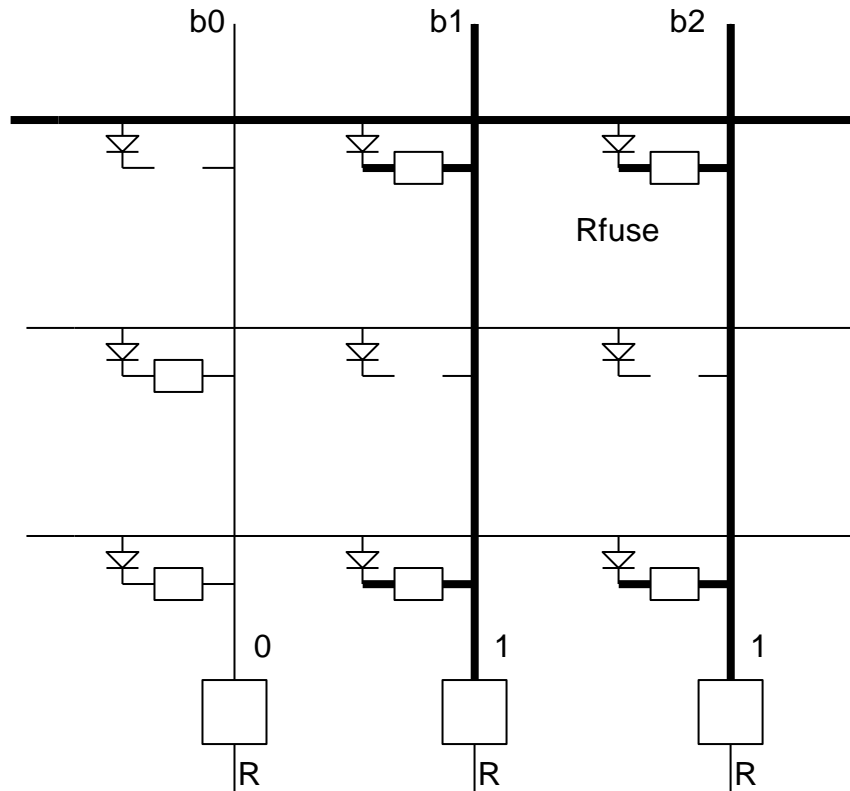


Abbildung 27: Lesevorgang

Abbildung 27 illustriert Lesezyklus. Dioden verhindern, dass ein Strom von den Bitlinien mit hohem Potential (b1 und b2) in die Write- und die Bit-Linie mit niedrigem Potential fließt.

Electrically Erasable PROM

Eine modernere Speichertechnologie ist EEPROM (electrically erasable and programmable read only memory). Die EEPROM Zellen können elektrisch programmiert und gelöscht werden. Eigentlich ist der Name „read only memory“ in dem Fall nicht ganz richtig, da EEPROM auch geschrieben werden kann. Das Schreiben ist aber nur mit höheren Spannungen möglich, nicht so schnell wie das Lesen und nur für eine begrenzte Zahl von Zyklen möglich. Wegen diesen Einschränkungen beim Schreiben kann man EEPROM auch als read only memory bezeichnen.

EEPROM basiert auf Transistoren mit variabler Schwellenspannung als Speicherelementen. Abbildung 28 zeigt wie ein EEPROM gelesen wird.

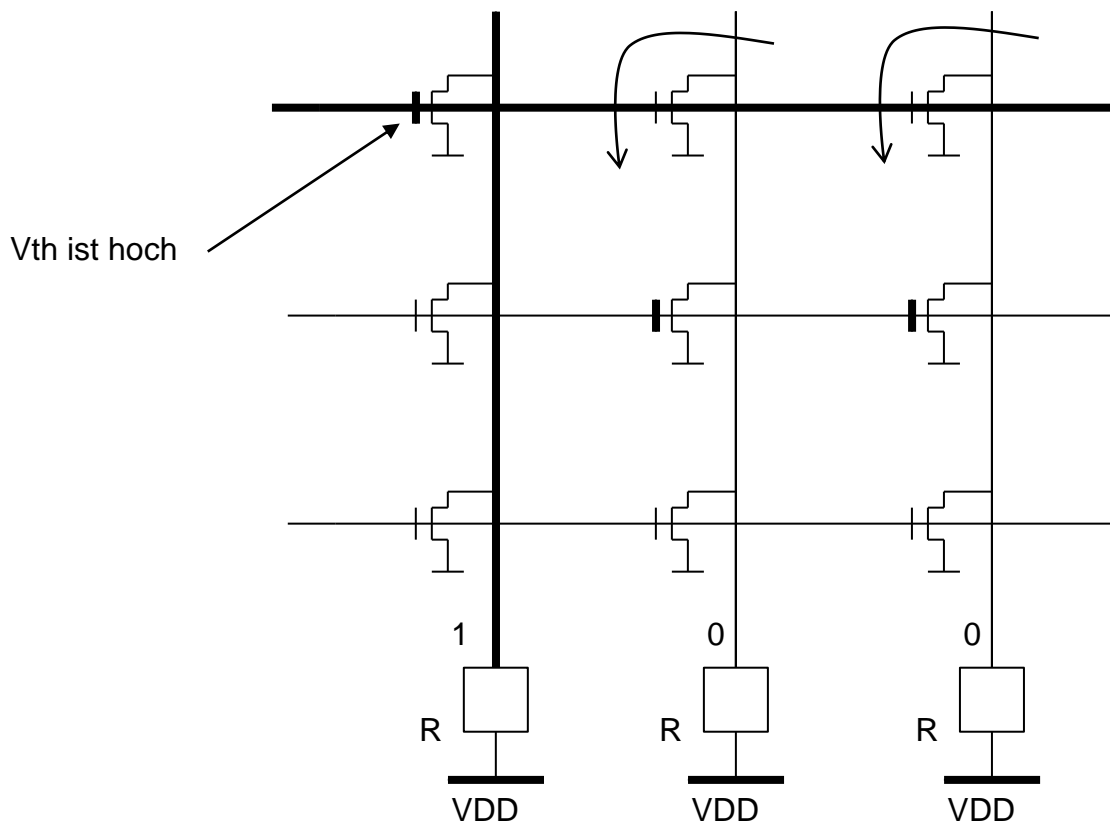


Abbildung 28: Lesen vom EEPROM

Transistoren mit „dickem“ Gate haben höhere Schwelle. Read-Linie der ersten Zeile wird auf 1 geschaltet. Transistoren mit niedrigerer Schwelle leiten und erzeugen logisch 0 auf den Bit-Linien. Ein leitender Transistor hat niedrigeren Widerstand als der pullup-Widerstand R . Der Transistor mit hoher Schwelle leitet nicht. Der pullup-Widerstand R erzeugt 1 auf der Bit-Linie.

Wie kann die Schwelle des Transistors verändert werden? Als Erinnerung: Transistorschwelle ist die Gate-Source Spannung, die benötigt wird, um einen leitenden Kanal zwischen den Source und Drain zu erzeugen. Die EEPROM Transistoren haben ein zweites Gate (Floating-Gate) zwischen dem Haupt-Gate und dem Silizium-Substrat (Abbildung 29). Floating-Gate ist an keine Metalleitung angeschlossen. Trotzdem kann es aufgeladen werden.

Um den Einfluss des Floating Gates auf die Schwelle zu erklären, nehmen wir an, dass SiO_2 Schicht zwischen dem floating Gate und Silizium sehr dünn ist. Nehmen wir auch an, dass der Kanal im Transistor bereits existiert.

Wenn das Floating-Gate negativ aufgeladen ist (Ladung $-Q_{FG}$), entsteht im Kanalbereich zusätzliche positive Ladung Q_{FG} . (Floating-Gate und Kanal bilden einen Plattenkondensator) Die positive Ladung im NMOS-Kanal verringert die effektive Kanalladung (das der NMOS Kanal negativ ist) und vergrößert die Schwelle des NMOS Transistors da man eine zusätzliche Spannung am Gate $V_G = Q/C_{ox}$ braucht, um die positive Ladung zu entfernen.

Es entsteht zwischen dem Floating-Gate und dem Kanalbereich auch eine negative Spannung (Minus am Floating-Gate) $V_{FG} = Q/C_{FG}$. Diese negative Spannung am Floating-Gate verringert die Schwelle ein bisschen, aber da die Kapazität C_{FB} größer als C_{ox} ist, kann man diesen Effekt vernachlässigen.

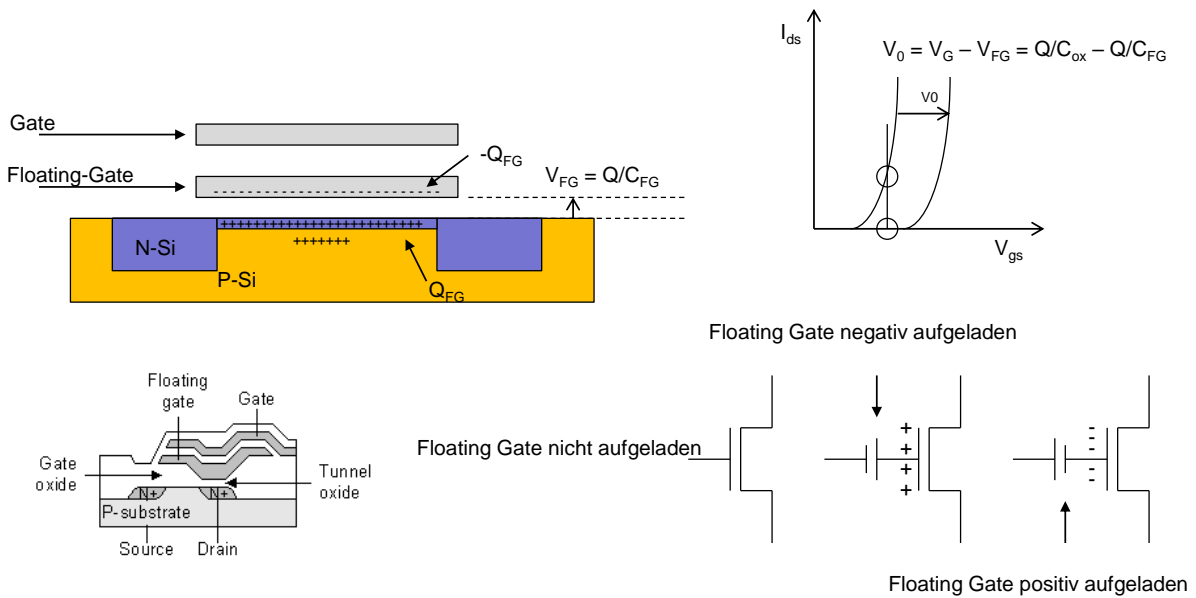


Abbildung 29: Floating Gate

Wie kann man Floating-Gate aufladen? Eine Möglichkeit basiert auf dem quantenmechanischen Tunneleffekt. Abbildung 30 zeigt die EEPROM-Struktur - einen Transistor mit Floating-Gate. Für einen Ausschnitt aus der Struktur zeigen wir das Energiediagramm. Beachten wir, dass sich Floating-Gate über Drain erstreckt. Das Energiediagramm links ist für Spannung $V_{bias} = 0$, und rechts für V_{bias} etwa 10V. Im Fall ohne Biasspannung ist die Potentialbarriere für Elektronen zwischen dem Drain und dem Floating-Gate zu breit und Elektronen können durch die Barriere nicht „tunneln“. Sie können die Barriere auch nicht überwinden da sie zu wenig Energie haben.

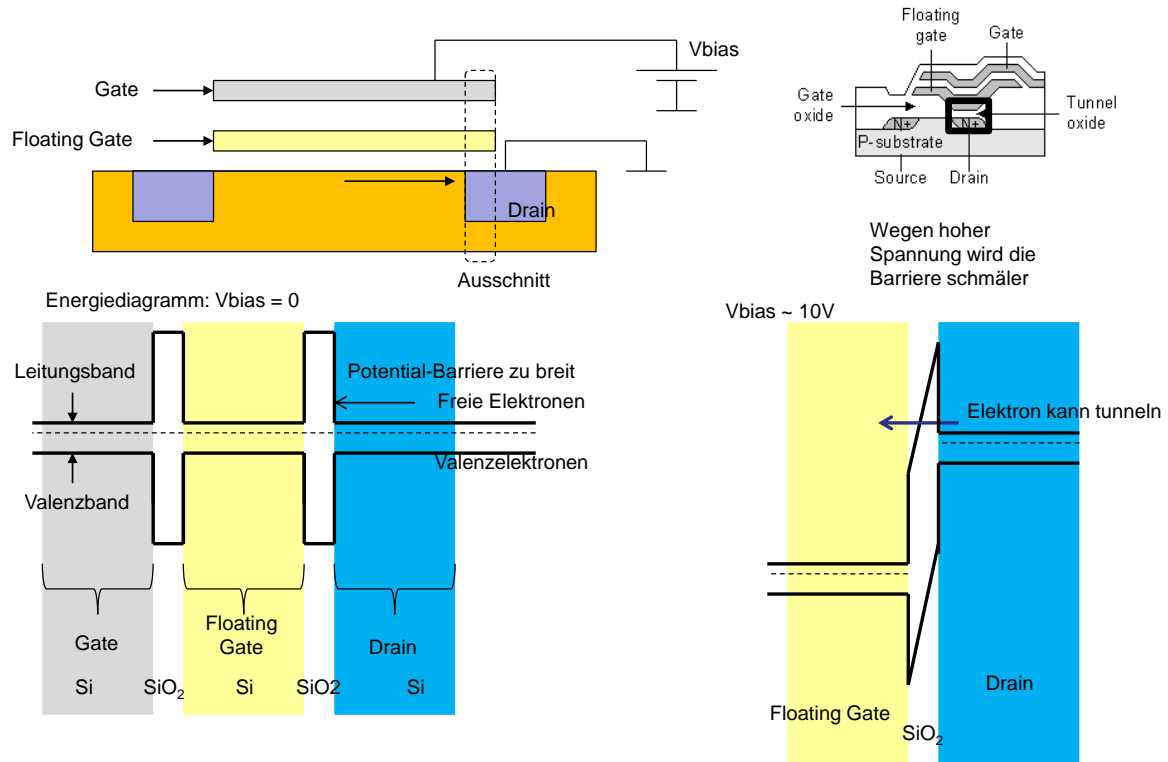


Abbildung 30: Negative Aufladung vom floating Gate mithilfe von Feldemission

Wenn man die Biasspannung einschaltet, verschieben sich die Energieniveaus wie in Abbildung 30 rechts gezeigt wurde. Dadurch wird die Barriere für Elektronen aus dem Drain schmaler und sie können durch die Barriere in das Floating-Gate gelangen, obwohl sie kleinere kinetische Energie als die hohe der Barriere besitzen (Austrittsarbeit). Die Elektronen erzeugen eine negative Ladung im Floating-Gate.

Der Effekt wo die Elektronen durch ein starkes E-Feld aus einer Kathode (Silizium) gelöst werden, obwohl sie kleinere Energie als die Austrittsarbeit besitzen nennt man Feldemission oder Fowler-Nordheim-Tunneln. [Feldemission – Wikipedia](#)

Man kann den Tunneleffekt (Feldemission) auch zunutze machen, um die negative Ladung aus dem Floating-Gate zu entfernen. Dafür würde man die Polarität der Biasspannung umdrehen, bzw. den Drain auf eine positive Spannung legen. In dem Fall können die Elektronen aus dem Leitungsband des Floating-Gates in den Drain tunneln. Damit wird die negative Ladung entfernt.

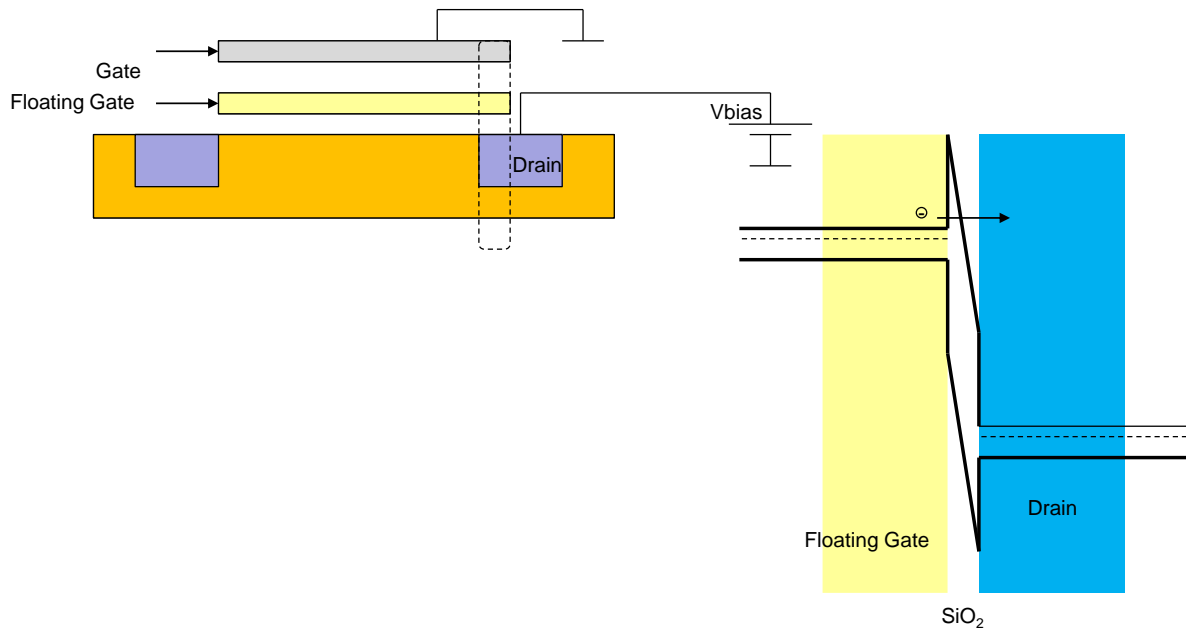


Abbildung 31: Negative Aufladung vom floating Gate mithilfe von Feldemission

Man kann auch die so genannte „hot carrier injection“ benutzen um das Floating-gate negativ aufzuladen. Dabei ist es wichtig eine hohe Stromdichte im Kanalbereich zu erreichen. Transistor wird mit $V_{gs} > \text{Schwelle}$ eingeschaltet. Eine V_{ds} -Spannung erzeugt den Strom. Durch die hohe Stromdichte im Kanalbereich nahe Drain, bekommen die Elektronen hohe kinetische Energie und überwinden die Barriere.

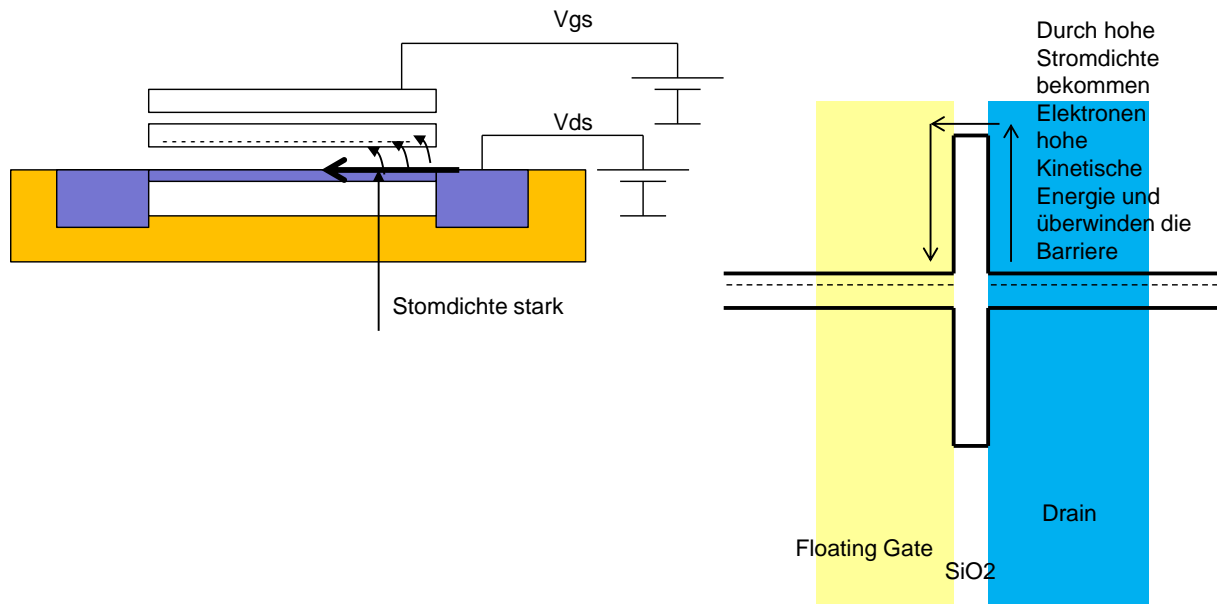


Abbildung 32: Hot carrier injection

Klassisch betrachtet ist es für ein Teilchen mit einer bestimmten mittleren thermischen Energie, die kleiner ist als die Höhe der Austrittsarbeit, unmöglich, das Kathodenmaterial zu verlassen. Quantenmechanisch betrachtet gibt es jedoch eine bestimmte Wahrscheinlichkeit, dass einzelne

Elektronen aus dem Festkörper austreten. Diesen Effekt nennt man allgemein auch Tunneleffekt. Das Elektron tunnelt durch den Potentialwall, der durch das äußere elektrische Feld verkippt wurde – diese spezielle Art von Tunneln nennt man auch „Fowler-Nordheim-Tunneln“ (benannt nach Ralph Howard Fowler und Lothar Nordheim).

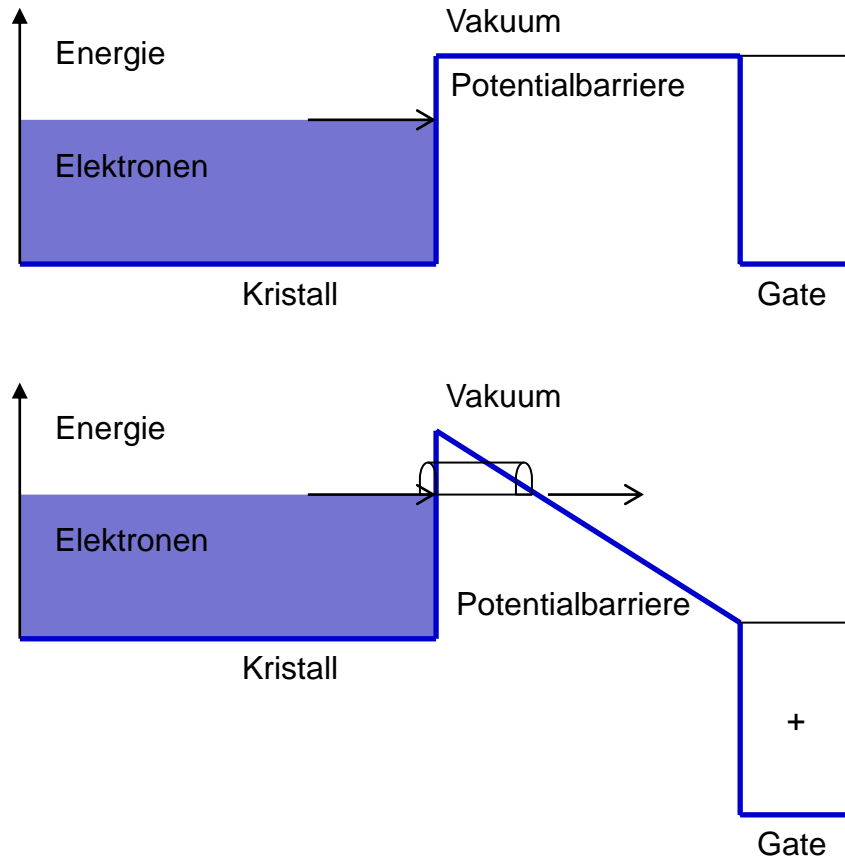


Abbildung 33: Feldemission - Illustration

Je nachdem wie die Ladung im Floating-Gate erzeugt oder gelöscht wird unterscheidet man zwischen einigen EEPROM Varianten. Wir zeigen hier eine Variante welche nur auf Tunneleffekt basiert.

Abbildung 34 zeigt den Löschvorgang: Gates werden auf hohe Spannung (~10V) gelegt. Elektronen tunneln aus den Source-Bereichen in das Floating-Gate. Auf diese Weise werden alle Floating-Gates gleichzeitig negativ geladen. Die Schalter-Transistoren werden beim Löschen nicht benutzt. Dünne Linien befinden sich auf niedrigen Potential.

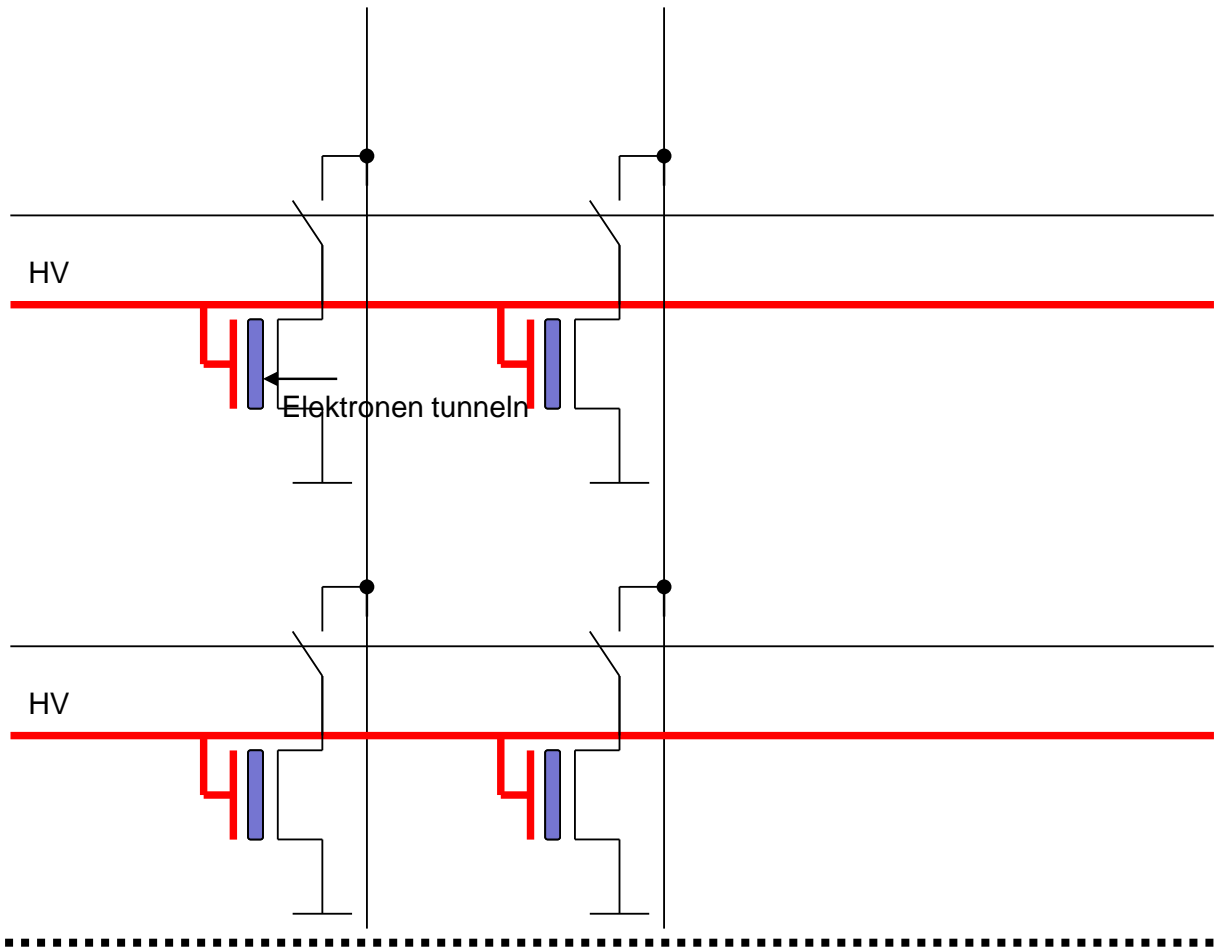


Abbildung 34: Löschen vom EEPROM

Abbildung 35 zeigt das Programmieren vom EEPROM. Es basiert ebenfalls auf dem Tunneleffekt. Beim Programmieren ist es wichtig, dass man einzelne Transistoren individuell programmieren kann, bzw. dass man die Ladungen an ihren Floating-Gates entfernen kann. Die Ladung wird nur dann entfernt, wenn die EEPROM Zelle hohe Spannung an der Programmlinie und an der Bit-Linie hat. Das ist der Fall für die eingekreiste Zelle. Die Elektronen tunneln vom Floating-Gate in den Drain.

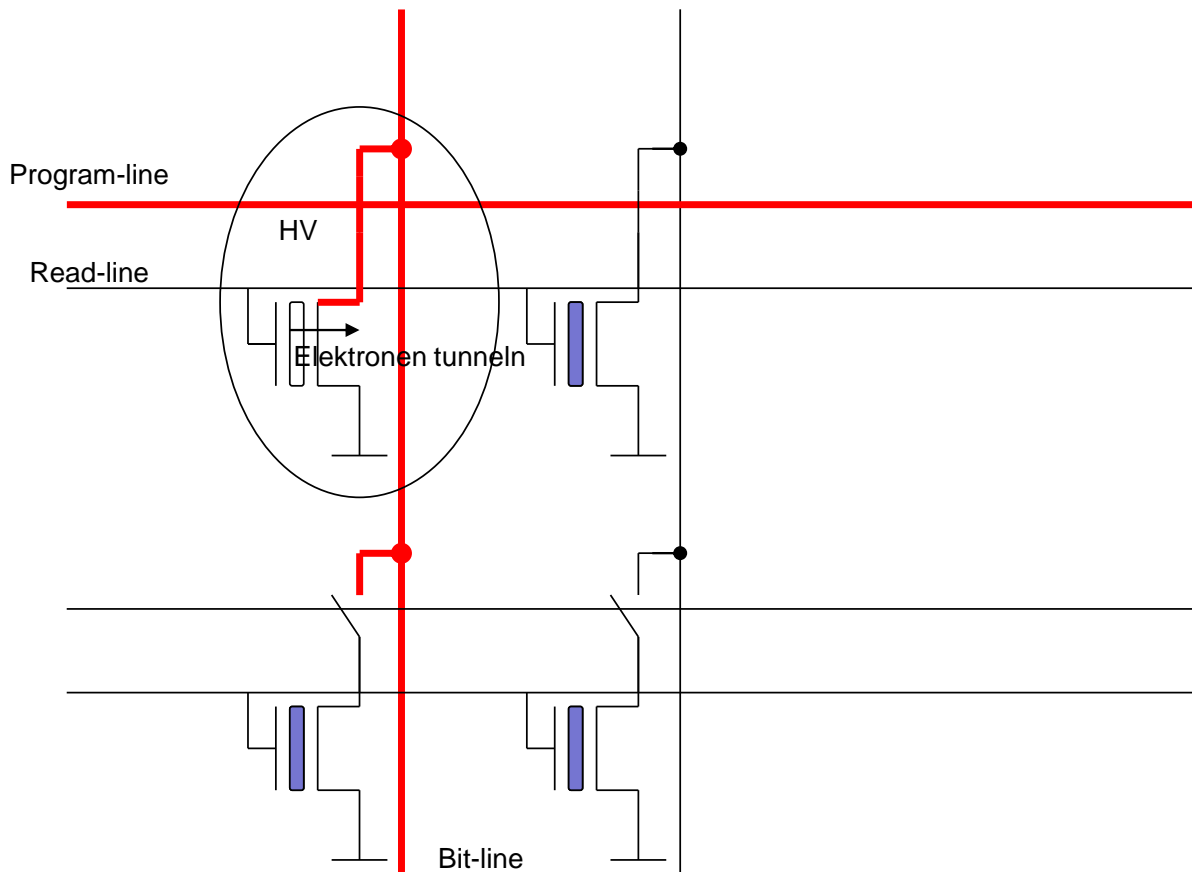


Abbildung 35: Programmieren (Schreiben) vom EEPROM

FLASH Speicher

Ein FLASH Speicher basiert auf dem gleichen Prinzip wie EEPROM. Der Hauptunterschied ist die Logik-Art.

EEPROMs verwenden Transistoren die NOR bilden – sie sind in parallel geschaltet. FLASH Speicher verwenden NAND Gates, die aus Transistoren mit Floating-Gate aufgebaut sind.

Die Zellen, die als NOR geschaltet sind, lassen sich kleiner machen da mehrere Speichertransistoren einen Read-Schalter teilen.

Folgender Text fasst die Eigenschaften von FLASH Speicher gut zusammen:

[Difference Between EEPROM and Flash | Difference Between](#)

Flash is a very popular term when it comes to storage media as it is used by portable devices like phones, tablets, and media players. Flash actually is an offspring of EEPROM, which stands for Electrically Erasable Programmable Read-Only Memory. The main difference between EEPROM and Flash is the type of logic gates that they use. While EEPROM uses the faster NOR (a combination of Not and OR), Flash uses the slower NAND (Not and AND) type. The

NOR type is a lot faster than the NAND type but there is the matter of affordability as the former is significantly more expensive than the NAND type.

Another advantage of EEPROM over Flash is in how you can access and erase the stored data. EEPROM can access and erase the data byte-wise or a byte at a time. In comparison, Flash can only do so block-wise. In order to simplify the whole thing, individual bytes are grouped into a smaller number of blocks, which can have thousands of bytes in each block. This is a bit problematic when you only want to read or write to a single byte at a time; which is what's typically needed in executing the code of a program. This is a reason why Flash cannot be used in electronic circuits that require byte-wise access to data. Data in Flash can also be executed, but it needs to be read as a whole and loaded into RAM beforehand.

EEPROM was designed to be read a lot more than it is written. This is in-line with programming for electronic circuits where you write to the chip a number of times while testing the program. Then, it is stored for good, only to be read every time the data is needed. This is not very suitable for storage media where data is routinely written and read.

In typical use, Flash is used mainly to refer to storage media and can range anywhere from a GB to hundreds of GB. In contrast, EEPROM is usually reserved for permanent code storage in electronic chips. Typical values range from kilobytes to a couple of megabytes

Summary:

1. Flash is just one type of EEPROM
2. Flash uses NAND type memory while EEPROM uses NOR type
3. Flash is block-wise erasable while EEPROM is byte-wise erasable
4. Flash is constantly rewritten while other EEPROMs are seldom rewritten
5. Flash is when large amounts are needed while EEPROM is used when only small amounts are needed

Ebenfalls folgender Text bringt viele interessante Infos:

[Flash-Speicher – Wikipedia](#)

Ein Flash-Speicher besteht aus einer bestimmten, von der Speichergröße abhängigen Anzahl einzelner Speicherelemente. Die Bytes oder Worte (typisch durchaus bis 64 Bit) können einzeln adressiert werden. Dabei können sie in einigen Architekturen auch einzeln geschrieben werden, wogegen bei anderen nur größere Datenmengen auf einmal programmiert werden können. In der Regel ist die entgegengesetzte Operation, das Löschen, aber nur in größeren Einheiten, sogenannten Sektoren (meistens ein Viertel, Achtel, Sechzehntel usw. der Gesamtspeicherkapazität) möglich.

Flash-Speicher haben eine begrenzte Lebensdauer, die in einer maximalen Anzahl an Löschkzyklen angegeben wird (10.000 bis 100.000 Zyklen für NOR-Flash und bis zu zwei Millionen für NAND-Flash). Dies entspricht gleichzeitig der maximalen Anzahl Schreibzyklen, da der Speicher jeweils blockweise gelöscht werden muss, bevor er wieder beschrieben werden kann. Diese Zyklenzahl wird Endurance (Beständigkeit) genannt. Verantwortlich für diese begrenzte Lebensdauer ist das Auftreten von Schäden in der Oxidschicht im Bereich des Floating-Gates, was das Abfließen der Ladung bewirkt.[3]

Eine andere wichtige Kenngröße ist die Zeit der fehlerfreien Datenhaltung, die Retention. Ein weiterer Nachteil ist, dass der Schreibzugriff bei Flash-Speicher erheblich langsamer erfolgt als der Lesezugriff. Zusätzliche Verzögerungen können dadurch entstehen, dass immer nur ganze Blöcke gelöscht werden können.

Der Flash-Speicher speichert seine Informationen auf dem Floating-Gate. Bei einem Löschkzyklus durchtunneln die Elektronen die Oxidschicht. Dafür sind hohe Spannungen erforderlich. Dadurch wird bei jedem Löschkvorgang die Oxidschicht, die das Floating-Gate umgibt, ein klein wenig beschädigt (Degeneration). Irgendwann ist die Isolation durch die Oxidschicht nicht mehr gegeben, die Elektronen bleiben nicht mehr auf dem Floating-Gate gefangen, und die auf der Speicherzelle gespeicherte Information geht verloren. Der Defekt einer einzelnen Zelle macht einen Flash-Speicher jedoch noch lange nicht unbrauchbar.

Als nichtflüchtiges Speichermedium steht der Flash-Speicher in Konkurrenz vor allem zu Festplatten und optischen Speichern wie DVDs und Blu-ray-Discs.

Ein wesentlicher Vorteil liegt in der mechanischen Robustheit von Flash-Speicher. Demgegenüber sind Festplatten sehr stoßempfindlich (Head-Crash). Häufig ist die Lebensdauer der Steckkontakte (USB-Stecker) der limitierende Faktor.

SD-Karten verwenden zwar die gleichen Flash-Speicherchips wie SSDs, aber die Art und Weise, wie der Speicher verpackt und verwaltet wird, ist ganz anders. Eine SSD verfügt über ein ausgefeilteres Controller-System, das mit den Einschränkungen des Flash-Speichers arbeitet, der nach einer beträchtlichen Anzahl von Schreibvorgängen verschleißt. Eine SSD "verteilt" diese Nutzung, sodass keine einzelne Stelle nacheinander oder übermäßig oft beschrieben wird. Durch die Nivellierung des Verschleißes wird die Lebensdauer einer SSD drastisch verlängert. Viele Laufwerke bieten optional "Trimmen", eine Funktion, bei der das Laufwerk und das Betriebssystem beim Löschen von Dateien Informationen weitergeben, die zur Verbesserung der allgemeinen Schreibgeschwindigkeit beitragen.

Abbildung 36 zeigt die FLASH Struktur (vier 2-bit Worte). Die eingekreisten Zellen werden gelesen.

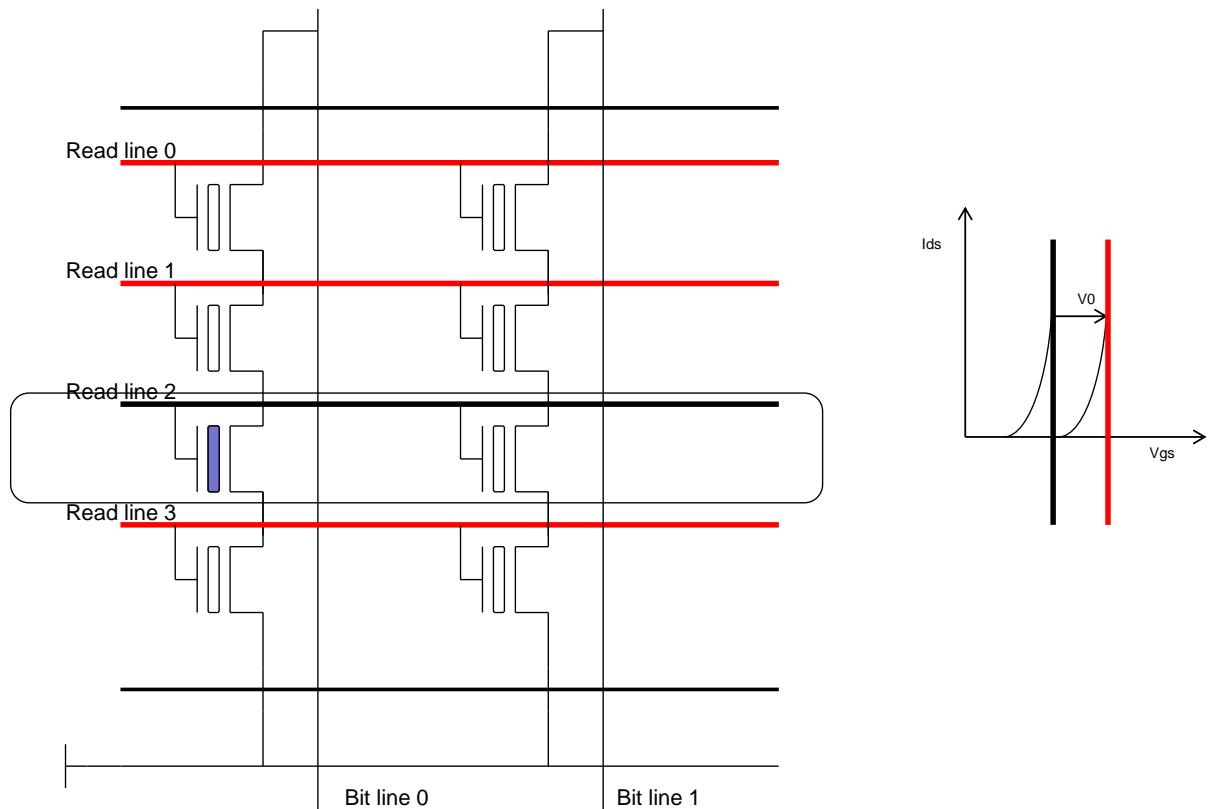


Abbildung 36: FLASH Speicher

Read-Linie 2 wird auf ein Niveau zwischen den Schwellen-Spannungen von Transistoren mit geladenen und nicht geladenen Floating-Gates gelegt. Alle anderen Read-Linien werden auf höheres Potential gelegt (z.B. 5V), so dass die entsprechenden Transistoren immer leiten (unabhängig davon ob ihre Gates aufgeladen sind). Auf diese Weise lässt sich messen, ob die Transistoren (angeschlossen an Read-Linie 2) leiten oder nicht, bzw. welches Bit sie speichern.

Abbildung 37 erklärt den Löschvorgang. Tunneleffekt wird benutzt. Beim Löschen wird positive Spannung von etwa 10V an Transistorsubstrat gelegt. Elektronen tunneln vom Floating-Gate weg. Das Löschen kann auch selektiv gemacht werden, indem man eine leicht höhere Spannung an Read-Linie legt (read-Linie 1). Die Transistoren in der Zeile 1 werden nicht gelöscht da die Spannung zwischen Substrat und Gate nicht ausreichend ist.

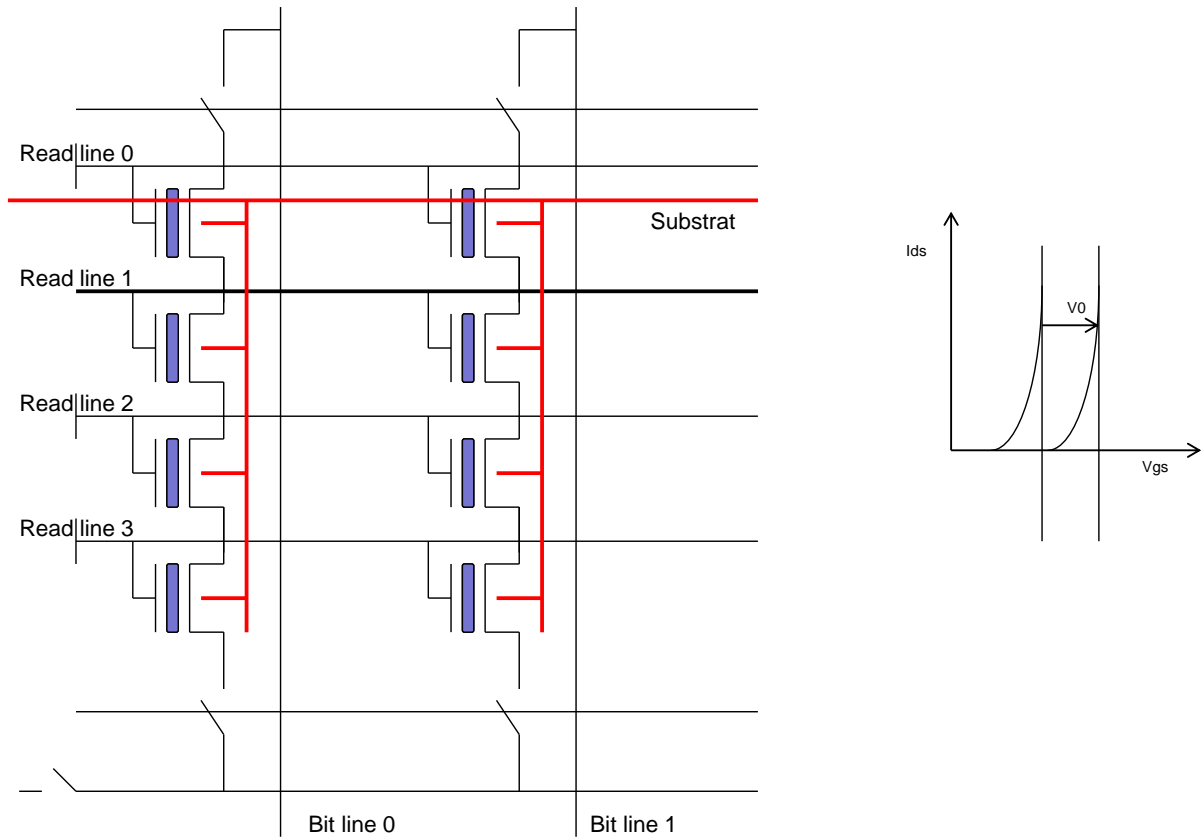


Abbildung 37: FLASH Speicher - Löschivorgang

Abbildung 38 zeigt den Schreibvorgang. Auch hier wird der Tunneleffekt verwendet. Read-Linie 1 wird auf etwa 10V gelegt und die anderen Read-Linien auf 5V. Der eingekreiste Transistor T2 sieht eine hohe Spannung am Gate und 0V am Drain und sein Floating-Gate wird aufgeladen. Transistor T1 sieht zwar ein hohes Potential am Gate aber da sein Drain-Potential höher als 0V liegt, ist die Gate-Drain Spannung unzureichend fürs Tunneln. Deswegen wird sein Floating-Gate nicht aufgeladen.

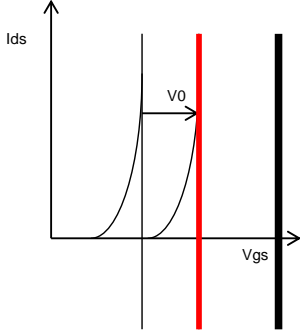
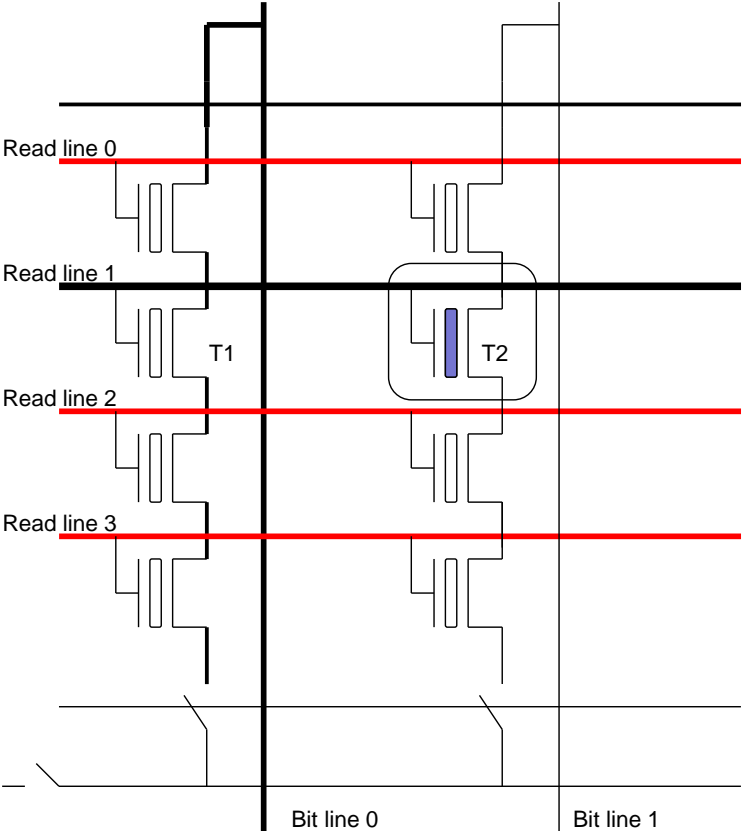


Abbildung 38: FLASH Speicher - Schreibvorgang