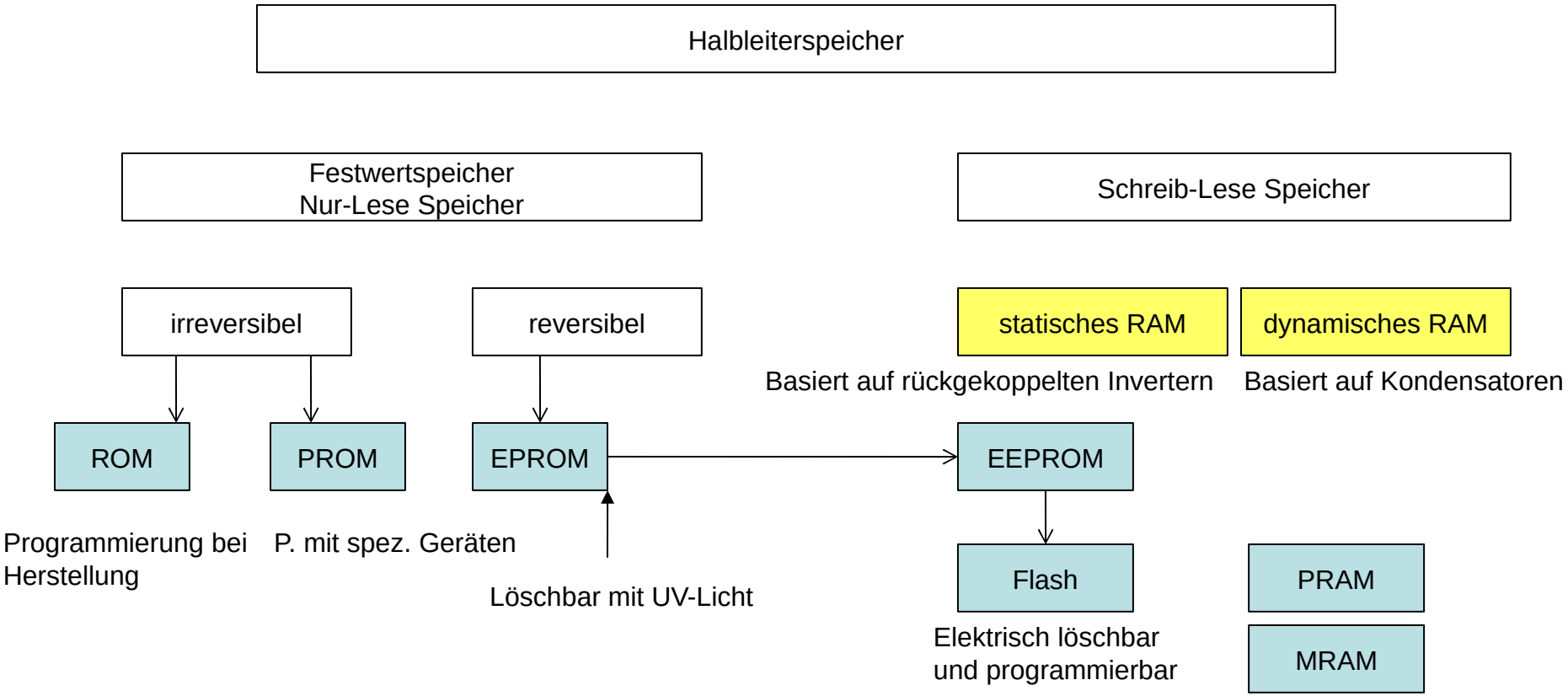
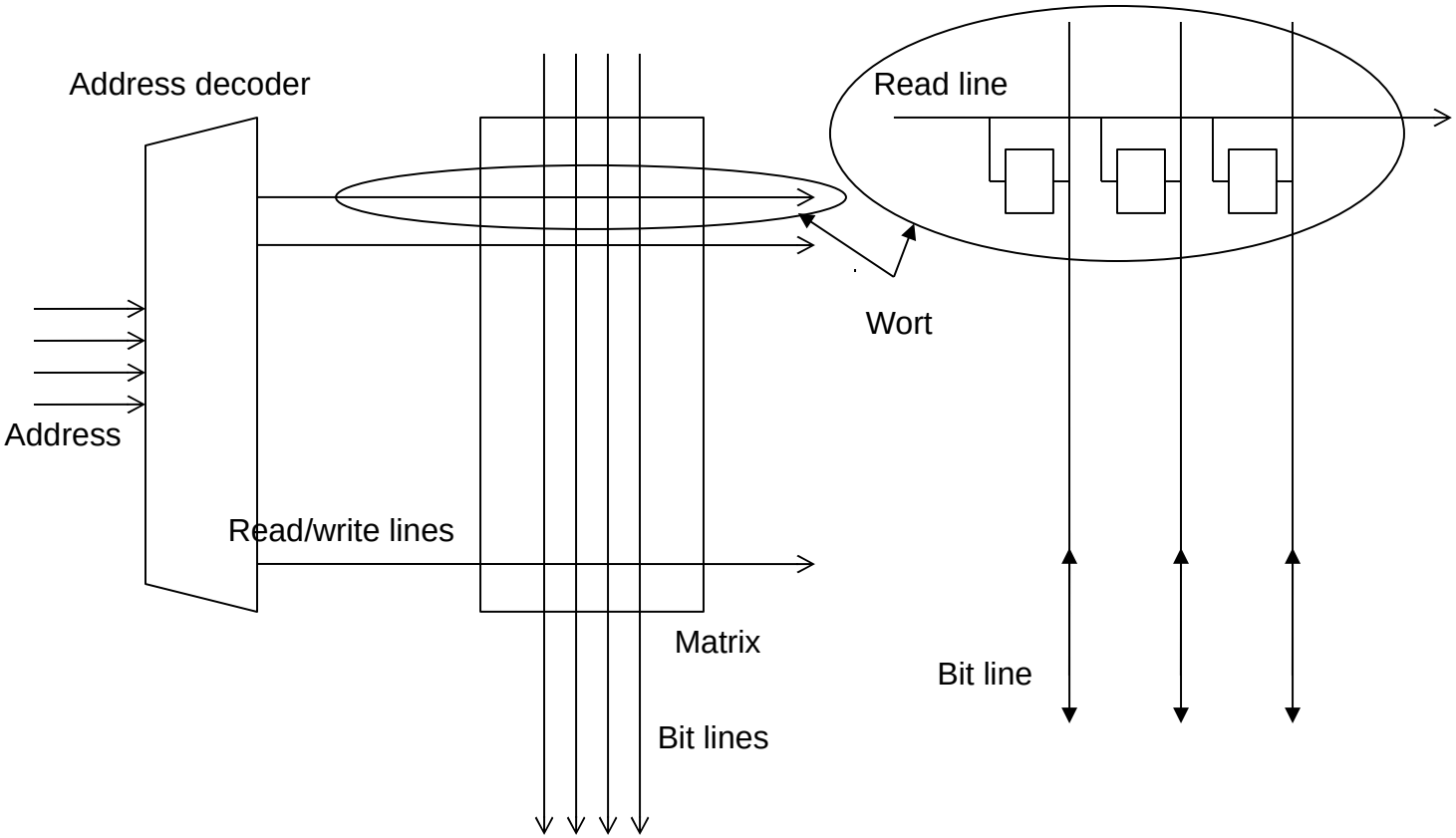
Design digitaler Schaltkreise

Speicherelemente - Klassifizierung

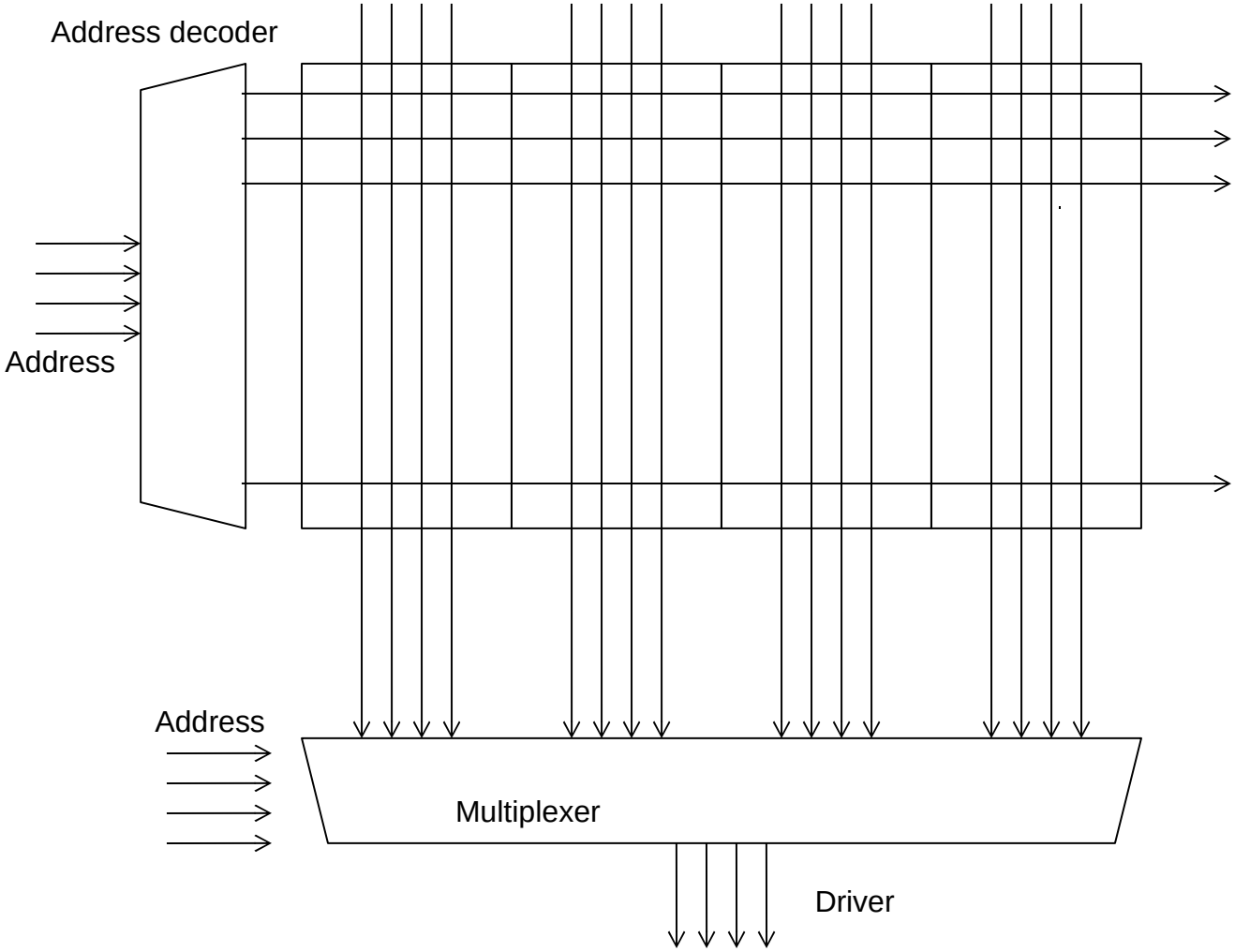


Einfache Struktur



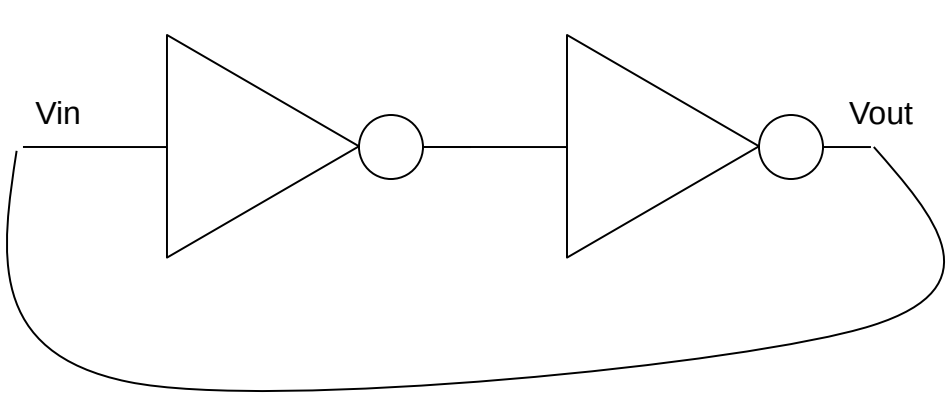
Aspect ratio (too high and too narrow layout) may be a problem!

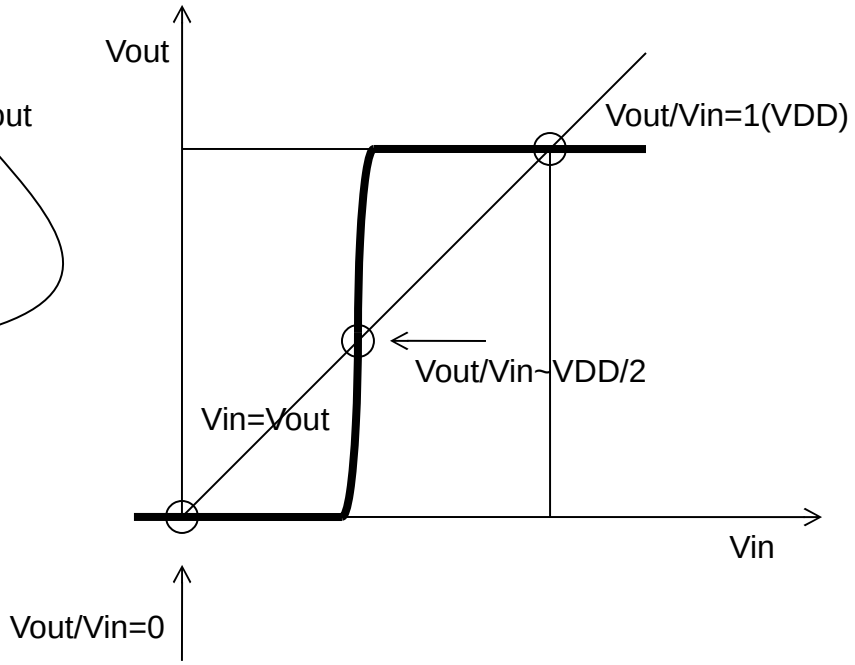
Komplexere Struktur

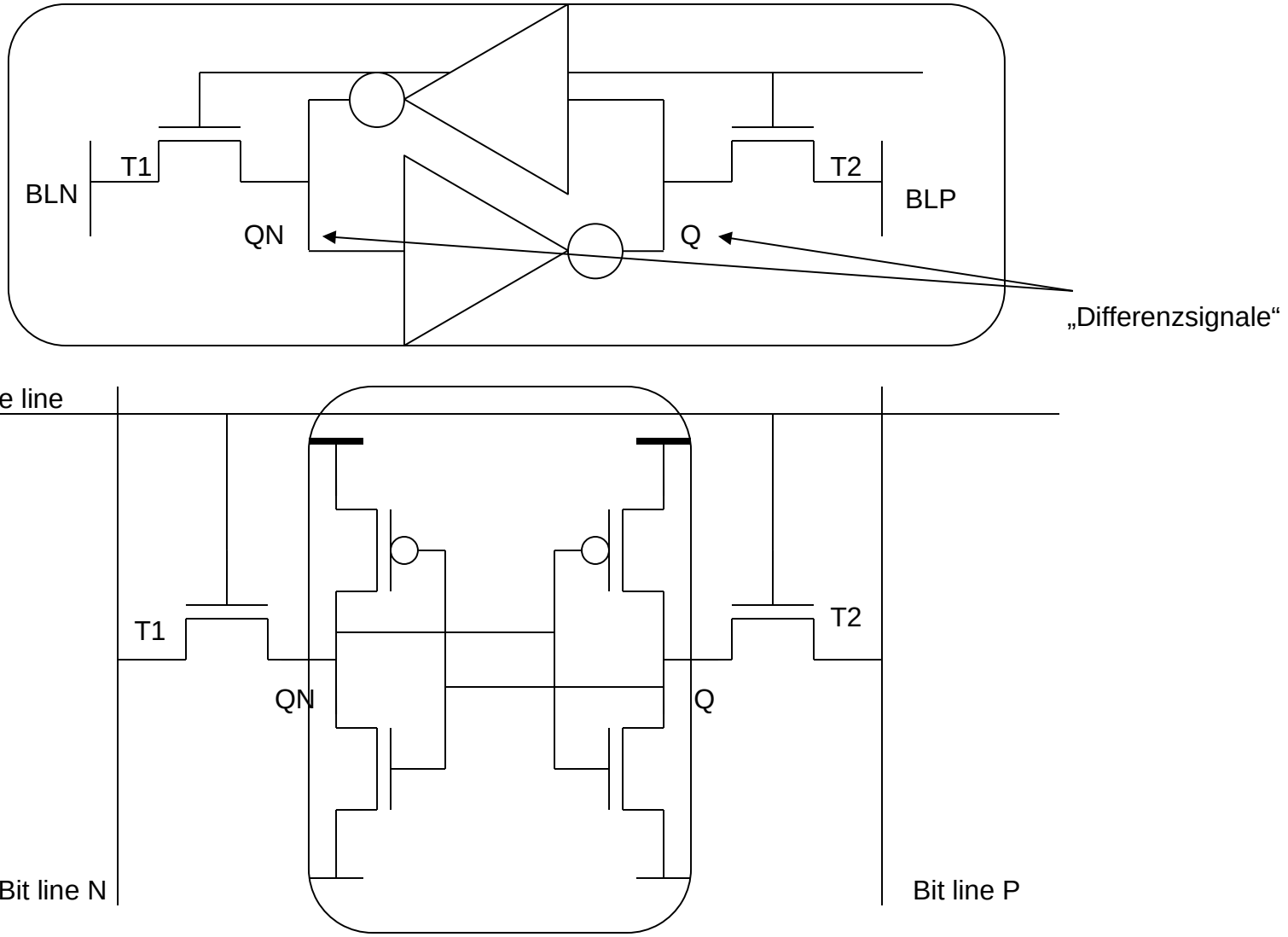


SRAM

- Statische Speicherzellen
- Wenn wir den Ausgang des zweiten Inverters mit dem Eingang des ersten verbinden, gilt Vin = Vout.
- Der Zustand der Schaltung liegt im Schnittpunkt der Kennlinie Vout = f(Vin) und der Gerade Vout = Vin.

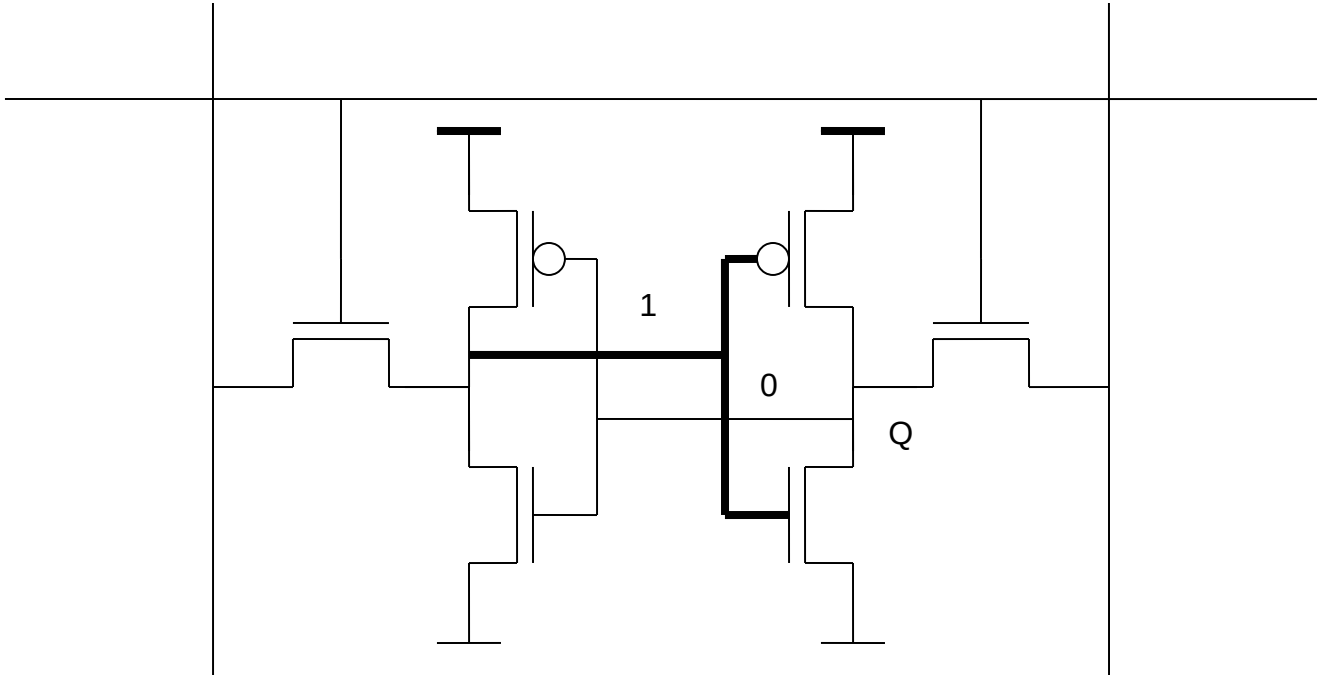




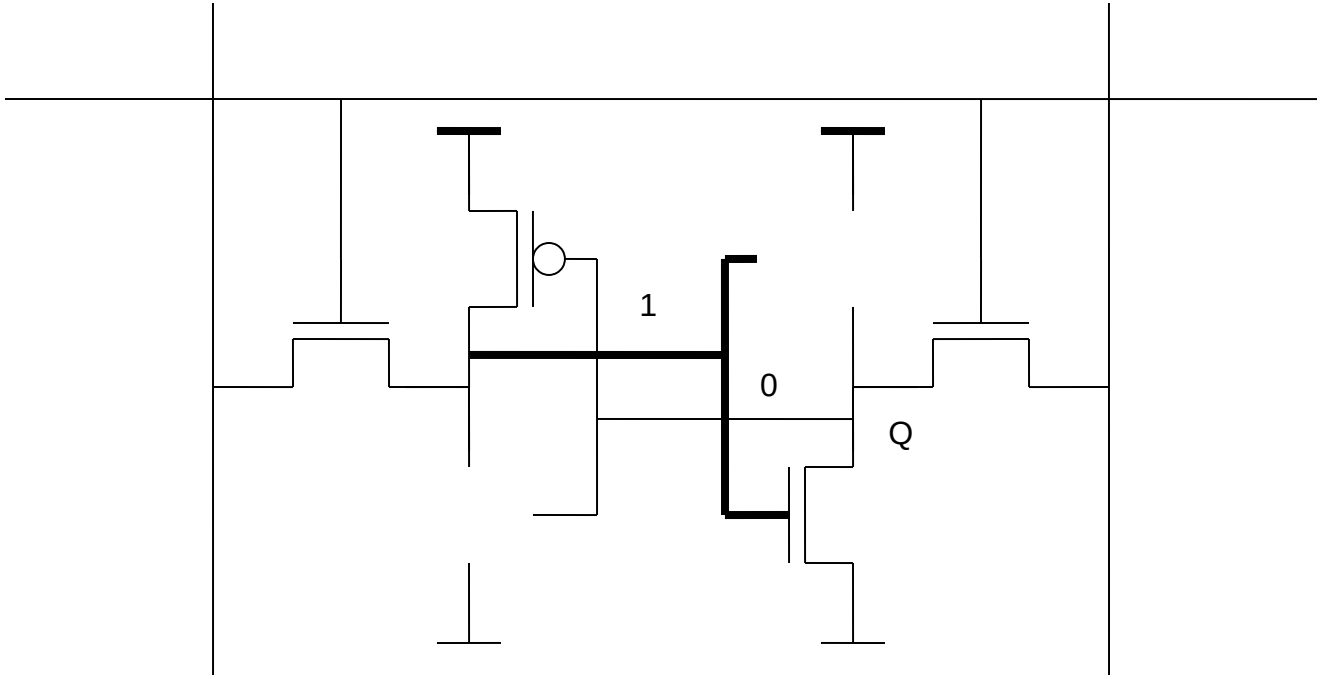


Diese 4 Transistoren speichern den Zustand

- SRAM befindet sich im Zustand Q=0

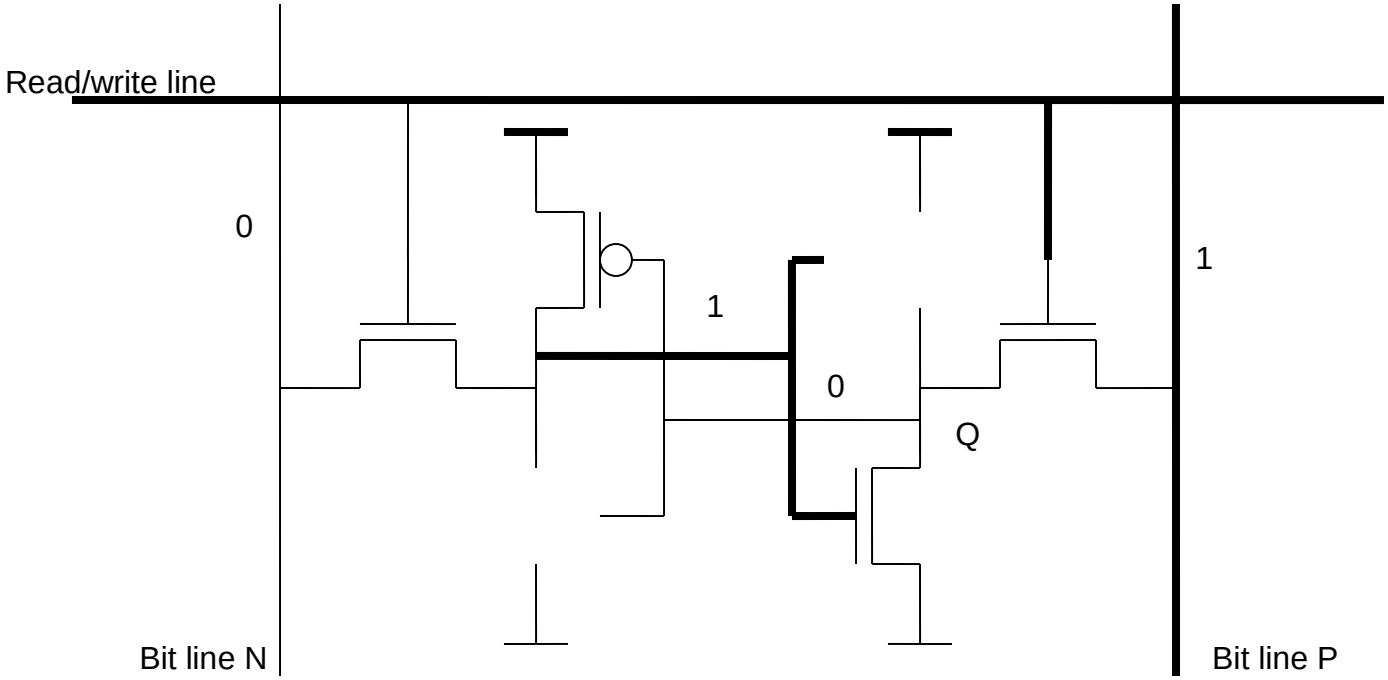


- SRAM befindet sich im Zustand Q=0



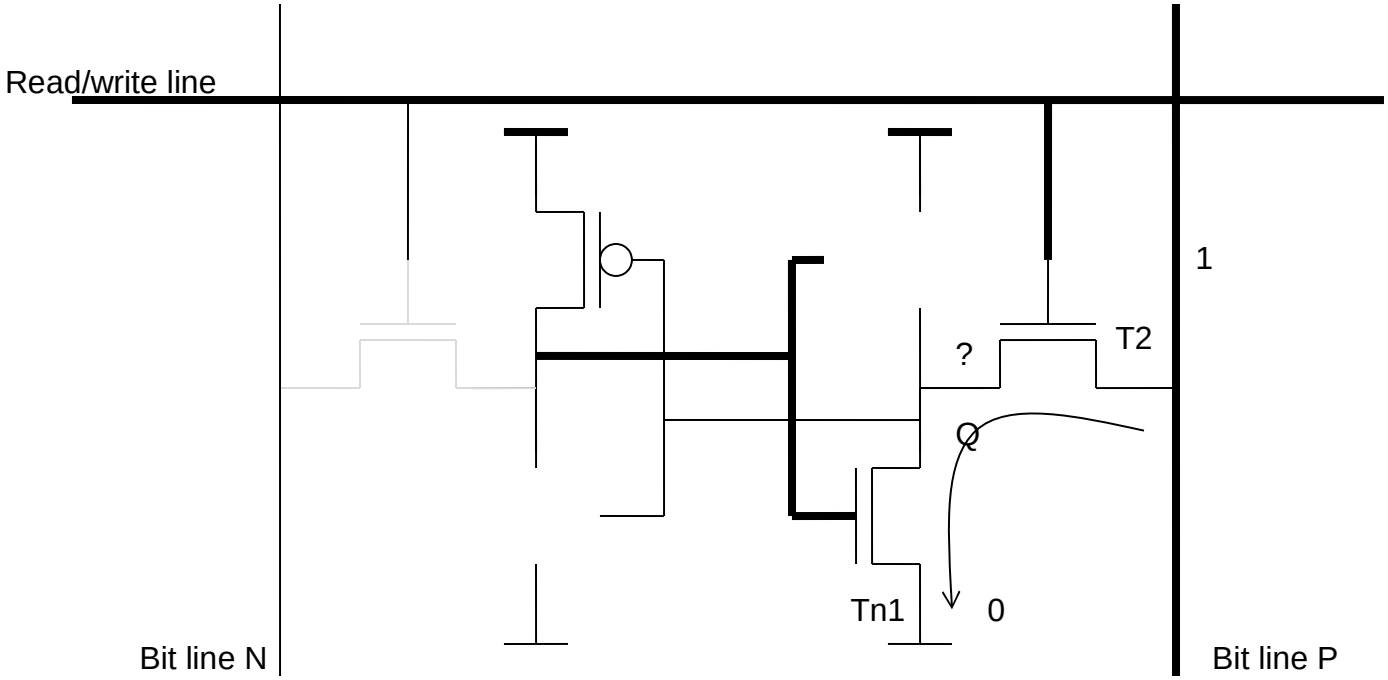
Program

- Versuchen wir die RAM Zelle zu überschreiben…
- Annahme: NMOS leitet starker als PMOS
- Bit line P wird auf 1 aufgeladen, bit line N auf 0 entladen
- Word line (read/write line) wird geschaltet

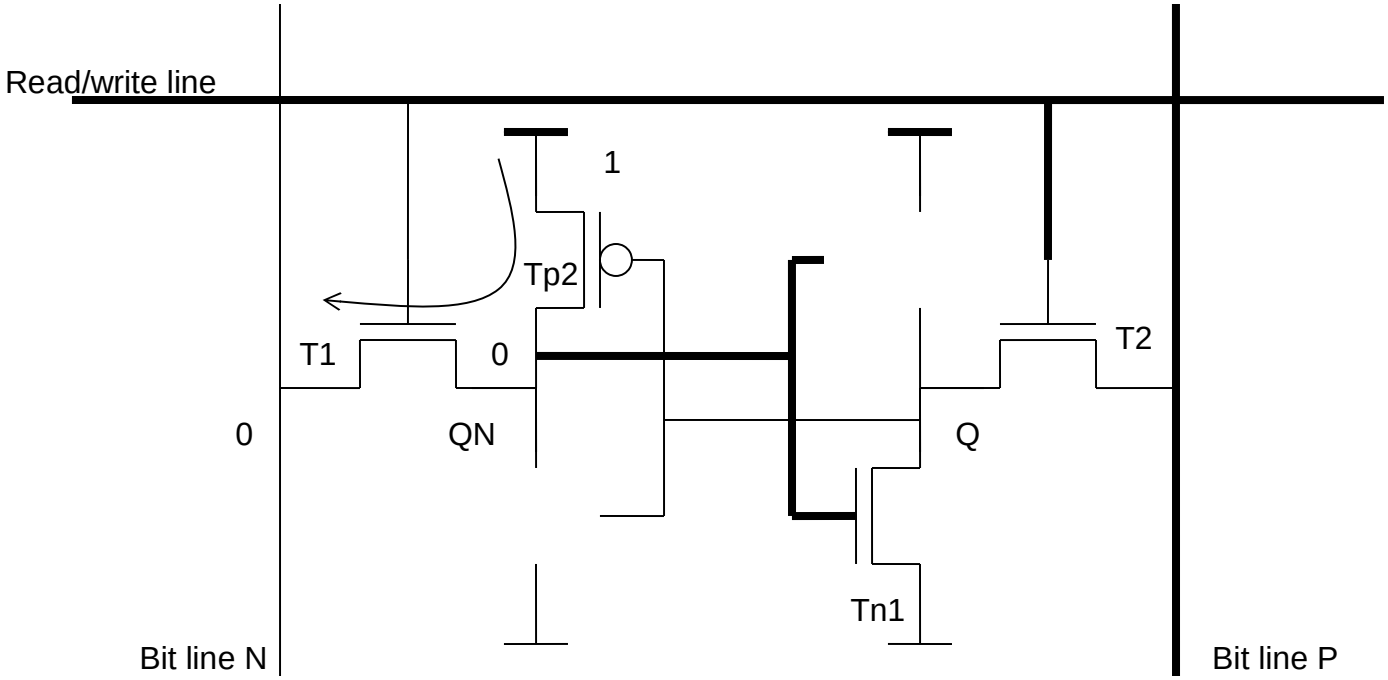


Program

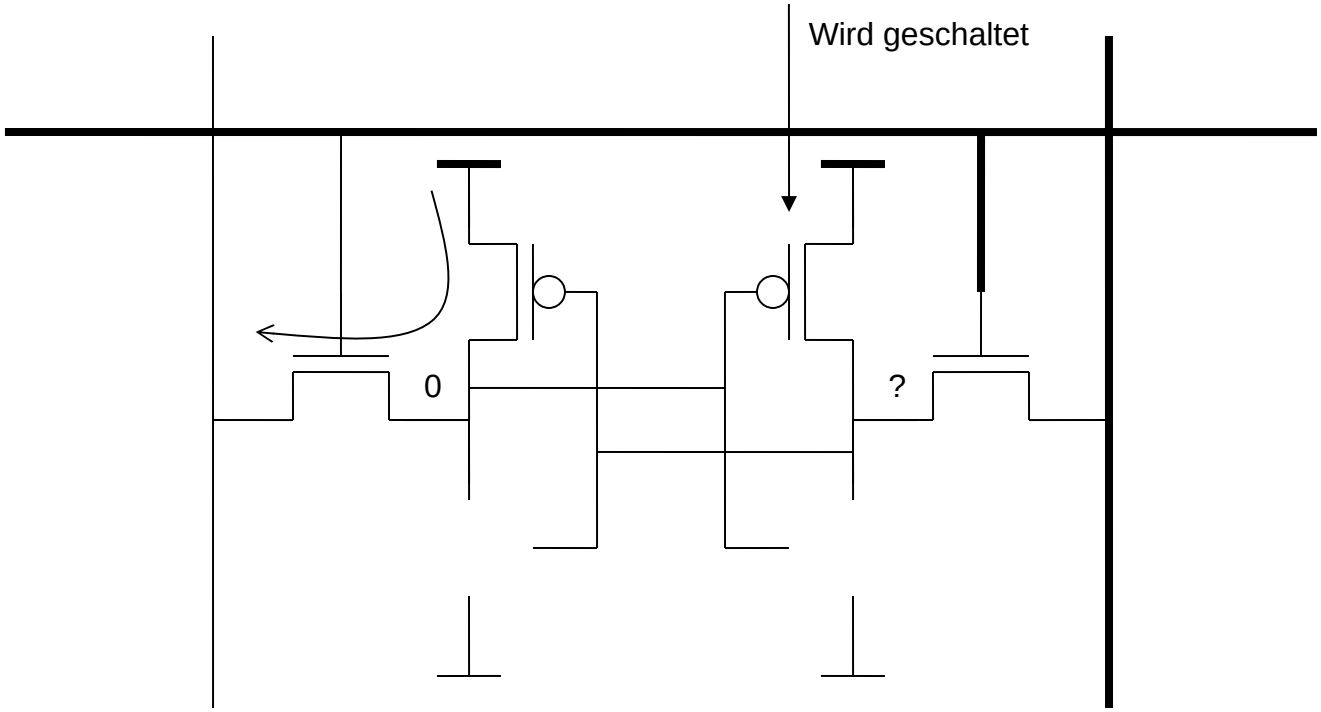
- Frage: Warum gibt es zwei Schalter-Transistoren und zwei bit lines?
- In unserem Fall ist es schwer nur mit der bit line P, RAM zu überschreiben. Transistor T2 leitet zu schwach wegen kleiner Vgs



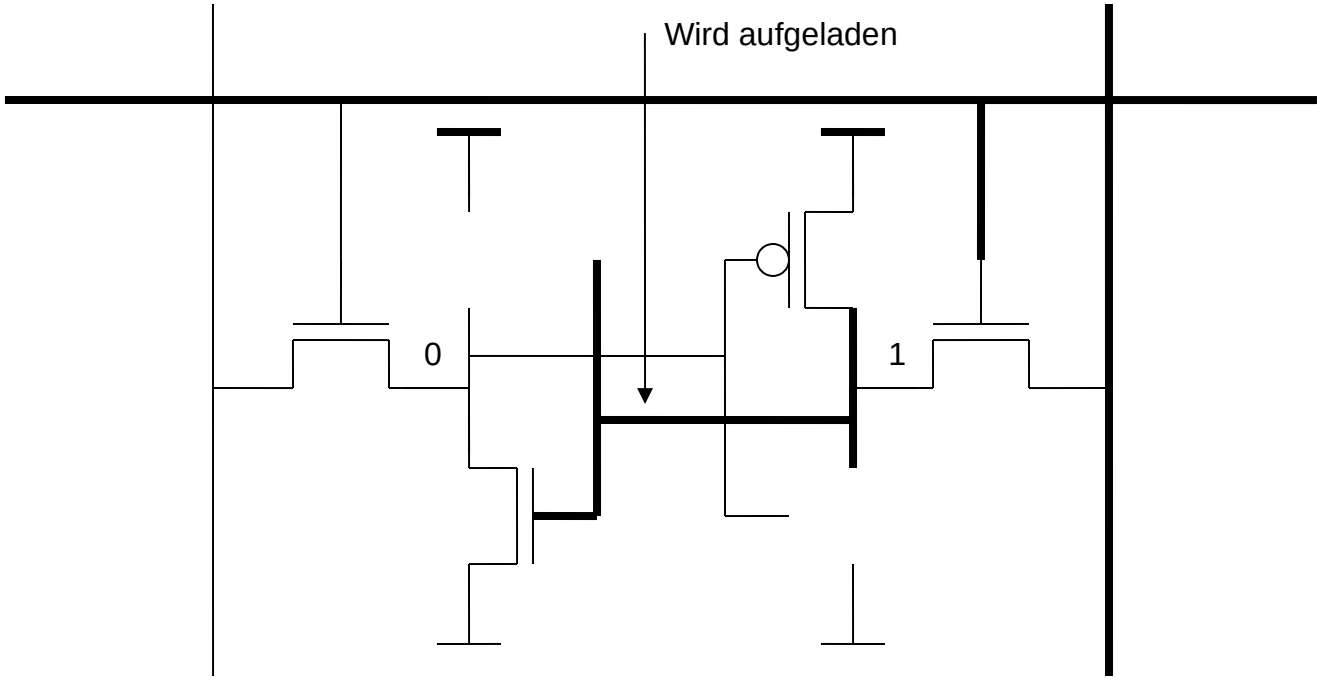
- Transistor T1 leitet gut und entlädt QN gegen 0



- Das führt zum “Umkippen” von RAM Zelle

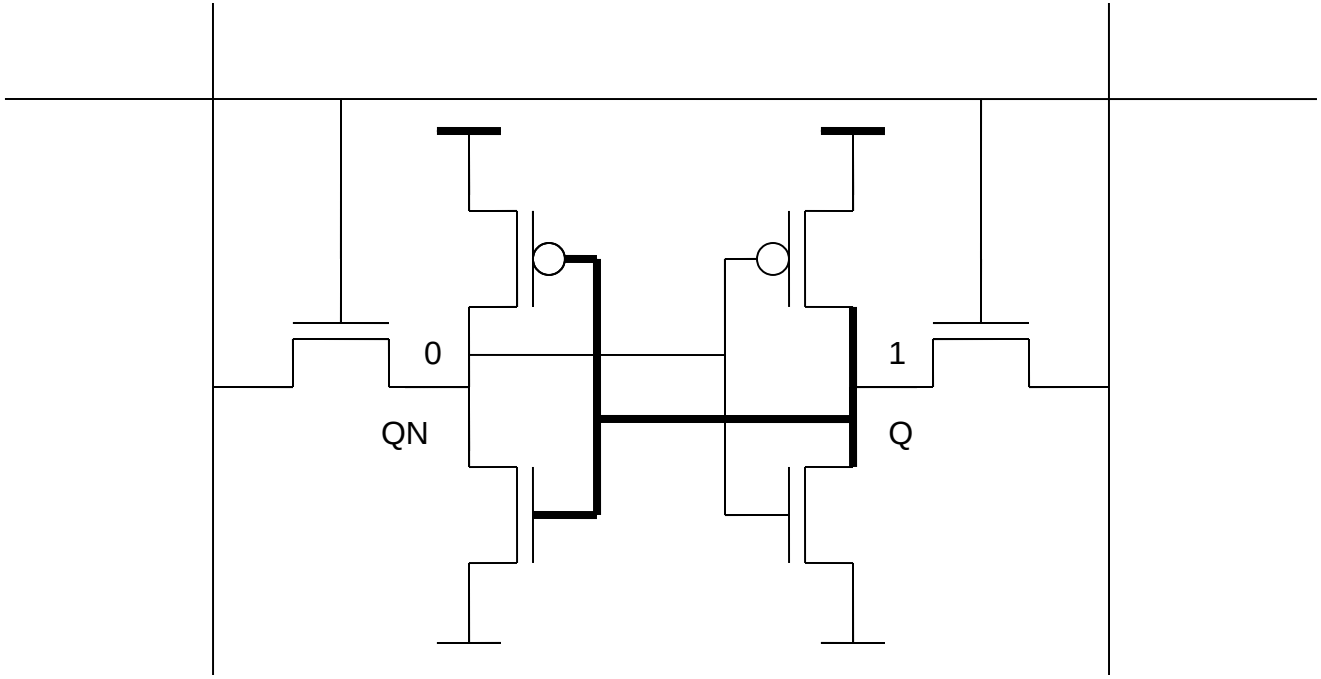


- Das führt zum “Umkippen” von RAM Zelle



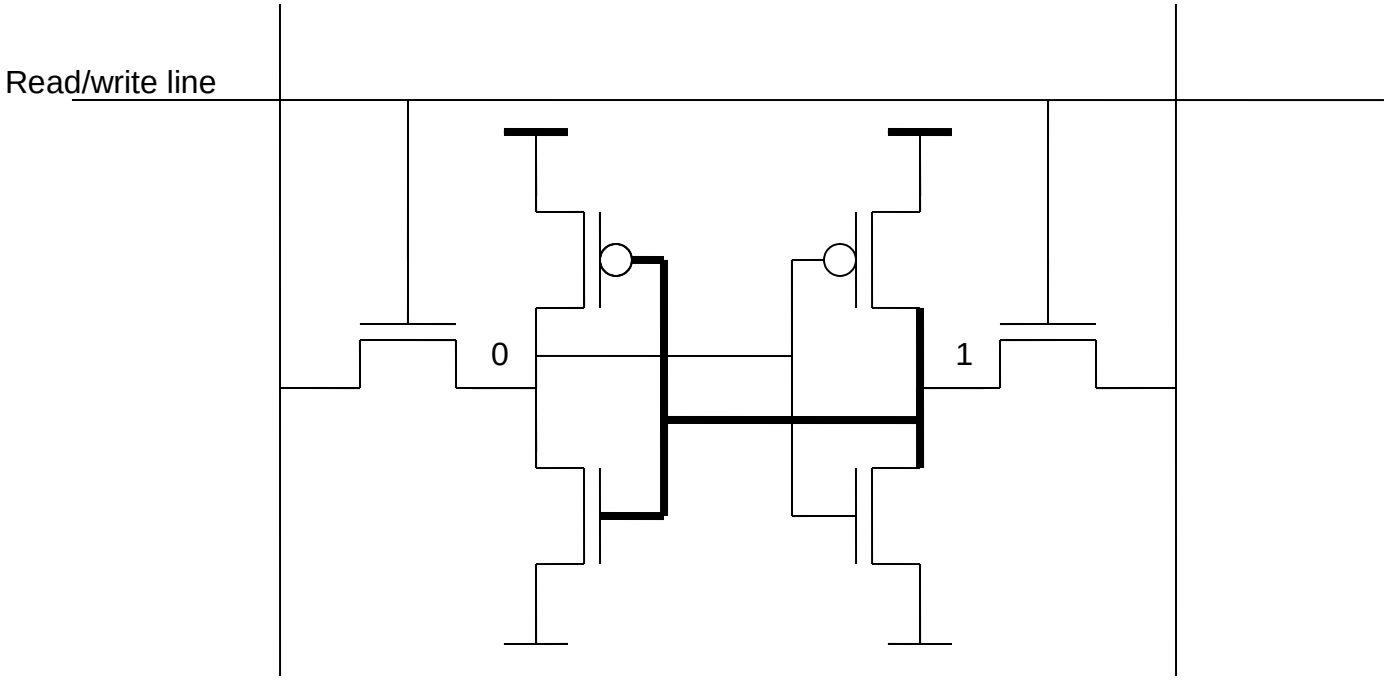
Program

- Zustand Q = 1/QN = 0 wird hergestellt



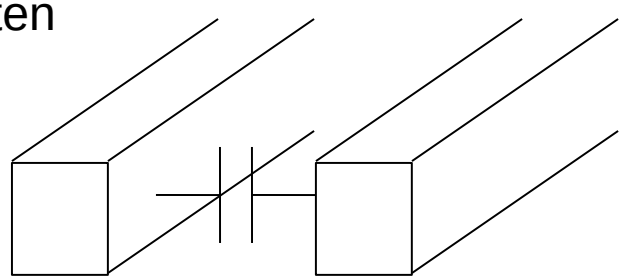
Read

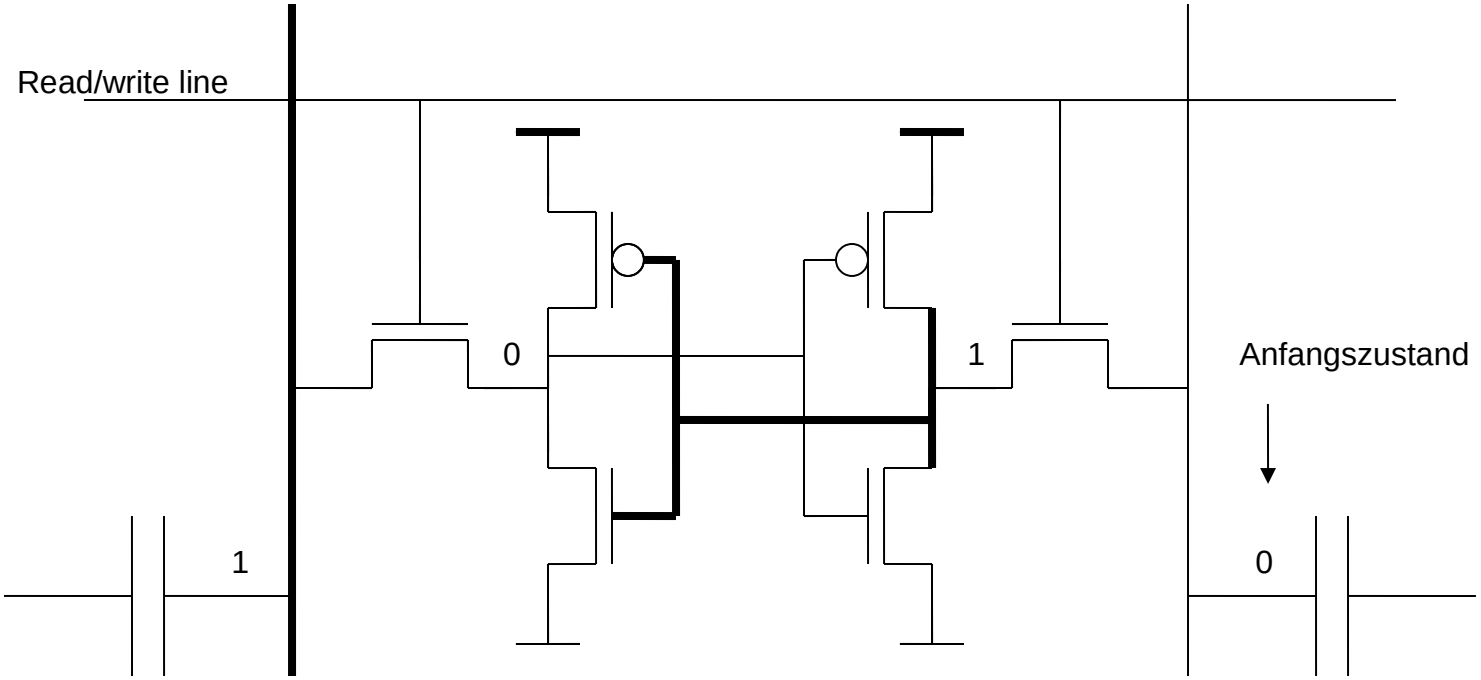
- Lesen von SRAM-Zellen



Read

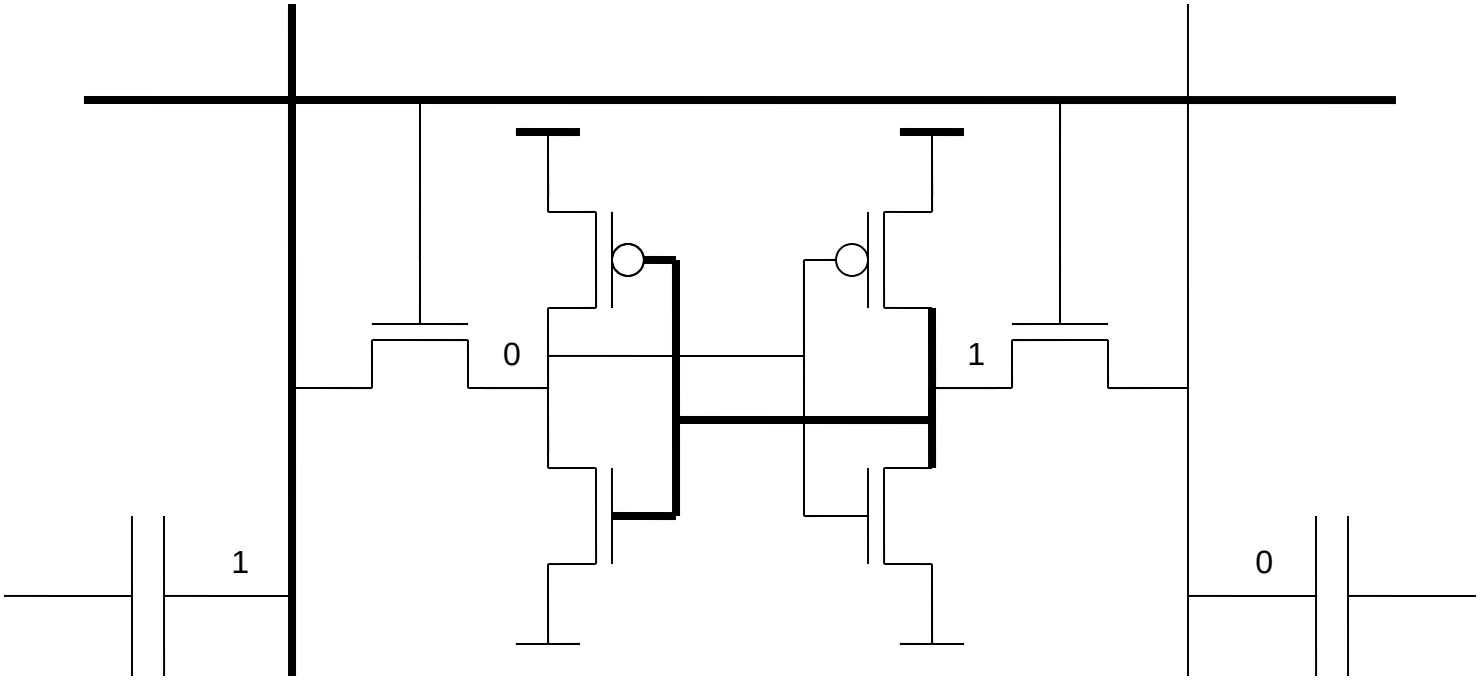
- Annahme: bit-Linien haben große Kapazitäten
- Die Bitlinien sind ungünstig aufgeladen





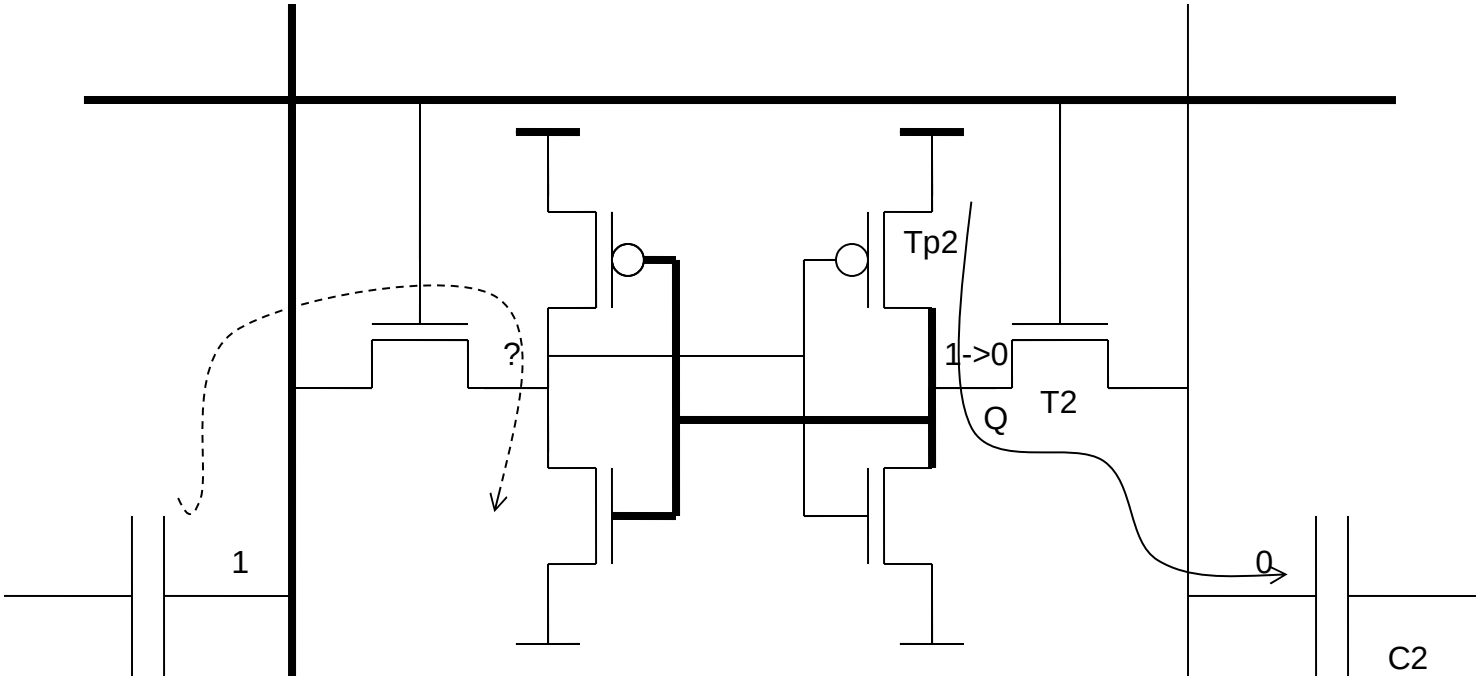
Read

- Read line wird geschaltet



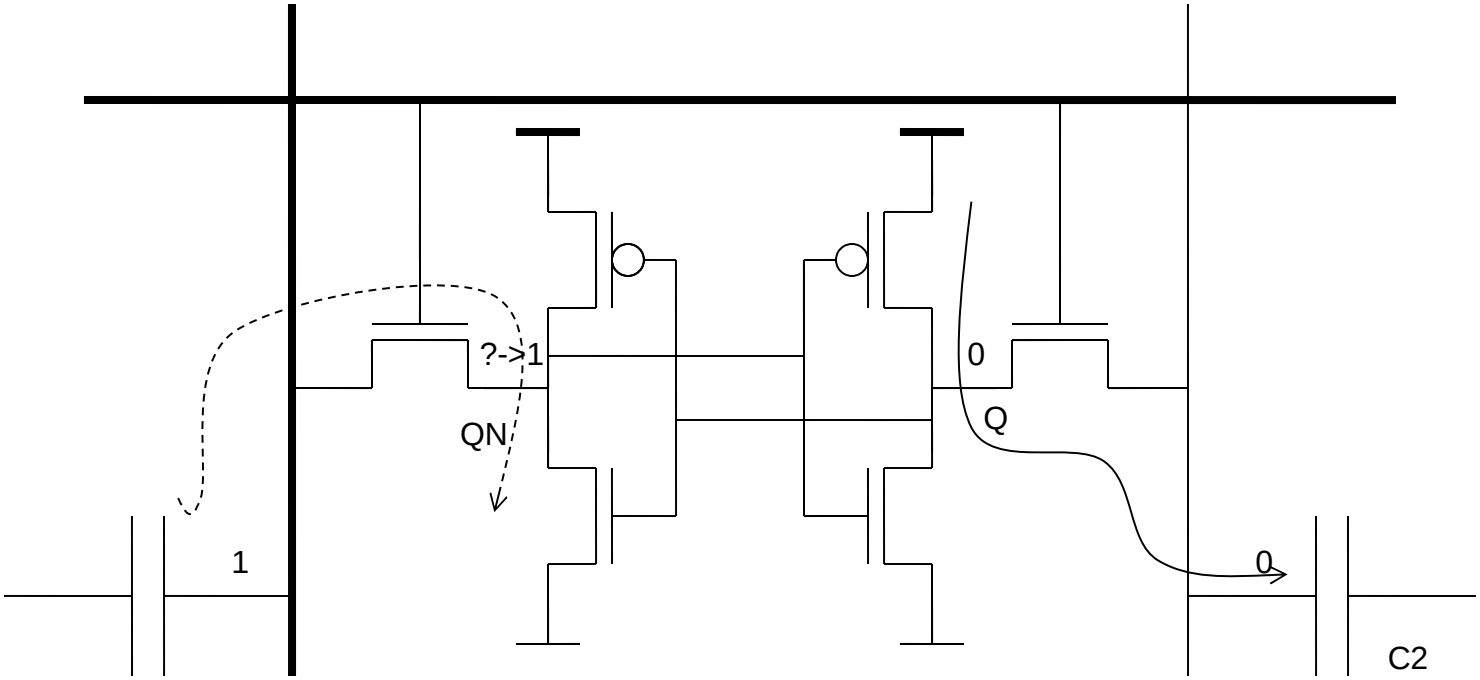
Read

- Problem
- Tp2 leitet schwächer als T2
- Tp2 muss C2 aufladen, wegen großer Kapazität C2 fließt viel Strom



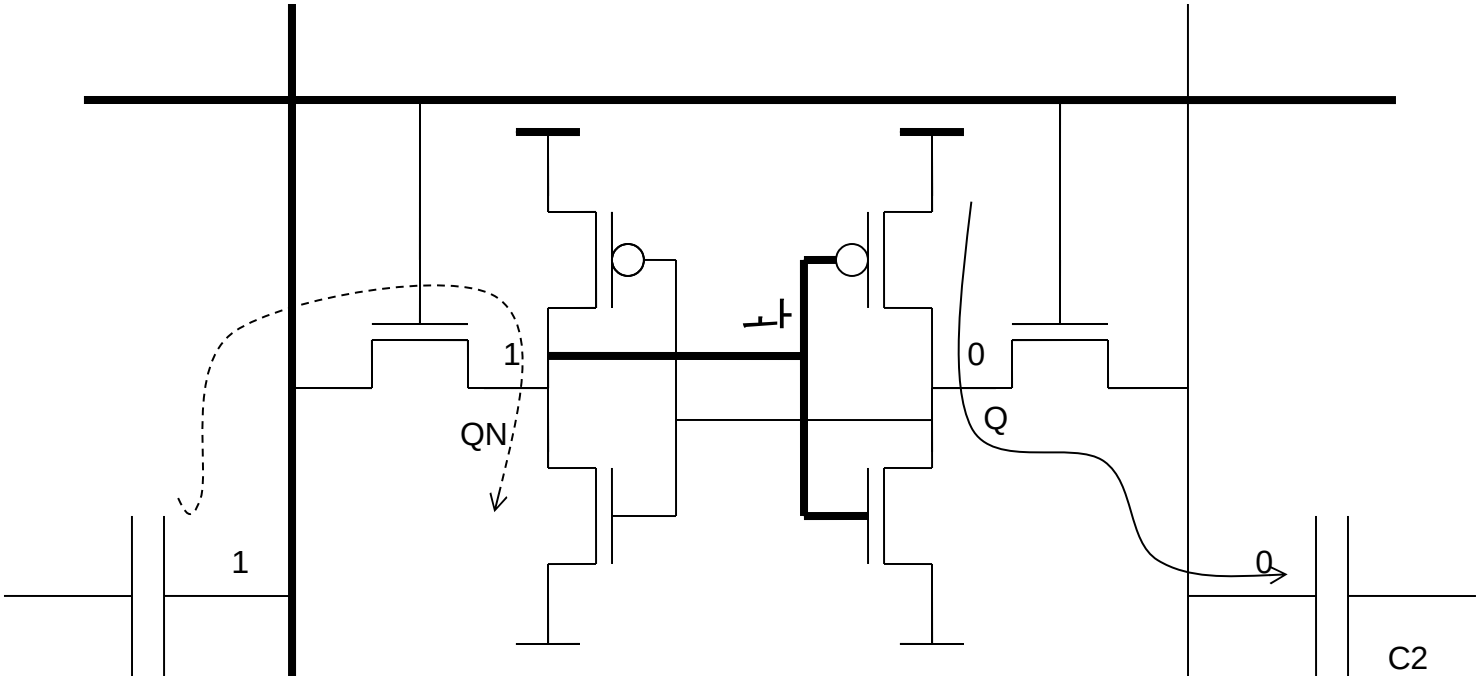
Read

- Problem
- Tp2 leitet schwächer als T2
- Tp2 muss C2 aufladen, wegen großer Kapazität C2 fließt viel Strom -> Potential Q fällt



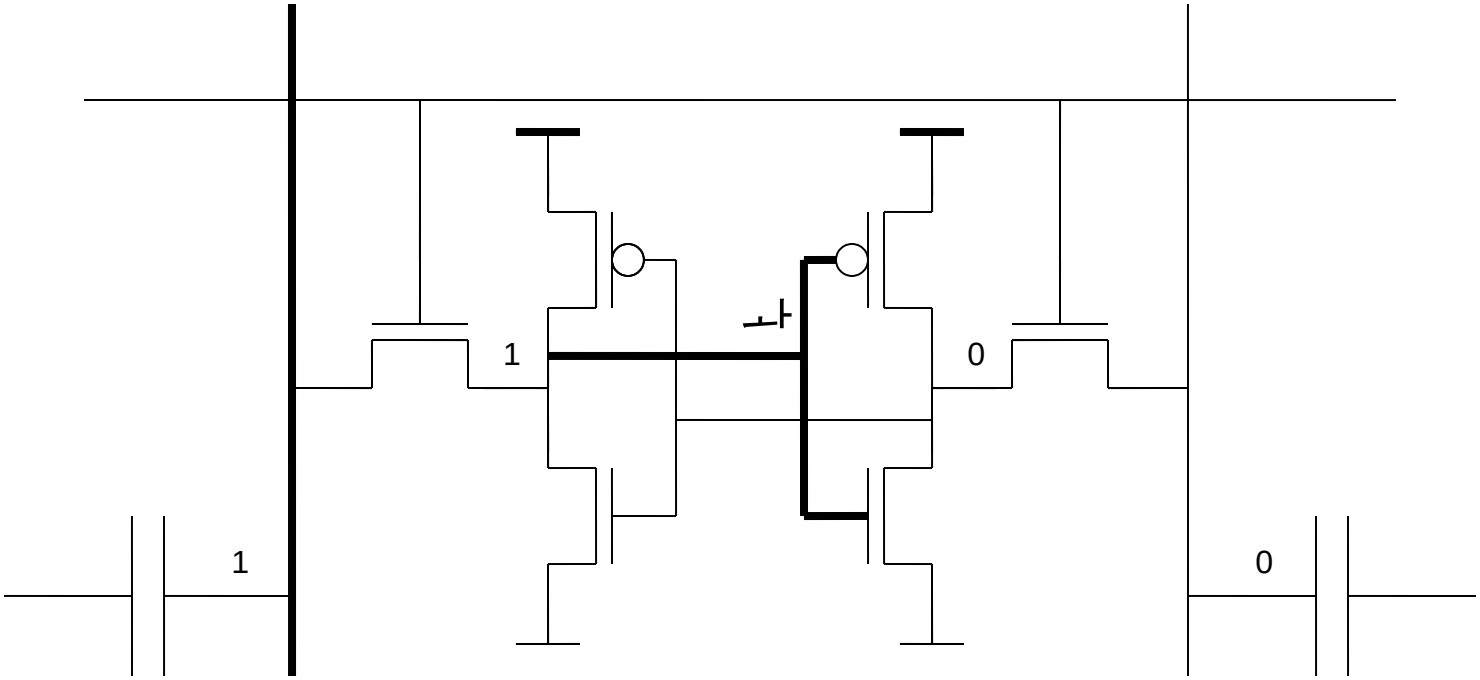
Read

- Problem
- Tp2 leitet schwächer als T2
- Tp2 muss C2 aufladen, wegen großer Kapazität C2 fließt viel Strom -> Potential Q fällt -> QN wird aufgeladen, RAM Zelle kippt um



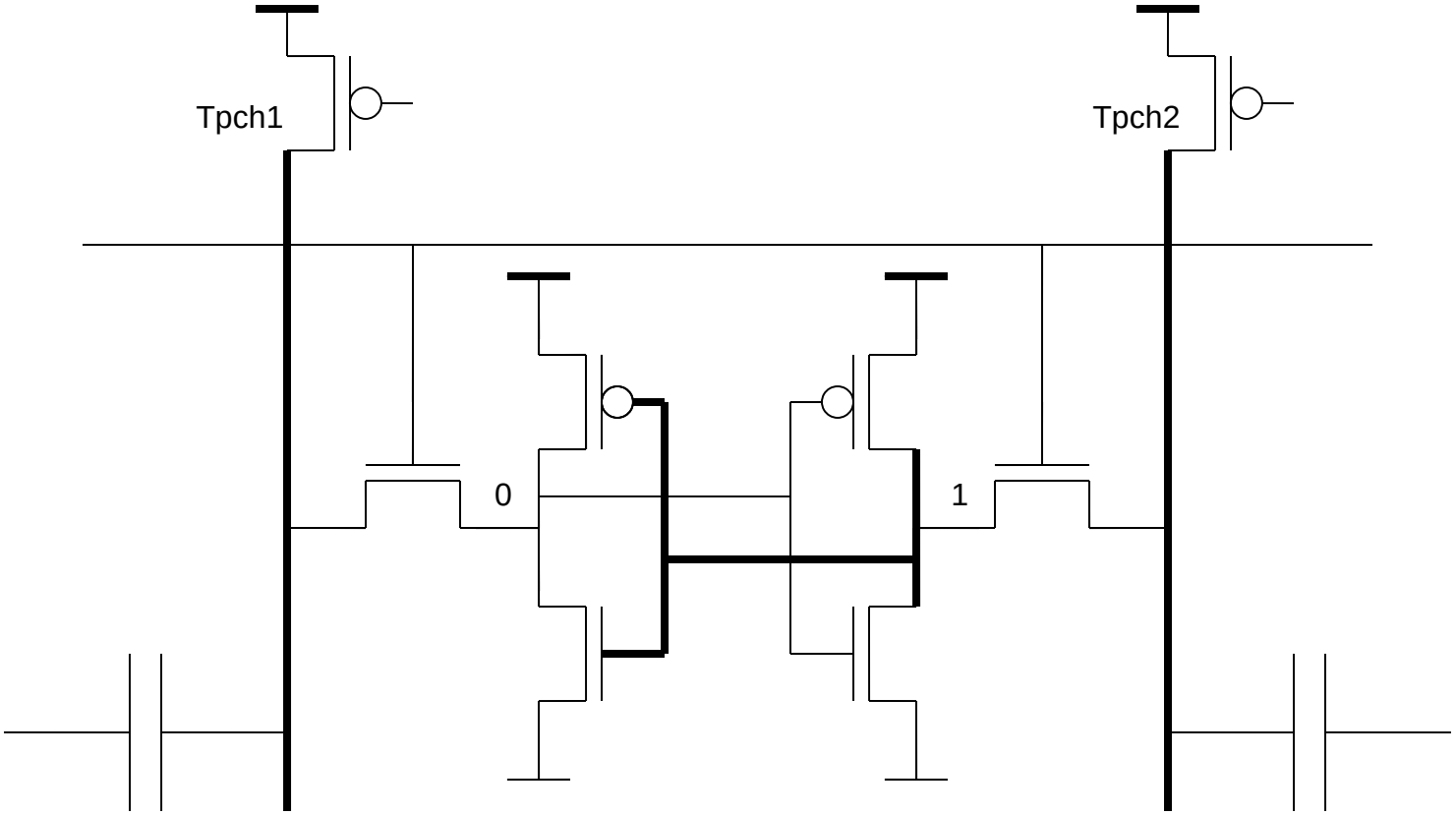
Read

- RAM Zelle bleibt überschrieben



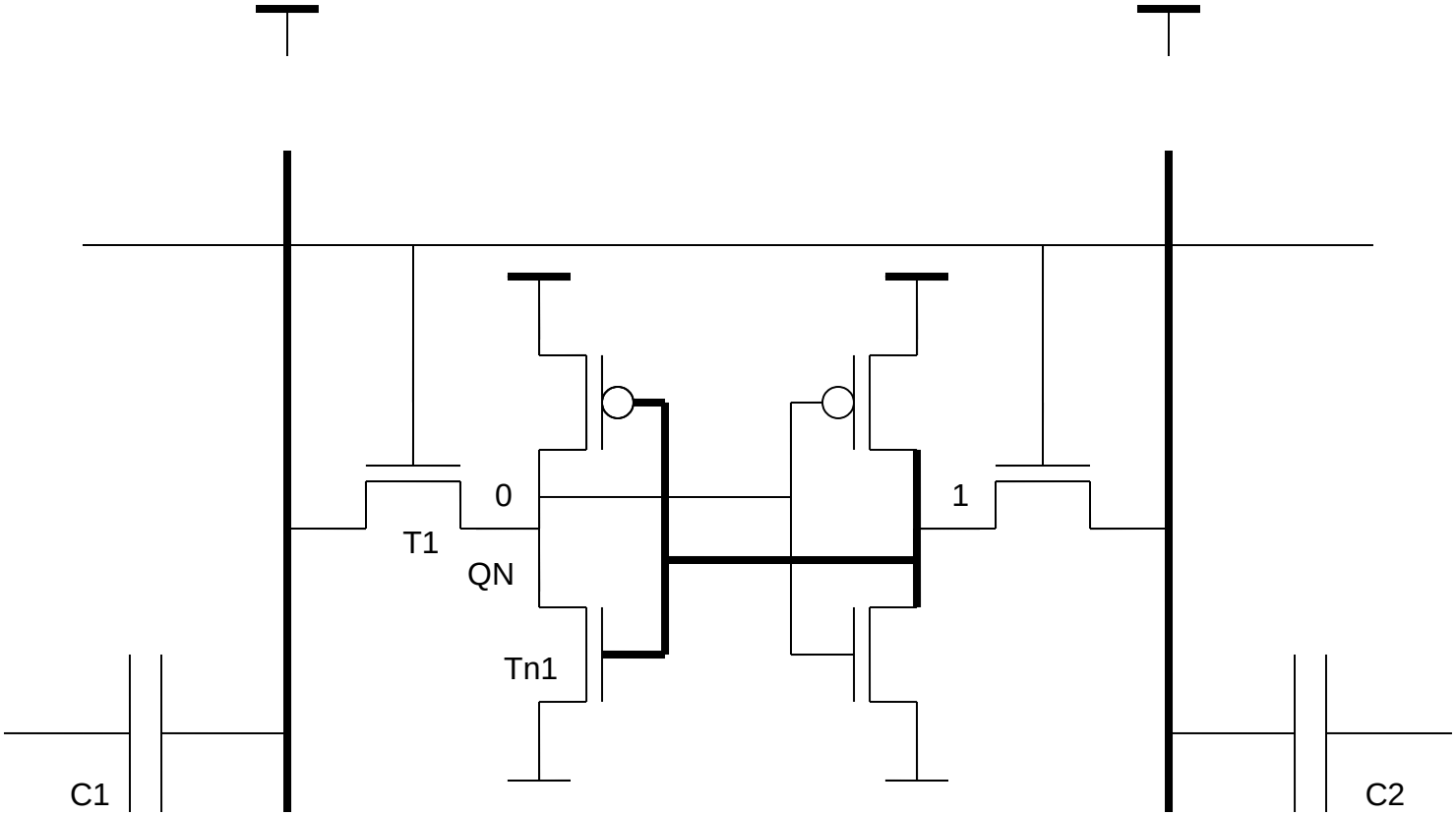
Read mit Precharge

- Lösung des Problems: Precharge
- Bitlinien werden auf 1 aufgeladen (mithilfe von Transistoren Tpch1 und Tpch2)



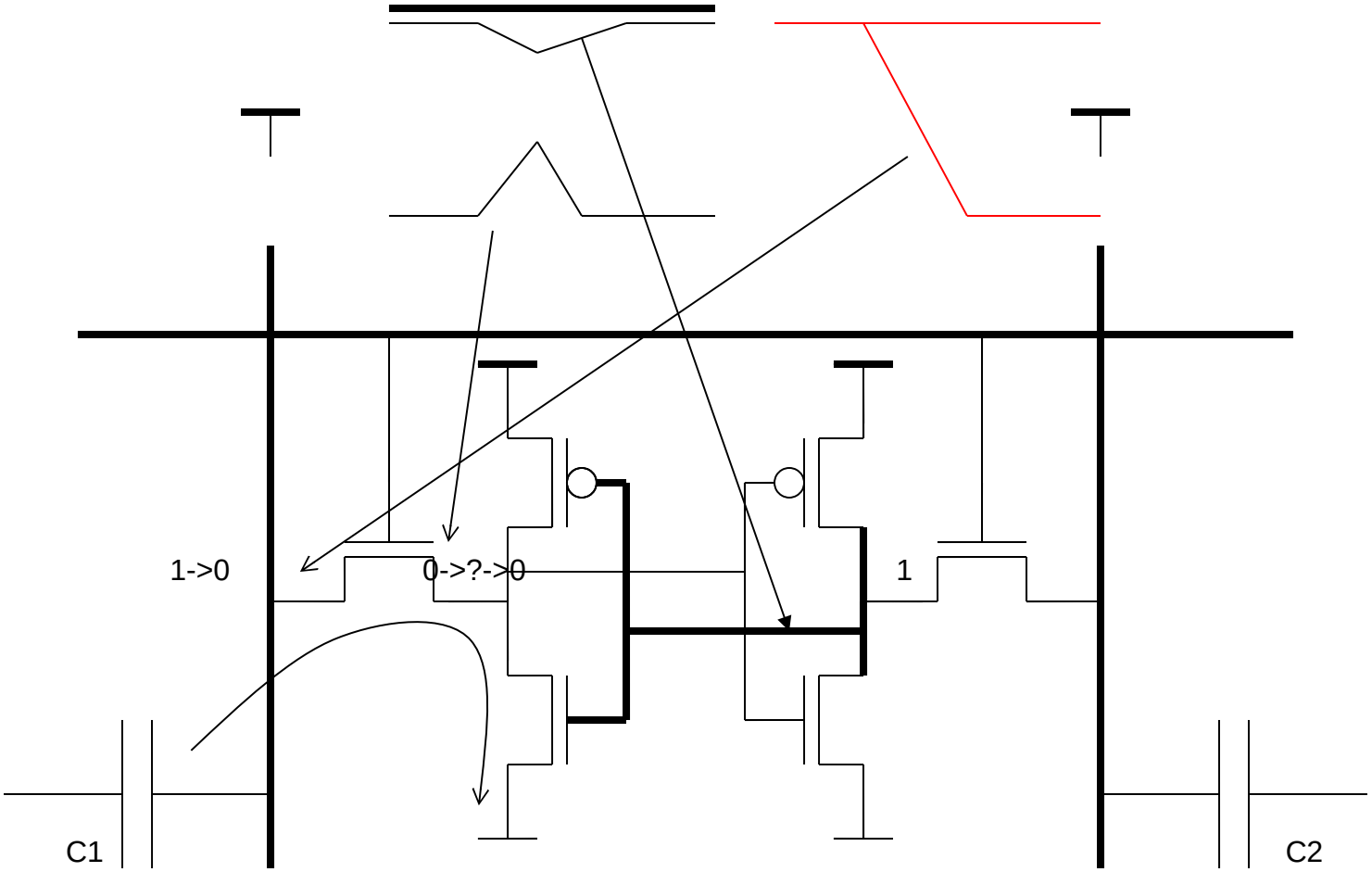
Read mit Precharge

- C2 ist „richtig“ aufgeladen und verursacht kein Problem beim Lesen
- C1 muss entladen werden
- Das gelingt weil Tn1 stärker leitet (R kleiner) als T1 -> QN bleibt 0



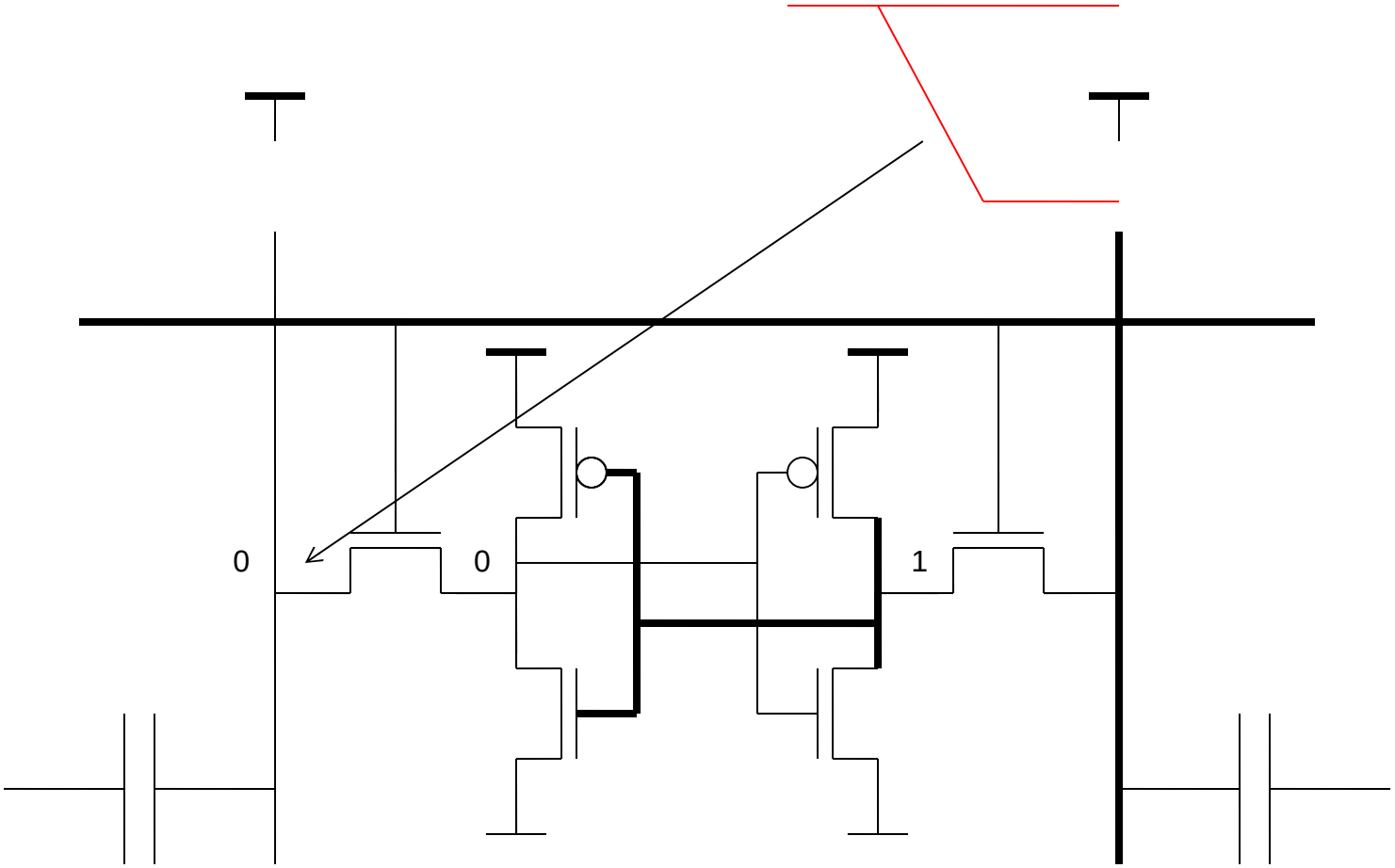
Read mit Precharge

- C1 wird entladen, RAM Zelle kippt nicht um und wird richtig geschrieben



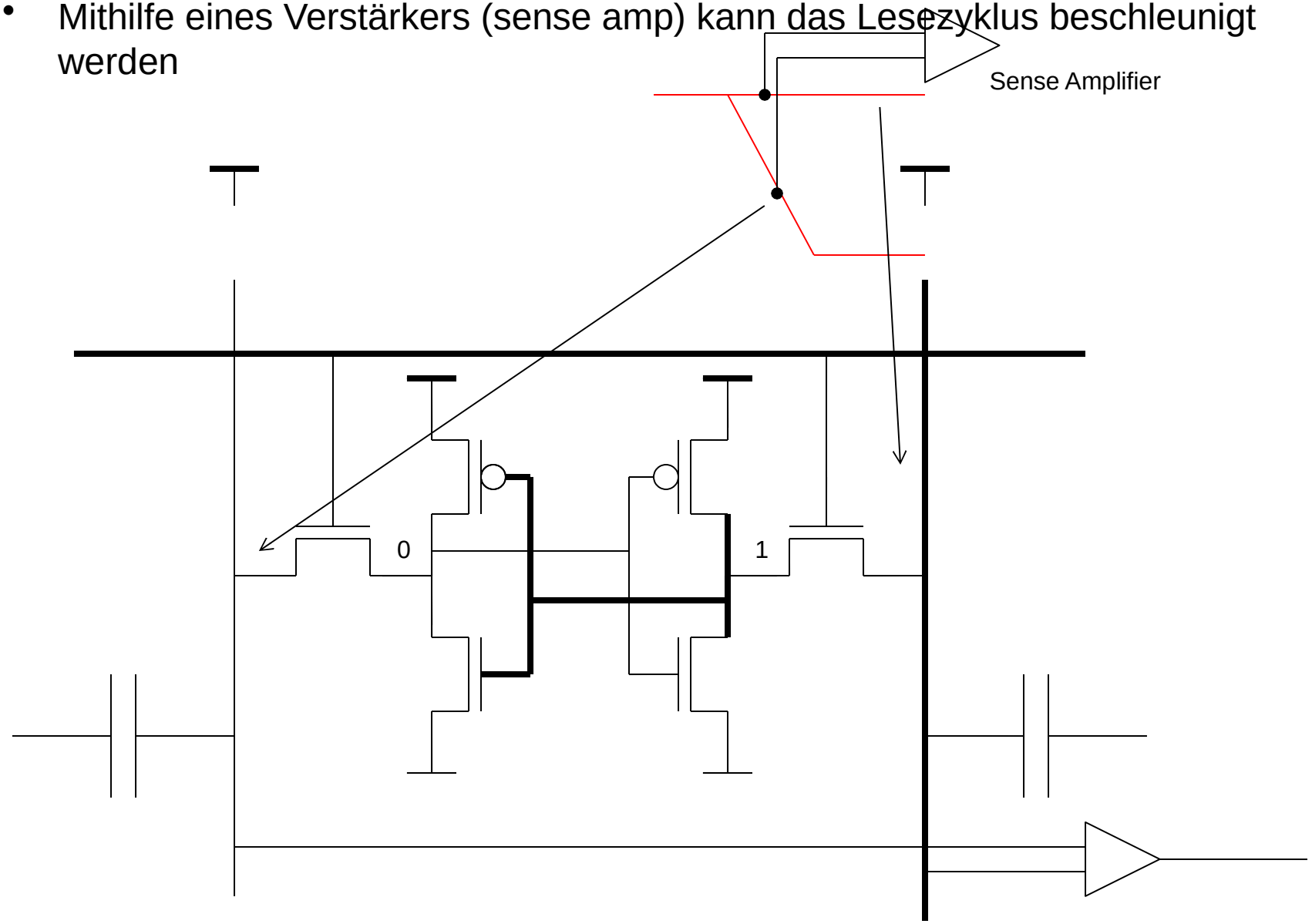
Read mit Precharge

- C1 wird entladen, RAM Zelle kippt nicht um und wird richtig geschrieben



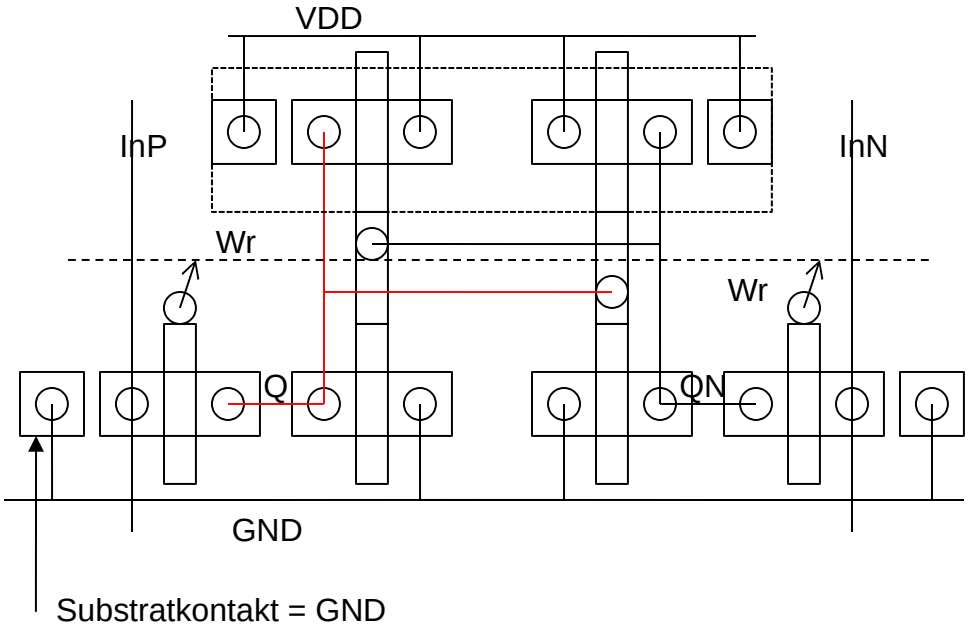
Read mit Precharge

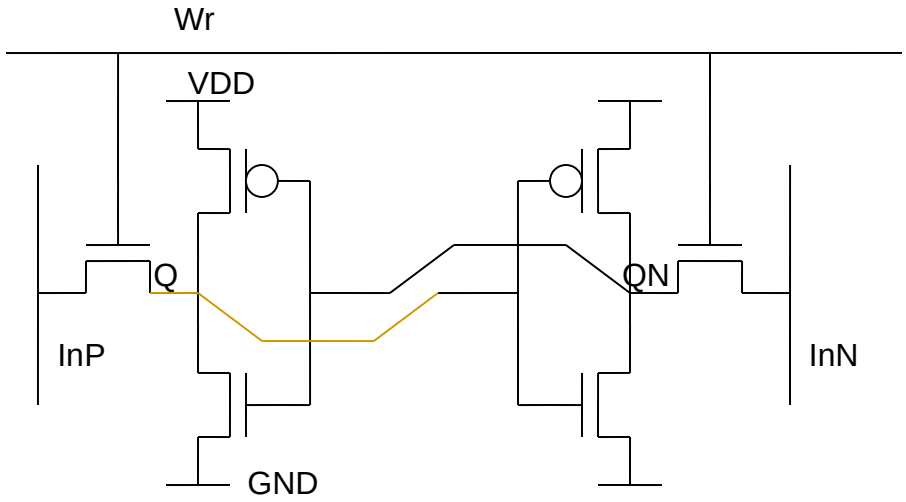
- Mithilfe eines Verstärkers (sense amp) kann das Lesezyklus beschleunigt werden



Layout

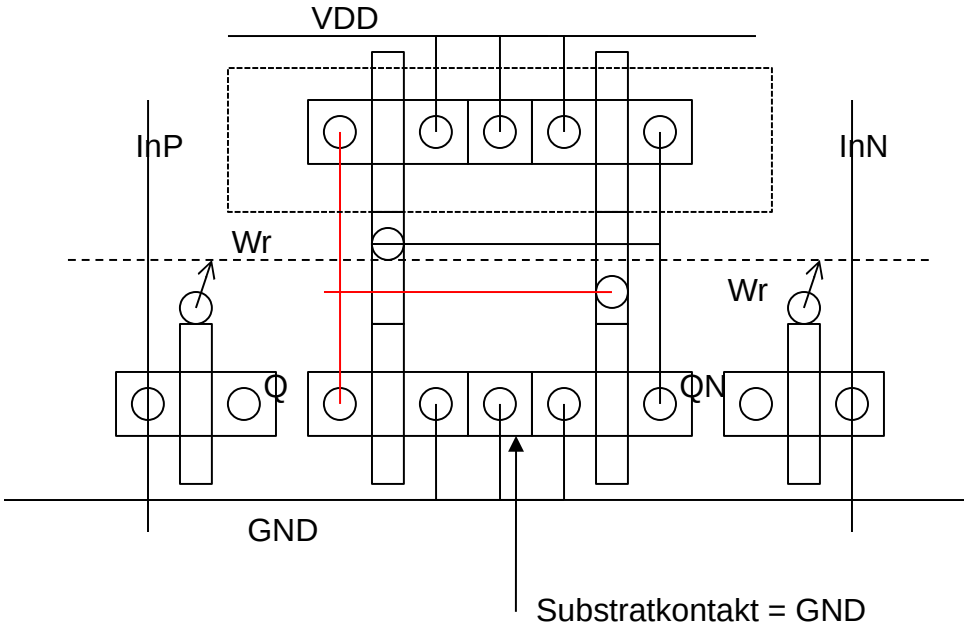
- …

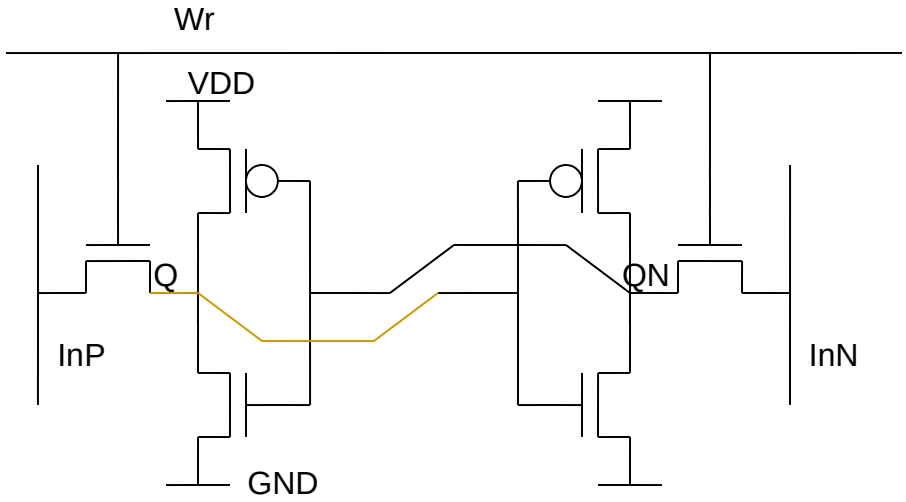

VDD
InP
InN
Wr
Wr
Q
QN
GND
Substratkontakt = GND


Wr
VDD
Q
QN
InP
InN
GND

Layout

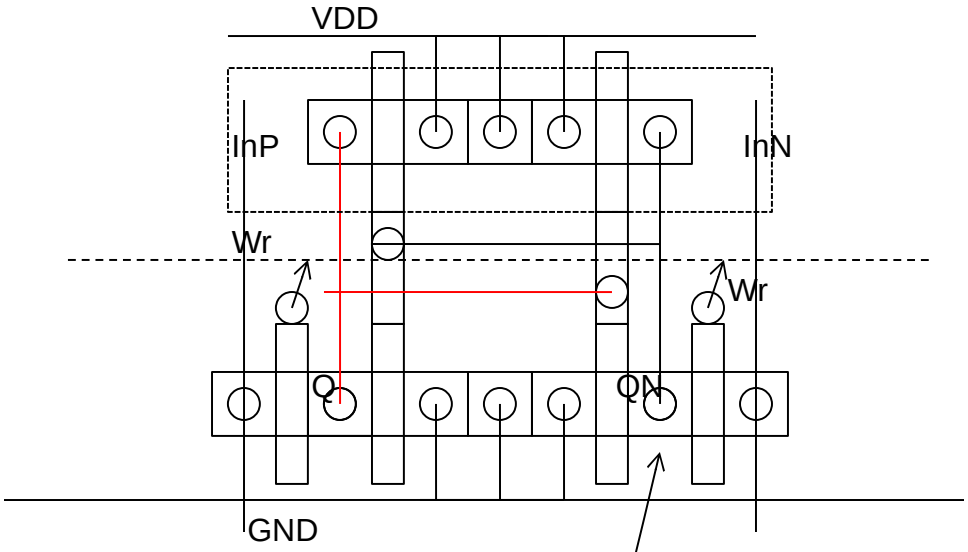
- …

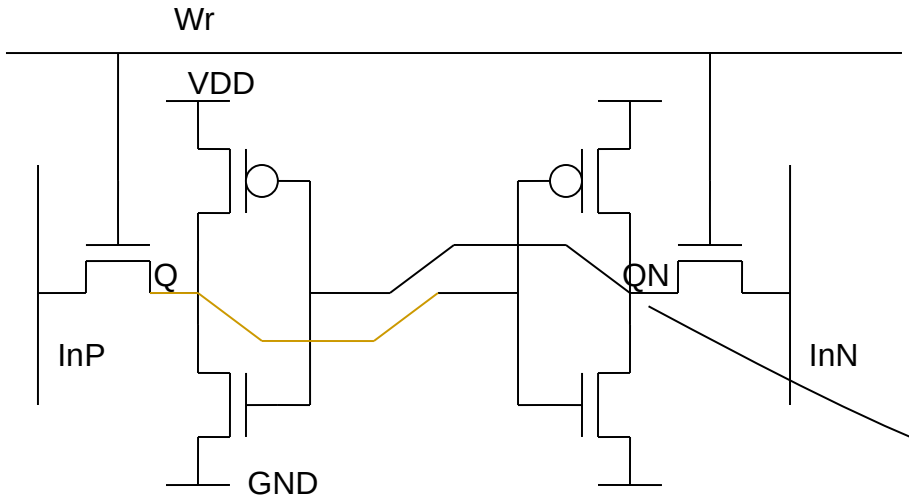

VDD
InP
InN
Wr
Wr
Q
QN
GND
Substratkontakt = GND


Wr
VDD
Q
QN
InP
InN
GND

Layout

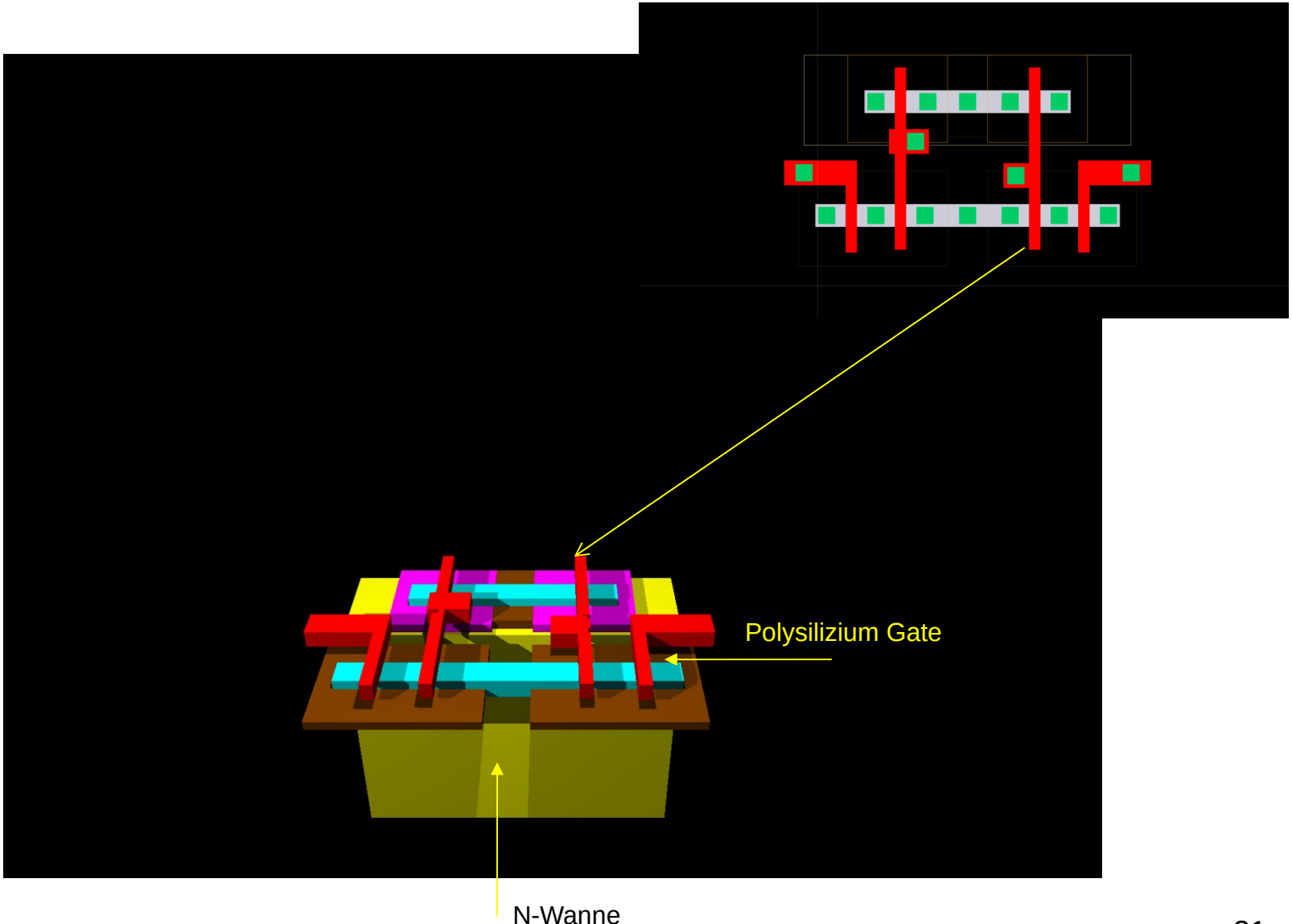
- …





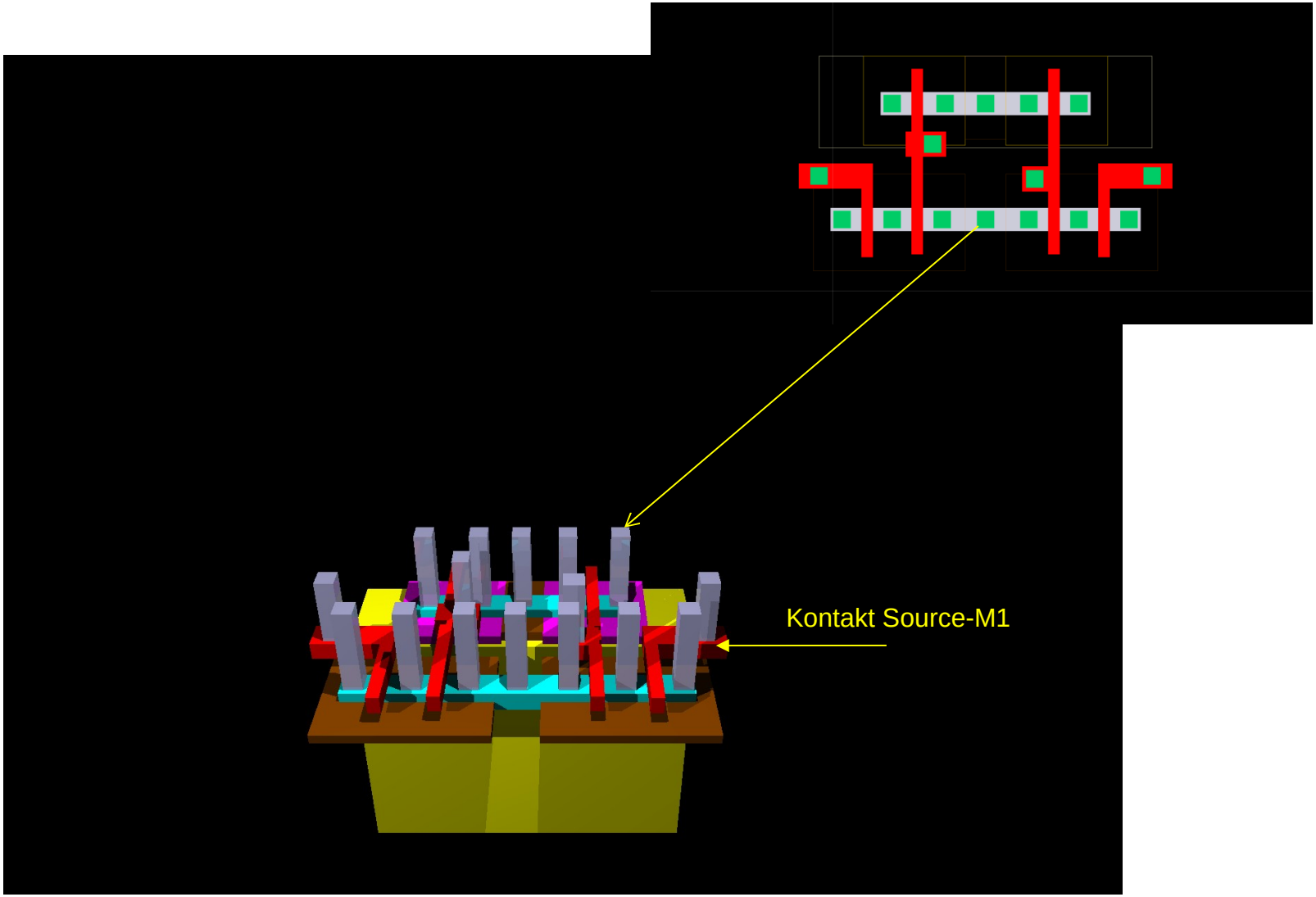
Layout

- …



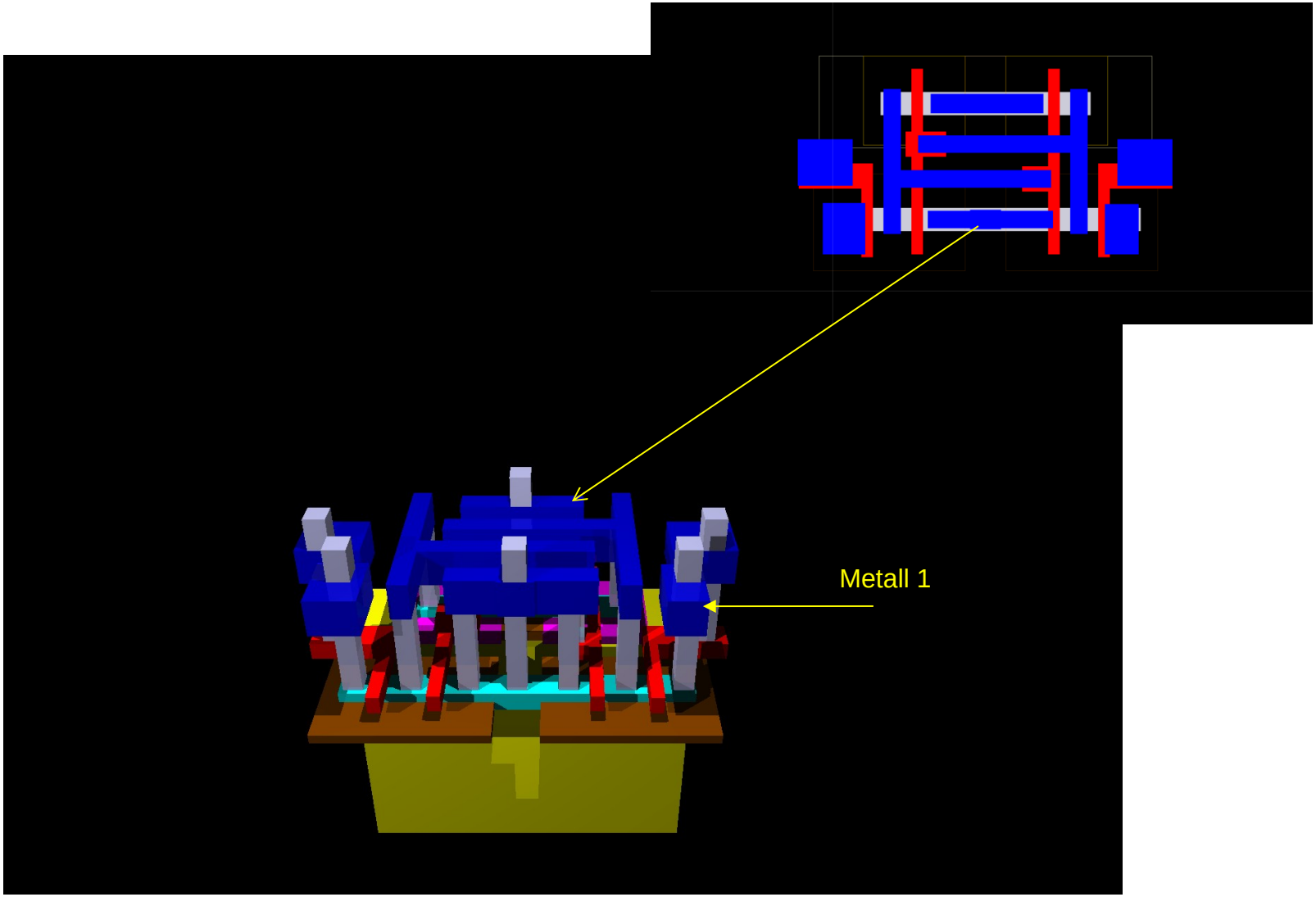
Layout

- …



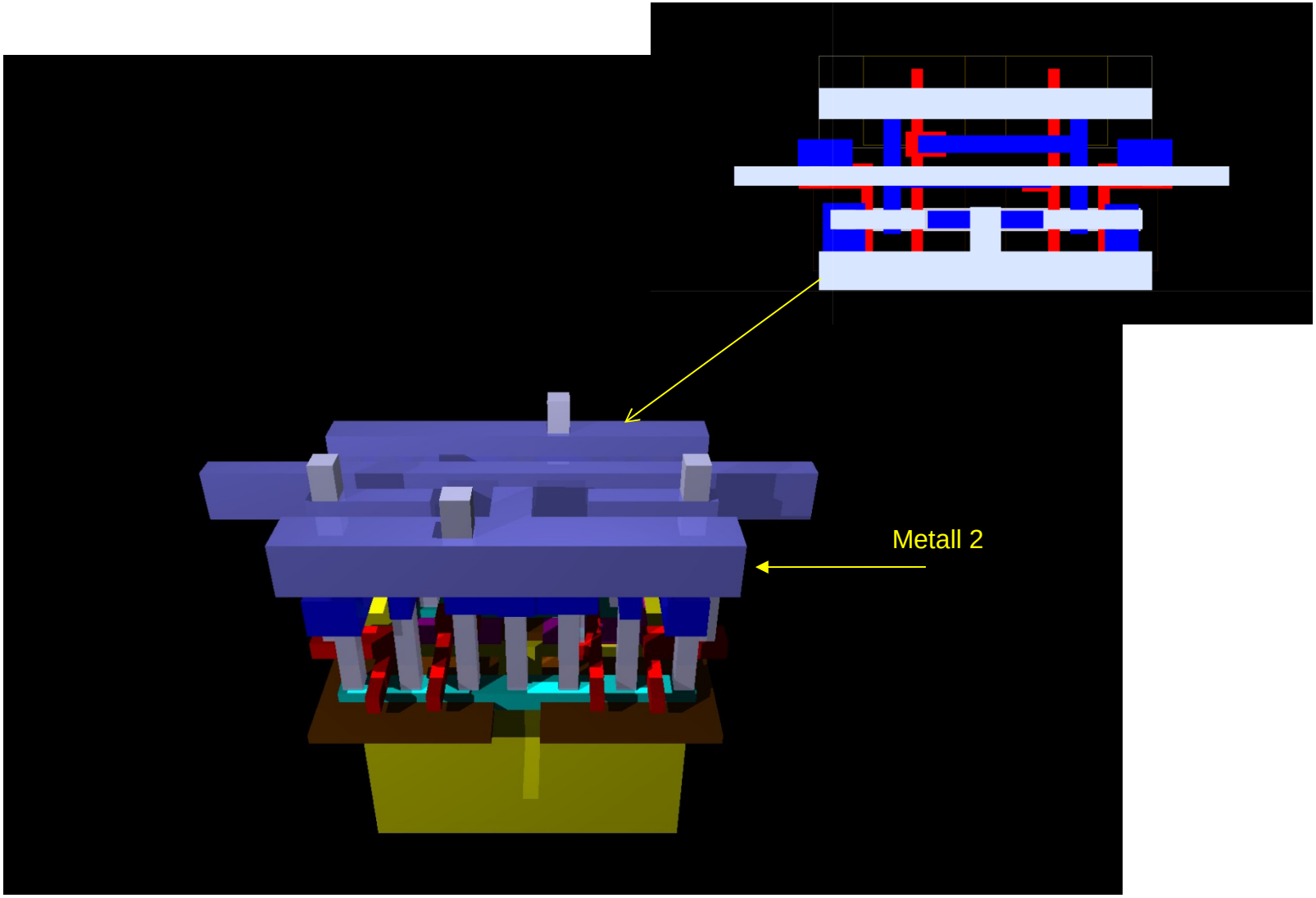
Layout

- …



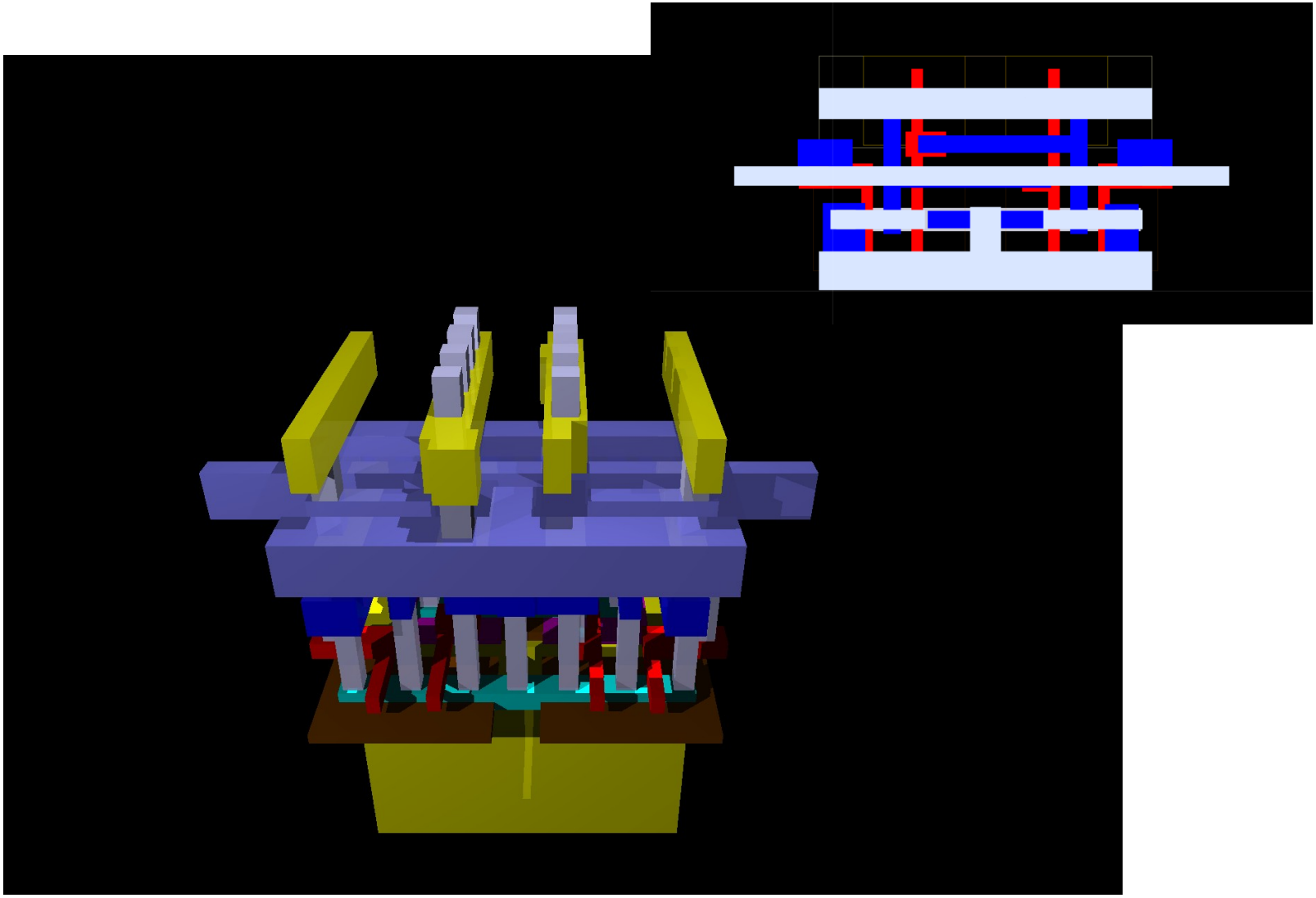
Layout

- …



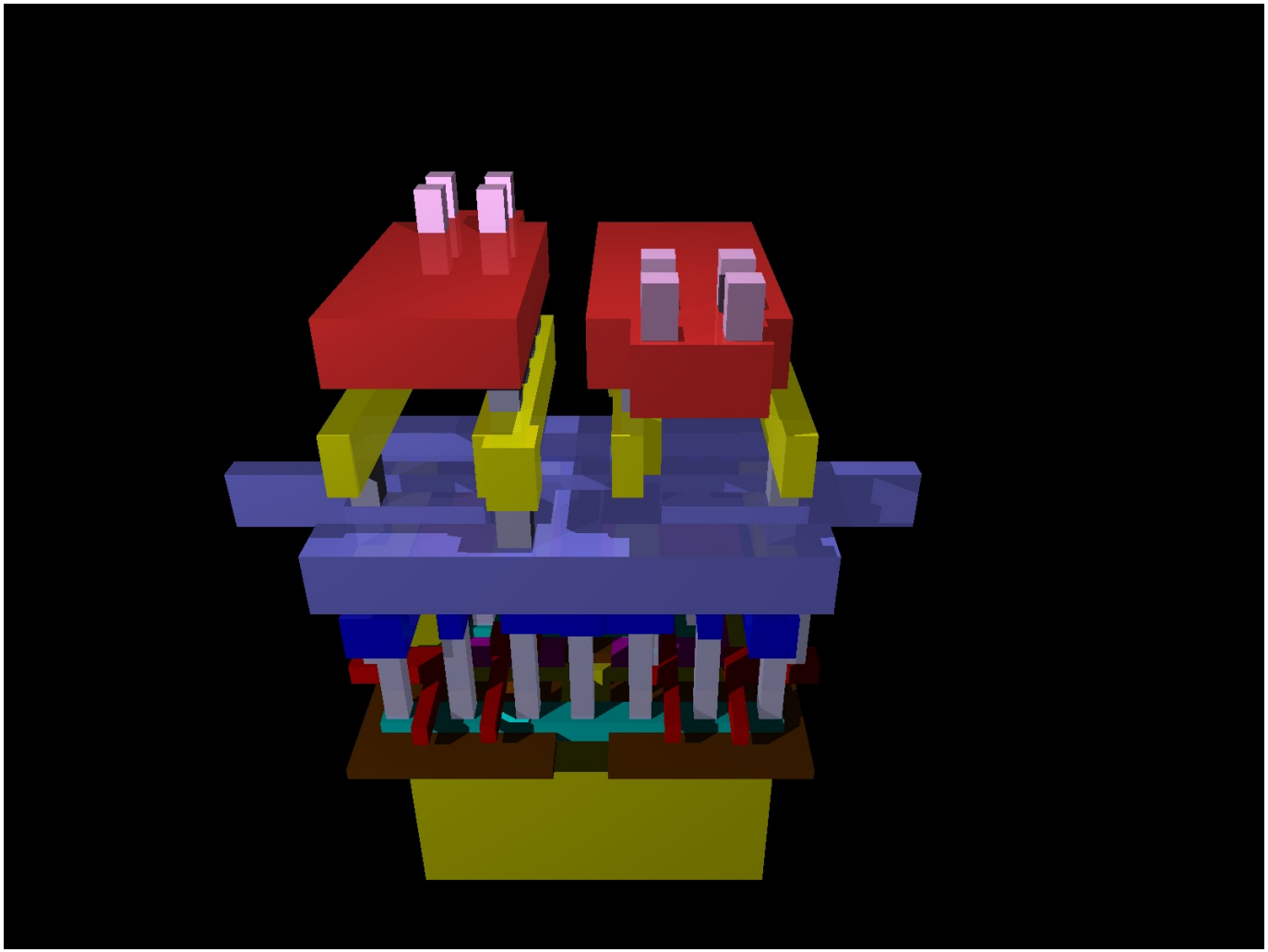
Layout

- …



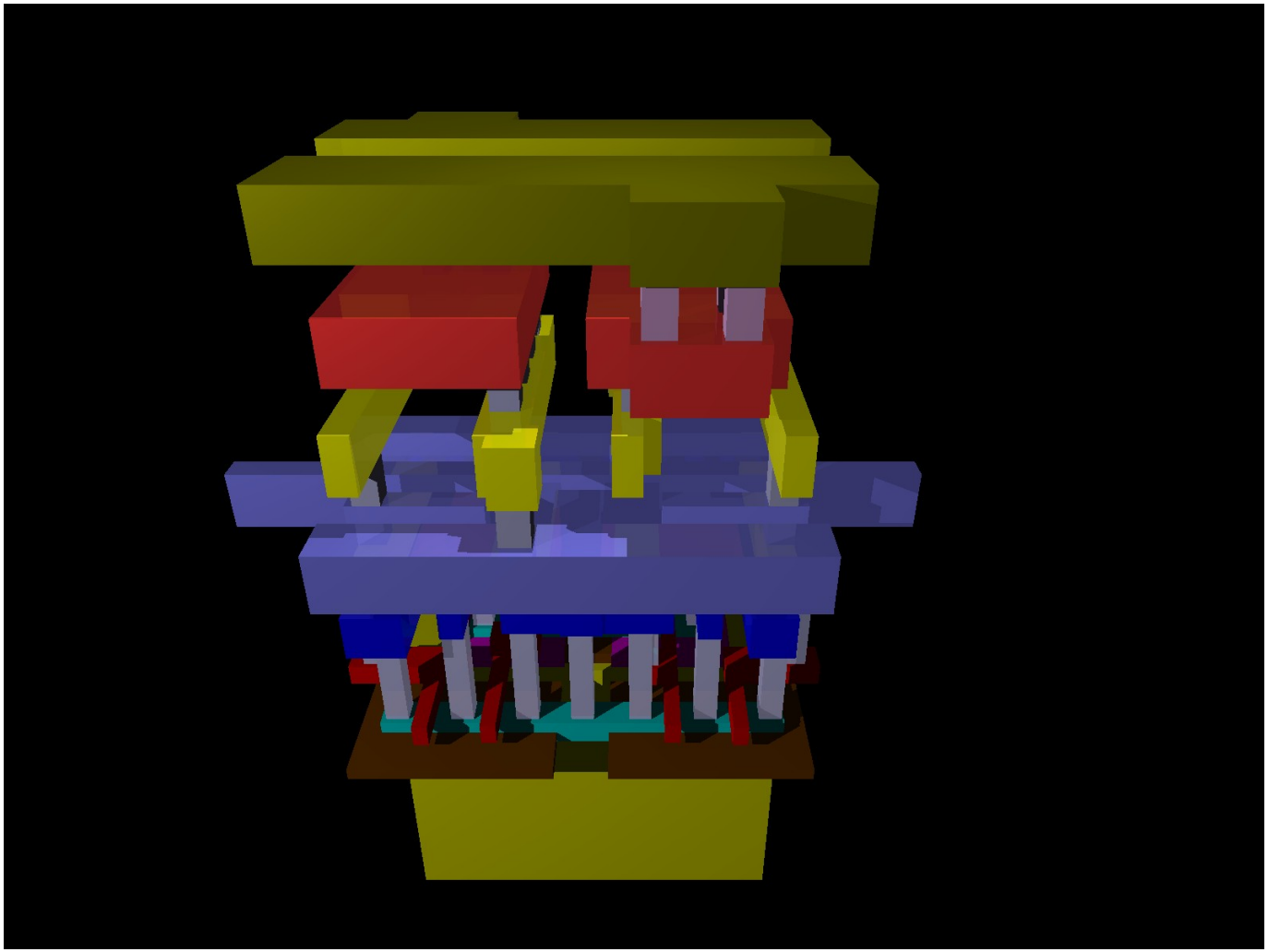
Layout

- …



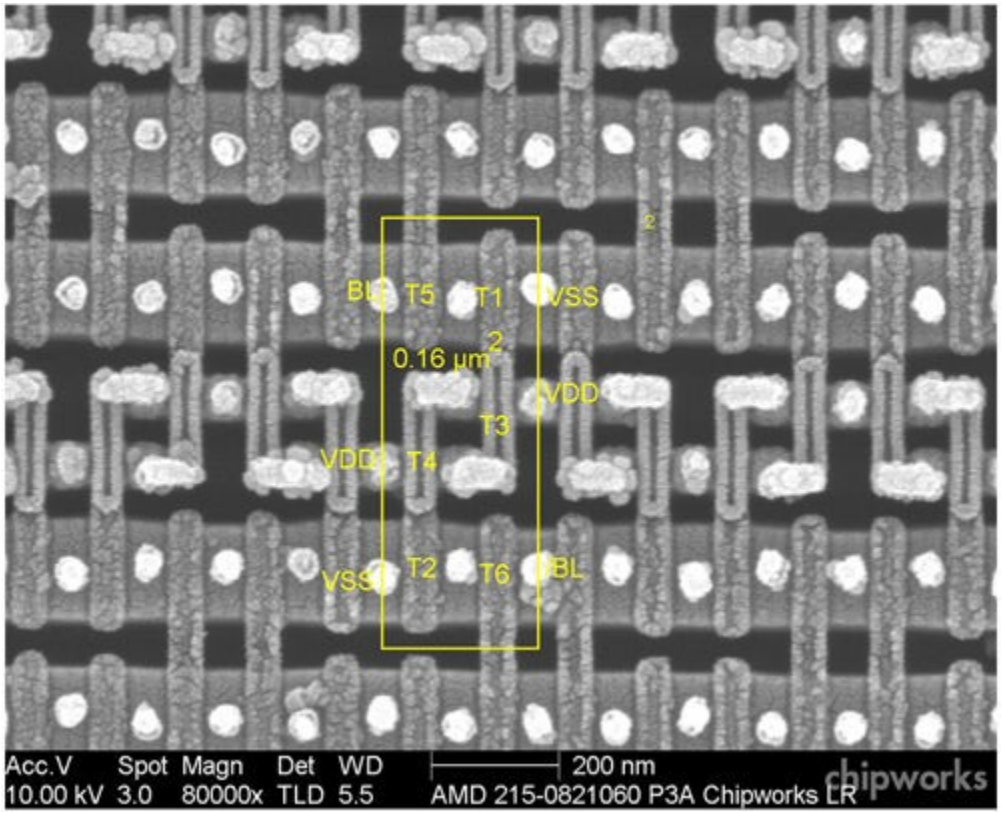
Layout

- …

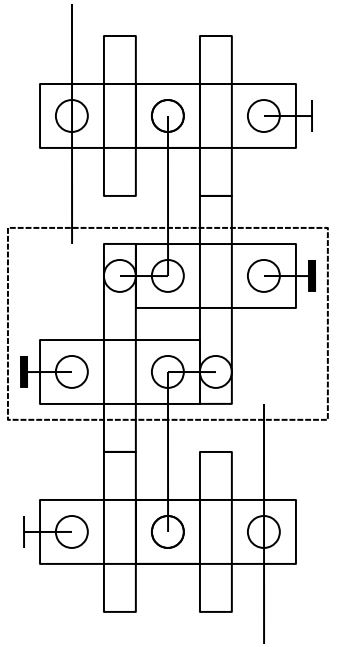


Layout 2

- SRAM



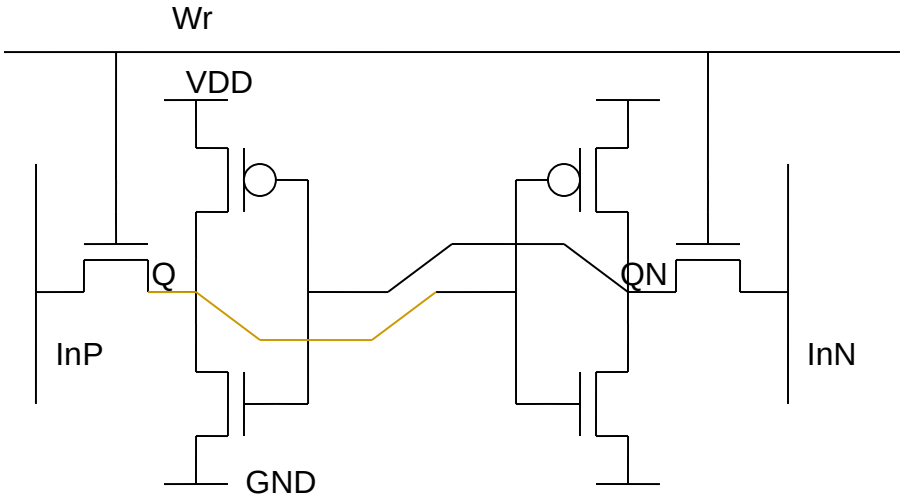
AMD 215-0821060 28 nm HP 6T-SRAM at Poly - Plan View SEM

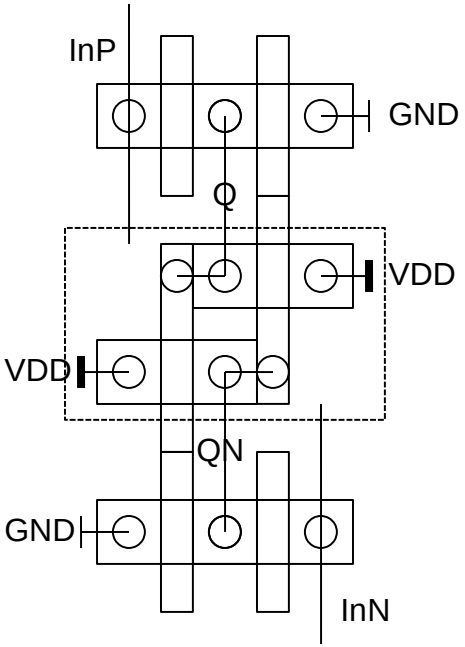


https://www.chipworks.com/about-chipworks/overview/blog/review-tsmc-28-nm-process-technology

Layout 2

- SRAM



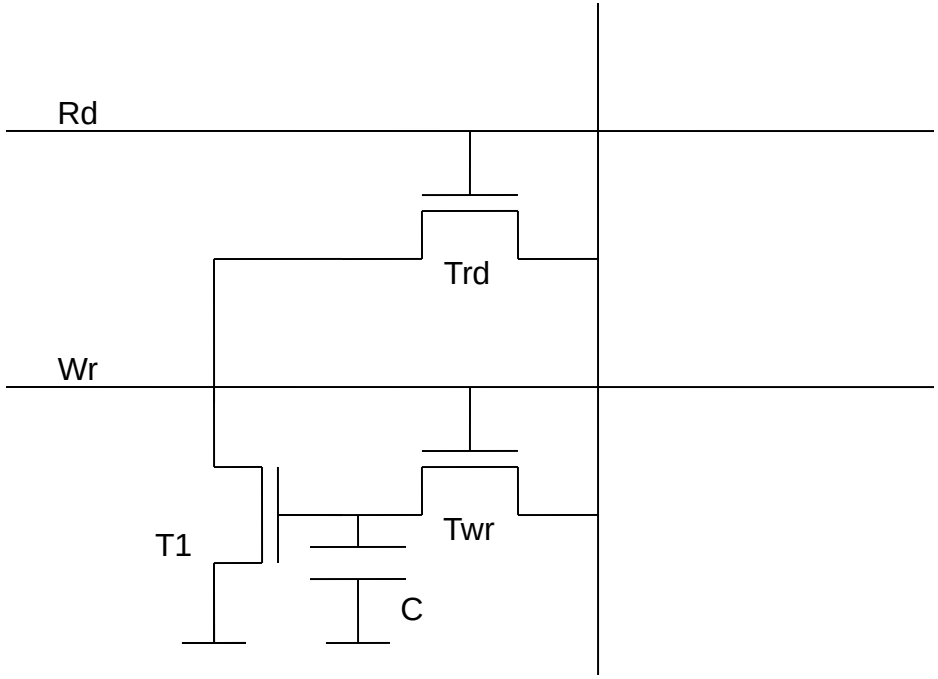


https://www.chipworks.com/about-chipworks/overview/blog/review-tsmc-28-nm-process-technology

DRAM

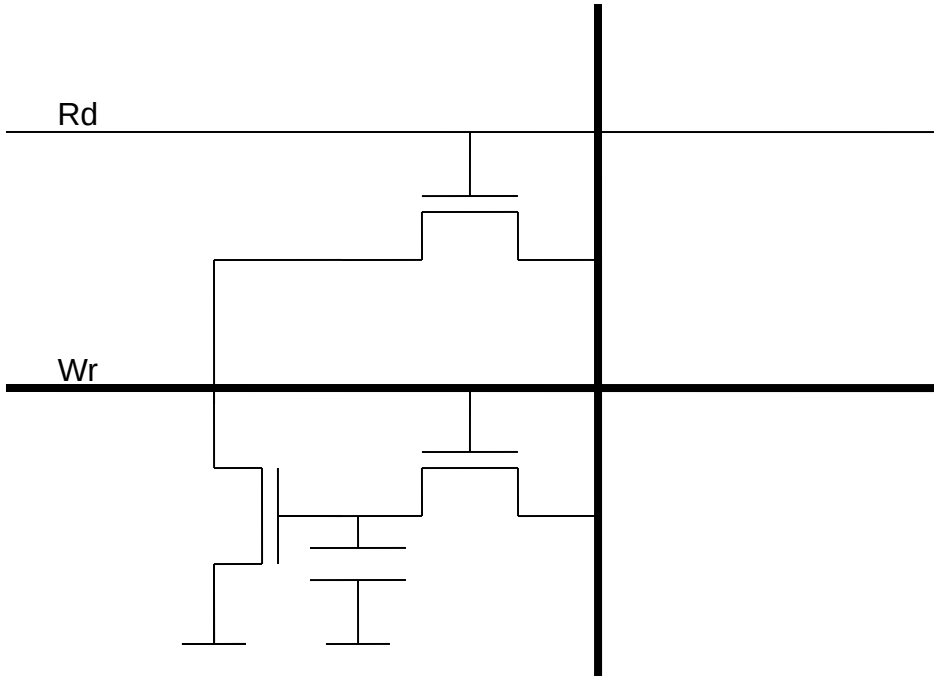
DRAM – Variante 1

- DRAM basiert auf Kondensatoren (C) als Speicherelementen
- Realisierung 1
- Transistor T1 dient als Verstärker
- Getrennte Rd und Wr Schalter Twr und Trd



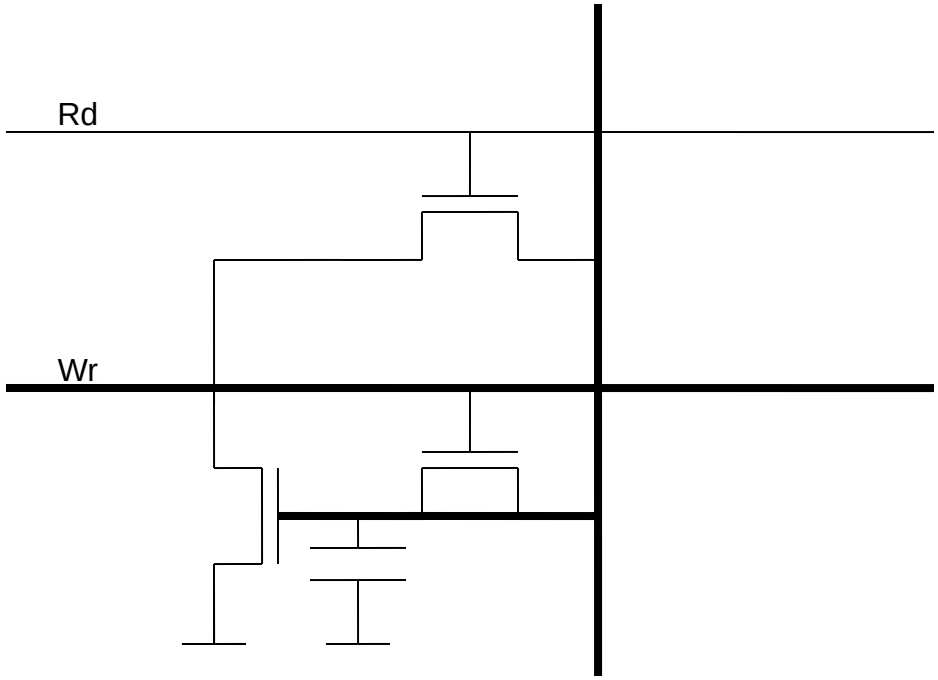
Write

- DRAM



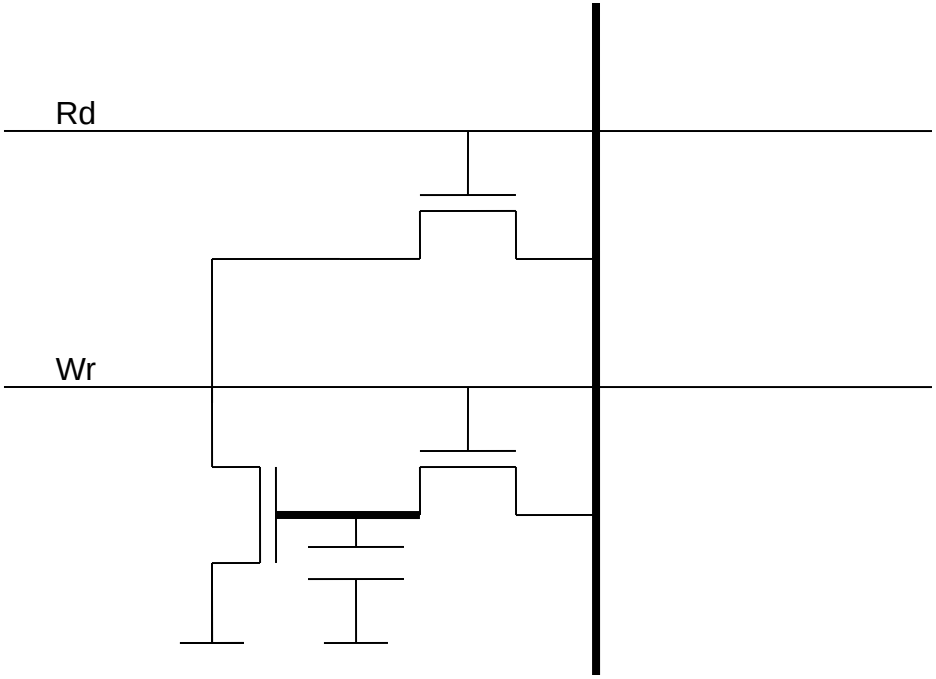
Write

- DRAM



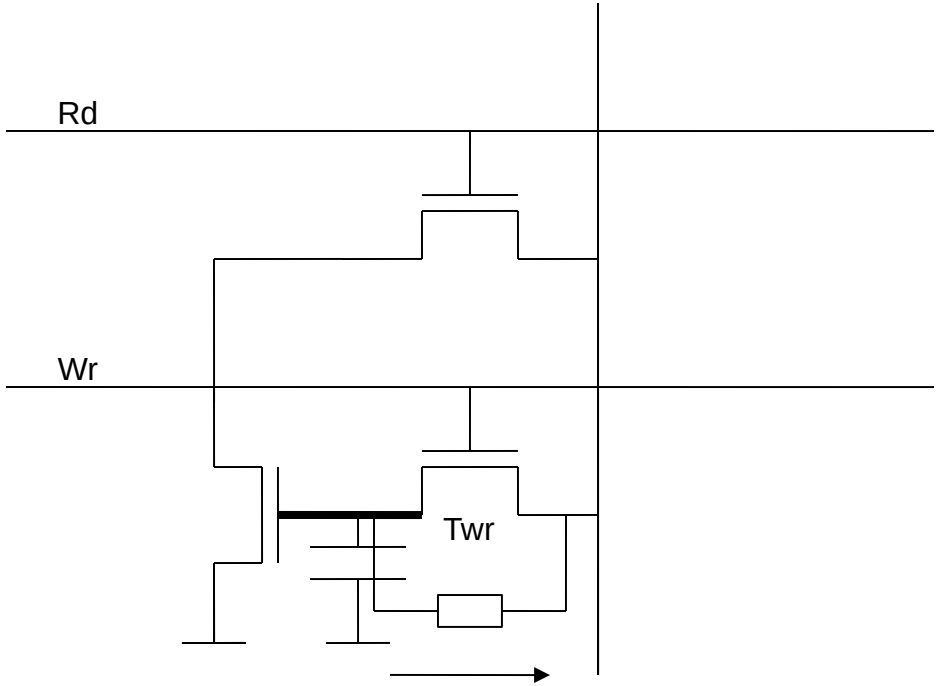
Write

- DRAM



Store

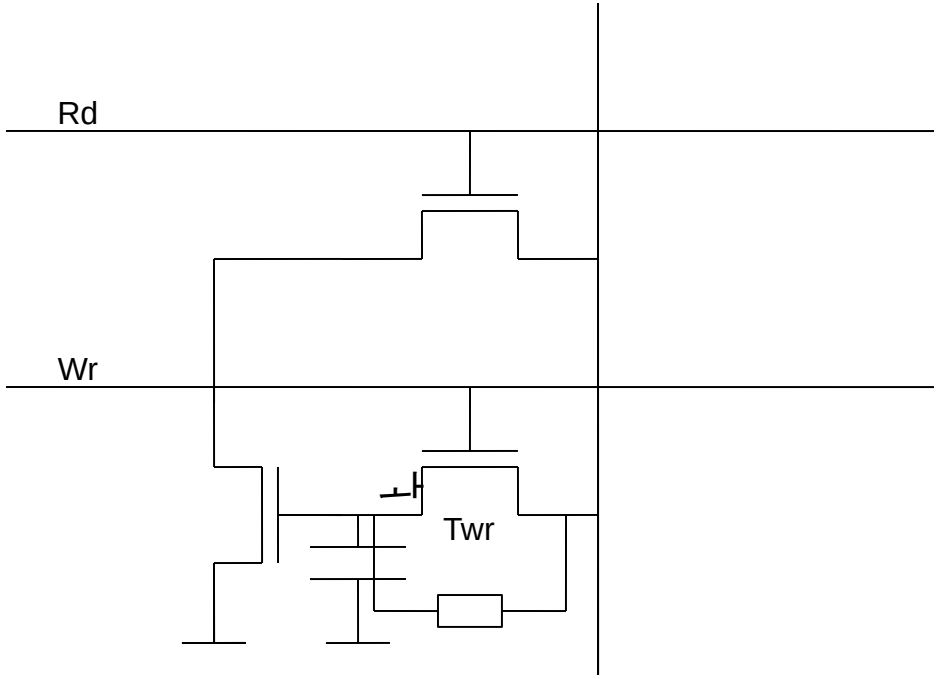
- DRAM



Leckstrom ist Problem, Ladung fliesst ab

Store

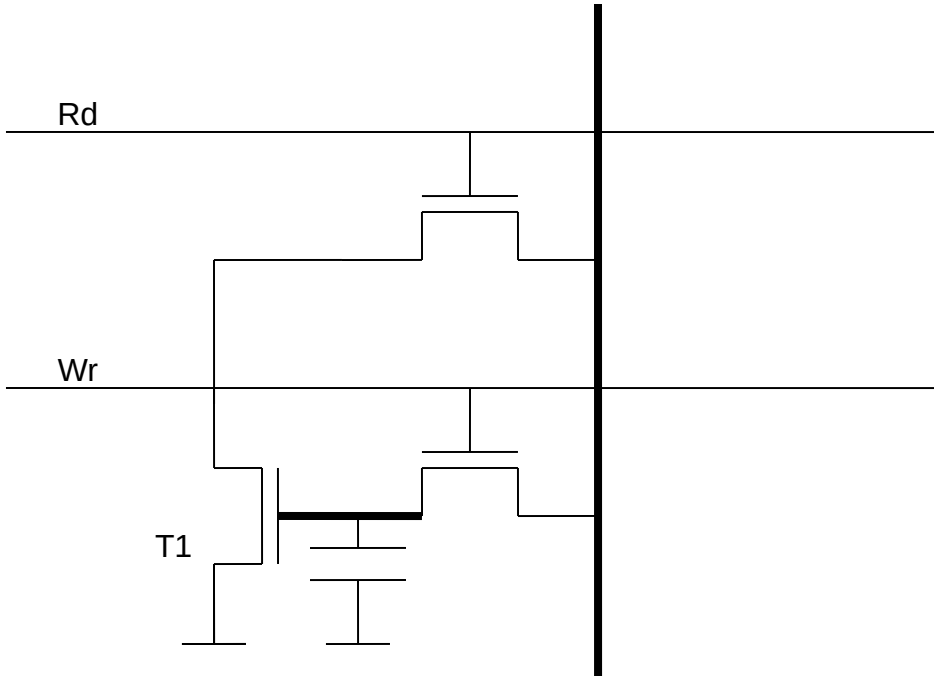
- DRAM



Leckstrom ist Problem, Ladung fliesst ab

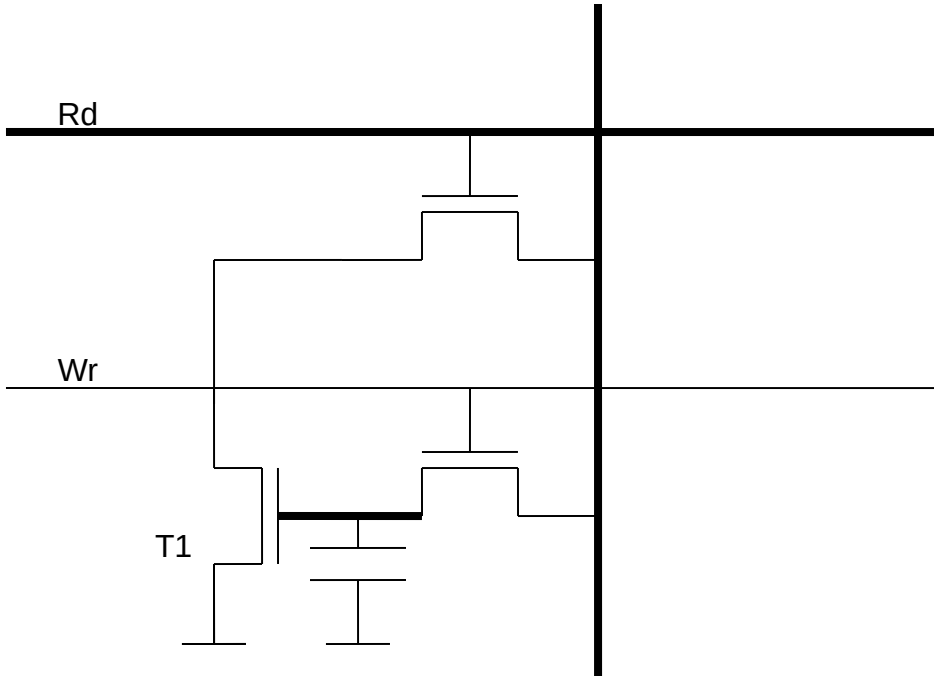
Read

- DRAM



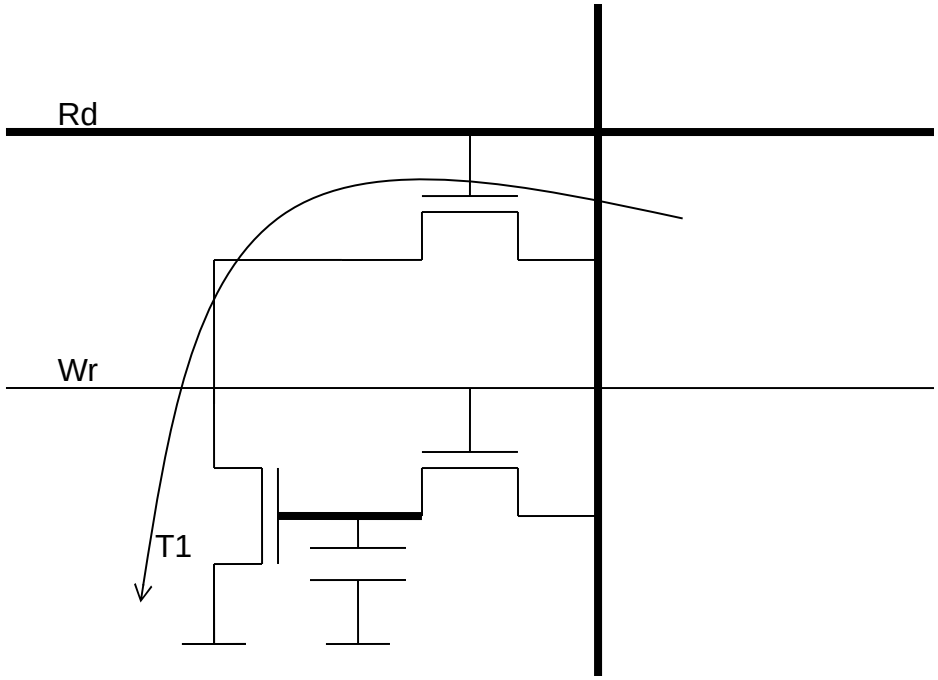
Read

- DRAM



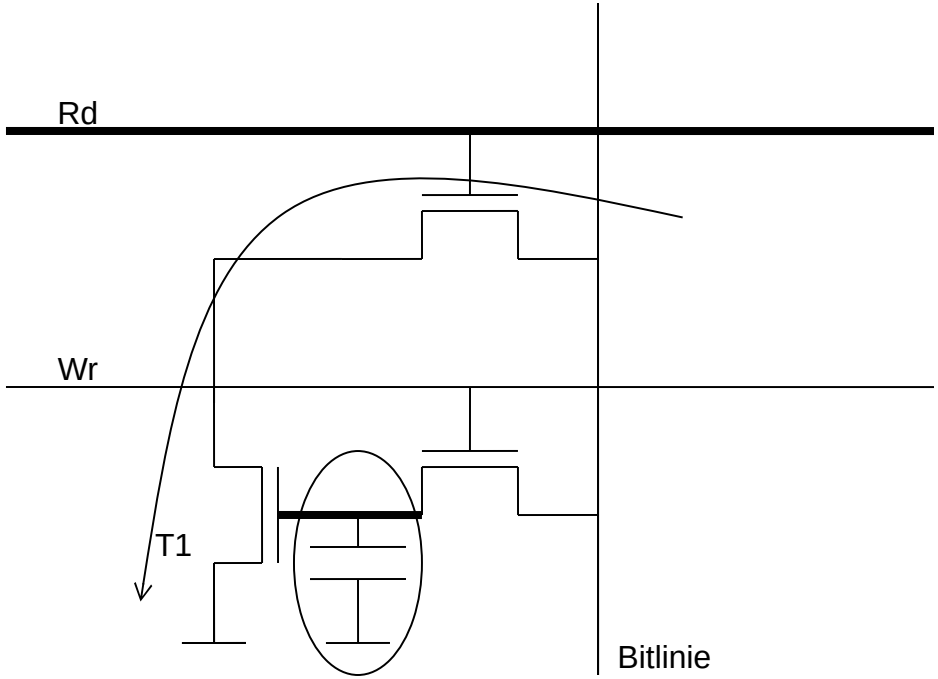
Read

- DRAM



Read

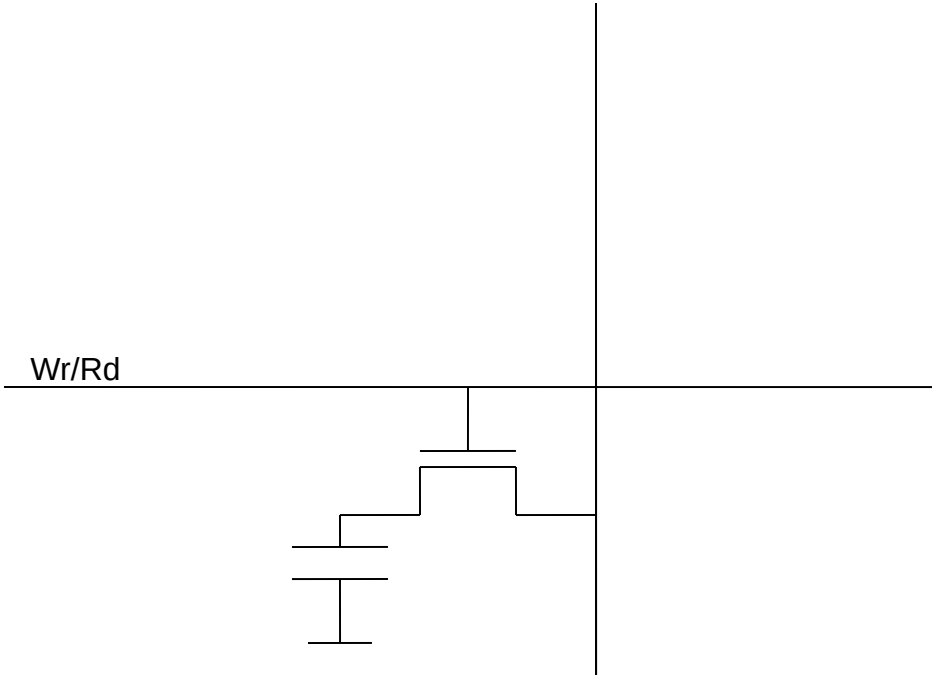
- DRAM



Ladung bleibt erhalten

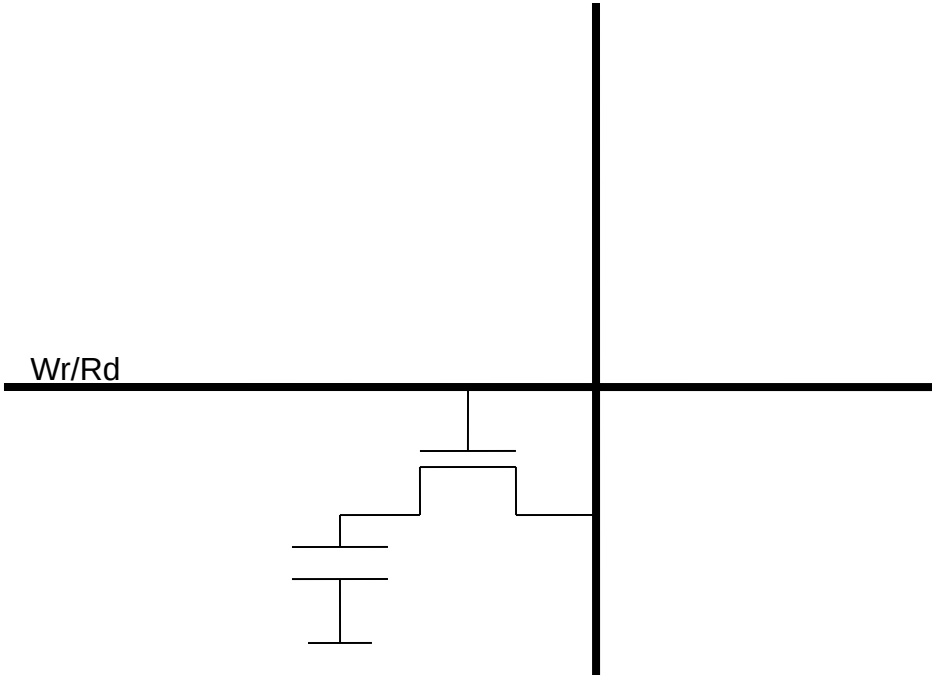
DRAM – Variante 2

- Realisierung 2
- 1-transistor DRAM



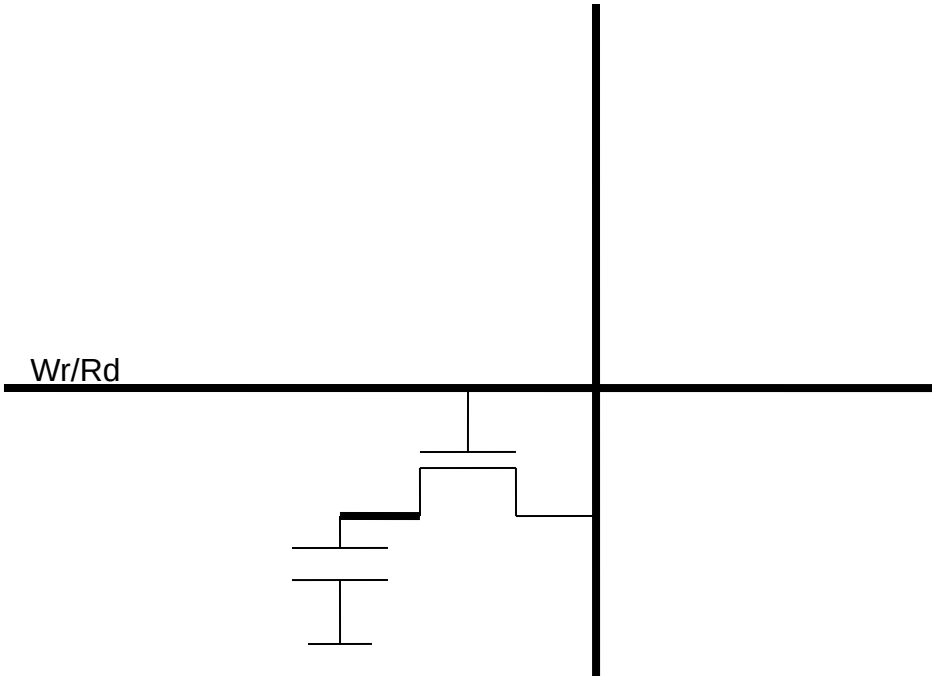
Write

- Write ist einfach



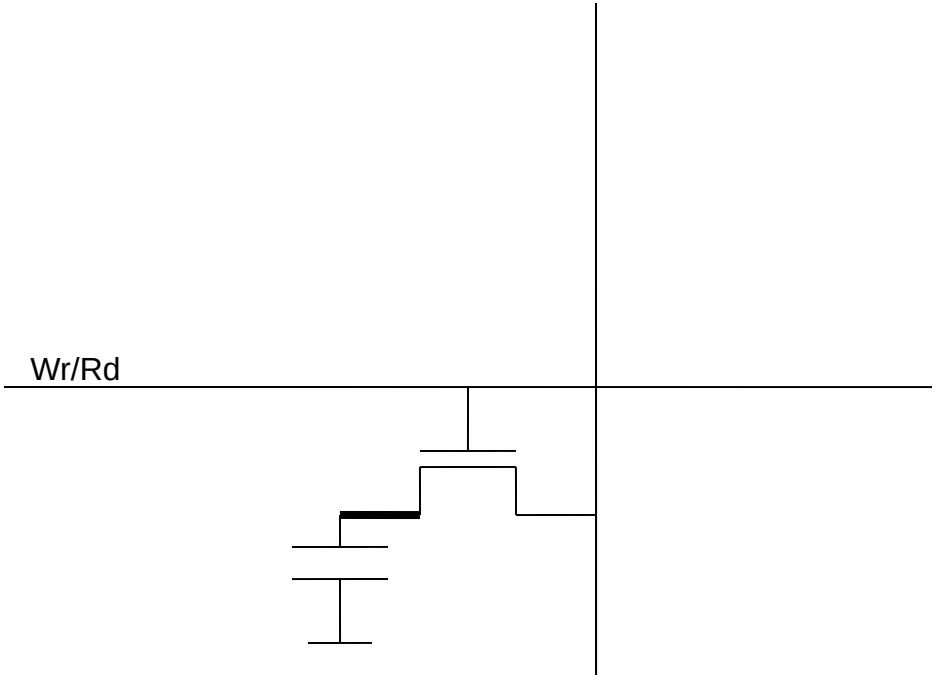
Write

- Write ist einfach



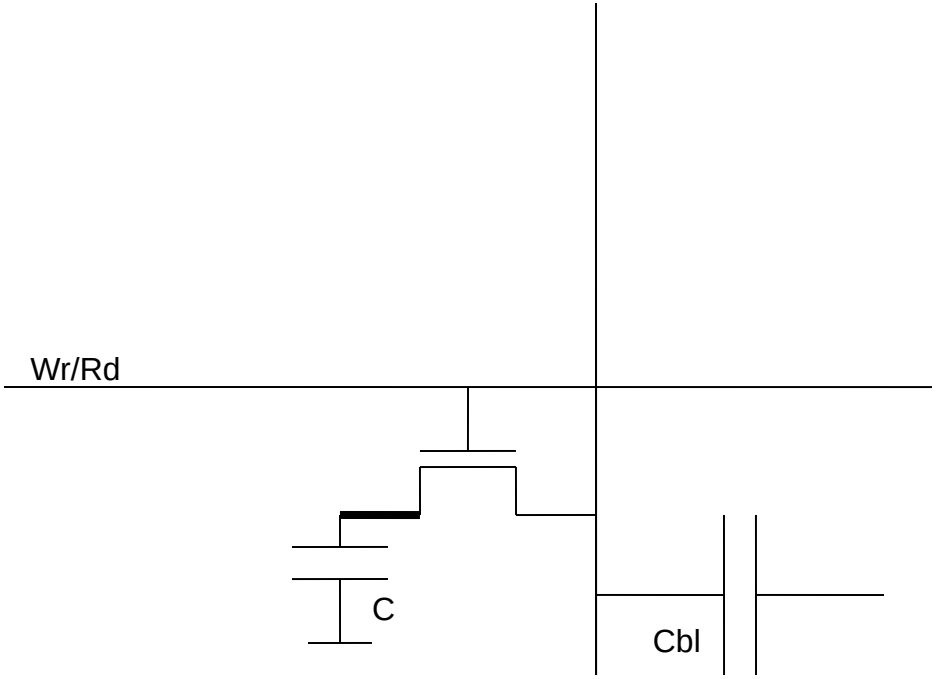
Write

- Write ist einfach



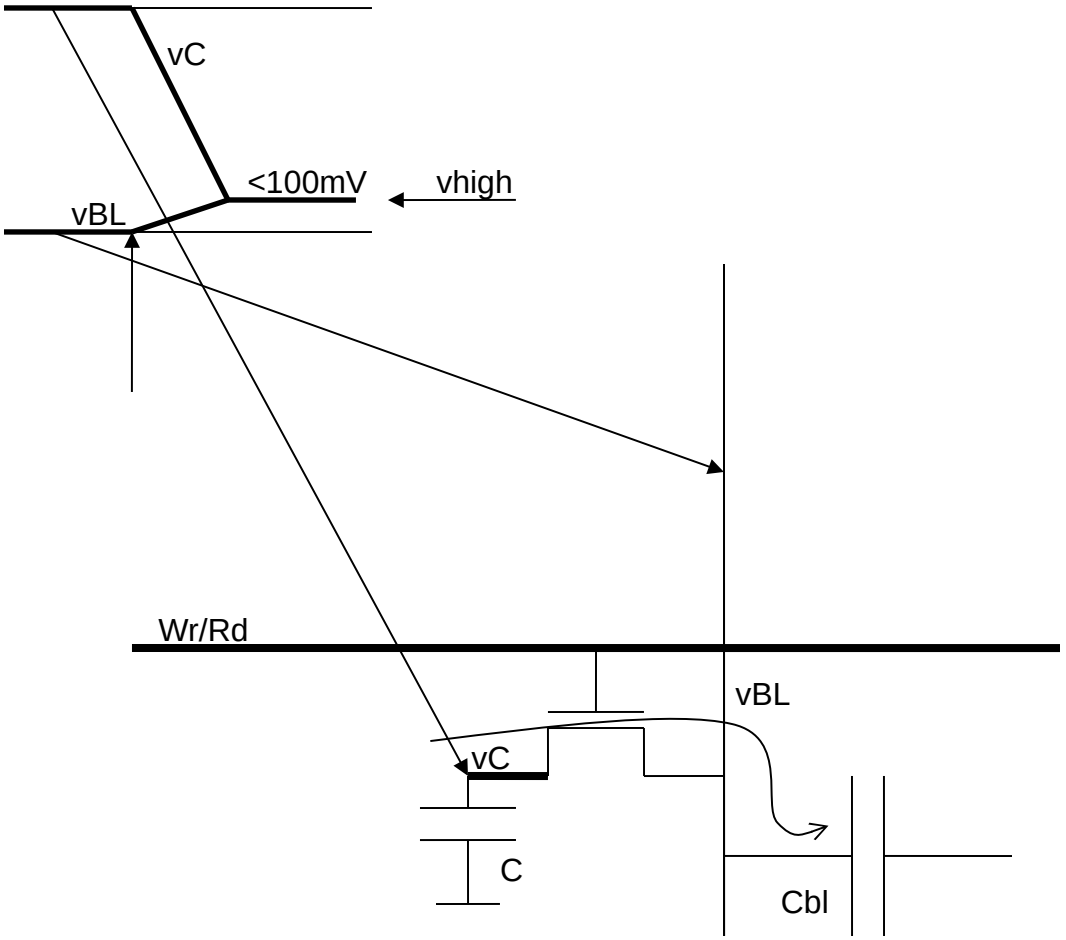
Read

- Lesezyklus
- Bit-line Kapazitäten Cbl führen zum Entladen von RAM Kapazitäten C



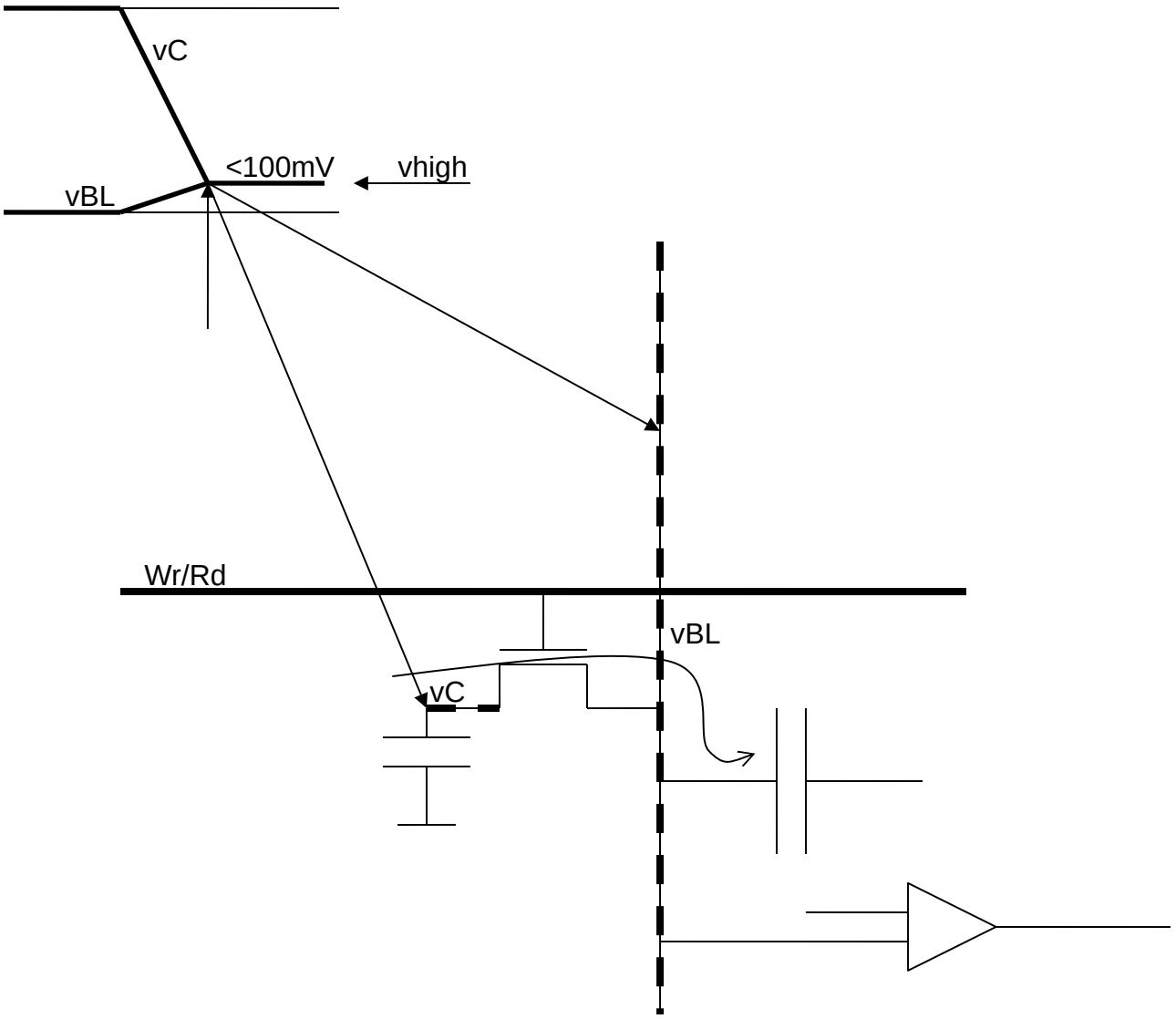
Read

- Bit-line Kapazitäten Cbl führen zum Entladen von RAM Kapazitäten C



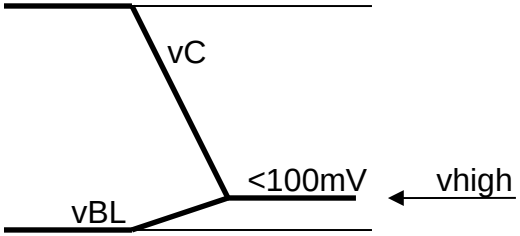
Read

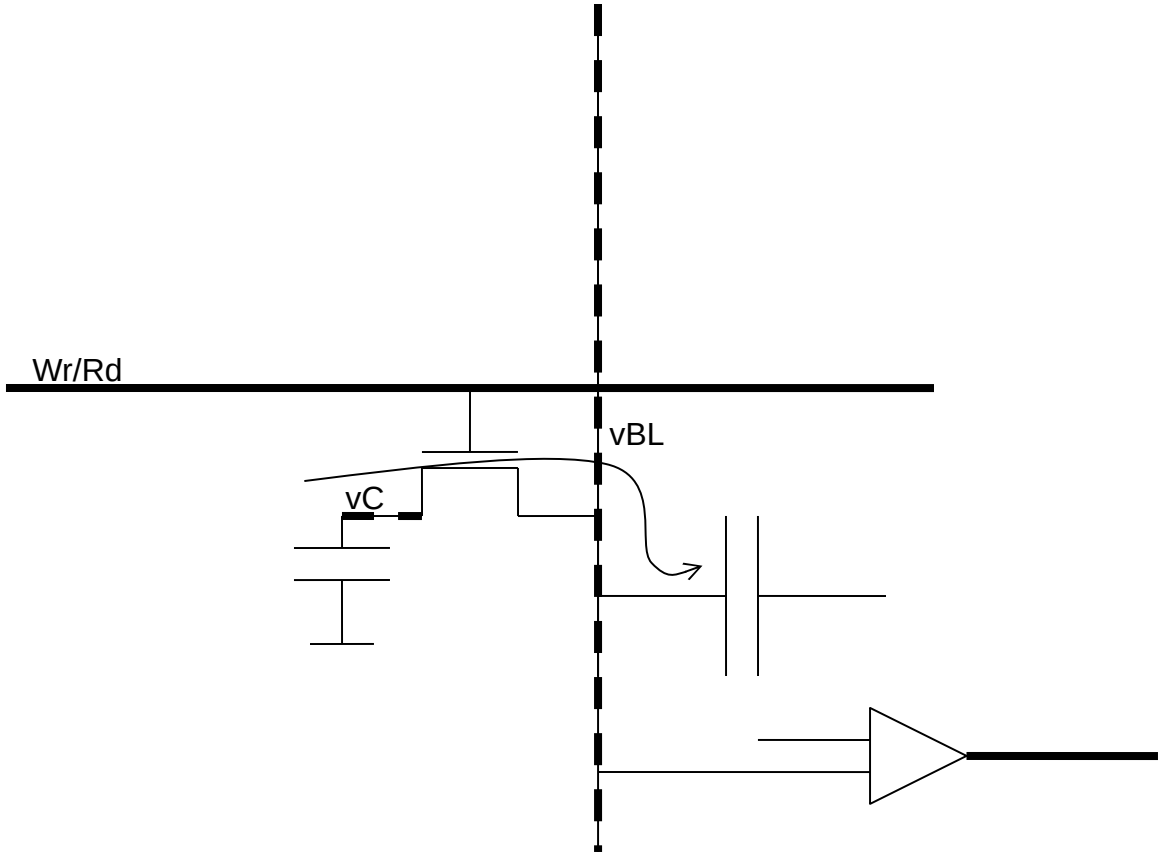
- Bit-line Kapazitäten Cbl führen zum Entladen von RAM Kapazitäten C



Read

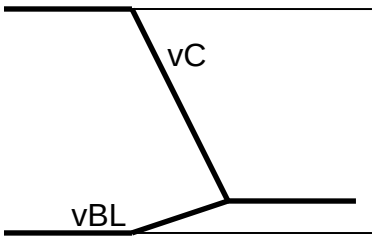
- Ein Verstärker oder analoger Komparator wird benutzt um Vhigh als logische 1 zu erkennen

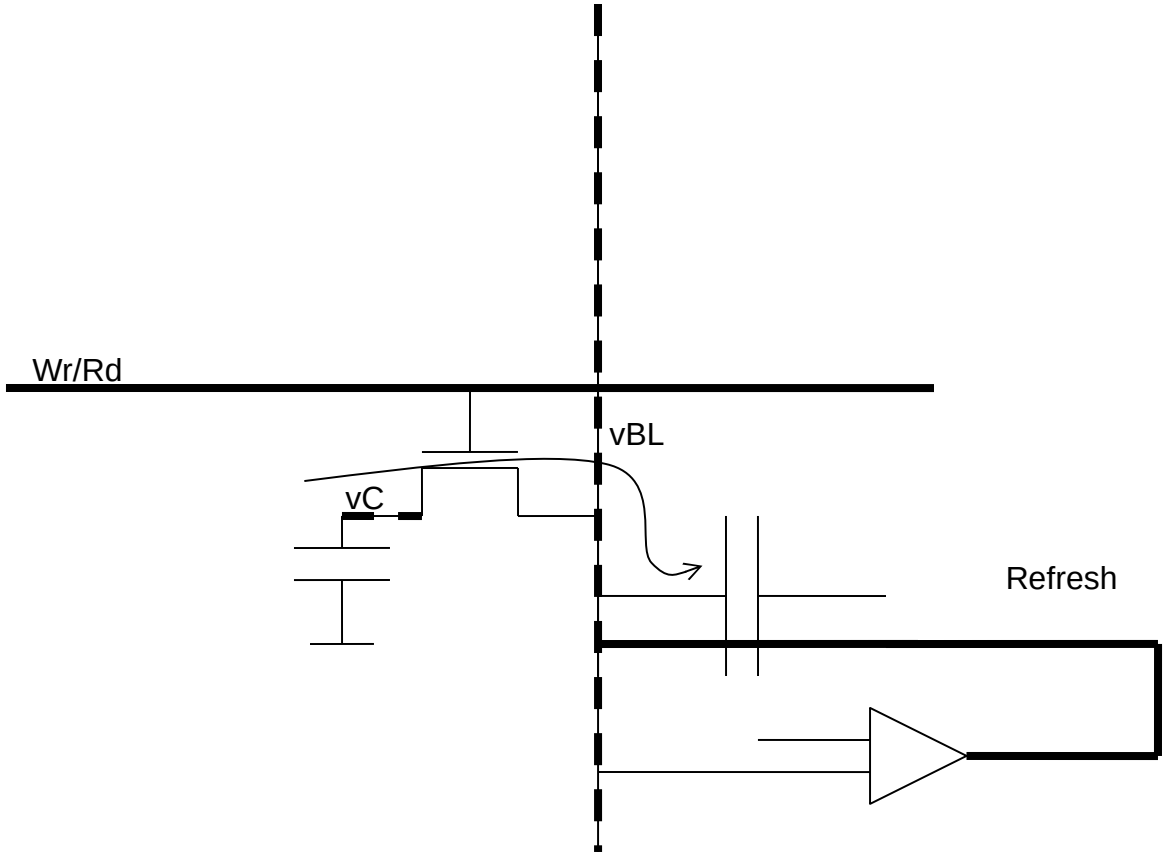




Refresh

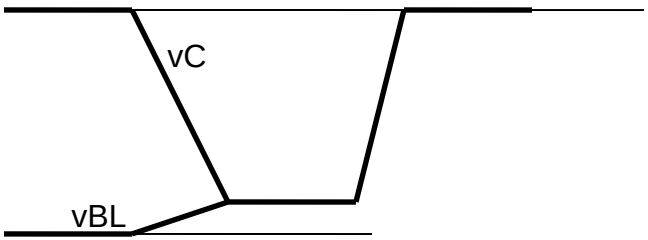
- Refresh

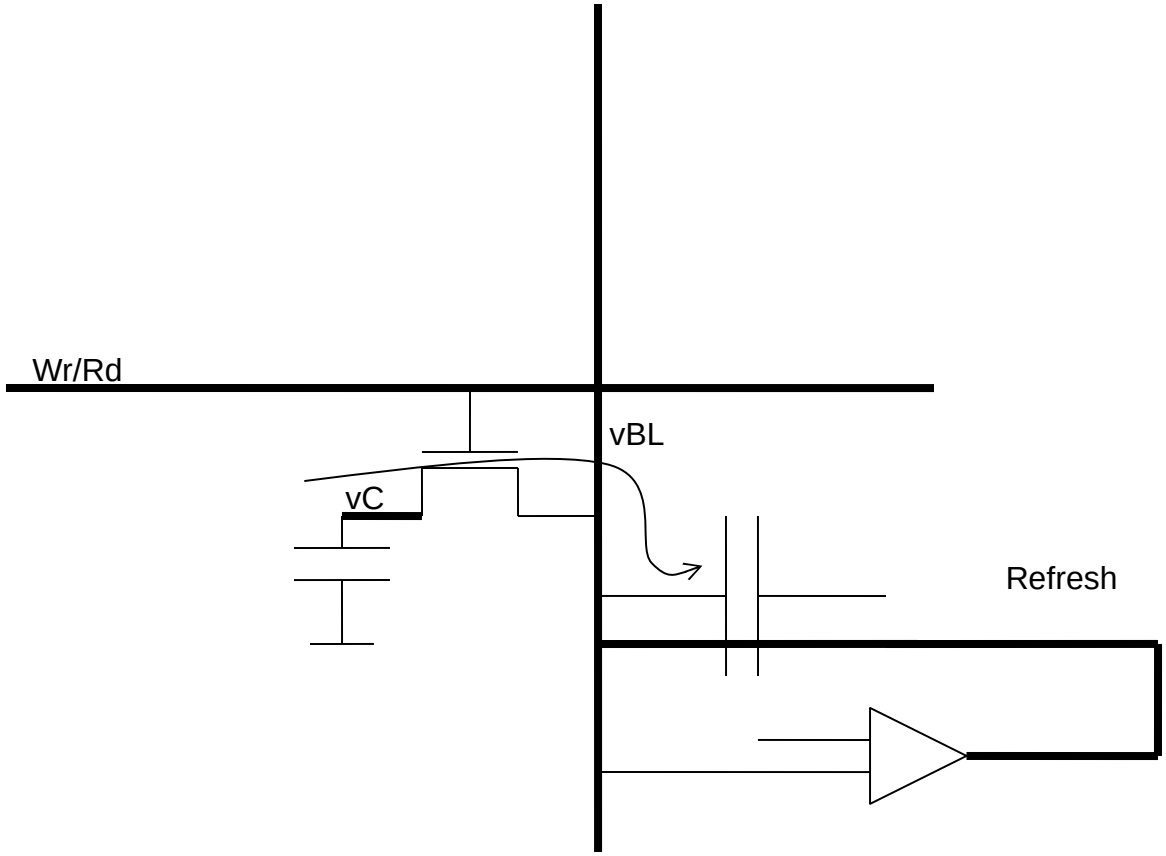




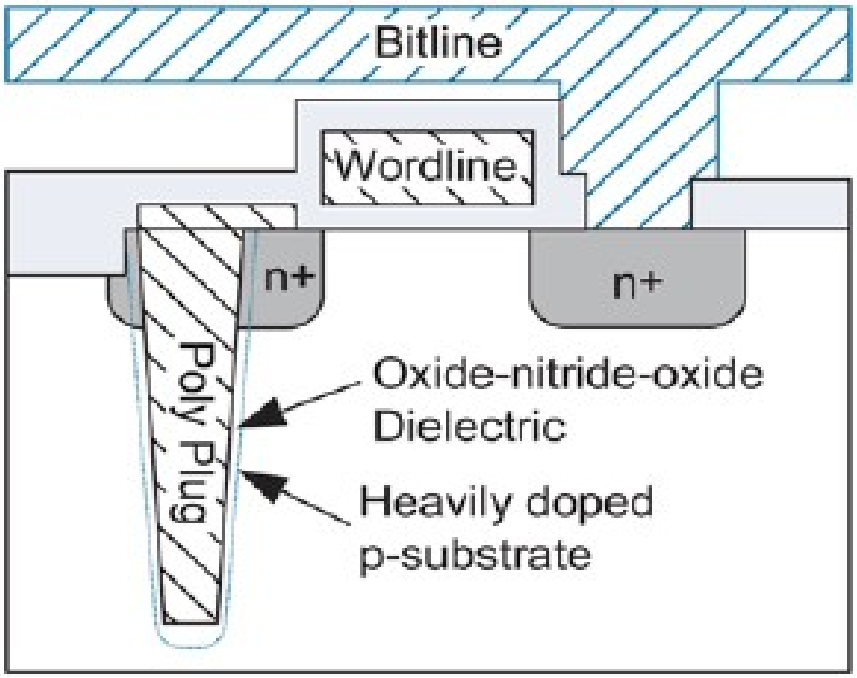
Refresh

- Refresh





Implementierung in Silizium



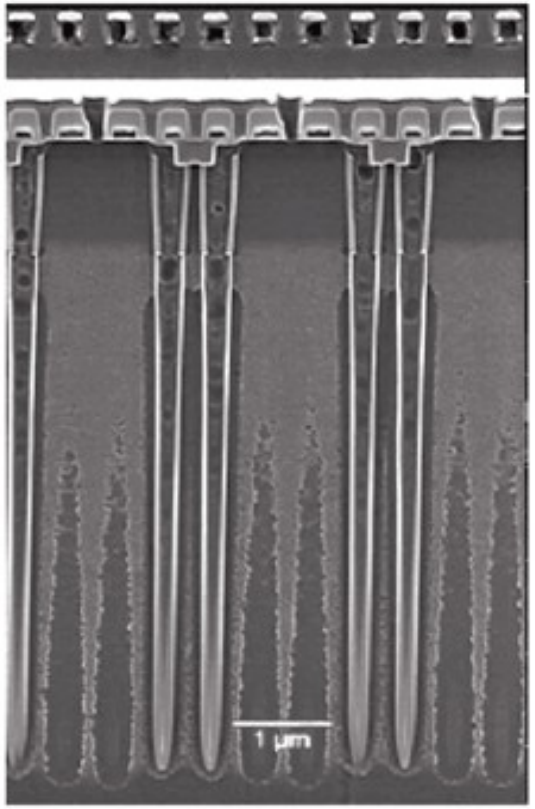
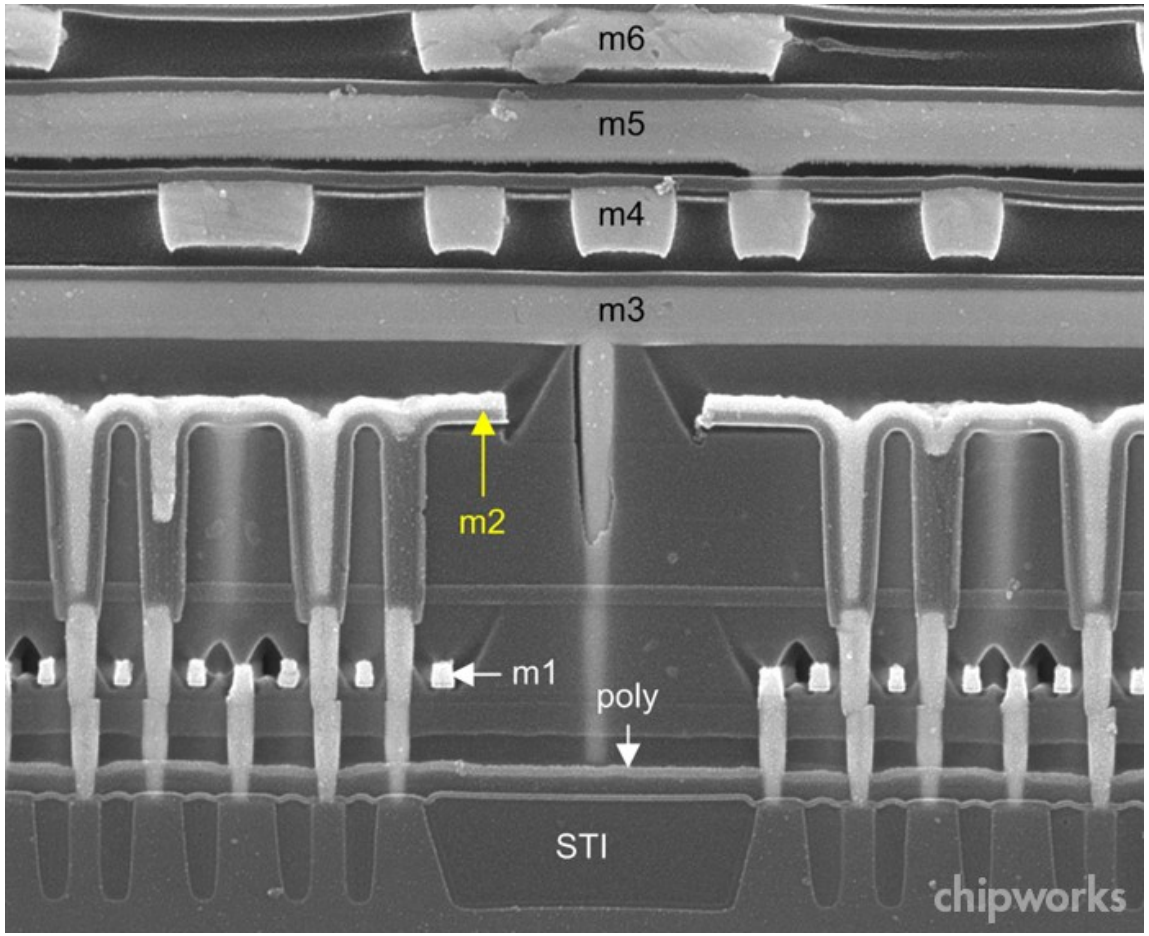


FIG 11.27 Trench capacitor

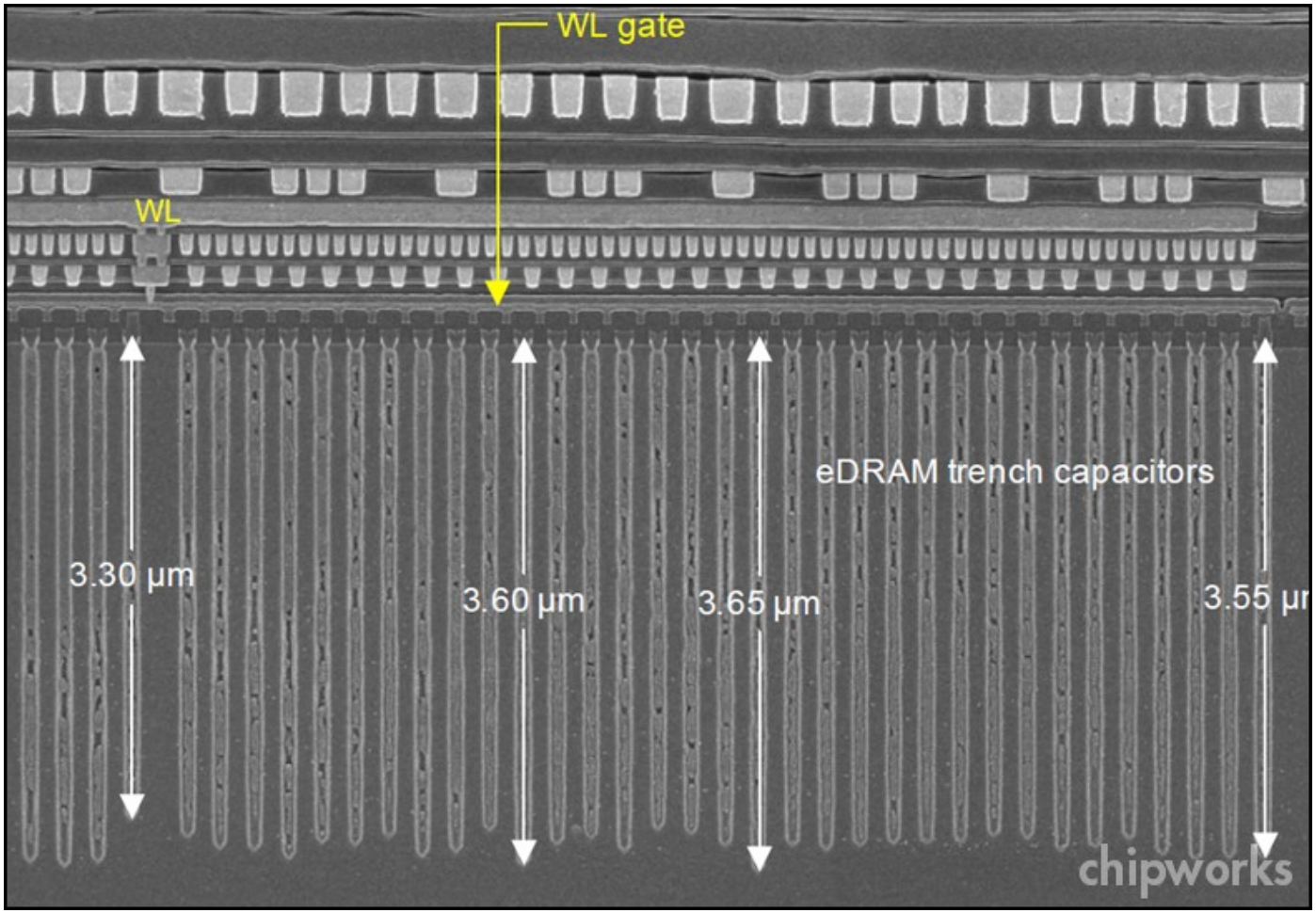
Implementierung in Silizium



Embedded DRAM in Nintendo Wii U GPU fabbed by Renesas (45-nm)

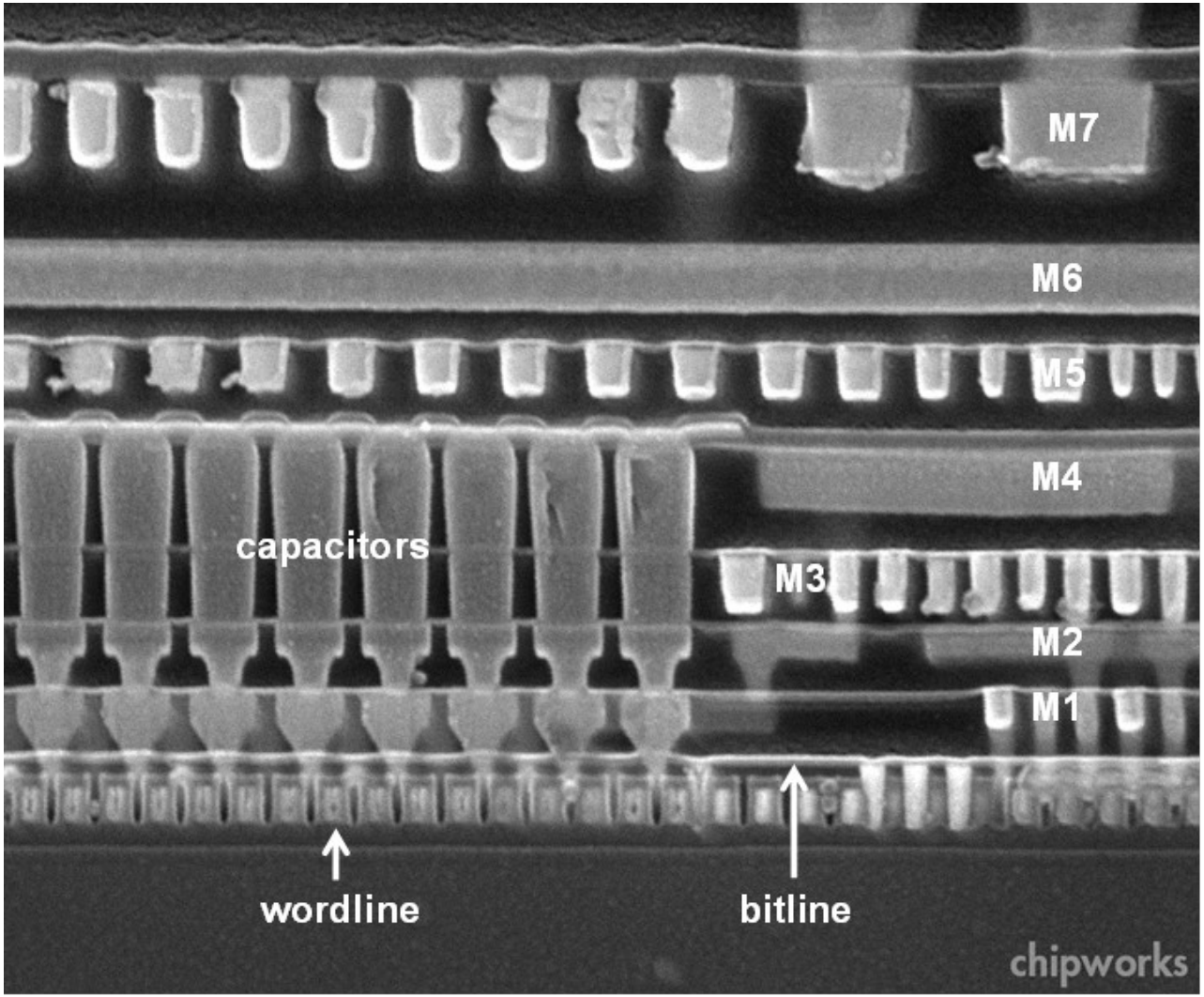
http://chipworksrealchips.blogspot.de/2014/02/intels-e-dram-shows-up-in-wild.html

Implementierung in Silizium



Embedded DRAM in IBM Power 7+ microprocessor (32-nm)

Implementierung in Silizium

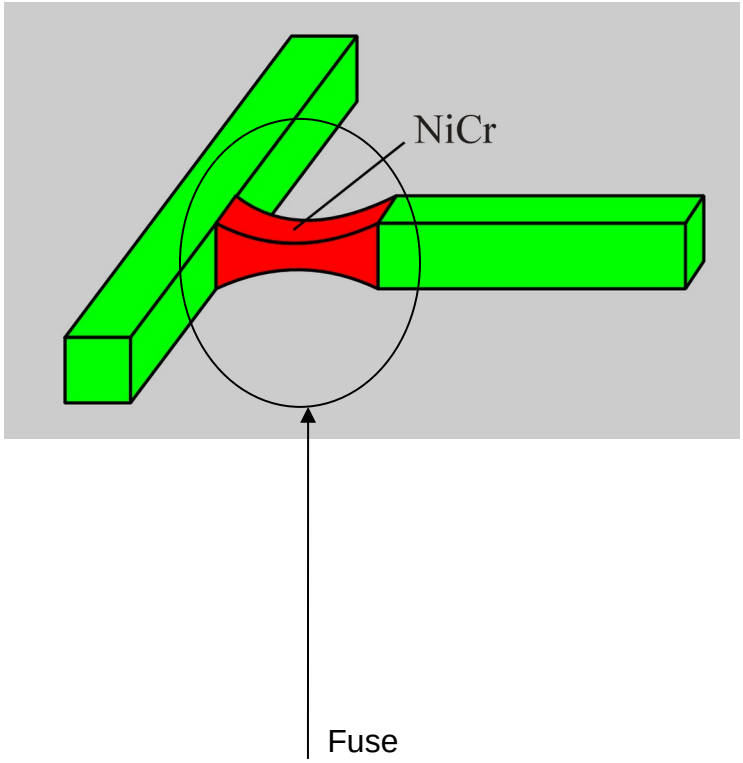


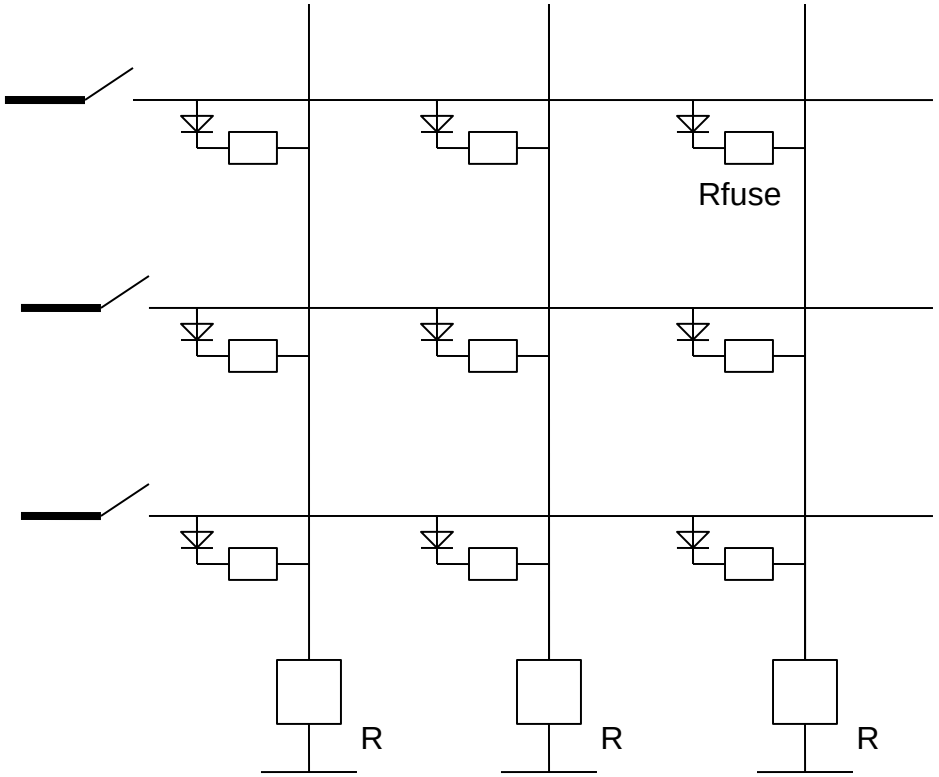
Intel’s 22-nm embedded DRAM stack

Permanentspeicher (non-volatile)

Programmiert vor Verwendung

PROM

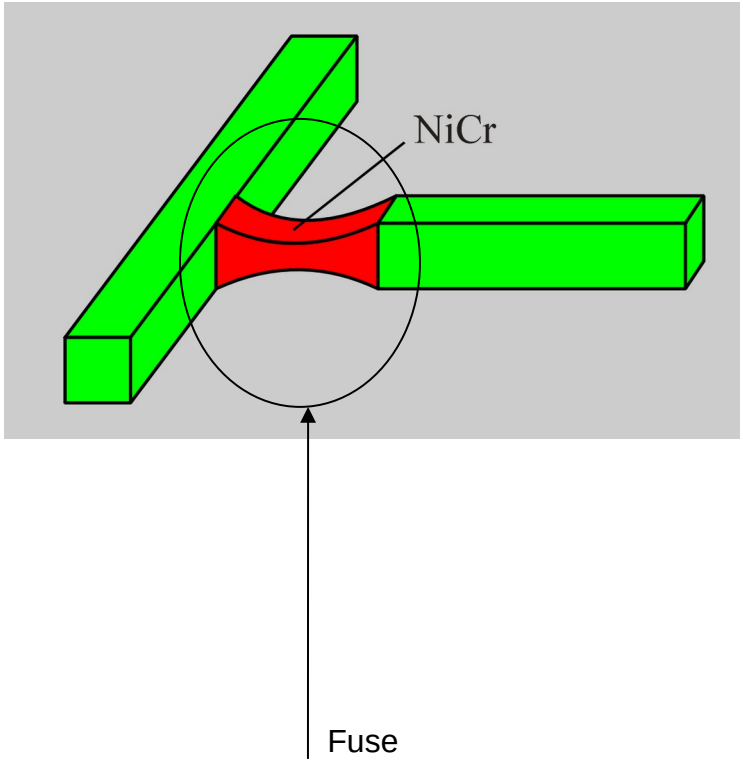


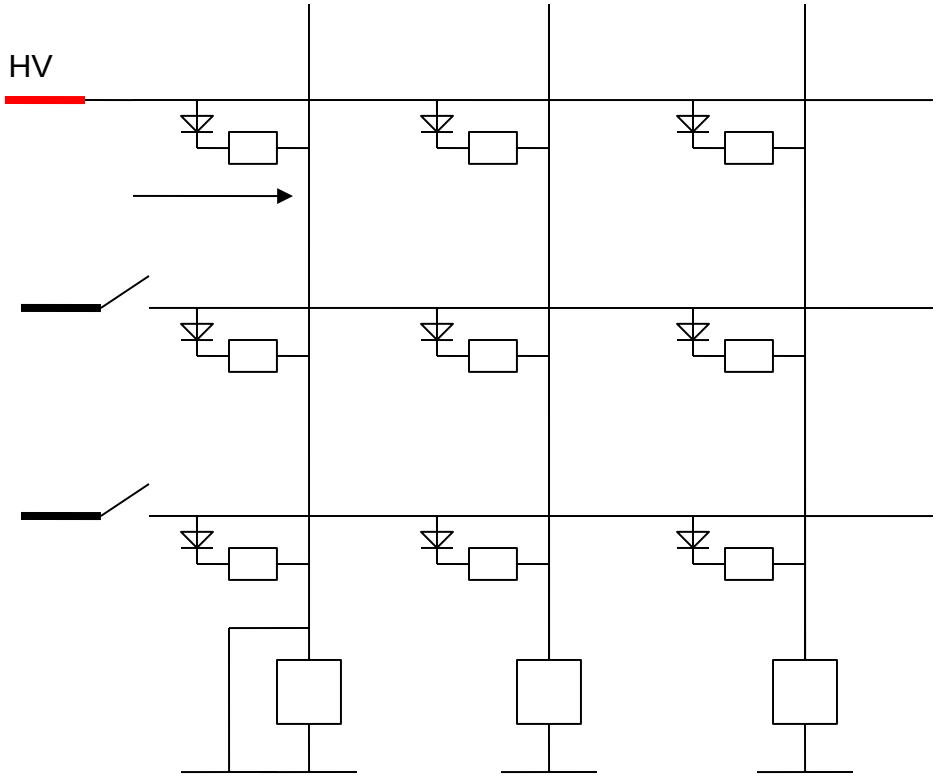


Programmierung

Programmiert vor Verwendung

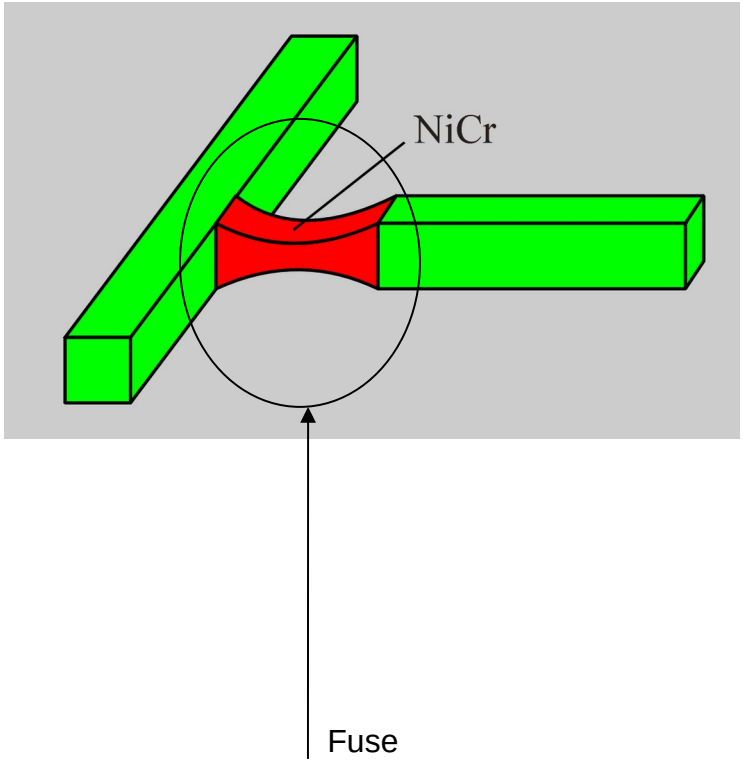
PROM

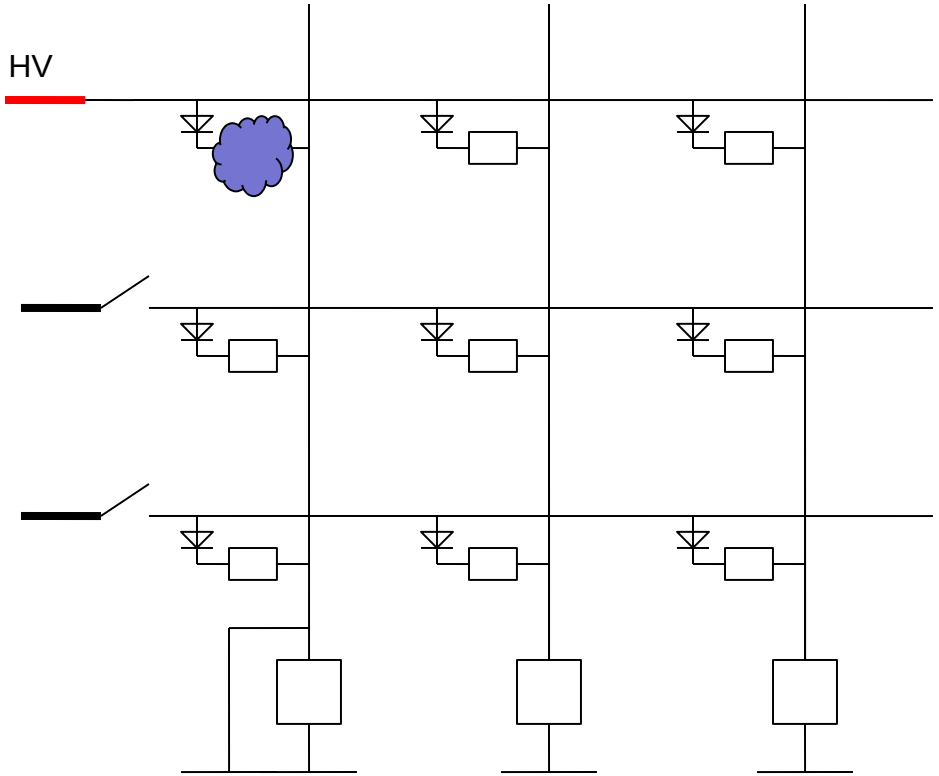




Programmiert vor Verwendung

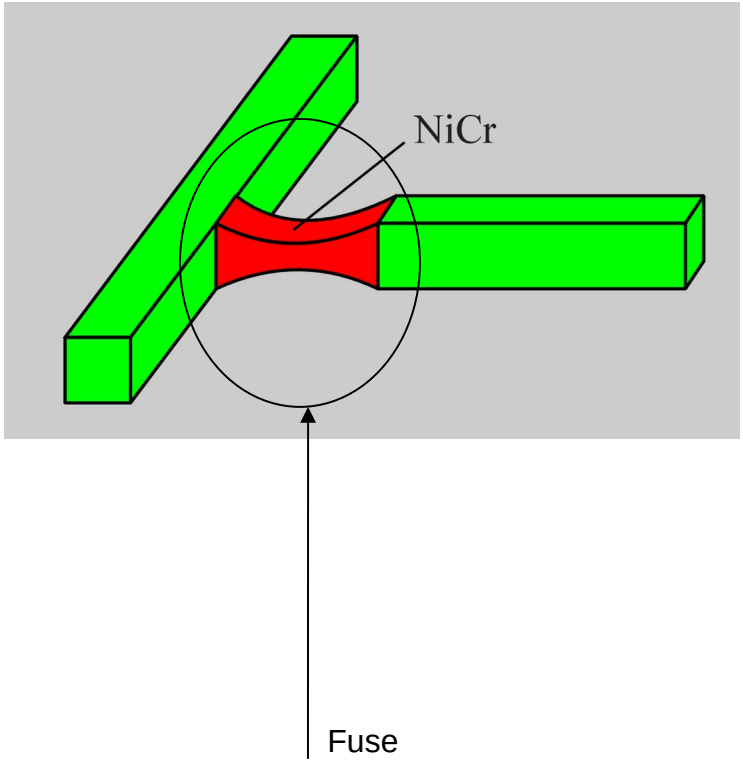
PROM

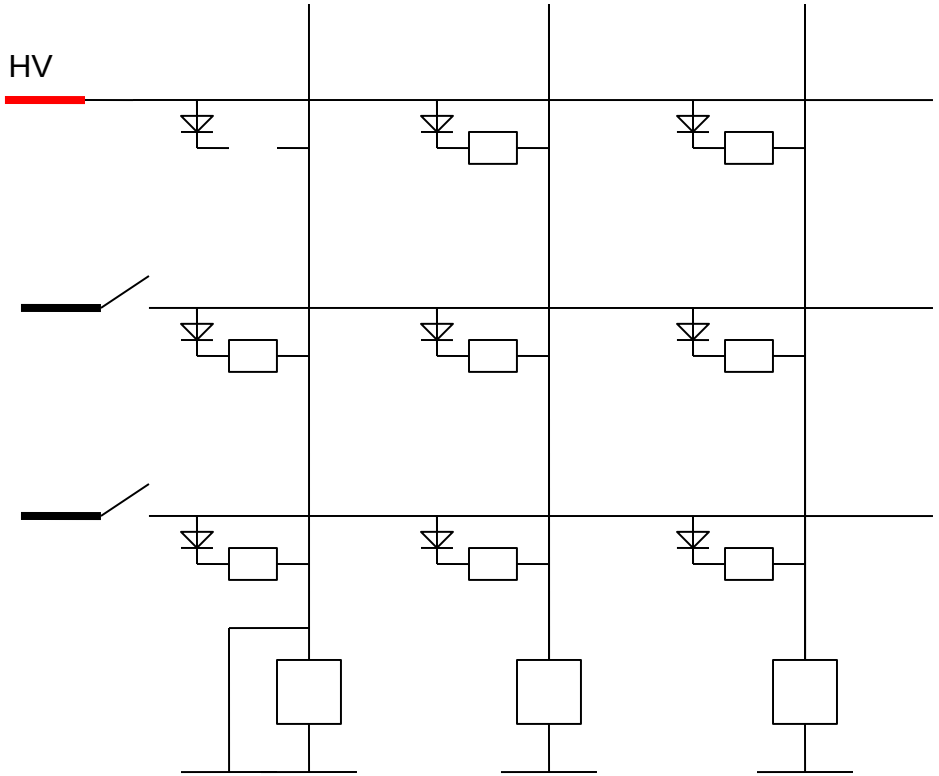




Programmiert vor Verwendung

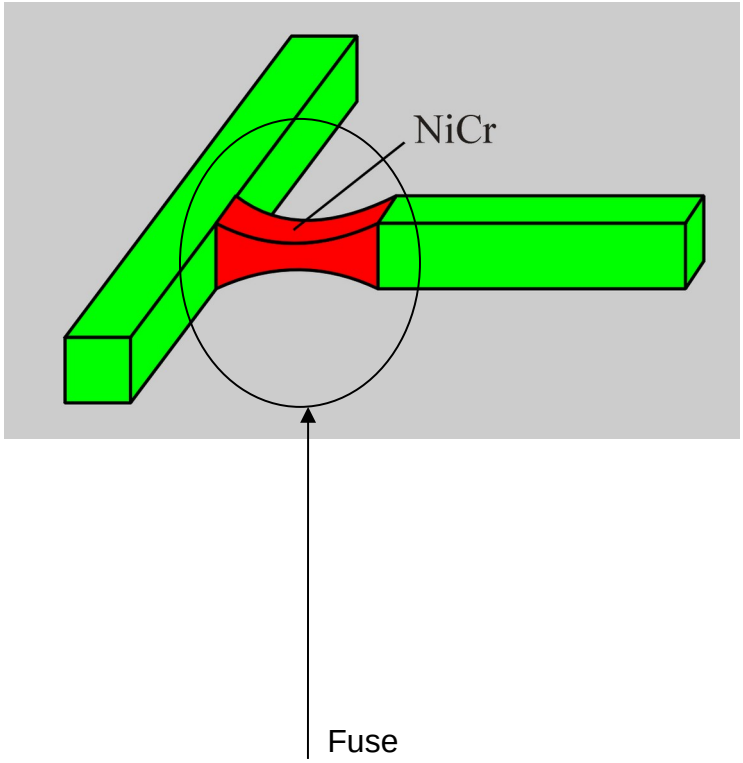
PROM

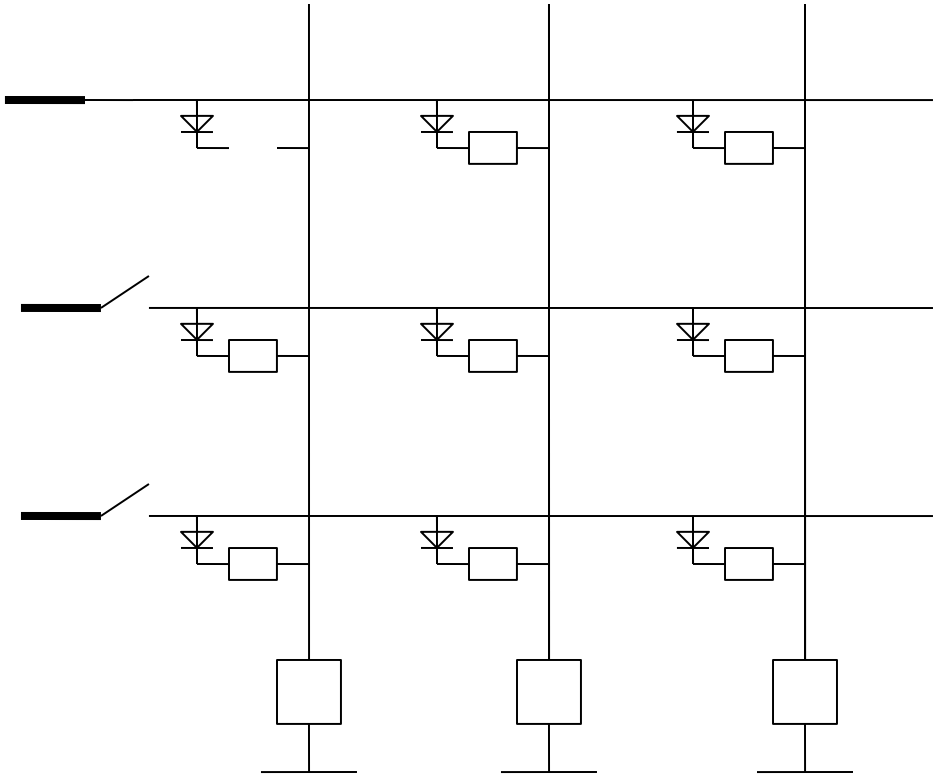




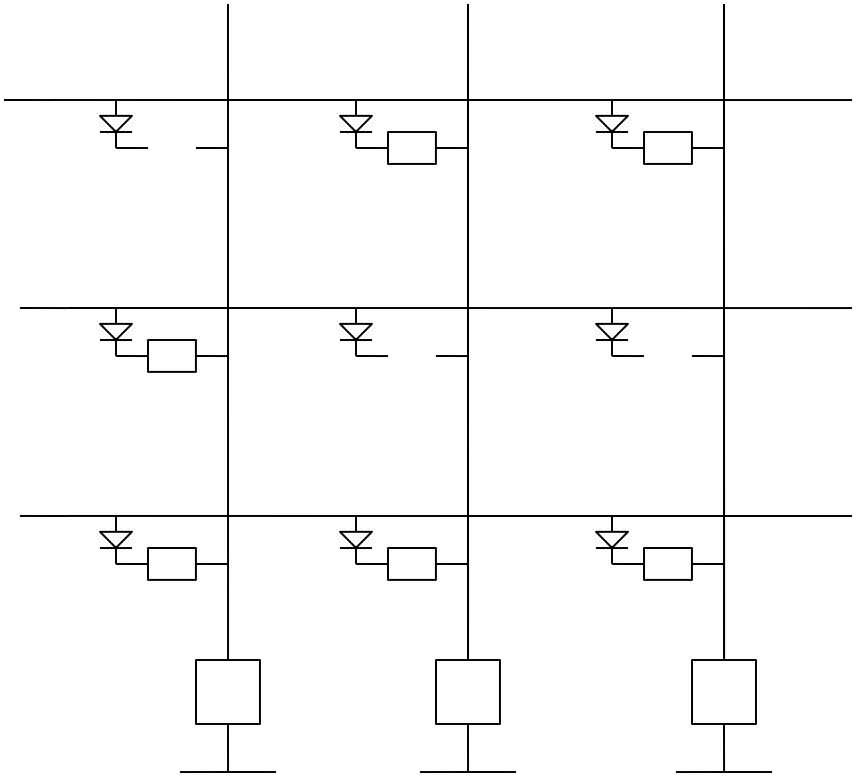
Programmiert vor Verwendung

PROM

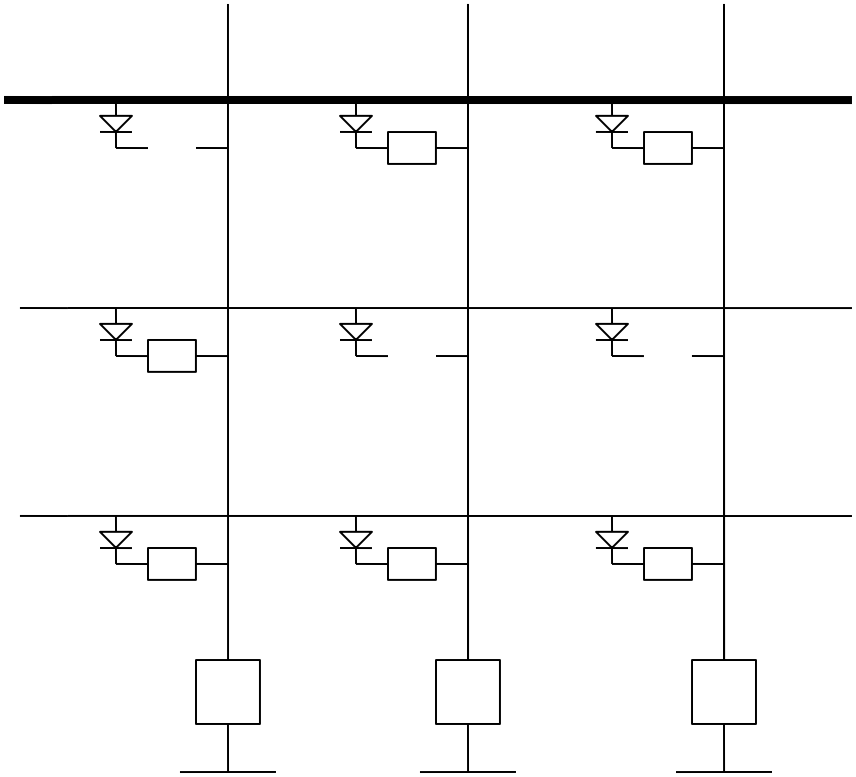




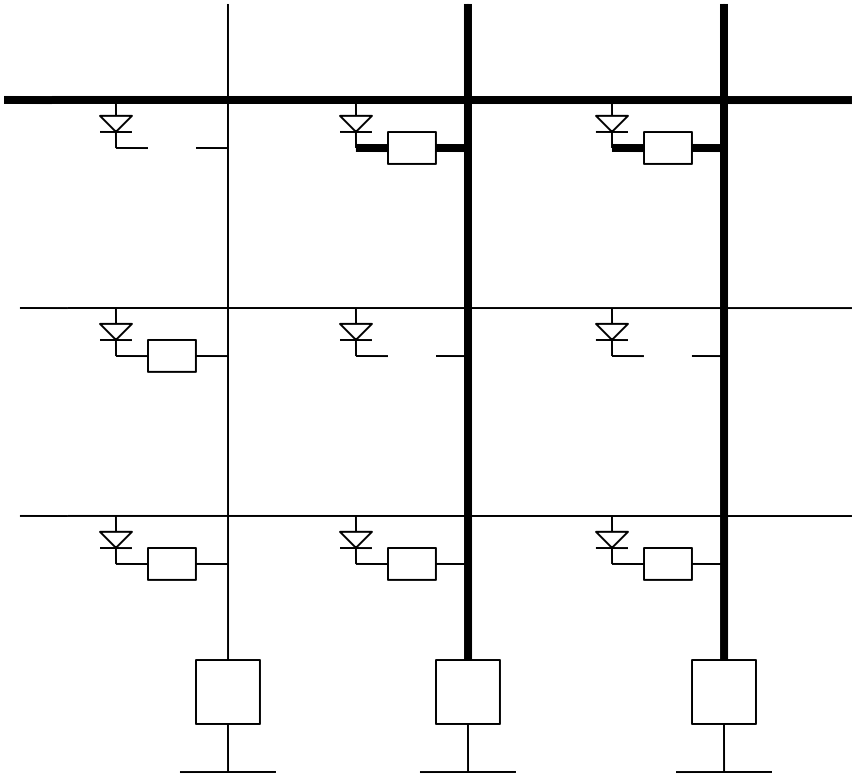
Lesen



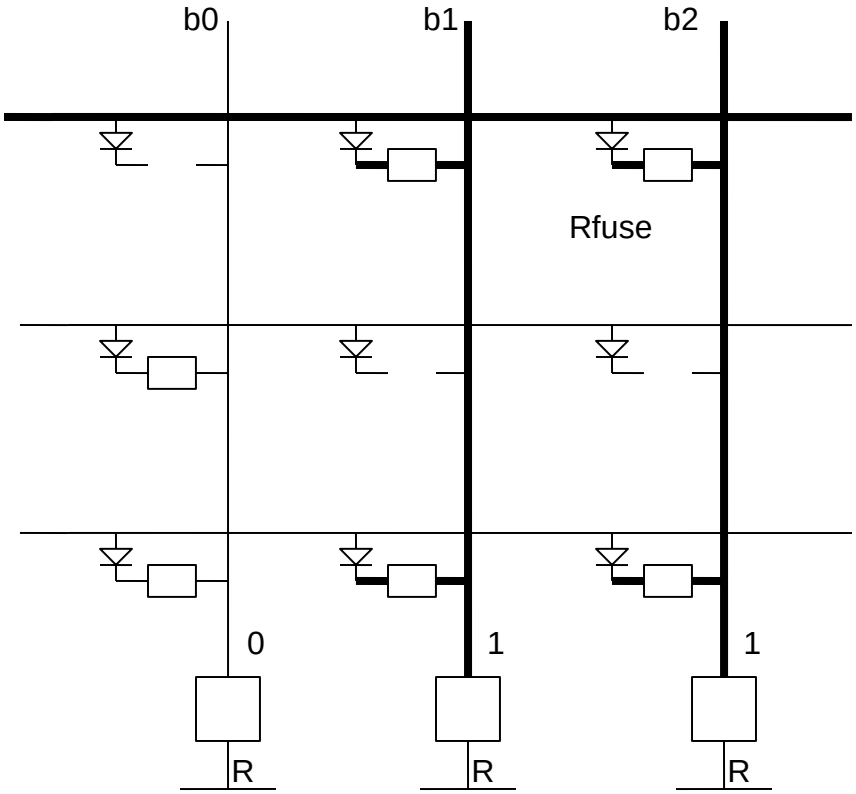
Lesen



Lesen



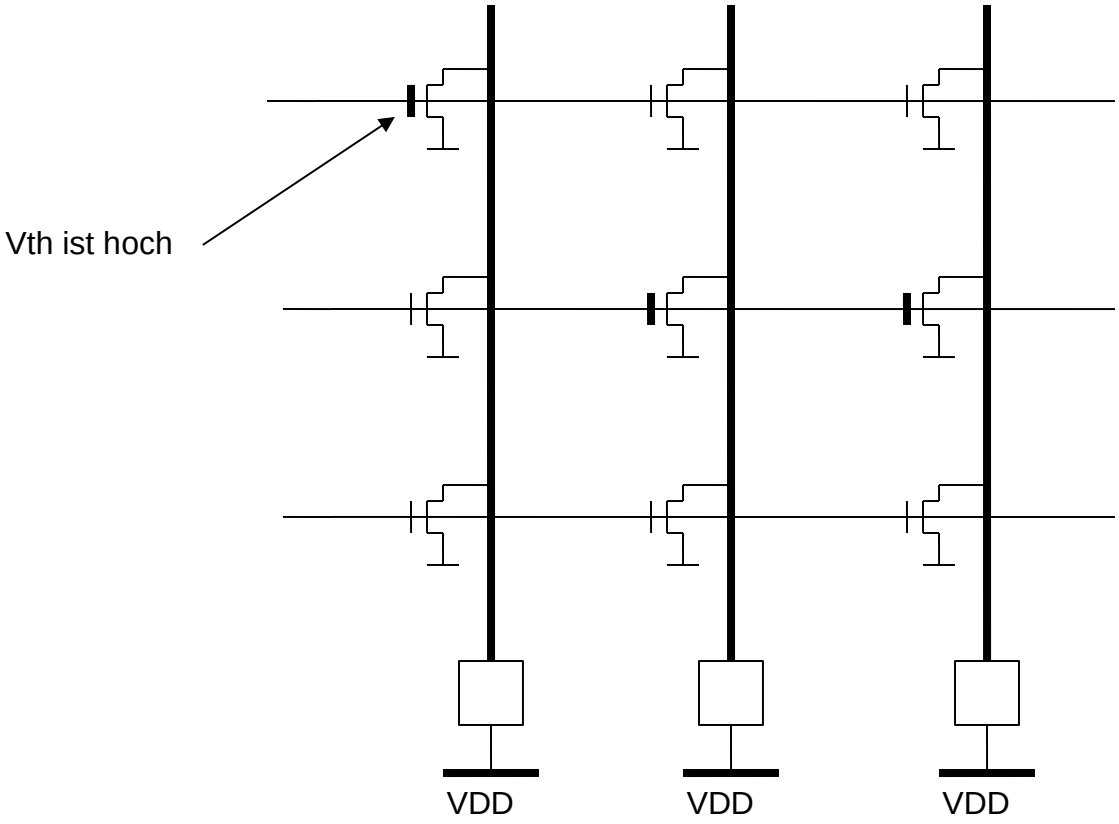
Lesen



EPROM, EEPROM und FLASH

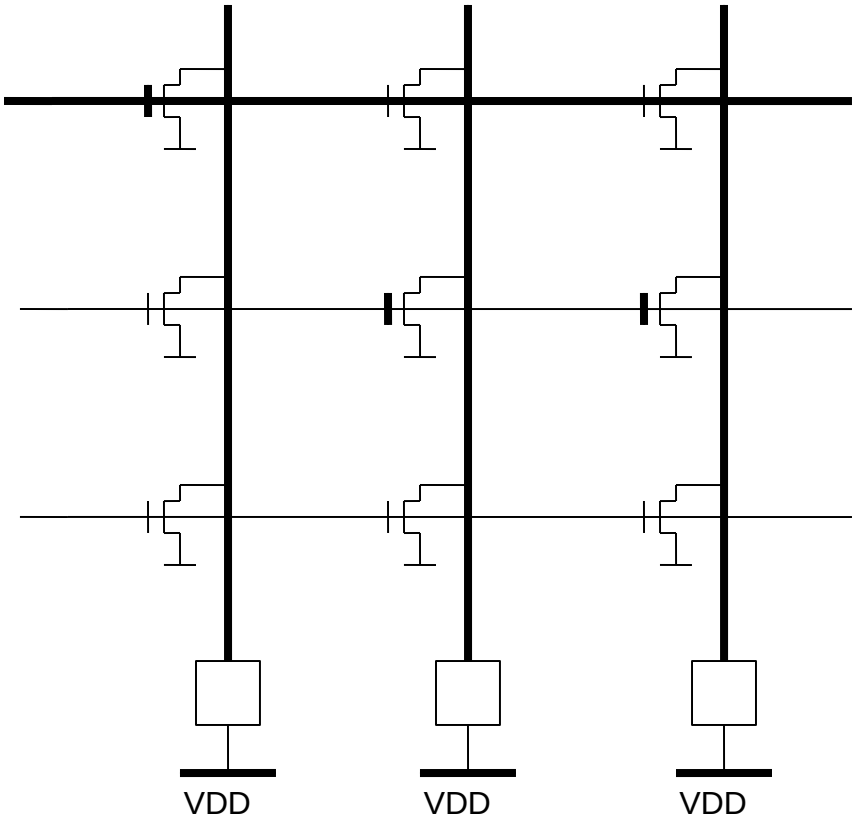
Lesen

- Transistor (ODER-Schaltung)

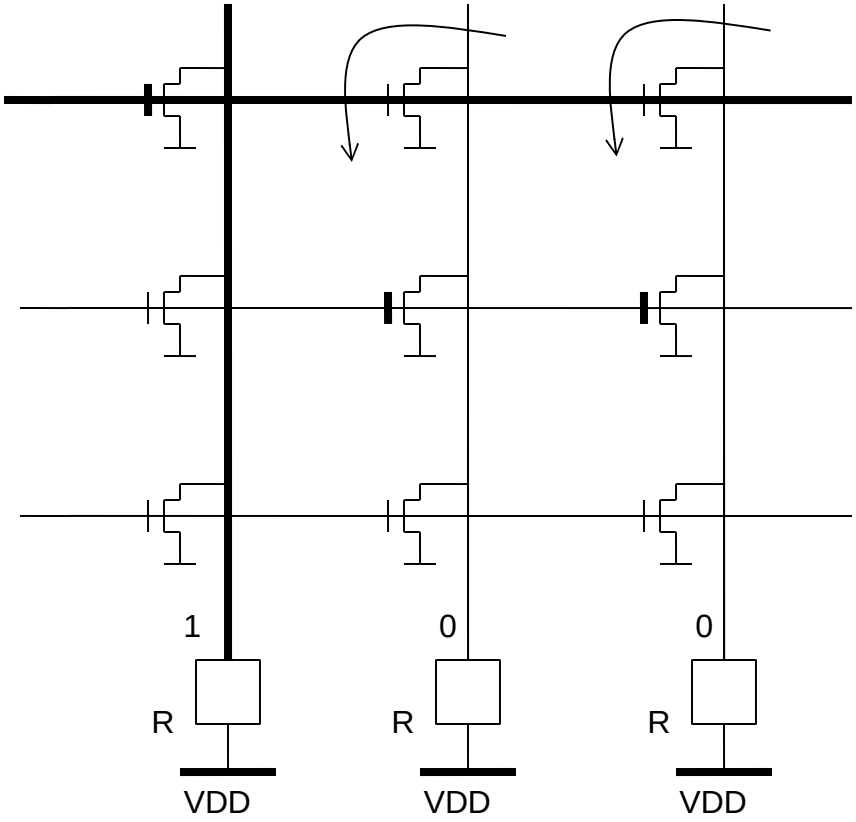


Lesen

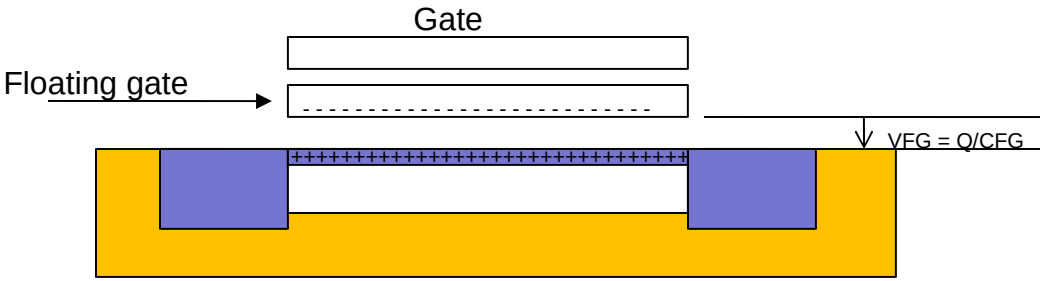
- Transistor (ODER-Schaltung)

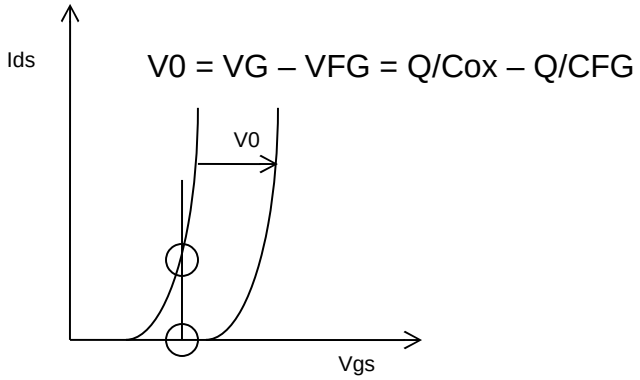


Lesen

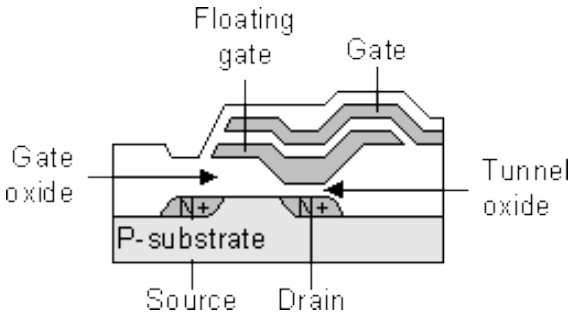


- Durch die gespeicherte Ladung wird die Schwellspannung des Transistors zu höheren Werten verschoben. Das Auslesen erfolgt bei niedrigen Spannungen. Ist U_{th} verschoben, schaltet der Transistor nicht ein.

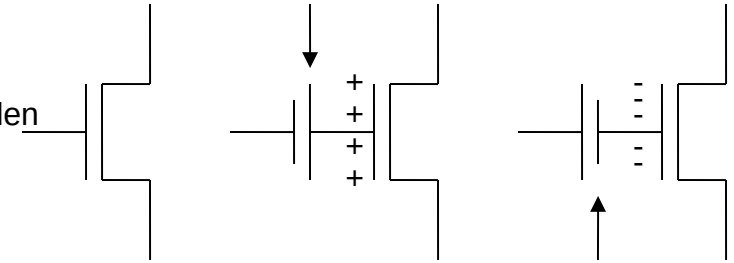




Floating gate negativ aufgeladen

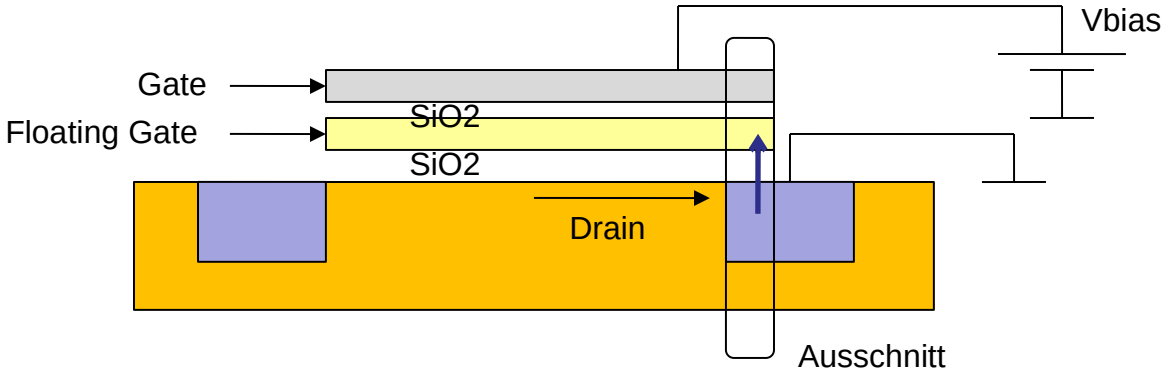


Floating gate nicht aufgeladen

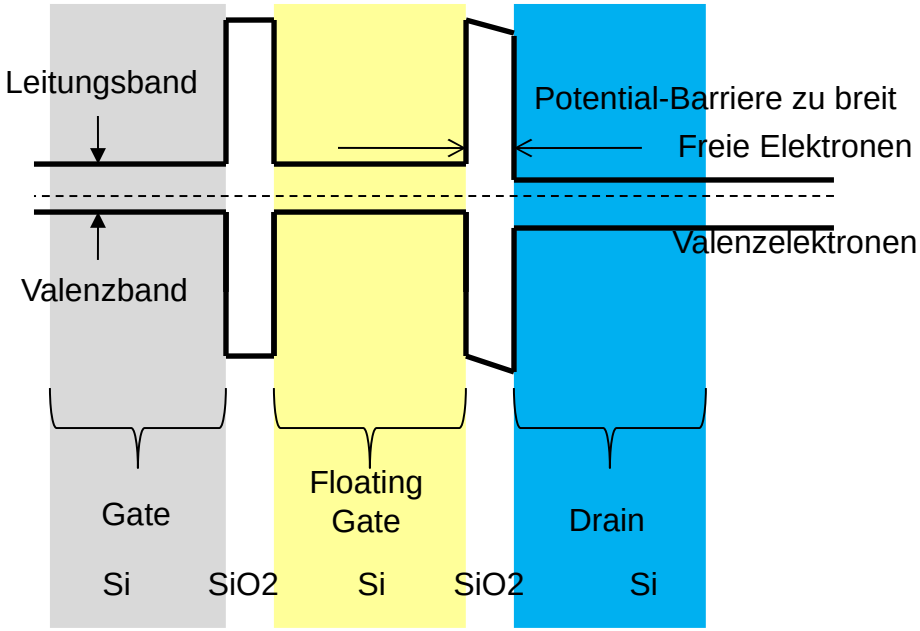


Floating Gate positiv aufgeladen

Programmierung - Tunneleffekt

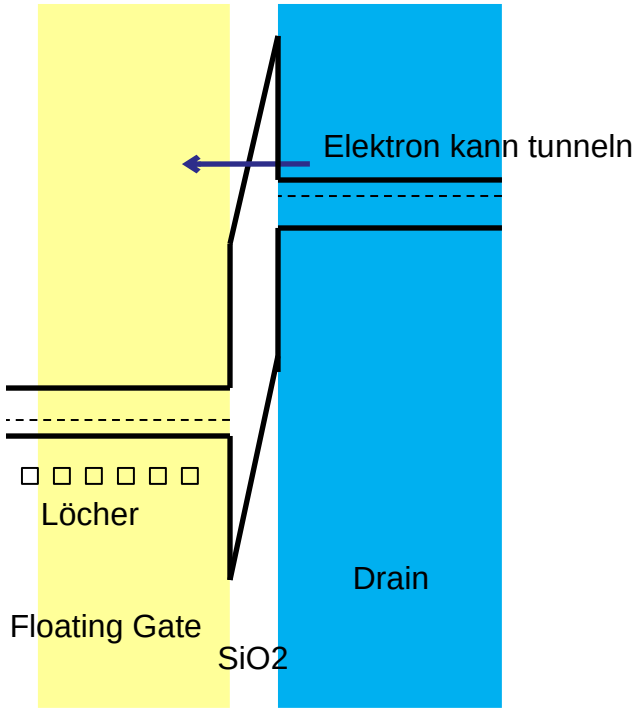


Energiediagramm: Vbias = 0



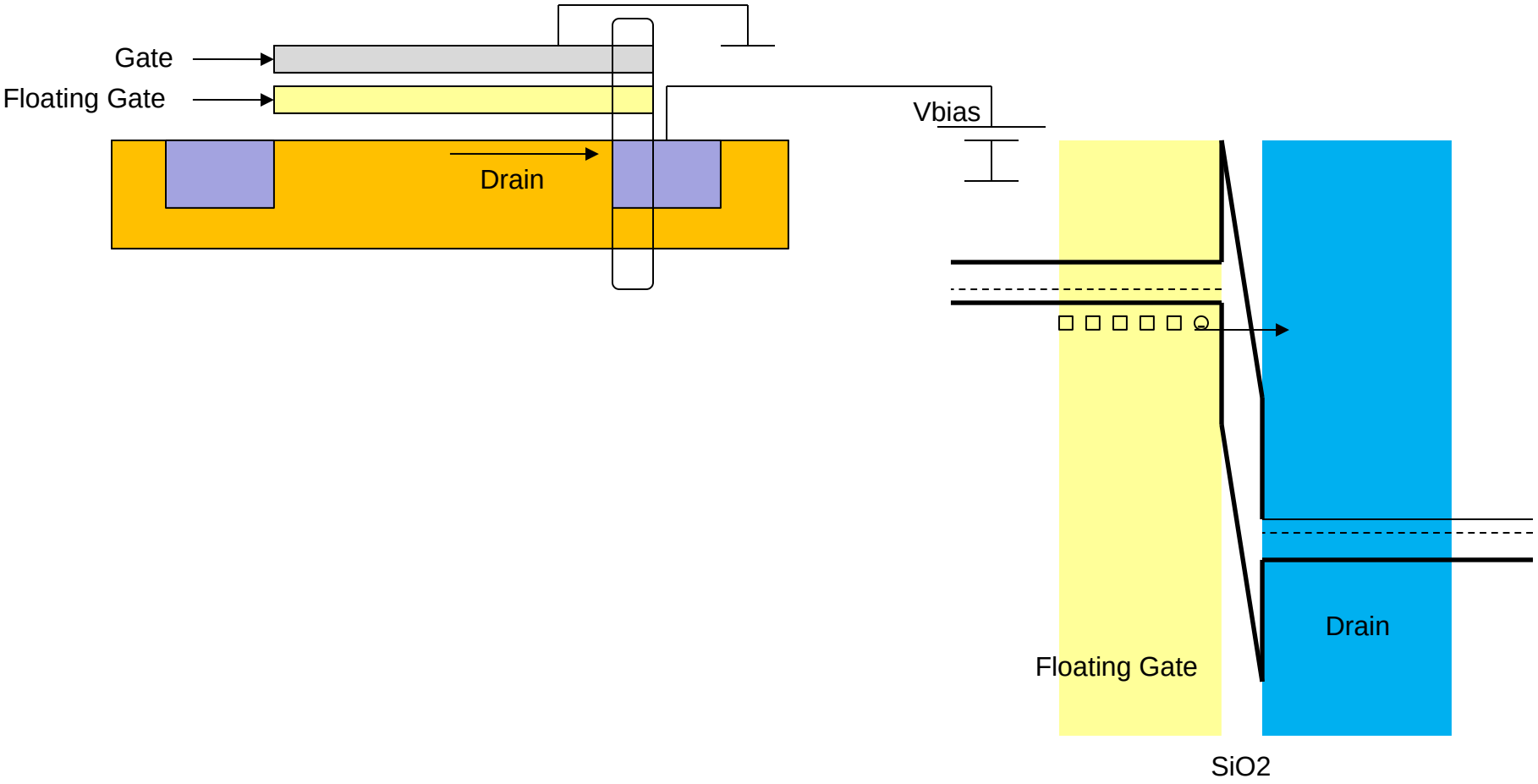
Wegen hoher Spannung wird die Barriere schmäler

Vbias ~ 10V

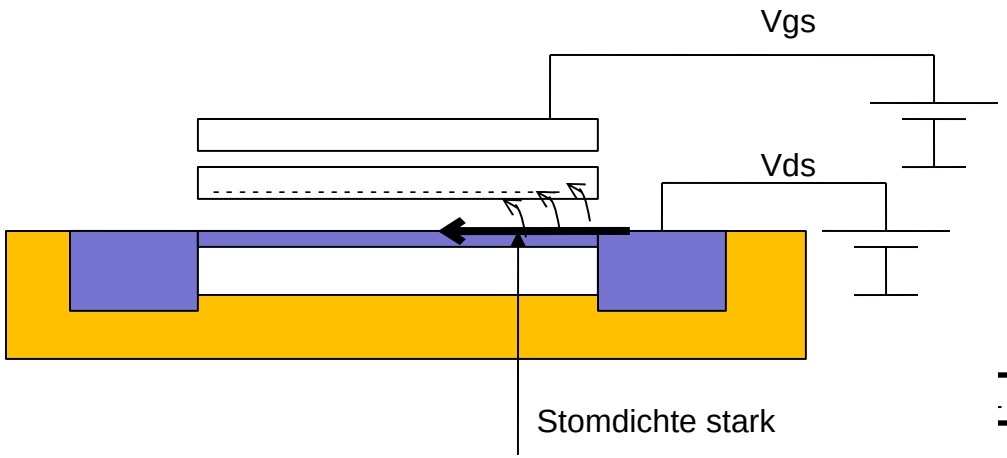


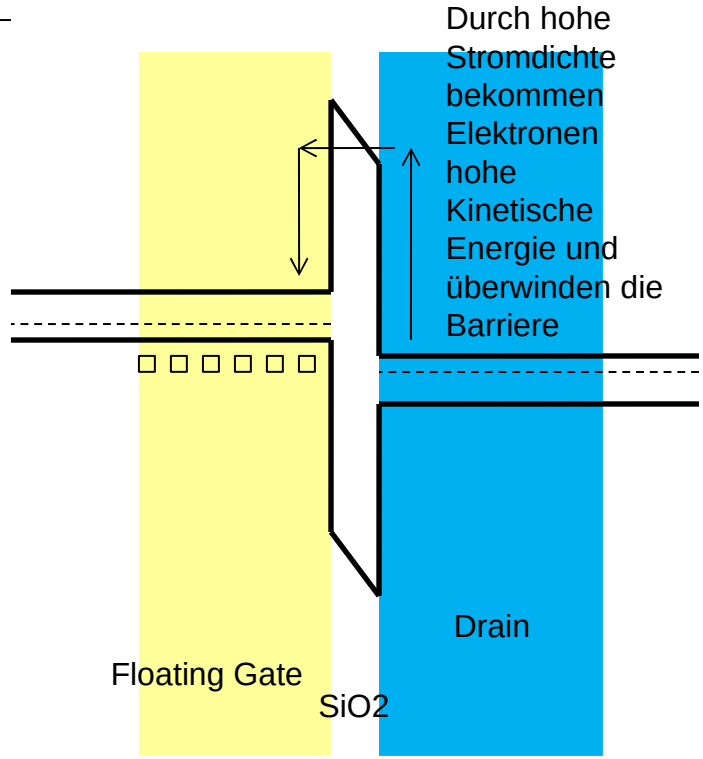
Löschen - Tunneleffekt

- EEPROM erase, Fowler-Nordheim tunnelling



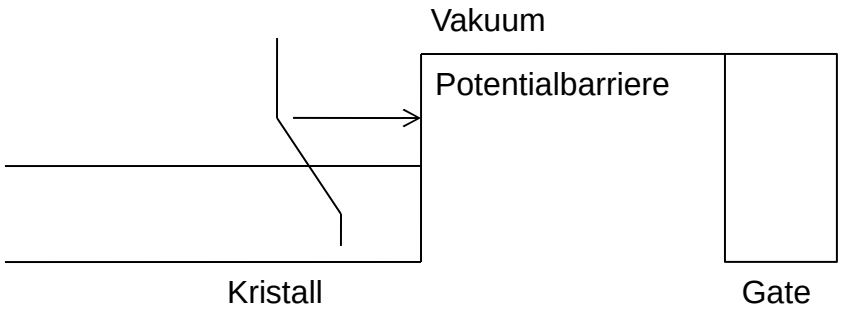
Programmierung – hot carrier injection





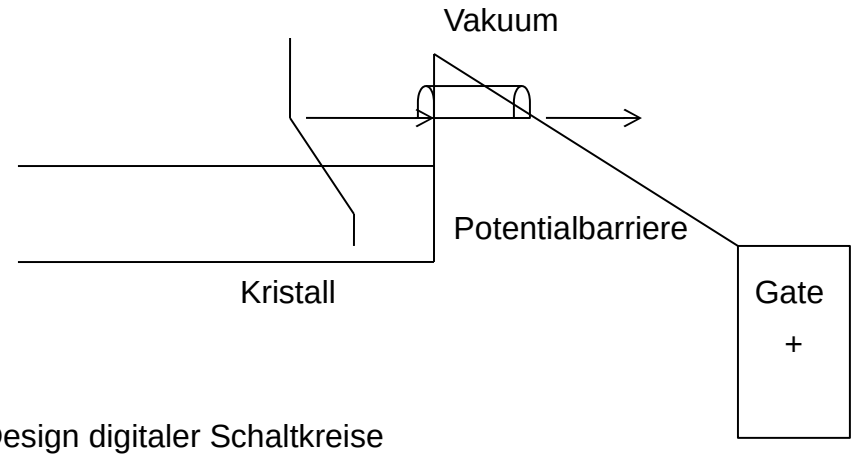
Theorie

- 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).



Theorie

- 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).

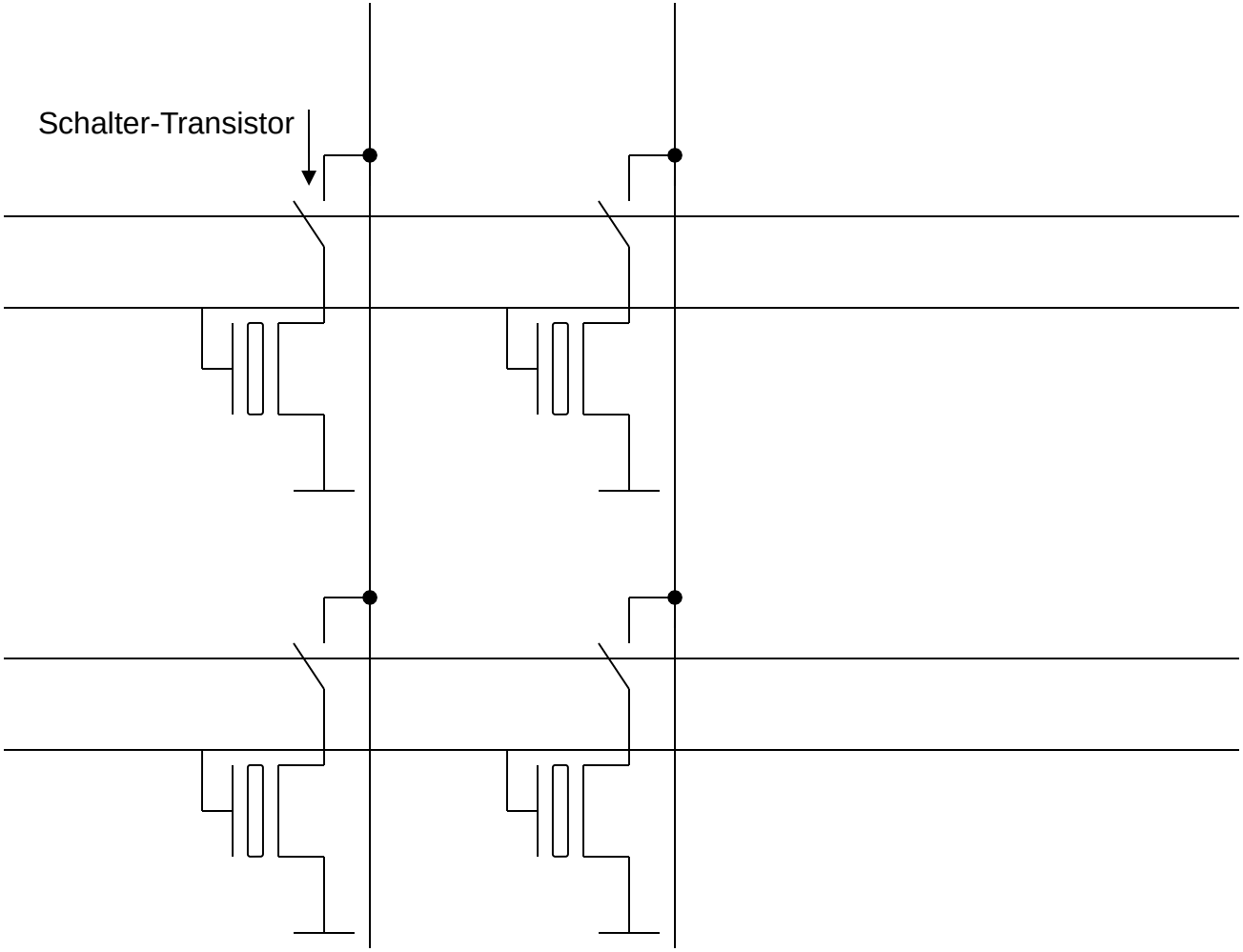


EEPROM – verschiedene Möglichkeiten

Variante 1
Erase and program based on tunnel effect

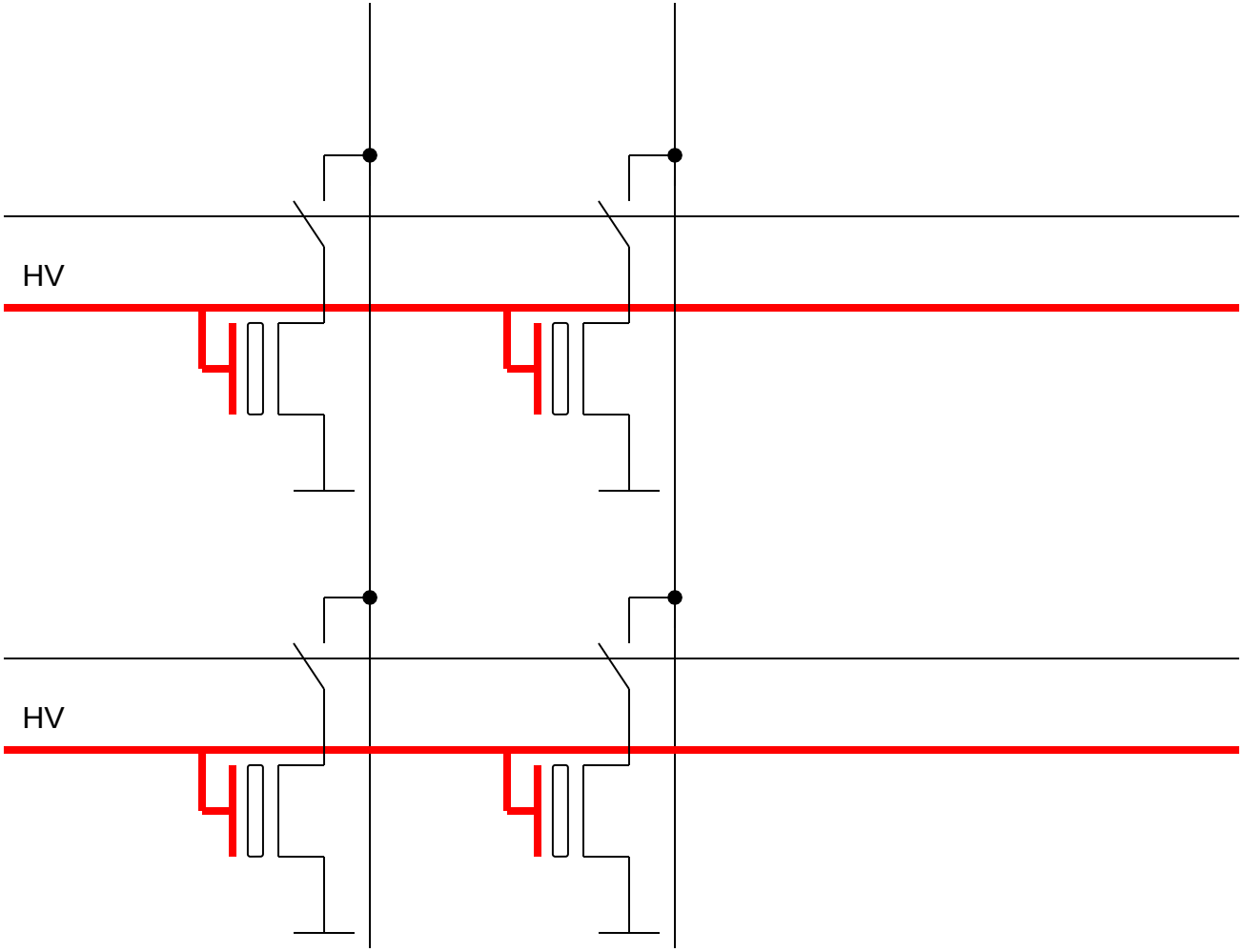
Erase

- EEPROM (erase and program based on tunnel effect)



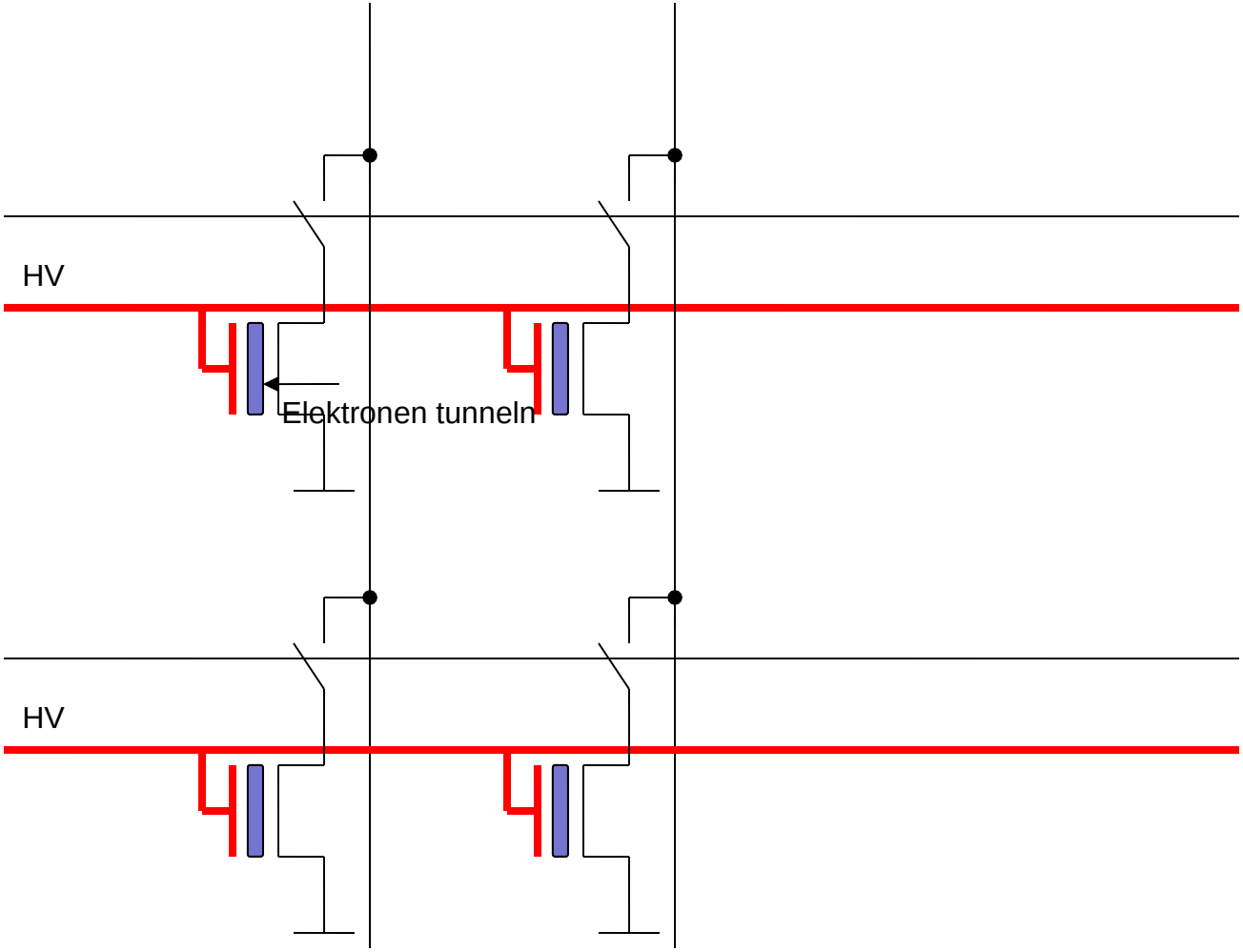
Erase

- EEPROM (erase and program based on tunnel effect)



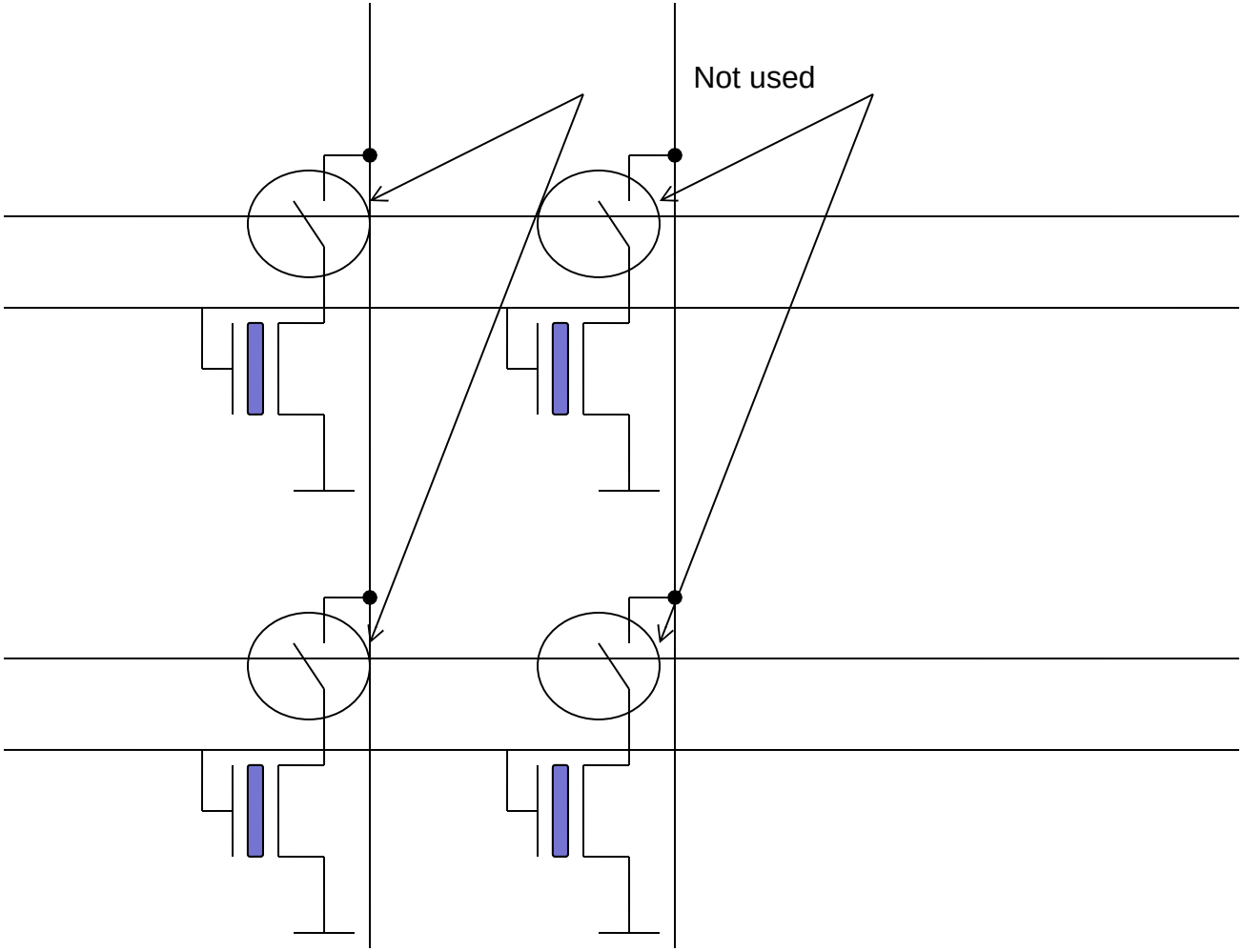
Erase

- EEPROM (erase and program based on tunnel effect)



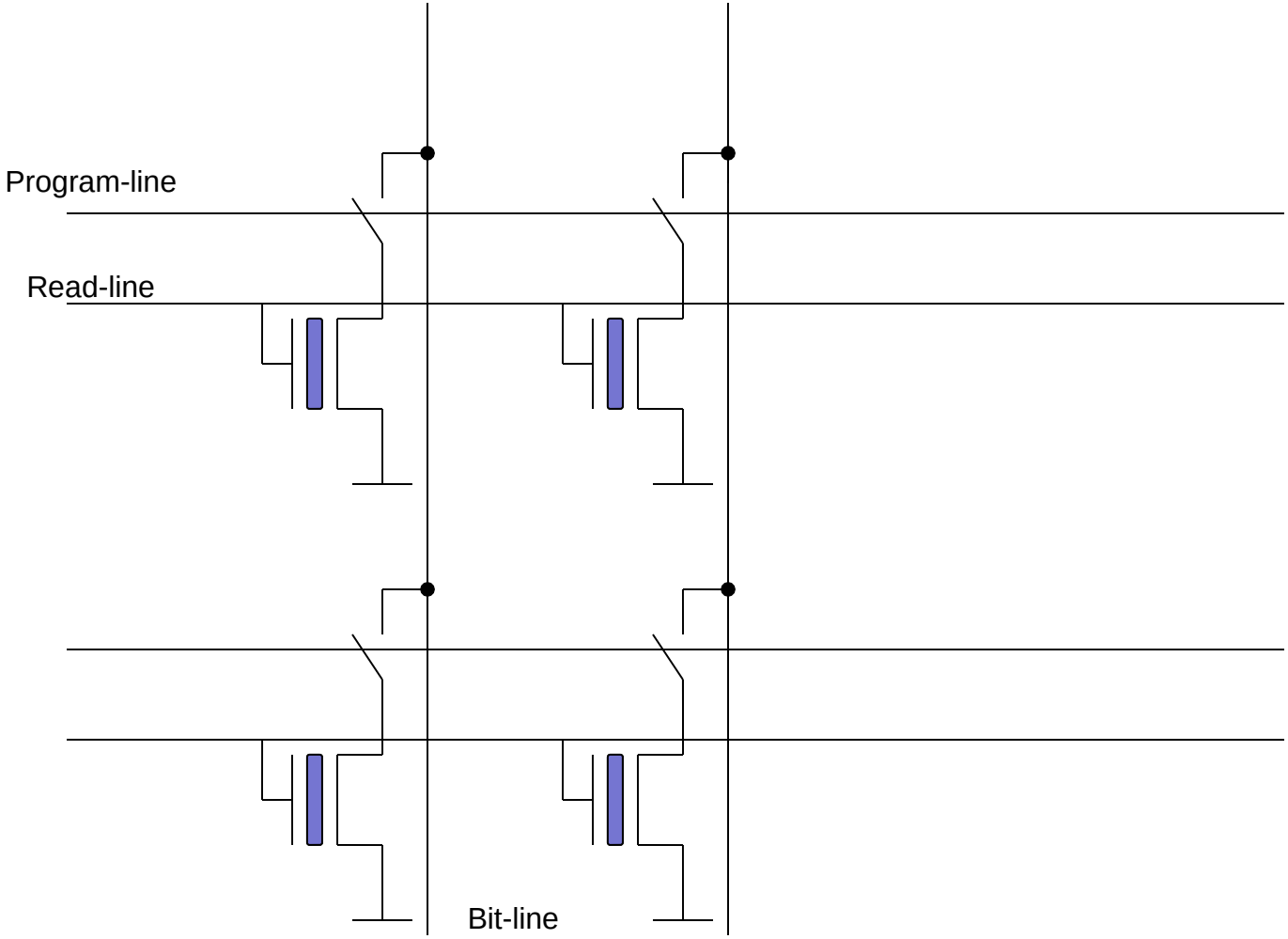
Erase

- EEPROM (erase and program based on tunnel effect)



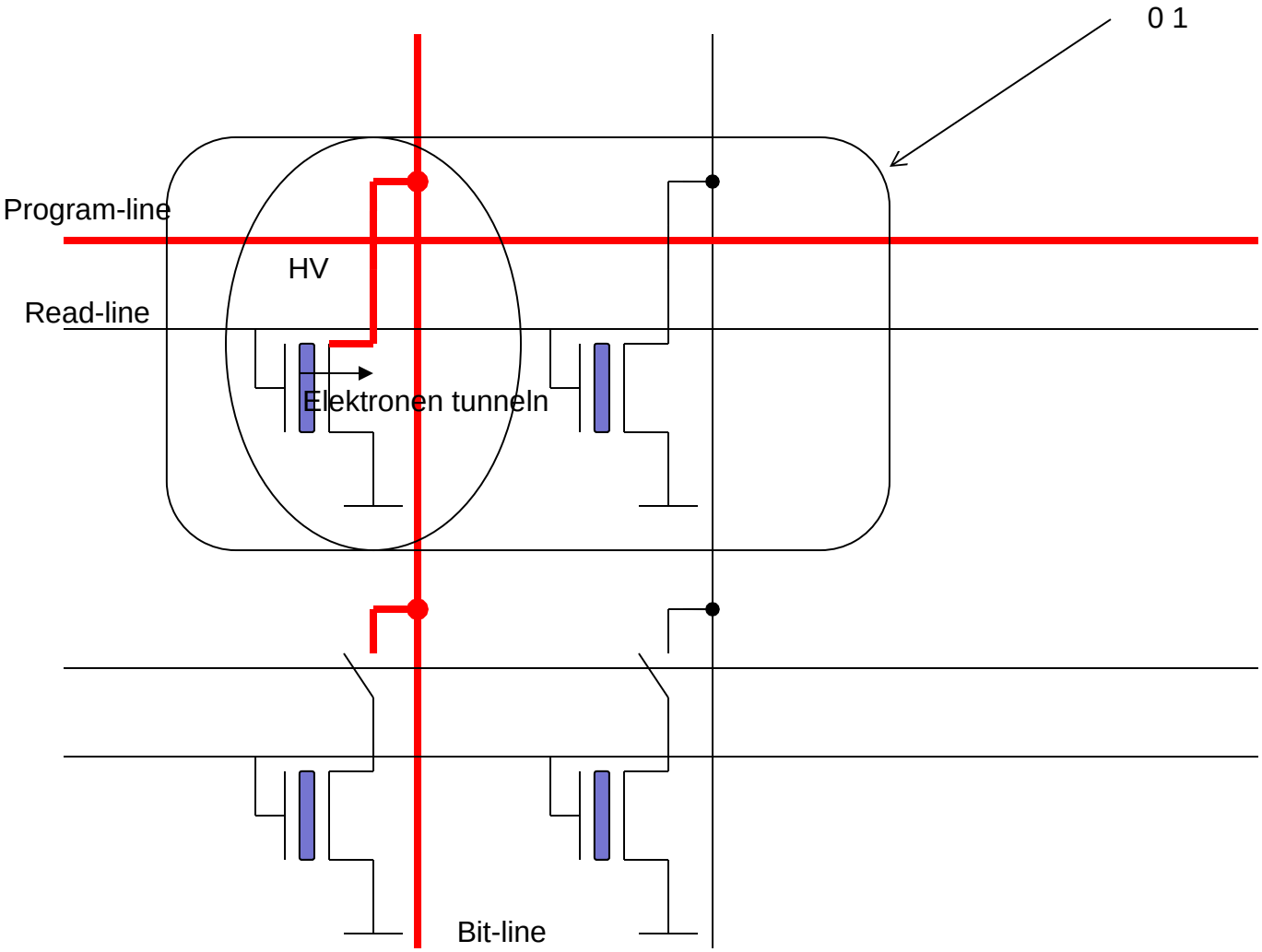
Program

- EEPROM (erase and program based on tunnel effect)



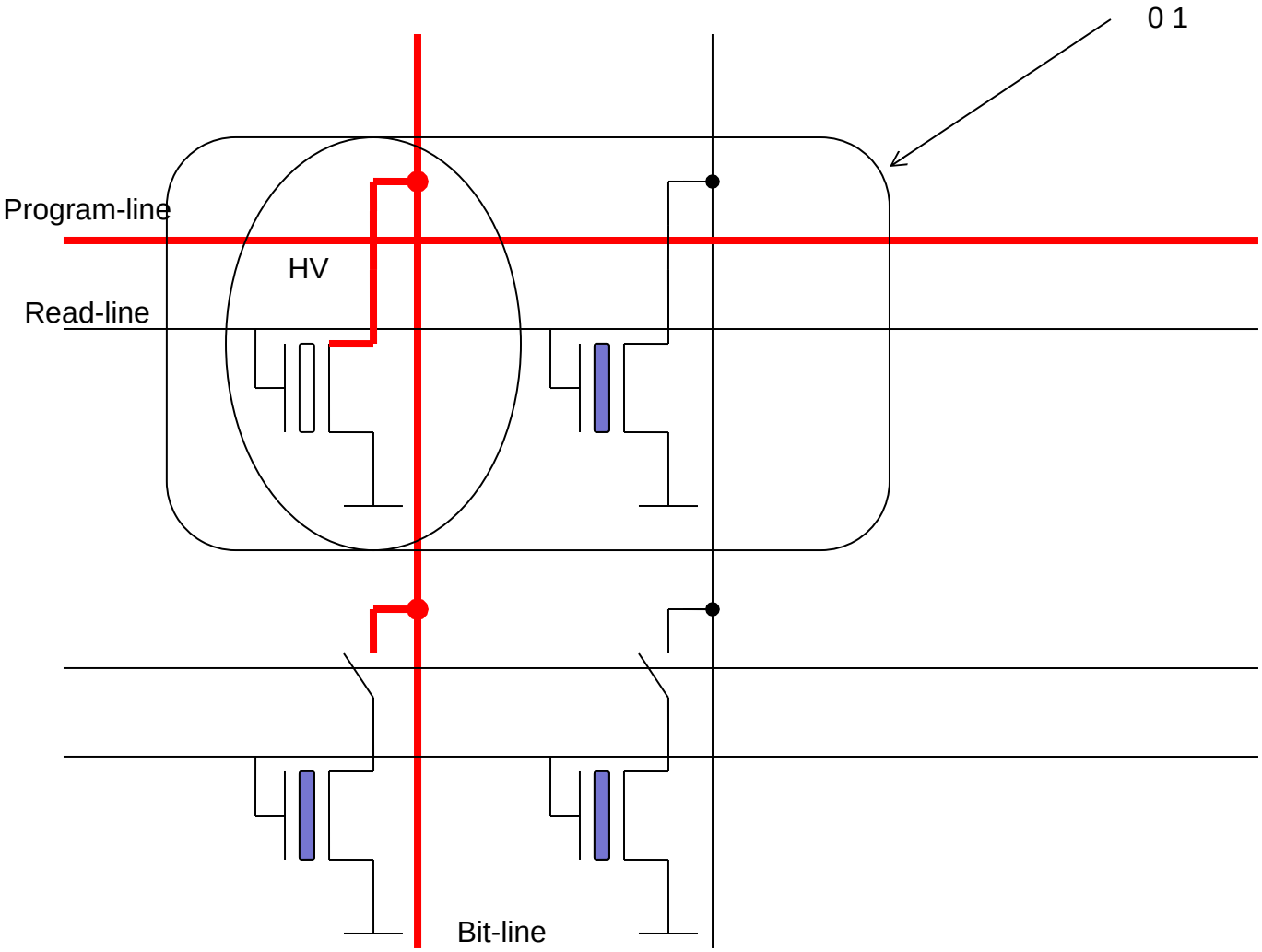
Program

- EEPROM (erase and program based on tunnel effect)



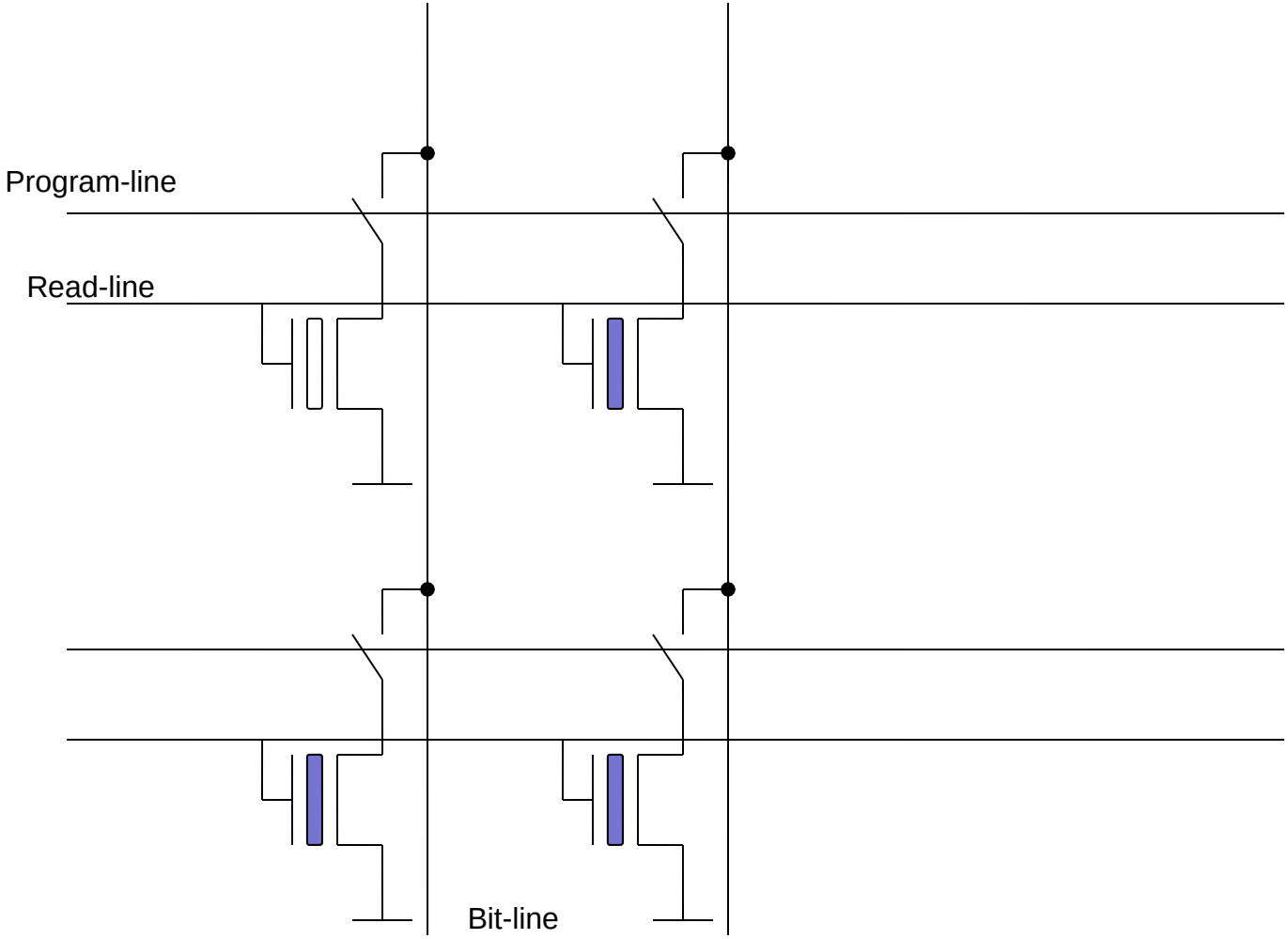
Program

- EEPROM (erase and program based on tunnel effect)



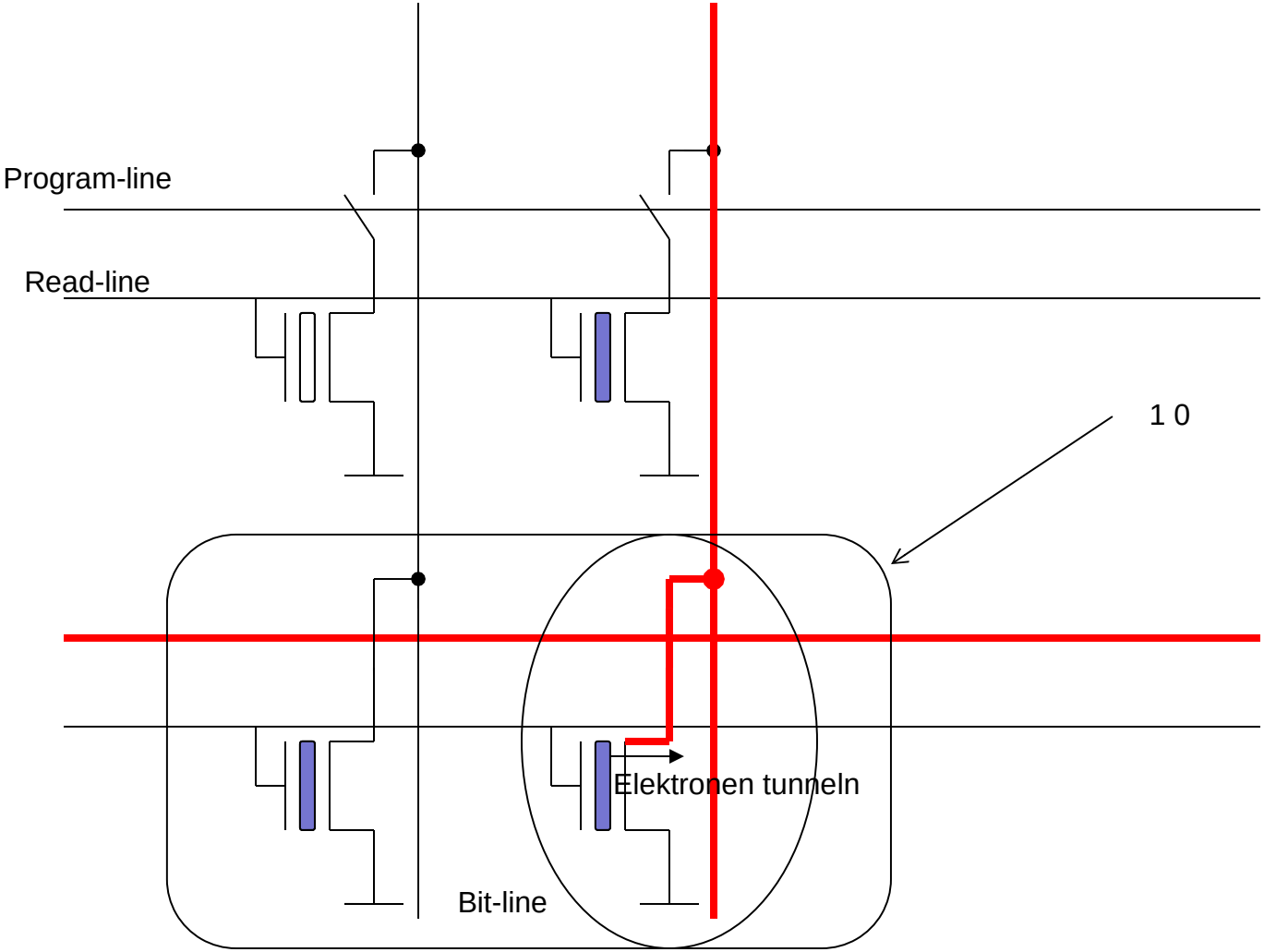
Program

- EEPROM (erase and program based on tunnel effect)



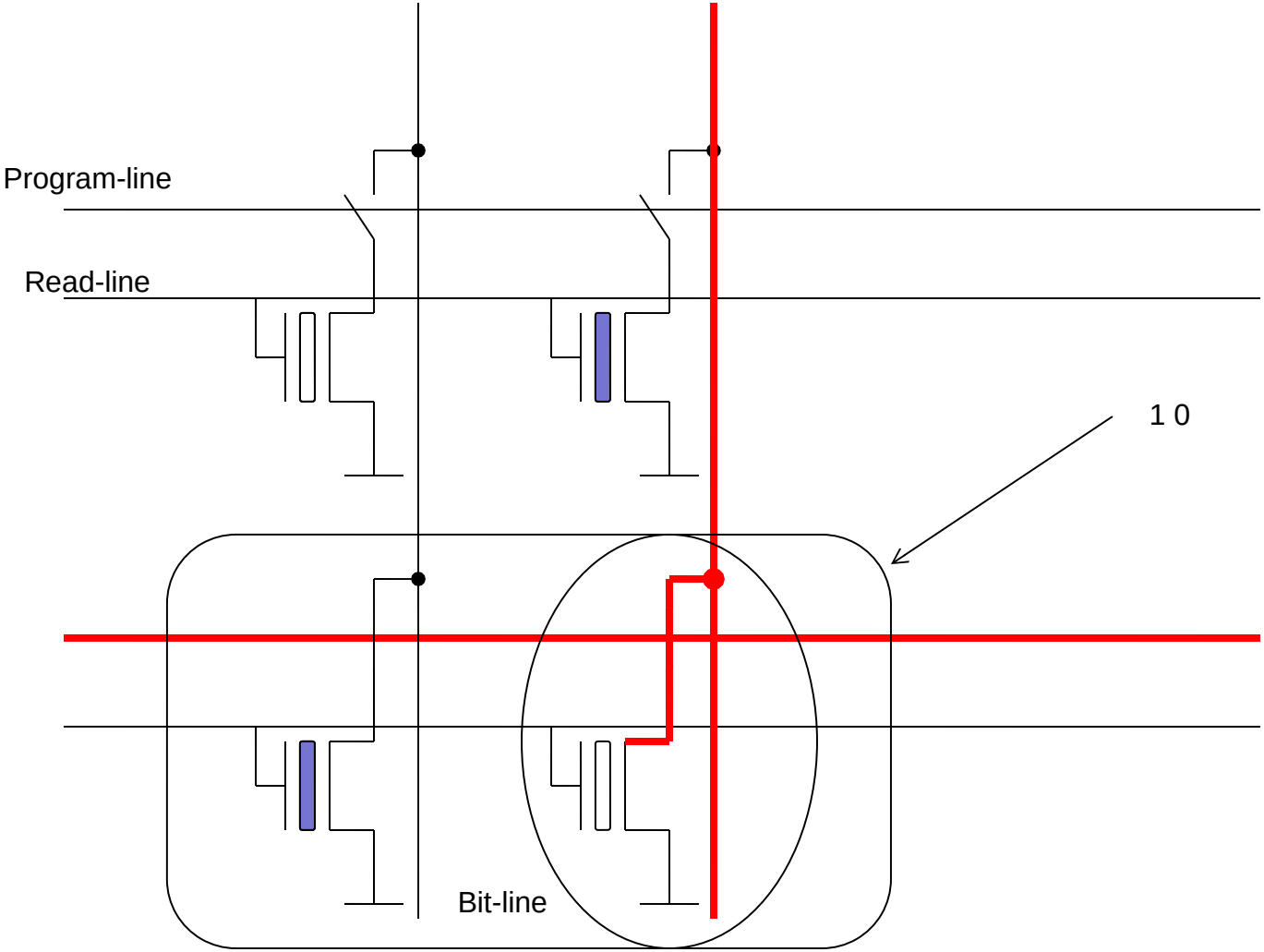
Program

- EEPROM (erase and program based on tunnel effect)



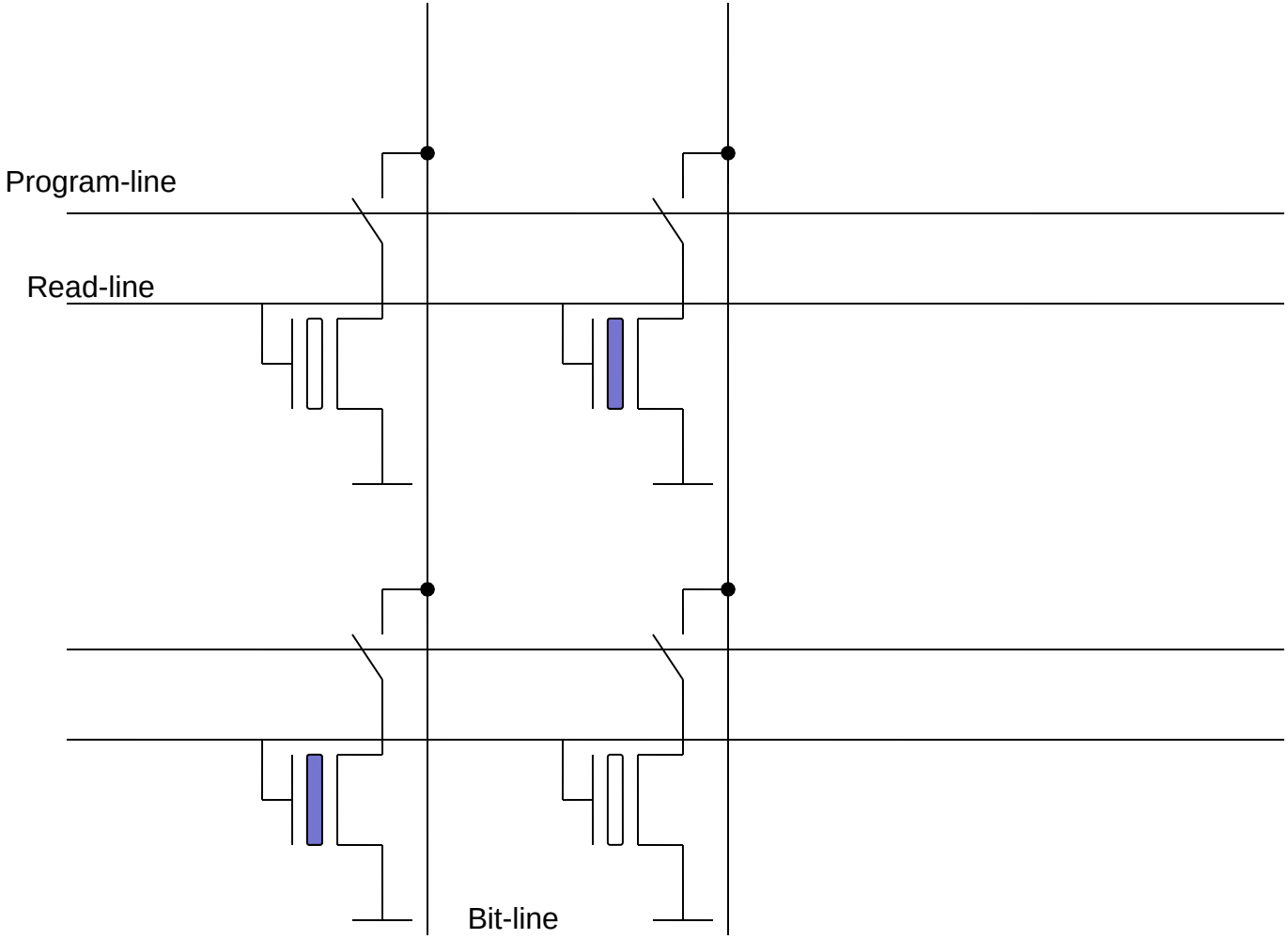
Program

- EEPROM (erase and program based on tunnel effect)



Program

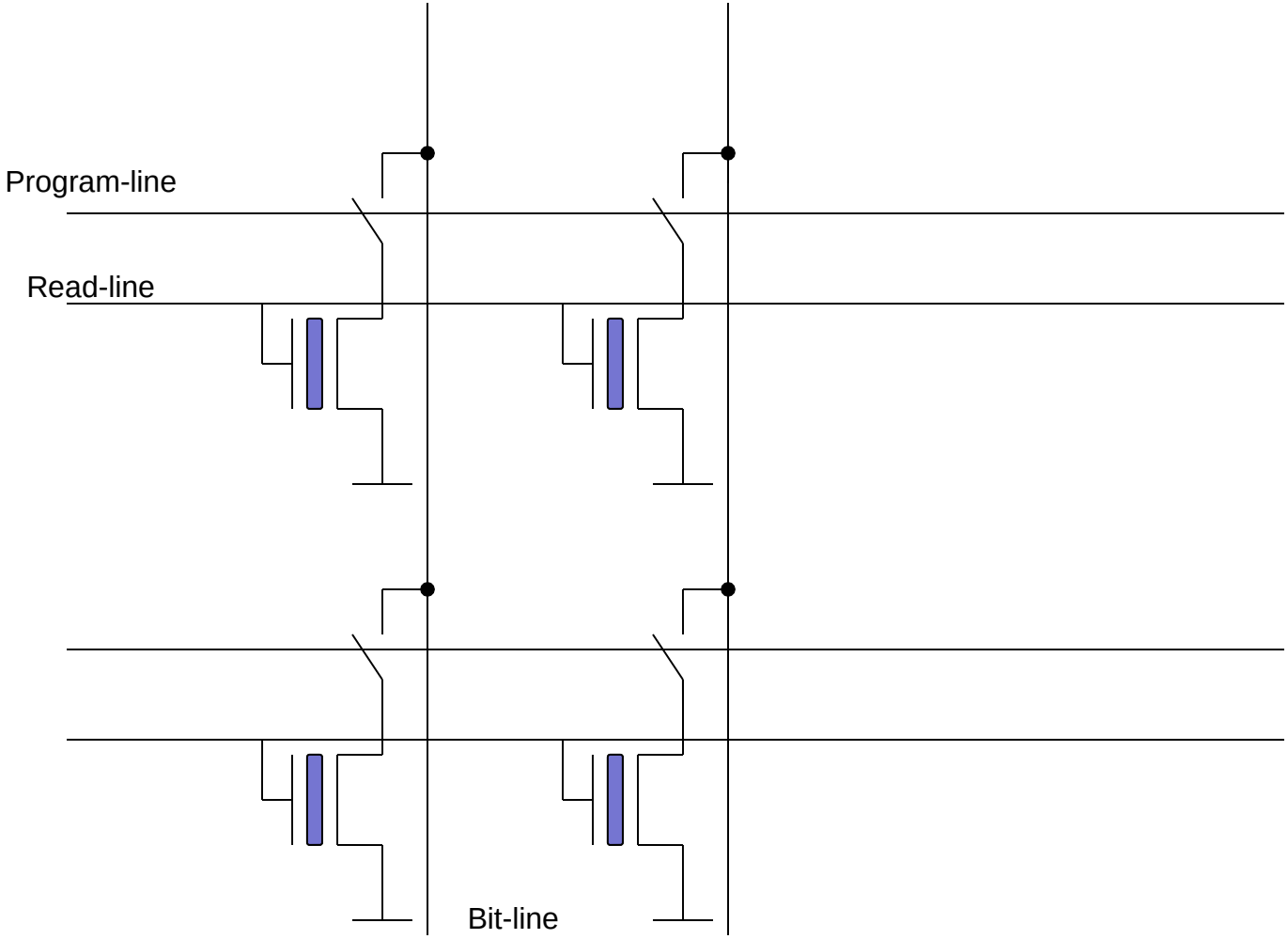
- EEPROM (erase and program based on tunnel effect)



Variante 2
Erase based on tunnel effect, program based on hot carriers injection

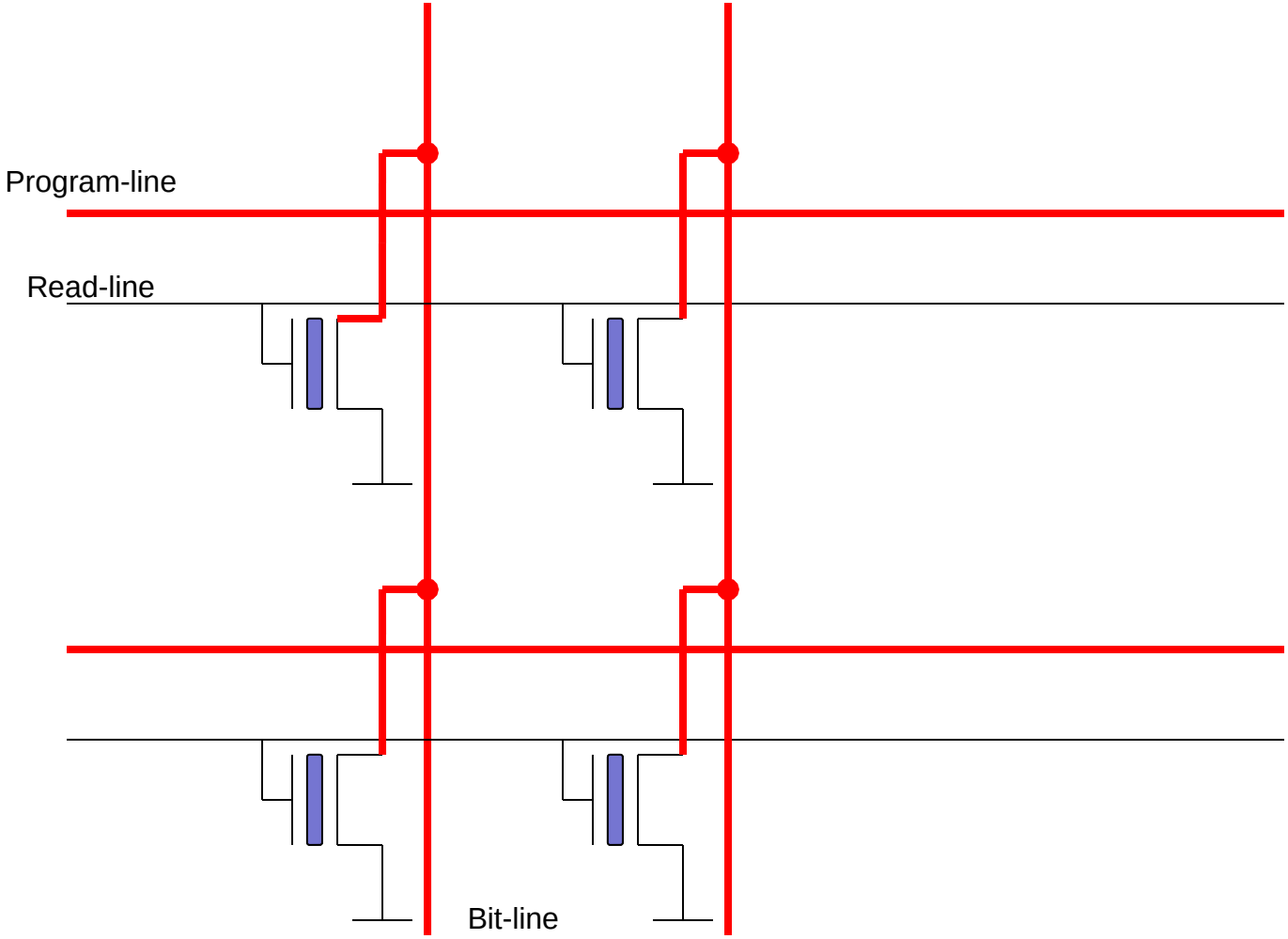
Erase

- EEPROM (erase based on tunnel effect, program based on hot carriers injection)



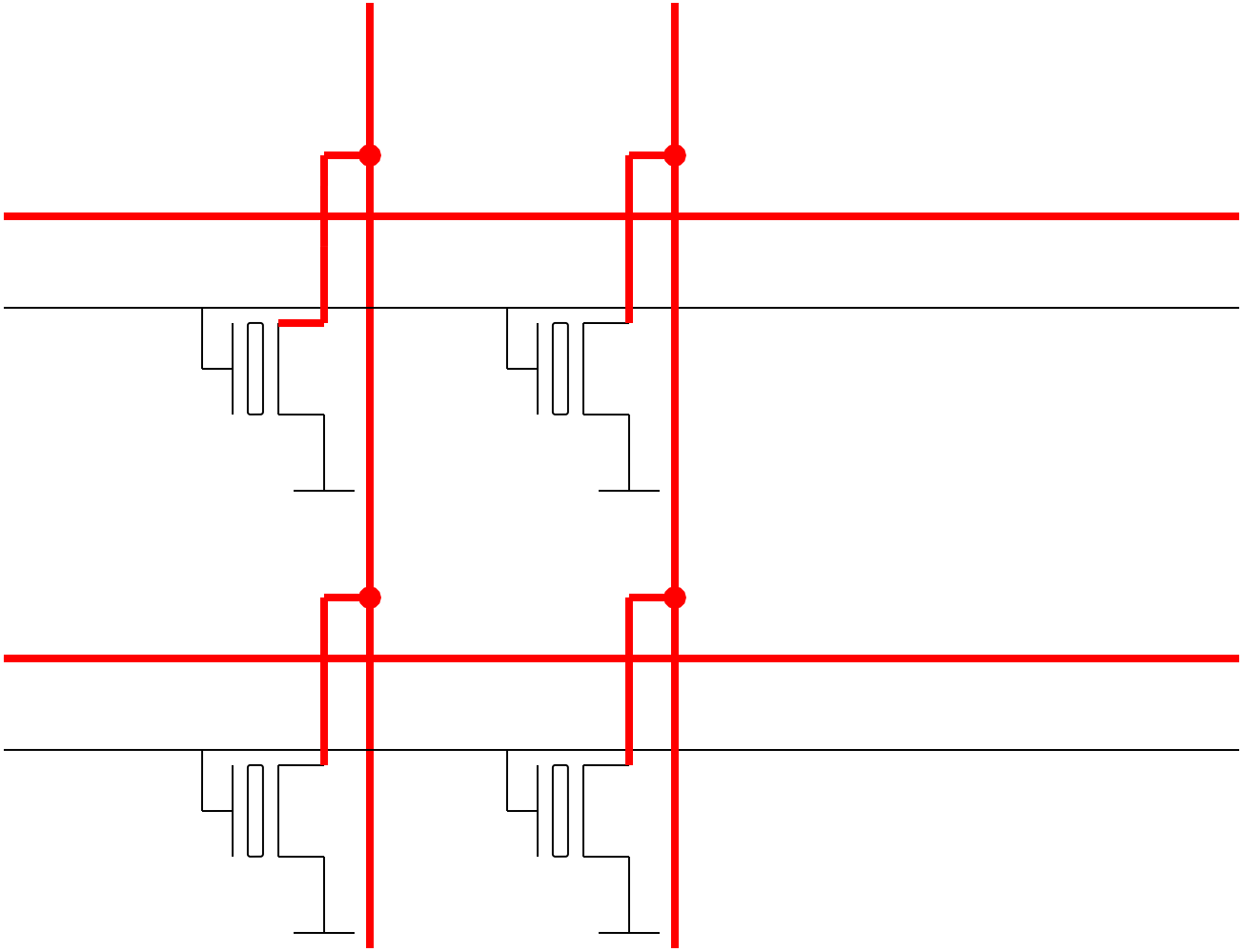
Erase

- EEPROM (erase based on tunnel effect, program based on hot carriers injection)



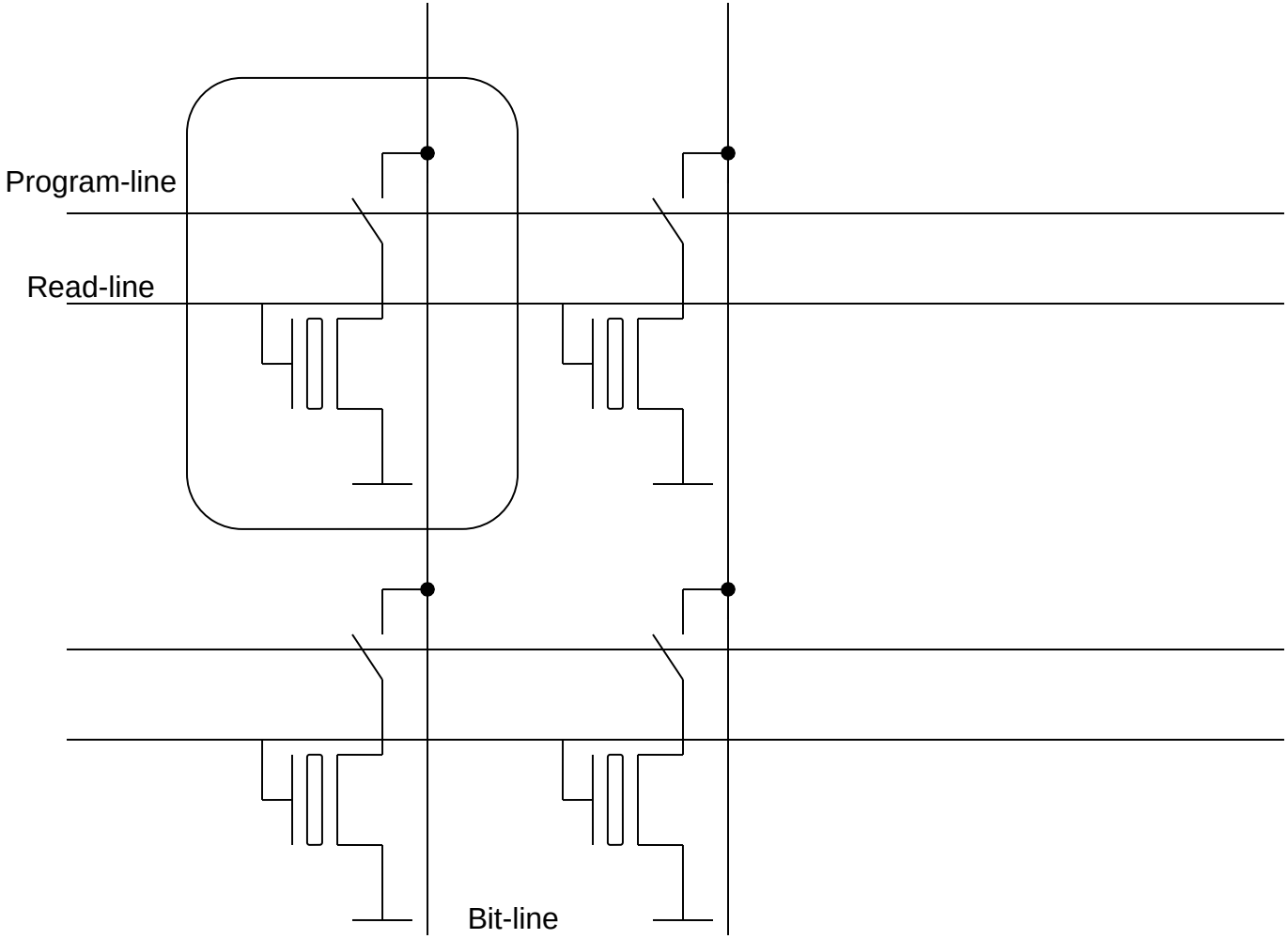
Erase

- EEPROM (erase based on tunnel effect, program based on hot carriers injection)



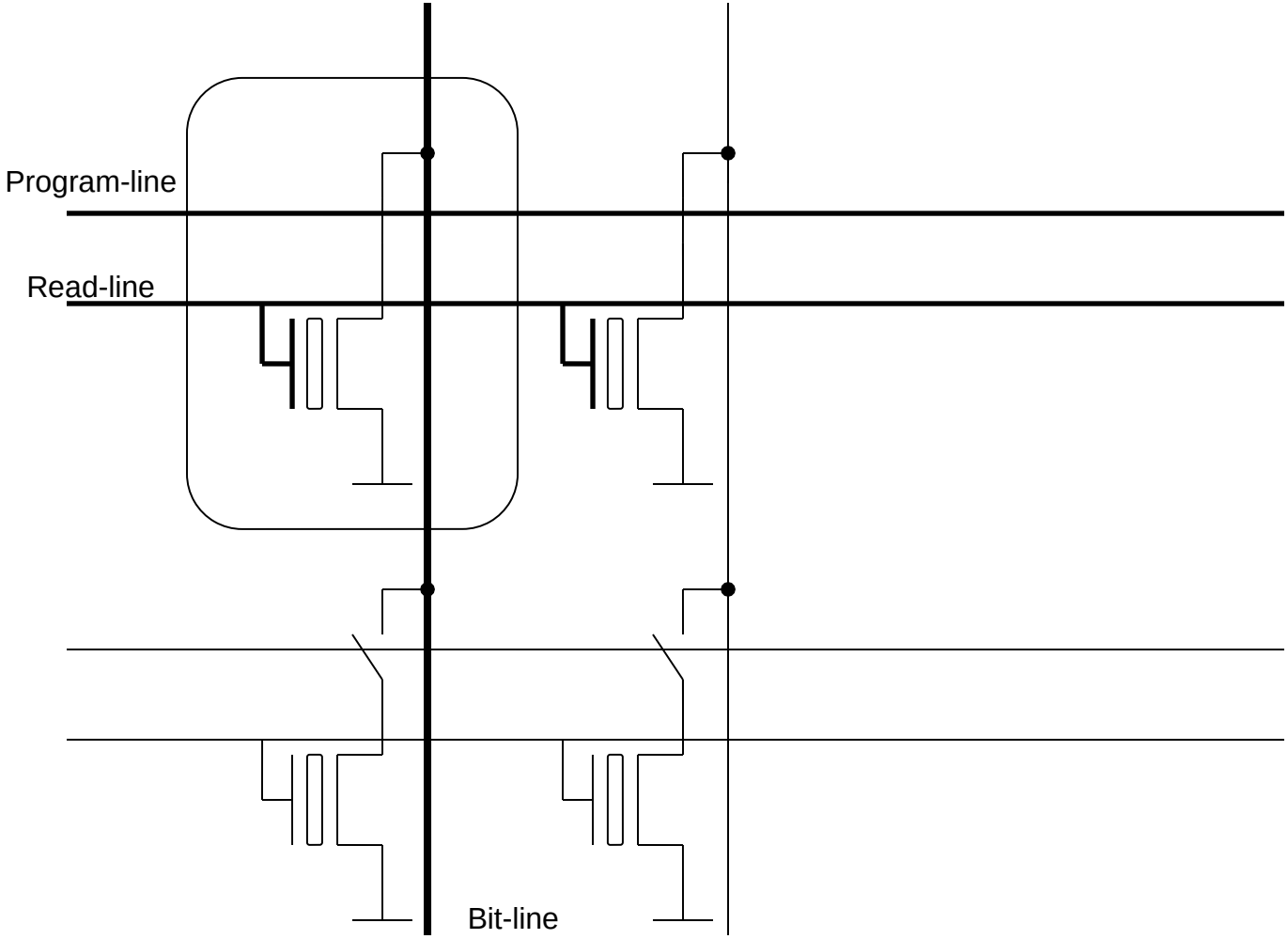
Program particular bit

- EEPROM (erase based on tunnel effect, program based on hot carriers injection)



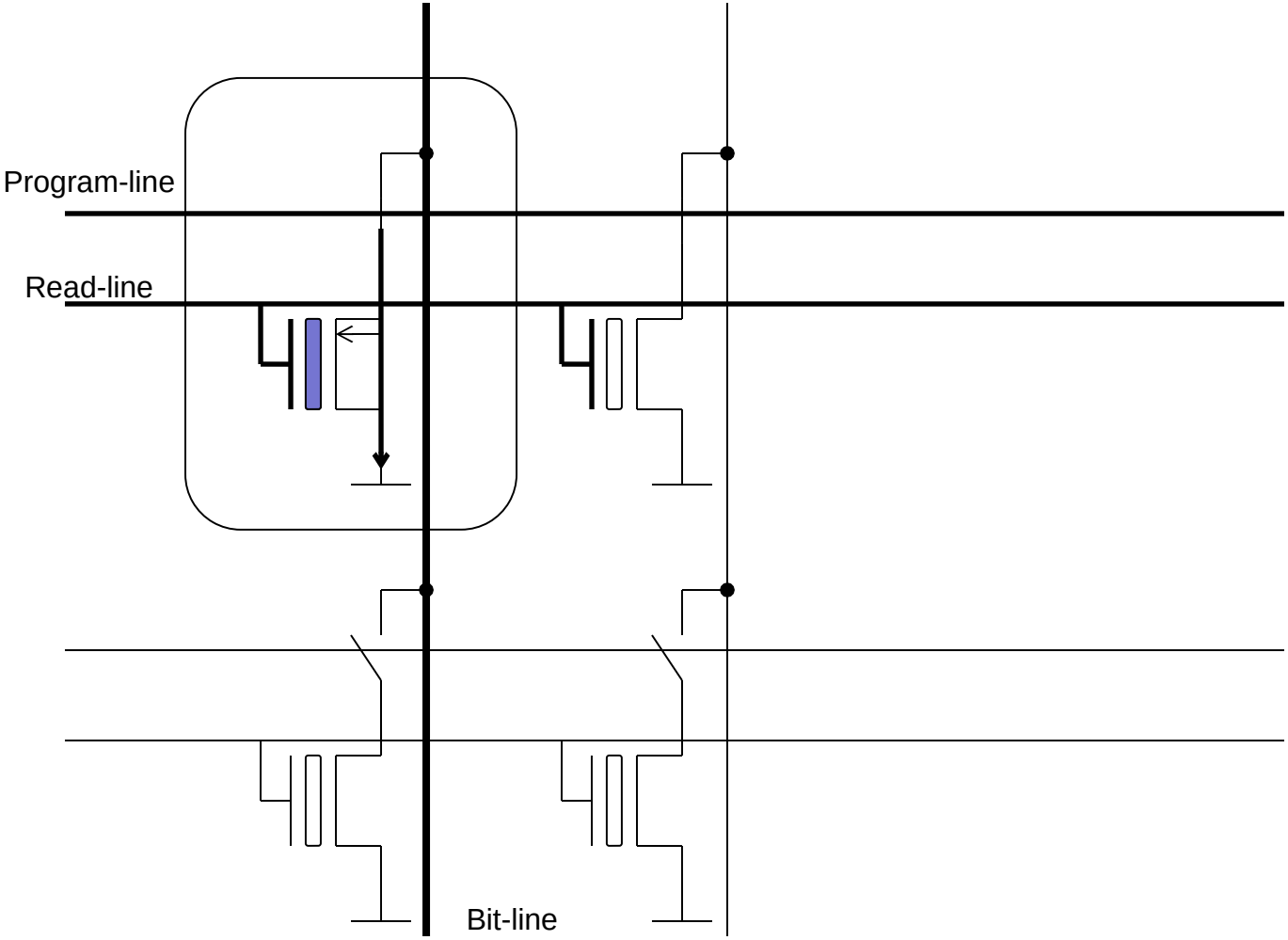
Program particular bit

- EEPROM (erase based on tunnel effect, program based on hot carriers injection)



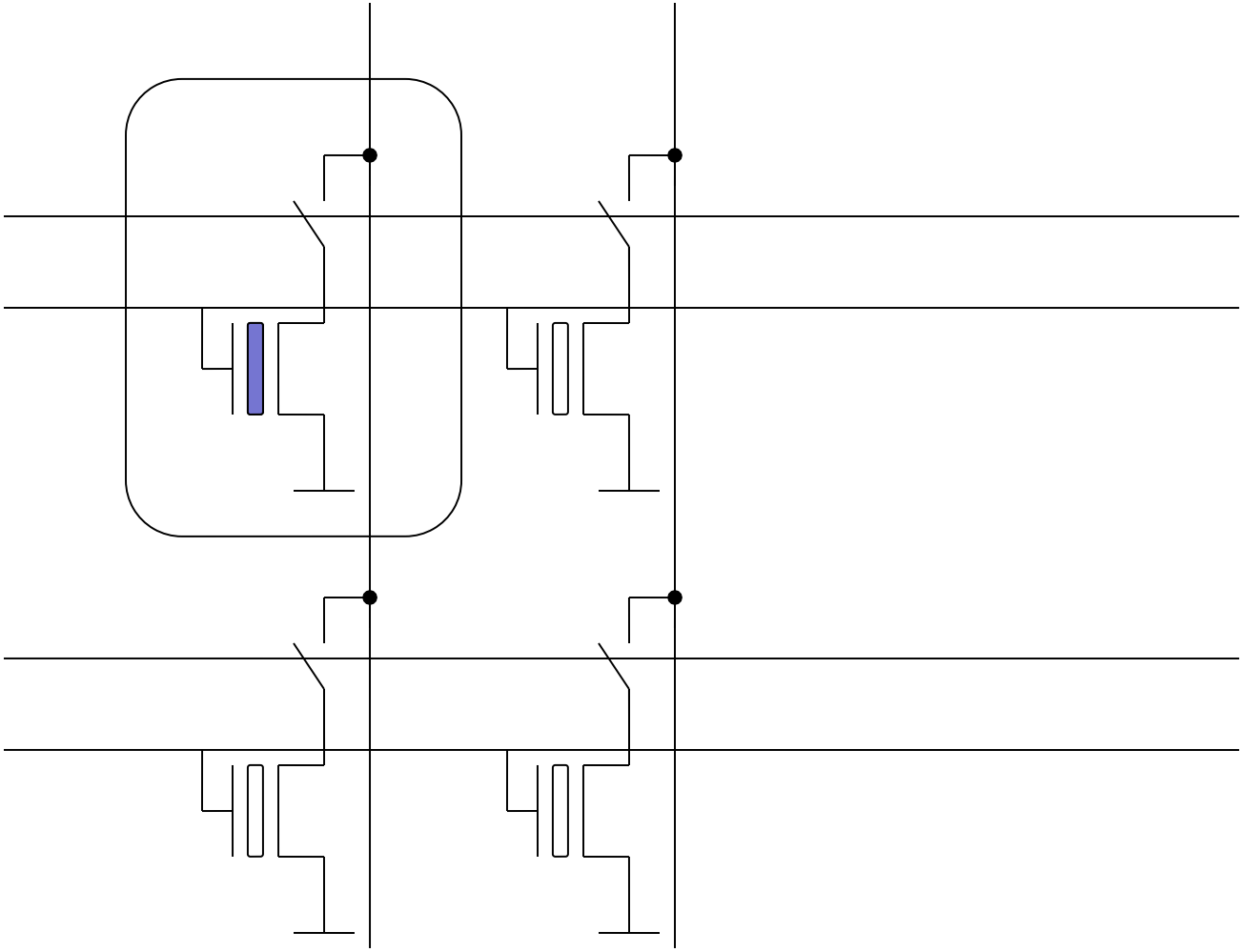
Program particular bit

- EEPROM (erase based on tunnel effect, program based on hot carriers injection)



Program particular bit

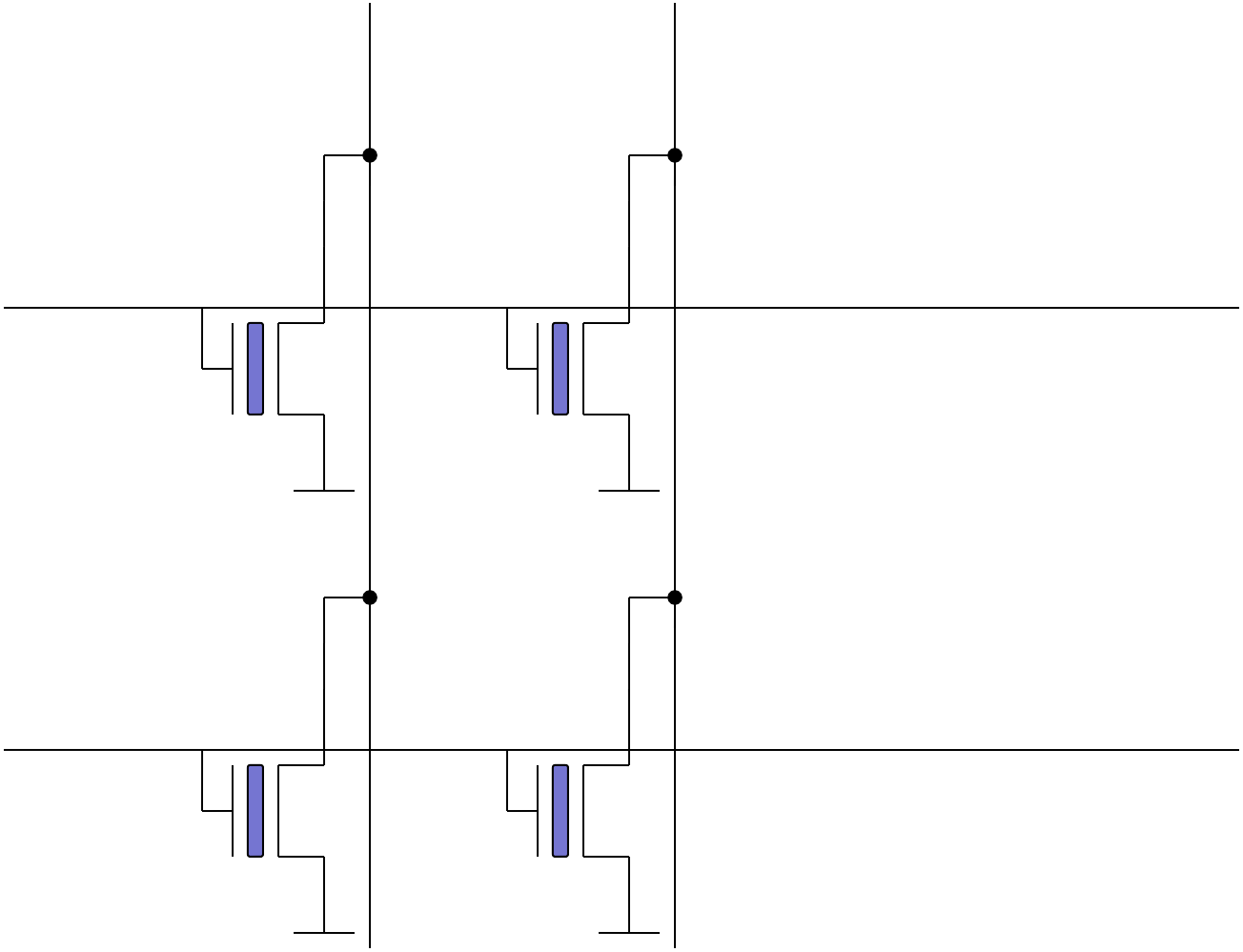
- EEPROM (erase based on tunnel effect, program based on hot carriers injection)



Variante 3
1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)

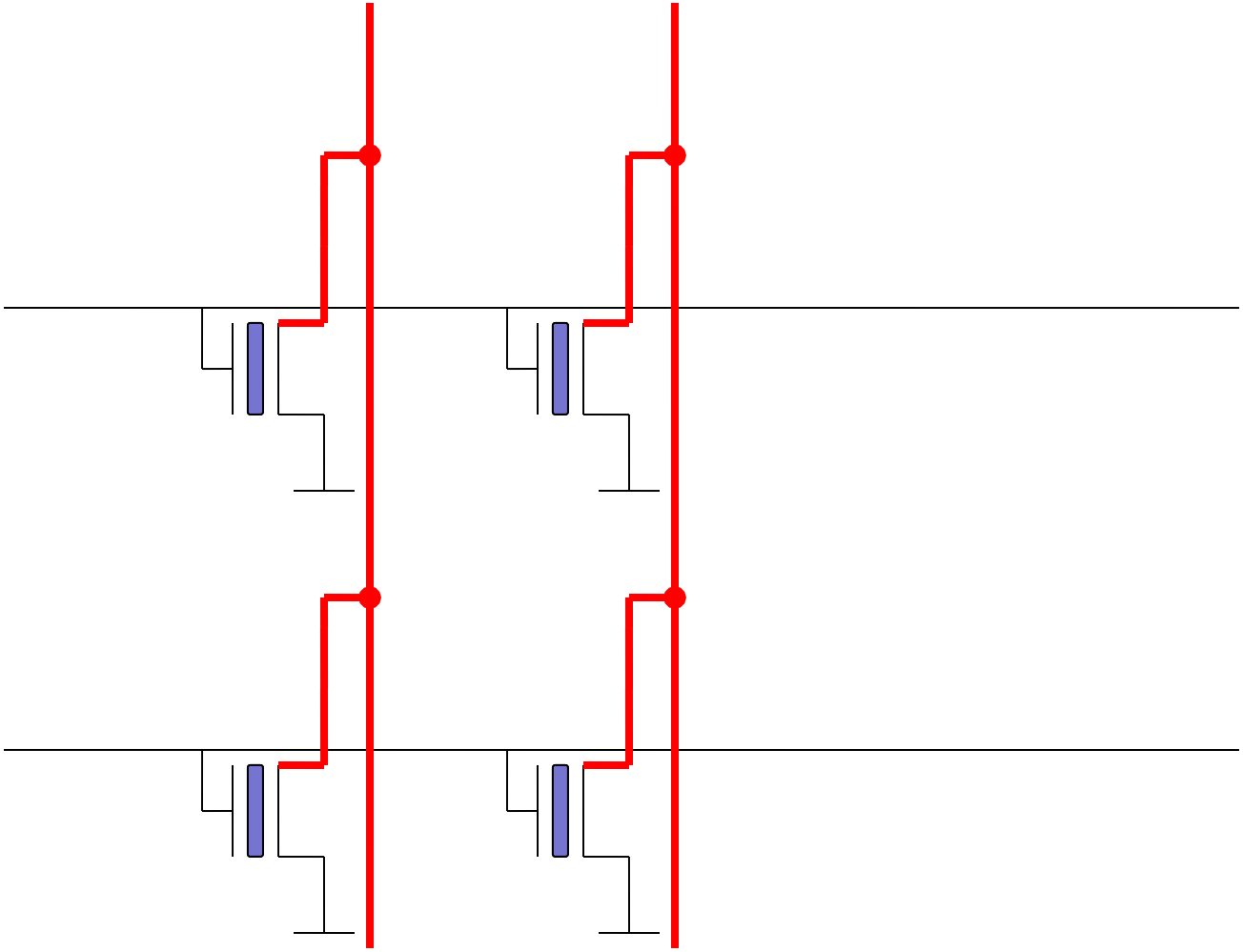
Erase

- 1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)



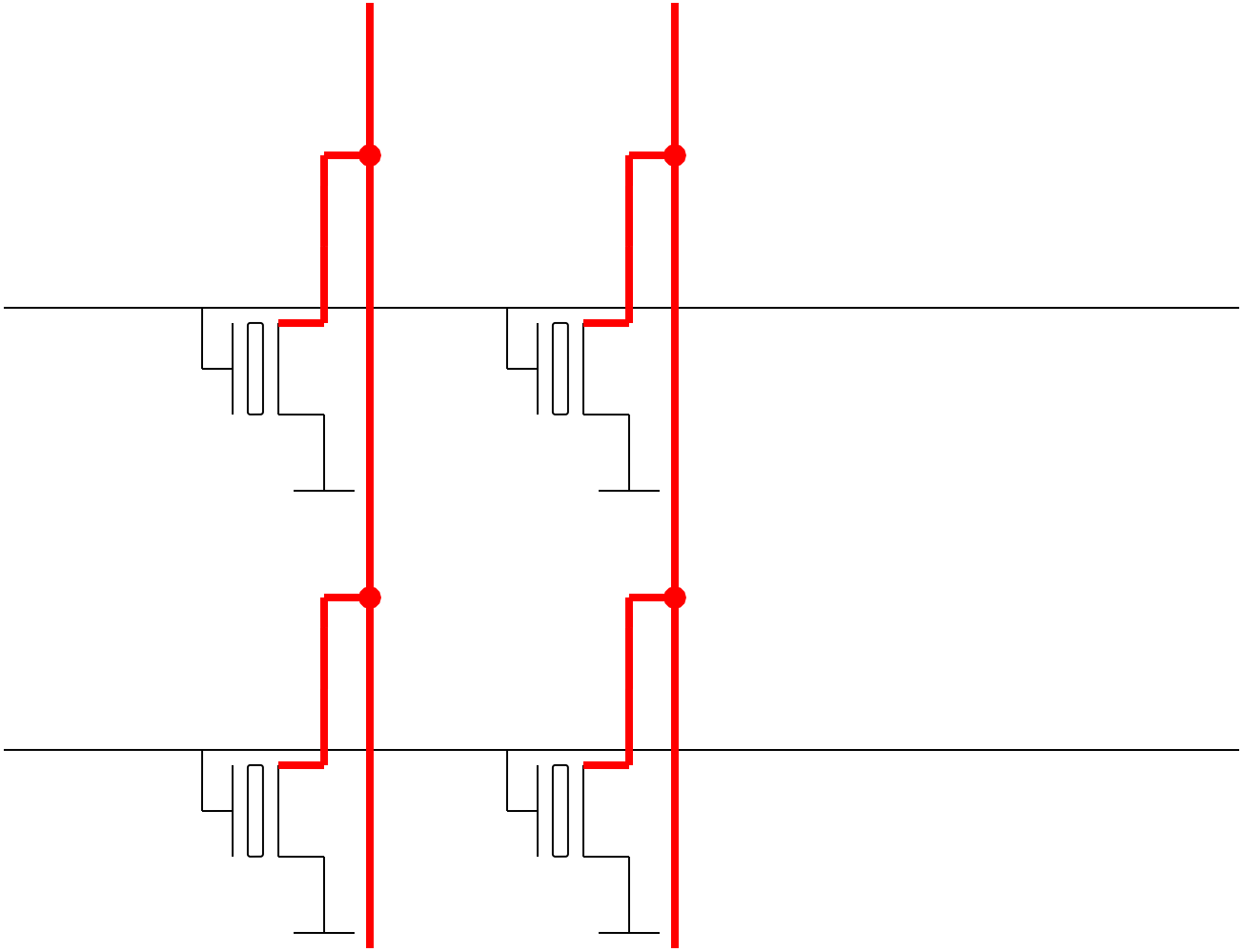
Erase

- 1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)



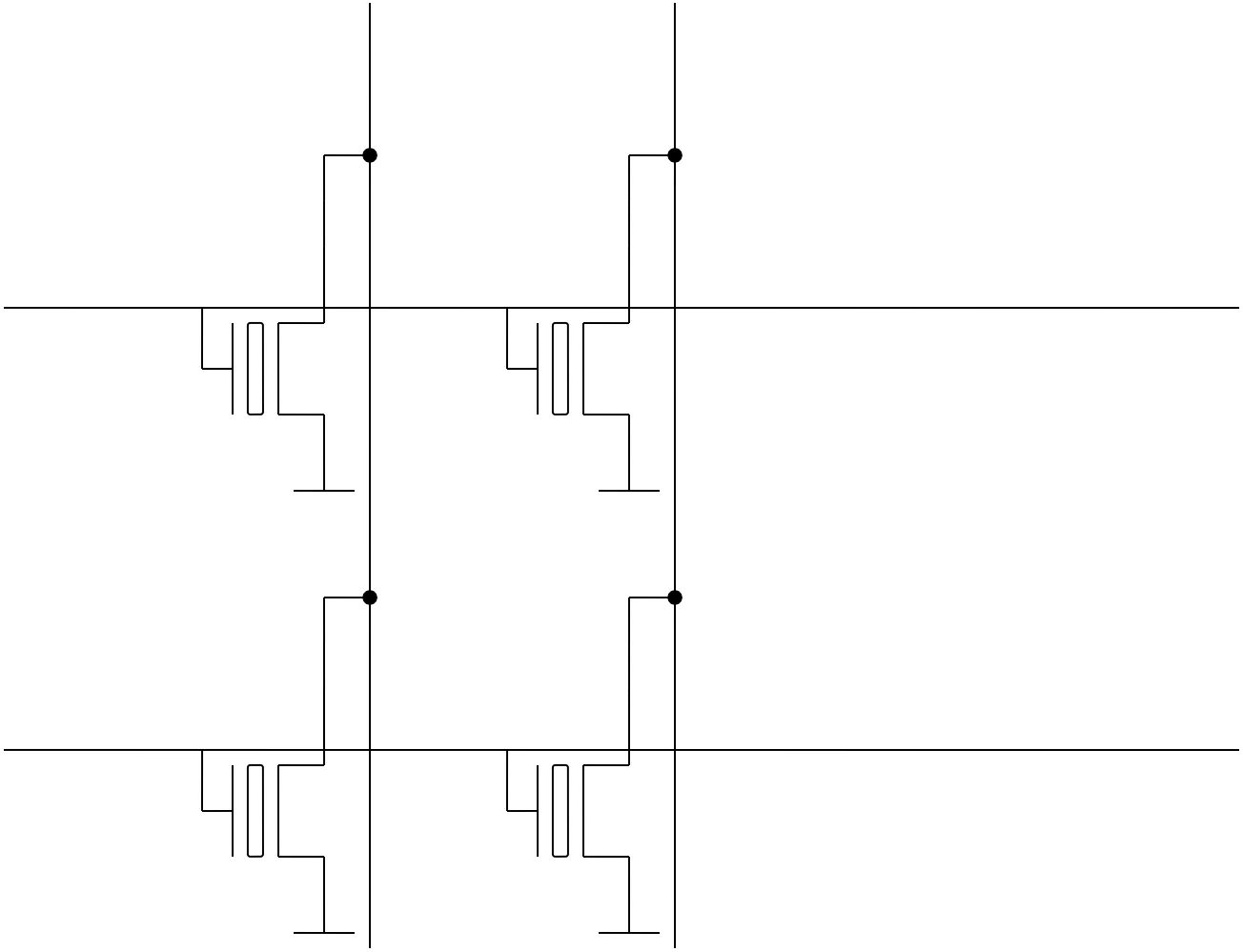
Erase

- 1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)



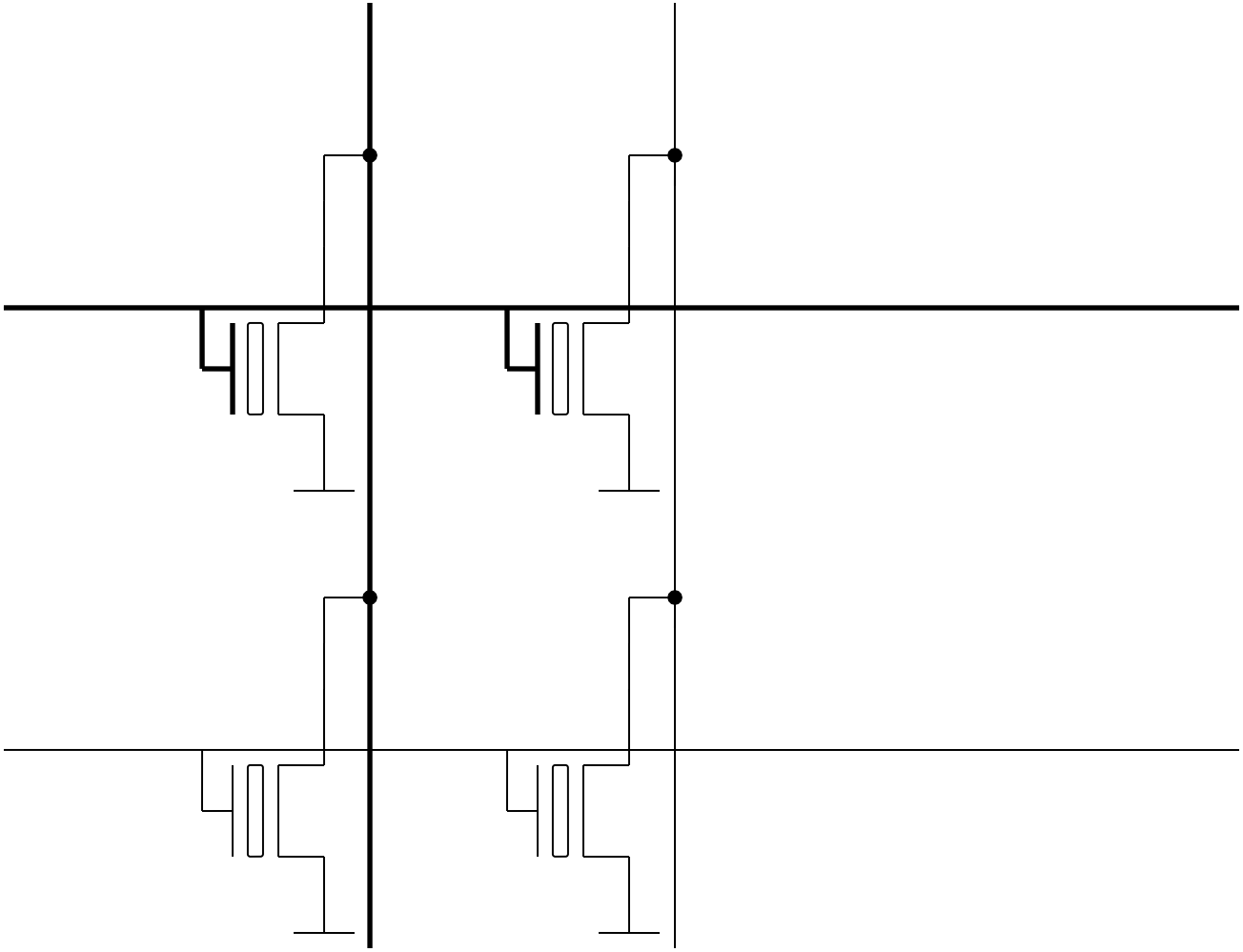
Erase

- 1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)



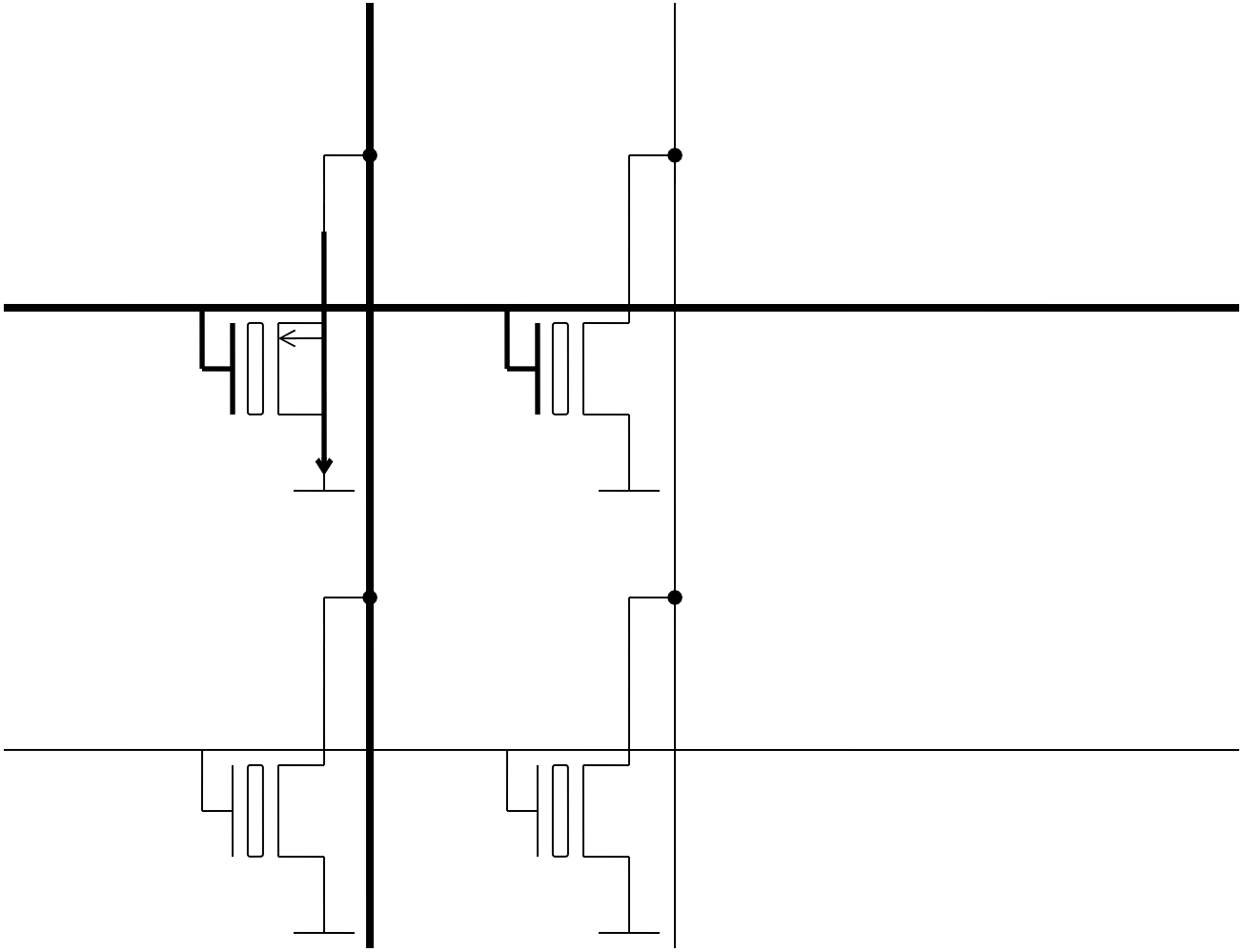
Program

- 1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)



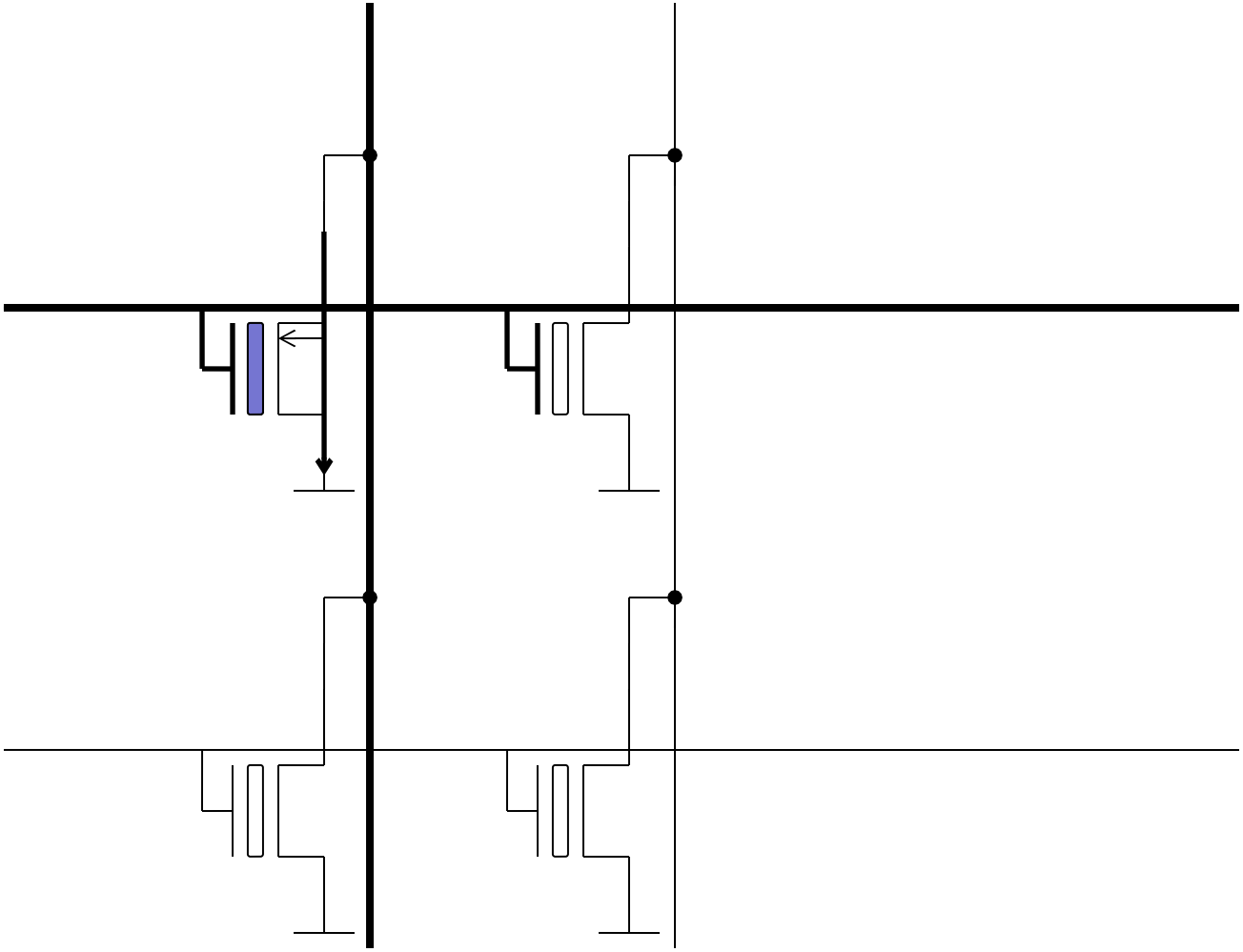
Program

- 1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)



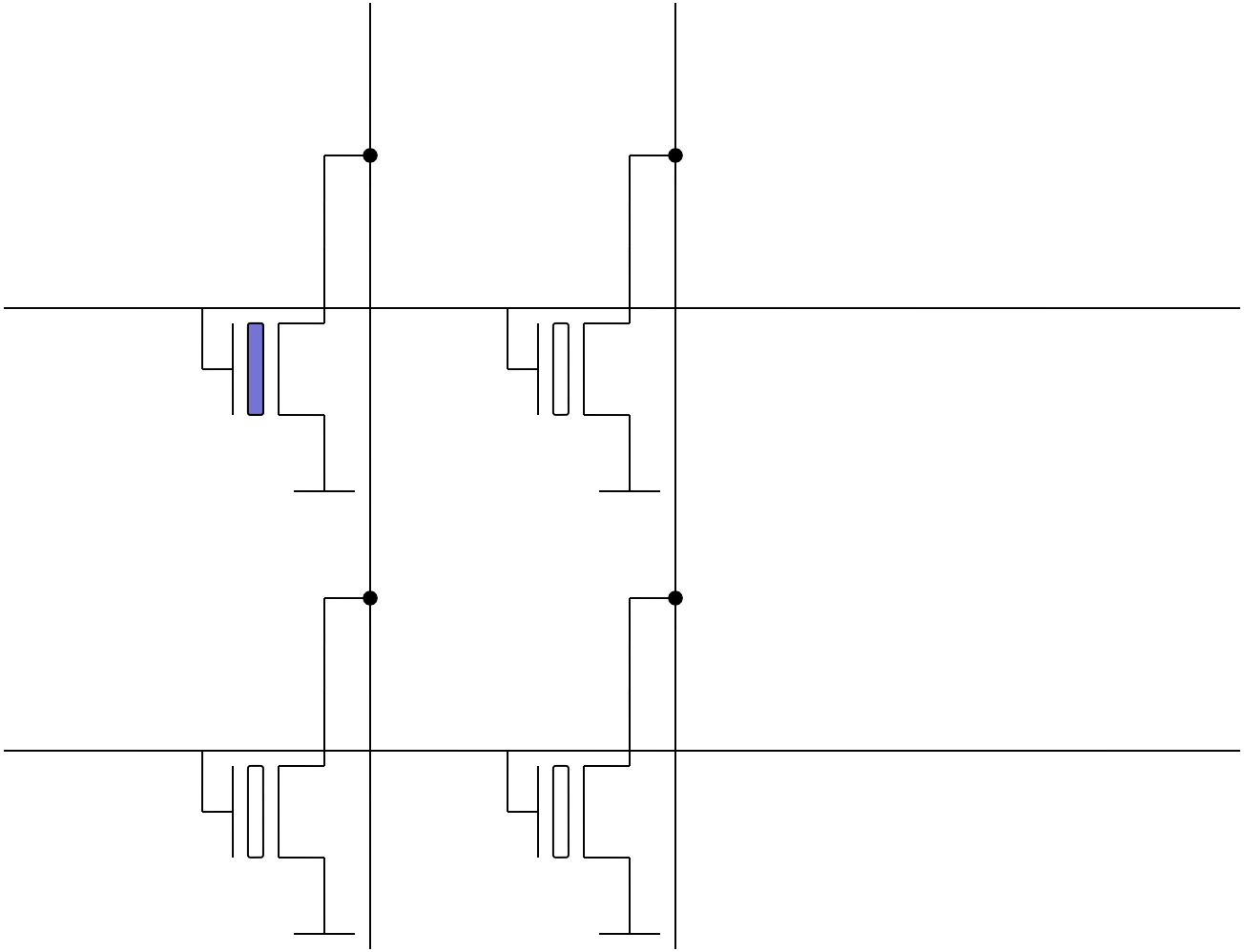
Program

- 1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)



Program

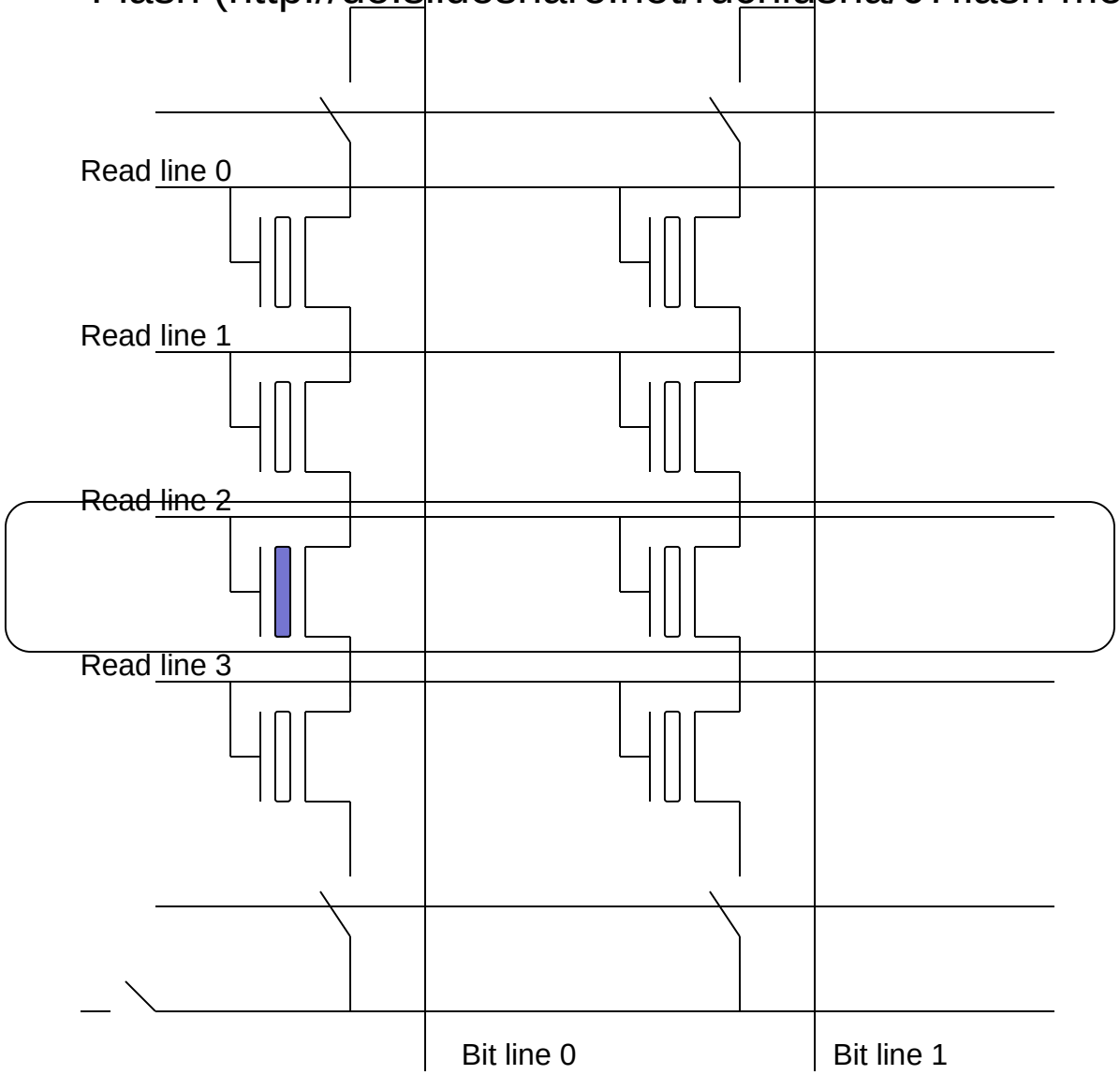
- 1 Transistor EEPROM (erase based on tunnel effect, program based on hot carriers injection)

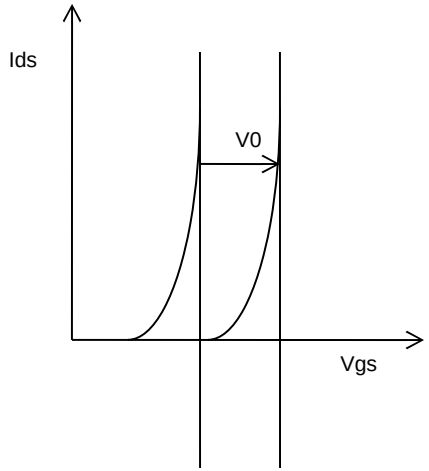


FLASH Speicher

Read

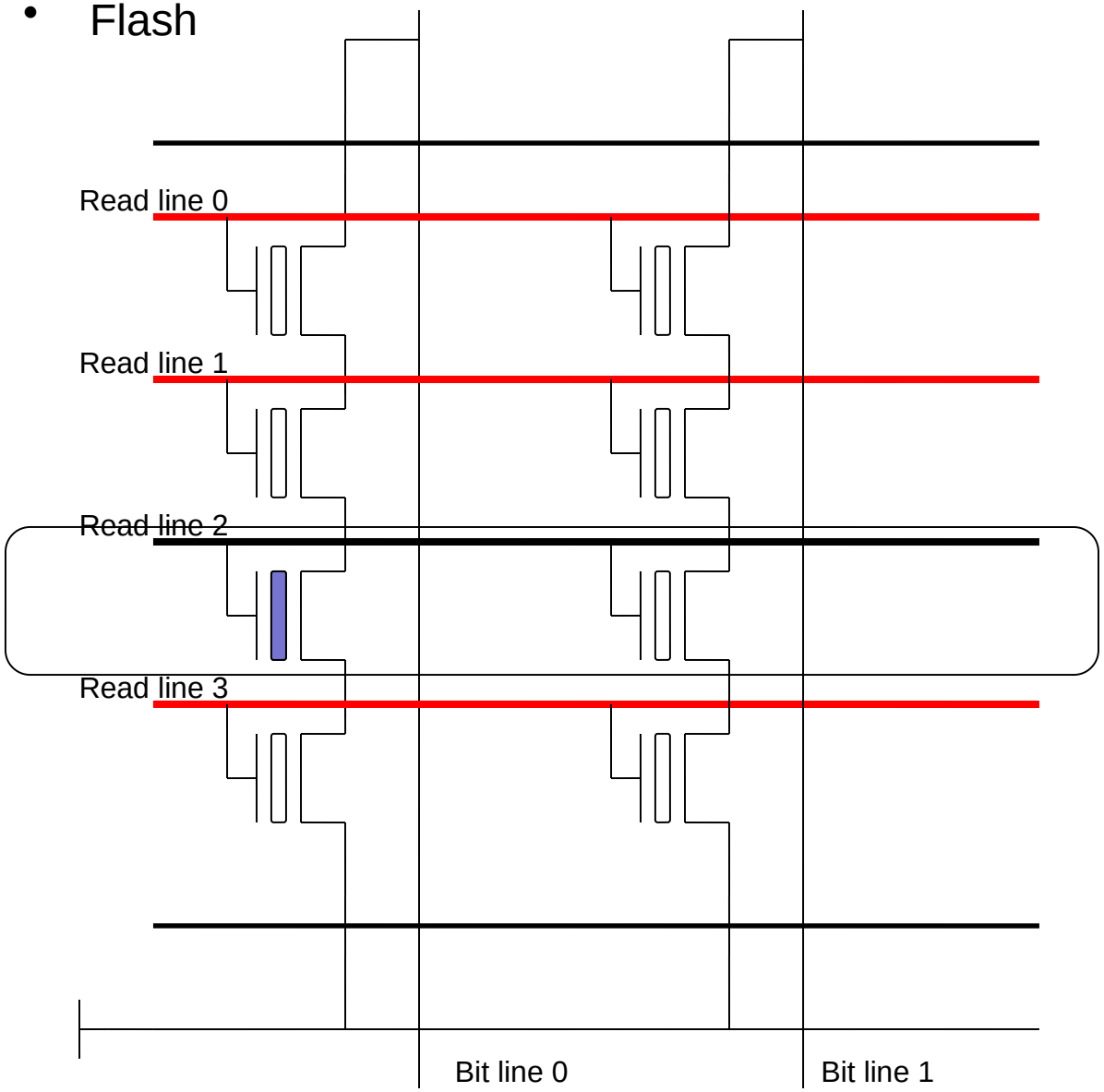
- Flash (http://de.slideshare.net/ruchiusha/07flash-memory-technology)

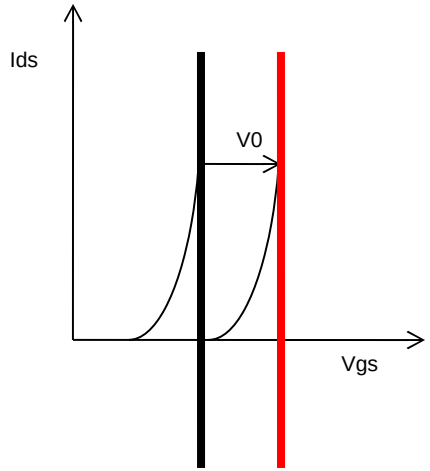




Read

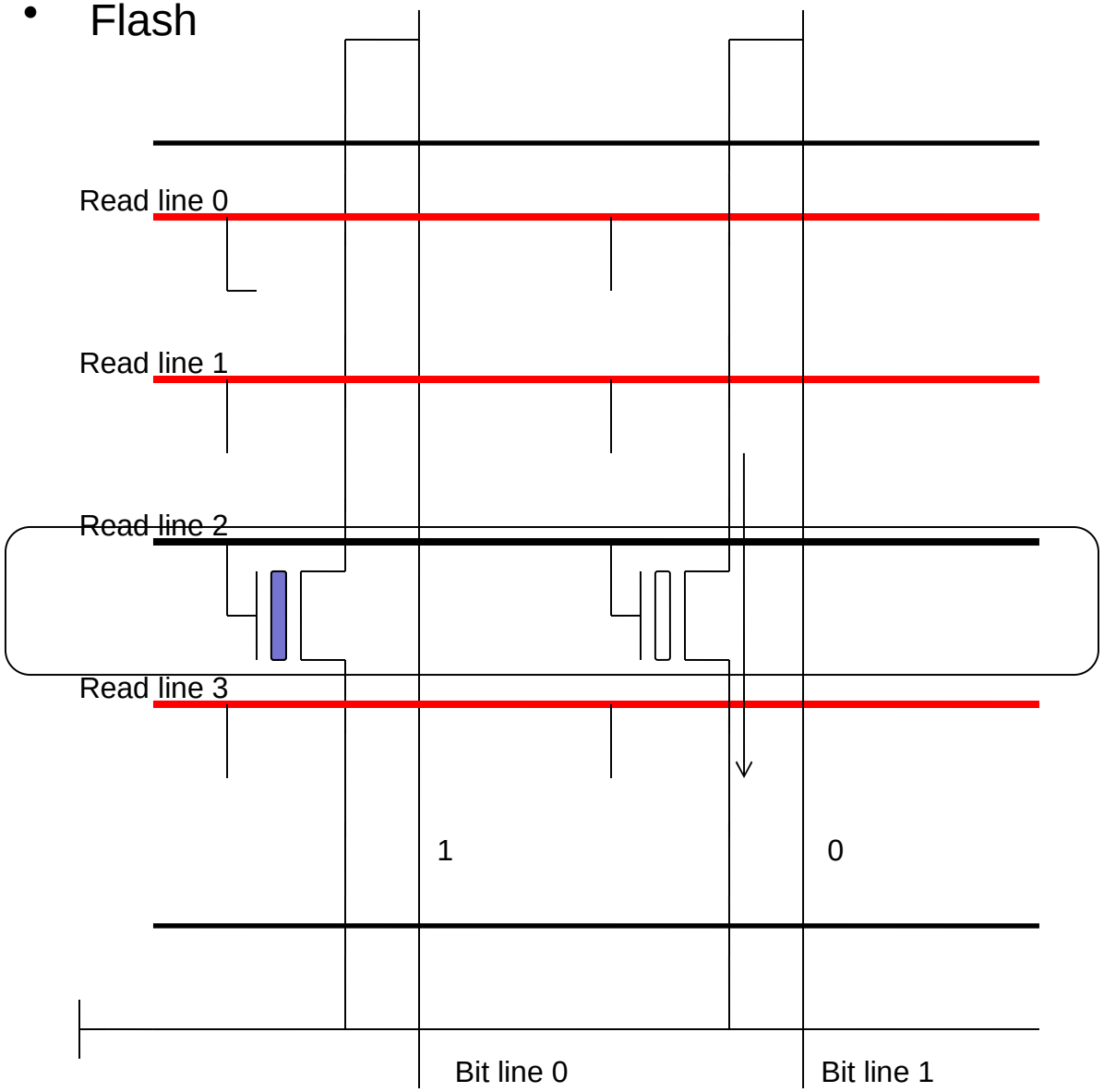
- Flash

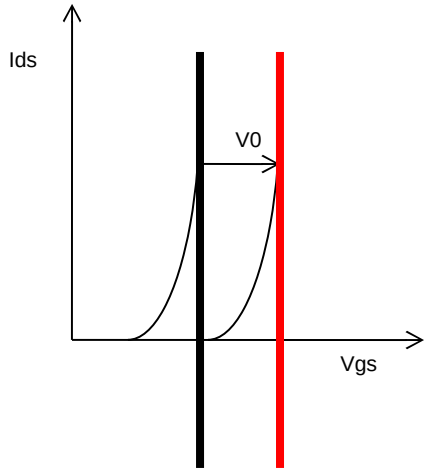

Read line 0
Read line 1
Read line 2
Read line 3
Bit line 0
Bit line 1


Ids
V0
Vgs

Read

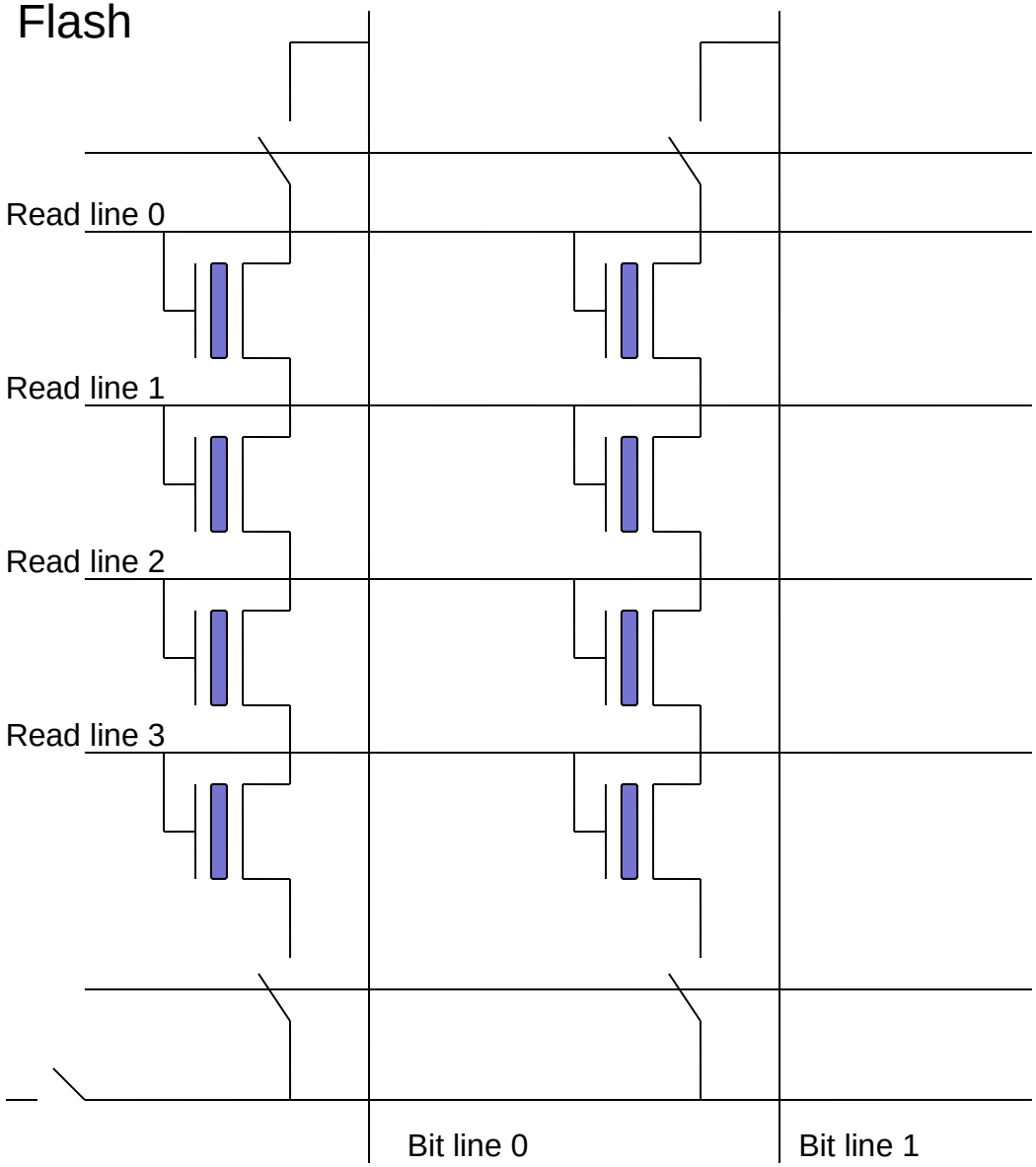
- Flash

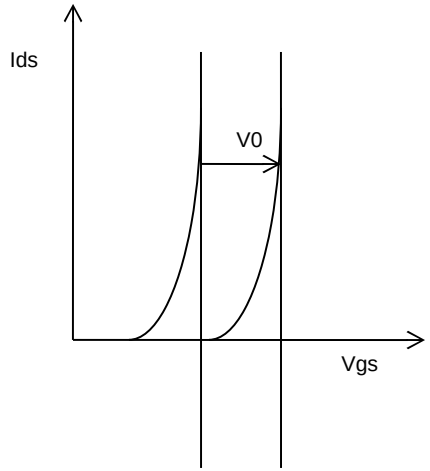

Read line 0
Read line 1
Read line 2
Read line 3
1
0
Bit line 0
Bit line 1


Ids
V0
Vgs

Erase

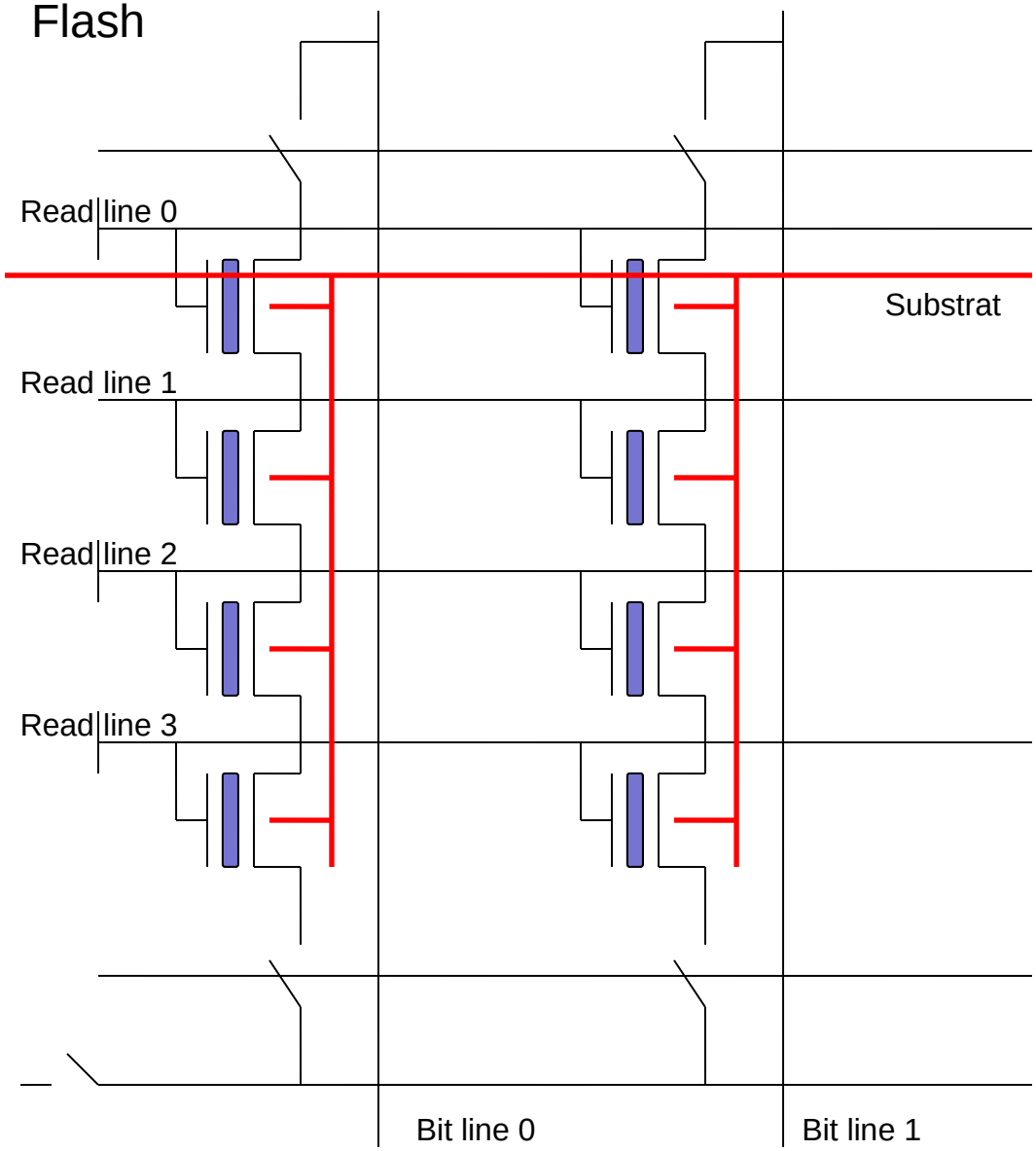
- Flash

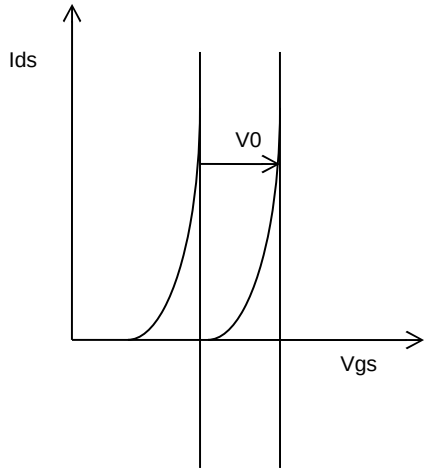

Read line 0
Read line 1
Read line 2
Read line 3
Bit line 0
Bit line 1


Ids
V0
Vgs

Erase

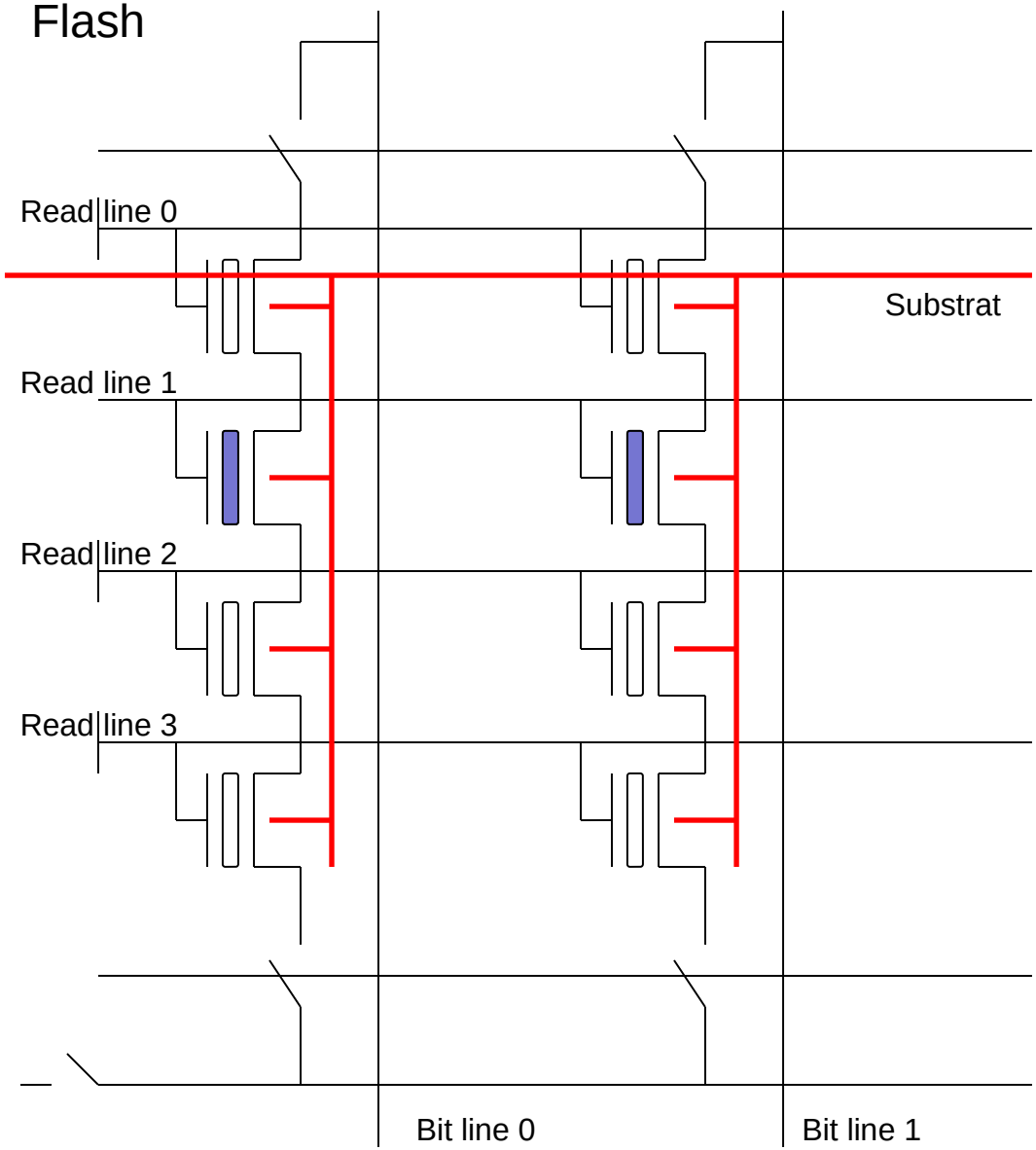
- Flash

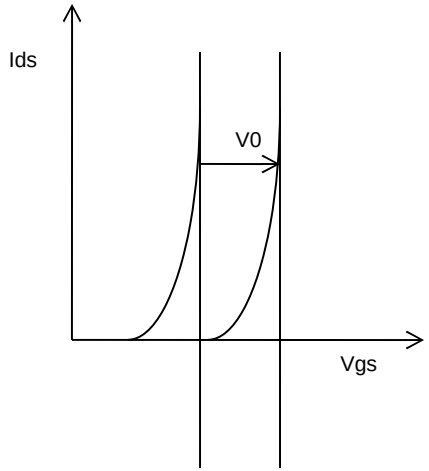

Read line 0
Substrat
Read line 1
Read line 2
Read line 3
Bit line 0
Bit line 1


Ids
V0
Vgs

Erase

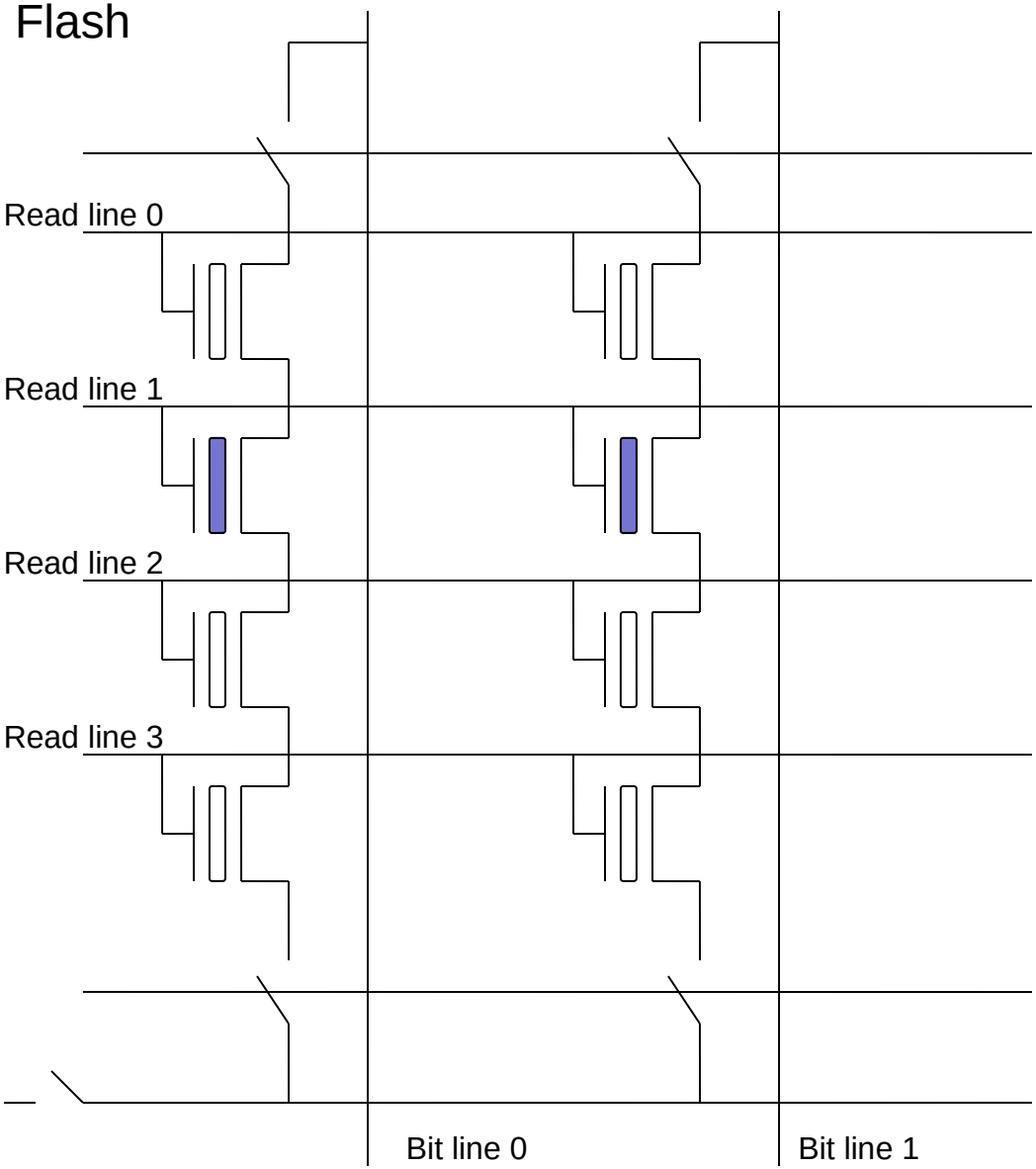
- Flash

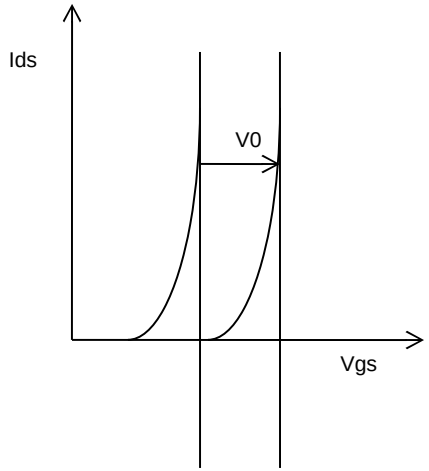

Read line 0
Substrat
Read line 1
Read line 2
Read line 3
Bit line 0
Bit line 1


Ids
V0
Vgs

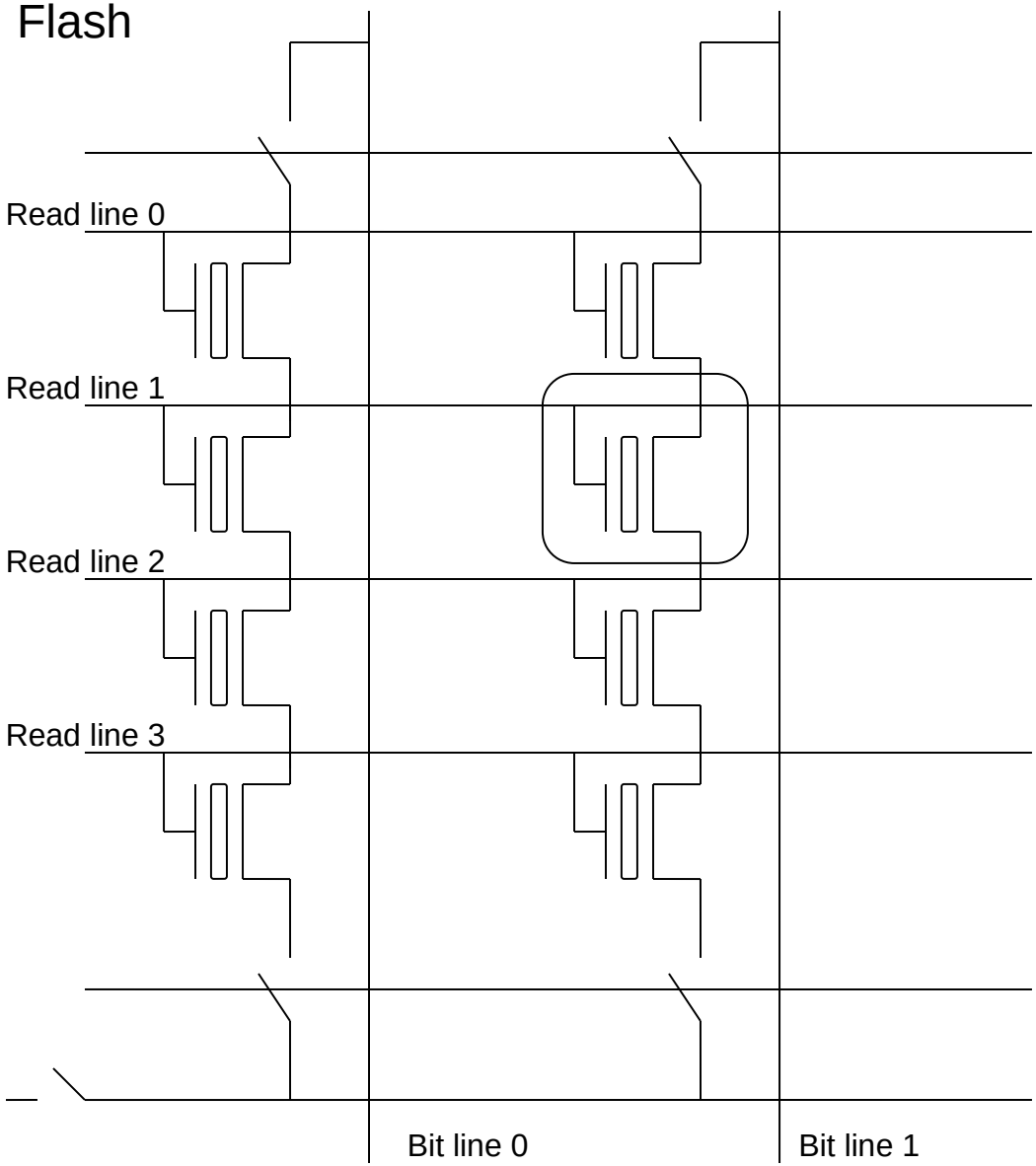
Erase

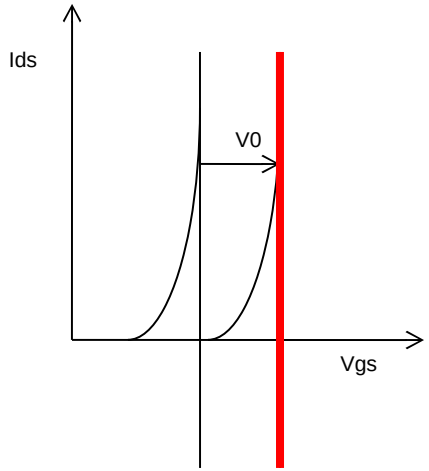
- Flash


Read line 0
Read line 1
Read line 2
Read line 3
Bit line 0
Bit line 1

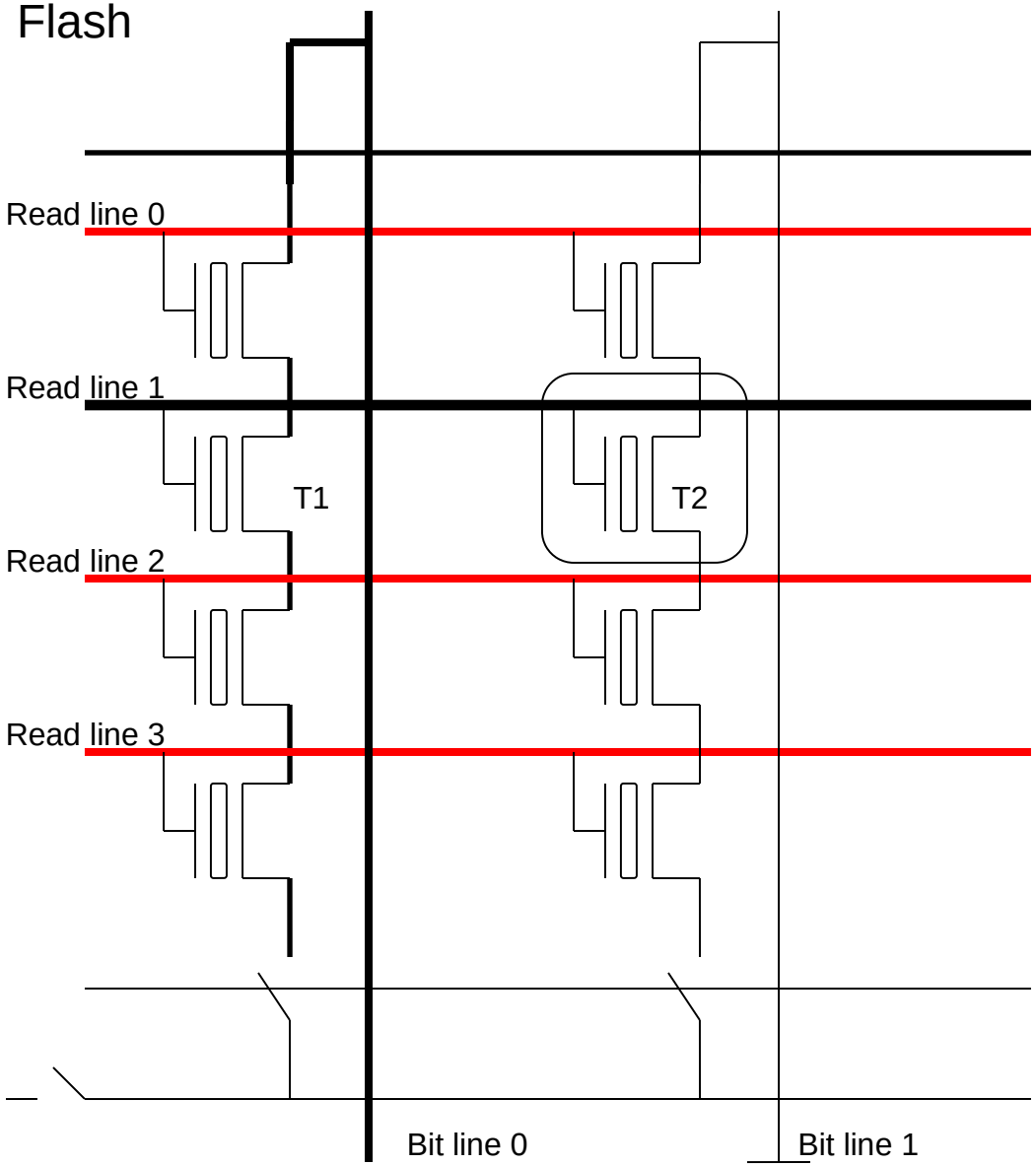

Ids
V0
Vgs

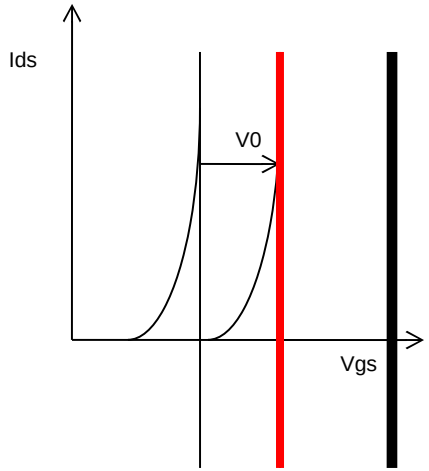
- Flash


Read line 0
Read line 1
Read line 2
Read line 3
Bit line 0
Bit line 1


Ids
V0
Vgs

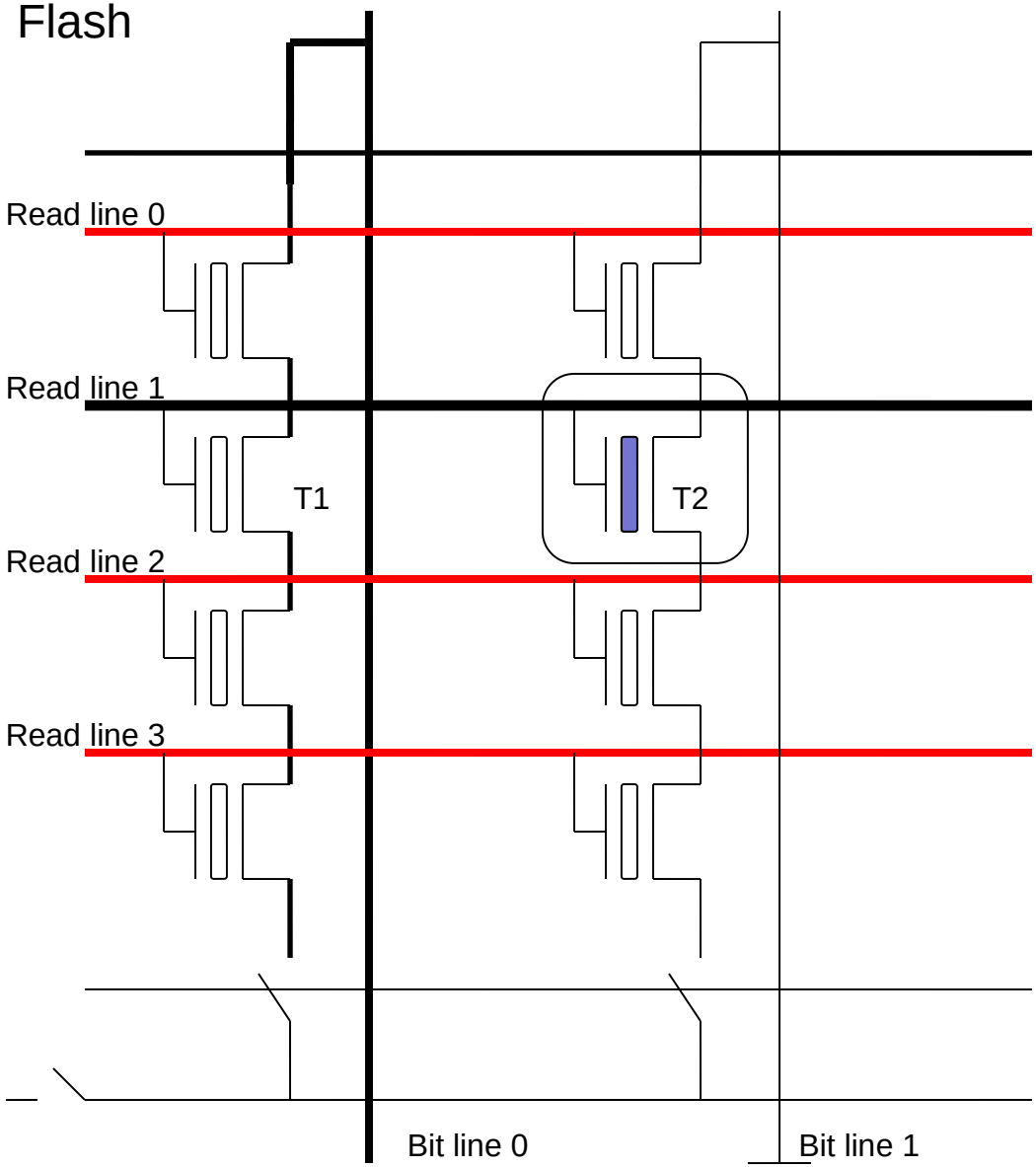
- Flash

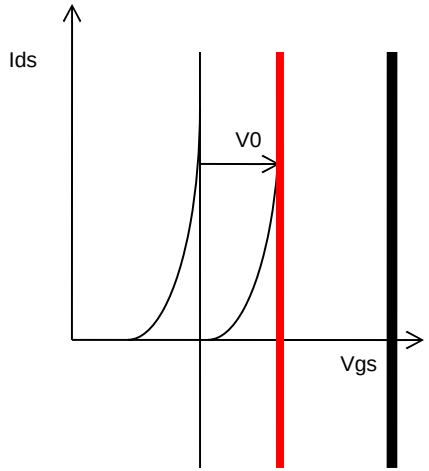

Read line 0
Read line 1
Read line 2
Read line 3
T1
T2
Bit line 0
Bit line 1


Ids
V0
Vgs

Program

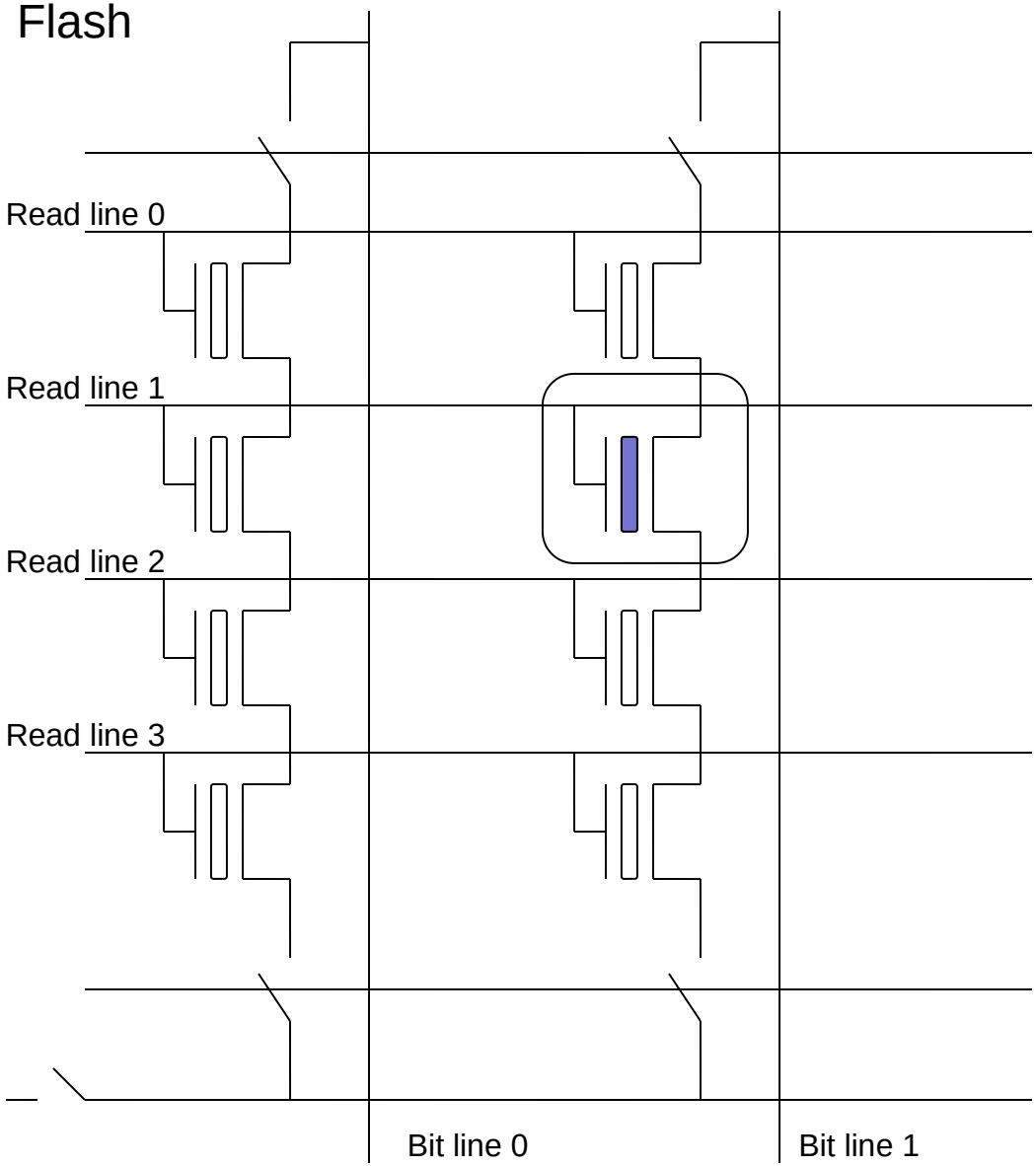
- Flash

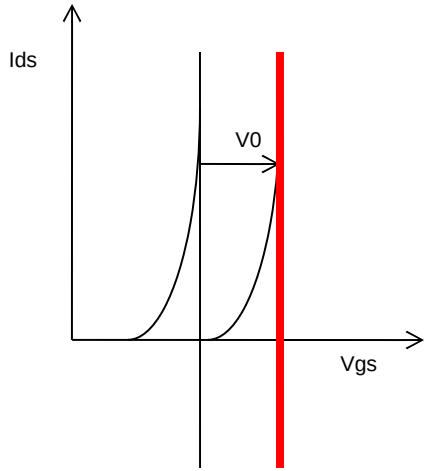




Program

- Flash

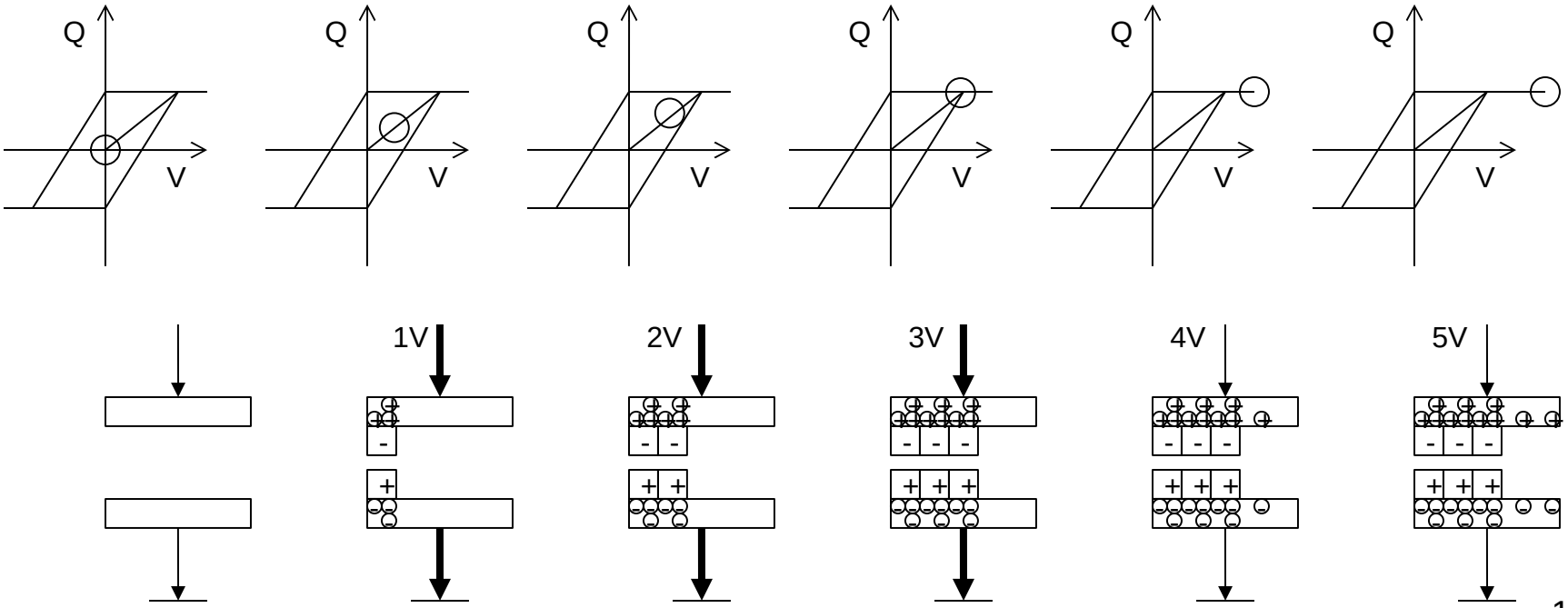




Ferroelectric RAM

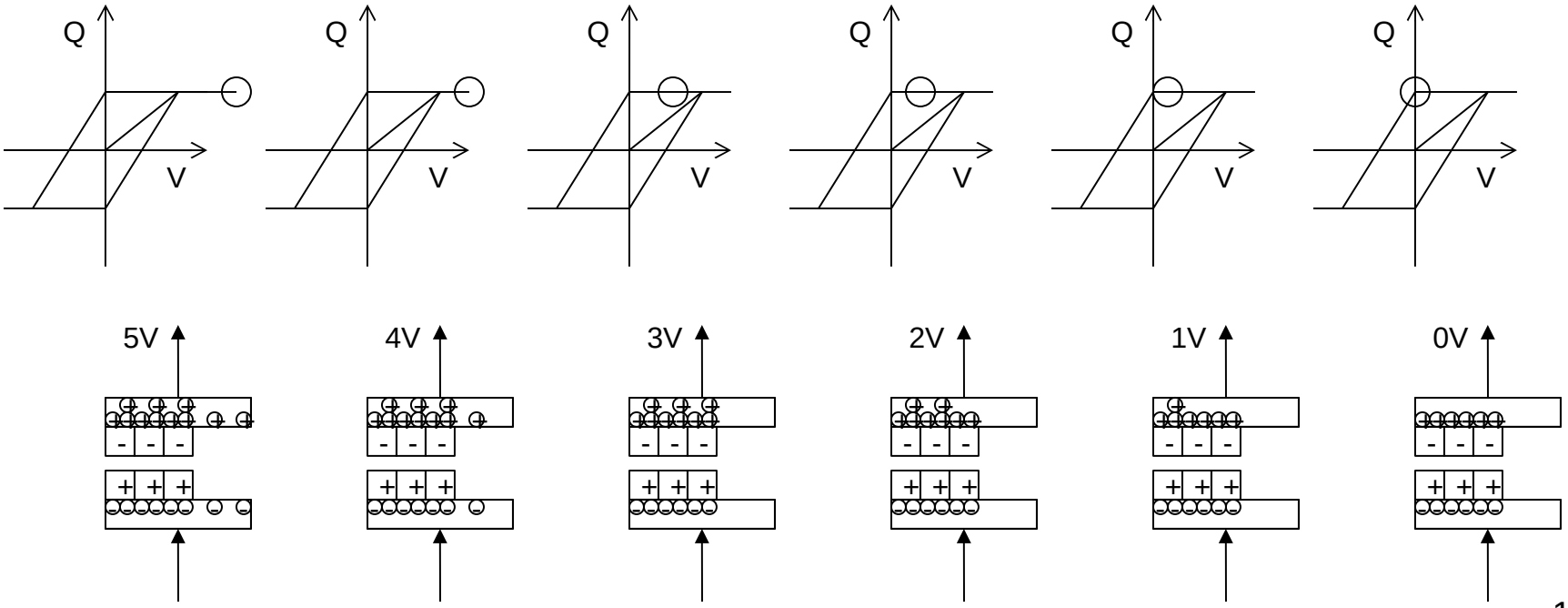
Ferroelectric SRAM

- Polarisation -> high current



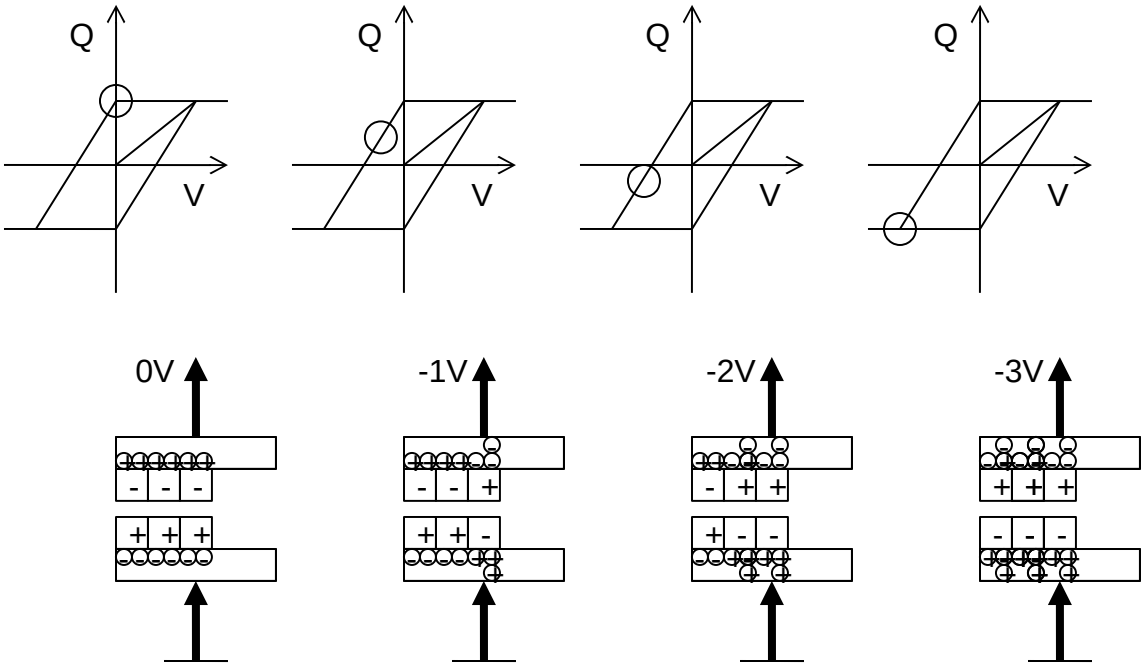
Ferroelectric RAM

- Maintaining of palarity -> low current



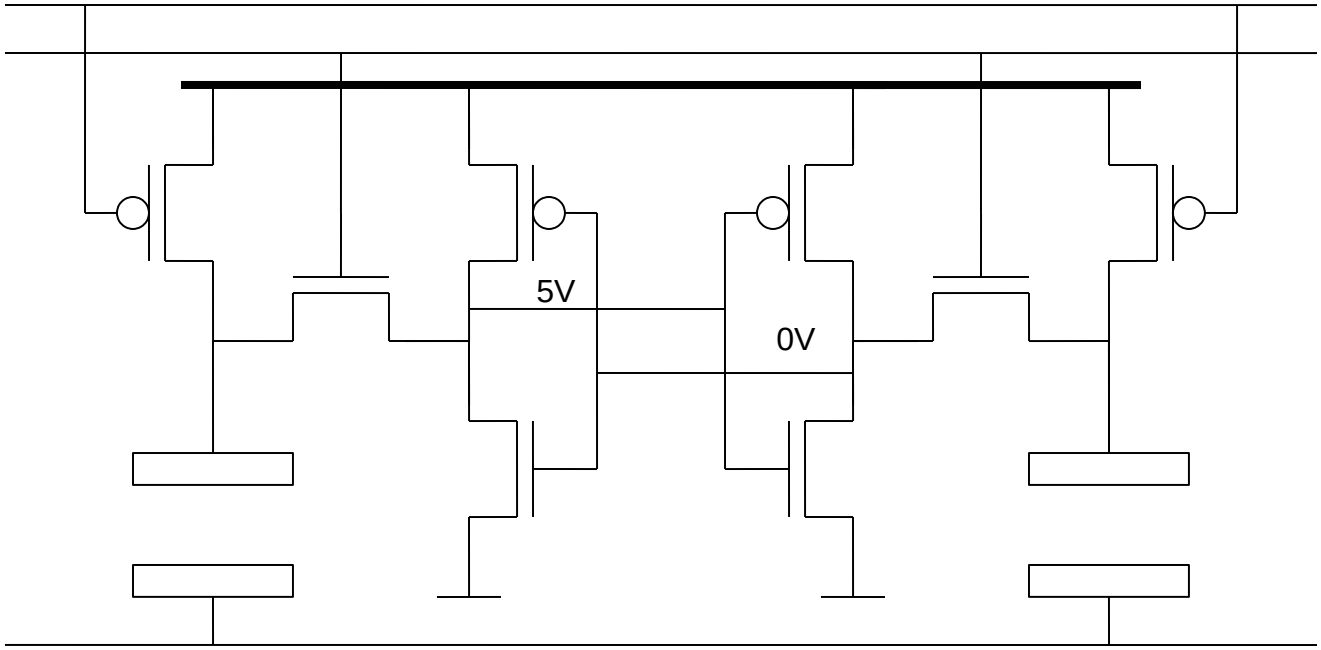
Ferroelectric RAM

- Reversal of polarity -> high current



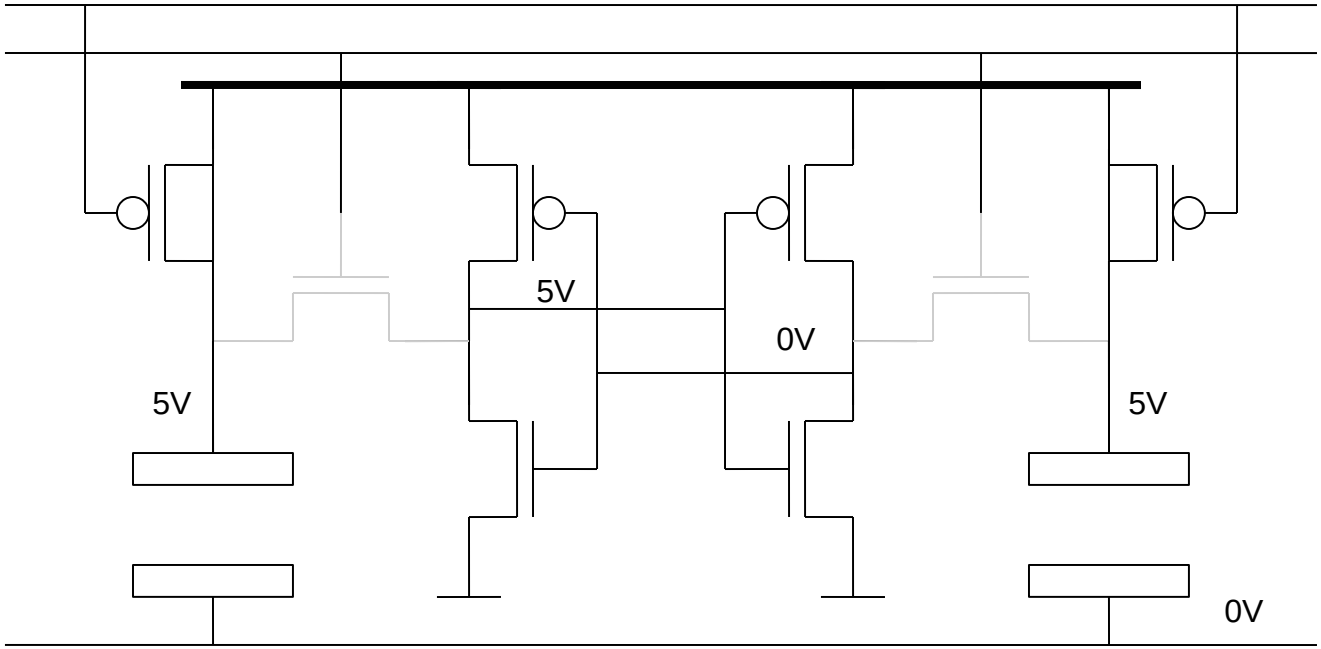
Ferroelectric RAM

- Precharge



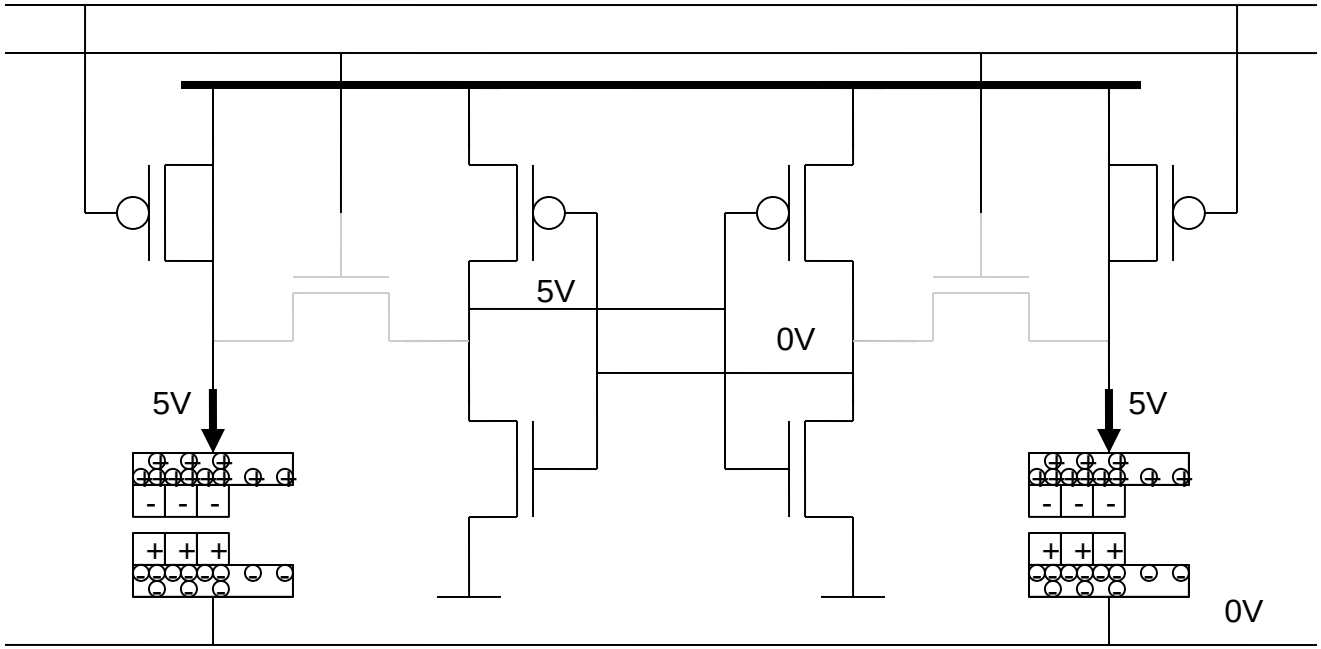
Ferroelectric RAM

- Precharge



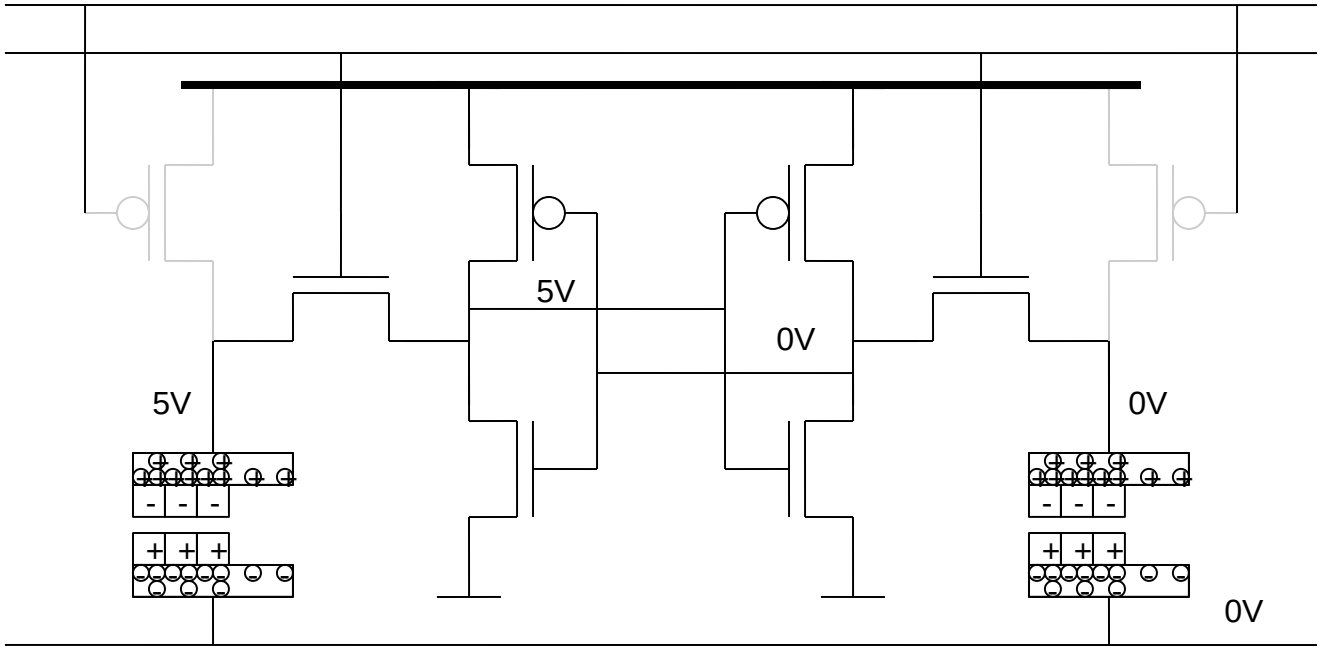
Ferroelectric RAM

- Precharge



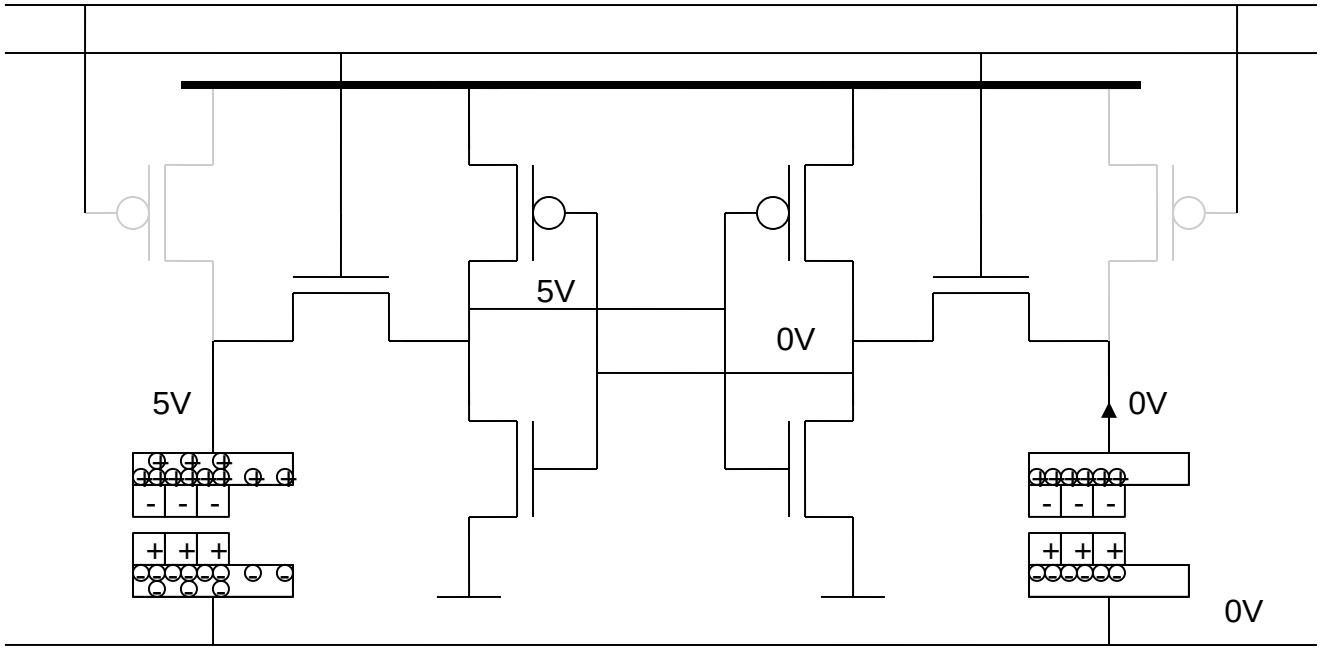
Ferroelectric RAM

- Write FERAM



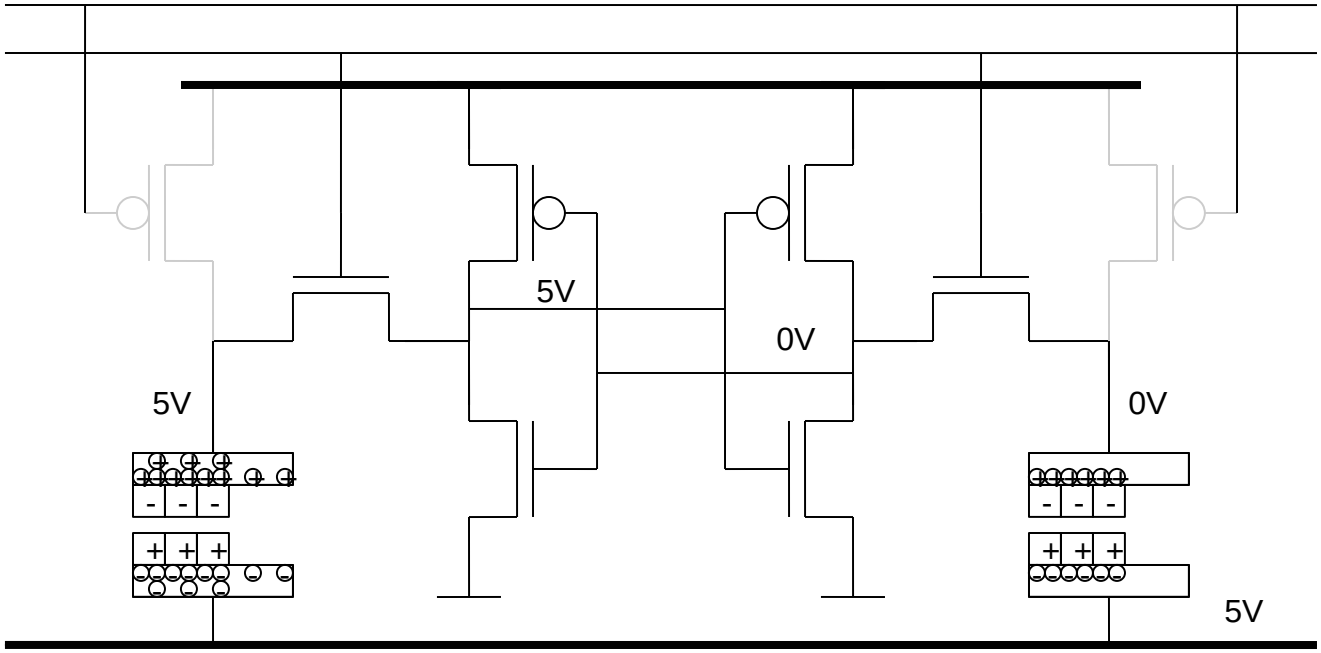
Ferroelectric RAM

- Write FERAM



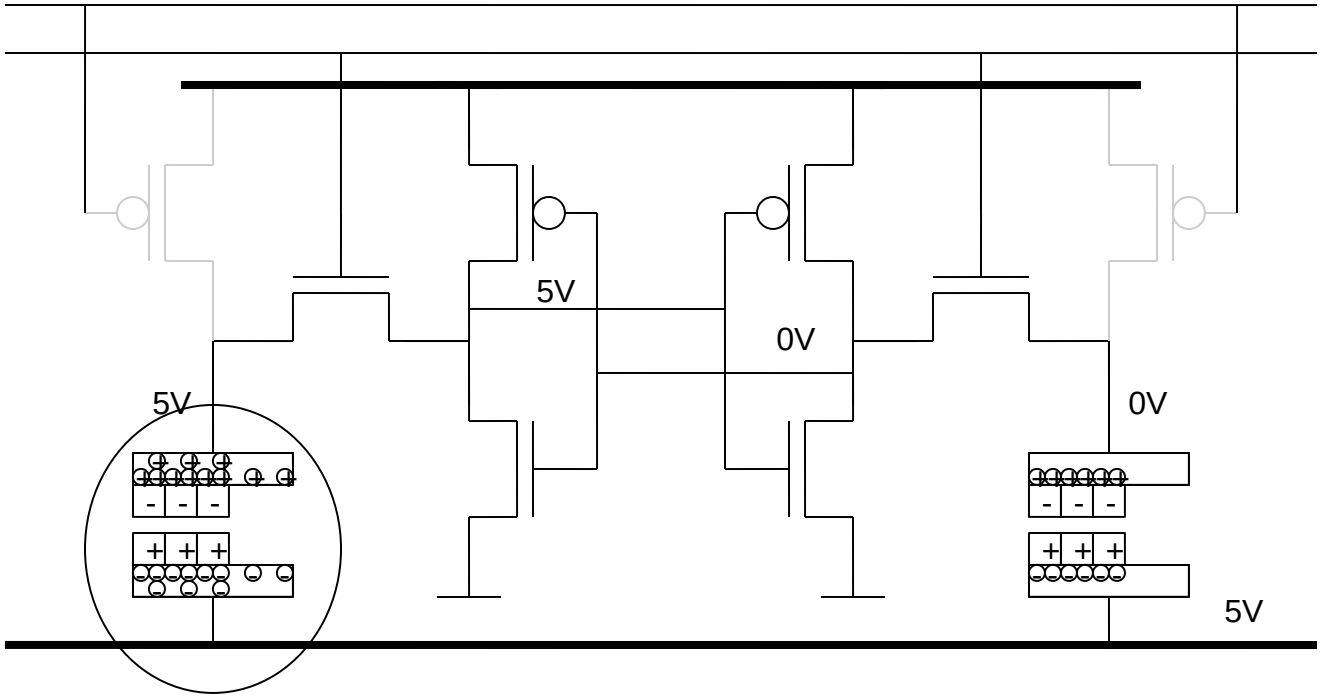
Ferroelectric RAM

- Write FERAM



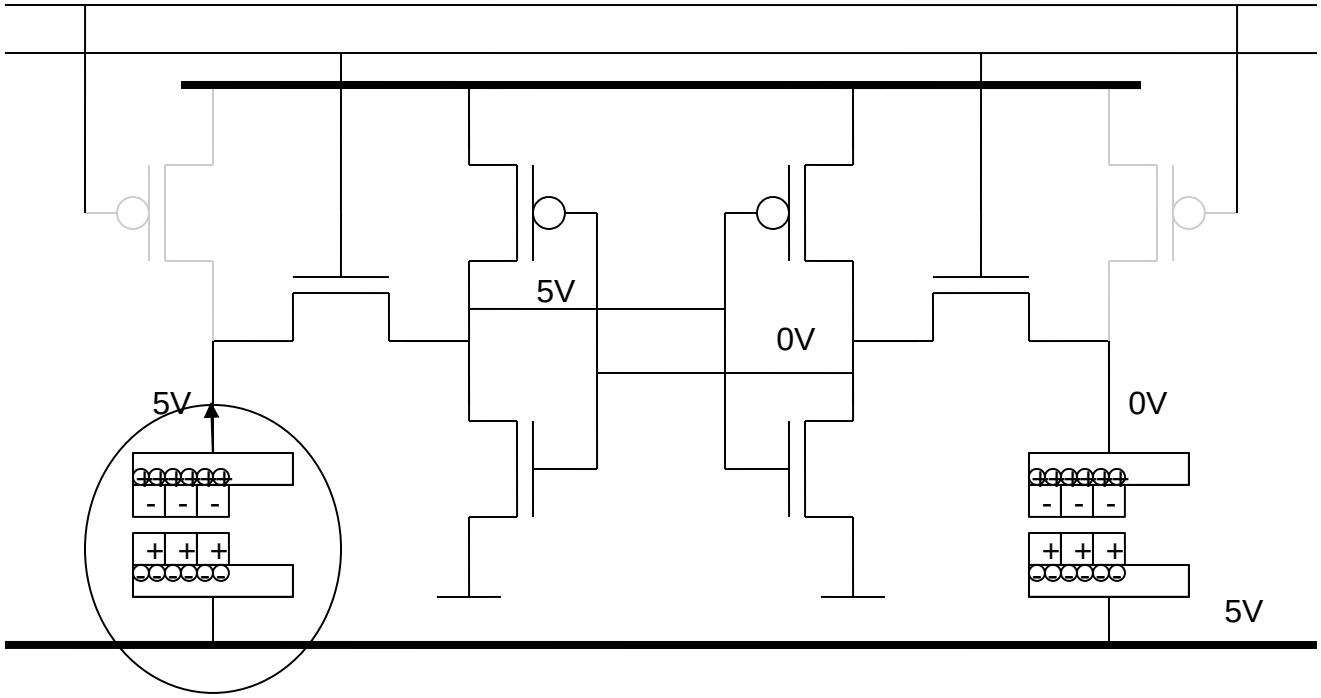
Ferroelectric RAM

- Write FERAM



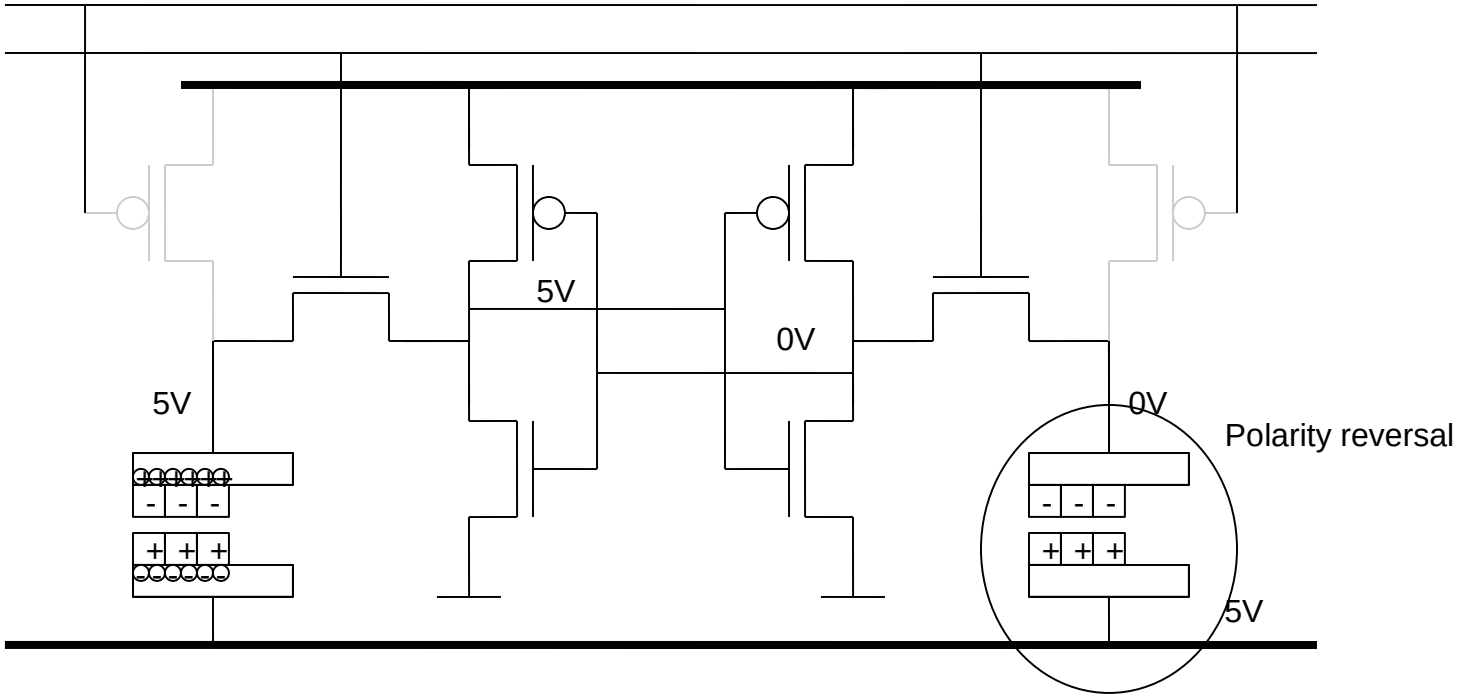
Ferroelectric RAM

- Write FERAM



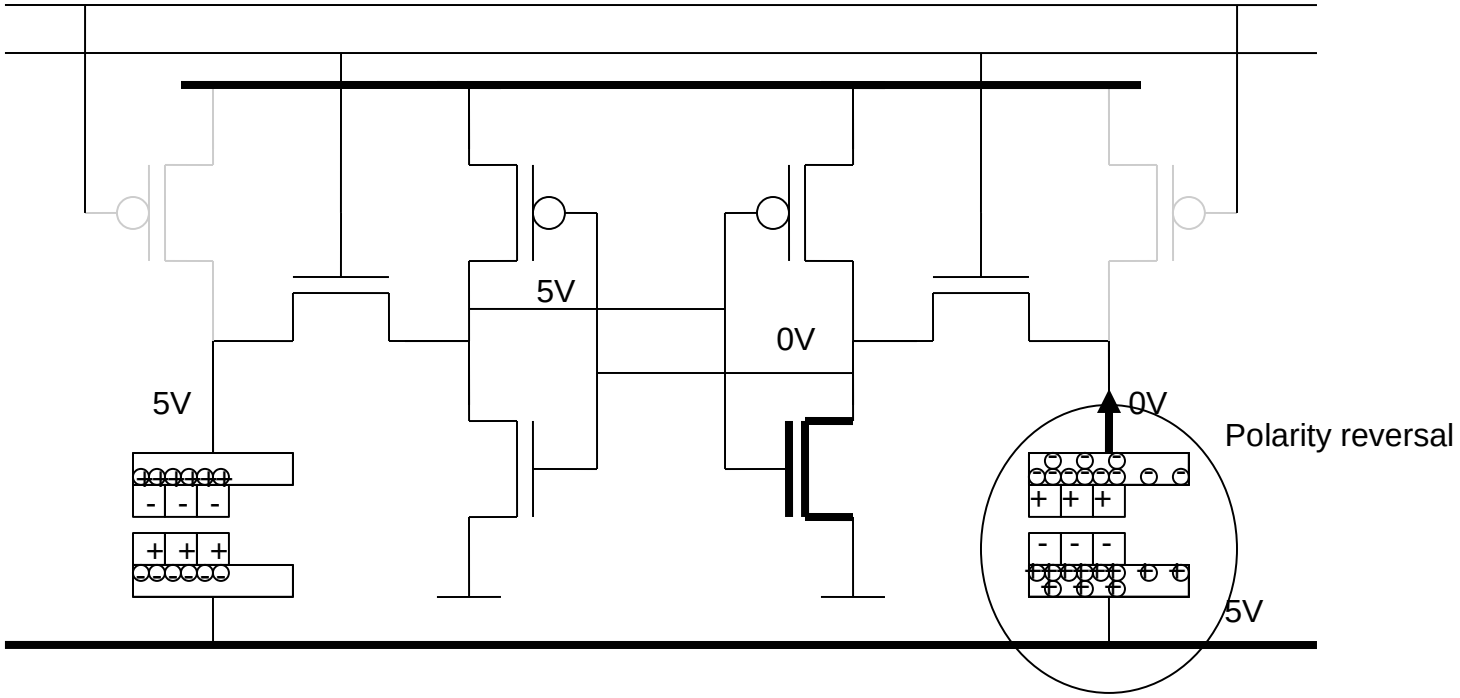
Ferroelectric RAM

- Write FERAM



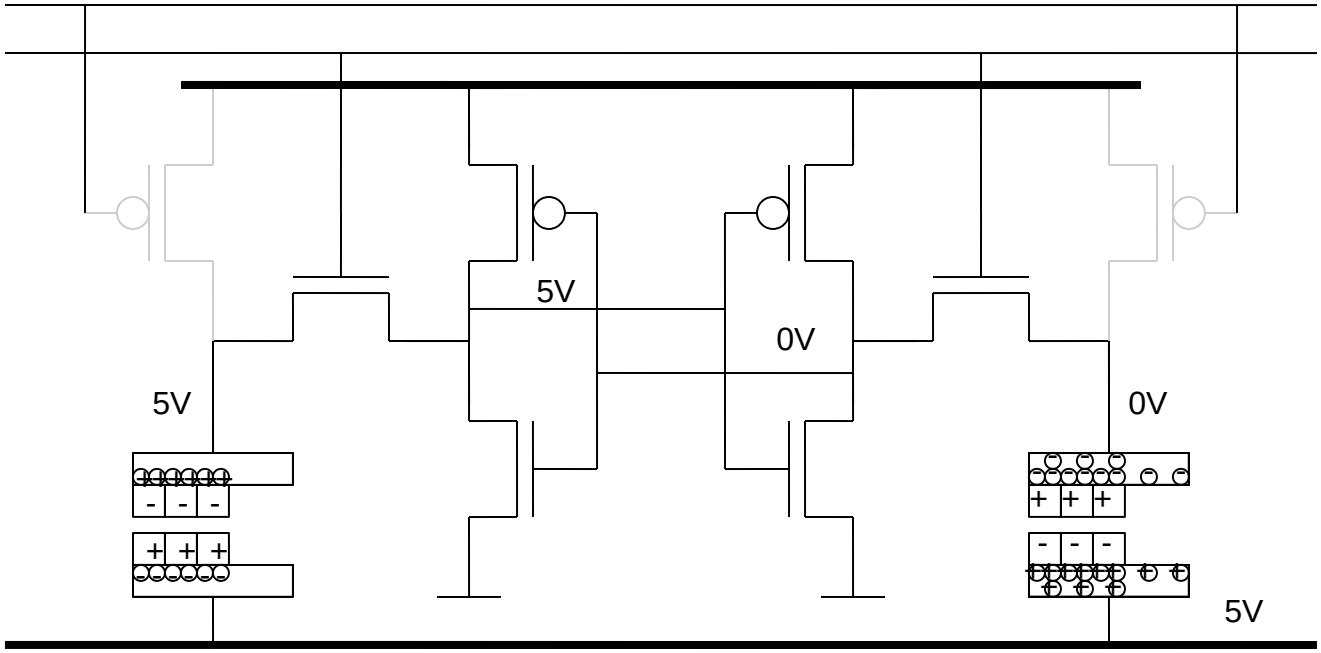
Ferroelectric RAM

- Write FERAM



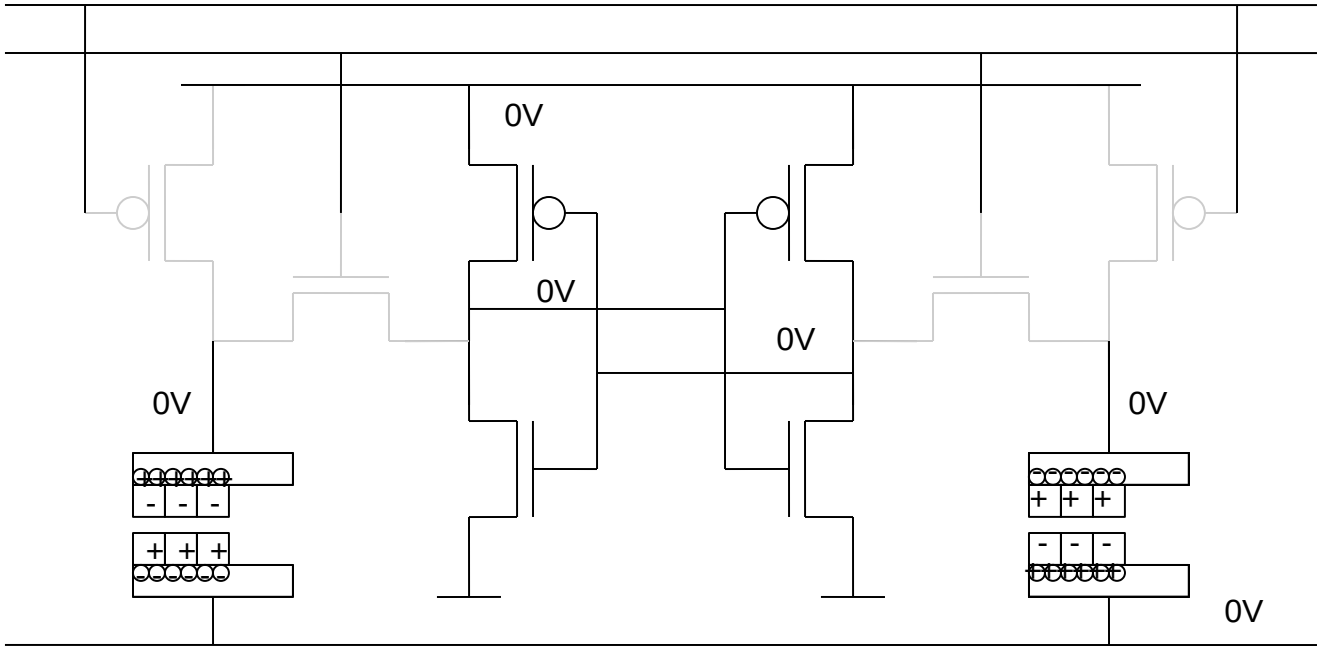
Ferroelectric RAM

- Write FERAM



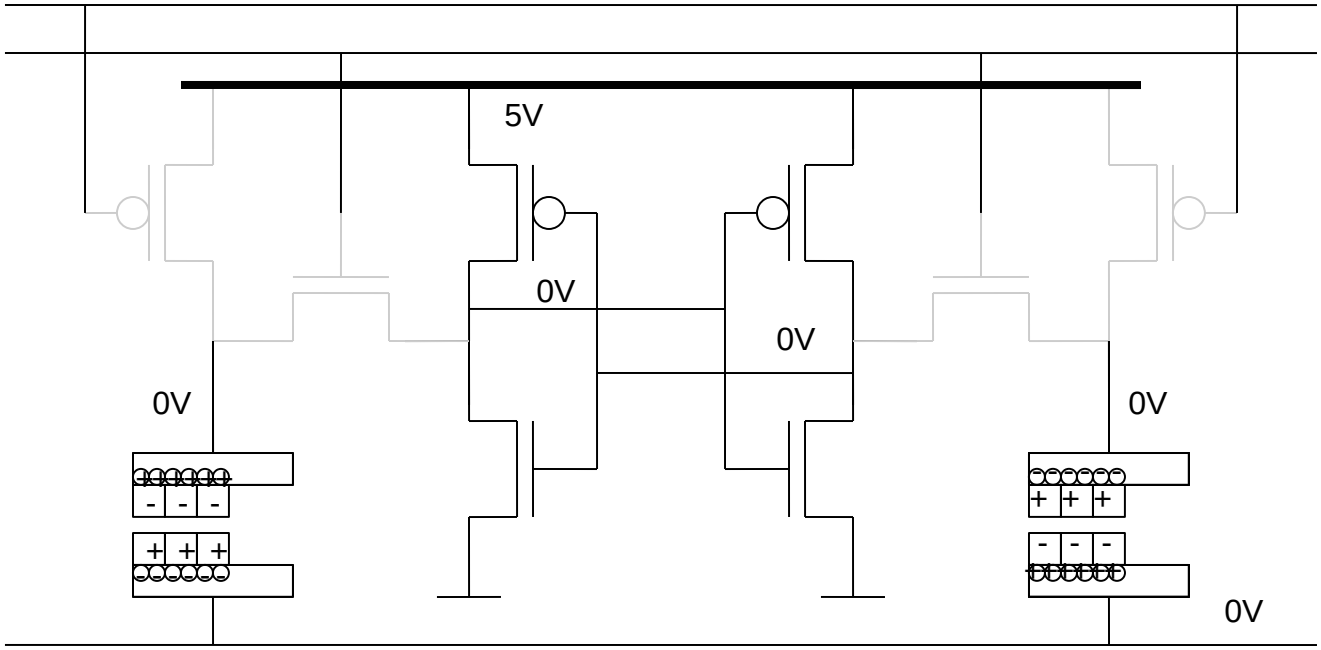
Ferroelectric RAM

- Power off



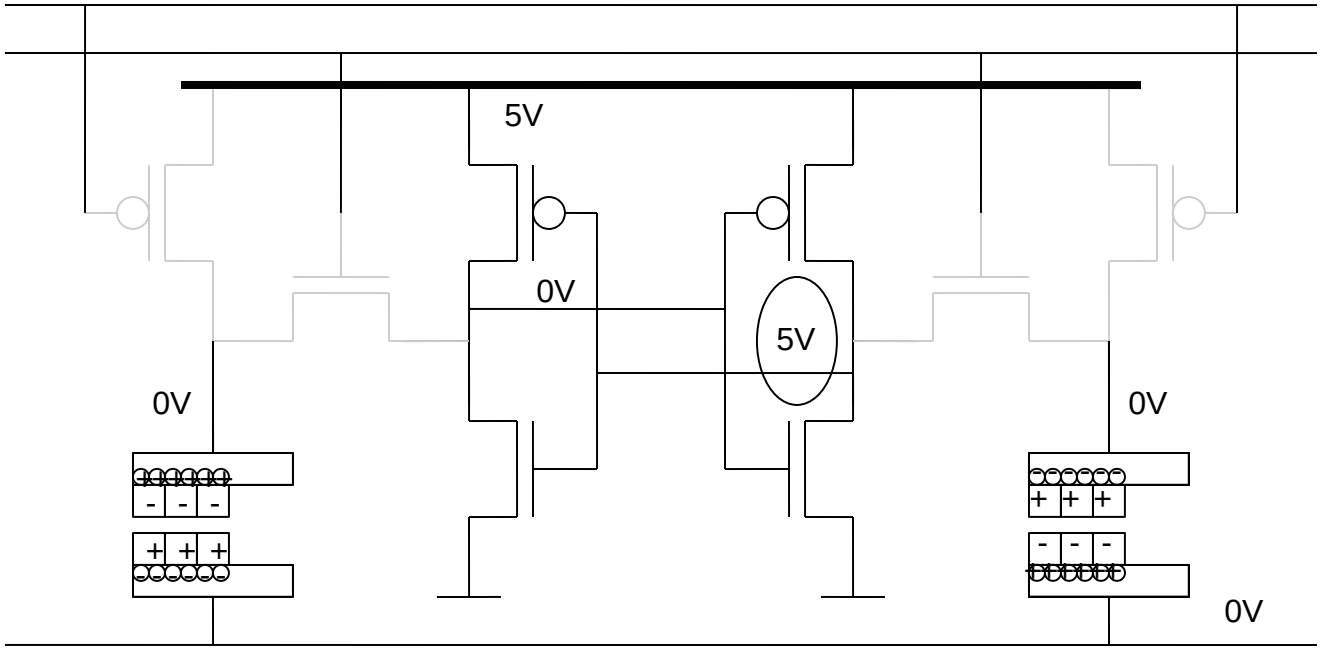
Ferroelectric RAM

- Power on



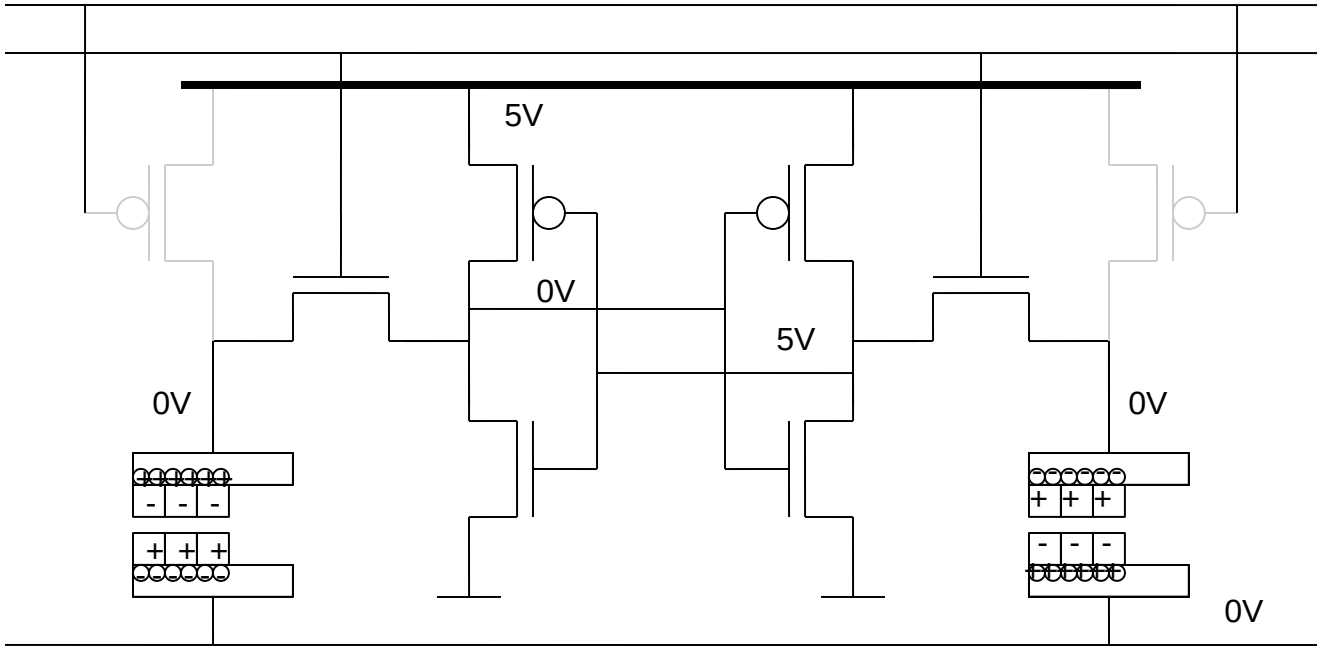
Ferroelectric RAM

- Power on



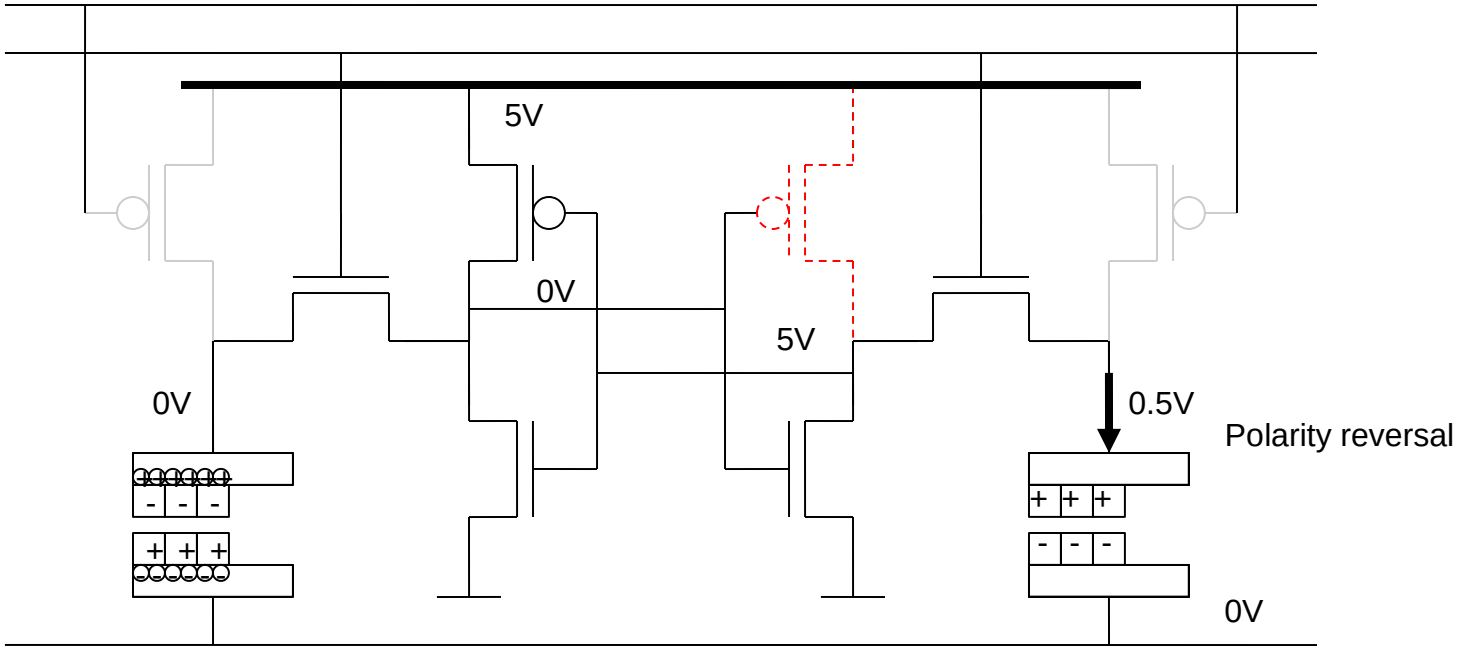
Ferroelectric RAM

- Write SRAM



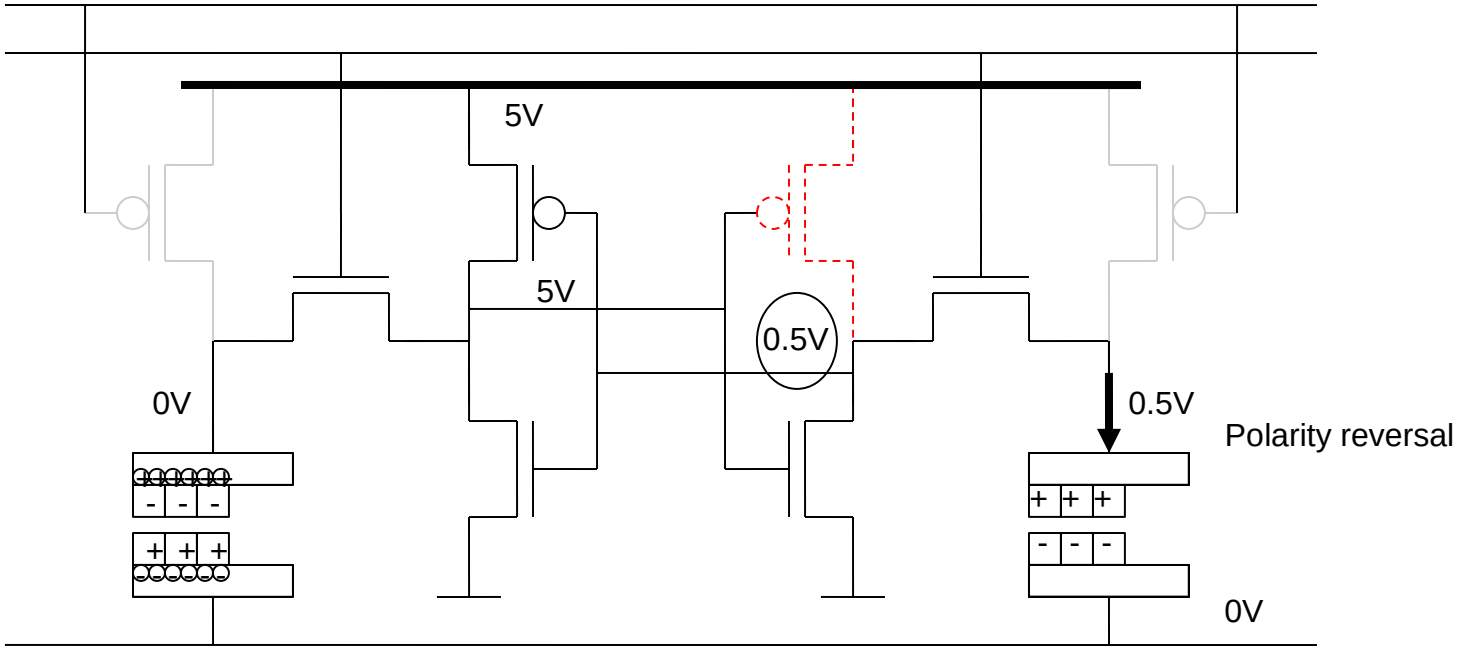
Ferroelectric RAM

- Write SRAM



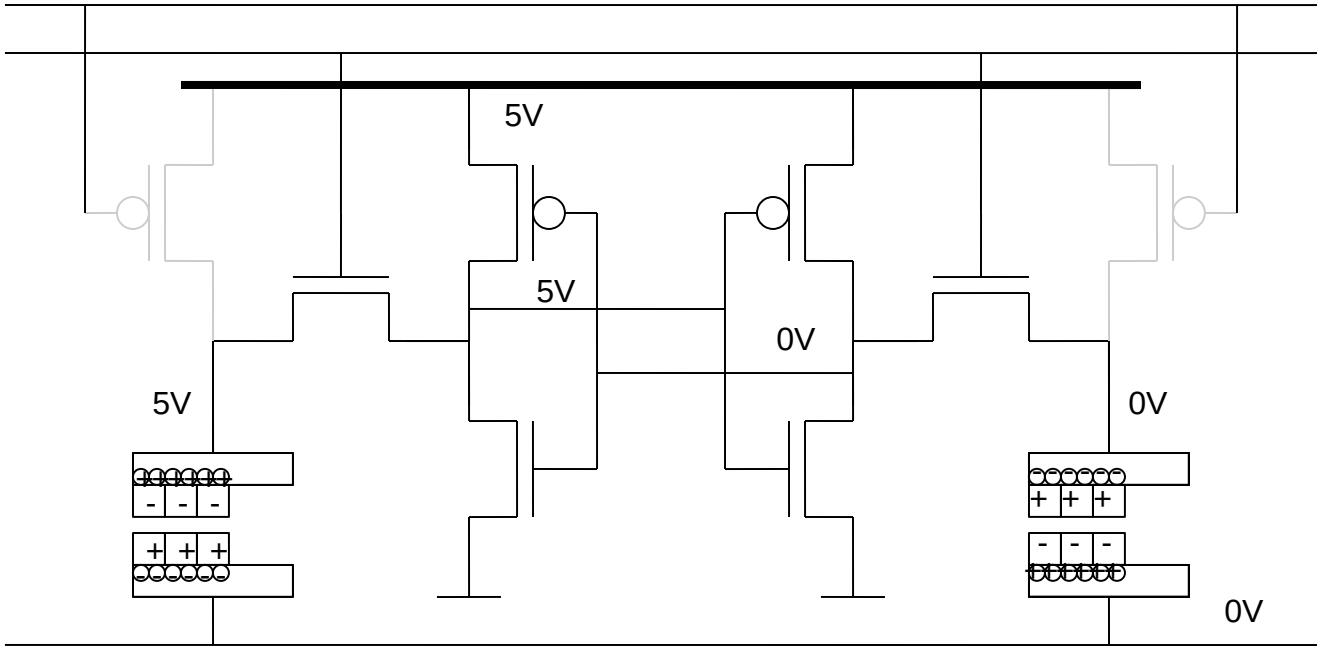
Ferroelectric RAM

- Write SRAM



Ferroelectric RAM

- Write SRAM



Ferroelectric RAM

- Write SRAM