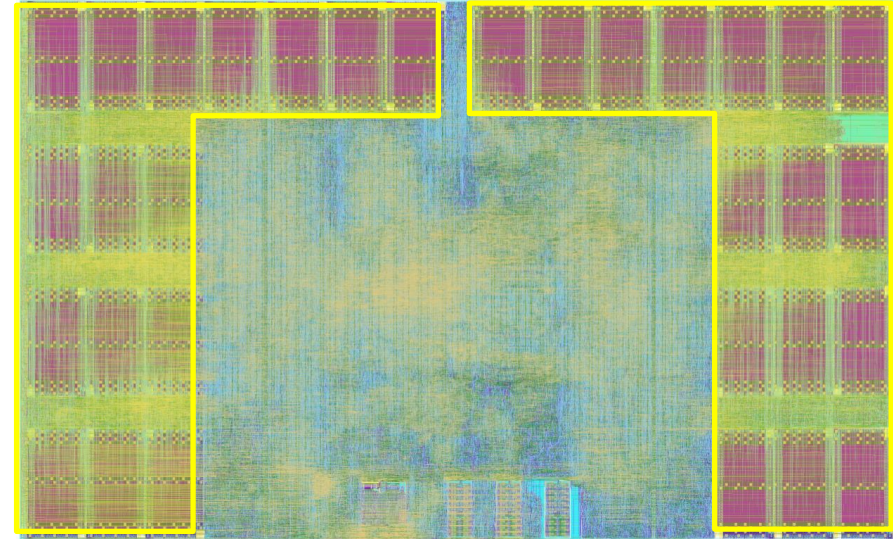


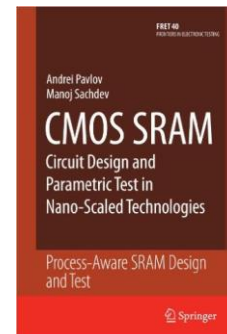
SRAM Compiler

- Vorteile:
 - Hohe Speicherdichten
 - Schnelles Schreiben und Lesen
- Nachteile:
 - Flüchtig Speicher
- Anwendung:
 - On-Chip Memory für Daten und Instruktionen, Caches, FIFOs, ...
- In aktuellen SoCs dominiert der SRAM die Chipfläche!

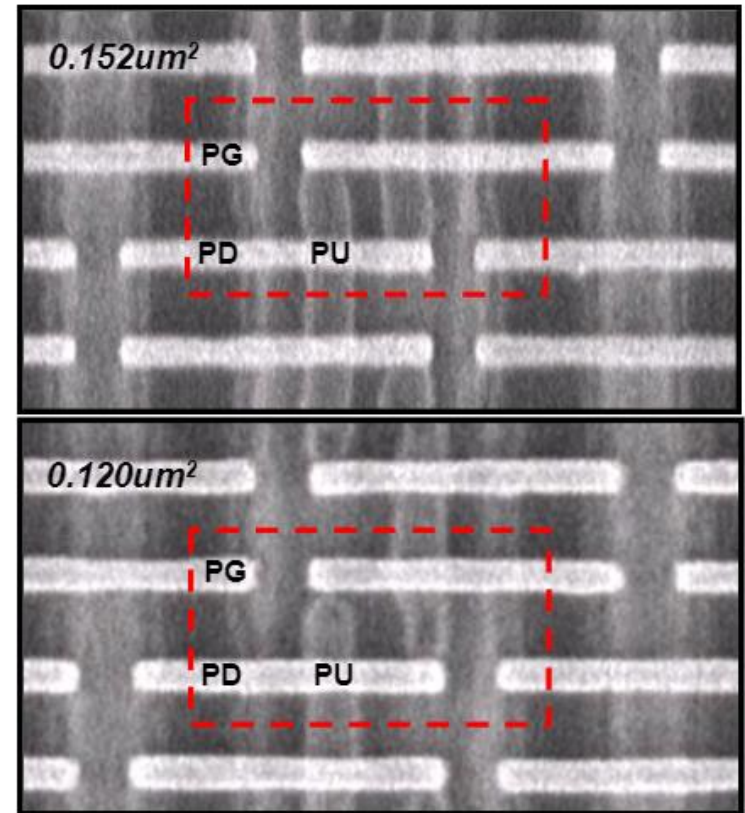
- Literatur: Andrei Pavlov, Manoj Sachdev; CMOS SRAM Circuit Design and Parametric Test in Nano-Scaled Technologies, SPRINGER



Beispiel: DuoPE auf Santos
≈50% der Fläche SRAM

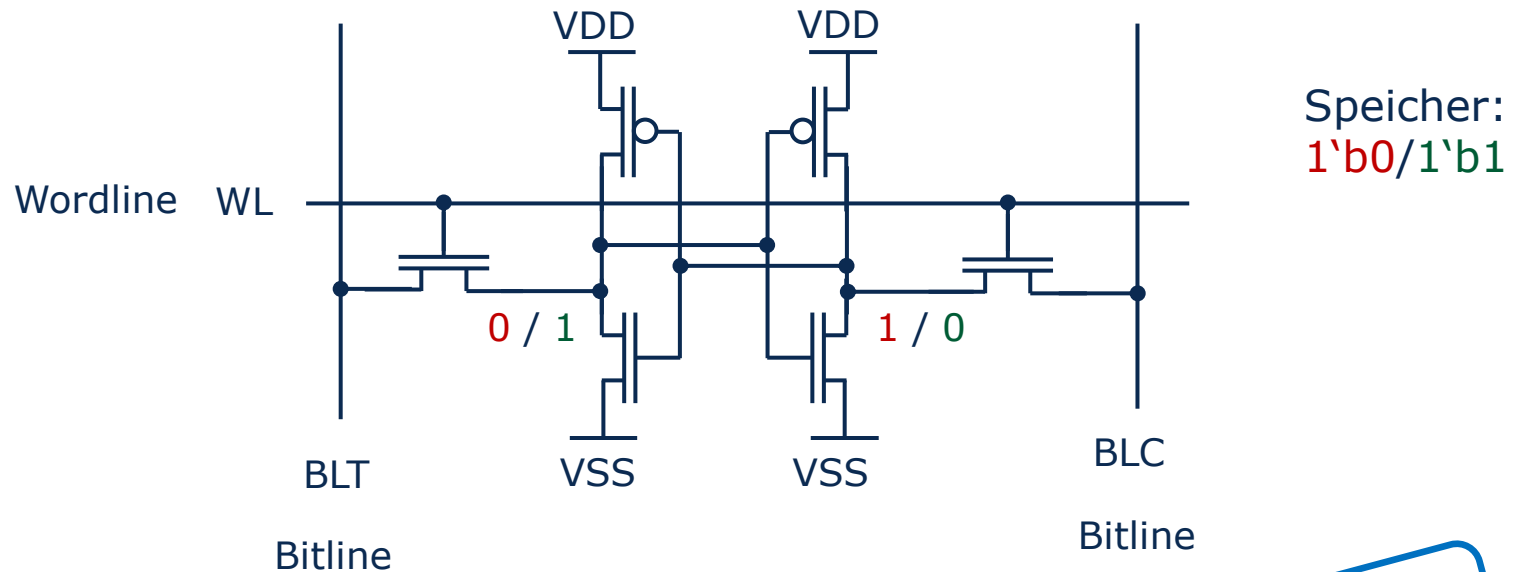


- SRAM Zellen beinhalten die **kleinsten Strukturgrößen** der jeweiligen Herstellungstechnologie
- Häufig sind SRAM Zellen nicht DRC clean mit den Standard Regeln
- Spezielle DRC Regeln für SRAMs
- Feste Layoutvorgaben für die SRAM Zellen („golden bitcell layout“)
- Beispiel: 28nm, Single Port 6T Zelle:
 - 0.152 μm^2 High Performance
 - 0.120 μm^2 High Density
 - ca. 7 Bits/ μm^2

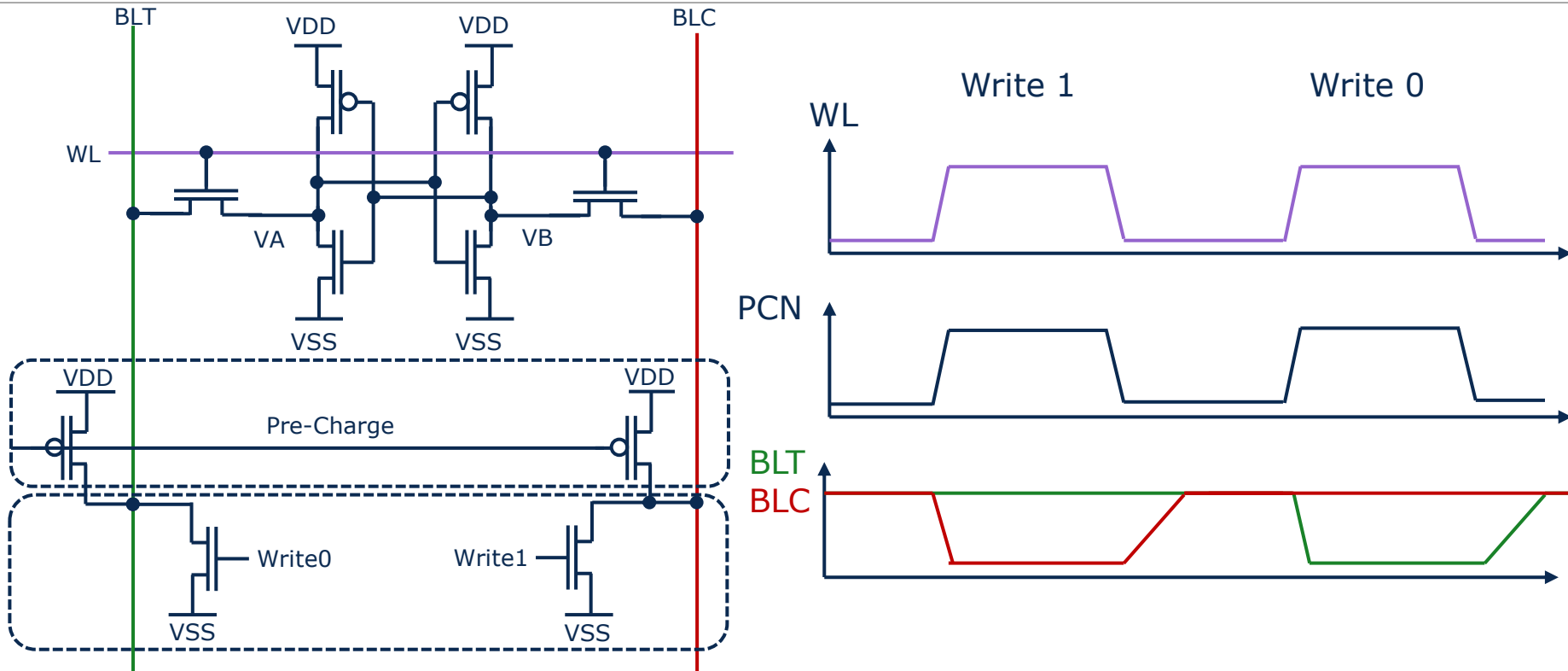


Quelle: Arnaud, F.; et.al. "Competitive and cost effective high-k based 28nm CMOS technology for low power applications," in Electron Devices Meeting (IEDM), 2009 IEEE International , vol., no., pp.1-4, 7-9 Dec. 2009

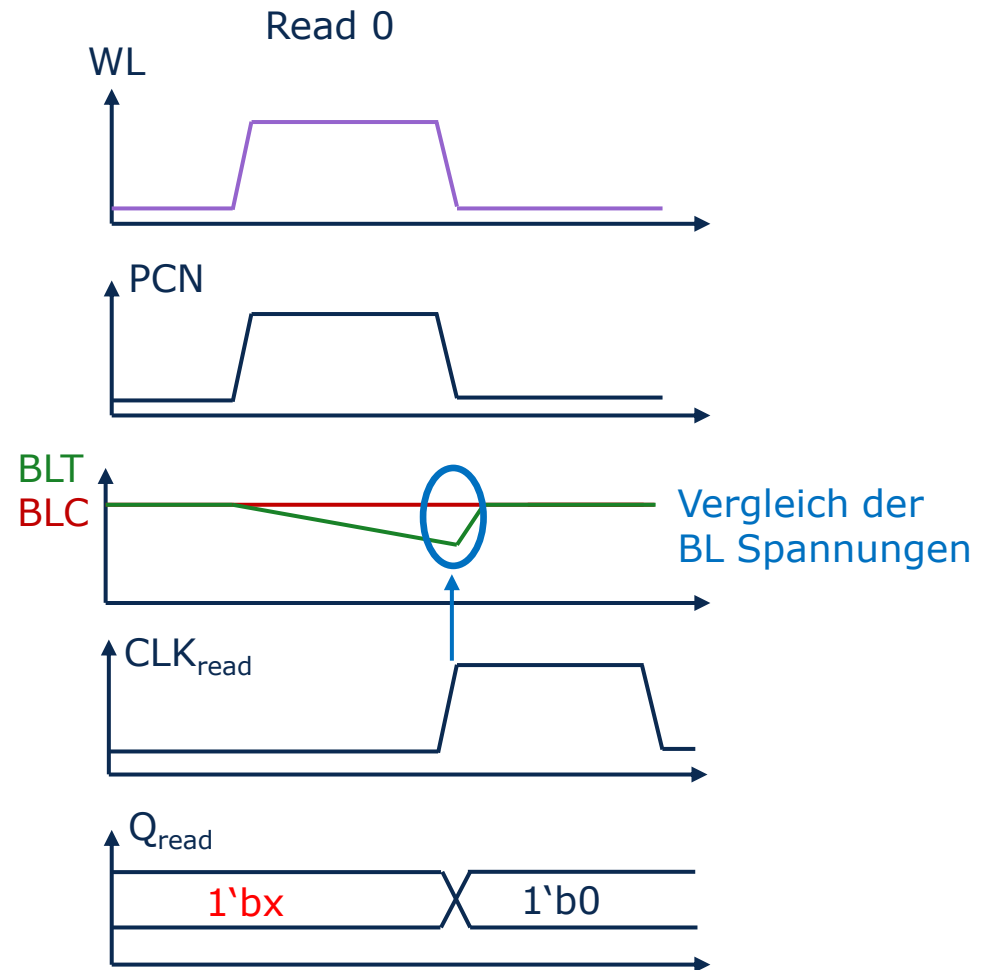
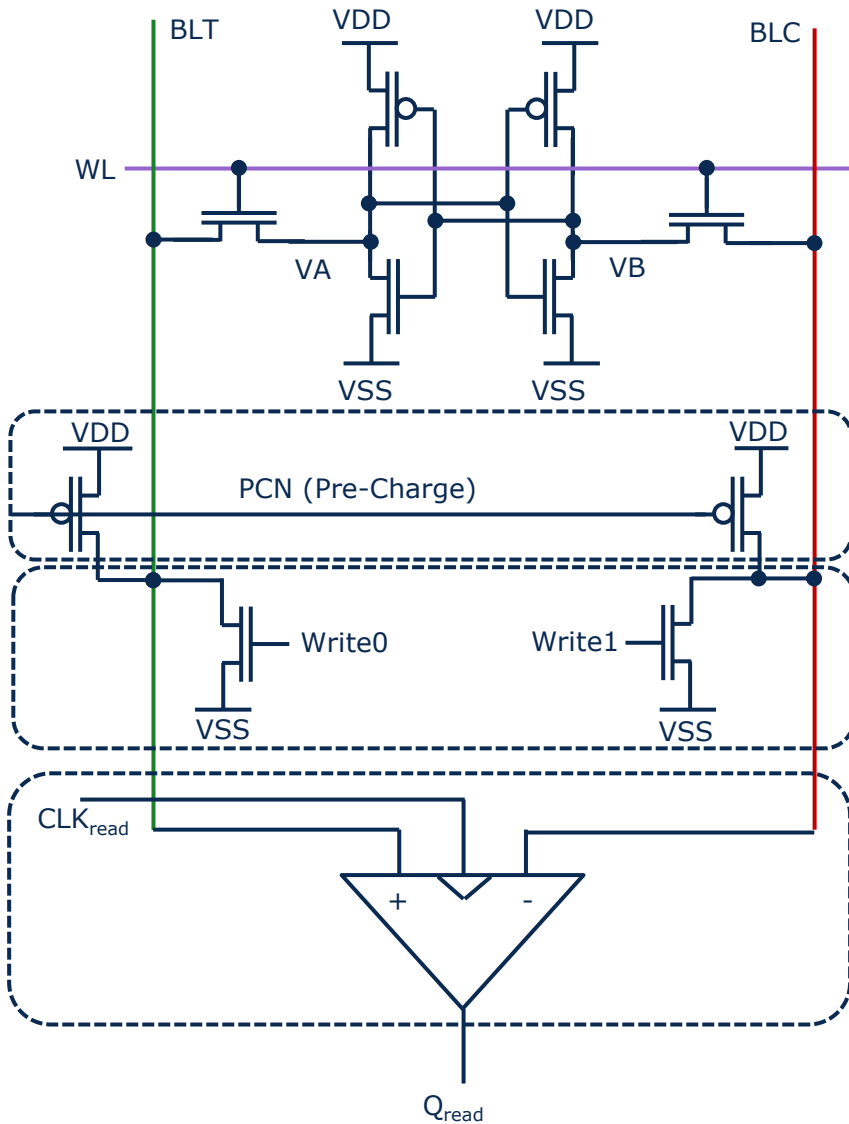
- Single Port 6-Transistor (6T) SRAM Bitzelle

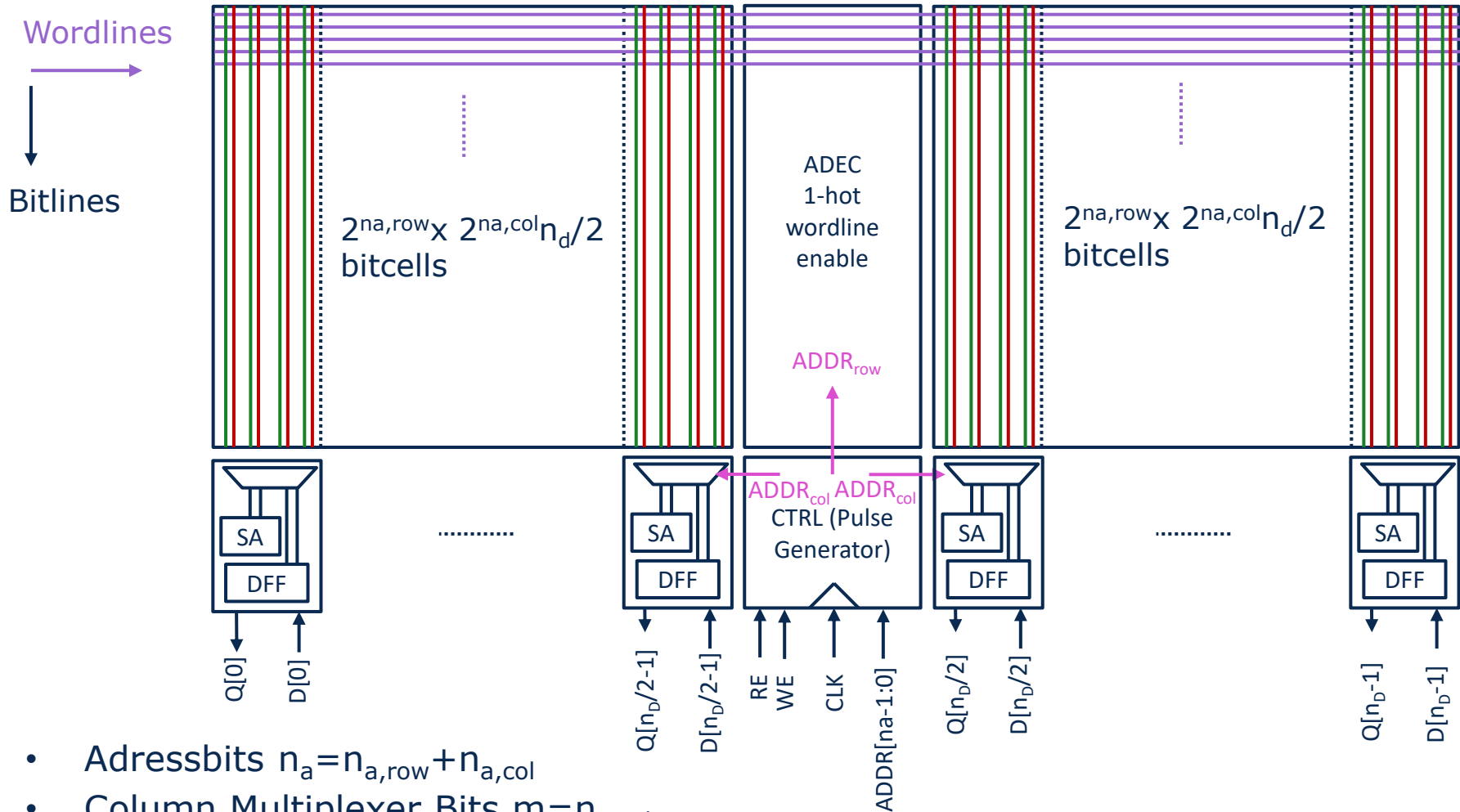


→ Details siehe Vorlesung
digitale Schaltungstechnik

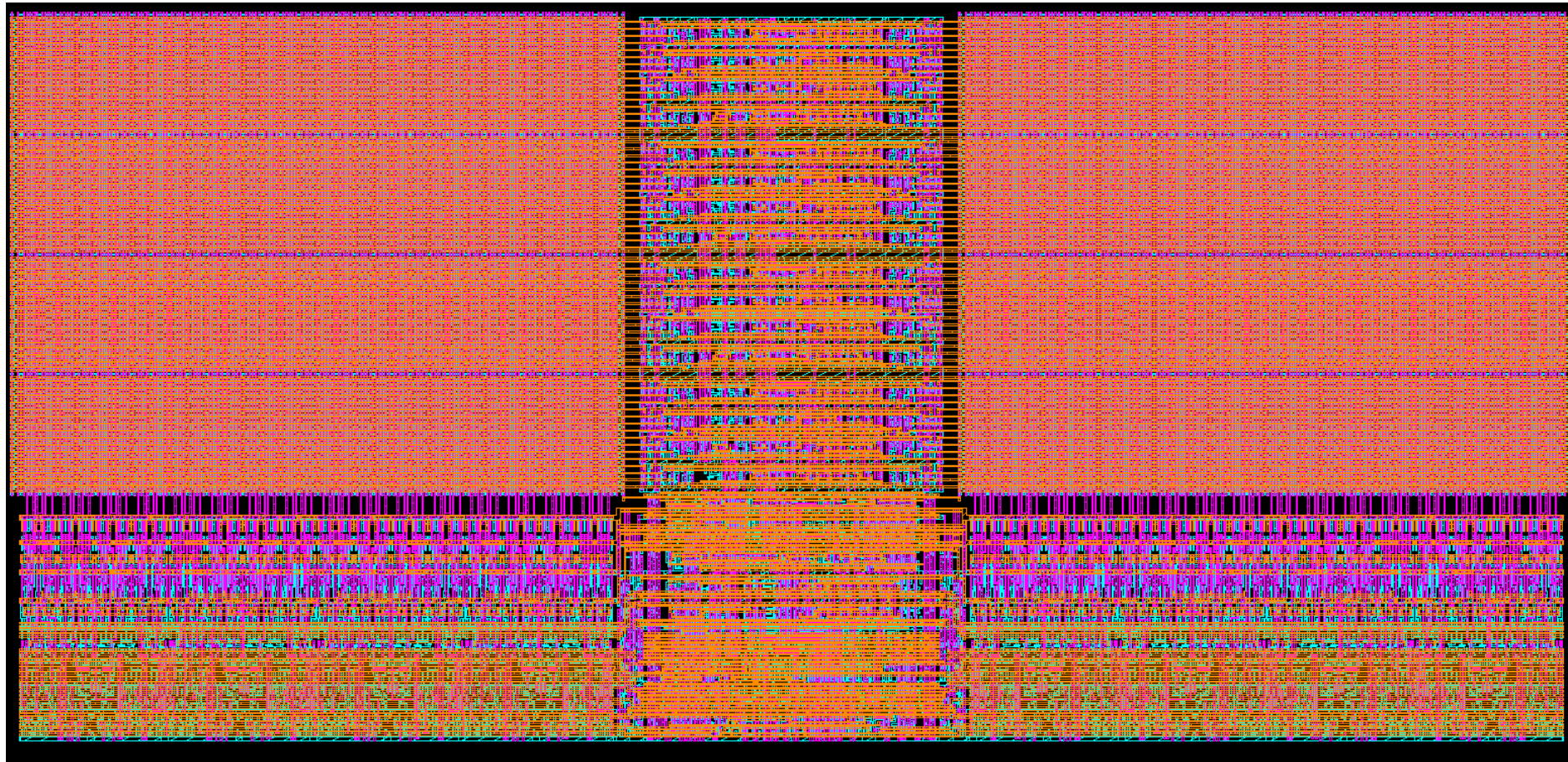


- Bitlines standardmäßig im Pre-Charge (V_{DD})
- Überschreiben der Speicherknoten durch die Pull-down getriebenen Bitlines
- Bitcell PMOS (klein!) übertrieben durch NMOS Access und Write Treiber
- Starker Pull-Down Write Treiber NMOS pro Spalte

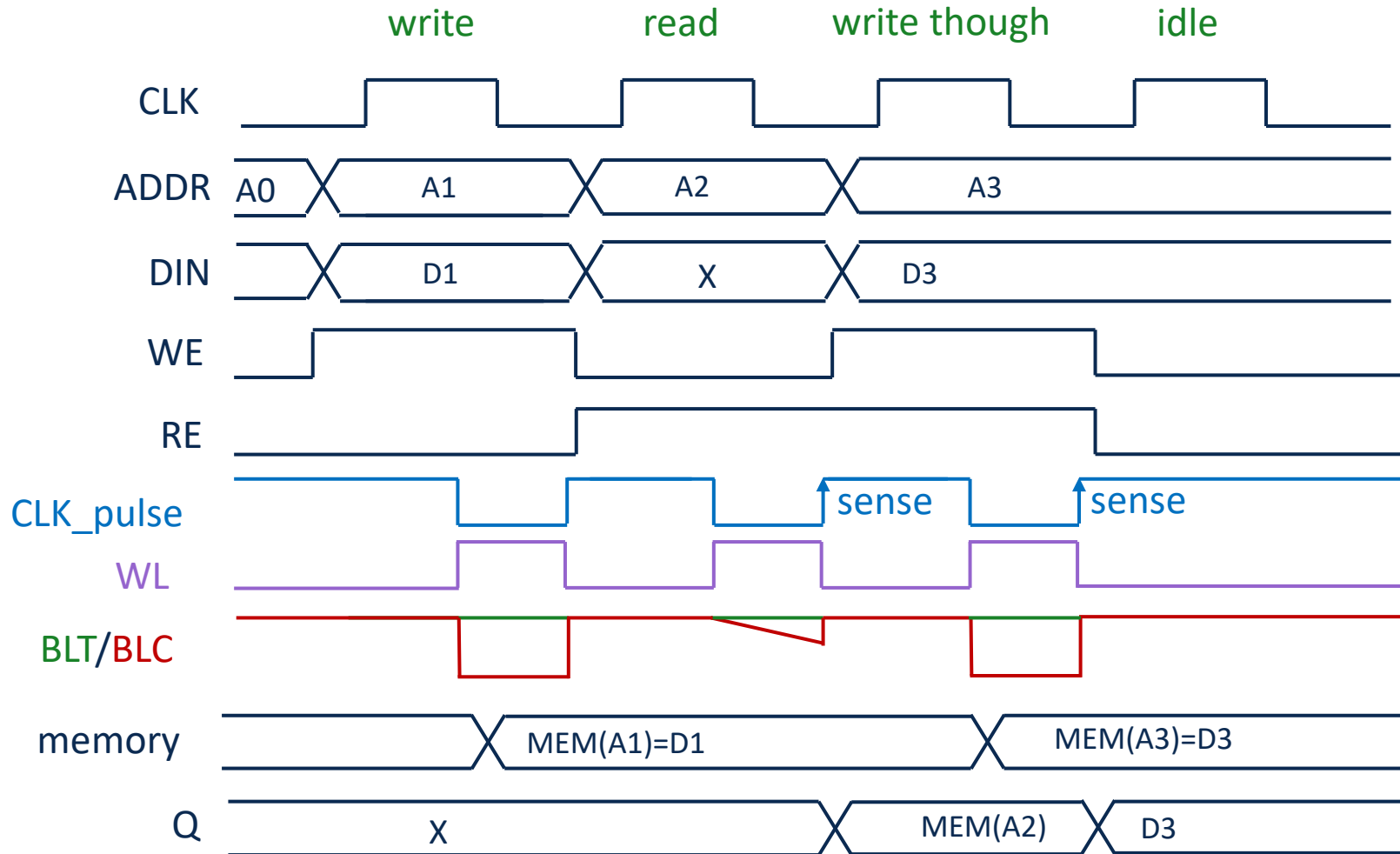




- Adressbits $n_a = n_{a,row} + n_{a,col}$
- Column Multiplexer Bits $m = n_{a,col}$
- Datenbits n_d



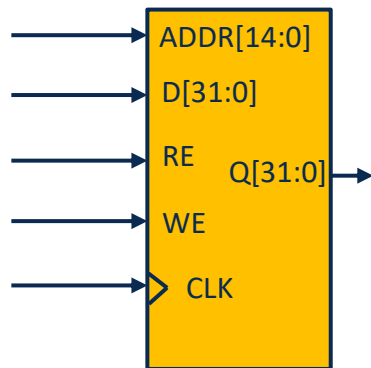
$$n_a=9, n_d=8, m=3$$



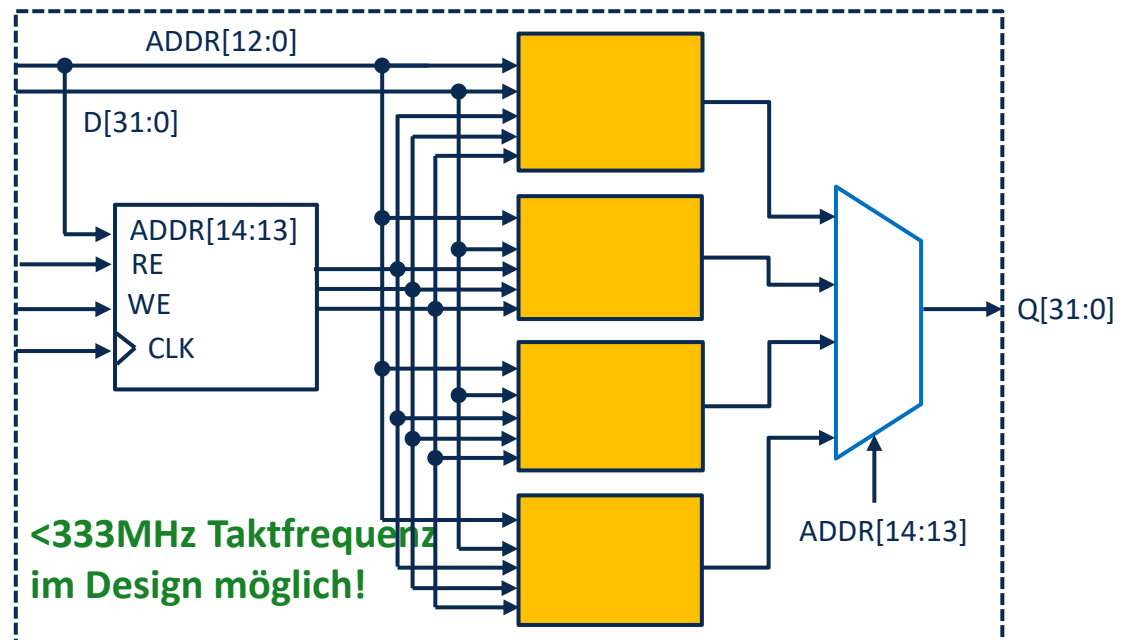
- Generieren von anwendungsspezifischen SRAM Makros
- Parameter: (n_a Adressbits, n_d Datenbits, m Column Mux Bits)
- Trade-off Fläche vs Speed (Höhe Bitlines) möglich
- Beispiel: $n_a=10$, $n_d=8$ → 1kByte Speicher



- Chips beinhalten typischerweise viele 100kByte SRAM → **Partitionierung**
- Beispiel: 128kByte Speicher mit 32 Bit Daten $n_a=15$, $n_d=32$
- Variante A
 - 32768x32 (128kByte Makro)
 - $M=5 \rightarrow 1024$ Wordlines
 - $\text{Clk} \rightarrow \text{Q} = 8\text{ns}$
- Variante B
 - 4x 8192x32 (32Kbyte Makros)
 - $M=5 \rightarrow 256$ Wordlines
 - $\text{Clk} \rightarrow \text{Q} = 2.5\text{ns}$, Mux Delay 0.5ns



<125MHz Taktfrequenz
im Design möglich!



<333MHz Taktfrequenz
im Design möglich!

