Emerging DRAM Technologies

Michael Thiems

amt051@email.mot.com

DigitalDNA Systems Architecture Laboratory

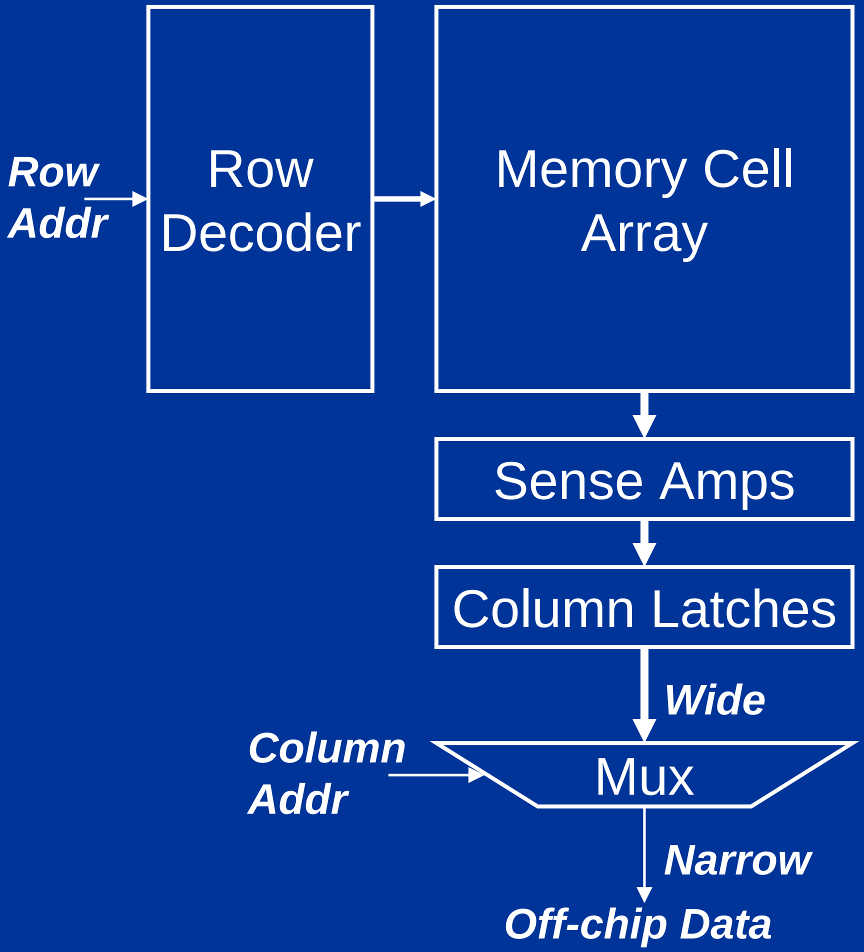
Motorola Labs

MOTOROLA

Motivation

- DRAM and the memory subsystem significantly impacts the performance and cost of a system
- Need to understand DRAM technologies
  - to architect an appropriate memory subsystem for an application
  - to utilize chosen DRAM efficiently
  - to design a memory controller

DRAM Basics



- Wide datapath on DRAM chip has potentially huge bandwidth
- Pin limitations restrict band-width available off-chip

MOTOROLA

Fill Frequency

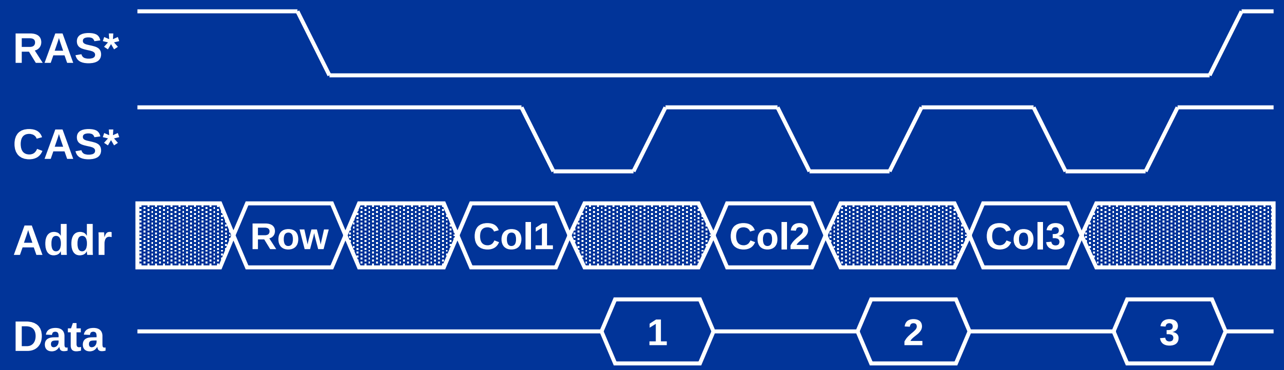
- Ratio of bandwidth to granularity
  - Minimum granularity impacts total system cost
  - Low-cost systems with large bandwidth needs (e.g. handheld consumer multimedia) have very high FF requirements
  - Increasing DRAM device sizes negatively impacts FF

Asynchronous DRAM

- No clock
- RAS* (Row Address Strobe)
- CAS* (Column Address Strobe)
- Multiplexed address lines
- System performance constrained:
  - by inter-signal timing requirements
  - by board-level inter-device skew

Asynchronous DRAM: Fast Page-Mode (FPM)

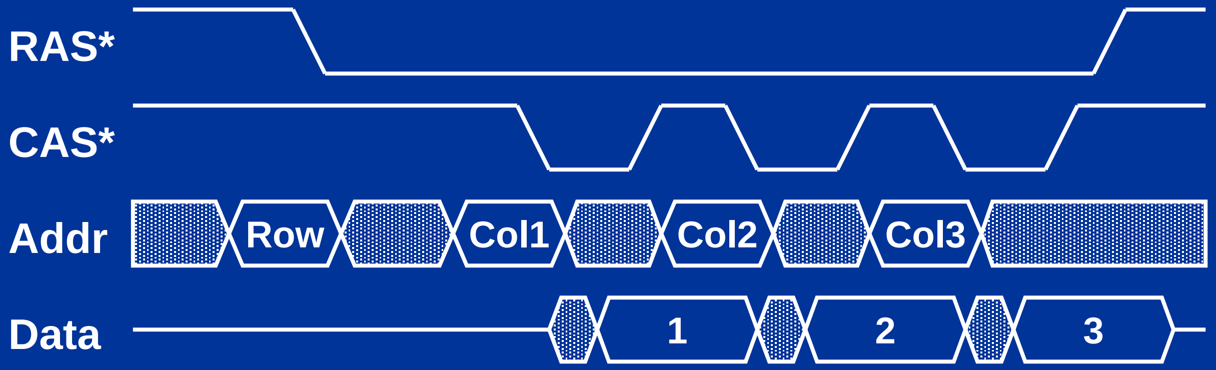
- Wide row transferred to column latches on-chip (RAS*)
- Columns in *same row* can be accessed more quickly (CAS*)



MOTOROLA

Asynchronous DRAM: Extended Data Out (EDO)

- Change in data output control reduces “dead time” between column accesses to same row
- Enables tighter timing, higher bandwidth



MOTOROLA

Synchronous DRAM (SDRAM)

- Clock signal distributed to all devices
- “Command” inputs (RAS*, CAS*, etc.) sampled on positive clock edge
- Higher performance synchronous operation
  - all inputs latched by clock
  - only need to consider set-up and hold time of signals relative to clock
  - on-chip column access path pipelined

MOTOROLA

SDRAM: On-Chip Multibanking

- Multiple, semi-independent banks on each SDRAM device
- SDRAM command scheme allows overlapped bank operations
  - one bank is activated and accessed
  - other bank is precharged
  - more efficient use of pin bandwidth

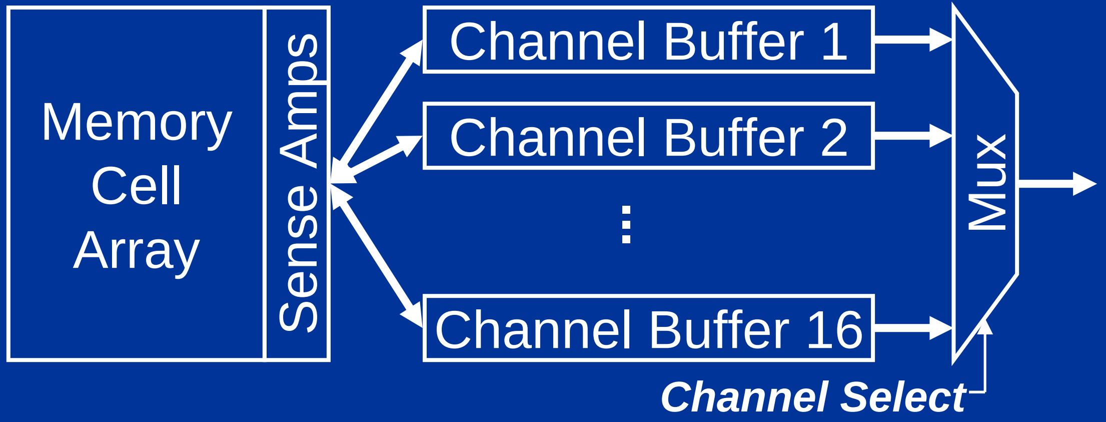
MOTOROLA

Double Data Rate (DDR) SDRAM

- Transfer data on both positive and negative clock edges
  - doubles peak pin data bandwidth
- Commands still sent only with positive clock edge
  - same pin command bandwidth
  - during random accesses, command bandwidth may limit usable data bandwidth

Virtual Channel (VC)

- Each device includes 16 channel buffers, each of which is very wide
  - each channel buffer can hold a large segment of *any* row (fully associative)
  - wide internal bus can transfer entire segment to/from memory array at once



MOTOROLA

Virtual Channel (cont.)

- Virtual Channels are associated with independent access streams
  - cache line refill, graphics, I/O DMA, etc.
  - reduces row penalties for multiple tasks interleaving accesses to different rows
- Data access is to channel buffers, allowing concurrent background operations
  - activate, precharge, prefetch, etc.

MOTOROLA

Virtual Channel (cont.)

- Maintains same command protocol and interface as SDRAM
  - command set is superset of SDRAM
  - extra commands encoded using unused address bits during column addressing
- Requires memory controller support
- Currently being implemented by NEC for single data rate SDRAM
  - techniques can also be applied to DDR, SLDRAM, etc.

Next-Generation Packet DRAMs

- Wrap extra logic and high-speed signaling circuitry around DDR-capable DRAM core
- Split-transaction “packet” protocols
  - sends command packets (including addresses) on control lines
  - sends or receives corresponding data packets some time later on data lines
  - synchronous with high-speed clock(s)

Packet DRAMs (cont.)

- Individual devices independently decode command packets and access DRAM core accordingly
- Other common features
  - in-system timing and signaling calibration and optimization
  - customized low-voltage, high-speed signaling
  - multiple internal banks
  - higher bandwidth with small granularity
- Direct RDRAM and SLDRAM

MOTOROLA

Direct Rambus DRAM (Direct RDRAM)

- Developed by Rambus Inc.
  - proprietary technology requires licensing and royalty payments
- 400 MHz differential transmit and receive clocks
- DDR signaling for both control and data, yielding 800 Mbits/sec/pin
- 16-bit data bus: peak data bandwidth of 1600 MBytes/sec from one device

Direct RDRAM (cont.)

- Control bus with *independent* row and column lines
  - 3 row lines and 5 column lines
  - control packets take 8 cycles (10 ns)
  - high control bandwidth and independent row and column control provide efficient bus utilization; claim 95% efficiency “over a wide range of workloads”
- 16 byte minimum data transfer size
  - 8 cycles (10 ns) on 16-bit data bus

MOTOROLA

Direct RDRAM (cont.)

- Rambus Signaling Logic (RSL)
  - 0.8 V signal swing
  - careful control of capacitance and inductance, transmission line issues
  - board design and layout constraints
- Memory subsystem architecture
  - 1.6 GBytes/sec for 1 to 32 devices on a single channel
  - multiply bandwidth by adding parallel channels

Synchronous-Link DRAM (SLDRAM)

- Developed by industry working group
  - IEEE and JEDEC open standard
  - claimed to be more evolutionary; fewer radical changes than Direct Rambus
- 200 MHz differential clocks
  - 1 command clock driven by controller
  - 2 data clocks driven by source of data
    - allows seamless change from one driving device to another

SLDRAM (cont.)

- DDR signaling for 400 Mbits/sec/pin, control and data
- 16-bit data bus: peak data bandwidth of 800 MBytes/sec from one device
- 10-bit control bus
  - no independent row / column control
  - control packets take 4 cycles (10 ns)
  - support for chip multicast addressing
- 8 byte minimum data transfer size
  - 4 cycles (10 ns) on 16-bit data bus

SLDRAM (cont.)

- Synchronous-Link I/O (SLIO)
  - 0.7 V signal swing
  - careful control of capacitance and inductance, transmission line issues
  - board design and layout constrain
  - less tightly constrained relative to RSL

MOTOROLA

SLDRAM (cont.)

- Memory subsystem architecture
  - 800 MBytes/sec for 1 to 8 devices on a single unbuffered channel
  - up to 256 devices on a hierarchical buffered channel
  - more bandwidth with parallel channels
    - parallel data channels can share single command channel

Embedded DRAM

- Fabricate DRAM on same die as processor / logic
- No pin limitations
  - can access full bandwidth of DRAM core using very wide on-chip busses
- System power savings
  - avoids off-chip I/O pin drivers/receivers and board-level traces
- Especially useful for ASICs
  - fixed, known DRAM requirements

MOTOROLA

Embedded DRAM (cont.)

- Requires process changes
  - DRAM and standard logic processes are not the same
  - processes integrating Embedded DRAM often lag newest process technology by one or more generations
- Larger die sizes with logic and DRAM
  - may negatively impact yields

References

- Y. Oshima et al., “High-speed Memory Architectures for Multimedia Applications,” IEEE Circuits & Devices, Jan 1997, pp. 8-13.
- S. Przybylski, “Full text excerpt of section 2.5 of *New DRAM Technologies, Second Edition*,” MicroDesign Resources Online, http://www.MDRonline.com/q/@16351162khsmcs/tl/dram/dram_excpt2.html
- A. Cosoroaba, “Double Data Rate Synchronous DRAMs in High Performance Applications,” Wescon/97 Conference Proceedings, IEEE, 1997, pp. 387-391.

References

- “MOS Integrated Circuit 64M-bit Virtual Channel SDRAM Data Sheet,” Document No. M13022EJ9V0DS00 (9th edititition), NEC, Japan, Oct 1998.
- D. Lammers, “Virtual Channel DRAM Gears Up to Duel Rambus,” EE Times Online, http://www.eet.com/news/98/1026news/virtual.html
- R. Crisp, “Direct Rambus Technology: The New Main Memory Standard,” IEEE Micro, Nov/Dec 1997, pp. 18-28.
- “64M/72M Direct RDRAM Data Sheet,” DL-0035-00.7, Rambus Inc., Aug 1998.

MOTOROLA

References

- “Direct Rambus Technology Disclosure,” DL-0040-00, Rambus Inc., Oct 1997.
- P. Gillingham and B. Vogley, “SLDRAM: High-Performance, Open-Standard Memory,” IEEE Micro, Nov/Dec 1997, pp. 29-39.
- “4Mx18 SLDRAM Data Sheet,” CORP400.P65, SLDRAM Inc., Feb 1998.
- D. Patterson and J. Hennessy, *Computer Organization and Design: The Hardware/ Software Interface*, Morgan Kaufmann, 1994, Appendix B.5.

MOTOROLA