# Memory

#### Introduction

- 1. This lesson covers the following:
  - Different types of memory
  - Why memory can't be large, fast, and cheap
  - How memory can store so many bits in a small physical space

### **How Memory Works**

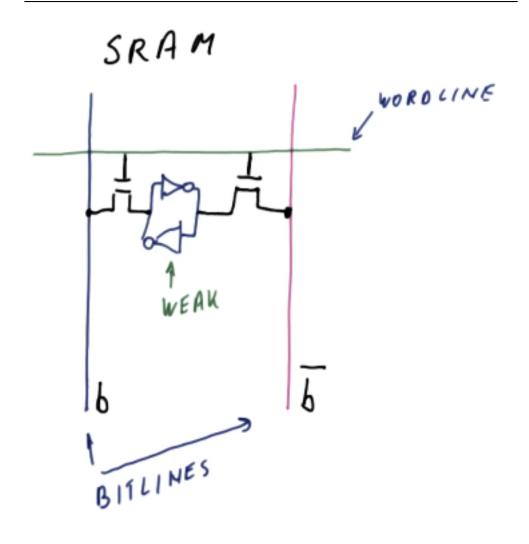
- 1. Memory technology: SRAM and DRAM
- 2. Why is memory slow?
- 3. Why don't we use cache-like memory?
- 4. What happens when we access main memory?
- 5. How do we make it faster?

### Memory Technology: SRAM and DRAM

- 1. SRAM: Static Random Access Memory
  - Static: SRAM retains data while power is supplied
  - Requires several transistors per bit
  - Typically faster
- 2. DRAM: Dynamic Random Access Memory
  - Dynamic: DRAM will lose data if we don't refresh it
  - Only requires one transistor per bit
  - Typically slower
- 3. Random access: Can access any memory location without going through all memory locations
  - As opposed to sequential access like a tape
- 4. SRAM and DRAM will lose data when power is not supplied

#### One Memory Bit SRAM

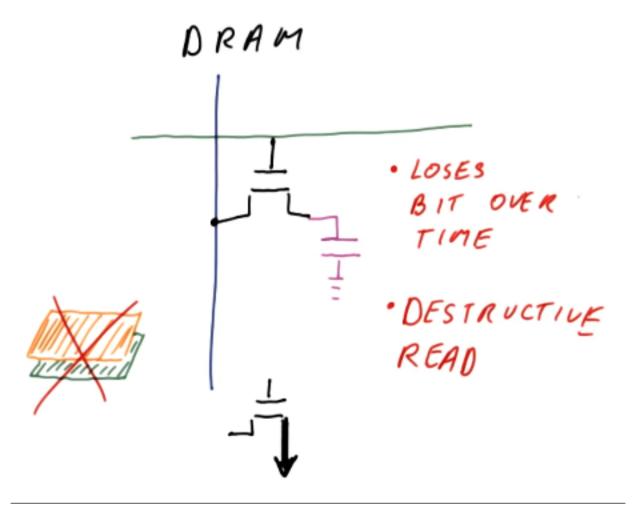
- 1. Memory works by manipulating a wordline and bitline to read/write to an array of transistors
- 2. In SRAM, each cell has a double inverter (one inverter requires two transistors)
  - This feedback loop is what makes the memory static
  - The voltage output by the transistor is greater than the voltage output by the inverter loop, so they are easy to overwrite
  - Typically connect the inverters to two bit lines with a transistor in between each
    - Allows us to more easily detect changes because we can observe the difference in voltage between the two bitlines
- 3. SRAM is a 6T cell (6 transistors)



One Bit Memory SRAM

# One Memory Bit DRAM

- 1. Implemented with a single capacitor
  - When we activate the bitline, the capacitor is charged with a 1
  - When we deactivate the bitline, the capacitor is discharged to a 0
- 2. Transistor is not a perfect switch, it's a little bit leaky
  - In SRAM, the inverter loop keeps the charge
  - In DRAM, the capacitor loses charge, so we need to periodically refresh it
- 3. Destructive read: Opening the wordline to read the value causes the capacitor to lose some charge
  - When we perform a read, we also need to recharge the capacitor
- 4. DRAM is a 1T cell (1 transistor)
- 5. Area of a DRAM cell is area of transistor plus area of capacitor
  - Bigger capacitors allow for the charge to be maintained longer
  - Capacitor is buried beneath the silicon, so the effective area is the area of the transistor ("trench cell")
  - DRAM also only requires one bitline



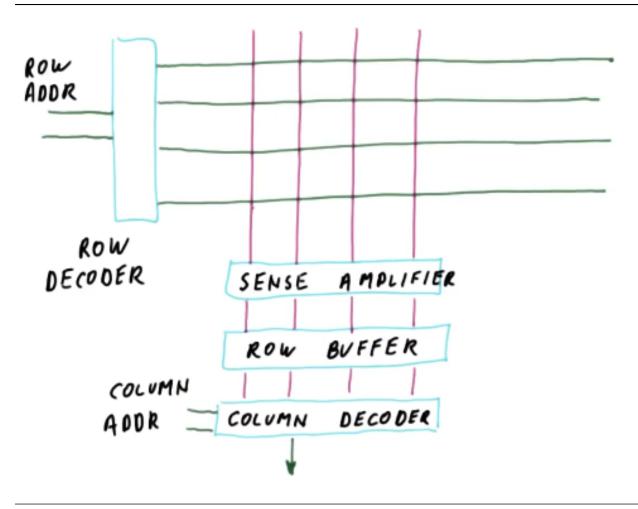
One Bit Memory DRAM

#### DRAM Technology Quiz

- 1. What not use a "normal" transistor and a capacitor when making DRAM?
  - Trench cell is easier to make (false)
  - Trench cell is more reliable (false)
  - Trench cell lets us make cheaper DRAM chips (true)

### Memory Chip Organization Part 1

- 1. Row decoder takes a row address and determines which wordline to read
  - Detects multiple cells to the bitline
- 2. Bitlines are connected to a sense amplifier
  - Helps the cell raise or lower the voltage on a bitline
  - Only need one at end of bitline, not one per cell
- 3. Output of sense amplifier goes to a row buffer
- 4. Row buffer goes to column decoder
  - Column address is input to the column decoder to pick one bit from the row
- 5. If we want to be able to read more than one bit at a time, we replicate this structure



Memory Organization

#### Memory Chip Organization Part 2

- 1. After the bit is read, the sense amplifier is reversed to write the values back
  - This is due to the fact that reads to DRAM are destructive
- 2. DRAM is slower for two reasons
  - Destructive read requires writing the value back (read-then-write)
  - Cell does not pull the bitline as strongly, so the sense amplifier needs more time to determine the correct value
- 3. Refresh: Make sure each row is read at some regular interval
  - Refresh row counter: Iterates through the rows reading-then-writing each
  - Refresh period T (typically less than a second)
  - N rows (modern memory has large N)

#### Memory Refresh Quiz

- 1. Consider memory with the following parameters: \*4096 rows, 2048 columns
  - Refresh period is 500 microseconds
- 2. Read timing:
  - 4 ns to select a row
  - 10 ns for sense amplifier to get bit values

- 2 ns to put data in row buffer
- 4 ns for column decoder
- 11 ns to write data from sense amplifier to memory row
  - This is overlapped with putting data in row buffer and decoding
- 3. How many reads per second can this memory support?
  - Read takes  $4 + 10 + \max(2 + 6, 11) = 25 \text{ ns} \rightarrow 40 \text{ M}$  refreshes per second
  - Refresh period is 500 microseconds, so we need to do 2000 refreshes per second
  - Refreshes per second = 2000 \* 4096 = 8.192 M refreshes per second
  - Total reads = 40M 8.192M = 31808000

#### Memory Chip Organization Part 3

- 1. How do we write to memory?
  - Use the row address to select a row
  - Read all of the bits in the row and latch into the row buffer
  - Use the column decoder to select the correct bitline
  - Write the value to the row buffer
  - When the sense amplifier writes the data back, the memory will store the correct value
- 2. Write is a "read-then-write" operation

#### Fast Page Mode

- 1. Once a word is read into the row buffer, we can read data directly from the row buffer instead of re-opening the page
- 2. Opening a page:
  - Row address
  - Select row
  - Sense amplifier
  - Latch into row buffer
- 3. Read/write
  - Reading and writing only modifies the row buffer
  - Can do this many times without having to close/reopen the page
- 4. Closing the page:
  - Write data from row buffer back to memory row

## DRAM Access Scheduling Quiz

- 1. DRAM has 32 1-bit arrays
- 2. Each array is 16 megabit array (2 ^ 12 rows \* 2 ^ 12 bits per row)
- 3. Assume the following:
  - Page open: 10 ns
  - Read from row buffer: 2 ns
  - Page close: 5 ns
- 4. Consider the following set of memory accesses:
  - F00F00
  - E00F00
  - F00E04
  - E04F00
  - E00E00
  - F00123
  - 123F00
  - The upper three hex digits are the row address, the lower three are the column address
- 5. How long do these memory accesses take in the current order?
  - F00F00 (10 + 2 + 5 = 17)

- E00F00 (10 + 2 + 5 = 17)
- F00E04 (10 + 2 + 5 = 17)
- E04F00 (10 + 2 + 5 = 17)
- E00E00 (10 + 2 + 5 = 17)
- F00123 (10 + 2 + 5 = 17)
- 123F00 (10 + 2 + 5 = 17)
- 17 \* 7 = 119 ns
- 6. How long do these memory accesses take in the optimal reordering?
  - F00F00 (10 + 2 = 12)
  - F00E04 (2)
  - F00123 (2 + 5 = 7)
  - E00E00 (10 + 2 = 12)
  - E00F00 (2 + 5 = 7)
  - E04F00 (10 + 2 + 5 = 17)
  - 123F00 (10 + 2 + 5 = 17)
  - Total = 12 + 2 + 7 + 12 + 7 + 17 + 17 = 74

### Connecting DRAM to the Processor

- 1. Want to be able to connect multiple memory chips to the same processor
  - Add a memory controller between the front-side bus and DRAM that connects the LLC to the memory
  - Memory channel connects memory controller to DRAM
- 2. Total memory latency includes the following:
  - Sending request to memory controller, its
  - Memory controller logic
  - Sending the data over the memory channel
  - Memory latency
  - Sending the data back over the memory channel
  - Sending the data back to the processor from the memory controller
- 3. Recent processor integrate the memory controller into the same chip as the processor and caches
  - Eliminates the need for the front-side bus to reduce latency (10-30%)
  - This means the processor is designed to work with a specific type of DRAM
    - Required standardizing memory construction

#### Conclusion

- 1. Examined how DRAM and SRAM works
- 2. Will examine disk drives which have a lower cost per bit but are much slower