ECE 571 – Advanced Microprocessor-Based Design Lecture 32

Vince Weaver

https://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

2 December 2024

Announcements

- HW#11 will be posted, GPU reading. Due next Monday.
- Will post preliminary project schedule
- Don't forget to do the course evaluation
- Bring some old disk technology to lecture

Re-writable Solid-state Storage

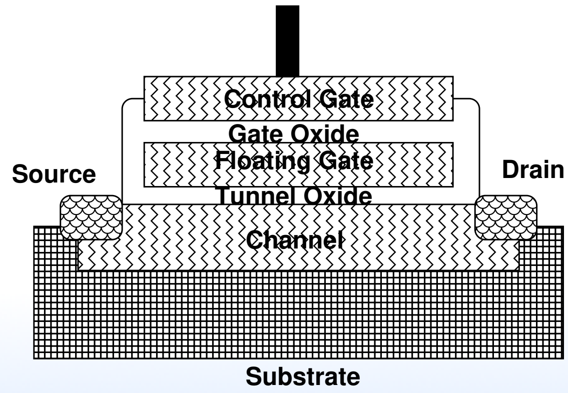
- Using transistors, no moving parts
- Old tech:
  - EPROM – erasable programmable read-only memory (erase with UV light)
  - EEPROM – electrically erasable PROM

Flash

- Invented at Toshiba in the 1980s
- NAND vs NOR
- Can erase, but in relatively large chunks
- Reading is fast (approaching RAM speeds)
- Writing is relatively slow (but still faster than magnetic hard drive)

Floating-gate transistors

- Each cell like a MOSFET, but with two gates
- Floating gate and control gate. Control to switch, float can trap electrons
- Floating gate raises the Vt by acting as a screen, so detect if 1 or 0 by putting an intermediate voltage



Charge Pumps

- Need high voltage to write. But usually this is done from single voltage supply
- 10-13V
- In space applications usually the charge pump that fails (so chips can still be read, but no longer write)

Writing Flash

- Starts as all 1s
- In general can change any 1 to 0 at any time, but if want to switch from 0 back to 1 have to erase whole block

Trapping Electrons in Gate

- Two common ways
  - Fowler-Nordheim Tunneling
    Strong electric field between negative source and positive gate, draw electrons into floating gate, trapped between insulators
  - Hot Electron Injection
    High current in channel, electrons boil up into floating gate. Positive charge on gate attracts them
- Resetting (actually to all 1 state). Same process,

opposite direction, large voltage. Can in theory be individually reset bits but in practice in blocks

Removing Electrons From Gate

- Fowler-Nordheim Tunneling
- Strong negative charge on control gate toward positive charge on drain.

NOR Flash

- Faster, more expensive, slower write/erase, but can individually address bytes and can directly execute code
- Long erase
- Random access
- 100x - 1000000x erase cycles
- Compact Flash (CF) was originally NOR (but NAND cheaper)
- Like a NOR gate, one end to ground
- low read latency, can be used bit-by-bit ROM

NAND Flash

- Bits are a bunch of nand gates connected serially, to read out have to read out whole row (sort of like a shift register). OK if streaming but tough if just want single bytes
- Read whole pages (4k-16k) but erase areas much larger?
- Reduced erase/write times
- Less chip area (higher density)
- 10x endurance of NOR
- Must read out in large blocks (not random access)

- Wired in series like a bunch of NAND gates
- Certain amount of errors allowed (unlike NOR)
- Tunnel injection for erasing
- ECC error correction

Bits-per-Transistor

- Vertical (3D) NAND, cells stacked vertically for more density
- SLC=single level (bit) per core
- MLC=multiple(2)
- TLC=triple level
- QLC=quad level cell
- In all cases the transistor is the same, just move values written. Harder to read/write because instead of a clear on/off amount of trapped electrons, it's in the middle.

Also wears out faster as error rate is higher making harder to distinguish as degrades.

Flash Issues

- Memory wear – can only write so many times before wears out. 100k?
- Memory disturb – a bit like rowhammer, write too many times can change nearby
- Xray can reset bits (problem when trying to see if BGA solder went well)
- Data retention – trapped electrons steadily leak away, especially at warm temperatures

SSD

- Solid-state disk
- No moving parts
- Faster, lower-latency, more resistant to shock
- Still more expensive
- SSDs not permanent, will gradually leak and lose data after 2-3 years (faster if worn? trapped electrons leak away)
- Originally was DRAM (battery backed) but these days NAND flash

Controller manages storage

- Bad-block remapping
- Read scrubbing
- Wear leveling

SSD Performance

- DRAM-based fastest
- Single NAND relatively slow
- Having lots in parallel helps

SSD Form Factor

- Can be SATA, but SATA while fast enough for magnetic disk cannot keep up with flash
- M.2 (formerly NGFF) – Intel
  Can provide PCIe, SATA 3.0, or USB 3, different keying to keep from plugging in wrong
- NVMe – non volatile memory express – hook up via PCIe
- U.2 (formerly) SSD Form Factor Working Group, provides SSD connector for enterprise. Hot-swap. mechanically identical to SATA-express. 3.3V or 12V

(M.2 only 3.3V)

Operating System support – Trim Operation

- On filesystem, erase file, usually just mark as deleted and blocks unused, even though never want the bytes again
- TRIM on flash lets you tell disk you don't want them anymore, and the drive can then reclaim them
- Also, when OS then re-uses freed block, flash sees this as an over-write of the block (expensive) rather than a fresh write to a new block

- Expensive because typically erases in big chunks (512kB) so over-writing you have to erase a whole big chunk, then do a write back of existing values

Hybrid Drive

- Can have Flash act as cache for traditional HDD
- Most HDDs also have DRAM cache for speed
- What happens if lose power when dirty data in DRAM cache?
- OS can send "flush" command to flush cache for safety. In past some manufacturers ignore this. Why? Looks better on benchmarks. What can go wrong?

SSD vs HDD comparison

- Data Durability – SSD loses in a few years – HDD lasts longer, but motors/mechanical might fail
- Startup – HDD has to spin up, takes a while
- Random Access – HDD bad, has to spin to location
- Read latency – SSD better
- Bandwidth – SSD often higher
- Read perf – SSD fast but goes down with use
- Noise – SSDs silent
- Heating – both don't like high temps

- Cooling – SSDs can operate at lower temps
- Air – SSDs don't require air
- Price – SSD cheaper
- Power –SSD usually better
- Storage size – HDD usually better

SSD Quality, DRAMless

- https://utcc.utoronto.ca/~cks/space/blog/tech/SSDsUnderstandingDramI
- SSDs for servers, people looking to use NVME instead (flash on the board)
- Use cheaper QLC flash
- All SSDs need to track which blocks are bad, wear leveling, etc
- Have embedded system that does this. Needs some sort of RAM to store this data
- Expensive to have on-board DRAM, can it instead off-

load this to the OS and use your system RAM?

- Host-Memory Buffer `https://phisonblog.com/host-memory-buffer-2/`
- Cost reduction, remove the DRAM for space and price
- Performance problem, without fast cache of LBA (address) to block mappings have to constantly pull it off of slow flash

Compact Flash

SD-card

PCIe – history

- Older busses on IBM PC
- ISA Bus, 4.7MHz 8-bit
- 16-bit
- 32-bit a mess. VLB? EISA? IBM tried to re-take the market with PS/2+microchannel
- Intel came out with PCI, 32-bit at 33MHz, also PCI-X 64-bit (parallel bus)
- We've discussed parallel busses problematic as speeds get faster

Timing skew, layout

PCIe (PCI-express)

- Serial bus
- Sends packets (almost like network)
- Can power 25W, additional power connectors to supply can have 75W, 150W and more

- Can transfer 8GT/s (giga-transfers) a second
- PCIe x4 abous same bandwidth as PCIx (64 bits at 133MHz) 1064M/s

PCIe Lanes

- Each lane 4 wires, differential pair each direction
- Can have 1 ... 16 lanes (x1, x2, x4, x8, x16)
- In theory card negotiates this at startup, can run an x16 card in an x1 slot it will just run 16 times slower

PCIe Versions

1	2003	NRZ	8b/10b	2.5GT/s
1	2007	NRZ	8b/10b	5.0GT/s
1	2010	NRZ	128b/130b	8.0GT/s
1	2017	NRZ	128b/130b	16.0GT/s
1	2019	NRZ	128b/130b	32.0GT/s
6	?	?	?	?
7	?	?	?	?