

Metric Prefixes

peta	P	10^{15}	1 000 000 000 000 000
tera	T	10^{12}	1 000 000 000 000
giga	G	10^9	1 000 000 000
mega	M	10^6	1 000 000
kilo	k	10^3	1 000
hecto	h	10^2	100
deca	da	10^1	10
one		10^0	1
deci	d	10^{-1}	0.1
centi	c	10^{-2}	0.01
milli	m	10^{-3}	0.001
micro	μ	10^{-6}	0.000 001
nano	n	10^{-9}	0.000 000 001
pico	p	10^{-12}	0.000 000 000 001
femto	f	10^{-15}	0.000 000 000 000 001

De Morgan's Laws

- $\overline{AB} = \overline{A} + \overline{B}$
- $\overline{A + B} = (\overline{A})(\overline{B})$

Silicon

- Si
- P-type:
 - * doped with material to remove electrons (add electron holes), usually Boron (B), Aluminum (Al), or Gallium (Ga)
- N-type:
 - * doped with material to add electrons, usually Antimony (Sb), Arsenic (As), or Phosphorous (P)
- Silicon dioxide: SiO_2
- Polysilicon is just silicon without the crystal structure

Transistors

- nMOS
 - * no bubble
 - * on when input is on, off when input is off
 - * base (of whole chip) is p-substrate
 - * spot of n+ for source and drain, joined by a small layer of oxide (SiO_2) and polysilicon (gate). also has a spot of p+ (base) connected to ground
- pMOS:
 - * has the bubble
 - * on when input is 0, off when input is 1
 - * whole thing sits in an n-well (inside the p-substrate base)
 - * source and drain are spots of p+ connected by gate. Gate is polysilicon layer separated from rest of chip by thin layer of SiO_2 on bottom. Also has a n+ spot for base (connected to VDD)
- CMOS: when you combine a nMOS and pMOS network together to make a gate, where one is the compliment of the other
- V_t : Threshold voltage. Nominal voltage below which the transistor is off
 - * below as in closer to 0, not less
 - * $V_t > 0$ for nMOS, $V_t < 0$ for pMOS
 - * this is compared to the gate to source voltage, V_{gs}
- regions: (for nMOS)
 - * accumulation: gate is negatively charged, attracts positive voids in p-substrate, which block flow in the channel
 - * depletion: small positive charge on gate repels positive voids from channel, forming a depletion below the gate
 - * inversion: higher positive charge ($> V_t$) is applied to gate, attracting electrons to the channel and allowing

flow

D Flip Flop vs Latch

- latch is level triggered
- flip flop is edge triggered

Fabrication

- n-well: use diffusion or ion implantation
- positive lithography: expose to UV where you want to remove material
- negative lithography: expose to UV where you want to keep material

Stick Diagram vs Boolean Function

- TODO
- there is more than one way of making a stick diagram for an expression
 - * stuff in series could be in different order, for instance
- if you do it manually, you need to check it with tools:
 - * LVS: Layout Vs Schematic
 - * DRC: Design Re-Check
 - * These aren't really needed for designs automatically generated from verilog code, because of course that's correct

Lithography

- the process of printing onto a chip at nanometer scale
- generally uses UV light, wavelength around 150nm
 - * must use fancy tricks to make 10nm features with 150nm light
 - * would be nice to use even lower wavelength X-rays, but those are hard to focus
- negative lithography: use the lithography mask to cover what you want to keep.
- positive lithography: mask what you want to remove
- a lens is used to focus the light
 - * ideally, want a point source for the light, but that is not practical
 - * Optical Proximity Effect: what happens when your focus from the lens is not just right
 - * results in rounded corners, inaccurate critical dimensions, and shorter wire ends
 - * can use Optical Proximity Correction to fix: basically over-emphasize all the features, and/or add extra lines at outset

MOS transitive I - V Characteristics and Parasitics

- I - V : current-voltage relationship
- Transistors are not really ideal switches, they have 3 zones of operation: cutoff, linear, saturation
- definitions:
 - V_{gs} : voltage gate to source
 - V_{gd} : voltage gate to drain
 - V_{ds} : voltage source to drain (across the channel)
 - V_t : critical voltage at which transistor is saturated
 - channel: space between the source and drain, where the electrons flow
- remember that the gate is insulated from the area under it by a thin layer of Silicon Dioxide (SiO_2)
- by convention, the source is the terminal at lower voltage
- **cutoff**:
 - * when $V_{gs} < 0$
 - * electrons on the gate attract positive voids in the silicon below, and inhibit current flow. Therefore, the transistor is closed.
 - * $I_{ds} = 0$
- **linear**:
 - * $V_{gs} > V_t, V_{gd} = V_{gs}, V_{ds} = 0$ or $V_{gs} > V_t, V_{gs} > V_{gd} > V_t, 0 < V_{ds} < V_{gs} - V_t$
 - * I_{ds} linearly proportional to V_{ds}
 - * channel of electrons forms, allowing current to flow

- **saturated:**

- * $V_{gs} > V_t, V_{gd} < V_t, V_{ds} > V_{gs} - V_t$
- * channel pinches off due to electrons attracting to source
- * I_{ds} is independent of V_{ds}

- **capacitor effect**

- gate and channel can have a parallel plate capacitor effect, with the thin layer of SiO_2 acting as the insulator
- $C = \frac{Q}{V} = \epsilon_{\text{SiO}_2} w l / t_{\text{SiO}_2}, V = V + g_c - V_t = (v_{gs} - V_{ds}/2) - V_t$
 - * l, w : length, width of section of gate above channel
 - * ϵ_{SiO_2} : permittivity of SiO_2 layer
 - * t_{SiO_2} : thickness of SiO_2 layer
- general capacitance per unit area: $C_{ox} = \frac{\epsilon_{ox}}{t_{ox}}$
 - * ϵ_{ox} : permittivity of oxidation layer
 - * t_{ox} : thickness of oxidation layer
- carrier velocity: velocity of the electrons?
 - * proportional to the electric field running horizontally between source and drain
 - * $v = \mu E, E = V_{ds}/L, t = L/v = L/(\mu E) = L/(\mu \frac{V_{ds}}{L})$
 - μ : mobility. electrons move about twice as fast as positive voids
 - L : length of channel
- actual velocity of electrons is the speed of light, but they don't travel in a straight line, they travel atom-to-atom
 - * this slowdown is called the **scattering** effect

- **Shockley model of transistor**

- $V_{dsat} = V_{gs} - V_t$
- 1st order model: $\beta = \mu C_{\text{SiO}_2} \frac{w}{l}$

$V_{gs} < V_t$	$I_{ds} = 0$	cutoff
$V_{ds} < V_{dsat}$	$I_{ds} = \beta(V_{gs} - V_t - \frac{V_{ds}}{2})V_{ds}$	linear
$V_{ds} > V_{dsat}$	$I_{ds} = \frac{\beta}{2}(V_{gs} - V_t)^2$	saturation
- Must be able to derive this model on exam!

- **Non-Ideal I-V Effects**

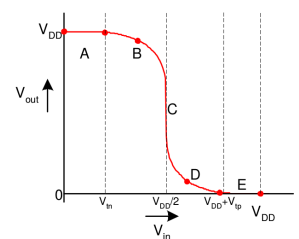
- velocity saturation (due to scattering)
 - * also called short channel effect
 - * this is hard to make into a mathematical formula
- sub-threshold leakage, junction leakage, gate tunneling
- **Body Effect**
 - * affected by V_{sb} : voltage of the p-substrate (which should be ground)
 - * $V_t = V_{t0} + \lambda(\sqrt{|-2\phi_F + V_{sb}|} - \sqrt{|-2\phi_F|})$
 - V_{t0} : threshold without body bias
 - ϕ_F : Fermi potential
 - negative for nMOS, positive for pMOS
 - λ : body effect coefficient
 - positive for nMOS, negative for pMOS (reversed)
 - * generally fixed/caused by biasing
 - * Forward Body Bias (FBB):
 - $V_{sb} < 0, V_t < V_{s0}$
 - gates switch faster, but leak more current
 - * Reverse Body Bias (RBB):
 - $V_{sb} > 0, V_t > V_{s0}$
 - gates switch slower, but consume less power (because less leakage)
- **Temperature**
 - * higher temperature means higher electron mobility, more leakage, and threshold decreases
- **Diffusion Capacitance**
 - * capacitance on the source/drain: C_{sb}, C_{db}
 - source/drain are diffusion nodes
 - C is comparable to C_g (gate) for connected nodes, $\frac{1}{2}C_g$ for unconnected
 - (depends on process)

- * this is the capacitance we care most about (because we must fill it every time the gate switches?)
- * if two capacitors share a source/drain, that reduces diffusion capacitance (reducing this is a good thing)
- * also, you can remove unconnected diffusion spots

- **DC Response**

- example: inverter
- must settle to $I_{dsn} = |I_{dsp}|$
- V_{tn}, V_{tp} : threshold voltages for nMOS and pMOS (defined in Non-Ideal I-V Effects)
- $V_{in} = V_{DD} \rightarrow V_{out} = 0, V_{in} = 0 \rightarrow V_{out} = V_{DD}$
- nMOS
 - * $V_{gsn} = V_{in}, V_{dsn} = V_{out}$
 - * cutoff: $V_{in} < V_{tn}$
 - * linear: $V_{in} > V_{tn}, V_{out} < V_{in} - V_{tn}$
 - * saturated: $V_{in} > V_{tn}, V_{out} > V_{in} - V_{tn}$
- pMOS
 - * $V_{gsp} = V_{in} - V_{DD}, V_{dsp} = V_{out} - V_{DD}, T_{tp} < 0$
 - * cutoff: $V_{in} > V_{DD} + V_{tp}$
 - * linear: $V_{in} < V_{DD} + V_{tp}, V_{out} > V_{in} - V_{tp}$
 - * saturated: $V_{in} < V_{DD} + V_{tp}, V_{out} < V_{in} - V_{tp}$
- to calculate actual output voltage, balance $I_{dsn} = I_{dsp}$ (easiest to do graphically)
- * end up with a graph of V_{out} as function of V_{in}

Region	nMOS	pMOS
A	Cutoff	Linear
B	Saturation	Linear
C	Saturation	Saturation
D	Linear	Saturation
E	Linear	Cutoff



- * don't want to switch nMOS and pMOS because then nMOS would saturate at far end of graph (and vice versa for pMOS)
- * horizontal position of graph can be varied by tuning β_p/β_n
 - called beta ratio, or skewed gate
 - $\beta_p/\beta_n > 1 \rightarrow$ right, $\beta_p/\beta_n \rightarrow$ left
- * unity gain slope: part of response graph where the slope is -1
 - want to tune β_p/β_n to put logic levels at these regions to maximize noise margins
- * Noise Margins:
 - $NM_H = |V_{OH} - V_{IH}|, NM_L = |V_{OL} - V_{IL}|$
 - that's just the higher/lower of the two axes

- **Transient Analysis**

- for instance, find step response of gate to determine rise time.
- rise/fall delay: time from when V_{in} crosses $\frac{V_{DD}}{2}$ to when V_{out} crosses it
- rise/fall time: (of V_{in} or V_{out}): time for that signal to go from $0.1V_{DD}$ to $0.9V_{DD}$ (or reverse)
- TODO many equations and such for inverter step response (from slides)
- TODO pass transistors (from slides)

- **Pass Transistors**

- for nMOS trying to pass V_{DD} or pMOS trying to pass 0
- nMOS can pull no higher than $V_{DD} - V_{tn}$ if $V_g = V_{DD}$
 - * more generally, $V_g - V_{tn}$
 - * called degraded 1
- pMOS can pull no lower than $|V_{tp}|$

- **Delay**

- generally estimated with RC models
 - * for nMOS with width k :
 - * resistance of R/k

- * caps of kC on all terminals
- * pMOS same except resistance is $2R/k$
- depends on effective R and C of transistors
 - * exactly what parasitic caps depends on exact layout (which stuff is shared between transistors)
- width:
 - * C proportional to width (approx $2 \text{ fF}/\mu\text{m}$)
 - * R inversely proportional to width (approx $6\text{k}\Omega \cdot \mu\text{m}$)
 - * TODO unity transistors
- find widths necessary for rise and fall resistance to be same as standard inverter
 - * pMOS is about half as conductive as nMOS, so the inverter has nMOS=1, pMOS=2
 - * larger width \rightarrow smaller R
 - * transistors in series: delay adds, so double the width
 - * transistors in parallel: same as a single transistor (because we assume worst case of only one being active)
- use effective resistance in RC model: $I_{ds} = V_{ds}/R$ (just good enough for a RC model, not for current at arbitrary time)
- find delay of circuit
 - * decompose to RC model (take into account widths of each transistor)
 - replace transistors with resistors
 - add parasitic caps on either side of every transistor (cap value = width of transistor)
 - caps with both pins short to ground don't count, are never charged
 - caps from V_{DD} to ground don't count, always charged
 - * add together all R and C 's to get delay
 - don't count R of nMOS and pMOS at the same time, because they're never on at the same time
- **Delay** $d = f + p$
- **Effort Delay** $f = gh$
 - * g : logical effort: relative ability of gate to deliver current
 - $g = 1$ for standard inverter
 - * h : electrical effort: ratio of input to output capacitance
 - also called fanout
- **Parasitic Delay** p
 - * independent of load (represents delay of gate driving no load)
 - * set by internal parasitic capacitance
- Elmore Delay:
 - * ON transistors look like resistors, so pullup/pulldown network is modeled as RC ladder
 - * $t = \sum_{i \in \text{nodes}} R_{i \rightarrow \text{source}} C_i$
 $= R_1 C_1 + (R_1 + R_2) C_2 + \dots + (R_1 + \dots + R_n) C_n$
- Ideal number of stages for inverter driving large load
 - * delay $= \left(\frac{C_{load}}{C_{inv}} \right)^{\frac{1}{k}} k R_{inv} C_{inv}$
 - * k : number of stages
 - * stage size ratio: $\left(\frac{C_{load}}{C_{inv}} \right)^{\frac{1}{k}}$

Static Timing Analysis

- worst case at each step
- in form arrival time / required arrival time / slack
- arrival time: input to output, take max
- required arrival time: output to input, take min
 - * work backward
 - * if a gate drives only one gate on it's output, this is trivial
- slack: required arrival time - arrival time
- Contamination delay: just best case delay (smallest delay)

- * in this case, you'll count parallel transistors as parallel resistors
- not sure about this stuff, Peter just said to not worry about this question on the homework
- Power Estimation**
- Dynamic Power
 - * power required to charge the load capacitor
 - * only counted when transistor switches
 - therefore this power usage is data dependent
 - therefore, you have to count the falling transitions in the output per time period
 - * $P_{dynamic} = \frac{1}{T} \int_0^T i_{DD}(t) V_{DD} dt$
 - * for any gate: $P_{dynamic} = C V_{DD}^2 f_{sw}$
- Activity Factor α
 - * α : how often this gate switches in terms of the base clock frequency
 - * $\alpha = 1$: clock; $\alpha = 0.5$: every other cycle, etc...
 - * for system clock at frequency f , $P_{dynamic} = \alpha C V_{DD}^2 f$
- Short Circuit Current
 - * nMOS and pMOS may both be on for a short instant during switching, leading to a short instant of short circuit current (from V_{DD} to ground)
 - * $P_s \propto (V_{DD} - 2V_t)^3 t_r f_p$
 assume $t_r = t_f$ for input
 f_p : frequency of input
 - * this is less than 10% of dynamic power if the rise/fall times are comparable
- Static Power
 - * leakage when gate is off
 - *

$$I_{ds} = I_{ds0} e^{\frac{V_{gs} - V_t}{n v_T}} \left(1 - e^{\frac{-V_{ds}}{v_T}} \right)$$

$$V_t = V_{t0} - \eta V_{ds} + \gamma \left(\sqrt{\phi_s + V_{sb}} - \sqrt{\phi_s} \right)$$

After exam 1

- exam 1 was on March 8 (2017.03.08)
- nothing specifically from previous exam
- Propagation vs Contamination delay**
- contamination delay (t_{cd}): time from input change to **any** output changing value
 - * kind of a best-case delay or smallest possible delay
 - * time counted when input crosses 50% of logical high voltage level
- propagation delay: time from when all inputs are stable to when all outputs are stable
 - * kind of like a worst-case delay
- Latch vs Flip Flop**
- Latch is level-sensitive, flip-flop is edge-sensitive
- register is flip-flop
- Latch/flip flop designs**
- pass transistor latch
 - * pros: tiny, low clock load
 - * con: voltage drop across is V_t , it's non-restoring
 - * con: not really a latch because doesn't hold signal
- transmission gate
 - * pro: no V_t drop
 - * con: requires inverted clock
- inverting buffer
 - * TODO
- tristate feedback
 - * first complete latch here
 - * risk of backdriving: downstream could change state if it is very strong
- buffered output
 - * no backdriving problem
 - * widely used

- * rather large and slow though, and large load on clock
- a flip-flop is just two latches back-to-back

Metastability

- stable works if it's not at a strong high or low but will settle somewhere
- metastable is where it might sit there, but if it must settle, you don't know which one it will settle to when perturbed

Sequential Circuits

- sequential means that the circuit holds state: the output depends on both current and past input
- to model a state machine, you can unroll it to [register] \rightarrow [combinational] \rightarrow ...
 - * this way you can use regular timing stuff for it: arrival time, slack, etc...
- Mealy FSM: output of circuit is from combinational logic
- Moore FSM: output is from registers
- pipelined circuit: uses registers to hold state between clock cycles, because not all of the combinational logic can happen fast enough to work in a single clock cycle
 - * pipelined circuits can use flip-flops or latches?
- the clock consumes 20-30% of the power on a chip
- the whole reason that registers are needed is because data (signals) moves through components at non-constant speed
- reset
 - * can be sync or async
 - * force low output when reset is high
- sequencing: it is (generally) equivalent to split the sequential logic in two, and then use two latches (one halfway through) instead of one large section of sequential logic with flip-flops at either end
 - * this is called two phase clocking
 - * you have to make sure the middle latch operates on clock-bar instead of clock
- timing:
 - t_{pd} logic propagation delay
 - t_{cd} logic contamination delay
 - t_{pcq} Clk \rightarrow Q propagation delay
 - t_{ccq} Clk \rightarrow Q contamination delay
 - t_{pdq} D \rightarrow Q propagation delay
 - t_{setup} setup time of flip-flop/latch
 - t_{hold} hold time of flip-flop/latch
 - * D \rightarrow Q delay only makes sense for latches, since for flip-flops it is simply one clock cycle
- sequencing overhead and max delay:
 - * T_c : cycle time
 - * need for combinational logic to be fast enough
 - * for single phase flip flop: $t_{pd} < T_c - (T_{setup} + T_{pcq})$
 - * for two-phase latch: $t_{pd} = t_{pd1} + t_{pd2} < T_c - 2t_{pdq}$
 - * max delay is dictated by cycle time and sequencing overhead; sequencing overhead does not effect minimum delay
- minimum delay:
 - * for single phase flip flop: $t_{cd} > t_{hold} - t_{ccq}$
 - combinational stuff must be slow enough that it doesn't violate hold time of second flip flop
 - * for two-phase latch:
 - $t_{cd1}, t_{cd2} > t_{hold} - t_{ccq} - t_{non-overlap}$
 - hold time applies twice each cycle, once per each combinational circuit
 - $t_{non-overlap}$: phase between clock signals?
- time borrowing: for two-phase latch-based system, you can borrow time from one segment to the other while the latch is transparent
 - * works (is possible) as long as the total circuit still completes within one clock cycle

- clock skew:
 - * when uncertain: causes tighter required CL propagation delay and minimum CL contamination delay, and reduces opportunity for time borrowing.
 - * when know constant skew: can help with same effect as time borrowing, except can also have effect for flip-flop based systems
- skew tolerance:
 - * flip-flop circuits are not very skew tolerant because the setup and hold times define a specific window of time in which the data must arrive
 - * latch circuits are more hold-time-tolerant because the data can arrive any time that the latch is transparent
- if setup times are violated, you can reduce clock speed to fix it
- if hold times are violated, it won't work at any clock speed

Clocking

- H-Tree (clock distribution network)
 - * laid out in such a way that the delay from the center to any leaf node is equal
- Grid clock distribution
 - * grid on two or more levels
 - * ensures low local skew, but there could still be larger chip-wide skew
- generally, the longer the delay between two point, the larger the worst-case clock skew
- Domino circuit: TODO
 - * a kind of dynamic logic
 - * when the clock pulse cascades through the circuit, activating the pMOS at each stage?

Memory

- random access or sequential access, or content addressable
- CAM: content-addressable memory
 - * basically like an associative array, or database
 - * designed to search all memory at once in a single operation (wikipedia)
 - * used in caches
- SRAM: TODO
 - * 12T and 6T cell designs
 - * 12T design: just a latch connected to a bitline
 - boring and impractical because it's too large
 - * 6T design: cross-coupled inverters (3T each) +2T for bit and bit-bar lines
 - we focus on 6T design
 - * read: pre-charge bit and bit-bar, raise wordline, read result
 - (need to pre-charge or else read is destructive?)
 - need sense amplifiers at the end of the bitlines because the signal will be weak, and the parasitic capacitance of the bitline depends on how many cells are 0 or 1
 - * write: drive data onto bit and bit-bar and raise wordline to write
 - * cross coupled inverters must be much larger than the bitline transistors so that it doesn't flip while reading
 - * per each column, you need:
 - piece of decoder
 - bitline sense amplifier
 - column multiplexer
 - * multi-port: means you can read from and write to the same cell in the same clock cycle and it works
 - implement by having multiple sequential read/write per cycle?
- decoders:
 - * must be pitch-matched to SRAM cell width for layout efficiency/simplicity (requires very skinny gates)

- * large ($n > 4$) decoders are inefficient, thus it is often helpful to use predecoding: factor out common gates into a second stage of decoders.
 - smaller area as alternative, but same logical effort (since same number of gates traversed)
- twisted bitlines: bitlines are long and right next to each other, so there will be significant crosstalk and parasitic cap
 - * solution: swap adjacent bit and bit-bar lines occasionally
 - * reduces Miller factor (MCF)
- DRAM: TODO
 - * one transistor and one capacitor
 - transistor gates capacitor
 - * no charge is 0, charge is 1
 - * open transistor to read/write
 - * read is destructive
 - * must be refreshed periodically due to leakage
 - thus, it's always consuming power
- shift register:
 - * can be implemented with cascade of flip-flops
 - can lead to hold time violations because no delay from clock to next data line? (TODO what? how?)
 - not very dense
 - * store data in SRAM and store begin/end pointers
 - more dense than individual flip-flops
- Tapped Delay Line
 - * shift register with programmable number of stages
 - * implement using sequence of flip-flops, each with a bypass MUX from data in to out
- Queues: are a thing (TODO)
- ROM: Read Only Memory:
 - * mask-programmed ROMs are one transistor per bit
- PROMs, EPROMs, etc
 - * generally burn out specific fuses to program ROM
 - * E for erasable

Wire Delay

- n -segment π model:
 - to model, find total R and C of wire; then split up according to n segment model, then combine adjacent caps

Miller Effect

- AKA crosstalk; it's when one wire affects another
- it's dynamic; depends on what signal the wires are carrying
- MCF: Miller factor (or something)
- for two wires A and B, model as

$$C_{eff} = C_{gnd} + MCF \cdot C_{adj}$$

	behavior of B	MCF
• MCF:	constant	1
	with A	0
	opposite A	2

Timing

- setup time: minimum time that signal must remain steady **before** clock edge
- hold time: minimum time that signal must remain steady **after** clock edge
- propagation delay: when driving another gate? largest (worst case) delay?
- contamination delay: best case delay (smallest delay)?
- parasitic delay: independent of load
- effort delay: proportional to load capacitance (nothing to do with the thing that is doing the driving of the load capacitance)
- maximum possible logic propagation delay for combinational circuit buffered on either side by flip-flops: $\text{clockCycle} - \text{Clk} \rightarrow \text{Q} - \text{FlipFlopSetup} - \text{WorstSkew}$

* minimum circuit delay: $\text{HoldTime} - \text{Clk} \rightarrow \text{Q}$

Memory

- serial access memory (SAM): accessed in a sequence, not randomly
 - * ex. shift register, queue, stack, etc...
 - * often implemented using random access memory
- Shift Register
 - * serial in, parallel out
 - * large number of transistors per cell when implemented using flip flop
 - * tapped delay line: shift register with variable number of stages
 - useful for allowing chips at different clock frequencies to communicate (apparently)
- queue or stack can be built using SRAM block with pointers to first/last and stuff
- SRAM
 - * static ram
 - * can be dual ported: means you can write to a cell while reading the old value from it
 - * designed in a cell grid, MUXes for input and then decoders for output connected to bit lines that run across the cells
 - * TODO note about pre charging bit line
- DRAM
 - * stores bits using capacitors
- decoders
 - * needed to switch to the right bit-line
 - * need to be pitch-matched to RAM cell width for efficiency
 - * TODO note about twisted bit lines
- ROM: read only memory
 - * used to be custom masked chips for data
 - *

PROM

Packaging

- flip chip: put connection pads on the surface of the die instead of the edges
 - * then flip the chip over and affix it directly to the package
 - * means the die must be positioned in the package more precisely
- Heat
 - * Thermal Resistance: $\Delta T = \theta_{ja} P$
 - ΔT : temperature rise in chip
 - θ_{ja} : thermal resistance of chip junction to ambient; units: C/W
 - P : power dissipation on chip
- IO
 - * ESD Protection: TODO
 - * output pads:
 - must drive large off-chip loads (2-50pF) do that using successively larger buffers
 - surround with guard ring to protect against lockup
 - * input pads:
 - may need level converters
 - Schmitt trigger: allow signal to cross high/low edge only after it goes a little over the tipping point
 - * bidirectional pads:
 - combine input and output
 - use tristate driver to set pin direction
 - * analog:
 - no buffering
 - any protection circuitry must be careful to not distort the signal at all
 - RF pads?