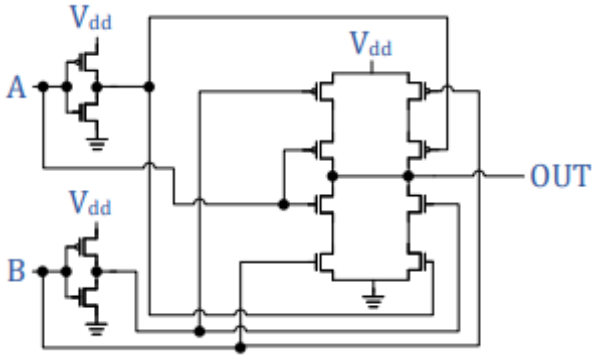
Multiples of bytes

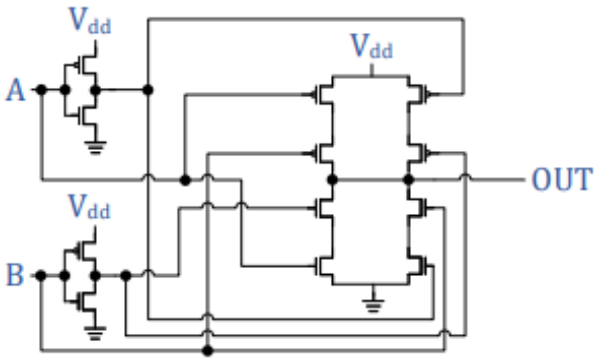
Value	Unit	Metric
1	B	byte
$10^3 \approx 2^{10}$	KB	kilobyte
$10^6 \approx 2^{20}$	MB	megabyte
$10^9 \approx 2^{30}$	GB	gigabyte
$10^{12} \approx 2^{40}$	TB	terabyte
$10^{15} \approx 2^{50}$	PB	petabyte
$10^{18} \approx 2^{60}$	EB	exabyte
$10^{21} \approx 2^{70}$	ZB	zettabyte
$10^{24} \approx 2^{80}$	YB	yottabyte

Transistor-Level Circuit Design: Every time you have PMOS in parallel the NMOS is in series, if PMOS in series then NMOS is in parallel.

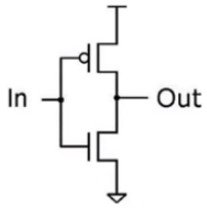
- **XOR**



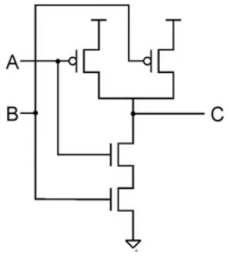
- **XNOR**



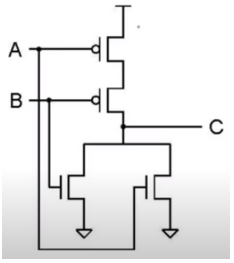
- **NOT**



- **NAND**



- **NOR**



Verilog:

- "always" statements are used to describe sequential circuits, because they remember the old state when no new state is prescribed. However always statements can also be used to describe combinational logic behaviorally if the sensitivity list is written to respond to changes in all of the inputs and the body prescribes the output value for every possible input combination.
- **Wire:** Can be read or assigned. No values stored in them. They need to be driven by either a continuous assign statement of from a port of a module
- **Reg:** Represent data storage elements in Verilog. They retain their value till next value is assigned to them (not through assign statement). They must be driven in an always block. They cant be assigned as the output when declaring a module.
- Modules cant be instantiated recursively.

Finite State Machines: Encoding types:

- **One-hot encoding:** Reduces next-state logic
- **Binary encoding:** Reduces FFs to hold state (only $log_2 K$ bits of state needed for K states)
- **Output encoding:** Reduces output logic

MIPS:

- The size of a MIPS instruction is 4 bytes
- when creating a function dont forget to allocate and deallocate space on the stack. (Allocation: addi $\$sp, \$sp, -16$ i.e for 4 regisiters, save the registers with sw and then load the arguments. Restoring the stack just add the amount subtracted at the beginning)

Microarchitecture: Connection between logic and architecture, it is the arrangement of regsiters,ALU, FSM, memories needed to implement the architecture. It is the actual underlying implementation of the machine. Changes to the microarchitecture are transparent to the compiler/programmer.

ISA level The interface a machine exposes to the software. Changes to the ISA affect the compiler/programmer

Processors:

- A processor with a lower frequency might be able to execute multiple instructions per cycle while a processor with a higher frequency might only execute one instruction per cycle.
- The total number of instructions required to execute a program could be different on different processors.
- the frequency of a processor is the number of cycles per second.