## Lecture 21:

## Combinational Circuits

- Integrated Circuits
- Combinational Circuits
- Multiplexer
- Demultiplexer
- Decoder
- Adders
- ALU


## Integrated Circuits

- Circuits use modules that contain multiple gates packaged together, rather than individual gates.
- These are called Integrated Circuits (ICs, chips)
- SSI (small scale integration): 1-10 gates/chip
- MSI (medium scale integration): 10100 gates/chip
- LSI (large scale integration): 100 100,000 gates/chip
- VLSI (very large scale integration): more than 100,000 gates/chip



## Integrated Circuits, cont.

- Current technology could put 5 million NAND gates on a chip!
- But... that chip would need 15,000,002 pins.
- With standard pin spacing, an 18 km long chip.
- Instead, circuits are designed with a high gate/pin ratio.


## Combinational Circuits

- Def: a set of interconnected gates whose output at any time is a function of the input at that time.
- The appearance of input is followed almost immediately by output, with only gate delays.



## Multiplexer (MUX)

- A circuit that goes from many inputs to one output.
- The select lines are used to pick one of the input lines to directly output to the output line.


## MUX, cont.

- S1 and S0 are connected to AND gates in such a way that for any combination of S0 and S1,3 of the AND gates will output 0
- The $4^{\text {th }}$ AND gate will output the value of the selected input line.
- So, 3 inputs to the OR-gate will always be 0 , and the output of the OR-gate will equal the value of the selected input gate.


## Multiplexer Uses

- Device controllers
- Parallel-to-serial data converter



## Decoder

- A circuit that asserts one output line, depending on a pattern of input lines.
- In this circuit, inputs are the select lines. The line they select gets a one, all other lines get zero.


## Decoder, cont.



- Truth table for 1-bit addition:

| A | B | Sum Carry |  |
| :--- | :--- | :--- | :--- |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |



## Adders

- This can be drawn using our previous method:


## Decoder Uses

- Address decoding
- Suppose you wish to construct a 1 K byte memory using $4256 \times 8$-bit RAM chips. Want a single unified address space.
Address Chip
000-0FF 0
100-1FF 1
200-2FF 2
300-3FF 3
- Each chip needs 8 address lines (256 bits). These are supplied by the loworder 8 -bits of the address.
- High-order 2 bits (of the 10-bit address) are used to select 1 of 4 chips.
- Or, we could notice that the Sum is the XOR of A and B:
- This circuit is known as a HalfAdder.


## Half-Adder vs. Full Adder

- To be useful for arithmetic, need to also consider carry-in:

1011
$+0011$
$1 / \begin{aligned} & \text { half-adder computes } \\ & \text { this correctly }\end{aligned}$
half-adder wouldn't calculate this correctly

## Full-Adder, cont.

- For sixteen bit words, wire together 16 1-bit full adders.
- Wire CarryIn for lowest bit to zero.
- CarryIn for the remaining bits should be wired to the CarryOut of the previous bit.


## Full-Adder

- For multiple-bit addition, need a full adder.
- Truth table:

| A B CarryIn | Sum | CarryOut |  |  |
| :---: | :---: | :---: | :---: | :---: |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

## Arithmetic Logic Units

- Most computers have a single circuit for performing AND, OR, and sum of two words.
- For n-bit words, built from n identical circuits or individual bit positions.
- These are known as 1-bit ALUs or bit slices.


