

# MIT 6.004 - Computation Structures Lecture 2 - Combinational Devices and Boolean Algebra

## Lecture 2 - Combinational Devices and Boolean Algebra

### Combinational Devices

A combinational device has to satisfy the following criteria in order to be a valid combinational device:

1. It has to have one or more digital inputs
2. It has to have one or more digital outputs
3. It has to have a functional specification that details the value of each output for every possible combination of valid inputs
4. It has to have a timing specification consisting at minimum of a propagation delay ( $t_{pd}$ ), which is the maximum amount of time the device needs to produce a valid output on receiving valid inputs

These criteria are called the static discipline. Moreover, a combinational device consisting of many devices also has to satisfy the following:

1. Each sub-device has to be combinational
2. Every input to a sub-device must either be driven by a single previous output or a constant 1 or 0
3. The circuit should have no cycles

### Functional Specifications

There are many ways to give a functional specification of a combinational device. The two main ones are:

- Truth tables
- Boolean expressions

When implementing the functional specification of a circuit, we usually convert the truth table representation to a boolean expression, and implement that. However, there are many ways to optimize a standard sum-of-products type boolean expression

into a minimal form that uses the fewest gates, so we use tools to synthesize circuits. These tools take as input the high-level circuit specification (in an HDL language), the set of available gates to be used, and the optimization goals (area, delay, power, etc.), and return an optimized circuit.

**Note**

The NAND and NOR gates together are universal gates, which means you can represent any boolean expression using just those two gates.

In CMOS technology, inverting gates (NAND, NOR, NOT) are faster and smaller than non-inverting gates (AND, OR).