Lect 16

Programmable Logic and Hardware Programming (Verilog HDL)

CS221: Digital Design

Dr. A. Sahu

Dept of Comp. Sc. & Engg.

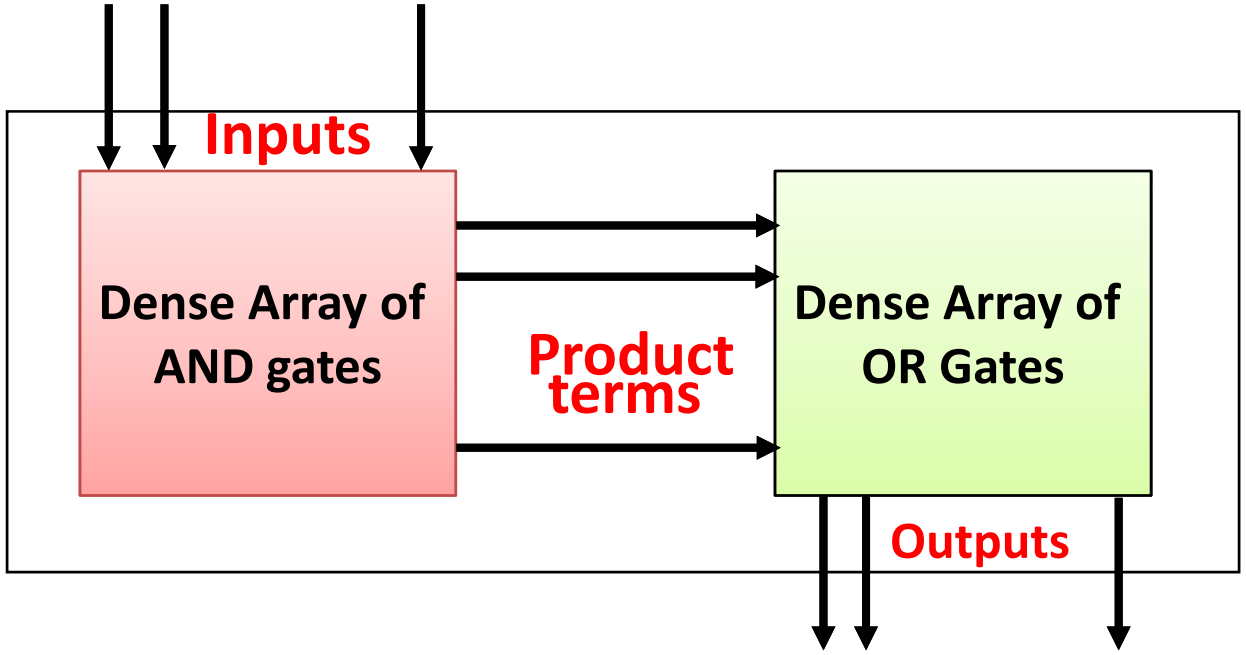
Indian Institute of Technology Guwahati

Outline

- Programmable Logic
- PAL, PLA,
- Memory
  - –ROM, PROM, EPROM, EEPROM
  - –SRAM : Memory Cell
- CPLD, CLB, FPGA
- FPGA/ASIC Design Flow
- HDL Programming : Verilog HDL

Programmable Logic Organization

- Pre-fabricated building block of many AND/OR gates (or NOR, NAND)
- "Personalized" by making or breaking connections among the gates

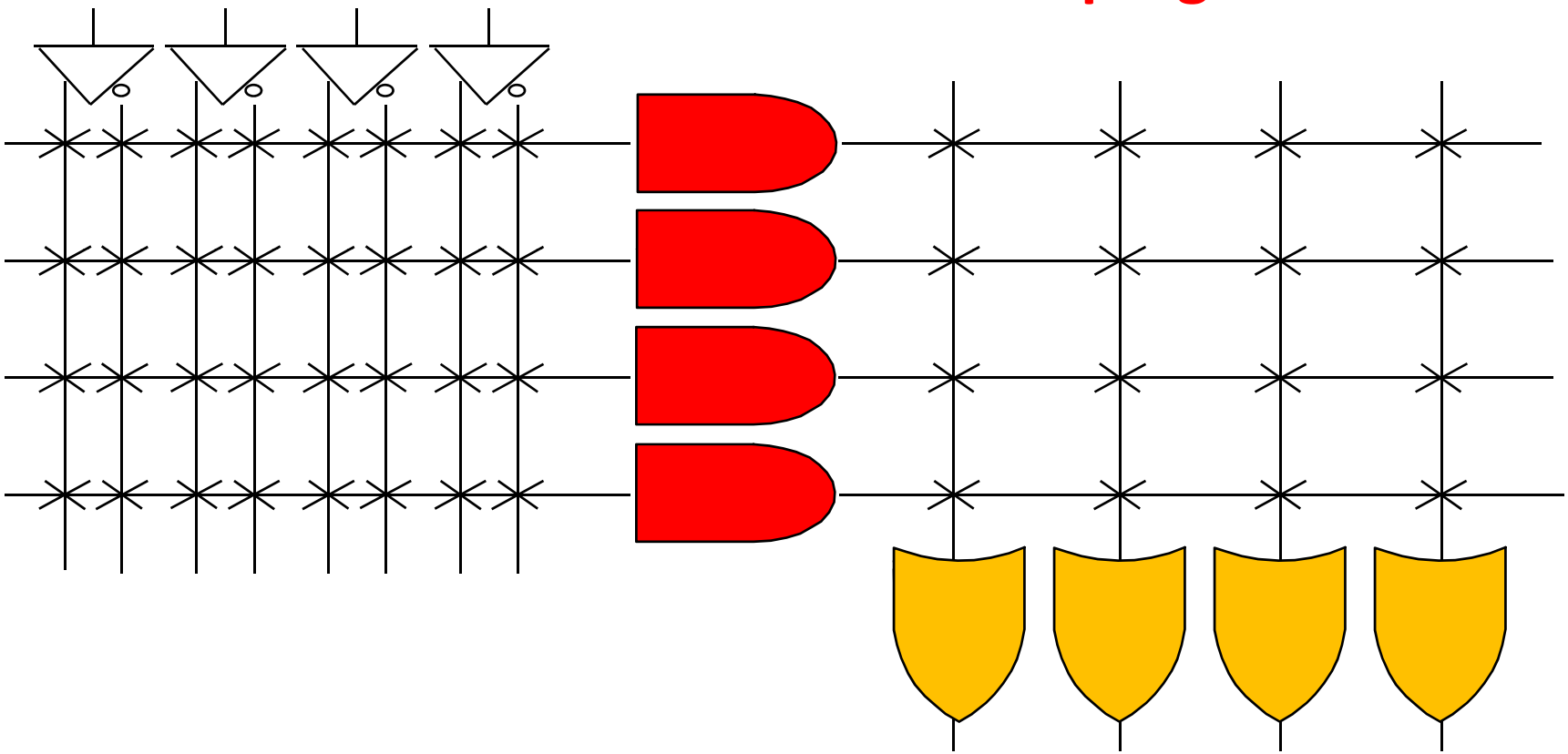


Programmable Array Block Diagram for Sum of Products Form

1. PLA Logic Implementation

Alternative representation

Un-programmed device



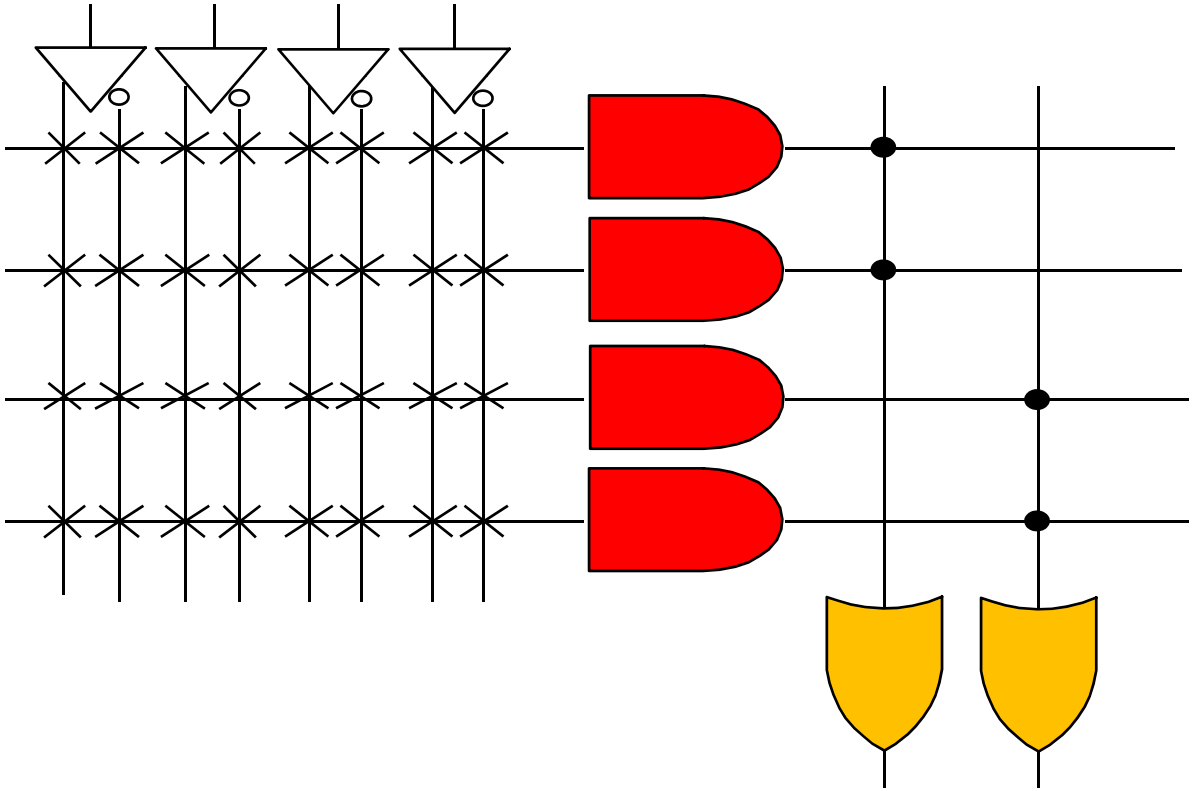
Short-hand notation so we don't have to draw all the wires!

X at junction indicates a connection

2. PALs

What is difference between Programmable Array Logic (PAL) and Programmable Logic Array (PLA)?

PAL concept — implemented by Monolithic Memories
AND array is programmable, OR array is fixed at fabrication

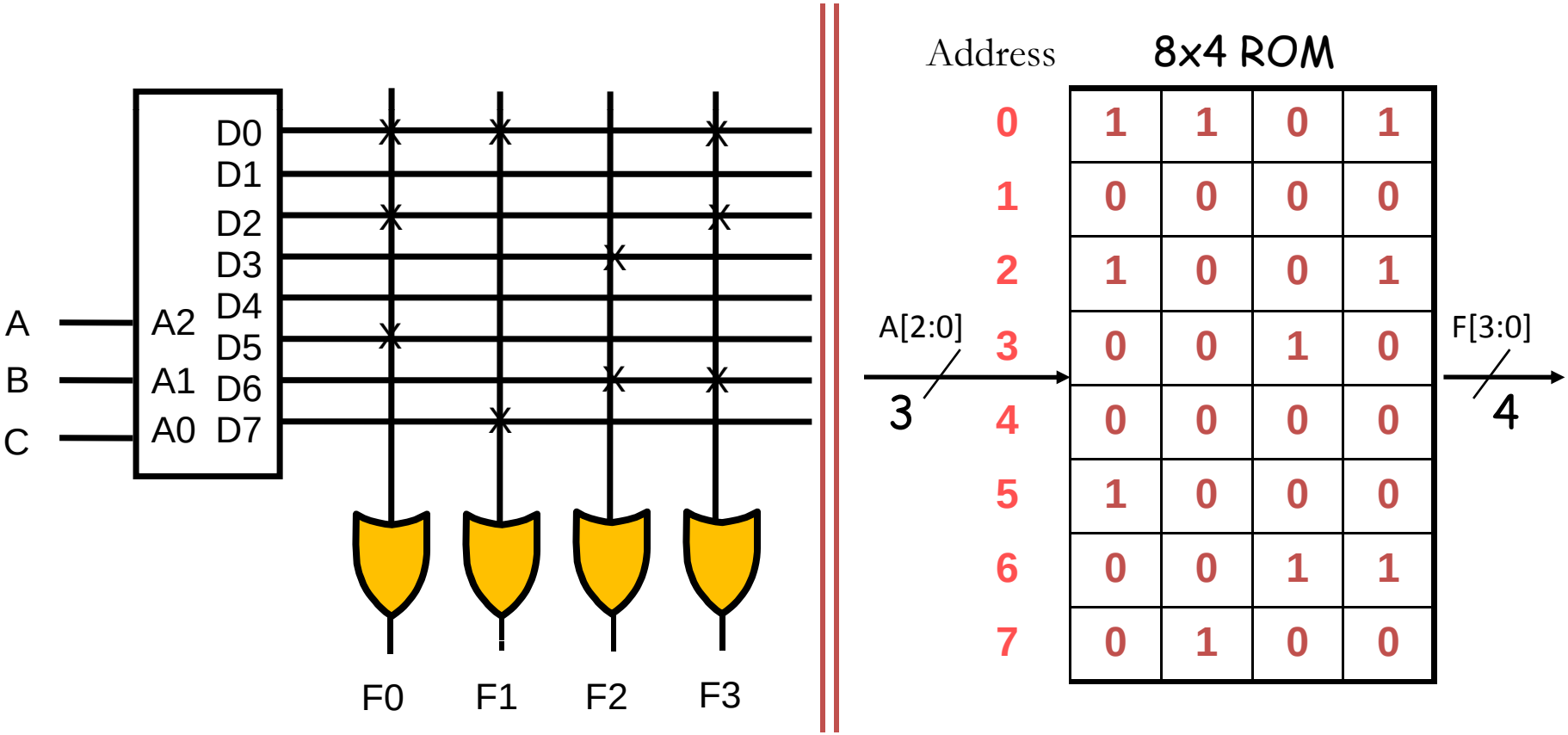


A given column of the OR array has access to only a subset of the possible product terms

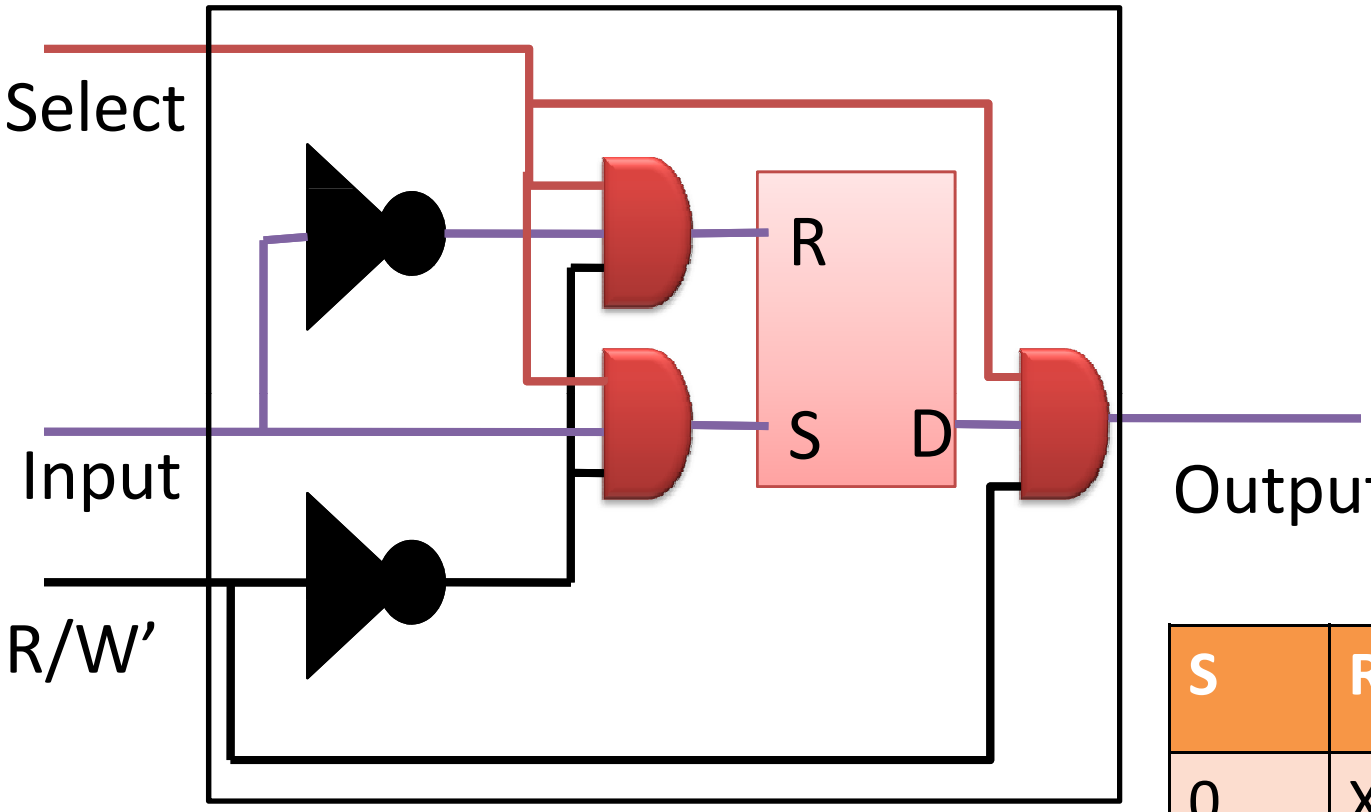
PLA concept — Both AND and OR arrays are programmable

3. ROM as Memory

- **Read Example**: For input $(A_2,A_1,A_0) = 011$, output is $(F_0,F_1,F_2,F_3) = 0010$.
- What are functions F_3, F_2 , F_1 and F_0 in terms of (A_2, A_1, A_0)?

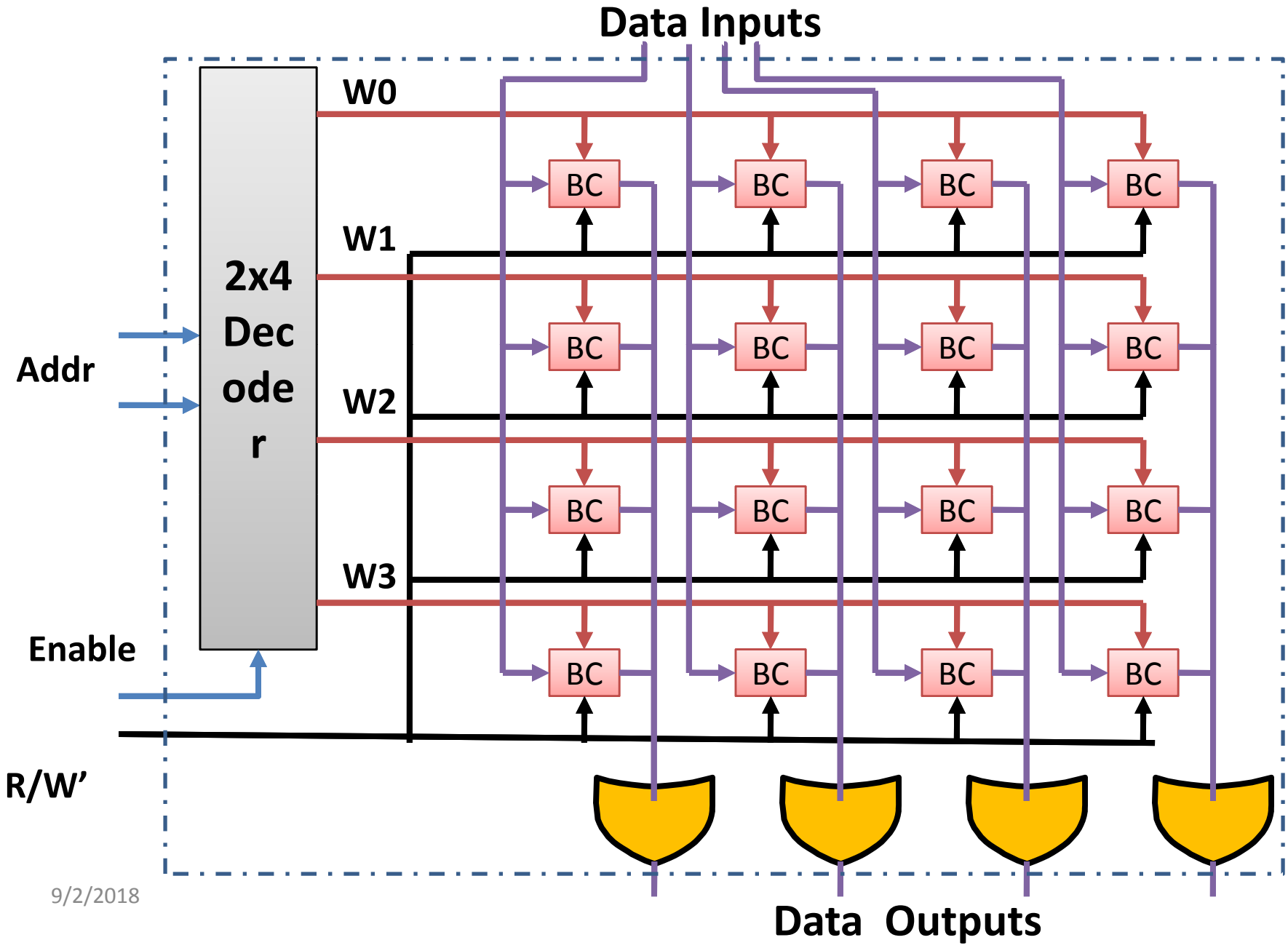


4. SRAM: Memory Cell



S	RW'	D	O/p
0	X	X	0
1	1	X	D
1	0	In	0

4. SRAM : Memory



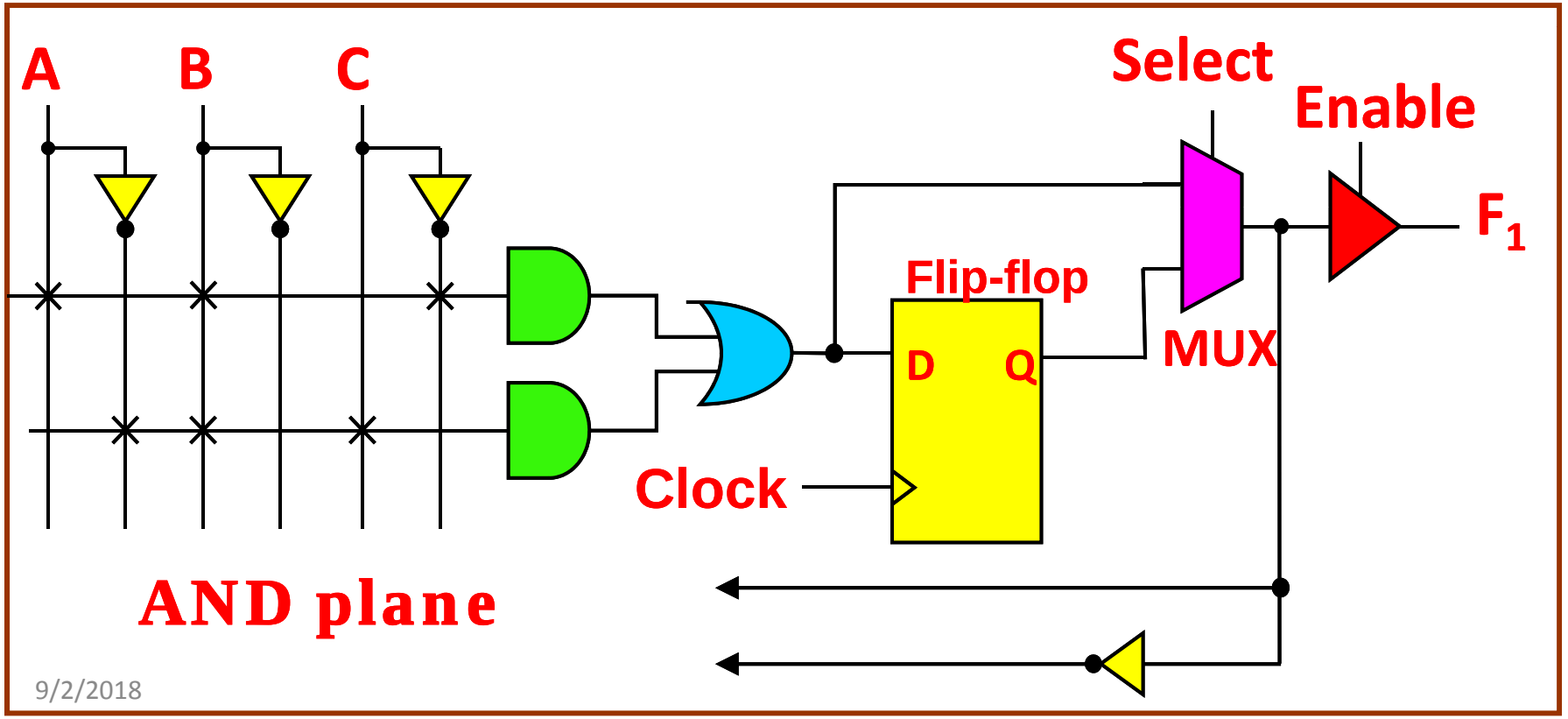
Advanced Programmable Logic Devices

SPLD , CPLD and FPGA

- Simple Programmable logic device
  - Single AND Level
  - Flip-Flops and feedbacks
- Complex Programmable logic device
  - Several PLDs Stacked together

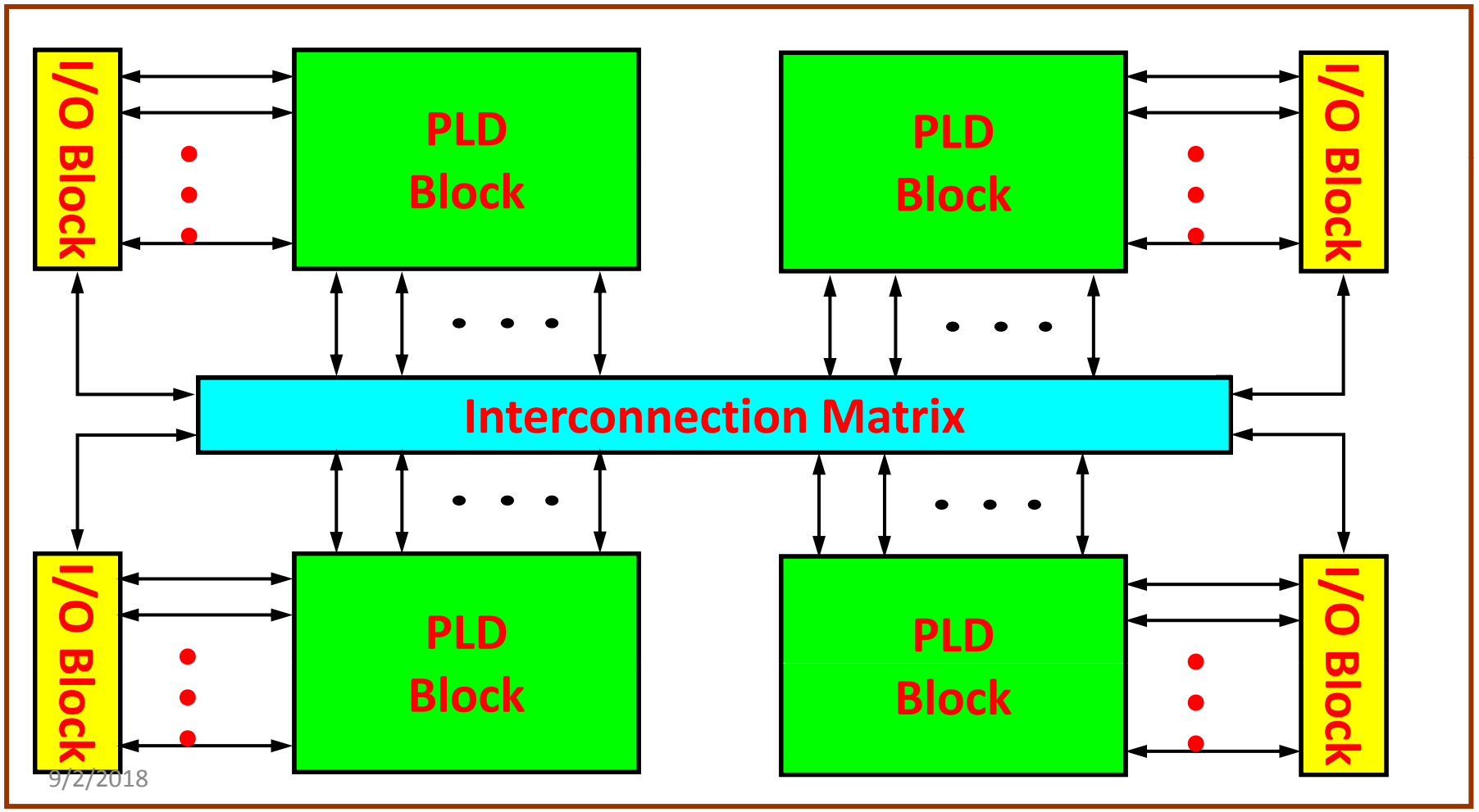
SPLD - CPLD

- Simple Programmable logic device
  - Single AND Level
  - Flip-Flops and feedbacks



SPLD - CPLD

- Complex Programmable logic device
  - Several PLDs Stacked together



FPGA

Field Programmable Gate Arrays (FPGAs)

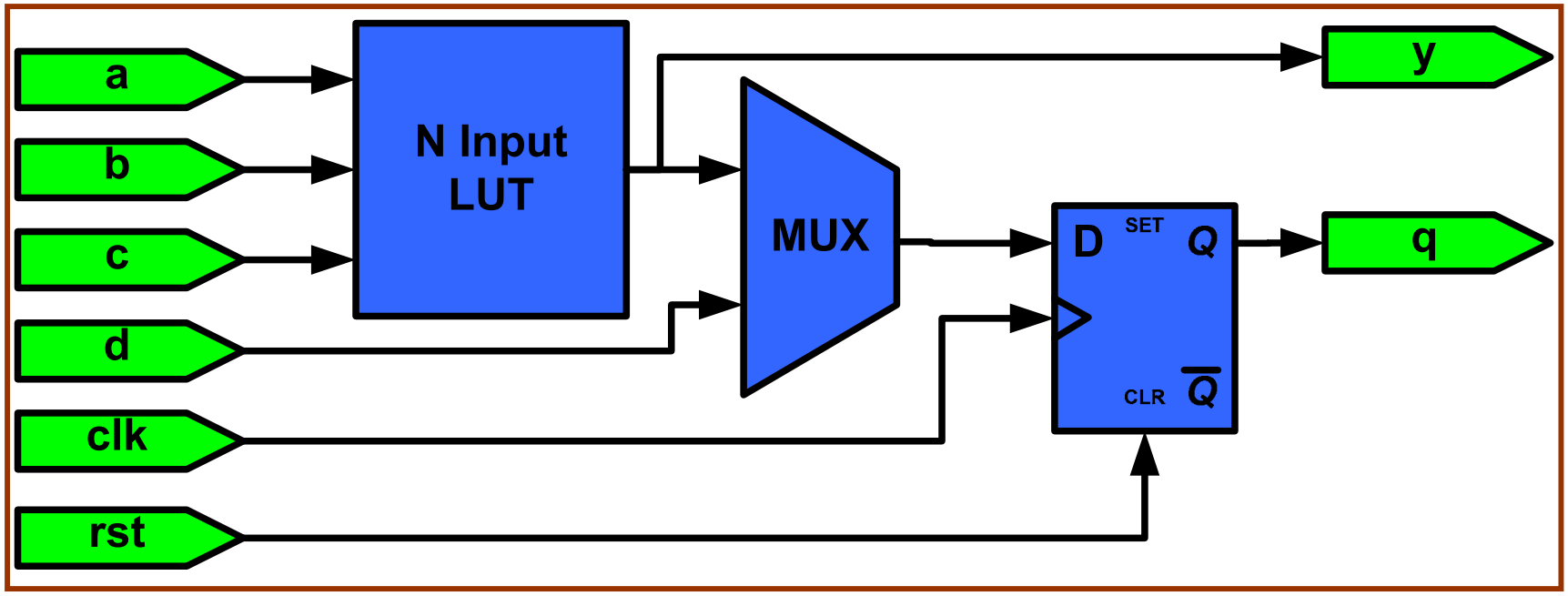
- FPGAs have much more logic than CPLDs
  - 2K to >10M equivalent gates
  - Requires different architecture
  - FPGAs can be RAM-based or Flash-based
    - RAM FPGAs must be programmed at power-on
      - External memory needed for programming data
      - May be dynamically reconfigured
    - Flash FPGAs store program data in non-volatile memory
      - Reprogramming is more difficult
      - Holds configuration when power is off

FPGA - Field Programmable Gate Array

- Programmable logic blocks (Logic Element “LE”) or CLB
  - Implement combinatorial and sequential logic. Based on LUT and DFF.
- Programmable I/O blocks
  - Configurable I/Os for external connections supports various voltages and tri-states.
- Programmable interconnect
  - Wires to connect inputs , outputs and logic blocks.
  - Clocks
  - short distance local connections
  - long distance connections across chip

FPGA - Field Programmable Gate Array

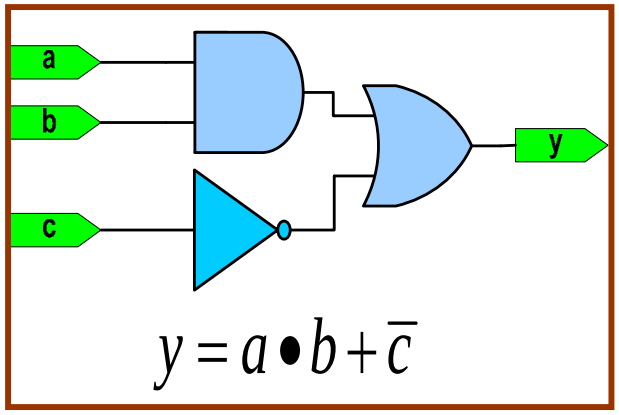
- Programmable logic blocks or CLB
  - (Logic Element “LE”)
  - Implement combinatorial and sequential logic. Based on LUT and DFF.



Configuring LUT

- LUT is a RAM with data width of 1bit.
- The contents are programmed at power up

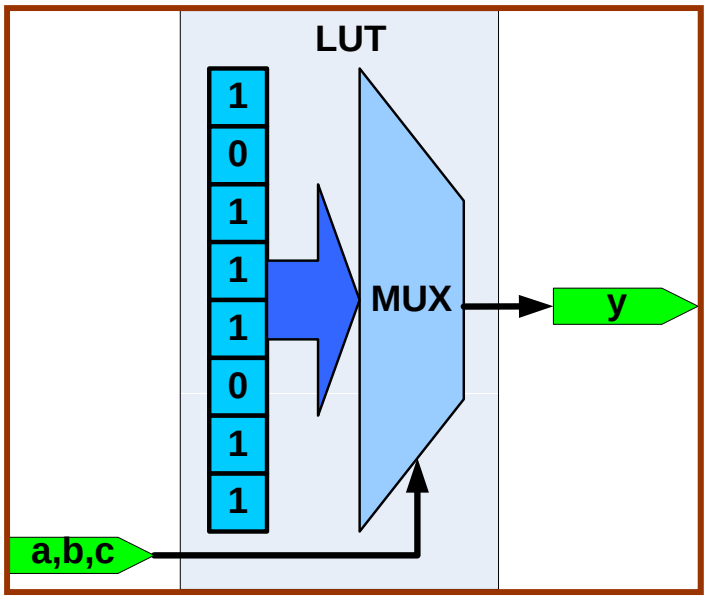
Required Function



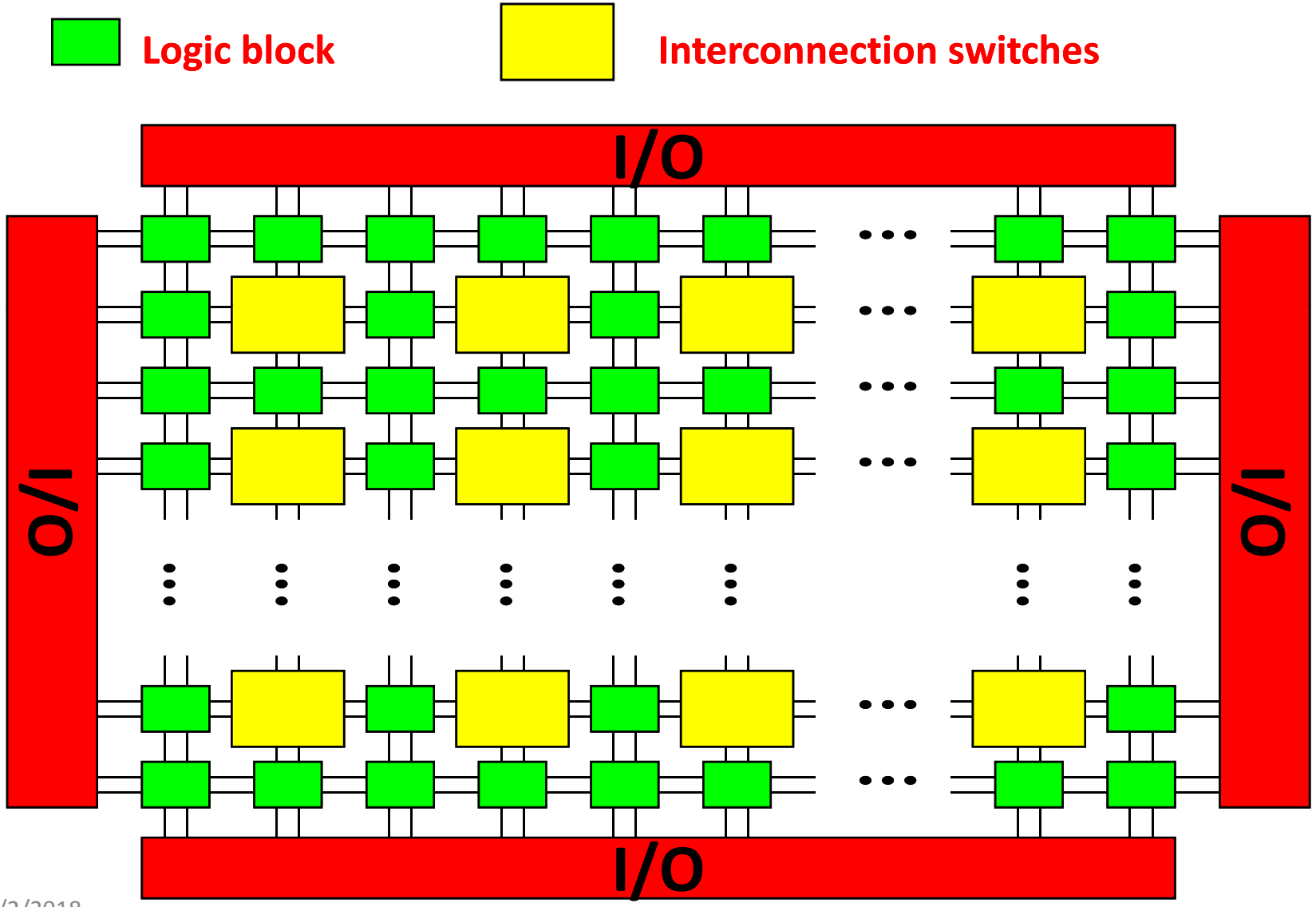
Truth Table

a	b	c	y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Programmed LUT

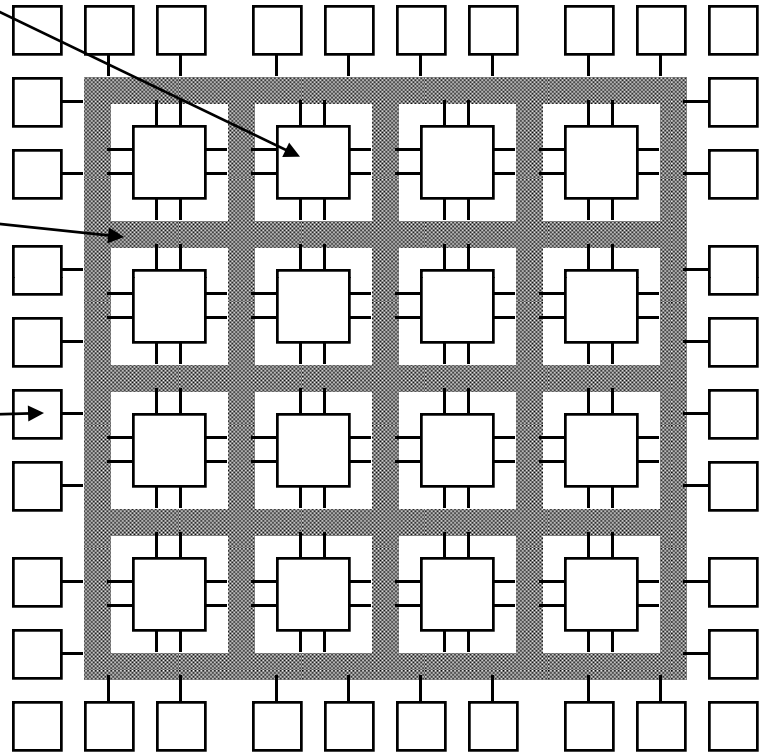


FPGA - Field Programmable Gate Array

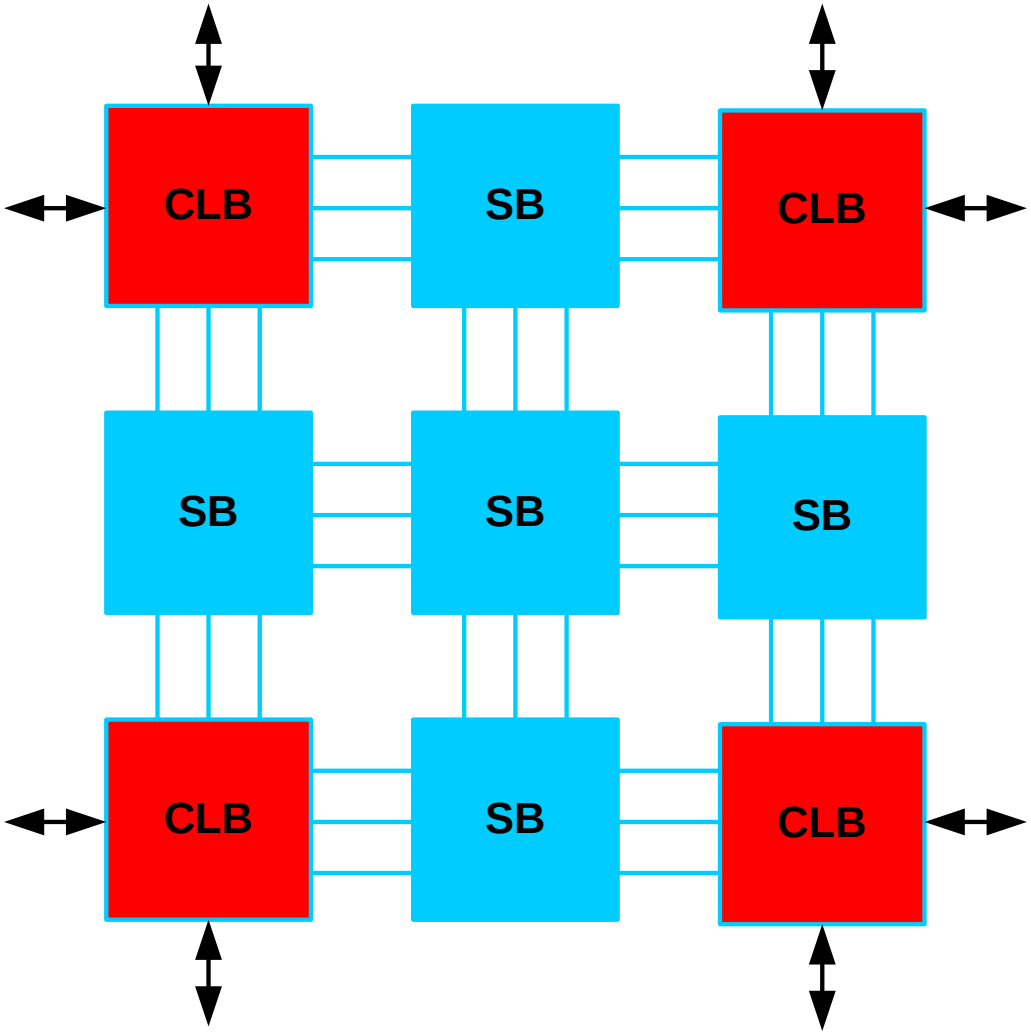


Field-Programmable Gate Arrays structure

- **Logic blocks**
  - To implement combinational and sequential logic
- **Interconnect**
  - Wires to connect inputs and outputs to logic blocks
- **I/O blocks**
  - Special logic blocks at periphery of device for external connections
- **Key questions:**
  - How to make logic blocks programmable?
  - How to connect the wires?



FPGA structure

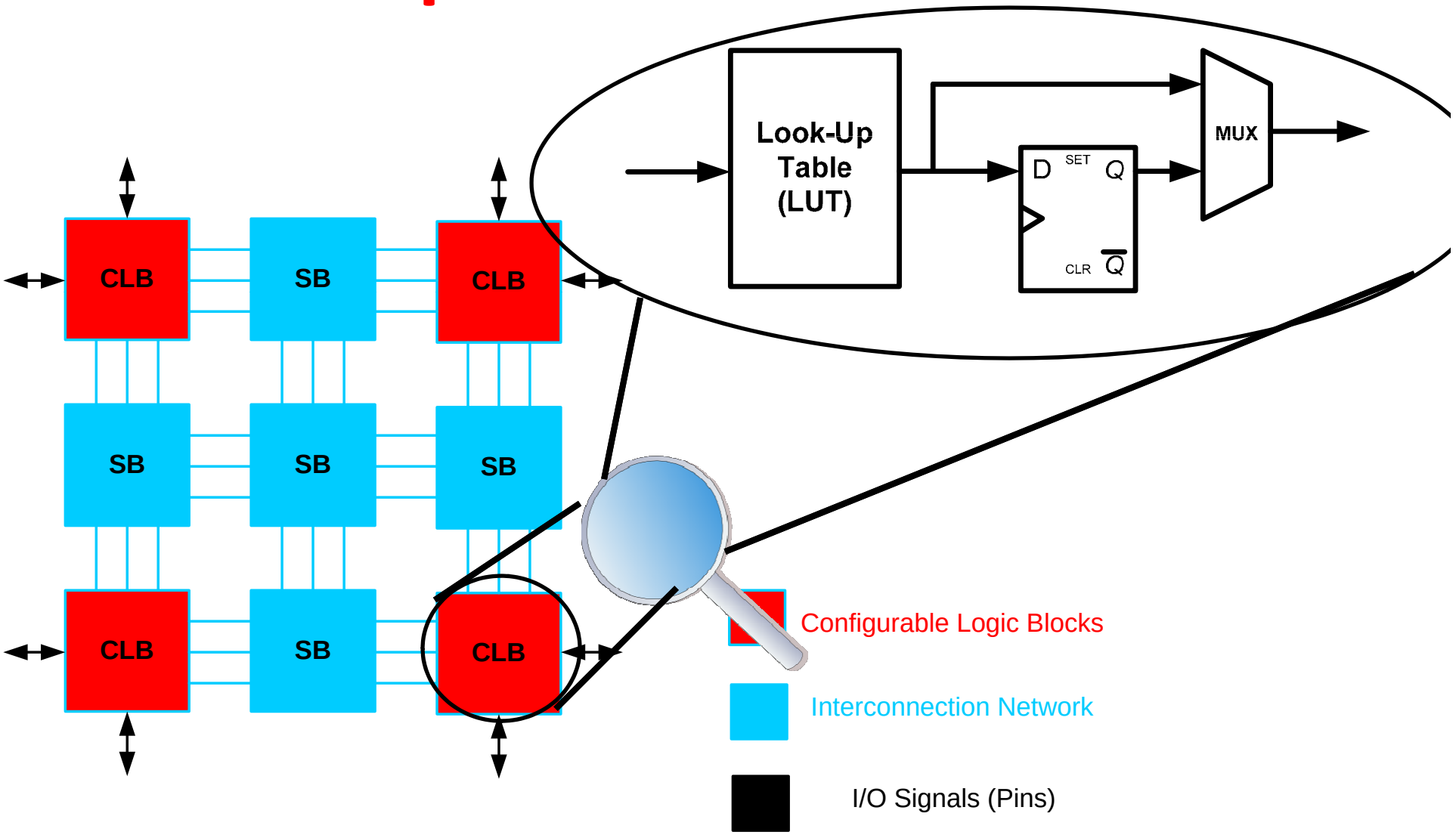


Configurable Logic Blocks

Interconnection Network

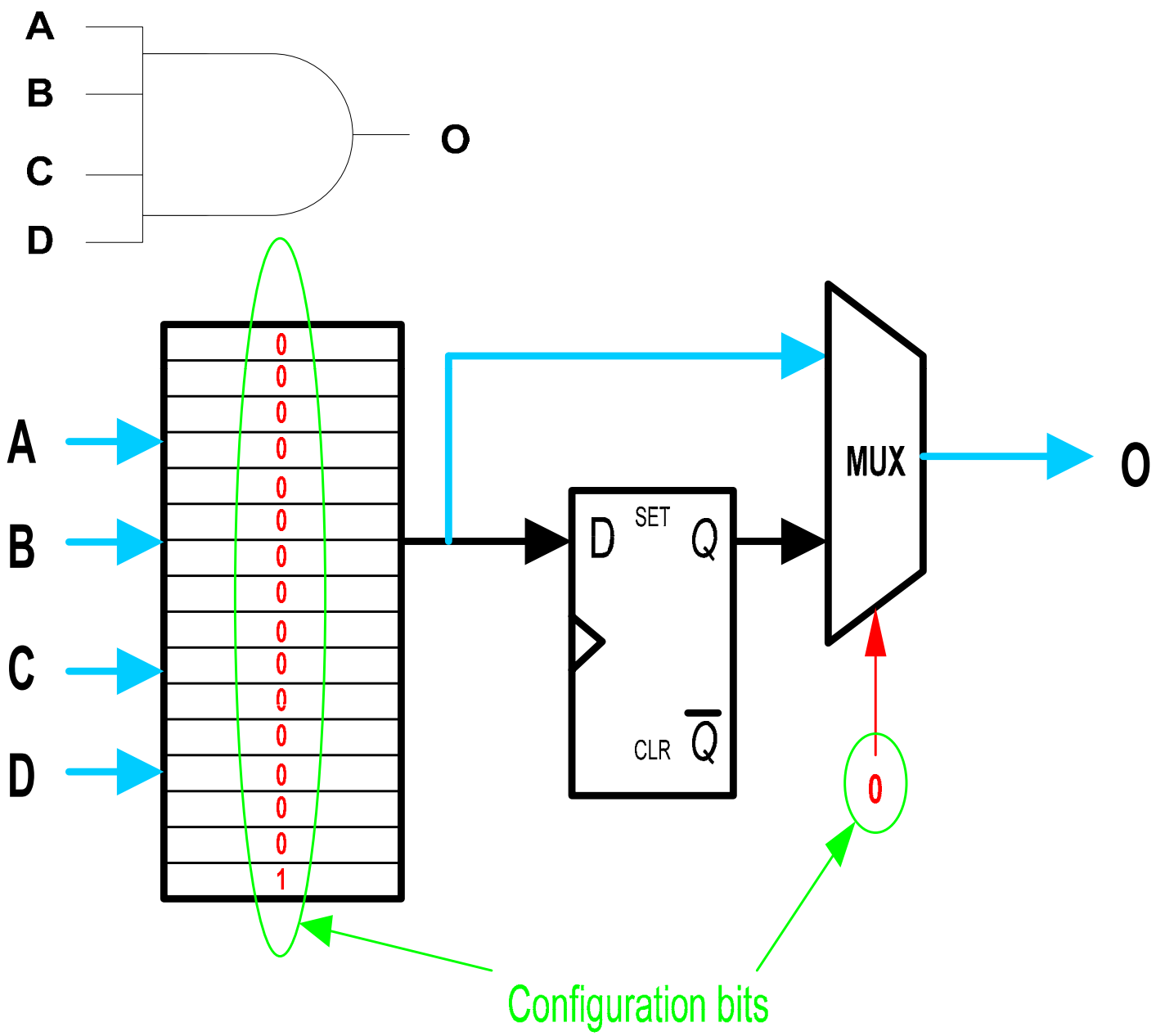
I/O Signals (Pins)

Simplified CLB Structure

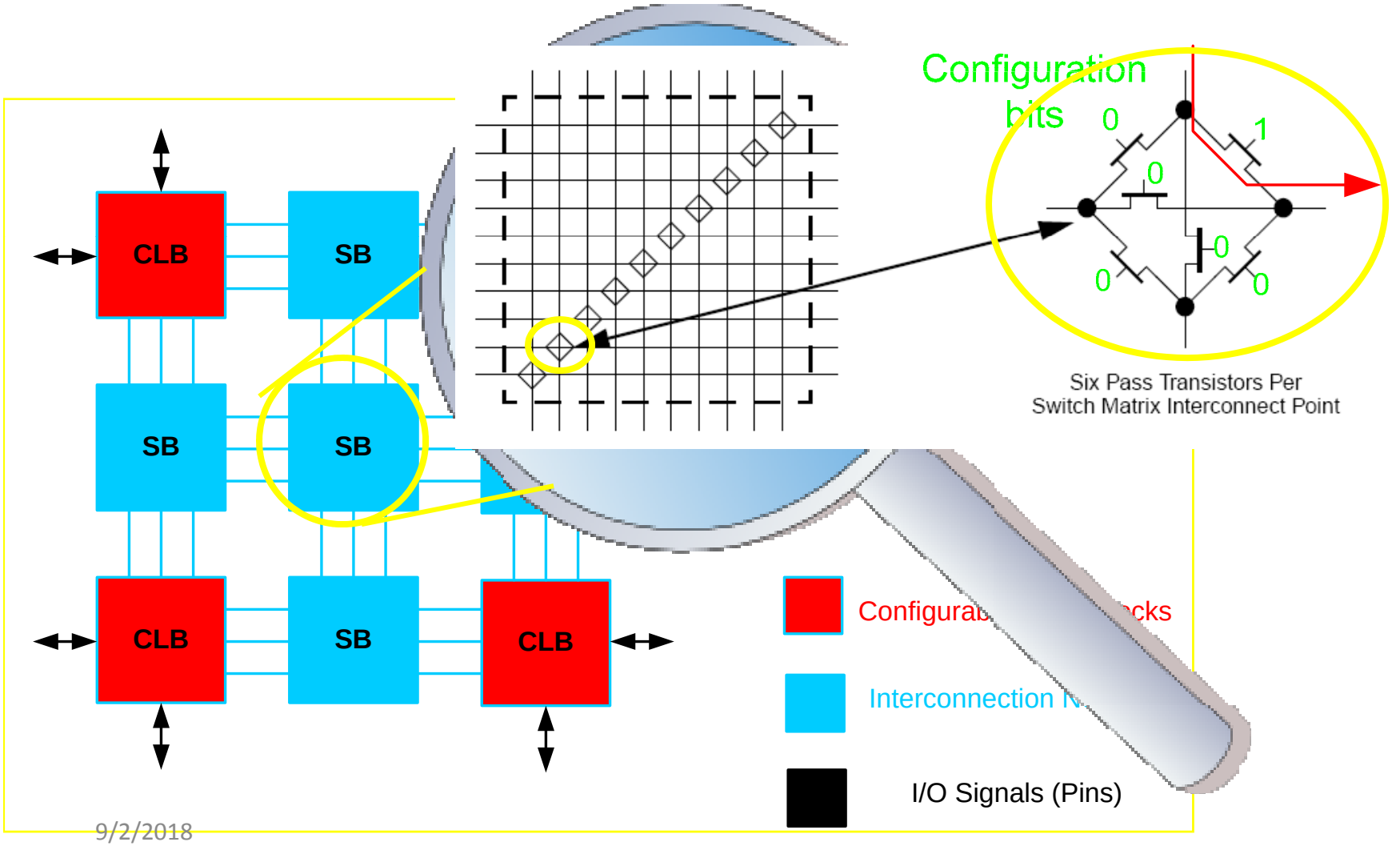


Example: 4-input AND gate

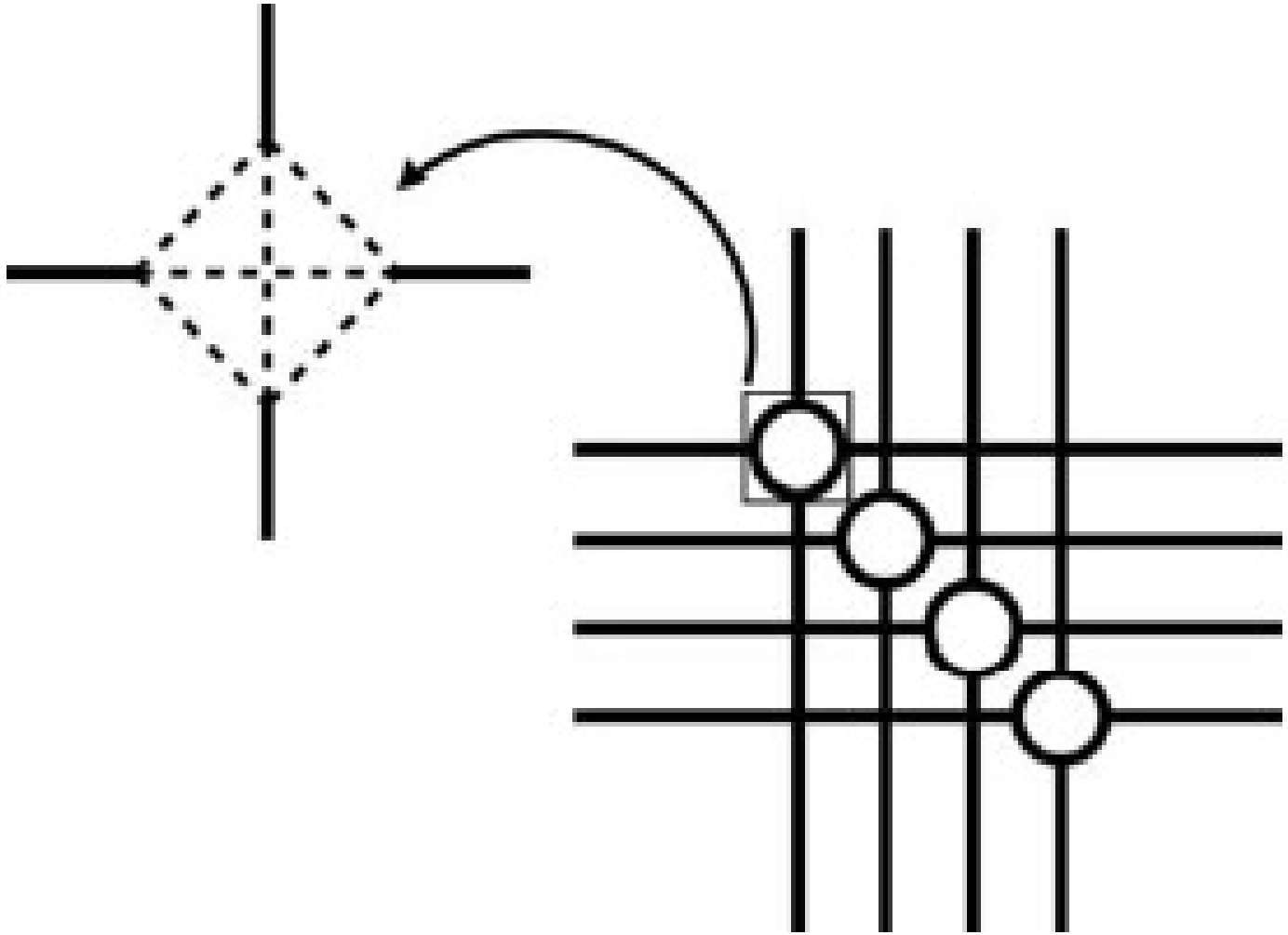
A	B	C	D	O
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1



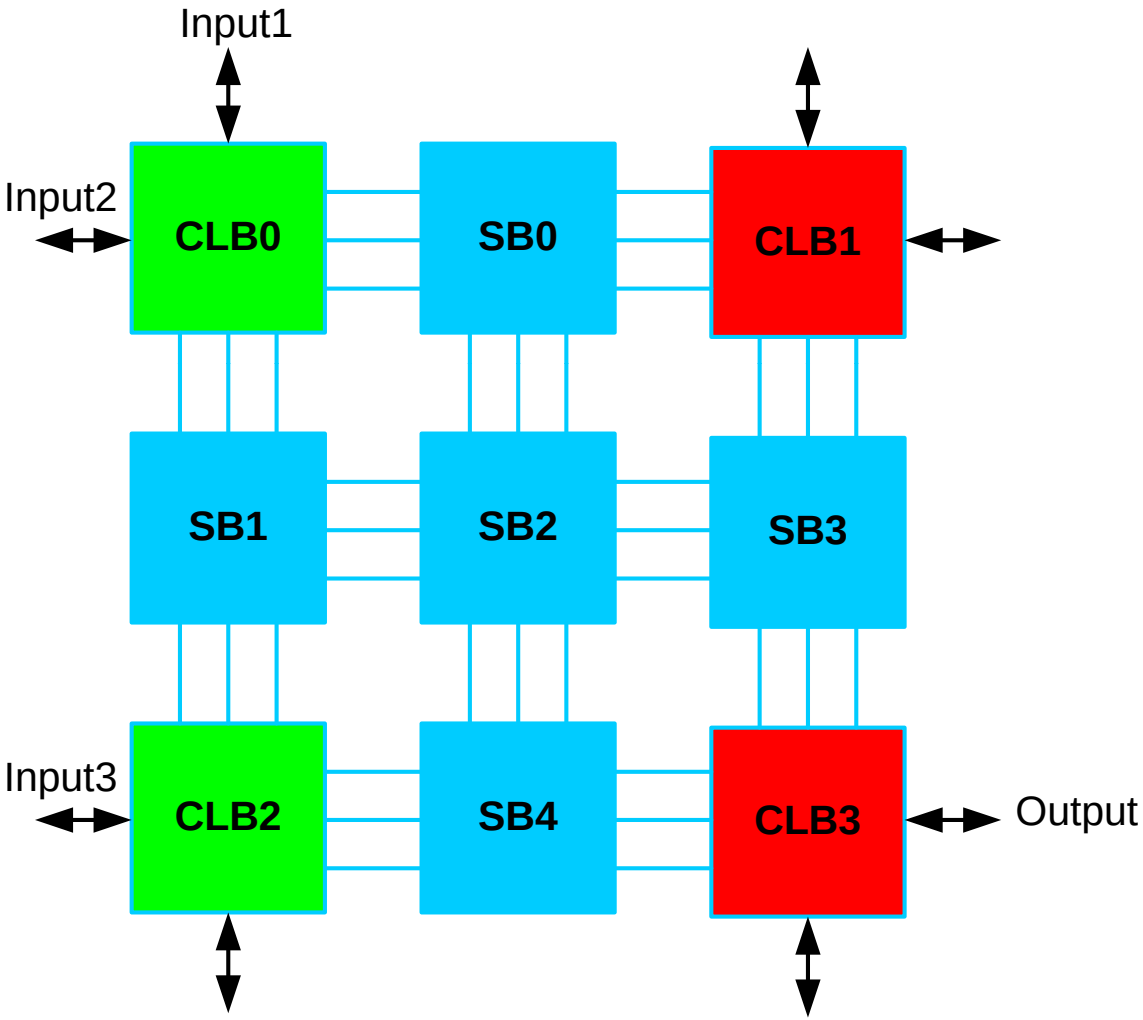
Interconnection Network



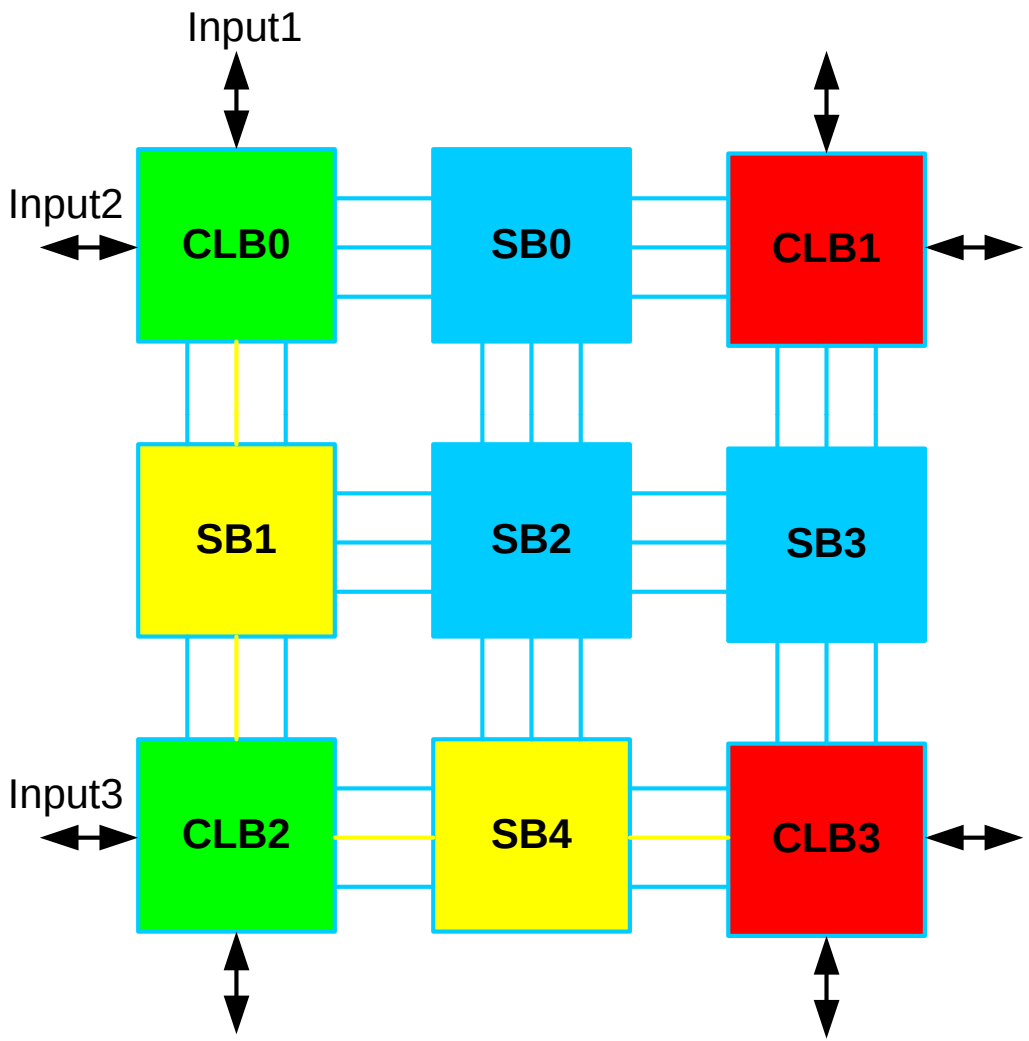
Configurable Interconnect

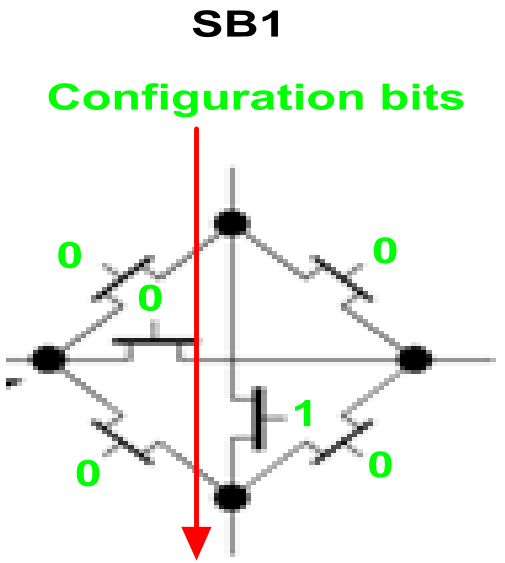


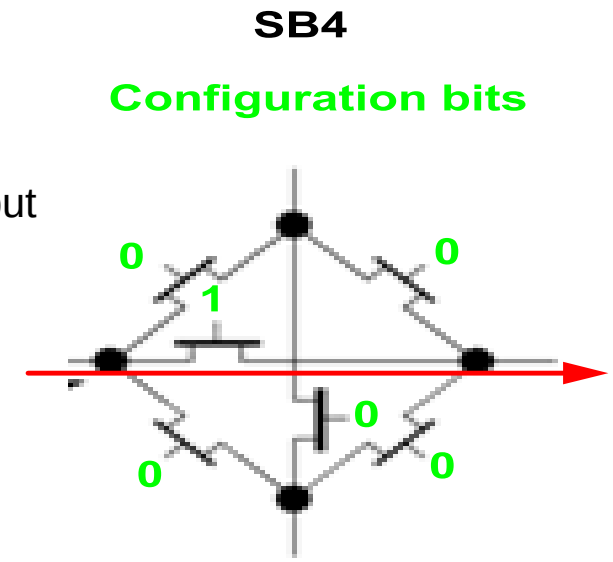
Placement: Select CLBs



Routing: Select path







Shannon's expansion theorem

Used to implement many variable logic functions using MUX and LUTs

$f(x1, x2, \ldots, xn) = x1\ f(0, x2, .., xn) + x1\ f(1, x2, \ldots, xn)$

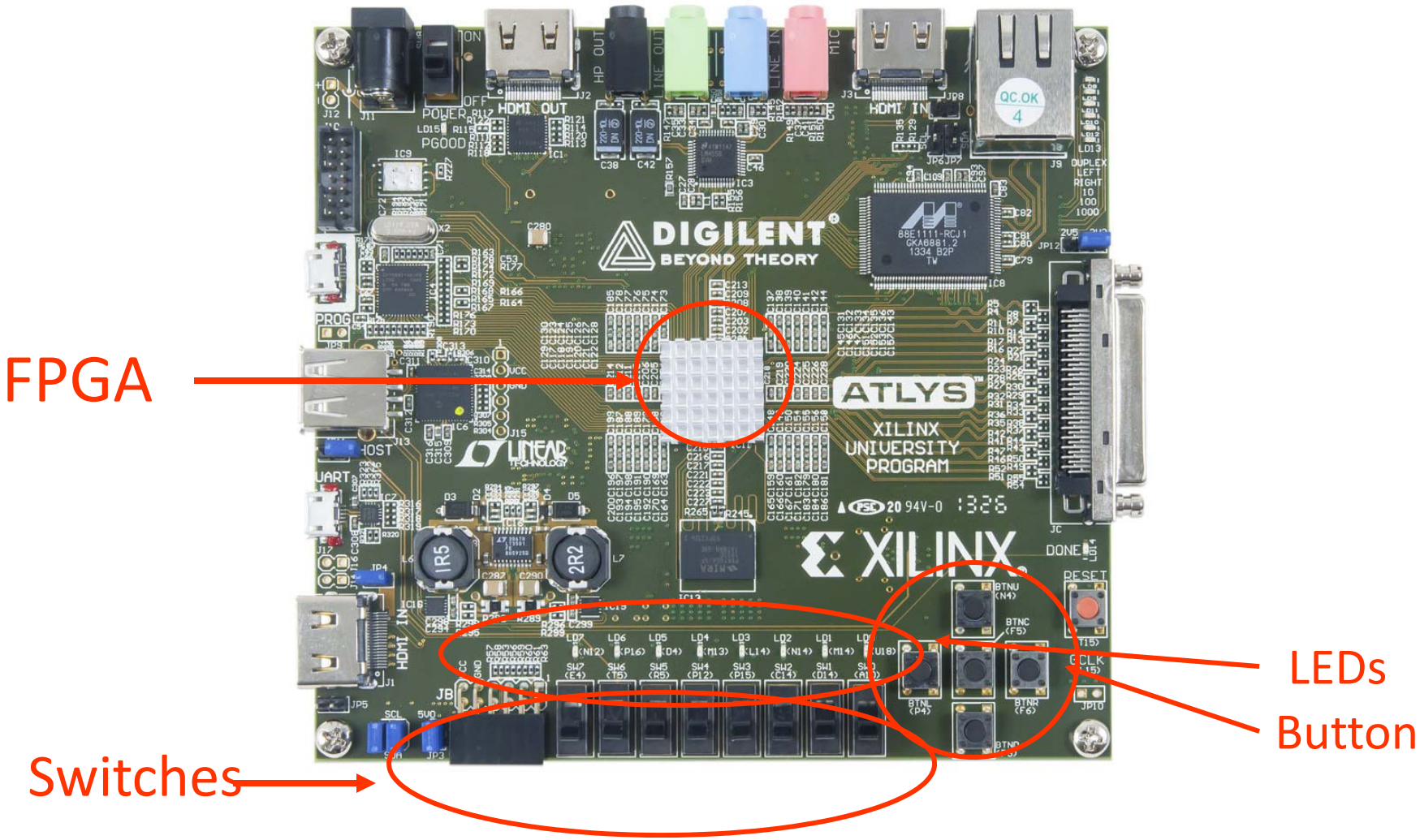
Since x1 is a boolean variable, we need to look at only two cases: x1 = 0 and x1 = 1.

- **Setting x1 = 0 in the above expression, we have:**
  - $f(0, x2, \ldots, xn) = 1\ f(0, x2, \ldots, xn) + 0\ f(1, x2, \ldots, xn)$
    $= f(0, x2, \ldots, xn)$
- **Setting x1 = 1, we have:**
  - $f(1, x2, \ldots, xn) = 0\ f(0, x2, \ldots, xn) + 1\ f(1, x2, \ldots, xn)$
    $= f(1; x2; \ldots; xn)$

FPGA Structures

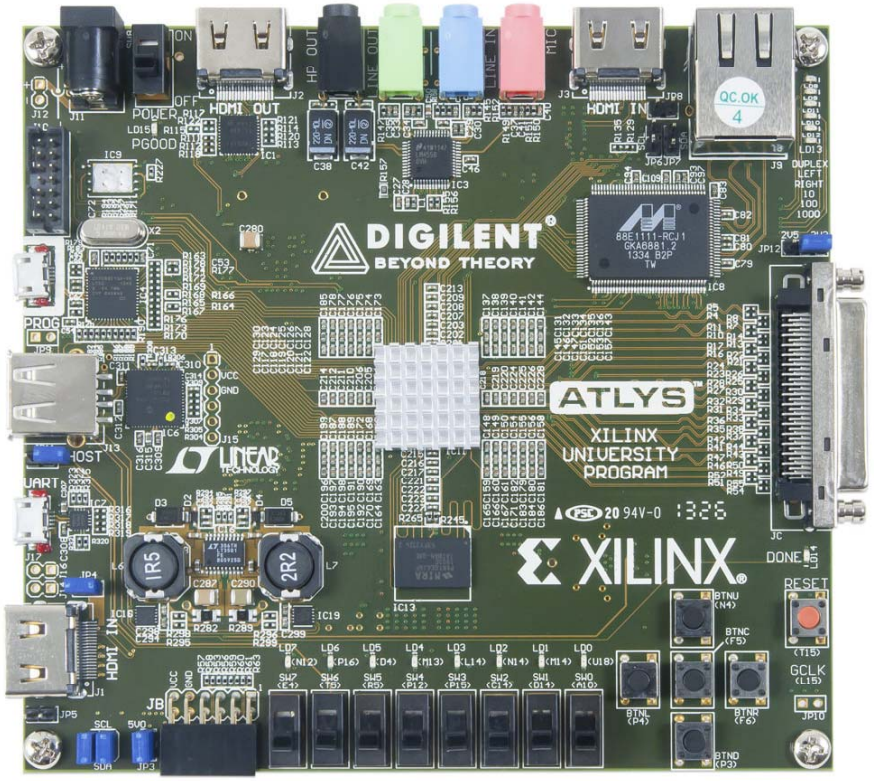
- Configurable Logic Block (CLB)
  - Two identical slices in each CLB
  - Two LUT in each slice
- CLK - Delay Locked Loop (DLL)

Xilinx Spartan-6 Digilent Atlys Board

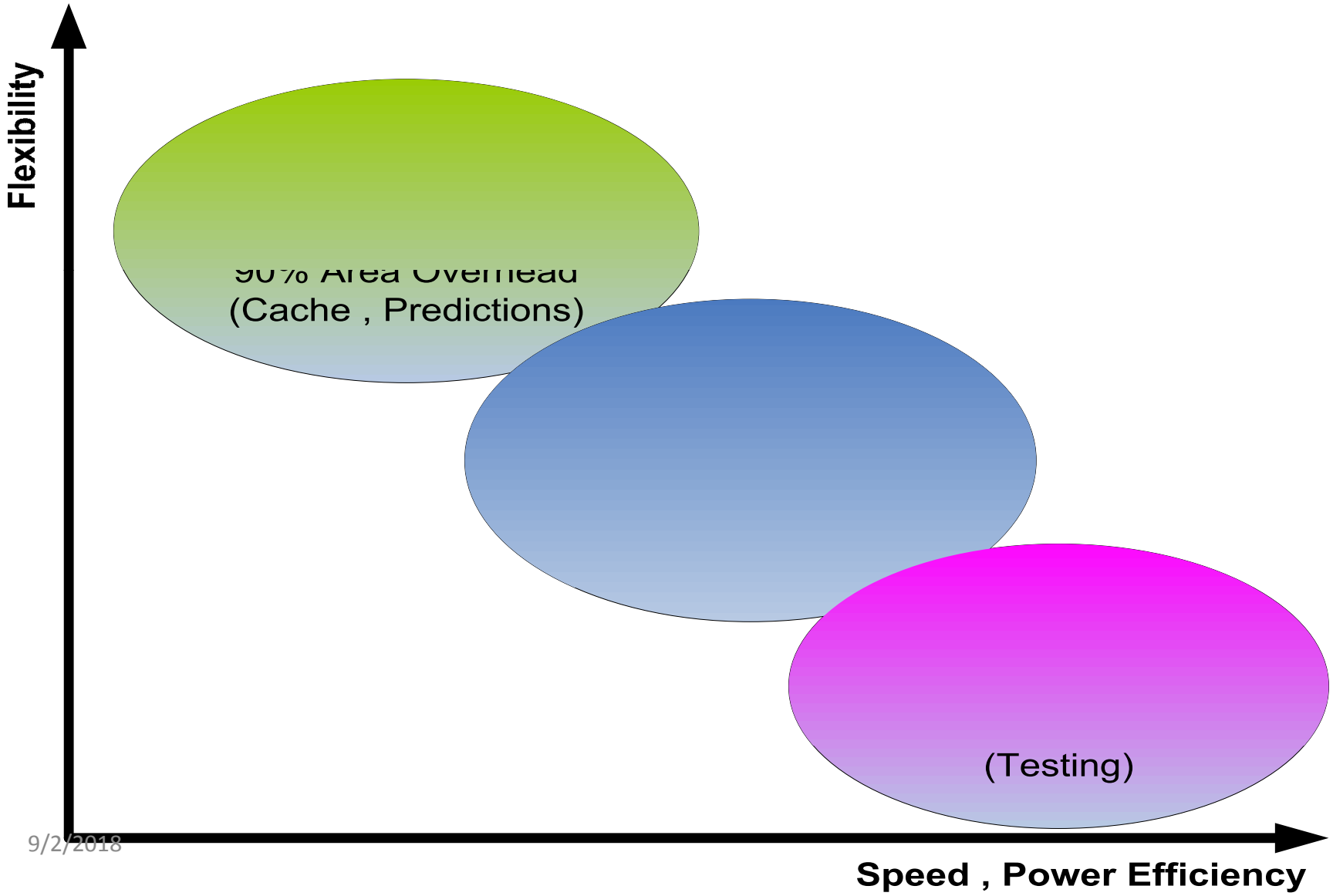


Special FPGA functions

- Internal SRAM
- Embedded Multipliers and DSP blocks
- Embedded logic analyzer
- Embedded CPUs
- High speed I/O (~10GHz)
- DDR/DDRII/DDRIII SDRAM interfaces
- PLLs



Comparison


Flexibility
90% Area Overhead
(Cache , Predictions)
(Testing)
Speed , Power Efficiency

Usages

- Digital designs where ASIC is not commercial
- Reconfigurable systems
- Upgradeable systems
- ASIC prototyping and emulation
- Education

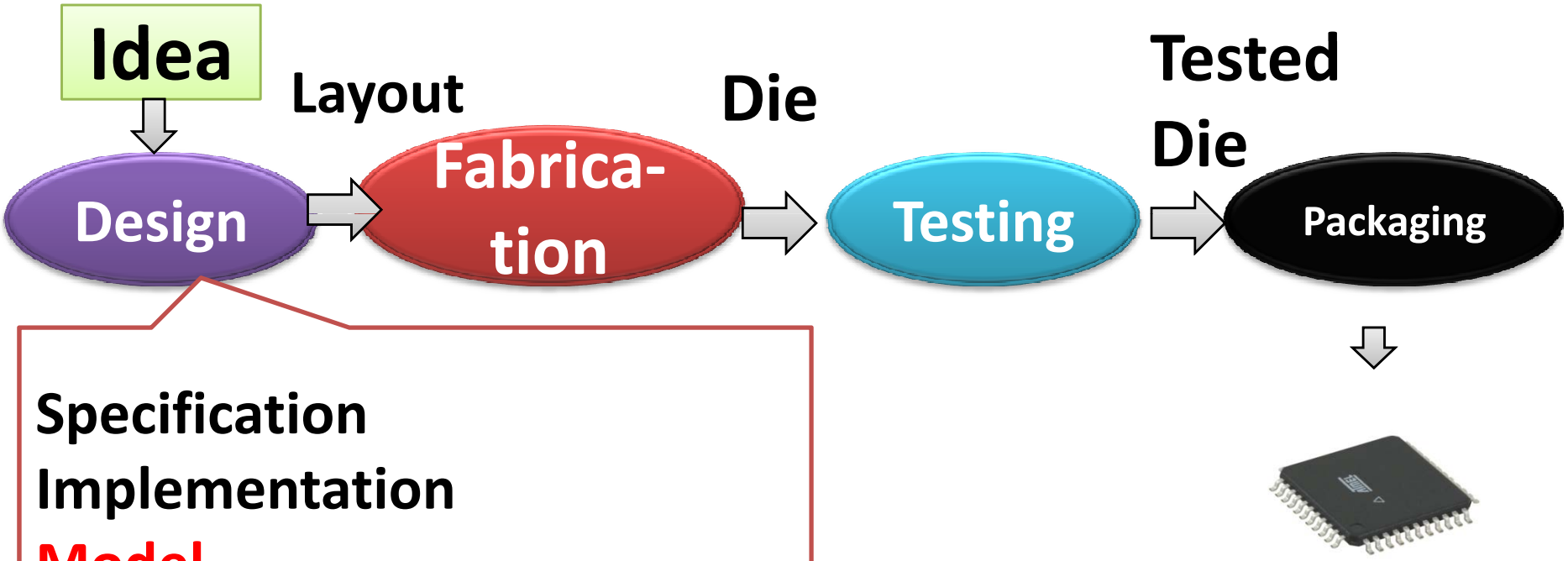
FPGA Manufacturers

- **Xilinx**
- Altera
- Lattice
- Actel

We will work with Xilinx FPGAs : Next Semester

FPGA and ASIC Design Flow

IC Design Process



Hardware/Software Design Flow