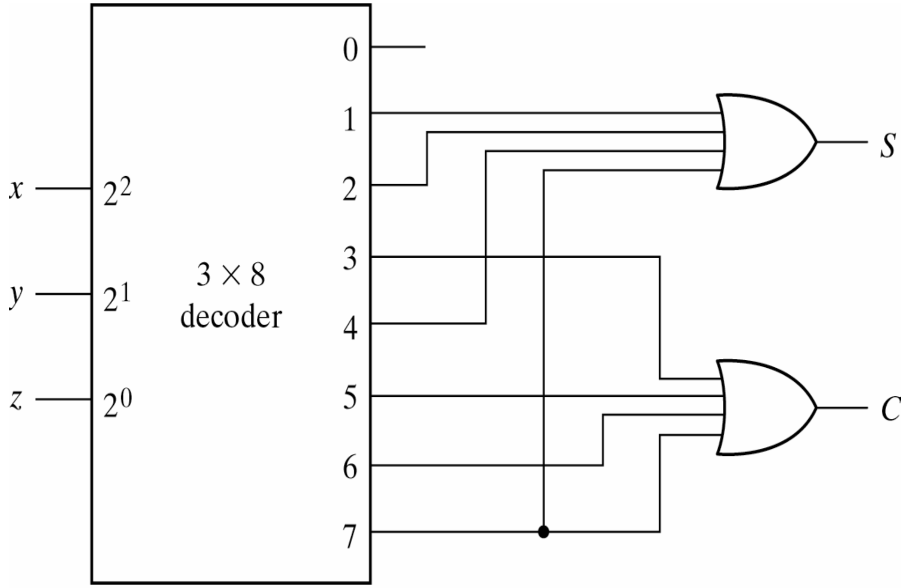
Programmable Logic

Programmable ROM

- In last lecture, we saw how a ROM circuit could be programmed by configuring the interconnection wires.
- In programming the ROM different hardware configurations were implemented, these configurations outputted the stored information.
- In this lecture we will treat the **PROM** not just as a information storage device, but as configurable hardware device.

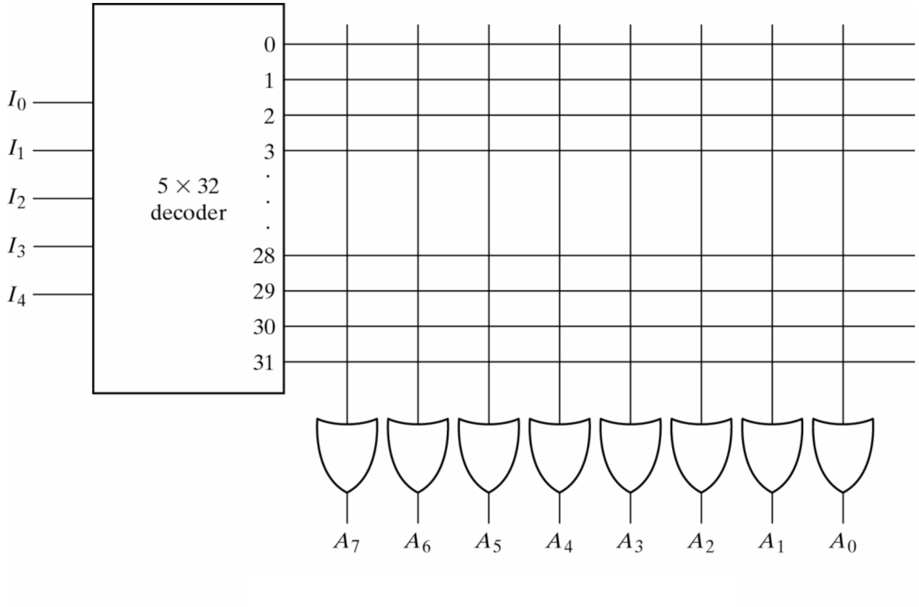
Combinational Circuit Implementation

- In Lecture 5, we learned that a decoder generates the 2^k minterms of k input variables.
- By using OR gates to sum the minterms, we were able to generate any desired combinational circuit.



Combinational Circuit Implementation

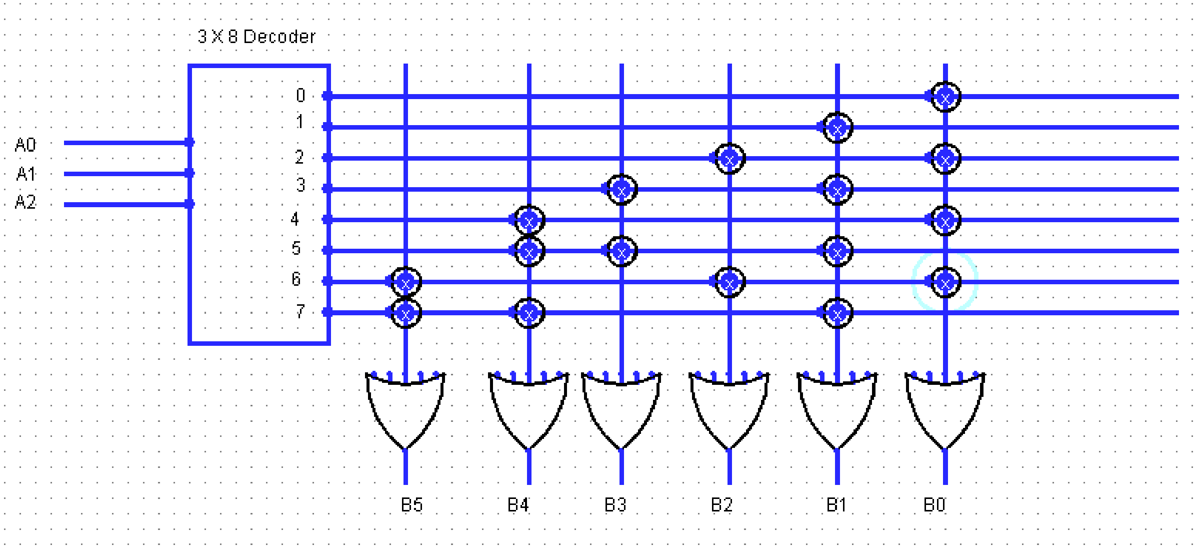
- The ROM includes both a decoder and OR gates.
- By choosing connections for those minterms, the ROM output can be programmed to represent the Boolean function of the output variables.



Combinational Circuit Implementation

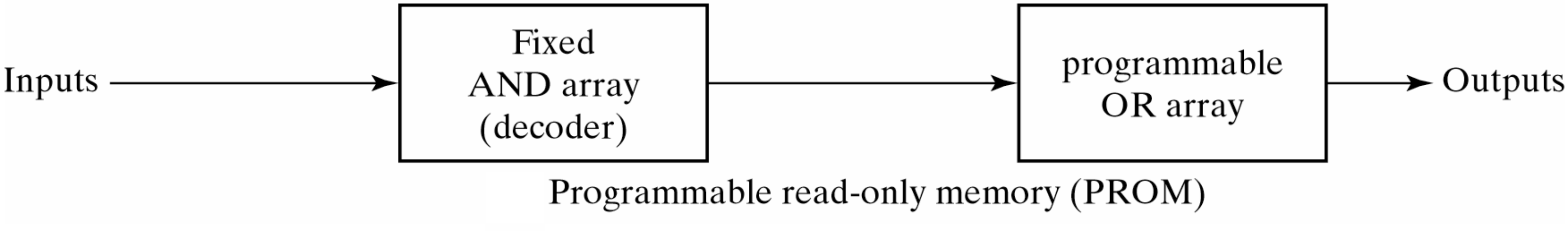
- We could program particular functions, e.g. to implement the function: $y = x^2+1$
- Rather than implementing a multiplier and adder we could use a ROM like a look-up table.

A_2	A_1	A_0	Input	B_5	B_4	B_3	B_2	B_1	B_0	Output
0	0	0	**0**	0	0	0	0	0	1	**1**
0	0	1	**1**	0	0	0	0	1	0	**2**
0	1	0	**2**	0	0	0	1	0	1	**5**
0	1	1	**3**	0	0	1	0	1	0	**10**
1	0	0	**4**	0	1	0	0	0	1	**17**
1	0	1	**5**	0	1	1	0	1	0	**26**
1	1	0	**6**	1	0	0	1	0	1	**37**
1	1	1	**7**	1	1	0	0	1	0	**50**



Programmable ROM

- Now the inputs do not necessarily refer to address lines, but inputs to a decoder, a fixed AND array.
- The outputs of the decoder are then combined using Boolean logic in the programmable OR array.
- The particular logical functions can be implemented in a completely configurable way.

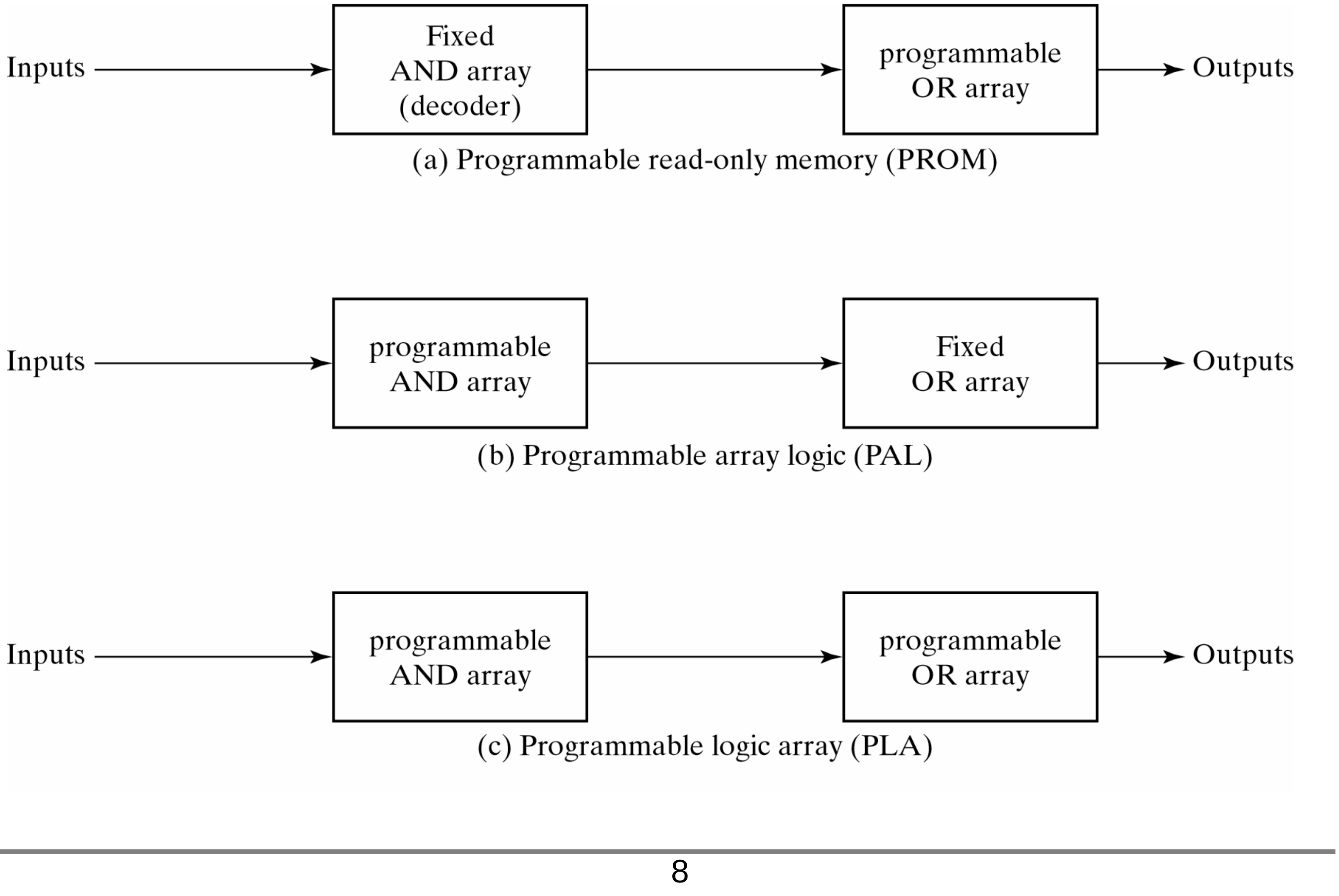


Programmable read-only memory (PROM)

Programmable Logic

- Other programmable logic devices include
  - the **PAL** (Programmable Array Logic) device which has a programmable AND array and fixed OR array and
  - the **PLA** (Programmable Logic Array) device which has programmable AND array and a programmable OR array.

Programmable Logic



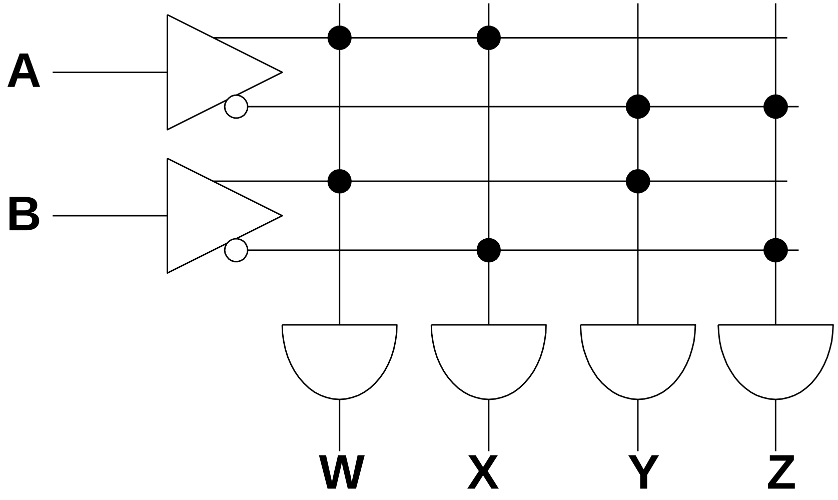
(a) Programmable read-only memory (PROM)

(b) Programmable array logic (PAL)

(c) Programmable logic array (PLA)

Programmable Array Logic

- The programmable AND array is used to implement product terms.
- Here the A and B inputs are combined to create four Boolean outputs.
- Note that the AND gates have four inputs but the convention is to draw only one line.

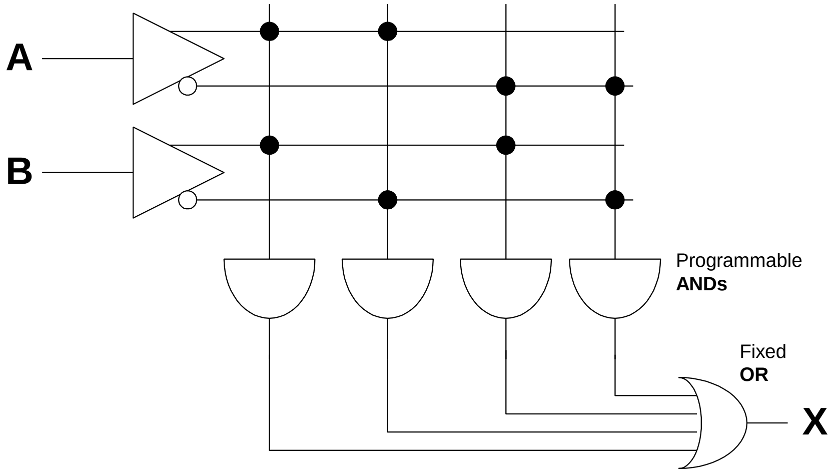


$$W = AB \quad X = A\overline{B}$$

$$Y = \overline{A}B \quad Z = \overline{A}\,\overline{B}$$

Programmable Array Logic

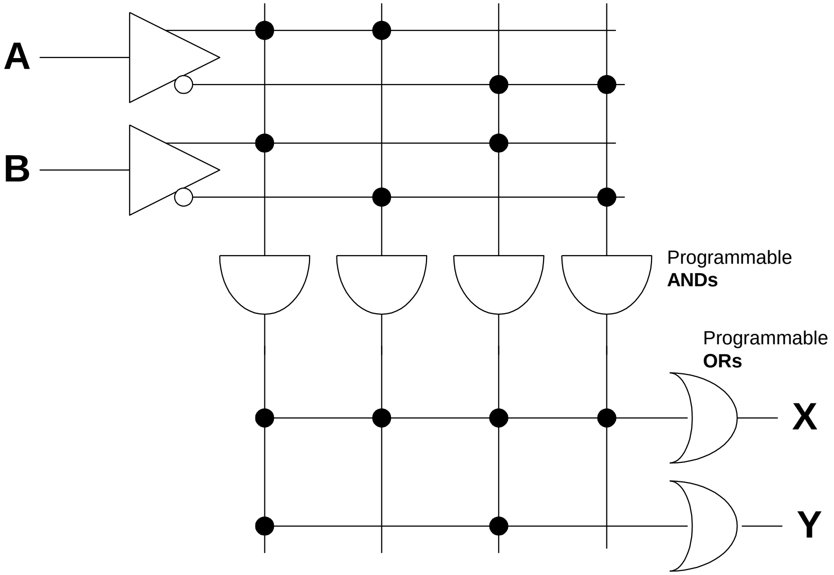
- In a PAL device the product terms are combined using a fixed OR gate.
- A resulting output is a sum of products.



$$X = AB + A\overline{B} + \overline{A}B + \overline{A}\,\overline{B}$$

Programmable Logic Array

- In a PLA device is similar to the PAL but now the product terms are combined using a programmable OR array.
- The resulting outputs X and Y are two different sum of products expressions.
- The PLA is more flexible than the PAL but more difficult to program.

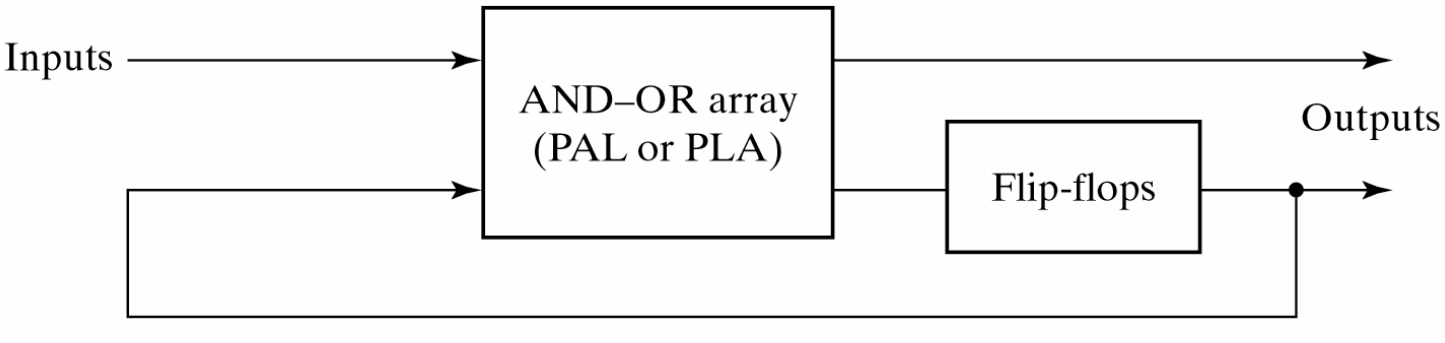


$$X = AB + A\overline{B} + \overline{A}B + \overline{A}\overline{B}$$

$$Y = AB + \overline{A}B$$

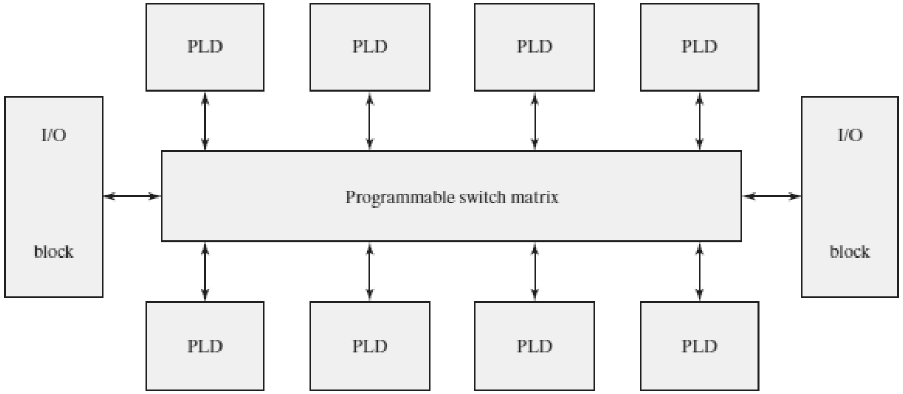
Sequential Programmable Logic

- The PAL and PLA devices consist only of gates and so can only be used to implement combinational logic.
- With the addition of flip-flop storage devices it is possible to create sequential programmable logic devices.
- Examples of sequential programmable devices include
  - CPLD (complex programmable logic device)
  - FPGA (field programmable logic devices)



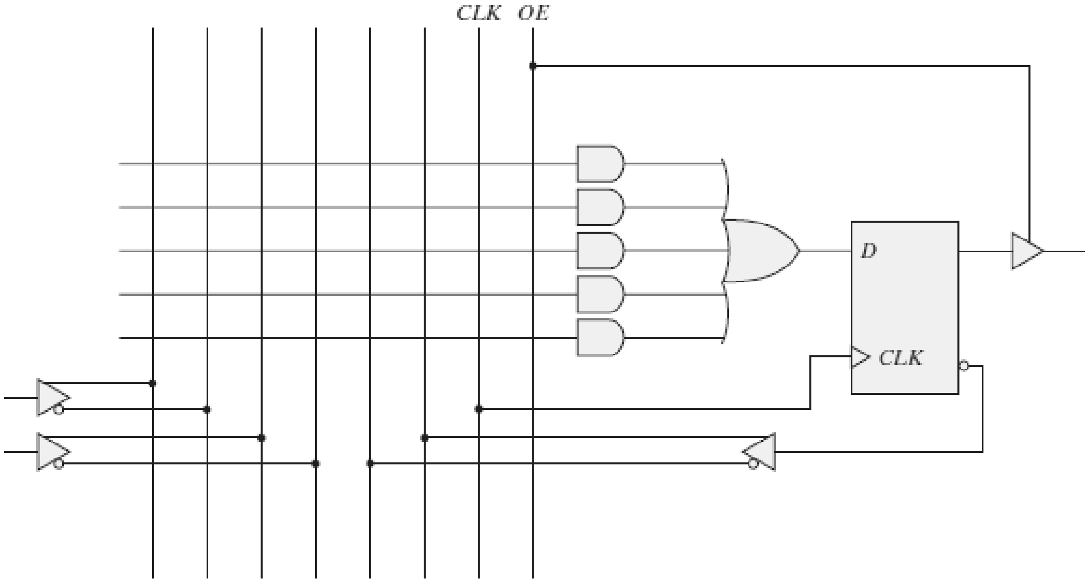
Complex Programmable Logic Device

- CPLD consists of multiple PLDs interconnected through a programmable switch matrix.
- The input/output (I/O) blocks provide the connections to the IC pins. Each I/O pin is driven by a three-state buffer and can be programmed to act as input or output.
- The switch matrix receives inputs from the I/O block and directs them to the individual macrocells in PLDs. Selected outputs from macrocells are sent to the outputs as needed.



Complex Programmable Logic Device

- Each PLD typically contains 8 to 16 macrocells.
- Basic macrocell contains a sum-of-products combinational logic function and an optional flip-flop.

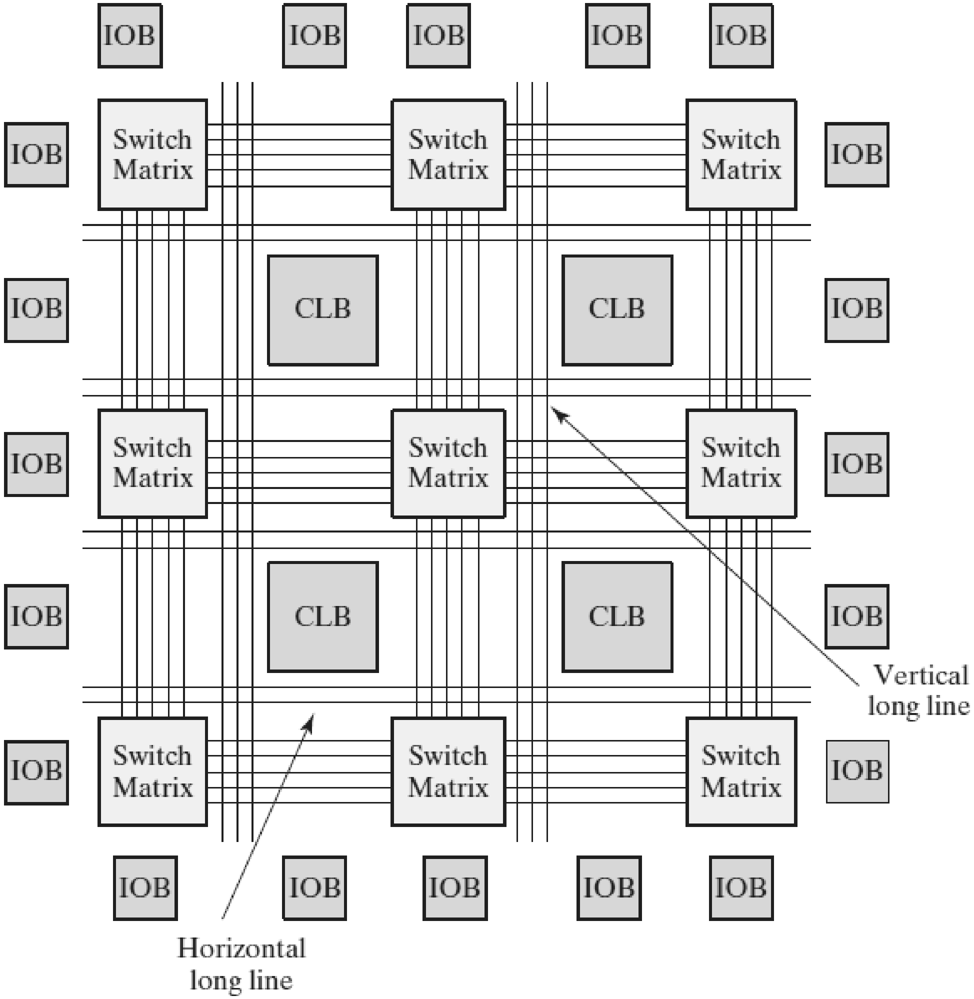


Basic Macrocell Logic

Field Programmable Gate Array

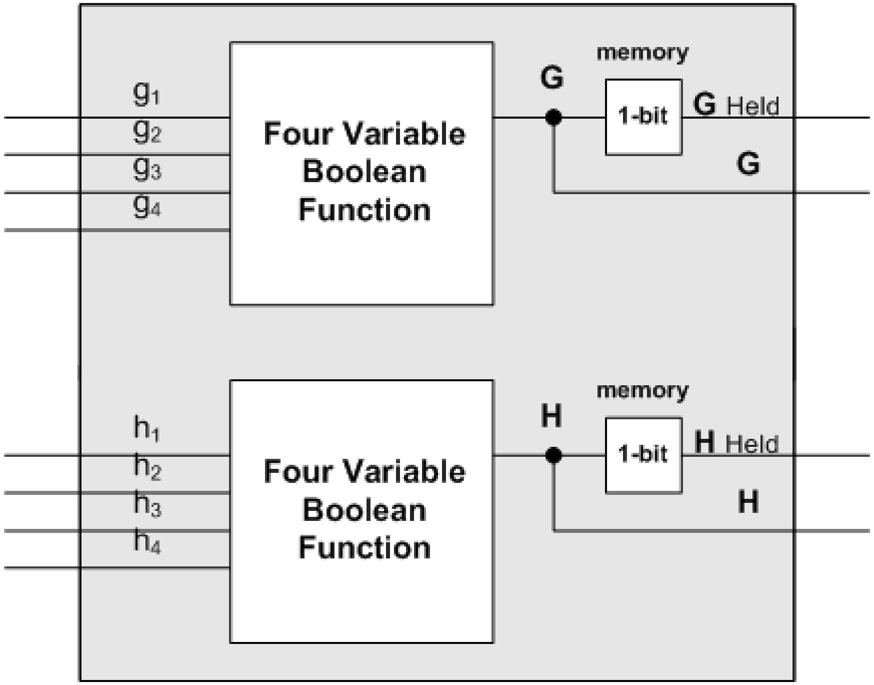
- For a large scale digital design, a field programmable gate array (FPGA) is normally used.
- The basic component of FPGA is the **gate array**, which consists of a pattern of gates, fabricated in a area of silicon.
- A typical FPGA consists of hundreds or thousands of **configurable/programmable logic blocks (CLBs/PLBs)**, surrounds by **programmable input/output blocks (IOBs)** and connected together via **programmable interconnections (Switch Matrix)**.

Basic Xilinx FPGA Architecture



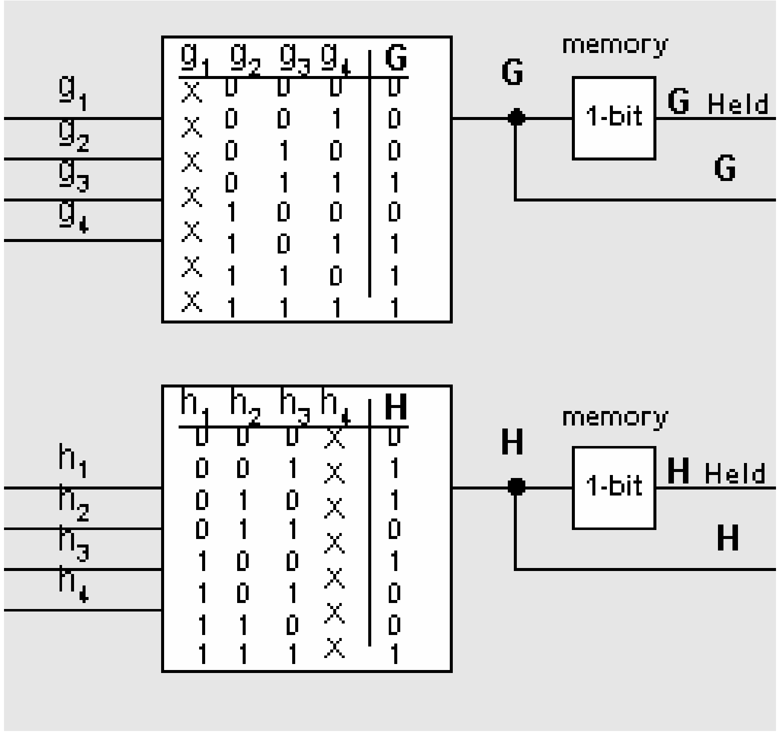
Configurable Logic Blocks

- Each CLB normally consists of a programmable lookup table, multiplexers, registers/flip-flops, and paths for control signals.
- Each logic block can be programmed to implement any Boolean function of four variables.
- The functional value it generates could be either directly outputted to another CLB or can be stored for later use in a 1-bit memory element.



Configurable Logic Blocks

- E.g., FPGA technology enables the designer to implement the full adder with one CLB.
- Carry and Sum functions could each be implemented with one logic block.
- NOTE : As the Carry and Sum functions are both 3-variable functions one input to each block is unused.

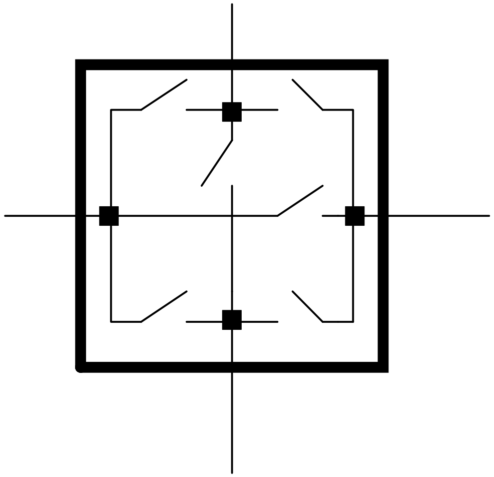


g_1	g_2	g_3	g_4	G
X	0	0	0	0
X	0	0	1	0
X	0	1	0	0
X	0	1	1	1
X	1	0	0	0
X	1	0	1	1
X	1	1	0	1
X	1	1	1	1

h_1	h_2	h_3	h_4	H
0	0	0	X	0
0	0	1	X	1
0	1	0	X	1
0	1	1	X	0
1	0	0	X	1
1	0	1	X	0
1	1	0	X	0
1	1	1	X	1

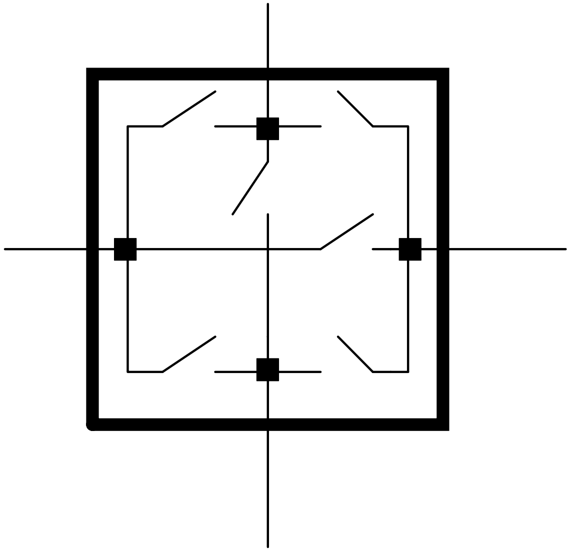
Switch Matrix

- Within an FPGA, the programmable interconnect contains a set of horizontal and vertical wires that can be interconnected through a Switch Matrix (SM)
- In each n x n Switch Matrix there will be m ≤ n x n interconnect points which can be used to connect the horizontal and vertical wires in several different ways.



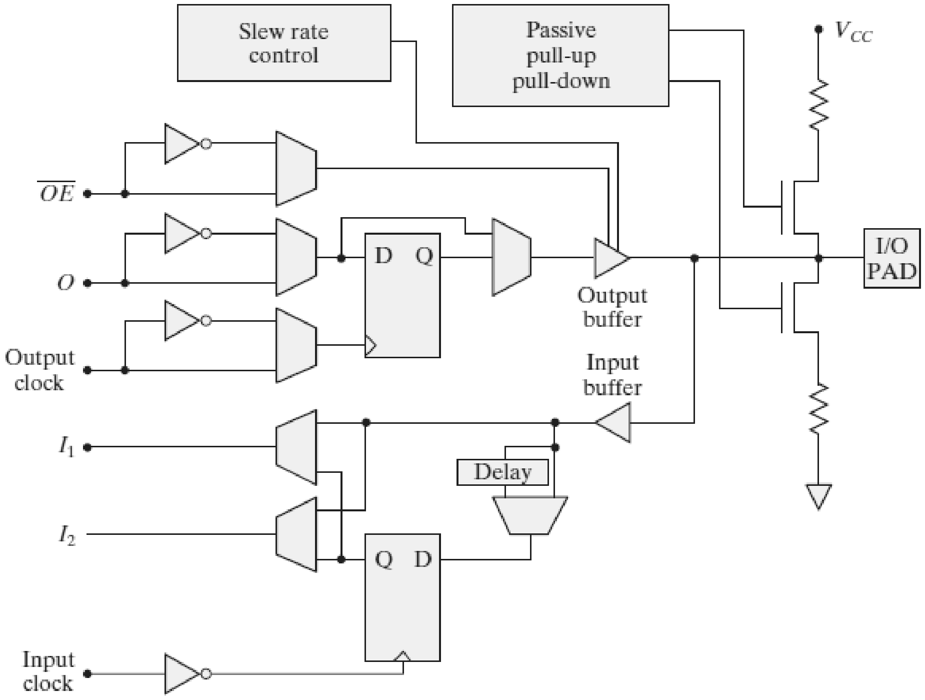
Switch Matrix

- Each interconnect point contains 6 switches, which can be programmed to be open or closed.
- 2^6 or 64 possible different connections between these 4 wires.
- The interconnect points can also be used to connect CLB's to horizontal and vertical wires.



I/O Block (IOB)

- IOB provides interface between the package pins and CLBs
- Each programmable I/O pin has a programmable IOB having buffers from compatibility with TTL and CMOS signal levels.
- It can be used as an input, an output, or a bidirectional port.



Xilinx XC4000 series IOB

FPGA Development

- FPGAs originally began as competitors to CPLDs.
- As size, capabilities, and speed increased, they began to take over larger and larger functions to the state where some are marketed as full systems on chips.
- Particularly with the introduction of dedicated multipliers into FPGA architectures in the late 1990s, FPGAs are increasingly used in conventional high performance computing applications where computational kernels such as FFT or Convolution are performed on the FPGA instead of a microprocessor.
- Major FPGA Vendors
  - Xilinx, Altera

Xilinx FPGA Families

- **Old families**
  - XC3000, XC4000, XC5200
  - Old 0.5µm, 0.35µm and 0.25µm technology.
- **Low Cost Family**
  - Spartan/XL – derived from XC4000
  - Spartan-II – derived from Virtex
  - Spartan-IIE – derived from Virtex-E
  - Spartan-3, Spartan 3E, Spartan 3L
- **High-performance families**
  - Virtex (220 nm)
  - Virtex-E, Virtex-EM (180 nm)
  - Virtex-II, Virtex-II PRO (130 nm)
  - Virtex-4 (90 nm)
  - Virtex 5 (65 nm)
  - Virtex-6 (40 nm)