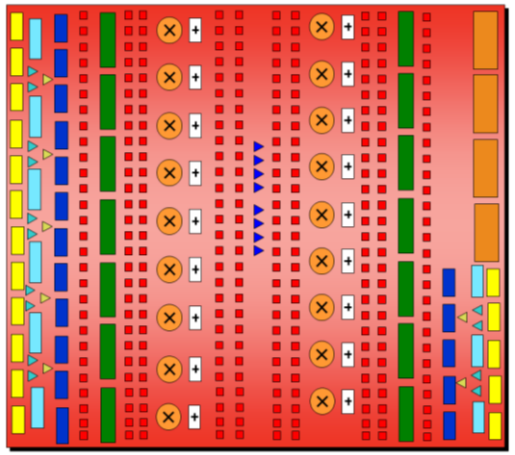
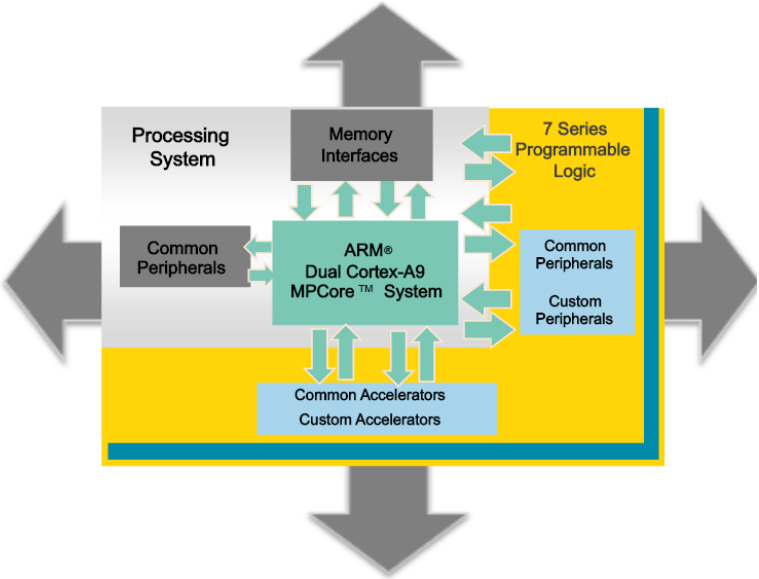
DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

A2A
Algorithms to Architecture

ECE 270: Embedded Logic Design



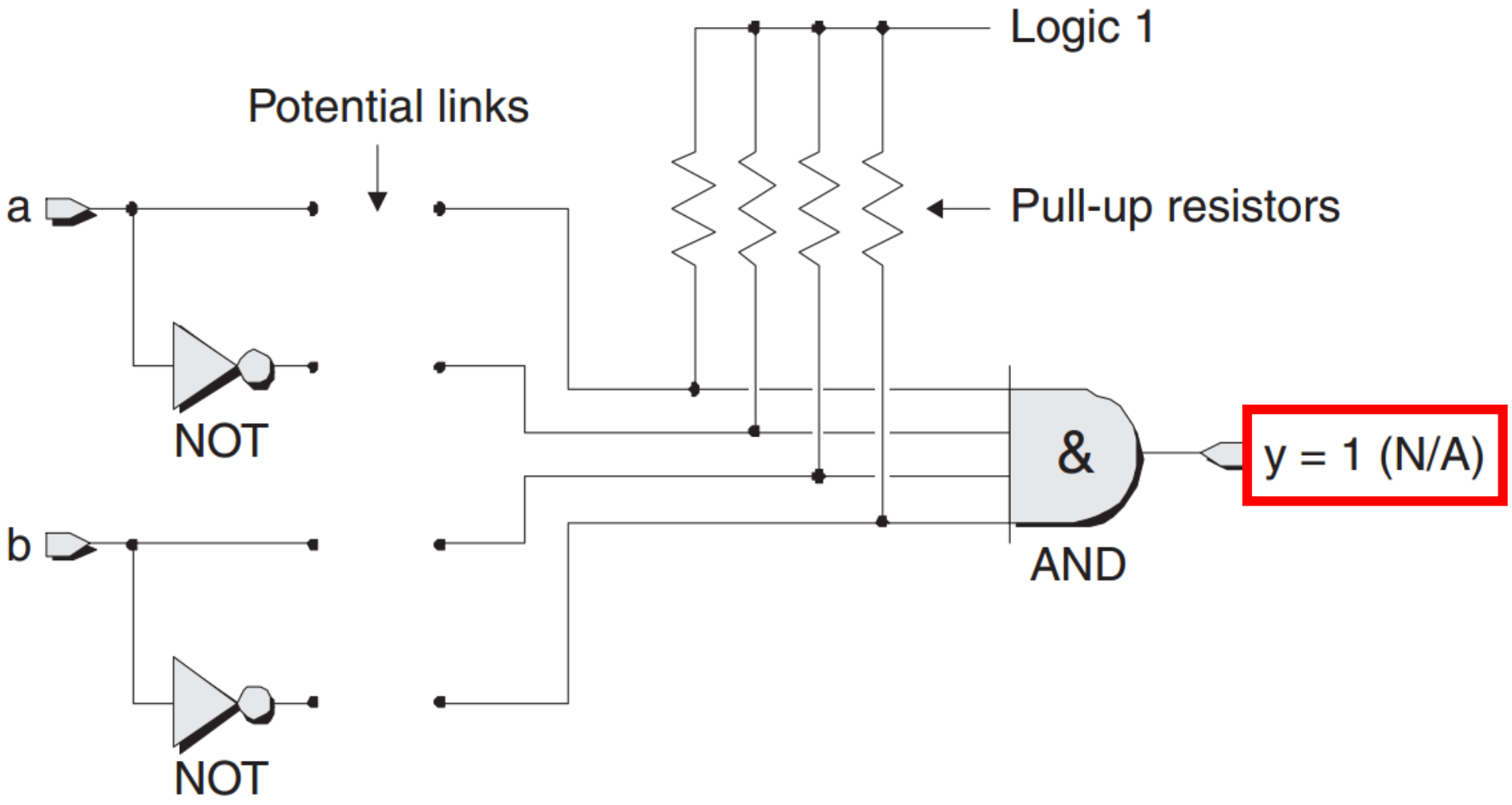


Evolution of Programmable Logic Device (PLD)

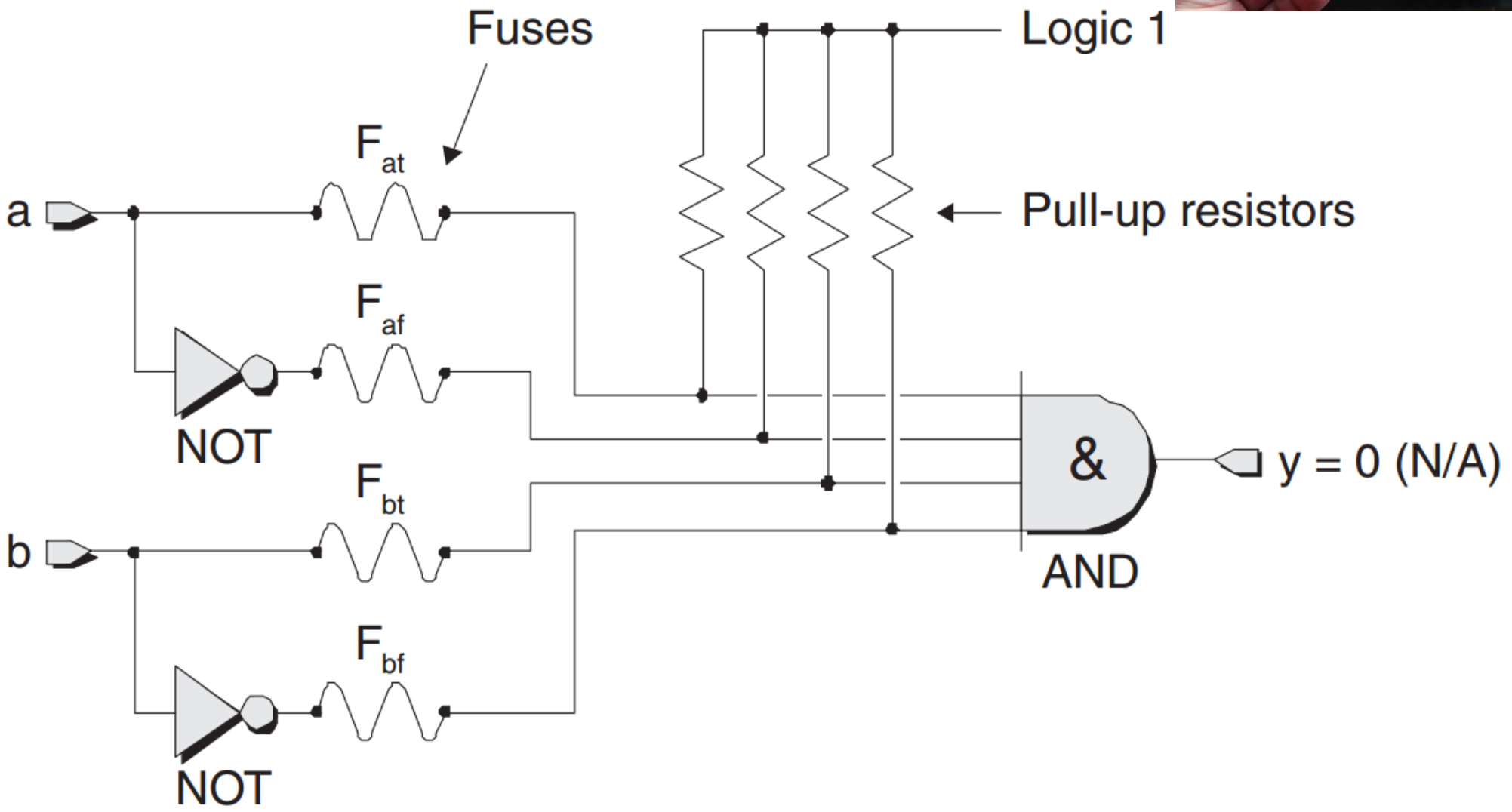
PLD

- **Programmable logic devices (PLD)**: Devices whose internal architecture is predetermined by manufacturer but which are created in such a way that they can be configured in the field to perform variety of functions
- How to make device field programmable?

PLD

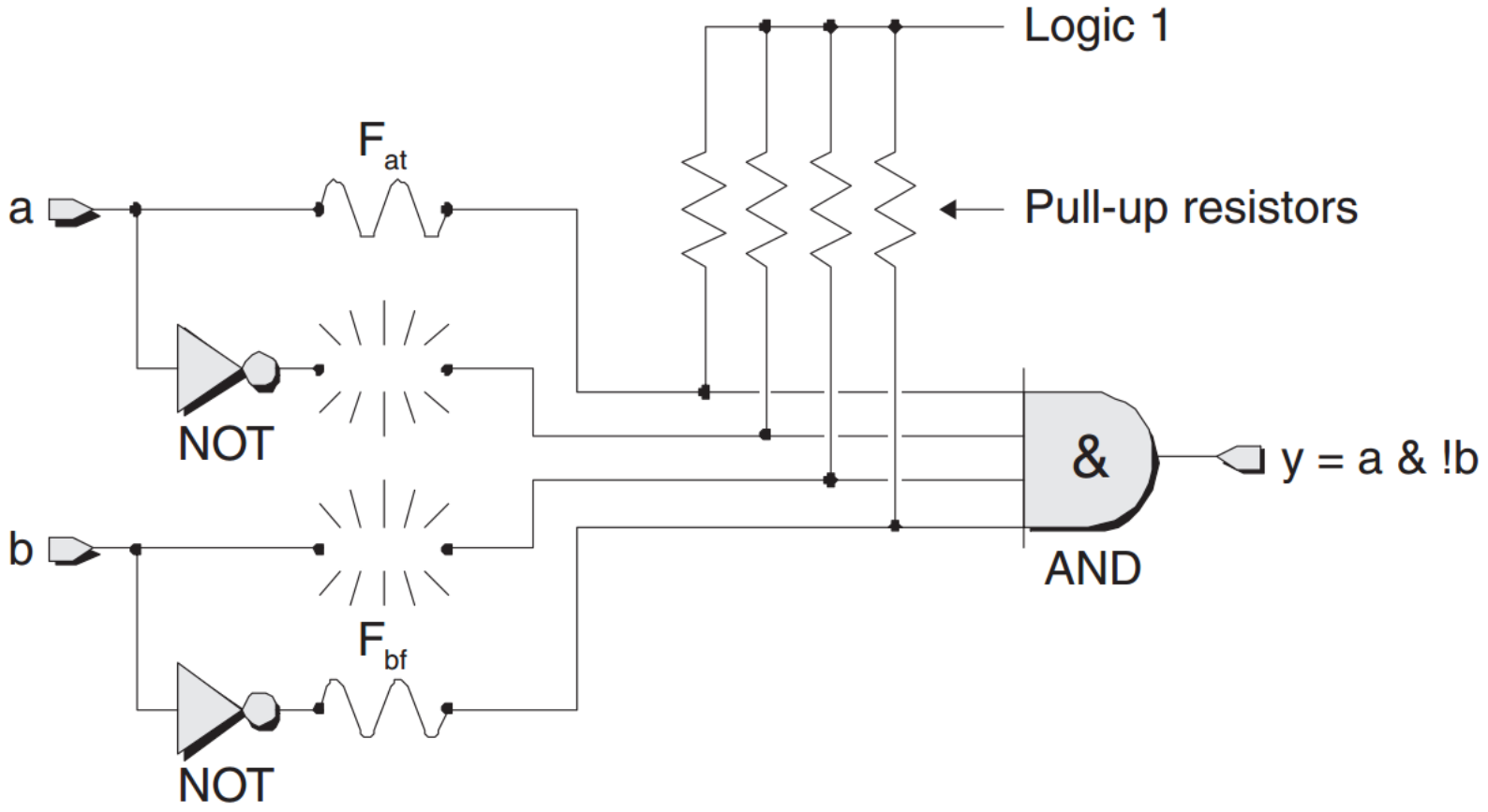

Logic 1
Potential links
Pull-up resistors
a
NOT
b
NOT
&
AND
y = 1 (N/A)

PLD (Fusible Link Technology)



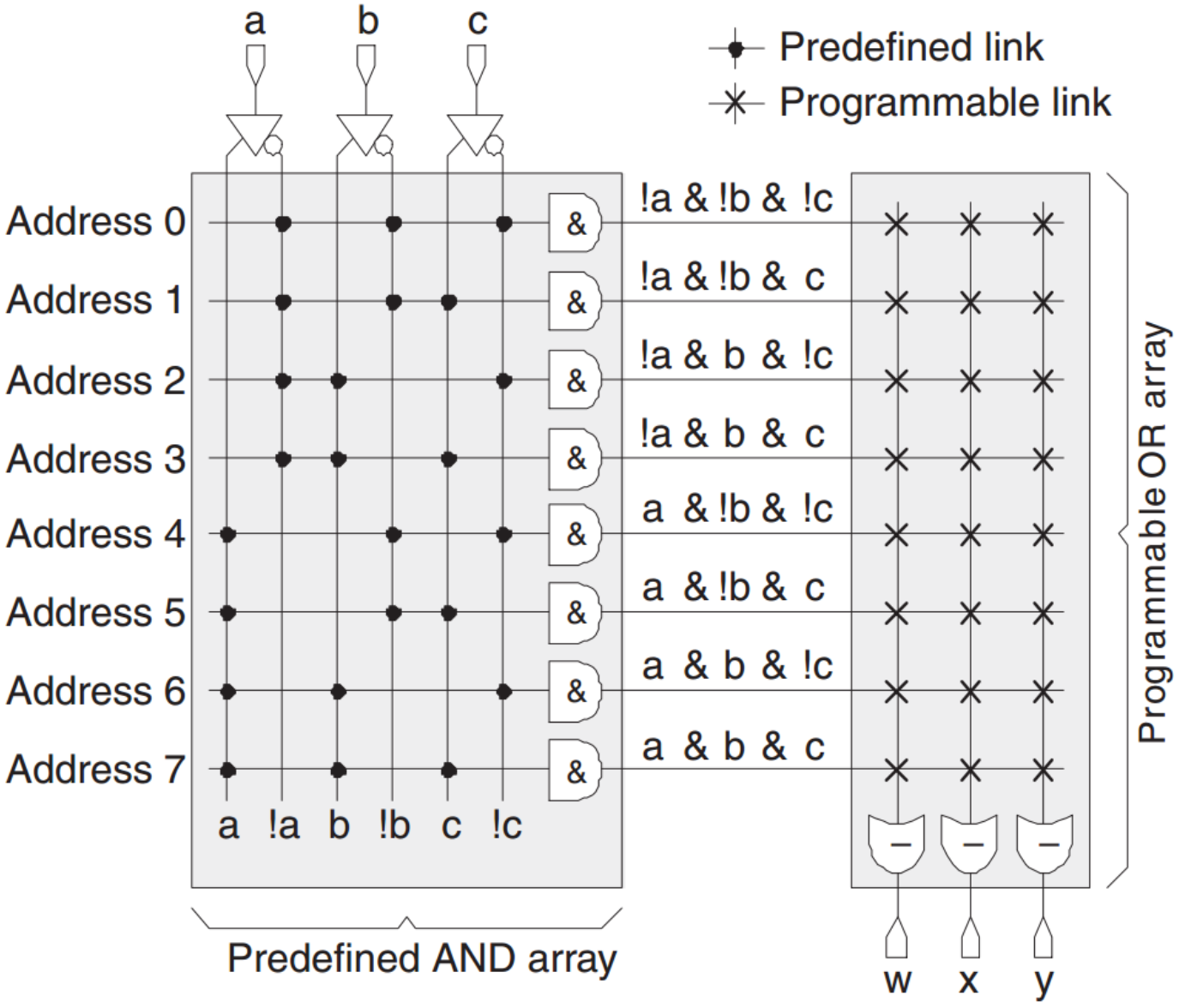
PLD

- Programming the device/blowing the fuses/burning the device
- One time programmable
- Antifuse technology

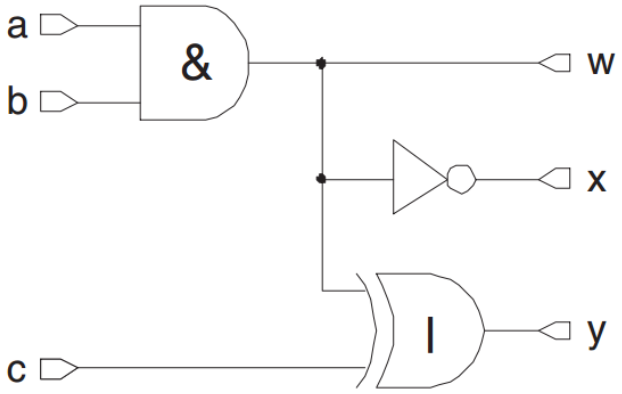


PLD

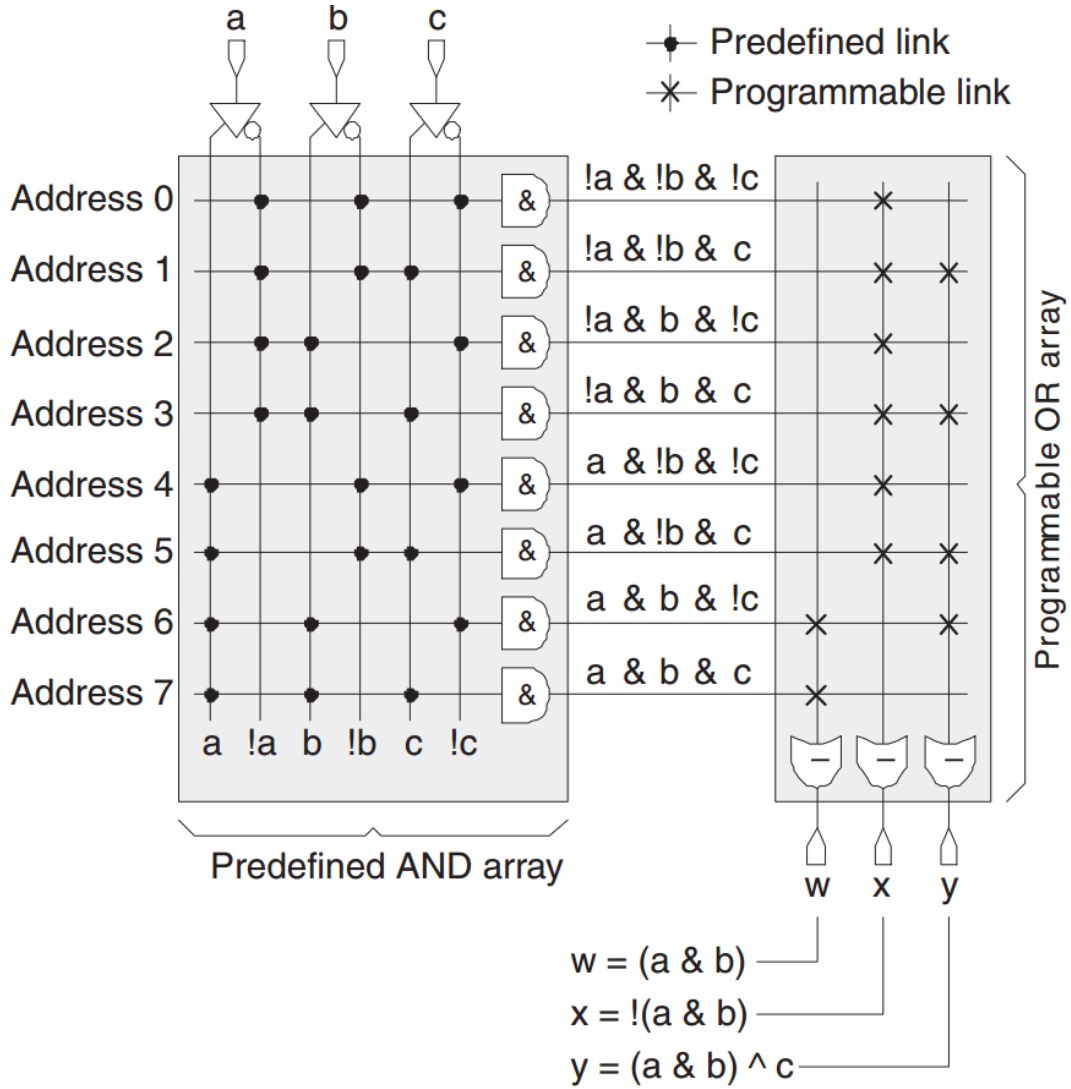
- PROM: Programmable read only memory (1970)



PROM

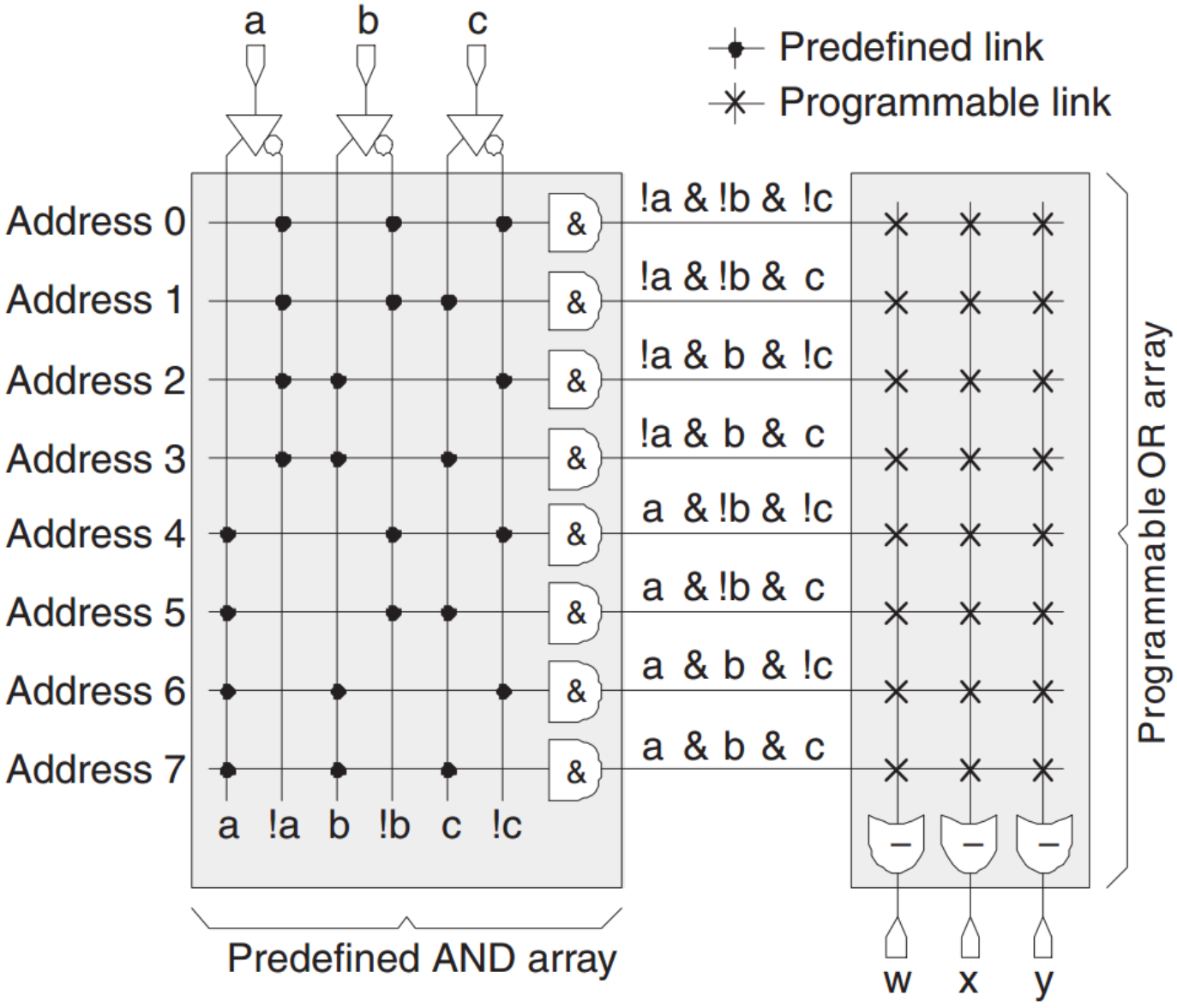


a	b	c	w	x	y
0	0	0	0	1	0
0	0	1	0	1	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	0	1	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	0	0



PLD

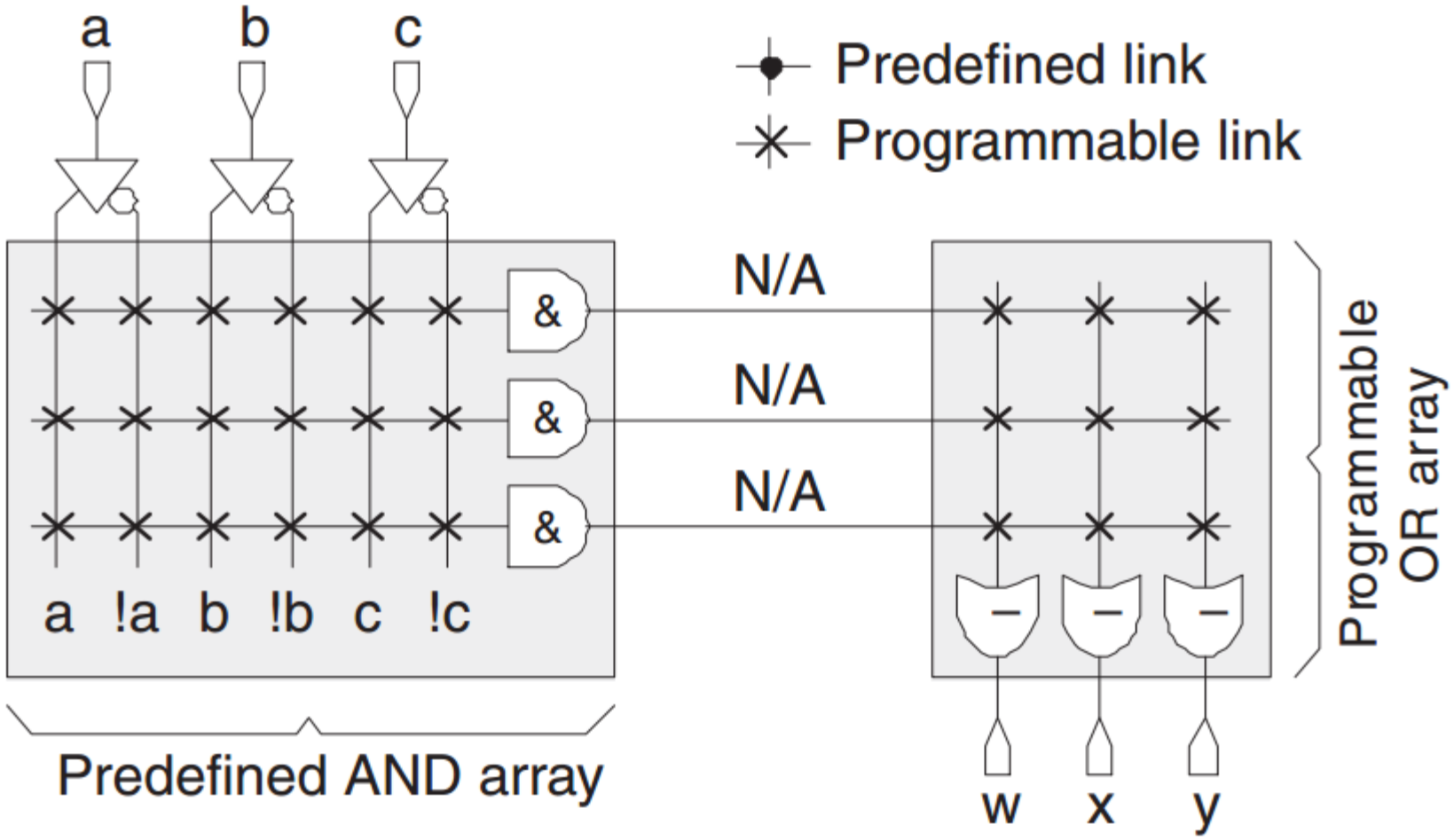
- PROM: Programmable read only memory (1970)
- One time programmable
- Single PROM instead of multiple chips: smaller, lighter, cheaper and less prone to errors (fewer soldier joints), easy to identify errors or correct errors



PLD

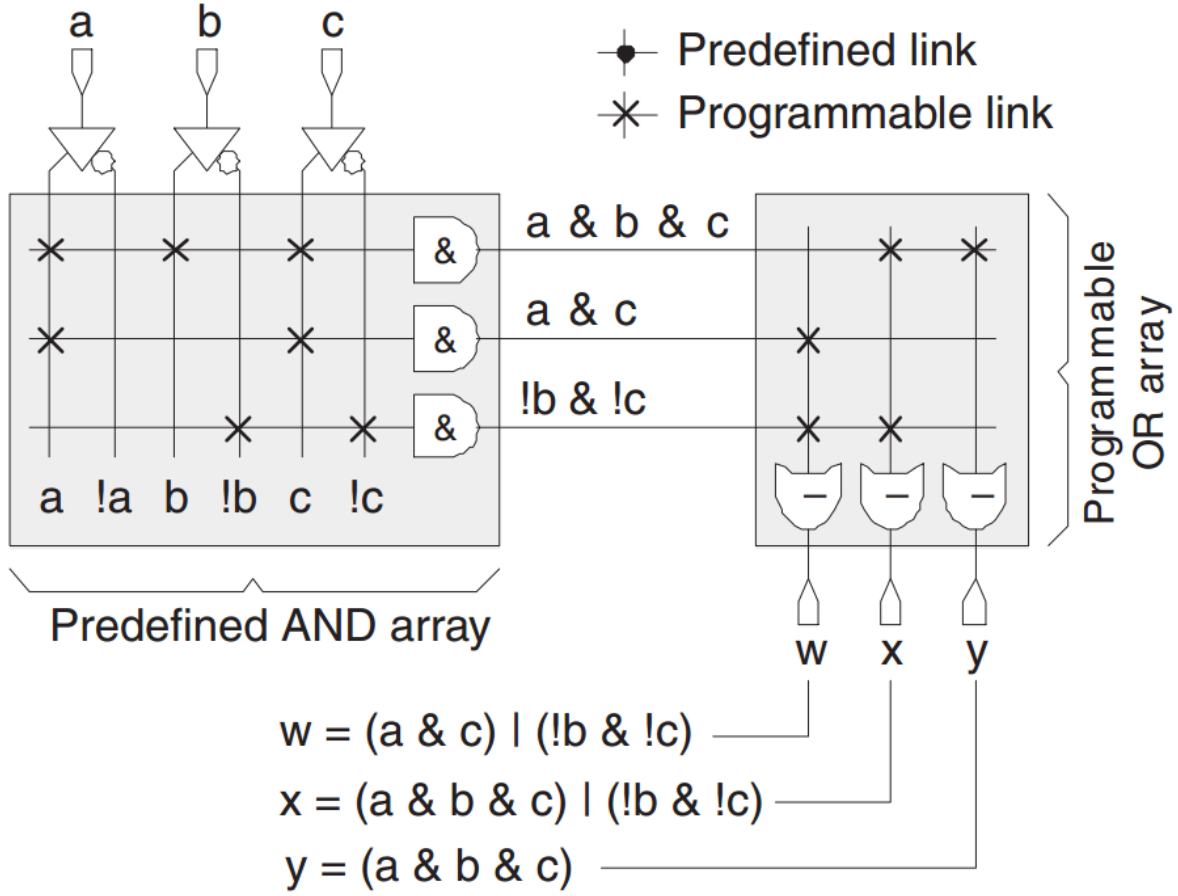
- PROM: Programmable read only memory (1970)
- Designed for use as memories to store computer programs and constant data values.
- Also useful for implementation simple logical function such as LUT and state machines
- EPROM (Intel 1971): Erasable PROM: Can be erased (UV rays) and re-programmed… smaller in size than fusible link devices
- Expensive and long erasure time (few minutes)
- Erasing process becomes complex as transistor density increases
- EEPROM: Electrically EPROM (larger than EPROM)

PLA: Programmable Logic Arrays (1975)



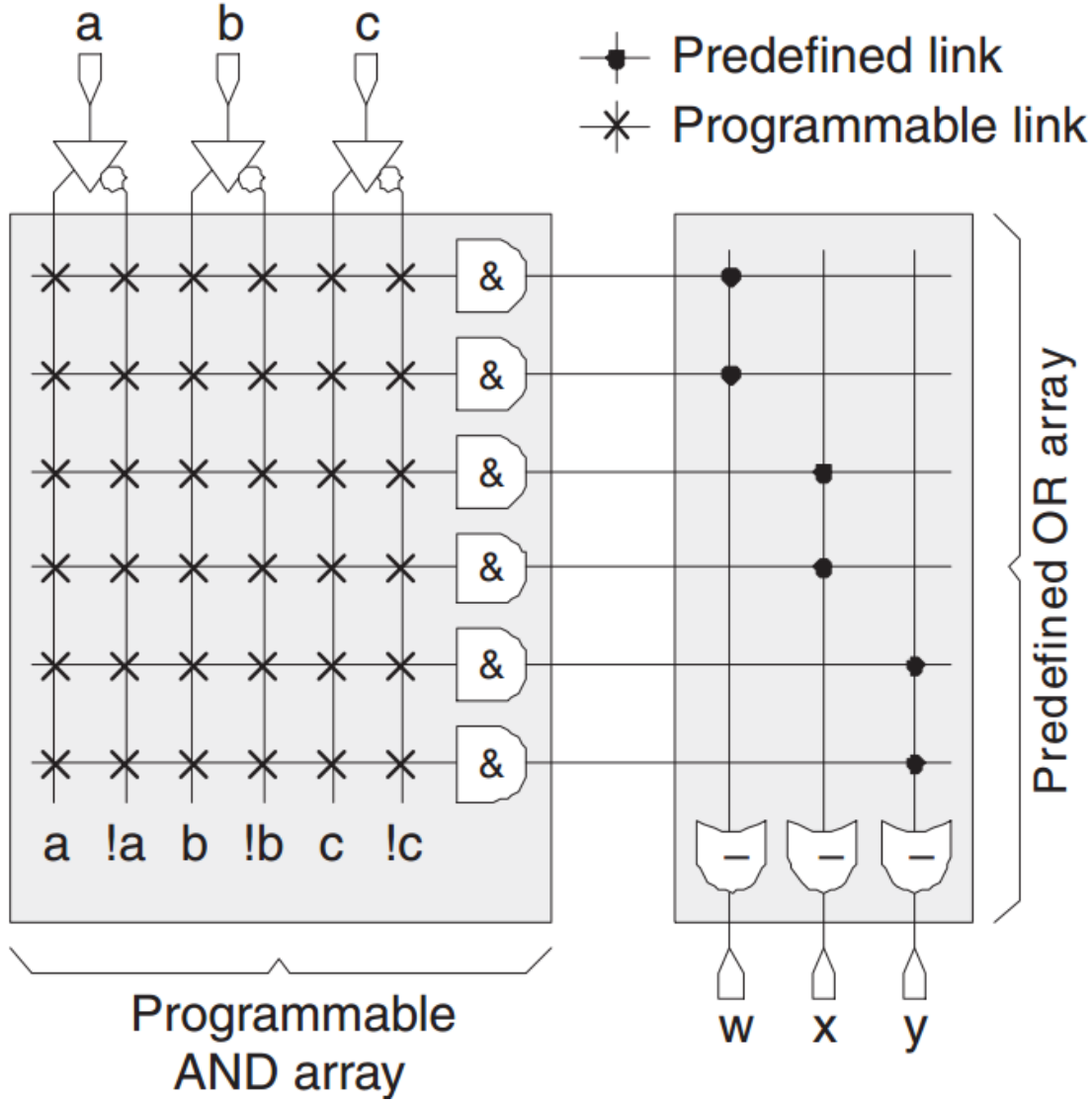
PLA: Programmable Logic Arrays (PLA)

- Not very popular because SOP was the most popular approach for representation of digital circuits
- Variations: AND-NOR, NAND-OR and so on…
- Slower than PROM: signal takes long time to pass through programmable link than predefined link



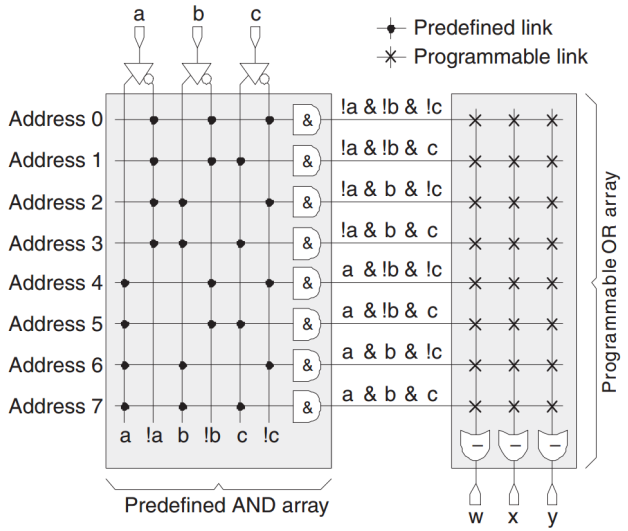
PAL: Programmable Array Logic (PAL)

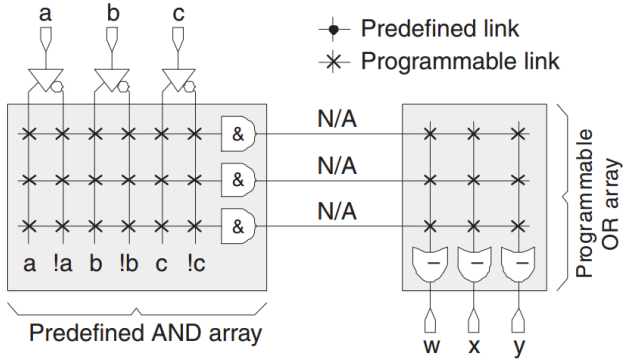
- Faster than PLA
- Limited applications due to fewer number of AND terms

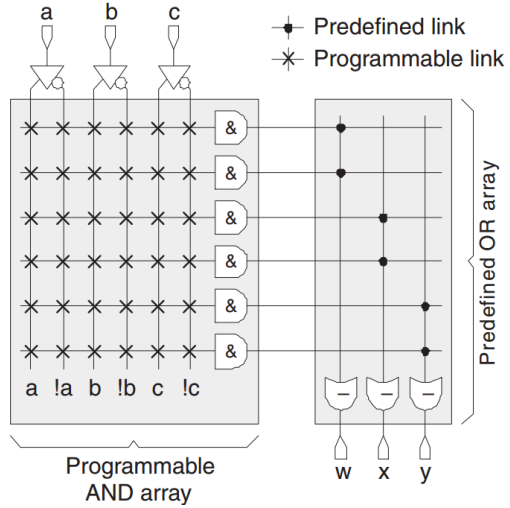


PLD

- **Programmable logic devices (PLD)**: Devices whose internal architecture is predetermined by manufacturer but which are created in such a way that they can be configured in the field to perform variety of functions

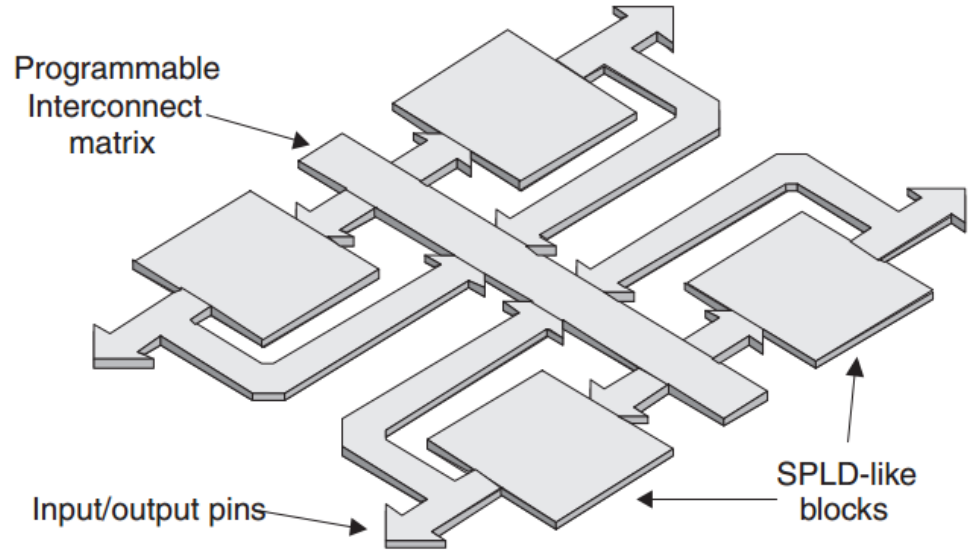






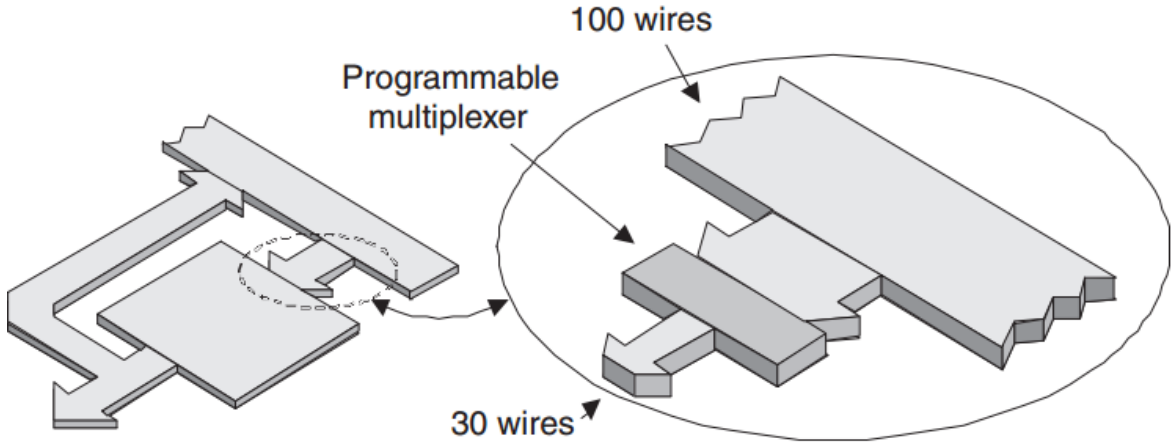
PAL: Complex PLDs (CPLD) 1984

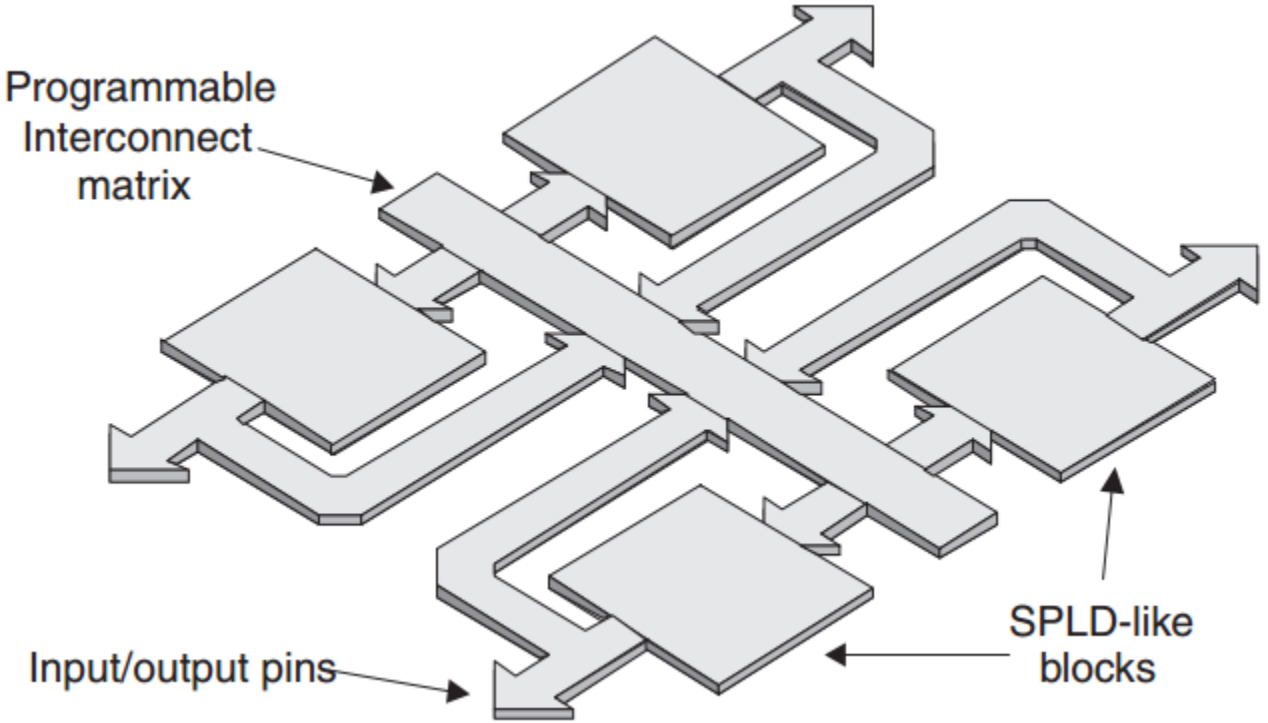
- Need for bigger (functional capability), smaller (size), faster and cheaper technology
- MegaPAL: interconnection of four PALs -> high power consumption
- 1984: Altera introduced **CPLD** using CMOS (high density, low power) and EPROM/EEPROM (programmability)
- Can implement complex functions, smaller size, cheaper and faster



PAL: Complex PLDs

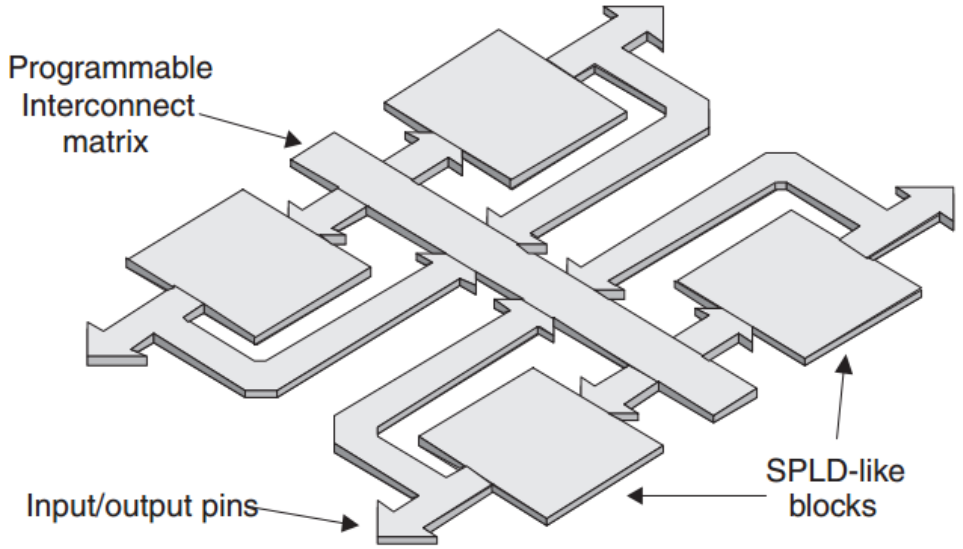
- Novelty: Central interconnect
- In addition to programming SPLD (PAL), connections can also be programmed using programmable interconnect matrix
- This leads to increase in the complexity of software tools

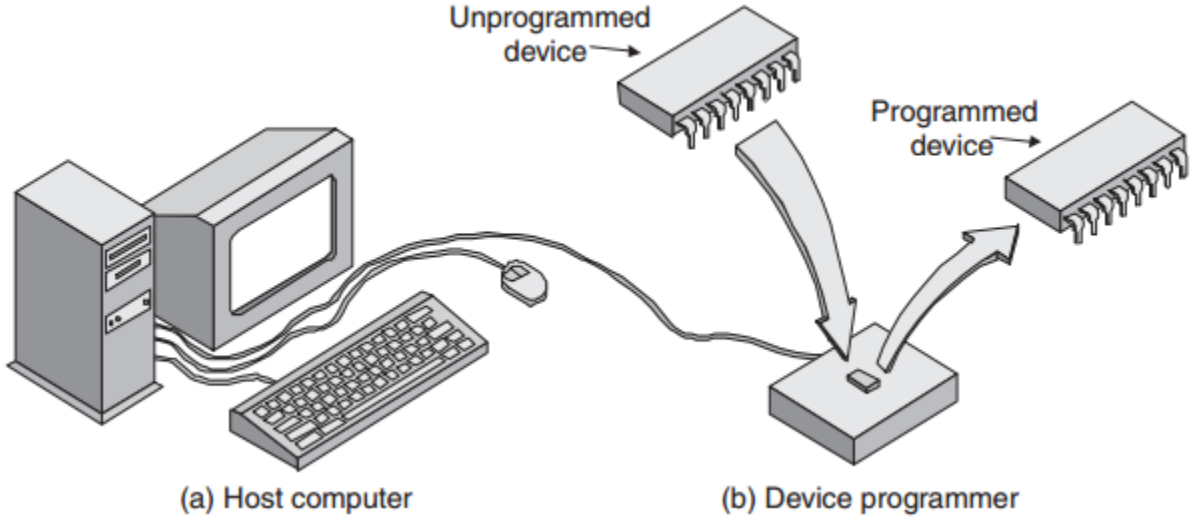




Programming CPLDs

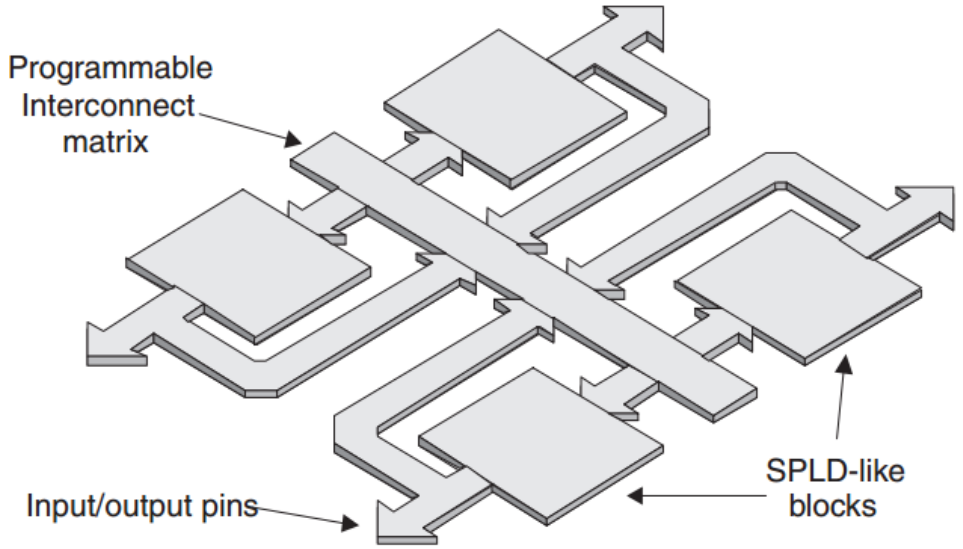
- Text file generation for programming (remember fuses) -> Device programmer
- Deep knowledge of architecture and file format used by device programmer
- Prone to error, time consuming process and difficult to locate error
- Lead to development of HDL (functional description to text file for device programmer)





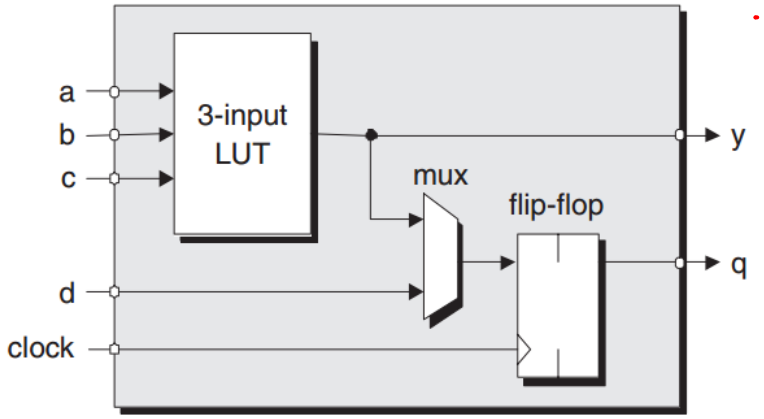
Why FPGAs?

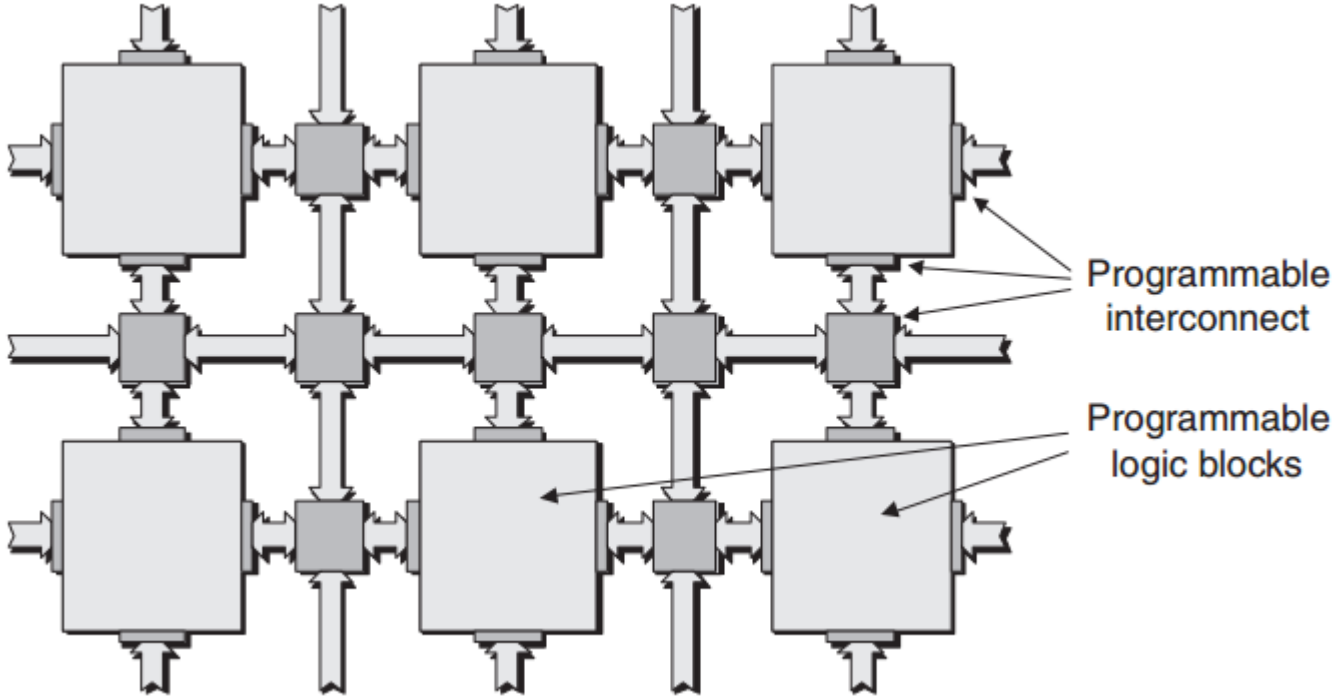
- **CPLD:** Based on EEPROM and hence, non-volatile
- Signals from a Logic Block can be connected to only the neighbouring Logic blocks (In FPGAs, they can travel great distances through the interconnect)
- Can not support large designs like ASIC



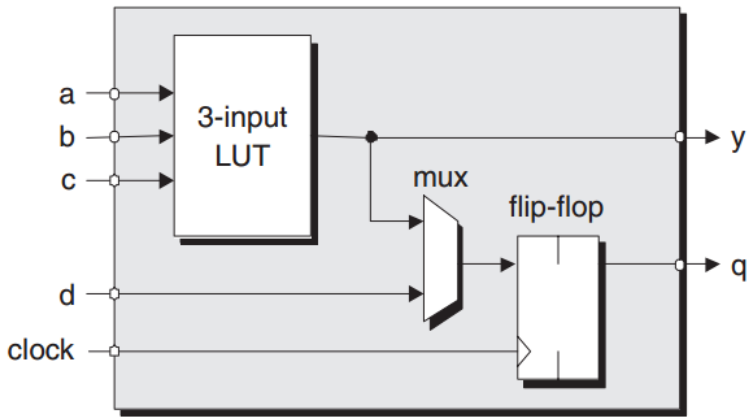
FPGA Architecture

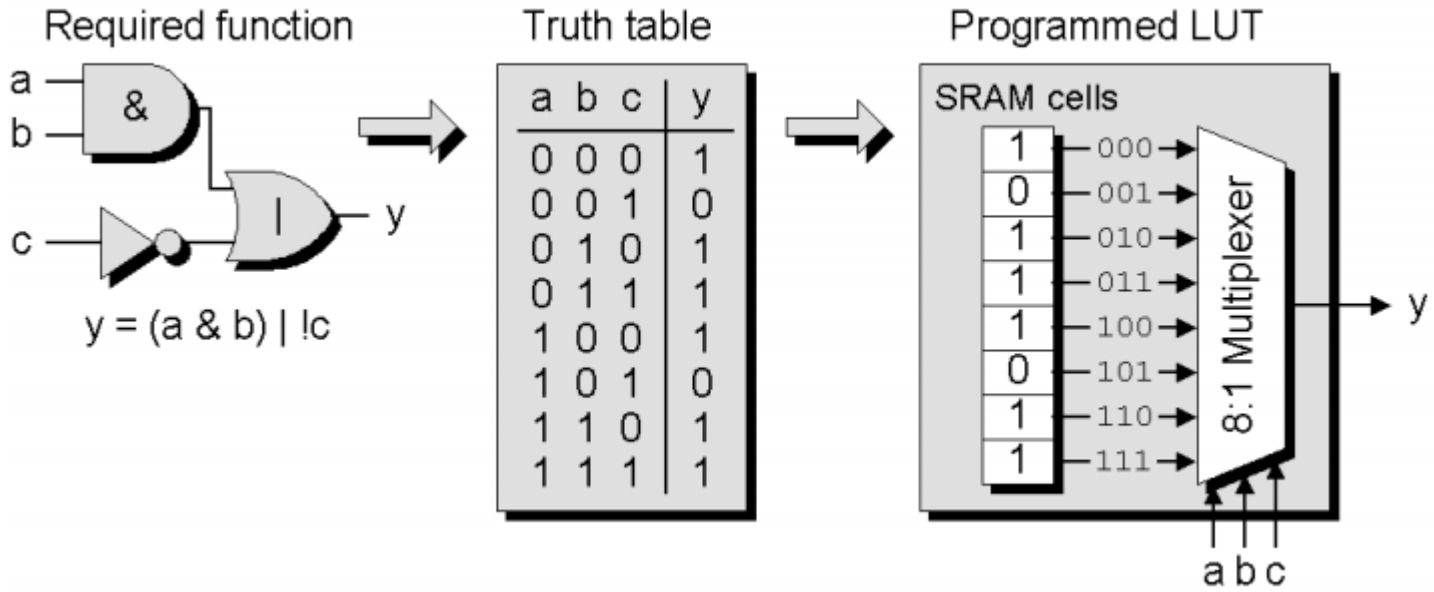
FPGA (1984)


a
b
c
d
clock
3-input
LUT
mux
flip-flop
y
q

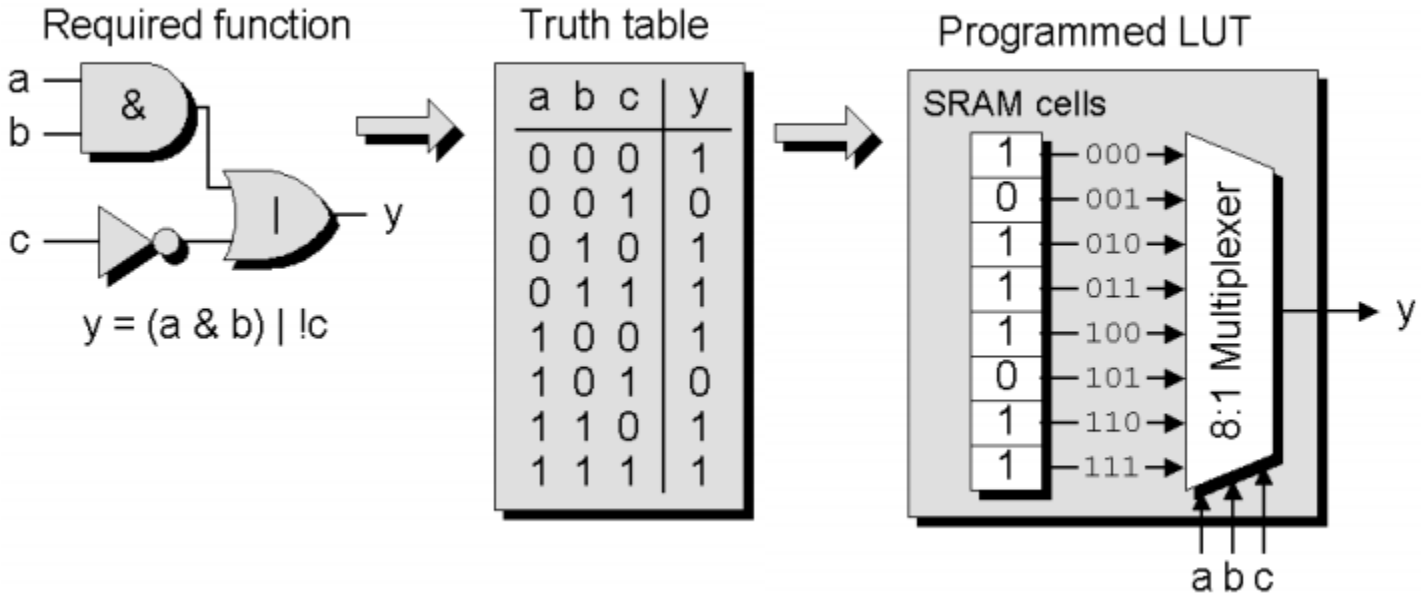

Programmable
interconnect
Programmable
logic blocks

FPGA (1984)

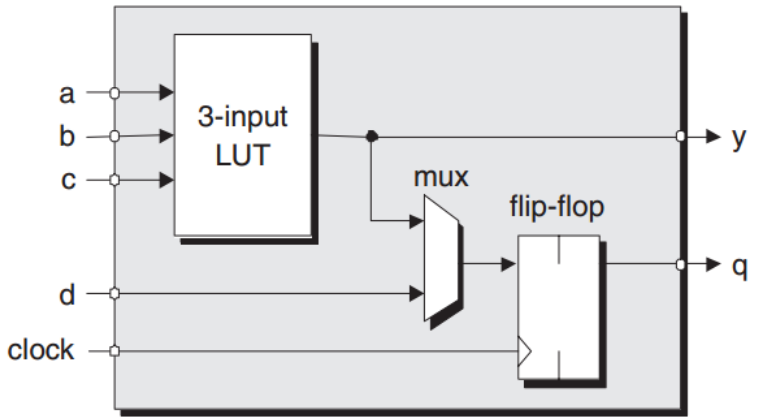

a
b
c
3-input
LUT
mux
flip-flop
y
d
q
clock


Required function
a
b
&
|
c
y
y = (a & b) | !c
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
SRAM cells
1
0
1
1
1
0
1
1
000
001
010
011
100
101
110
111
8:1 Multiplexer
y
a b c

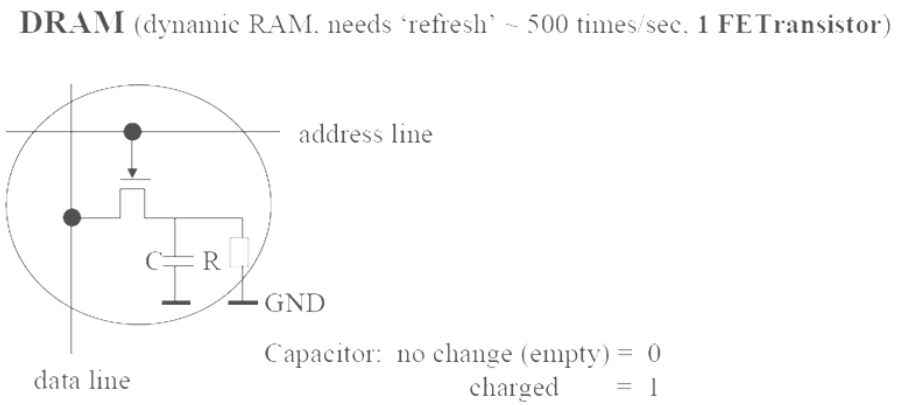
LUT



- What is the difference between *y* and *q* outputs?



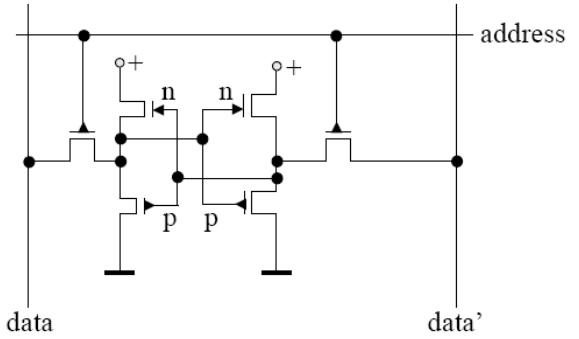
Memory



- DRAM: Dynamic RAM
- A DRAM chip consists of a number of memory cells which can each hold 1-bit of data, stored in a capacitor.
- Due to the fact that capacitors leak electric charge, the state of the bit of information held by each memory cell will eventually fade unless the charge of the capacitor is periodically refreshed by the memory controller.
- The memory controller does this by reading the state of each memory cell and then writing the state back again.
- This is where dynamic RAM gets its name.

Memory

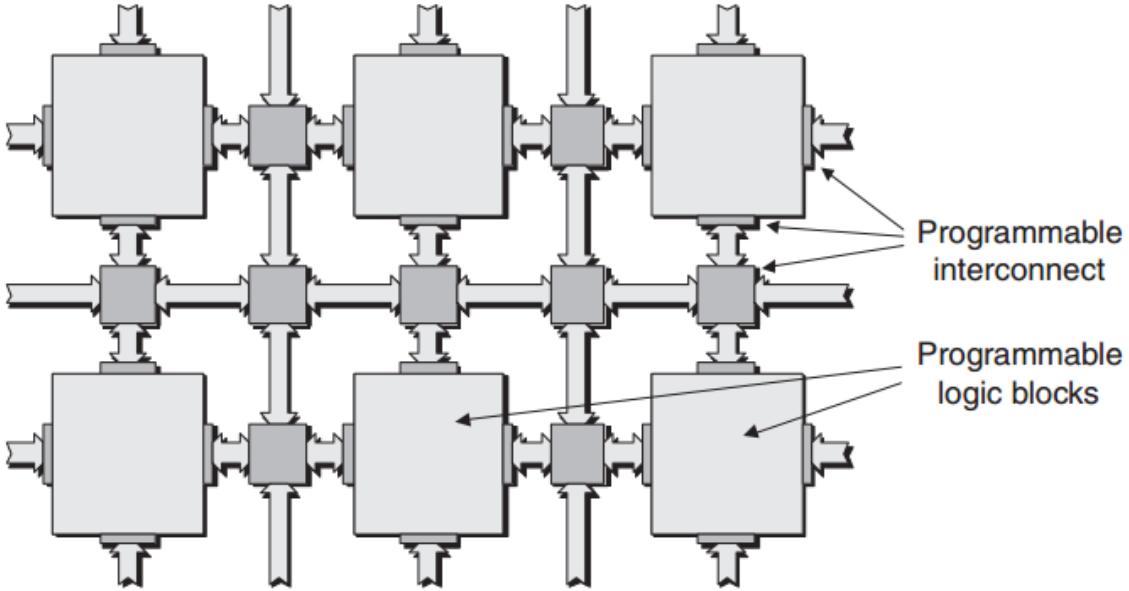
SRAM (static RAM, no 'refresh' required, **6 FETransistors**)
CMOS = complementary metal oxide silicon



- SRAM: Static RAM
- Whereas each bit of memory in DRAM is stored in a capacitor, SRAM uses **latches** to store the data.
- More space than DRAM and expensive
- No refreshing and hence, faster than DRAM
- Used in high-speed, low-capacity memory chips (L1 Cache)

FPGA

- Islands of PLD surrounded by sea of programmable interconnects
- Based on CMOS and SRAM (EPROM or EEPROM in CPLD)
- FPGA Logic Blocks are implemented as LUTs (Look Up Tables) which are RAM based, while CPLDs implement sum-of-product style logic
- SRAM is mature technology and hence, efficient
- Easy and fast to program
- Less delay in data communication from one logic block to another



FPGA

Feature	SRAM	Antifuse	E2PROM / FLASH
Technology node	State-of-the-art	One or more generations behind	One or more generations behind
Reprogrammable	Yes (in system)	No	Yes (in-system or offline)
Reprogramming speed (inc. erasing)	Fast	----	3x slower than SRAM
Volatile (must be programmed on power-up)	Yes	No	No (but can be if required)
Requires external configuration file	Yes	No	No
Good for prototyping	Yes (very good)	No	Yes (reasonable)
Instant-on	No	Yes	Yes
IP Security	Acceptable (especially when using bitstream encryption)	Very Good	Very Good
Size of configuration cell	Large (six transistors)	Very small	Medium-small (two transistors)
Power consumption	Medium	Low	Medium
Rad Hard	No	Yes	Not really