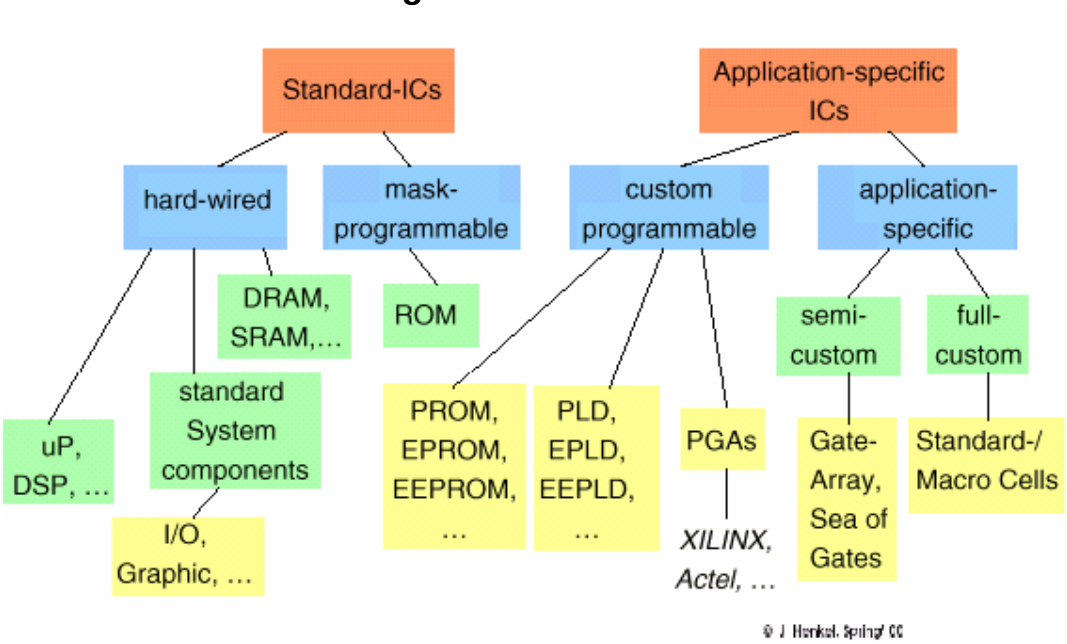
Introduction to Programmable Logic Devices PLDs

1. Programmable Logic devices

Logic Device Families



The main differences in programmable devices are between:

- mask-programmable and field-programmable
- erasable and non-erasable

The mask-programmable types are programmed when they are manufactured whereas the user sets up the field-programmable device with some form of programmer.
Mask-programmable devices are expensive in low production runs but are relatively cheap for large production runs, which is opposite for the field-programmable devices.
An erasable device allows the stored set-up to be changed whereas the non-erasable type is permanent.

Programmable Read-Only Memory : PROM

- First type of user-programmable chip
- Device has a fixed, fully decoded AND plane and a programmable OR plane
- One-time programmable
- A logic circuit can be implemented by using the PROM's address lines as the circuit's inputs , and the circuit's outputs are then defined by the stored bits .
- Any truth table function can be implemented in this way .
- Two basic versions of PROM :
    1) **Mask-Programmable :** can be programmed Only by the manufacturer . Mask-programmable chip has less delay because connections within the device can be hardwired during manufacture .
    2) **Field-Programmable :** can be programmed by the end-user .Field-programmable chips are less expensive , and can be programmed immediately. The Field Programmable PROM developed into two types, the Erasable Programmable Read-Only Memory (EPROM) and the Electrically Erasable Programmable Read-Only Memory (EEPROM). The EEPROM has the advantage of being erasable and reprogrammable many times.

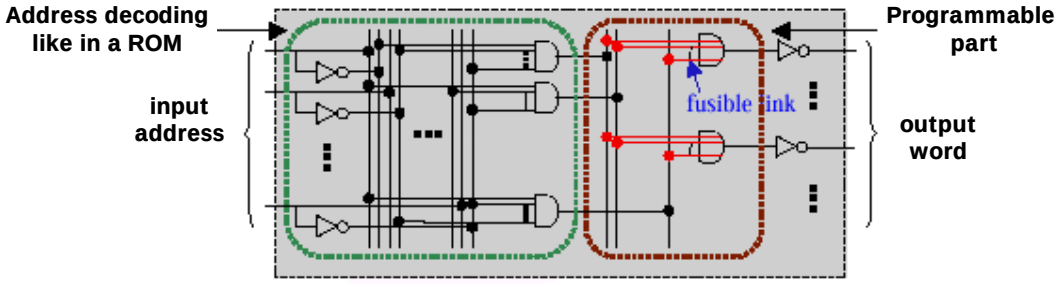
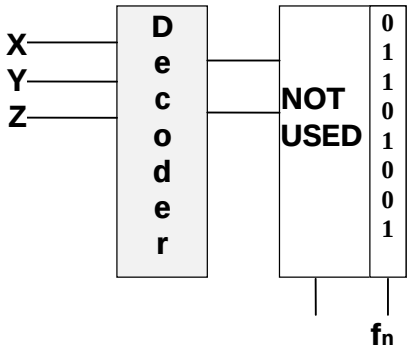


Fig. 1-1 : PROM structure

PROM Example:

f=x^y ^z

x	y	z	f_n . . . f_1 f_0
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1



Simple Programmable Logic Device : SPLD

PLA : (Programmable Logic Array)

A PLA consists of two levels of logic gates: a programmable, “wired” AND-plane followed by a programmable, “wired” OR-plane. A PLA's structure allows any of its inputs (or their complements) to be AND-ed together in the AND plane; each AND plane output can thus correspond to any product term of the inputs. Similarly, users can configure each OR plane output to produce the logical sum of any AND plane output. With this structure, PLAs are well-suited for implementing logic functions in sum-of-products form. They are also quite versatile, since both the AND and OR terms can have many inputs (product literature often calls this feature "wide AND and OR gates")

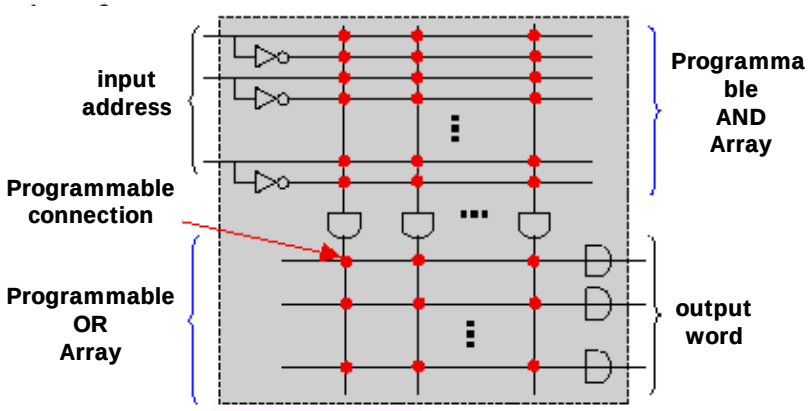


Fig. 1-2: PLA structure

In a PLA, the number of AND functions is independent of the number of inputs, and the number of OR functions is independent of both the number of inputs and the number of AND functions. Also, SPLDs need not have AND input arrays feeding OR output arrays; some devices have two NAND arrays, others have two NOR arrays, and some have a NAND array driving a NOR array.

PAL : (Programmable Array Logic)

It consists of Programmable AND-Plane followed by a fixed OR-Plane .The number of products in an SOP form will be limited to a fixed number (usually 4-10 product

terms). The number of variables in each product term limited by number of input pins (>=10 inputs) .The number of independent functions limited by number of output pins.

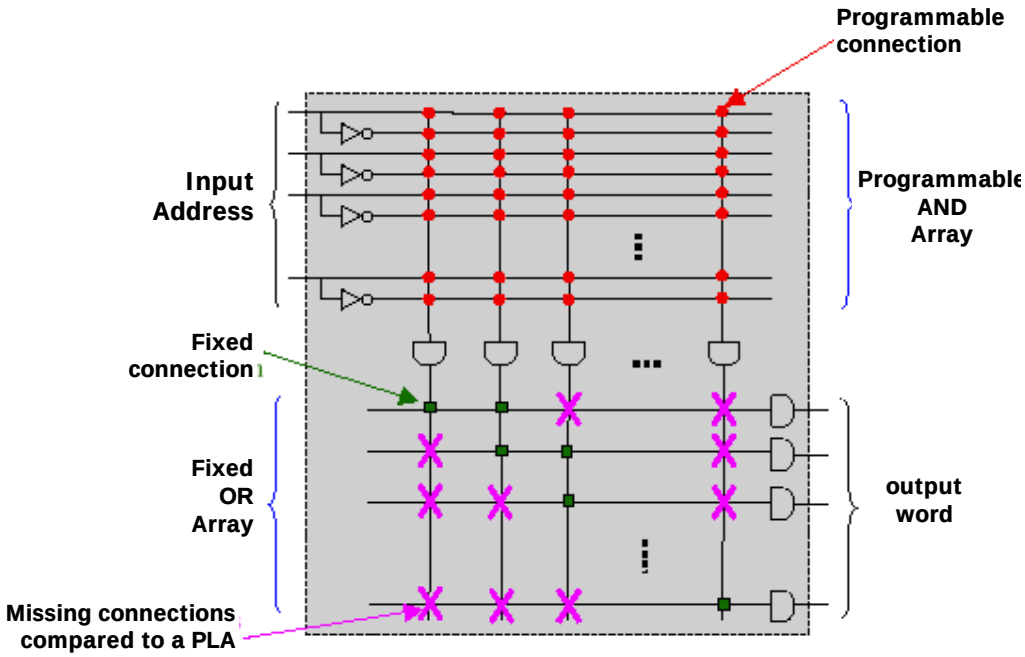


Fig. 1-3: PAL structure

PLAs are more flexible than PALs, but PALs operate faster, because hard-wired connections take less time to switch than their programmable equivalents. Due to the fact that they are fast and cheap to manufacture, PALs are the most common of all the SPLDs.

Complex Programmable Logic Device : CPLD

- also known as:
  - EPLD (Erasable Programmable Logic Device)
  - PEEL (Programmable Electrically Erasable Logic)
  - EEPLD (Electrically-Erasable Programmable Logic Device)
  - MAX (Multiple Array matriX, Altera)
- CPLDs consist of multiple PAL-like logic blocks interconnected with a programmable switch matrix .
- Typically, each logic block contains 4 to 16 macrocells depending on the vendor and the architecture.
- A macrocell on most modern CPLDs contains a sum-of-products combinatorial logic function and an optional flip-flop. The combinatorial logic function typically supports four to 16 product terms with wide fan-in. In other words, a macrocell function can have many inputs, but the complexity of the logic function is limited. Contrast this

structure to an FPGA logic block where complexity is unlimited, but the lookup table has only four inputs.

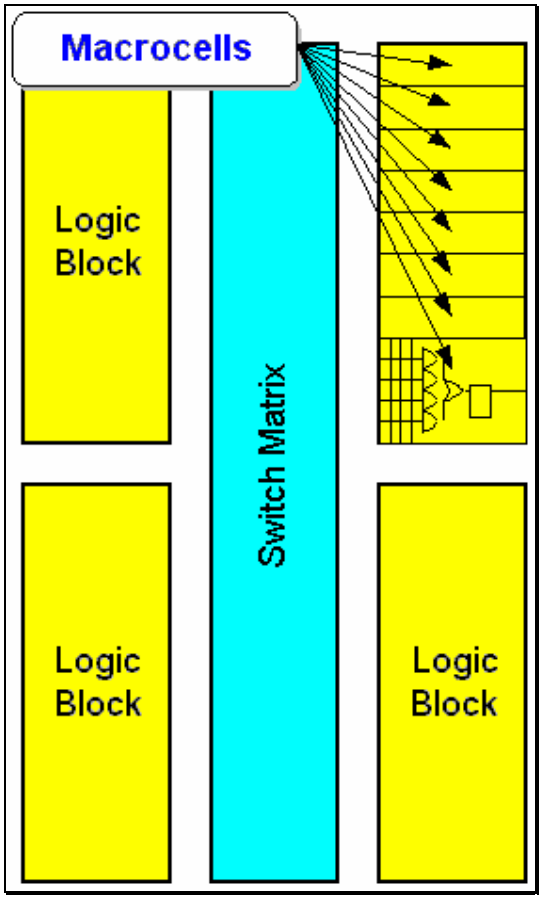


Fig 1-4. The CPLD (or EPLD) architecture uses a few large, PAL-like blocks with I/O fixed inside of each block.

CPLDs are based on one of three process technologies

- EPROM,
- EEPROM, or
- FLASH.

EPROM-based CPLDs are usually one-time programmable (OTP) unless they come in a UV-erasable windowed package.

EEPROM and FLASH processes are erasable technologies. However, not all EEPROM- and FLASH-based devices are programmable while soldered on a board. In-system programmability (ISP) requires special on-chip programming logic, and not all CPLDs come with it, even when built with EEPROM and FLASH technologies. You can erase and program those lacking that circuitry in a device programmer.

CPLDs are generally best for control- oriented designs due in part to their fast pin-to-pin performance. The wide fan-in of their macrocells makes them well-suited to complex, high-performance state machines . At the high end (in terms of numbers of gates), a lot of overlap with FPGAs exists in potential applications. Traditionally, CPLDs have been chosen over FPGAs whenever high-performance logic is required. Because of its less flexible internal architecture, the delay through a CPLD (measured in nanoseconds) is more predictable and usually shorter .

Mask-Programmable Gate Array : MPGA

- Mask-Programmable Gate Array (MPGA) was developed to handled larger logic circuits.
- A common MPGA consists of rows of transistors that can be interconnected to implement desired logic circuits. User sepcified connects are available both within the rows and between the rows. This enabled implementation of basic logic gates and the ability to interconnect the gates.
- As the metal layers are defined at the manufacturer, significant time and cost are incurred in producing the run. In 1985, Xilinx Inc. introduced the FPGA (Field Programmable Gate Array). The interconnects between all the elements were designed to be user programmable

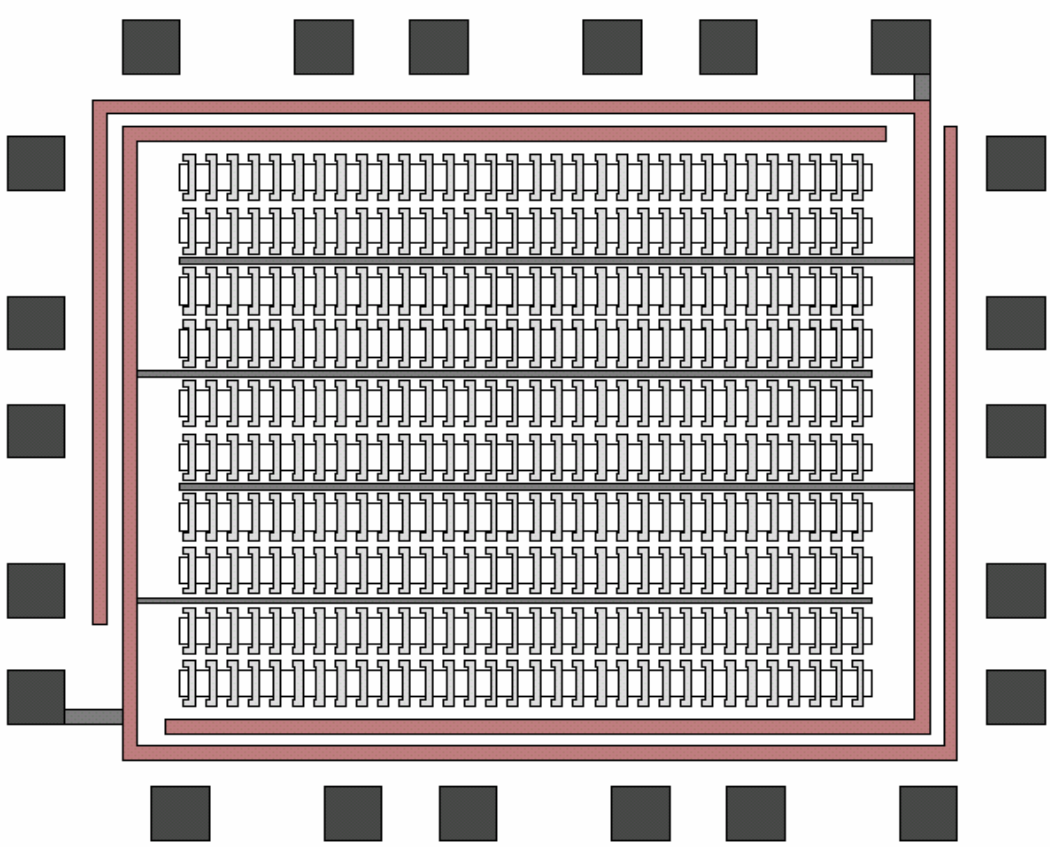
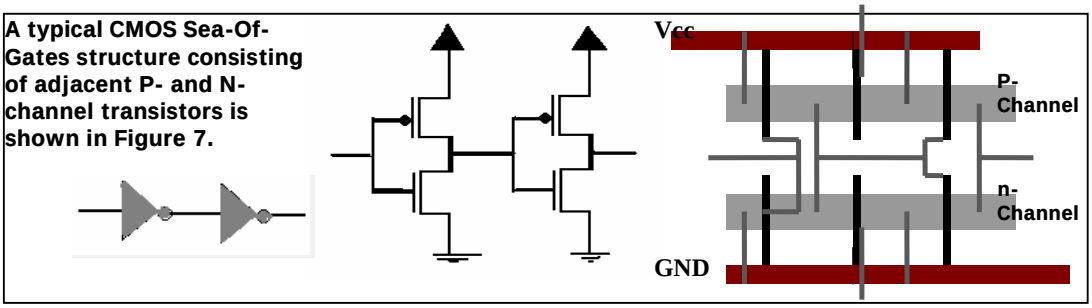


Fig. 1-7: Sea of Gates (SOG) MPGA Architecture



Field-Programmable Gate Array : FPGA

- Like MPGA, an FPGA consists of an array of uncommitted elements that can be interconnected in general way .
- Like a PAL, the interconnections between elements are user-programmable .
- FPGAs are approximately 10 times less dense \ and 3 times slower than MPGAs .
- Consist of an array of logic blocks, surrounded by programmable I/O blocks, and connected with programmable interconnect .

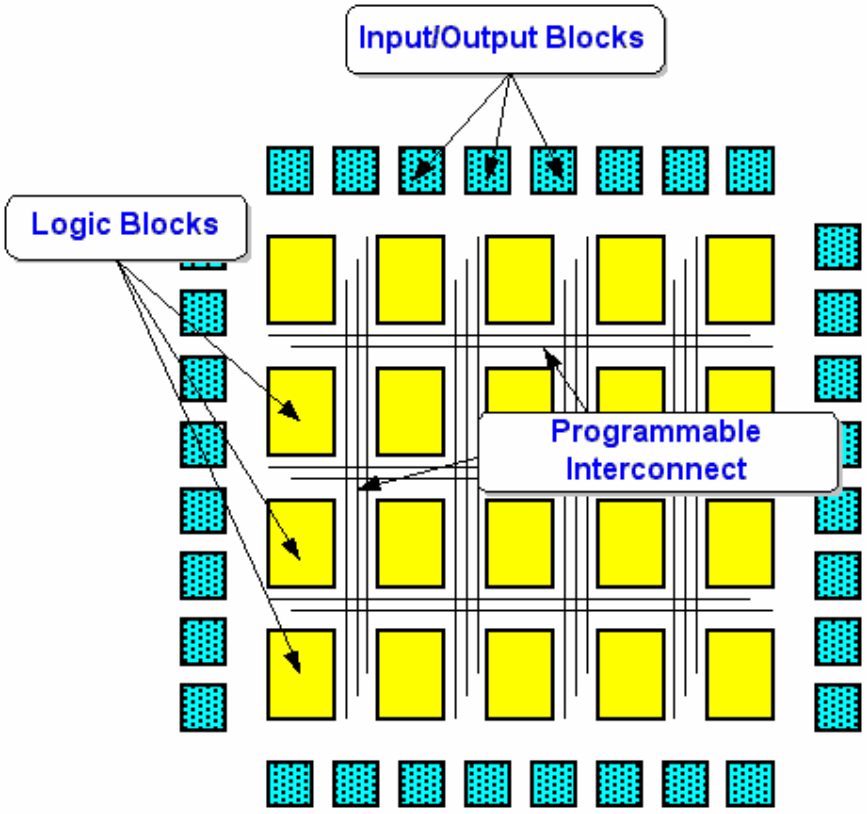


Fig. 1-8 : The FPGA

It consists of a two-dimensional array of logic blocks that can be connected by general interconnection resources . The ***interconnect*** comprises segments of wire , where the segments may be of various length . Present in the interconnect are the programmable switches that serve to connect the logic blocks to the wire segments , or one wire segment to another .

Logic Block

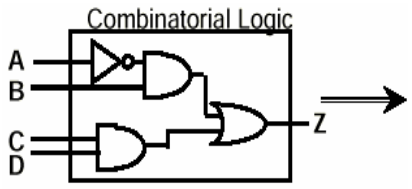
The structure and content of a logic block is called its *architecture*. Logic block architecture can be designed in many different ways .

There are two primary classes of FPGA architectures:

➢ First, **"coarse-grained"** architectures consist of fairly large logic blocks, often containing two or more lookup tables and two or more flip-flops. In these architectures, a 4-input lookup table (think of it as a 16 x 1 ROM) implements the actual logic .

Look Up Tables

- **Combinatorial Logic is stored in 16x1 SRAM Look Up Tables (LUTs) in a CLB**
- **Example:**



- **Capacity is limited by number of inputs, not complexity**
- **Each function generator can be used as 4 input logic (LUT) or as high speed sync.dual port RAM**

Look Up Table
4-bit address

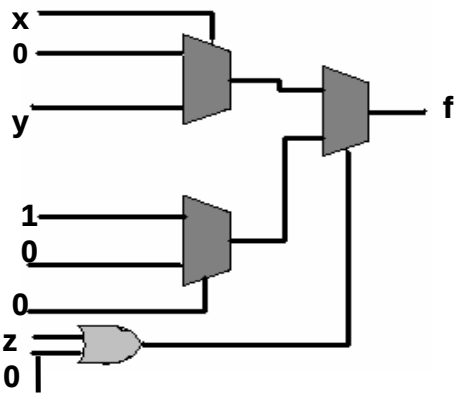
A	B	C	D	Z
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
	.	.	.	
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

$2^{(2^4)}$
= 64K !

➢The other architecture is called **"Fine-grained"**. These devices hold a large number of relatively simple logic blocks. Each block usually contains a flip-flop and either a 2-input logic function or a 4:1 multiplexer.

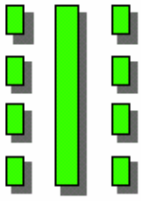
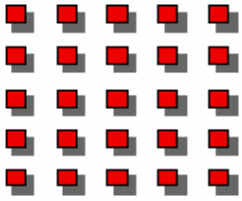
4:1 Multiplexer

f=x.y+z	x	y	z	f
	0	0	0	0
	0	0	1	1
	0	1	0	0
	0	1	1	1
	1	0	0	0
	1	0	1	1
	1	0	0	0
	1	1	0	1
	1	1	1	1



- **Each logical function can be implemented by a Multiplexer .**

Generally, FPGAs have many more registers and I/O than CPLDs and typically use less power. FPGAs are usually best for datapath-oriented design but don't have the fast pin-to-pin performance associated with CPLDs

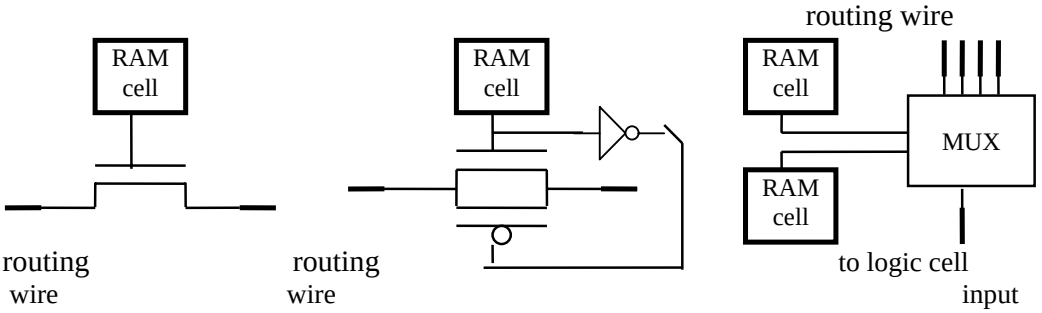
	CPLDs	FPGAs
		
Architecture	**Large, wide fan-in blocks of AND-OR logic**	**Array of small logic blocks surrounded by I/O**
Applications	**Bus interfaces Complex state machines Fast memory interfaces Wide decoders PAL-device integration**	**Logic consolidation Board integration Replace obsolete devices Simple state machines Complex controllers/interfaces**
Key Attributes	**Fast pin-to-pin performance Predictable timing Easy to use**	**Very high density Lots of I/Os and flip-flops Generally lower power SRAM devices are reprogrammable**
Gate Capacity	**300-6,000 gates**	**800-100,000 gates**
Design Timing	**Fixed, PAL-like Very fast pin-to-pin performance**	**Application dependent Very high shift frequencies**
Number of I/Os	**30-200**	**50-400**
Number of Flip-flops	**30-200**	**100-5,500**
Process Technology	**EPROM EEPROM FLASH**	**SRAM Anti-fuse EEPROM**
In-System Programmable	**Some EEPROM- and FLASH-based devices**	**SRAM-based devices and some EEPROM-based devices**
One-Time Programmable (OTP)	**EPROM devices in plastic packages. Some EEPROM- and FLASH-based devices**	**All anti-fuse-based devices**
Performance	**Predictable timing Up to 200MHz today**	**Application dependent Up to 100 MHz today**
Power Consumption	**0.5-2.0W static 0.5-4.0W dynamic**	**Very low static Dynamic consumption is application dependent, 0.1-2W typical**

comparing CPLDs & FPGAs

1. Programming Technology

Static RAM Technology

In the Static RAM FPGA, programmable connections are made using pass-transistors, transmission gates, or multiplexers that are controlled by SRAM cells.



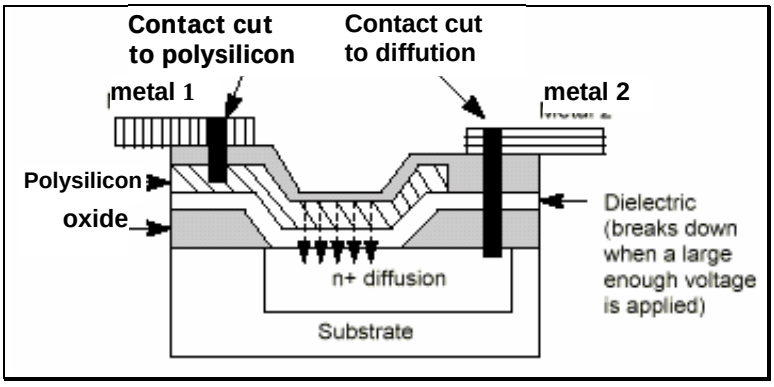
a) pass-transistor b) transmission gate c) multiplexer

In the case of pass-transistor approach RAM cell controls whether pass-gates are on or off. When off, the pass-gate presents a very high resistance between the two wires to which it is attached. When the pass gate is turned on, it forms a relatively low resistance connection between the two wires. The multiplexer would typically be used to optionally connect one of the several wires to a single input of a logic block .

Anti-Fuse Technology

An anti-fuse resides in a high-impedance state; and can be programmed into low impedance or ***"fused"*** state .

The Anti-fuse FPGA vendor, Actel, uses a poly-diffusion antifuse and the high current density causes a large power dissipation in a small area, which melts a thin insulating dielectric between poly-silicon and diffusion electrodes and forms a thin, permanent, and resistive silicon link of about 20nm in diameter. Actel calls it's antifuse a *programmable low-impedance circuit element* or PLICE.

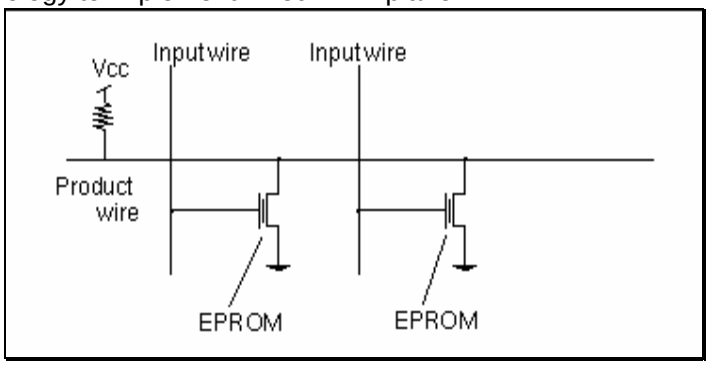


Actel Antifuse

EPROM / EEPROM Technology

This method is the same as used in the EPROM memories.

EPROM transistors are used in FPGAs in a different manner than are static RAM cells or anti-fuses . They are used as **"pull down"** devices for logic block input . One wire , called the "word line" is connected to the select gate of the EPROM transistore . As long as the transistor has not been programmed into the OFF state, the word line can cause the "bit line", which is connected to a logic block input to be pulled to logic zero. Since a pull-up resistor is present on the bit line, this scheme allows the EPROM transistors realize ***"wired-AND"*** logic function. CPLDs (and many SPLDs), use EPROM or EEPROM technology to implement wired-AND plane.



EPROM Application