**SOPEJU OLUWABUSUYI FEYIKEMI**

**COMPUTER ENGINEERING**

**15/ENG02/046**

**MAY,2020**

**Digital system design**

There are various hardware choices available to digital system designers. Usually, several types of digital hardware are used to achieve one’s desired circuit functionality. The designer considers many factors like the circuit’s necessary speed of operation, cost of manufacturing, system power consumption, system size, amount of time available to design the product.

The major digital system categories

1. Standard logic: these devices refer to the basic functional digital components that are available as SSI and MSI chips. They are inexpensive and are useful for simple designs. The three major families of standard logic devices are TTL, CMOS and ECL
2. Microprocessor/digital signal processing (DSP): these devices contain different types of functional blocks. Devices can be controlled electronically. It is slow in execution and offers a great deal of flexibility.
3. Application-specific integrated circuits (ASICs). Here, an integrated circuit is designed to implement a specific application. Four subcategories of ASIC devices are available to create digital systems:
4. Programmable logic devices: they can be custom-configured to create any desired digital circuit, from simple logic gates to complex digital systems.
5. Gate arrays: they are ULSI circuits that offer hundreds of thousands of gates. The desired logic gates are created by interconnections of these prefabricated gates. They are less expensive than PLDs of comparable gate count.
6. Standard cell: they use predefined logic function building blocks to create the desired digital system. These functions have been designed to be much smaller than equivalent functions in gate arrays.
7. Full custom: here, all components and interconnections between them are custom-designed by the IC designer. It results in ICs with high speed and small size.

Programmable Logic Devices (PLD) contain the necessary circuitry to create logic functions and so, are being used to implement digital systems. With PLD, same functionality can be obtained with one IC rather than using several individual logic chips. It has advantages like less board space and less power required, greater reliability, less inventory, and overall lower manufacturing cost.

PLDs are generally described as one of three different types:

1. Simple Programmable Logic Devices (SPLDs)
2. Complex Programmable Logic Devices (CPLDs)
3. Field Programmable Gate Arrays (FPGAs)

Together, CPLDs and FPGAs are often referred to as High-Capacity Programmable Logic Devices (HCPLDs). PLD devices variation are based on the types of semiconductor memory.

|  |  |
| --- | --- |
| SPLDs | HCPLDs |
| They typically contain 600 or fewer gates | They have thousands and hundreds of thousands of gates |
| They are less complicated | They are more complicated |
| They are less expensive | They are more expensive |
| Resources are useful for small digital applications | They provide circuit resources for complete complex digital systems. |

**Fundamentals of PLD circuitry**

Consider a simple PLD device that has two input variables. Each output function is programmed with the fuses located between the AND gates and each of the OR gates. Each of the inputs feed both a non-inverting buffer and an inverting buffer to produce the true and inverted form of each variable. These are the input lines to the AND gate array. Each AND gate is connected to two different input lines to generate product lines, which are a unique product of the input variables. Each of the product lines is connected to one of the four inputs of each OR gate. Each OR output will be a constant 1. Each of the OR outputs can be programmed to any desired function. After programming, the device will permanently generate the selected output functions.



Figure : An example of PLD

**PLD Architectures**

1. Programmable Read-Only Memory (PROM)

PROM can generate any possible logic function of the input variables because it generates every possible AND product term. In general, any application that requires every possible input combination to be available in order to generate output function can use a PROM. PROMs become impractical when many input variables must be accommodated because the number of fuses doubles for each added input variable.



Figure : PROM architecture

1. Programmable Array Logic (PAL)

The PAL has an AND and OR structure like a PROM but in the PAL, inputs to the AND gates are programmable, whereas the inputs to the OR gates are hard-wired. Every AND gate can be programmed to generate any desired product of the four input variables and their complements. Each OR gate is hardwired to only four AND gates. This limits each output function to only four product terms. If a function requires more than four product terms, it cannot be implemented with this PAL. If fewer than four product terms are needed, the unneeded ones can be made 0.

The PAL family also contains devices with variations of the basic SOP circuitry.



Figure : Typical PAL architecture

1. Field Programmable Logic Array (FPLA)

It was developed in the mid-1970s as the first nonmemory programmable logic device. It is more flexible than the PAL architecture. They are mostly used in state-machine designs where a large number of product terms are needed in each SOP expression.

**The GAL 16V8 (Generic Array Logic)**

The GAL chip uses an EEPROM array to control the programmable connections to the AND matrix, allowing them to be erased and reprogrammed at least 100 times. The GAL 16V8 has an architecture very similar to the PAL devices. It contains optional flipflops for register and counter applications, tristate buffers and control multiplexers.

These devices are inexpensive. Designs requiring more hardware resources than is contained in the 16V8 can be broken into smaller blocks and implemented in several 16V8 chips.

The GAL 16V8 has three modes of operation:

1. Simple mode
2. Complex mode
3. Registered mode