A SUMMARY ON PROGRAMMABLE LOGIC DEVICE ARCHITECTURE

PREPARED BY

DERRI COLUMBUS BOMAYE

15/ENG02/017

TO

DEPARTMENT OF ELECTRICAL/ELECTRONICS AND COMPUTER ENGINEERING

AFE BABALOLA UNIVERSITY ADO EKITI

MAY, 2020

INTRODUCTION

Many applications require faster response than a microcomputer/DSP architecture can accommodate and in the cases, a conventional digital circuit must be used. In today's rapidly advancing technology market, most conventional digital systems are not being implemented with standard logic device chips containing only simple gates or MSI-type functions. Instead, programmable logic devices, which contain the circuitry necessary to create logic functions are being used to implement digital systems. These devices are not programmed with a list of instructions, like a computer or DSP. Instead, their internal hardware is configuring by electronically connecting and disconnecting points in the circuit.

The reason why PLDs have taken over so much of the market with programmable devices, the same functionality can be obtained with one IC rather than using several individual logic chips. This characteristic means less board space, less power required, greater reliability, less inventory, and over all lower cost in manufacturing.

DIGITAL SYSTEMS FAMILY TREE

A digital system family tree as shown below in the figure, shows most of the hardware choices that are currently available can be useful in sorting out the many categories of digital devices. The graphical representation in the figure does not show all the details – some of the more complex device types hence many additional subcategories, and older, obsolete device types have been omitted for clarity. The major digital system categories include standard logic, application-specific integrated circuits (ASICs) and microprocessor digital signal processing (DSP) devices.

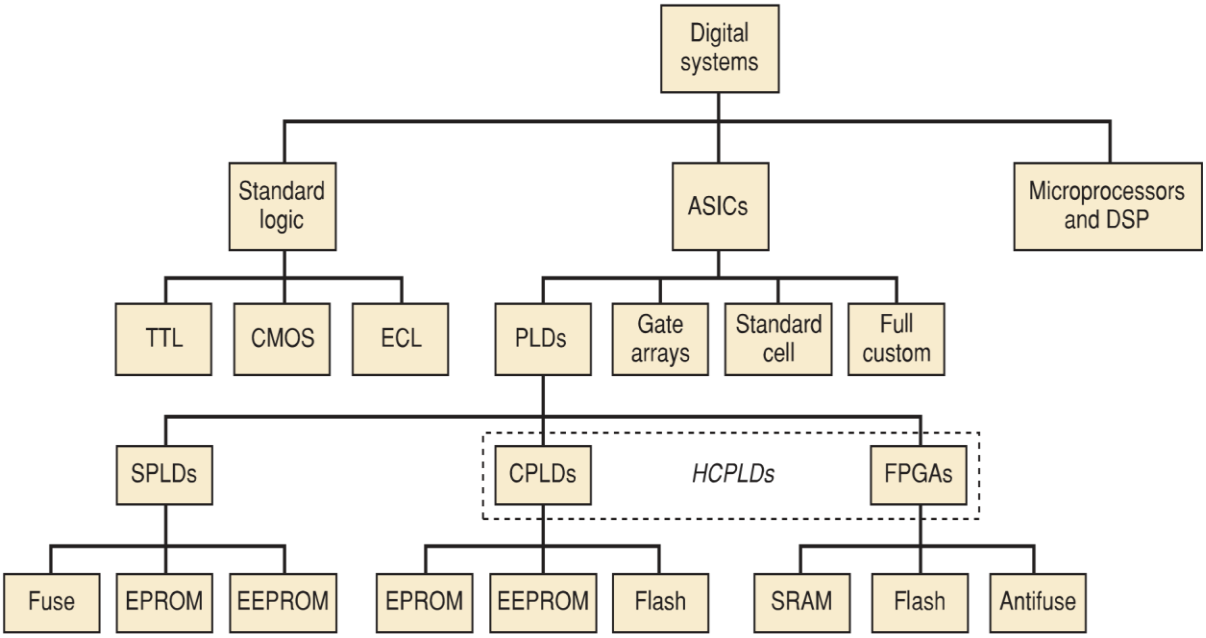

FIGURE 13-1 Digital system family tree.

The first category of standard logic devices refers to the basic functional digital components (gates, flip-flops, decoders, multiplexers, registers, counters, etc.) that are available as SSI

and MSI chips. These devices have been used for many years (some more than 30 years) to design complex digital system.

The microprocessors/digital signal processing (DSP) which is the second category is a much different approach to digital system design. These devices actually contain the various types of functional blocks that have been discussed throughout this text. With microcomputer/DSP systems, devices can be converted electronically, and data can be manipulated by executing a program of instructions that has been written for the application.

The third major digital system category is called application-specific integrated circuits (ASICs). This broad category represents the modern hardware design solution for digital systems. As the acronym implies, an integrated circuit is designed to implement a specific desired application. Four subcategories of ASIC devices are available to create digital systems; programmable logic devices, gate arrays, standard-cell, and full-custom.

FUNDAMENTALS OF PLD CIRCUITRY

A simple PLD device is shown in the figure below; each of the four OR gates can produce an output that is a function of the two input variables A and B. each output function is programmed with the fuses located between the AND gates and each of the OR gates.

Each of the inputs A and B feed both a noninverting buffer and an inverting buffer to produce the time and inverted forms of each variable. These are the input lines to the NAND gate array. Each AND gate is connected to two different input lines to generate a unique product of the input variables. The outputs are called the product lines.

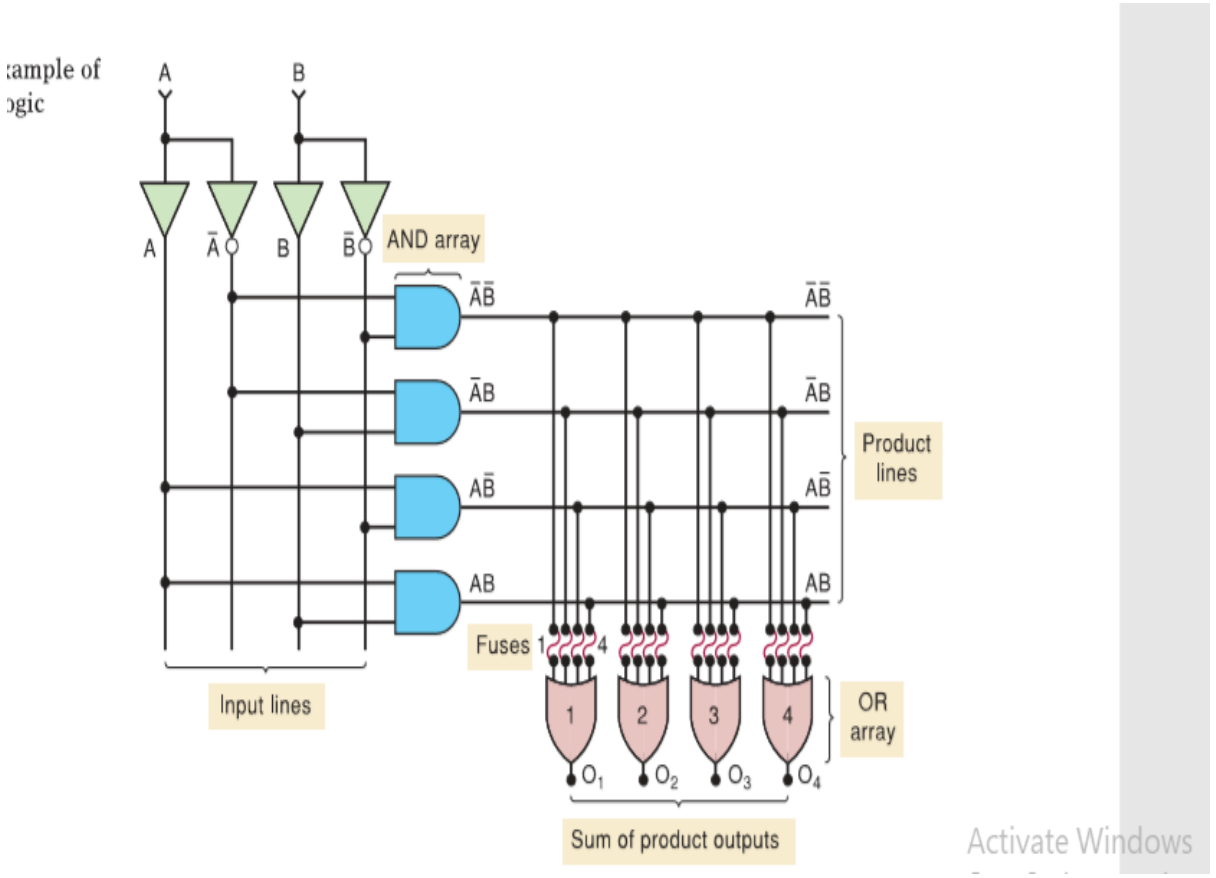

Figure: Example of a programmable logic device

PLD SYMBOLOGY

The example in the figure above has only two input variables and the circuit diagram is already quite cluttered. You can imagine how messy the diagram would be for PLDs with many more inputs. For this reason, PLD manufacturers have adopted a simplified symbolic representation of the internal circuitry of these devices.

PLD ARCHITECTURES

The concept of PLDs has led to many different architectural designs of the inner circuitry of the devices. In this section, we will explore some of the basic different in architecture.

PROMs

The architecture of the programmable circuits in the previous saction involves programming the connections to the OR gate. The AND gates are used to decode all the possible combinations of the input variables., as shown in the figure below. For any given input combination the corresponding row is activated (goes HIGH). If the OR input is connected to that row, a HIGH appears at the OR output. If you think of the input variables as address inputs and the intact/blown fuses as stored 1s and 0s, you should recognize the architecture of a PROM.

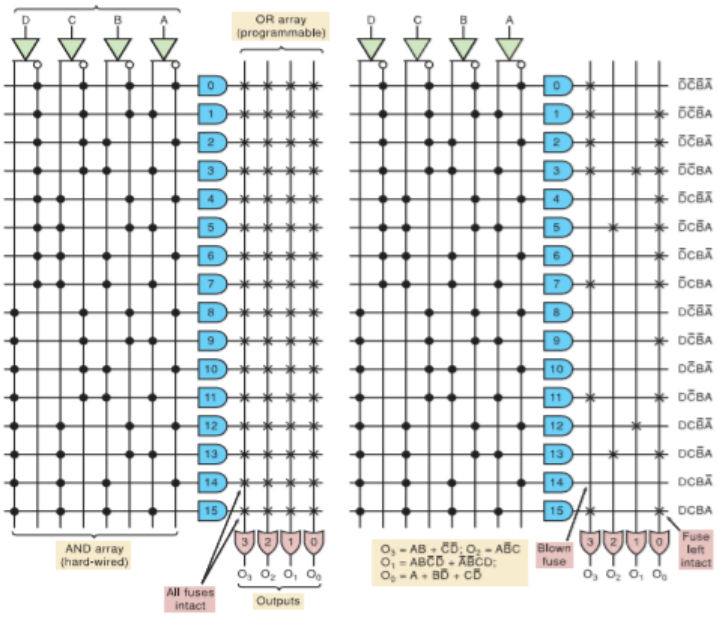

FIGURE 13-5 (a) PROM architecture makes it suitable for PLDs; (b) fuses are blown to program outputs for given functions.

PROGRAMMABLE ARRAY LOGIC (PAL)

The PAL has an AND and OR structure similar to a PROM but in the PAL, inputs to the AND gates are programmable, whereas the inputs to the OR gates are hard-wired. This means that every AND gate can be programmed to generate any desired product of the four input variables and their complements. Each OR gate is hard-wire to only four AND outputs. This limits each output function to four product terms.

FIELD PROGRAMMABLE LOGIC ARRAY (FPLA)

The field programmable logic array (FPLA) was developed in the mid-1970s as the nonmemory programmable logic device. It used a programmable AND array as well as a programmable OR array. Although the FPLA is more flexible that the PAL architecture, kit has not been as widely accepted by engineers. FPLAs are used mostly in state-machine design where a large number of product terms are needed in each SOP expression.

THE GAL 16V8 (GENERIC ARRAY LOGIC)

The GAL 16V8, introduced by Lattice Semiconductor, has an architecture that is very similar to the PAL devices described in the previous section. Standard, low-density PALs are one-time progframmable. The GAL chip, on the other hand, uses an EEPROM array to control the programmable connections to the AND matrix, allowing then to be erased and programmed at least 100 times.

The flexibility of the GAL 16V8 lies in its programmable output logic macrocell. Eight different products (outputs of AND gates) are applied as inputs to each of the eight output logic macrocells. Within each OLMC, the products are ORed together to generate the sum of products (SOP).

THE ALTERA EPM7128S CPLD

We will investigate the architecture of the EPM7128S, an EEPROM-based device in the Altera MAX7000S CPLD family. This device is found on several educational development boards, including the Altera UP2, De Vry eSOC, and RSR PLDT-2. The major structures in the MAX7000S are the logic array blocks (LABs) and the programmable interconnect array (PIA). A LAB contains a set of 16 macrocells and looks very similar to a single SPLD device. Each macrocell consists of a programmable AND/OR circuit and a programmable register (flip-flop).

Logic signals are routed betweeen LABs via the PIA. The PIA is a global bus that connects only signal source to any destination within the device. All inputs to the MAX7000S device and all macrocell outputs feed the PIA. Up to 36 signals can feed each LAB from the PIA. Only signals needed to produce the required functions for any LAB are fed into that LAB.