Subr ADDV.
Class Btech 3rd yr ECE
faculty: Mandeep singh
Date 8/4/2020

### 2.2.4Field-programmable gate array

A **field-programmable gate array** (FPGA) is a semiconductor device that can be configured by the customer or designer after manufacturing—hence the name "field-programmable". To program an FPGA you specify how you want the chip to work with a logic circuit diagram or a source code in a hardware description language (HDL). FPGAs can be used to implement any logical function that an application-specific integrated circuit (ASIC) could perform, but the ability to update the functionality after shipping offers advantages for many applications.

FPGAs contain programmable logic components called "logic blocks", and a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together"—somewhat like a one-chip programmable breadboard. Logic blocks can be configured to perform complex combinational functions, or merely simple logic gates like AND and XOR. In most FPGAs, the logic blocks also include memory elements, which may be simple flip-flops or more complete blocks of memory.

For any given semiconductor process, FPGAs are usually slower than their fixed ASIC counterparts. They also draw more power, and generally achieve less functionality using a given amount of circuit complexity. But their advantages include a shorter time to market, ability to re-program in the field to fix bugs, and lower non-recurring engineering costs. Vendors can also take a middle road by developing their hardware on ordinary FPGAs, but manufacture their final version so it can no longer be modified after the design has been committed.

- For all but the largest devices, routing constrains most logic blocks to have input and output signals connected to external pins (little opportunity for internal state storage or deeply layered logic).

## 1.7 Features in common with FPGAs:

Large number of gates available. CPLDs typically have the equivalent of thousands to tens of thousands of logic gates, allowing implementation of moderately complicated data processing devices. PALs typically have a few hundred gate equivalents at most, while FPGAs typically range from tens of thousands to several million.

Some provisions for logic more flexible than sum-of-product expressions, including complicated feedback paths between macro cells, and specialized logic for implementing various commonly-used functions (such as integer arithmetic).

The most noticeable difference between a large CPLD and a small FPGA is the presence of on-chip non-volatile memory in the CPLD. The differences in architectural approach become more apparent farther from this intermediate region.

This characteristic of non-volatility means that CPLDs are often used in modern digital design to perform 'boot loader' functions before handing over control to other devices not having this capability. A good example is where a CPLD is used to load configuration data for an FPGA from non-volatile memory.

CPLDs were an evolutionary step from even smaller devices that preceded them, PLAs (first shipped by Signetics), and PALs.

As FPGA's become more advanced the differences between the two device types become blurred this trend will continue until the two types are essentially indistinguishable.

### 2.2.3Complex programmable logic device

A **complex programmable logic device (CPLD)** is a programmable logic device with complexity between that of PALs and FPGAs, and architectural features of both. The building block of a CPLD is the **macro cell,** which contains logic implementing disjunctive normal form expressions and more specialized logic operations.

Features in common with PALs:

□ Non-volatile configuration memory. Unlike many FPGAs, an external configuration ROM isn't required, and the CPLD can function immediately on system start-up.

□ For many legacy CPLD devices, routing constrains most logic blocks to have input and output signals connected to external pins, reducing opportunities for internal state storage and deeply layered logic. This is usually not a factor for larger CPLDs and newer CPLD product families.

Features in common with FPGAs:

□ Large number of gates available. CPLDs typically have the equivalent of thousands to tens of thousands of logic gates, allowing implementation of moderately complicated data processing devices. PALs typically have a few hundred gate equivalents at most, while FPGAs typically range from tens of thousands to several million.

□ Some provisions for logic more flexible than sum-of-product expressions, including complicated feedback paths between macro cells, and specialized logic for implementing various commonly-used functions, such as integer arithmetic.

The most noticeable difference between a large CPLD and a small FPGA is the presence of on-chip non-volatile memory in the CPLD. This distinction is rapidly becoming less relevant, as several of the latest FPGA products also offer models with embedded configuration memory.

The characteristic of non-volatility makes the CPLD the device of choice in modern digital designs to perform boot loader.