# Implementation Technologies

*After this lecture you should be able to: explain the growth of digital electronics; select software, PLDs, or ASICs as most appropriate solve a particular problem; explain the terms: Moore's Law, ASIC, CPLD, FPGA, feature size, VLSI, fabless, wafer, die, NRE, FPGA, SoC, LE and LUT.*

## Introduction

Digital Integrated Circuits (ICs) have increased in complexity at an exponential rate over the last 40 years. This growth has been at a rate predicted by "Moore's Law" – an observation that digital IC complexity per unit area seems to double every 2 to 3 years.

Moore's law does not apply to analog ICs. This is because the die area required for an analog IC (e.g. an op-amp) is determined by factors such as its voltage and power rating rather than by the minimum transistor size.

The steady decline in the cost of digital relative to analog electronics has resulted in modern electronic devices implementing almost all functionality using digital rather than analog electronics. The main exception is interface electronics, including power electronics.

Digital ICs can be classified by the number of gates or transistors in an IC (e.g. SSI, LSI and VLSI standing for small, large and very large scale integration). Application-specific ICs (ASICs) are ICs designed for a specific application although the term today is used for any complex and non-programmable logic IC. Many ASICs include a general-purpose CPU as well as memory and peripheral interfaces and application-specific components such as graphics or signal processors. These are often called "System on a Chip" (SoC).

Digital ICs are also classified by the "feature size" of their masks measured in nanometers. In 2023, ICs with 3 nm features sizes started manufacturing.

Due to the cost of the equipment required to manufacture ICs with such small feature sizes, only a handful of companies (e.g. Intel, Samsung) own fabrication plants ("fabs") that manufacture digital ICs. Instead, most semiconductor companies are "fabless." These companies design and sell their ICs but use "foundries" to manufacture them. The largest foundry is TSMC (Taiwan Semiconductor Manufacturing Co.) with about 60% of the foundry market.

## IC Manufacturing

IC's are manufactured on (typically) 300 mm diameter wafers of crystalline silicon. Each wafer is put through dozens of manufacturing steps where dopants are diffused into the silicon and alternating layers of insulating (dielectric) and conductive (metal) materials are deposited to build the circuit. Each step requires a "mask" that is used to expose a photoresist so the processing can be limited to specific areas.

Costs of preparing for production are called "Non-Recurring Engineering" (NRE) costs, meaning they do not vary with the number of units produced. ASIC NRE is primarily preparing a mask set for a digital IC and is very high (e.g. $\$10^6$) because the very small dimensions involved requires using precise and expensive equipment. NRE will also include design labour, CAD tools, and IP licensing.

**Exercise 1:** What improvement in number of transistors per unit area would be achieved by reducing the feature size from 7 nm to 5 nm? Approximately how many 5x5 mm die fit on a 300 mm wafer? How many 200x200 nm gates fit on the die?

## PLDs

Programmable Logic Devices PLDs[1] are ICs that can be configured after manufacturing to implement different logic functions. Unlike software that consists of operations on fixed word sizes, drawn from a limited instruction set and performed in sequence, a PLD's logic functions can be defined in more detail and can be performed in parallel if necessary.

PLDs are often categorized into three major types: PALs (Programmable Array Logic, now ob-

[1] Not to be confused with a Programmable Logic Controller, or PLC, which is a piece of equipment used for industrial control applications rather than an IC.

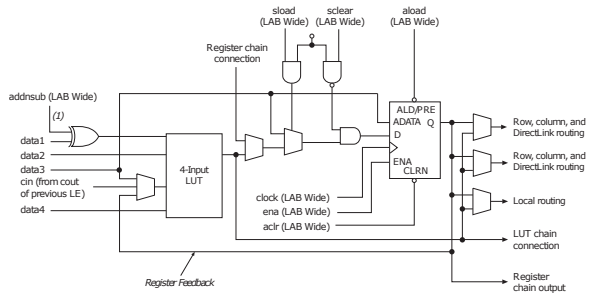
solete), CPLDs (Complex PLDs) and FPGAs (Field-Programmable Gate Arrays). CPLDs have limited functionality and are often used as "glue logic." FPGAs are more capable and can often replace ASICs. The different types of PLDs are described below.

The two largest PLD companies are Xilinx[2] with about 50% of the market and Altera [3], with about 30% of the market.

## PLD Architectures

### CPLD

A CPLD is composed of small programmable logic blocks with combinational logic at the input to a flip-flop. The Altera CPLD logic blocks are called Logic Elements. An example is the Altera MAX-V LE:




Each LE implements combinational logic using a 4 bit in, 1 bit out look-up table (LUT). The LUT output is fed to a single flip-flop. Programmable multiplexers allow groups of LE's to be efficiently configured into a wide range of more-complex logic functions such as registers and adders.

**Exercise 2:** How many bits are required to program the MAX-V LUT? How many additional bits are required to program the 7 two-input multiplexers that configure the LUT?

Additional multiplexers implement an interconnect matrix that allows LE's to be connected to each other and to I/O pins over buses. Logic synthesis software converts an HDL description into a file that programs the LUTs and the interconnect matrix.

Small CPLDs sell for about $1 or $2 in small quantities (e.g. the Intel MAX V 5M40 with 64 I/O pins and 40 LE).

[2]Acquired by AMD.

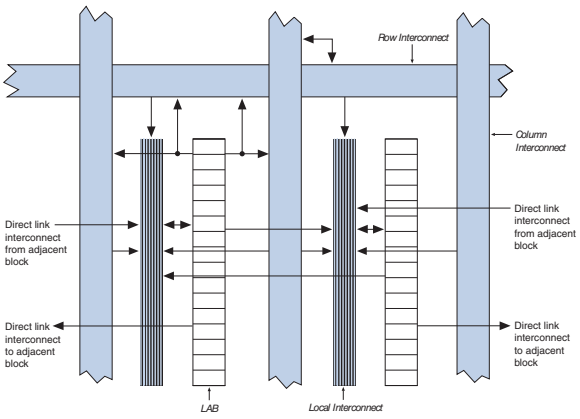
[3]Formerly Intel, formerly Altera.

### FPGA

Gate arrays were an attempt to bridge the gap between fully custom ICs and PLDs. The idea was that gates would be laid out in predefined locations on the die and only the last few layers of interconnect would need to be customized for each IC thus reducing the number of custom masks needed for each IC and thus the NRE. This approach is no longer popular.

This idea developed into a Field-Programmable Gate Array. An FPGA is a PLD that contains a large number (thousands to hundreds of thousands) of simple logic elements similar to those in an Intel CPLD. Note that modern FPGAs have no gates – the combinational logic for each LE is implemented in LUTs.

However, an FPGA's interconnection resources are more limited than those in a CPLD. In keeping with the idea that multiple LEs will be combined to form multi-bit logic functions, logic elements can be connected to their neighbours in logic array blocks (LABs) and these to row and column interconnect buses:




Complex software is required to fit a design into an FPGA and route the signals between logic elements. Because of the multiple levels of interconnect the propagation delays are harder to predict than for a CPLD. Even if there is sufficient logic, some designs may not 'fit' into an FPGA because of insufficient routing resources.

Modern FPGAs include special-purpose components such as RAM, multipliers, PLL clock generators and high-speed serial I/O in addition to general-purpose logic elements.

Most modern FPGAs have enough logic elements and memory that they can be configured with a

"soft" CPU (e.g. Altera's Nios and Xilinx's MicroBlaze). This allows the FPGA to include both software and hardware functions. Some FPGAs include a (hardware-based, typically ARM) CPU core for applications that require both an SoC and programmable logic (a "programmable SoC" or PSOC).

Depending on the version, the FPGA might contain between 6 k and 114 k logic elements and between 180 and 530 I/O pins. Smaller FPGAs sell for under $10 in small quantities. However, large high-performance FPGAs, often used for ASIC prototyping, can cost many thousands of dollars. Due to the high I/O count most FPGAs use ball grid array (BGA) packages.

## PLD Configuration

Although most CPLDs have on-board non-volatile configuration memory, most FPGAs use volatile configuration memory which must be reloaded each time the device powers up. The FPGA can load itself from an external, typically serial, EEPROM or it can be configured through a "JTAG" serial interface. On larger systems that include processors the FPGA is often configured by software running on the processor and in this case the FPGA configuration can be changed as part of a firmware update.

## Technology Selection

A decision must often be made as to whether to use software running on a microcontroller, a hardware design implemented on programmable logic, or a custom ASIC. The following factors affect this decision:

- Performance. This includes clock speed and power consumption. An ASIC (a custom IC) will execute faster and use less power that the same function programmed into an FPGA which in turn will be faster and use less power than the same function implemented in software.
- Sales volume. The NRE (e.g. foundry NRE, logic design, and/or software) must be recovered from sales. A software-based solution typically has the lowest NRE. Programmable logic requires more design effort. A custom ASIC will have much higher NRE than either software or programmable logic.
- Time to Market (TTM). It takes many months to design, verify, and manufacture an ASIC. If TTM is short then a software or programmable logic solution may a better option.
- Flexibility. If the functionality of the IC is likely to change then software or PLD may be a better option. The design can even be changed in the field after the product has been delivered to the customer through software or PLD programming file updates.
- Risk. Errors in an ASIC design will require a "respin" of the IC resulting in additional costs and delays. This can be fatal to a project.

Because of the above considerations, software tends to be used in applications with low-volume or with low-performance requirements. Examples include appliances, and industrial control applications. Programmable logic is used where higher performance is required but where the volumes and are relatively low. Examples of typical application areas include telecommunications infrastructure (e.g. large Internet routers and cellular base stations), medical equipment (e.g. CAT scanners), military/aerospace (e.g. avionics, satellites), and new applications (e.g. artificial intelligence, quantum computing, autonomous vehicles). ASICs are used for high-performance, high-volume applications such as CPUs, graphics processors, and cell phones.

**Exercise 3:** Would you use a PLD or ASIC for: A project that had to be completed within a month? That would be expected to sell 100 million units? Whose complete requirements aren't known? A state-of-the-art general-purpose CPU?