

NAME: KAREEM JADESOLA 'TANWA

MATRIC NUMBER: 15/ENG02/033

COE 506

VHDL ASSIGNMENT

Q.1

- ASIC - Application-specific integrated circuit, it is an integrated circuit (IC) chip customized for a particular use, rather than intended for general-purpose use.
- PAL - Programmable Array Logic is a family of programmable logic device semiconductors used to implement logic functions in digital circuits.
- PLA -A programmable logic array (PLA) is a kind of programmable logic device used to implement combinational logic circuits.
- PLD- Programmable logic device
- CPLD- A complex programmable logic device (CPLD) is a programmable logic device with complexity between that of PALs and FPGAs, and architectural features of both.
- FPGA- A Field-programmable gate array (often shortened to FPGA) is an electronic component used to build reconfigurable digital circuits. That means that an FPGA is different from a logic gate, because a logic gate has a fixed function.

2. Granularity of logic block has influence on performance of an FPGA. Typically, higher granularity level results in lesser delay between input and output. As the granularity of logic block increases, number of levels of logic in critical path decreases, and hence delay in critical path decreases. Granularity, the condition of existing in granules or grains, refers to the extent to which a material or system is composed of distinguishable pieces.

Q3a. Why would anyone use programmable logic devices (PLD, PAL, PLA, CPLD, FPGA, etc.) in place of traditional "hard-wired" logic such as NAND, NOR, AND, and OR gates?

- a few parts instead of a large number of parts
- A PCB design can be fixed for the known inputs and outputs once defined, before the entire working logic schematic is known. Thus, hardware and logic design can be carried on in parallel rather than in sequence which takes longer development time
- As long as the I/O's are connected right, the PCB can be used for development while making changes in code on PCs and not having to make hardware cuts and jumps to fix problems as they are found.

- Easy to add field modifications by loading new code to the in-circuit programmable device or replacing a chip in a socket.

Traditional logic gets displaced mostly by microcontrollers. FPGAs are niche applications because of high costs, but microcontrollers are everywhere. In general, its faster and cleaner and easier to modify and upgrade. And physically smaller and more easily configurable.

b. Are there any applications where hard-wired logic would do a better job than a programmable device?

Programmable IC's cost more than discrete logic and require extra design tools. Hardwired circuits don't need programming. Unless the circuit is extensive, discrete logic is lower cost and faster to produce.

Hard wired may be better for lower power applications (cmos will run down to microamps), where small numbers of gates are required (where you might just need a few nand gates) particularly where cost is critical (standard functions can be purchased for just a few pennies).

Q4. Some programmable logic devices (and PROM memory devices as well) use tiny fuses which are intentionally "blown" in specific patterns to represent the desired program. Programming a device by blowing tiny fuses inside of it carries certain advantages and disadvantages - describe what some of these are.

- the stored program will be non-volatile, but it will also be read-only. This is why fuse-programmed devices are sometimes called "OTP". The **OTP fuse** block is a 128-bit non-volatile memory that can only be **programmed** once, and data is stored permanently. The **OTP** feature is commonly used for customizing a **device** with unique codes. Some examples of **OTP** applications are the storing of manufacturing codes and unique user serial numbers.

$$F_1(w, x, y, z) = wx^2y^2z + wx^2yz^2 + wx^2y^2z^2$$

$$F_2(w, x, y, z) = wx^2y^2z^2$$

w	x	y	z	F <sub>1</sub>	F <sub>2</sub>
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	0
0	1	0	1	0	0
0	1	1	0	0	0
0	1	1	1	0	0
1	0	0	0	0	0
1	0	0	1	1	1
1	0	1	0	1	1
1	0	1	1	1	1
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	1
1	1	1	1	1	1

