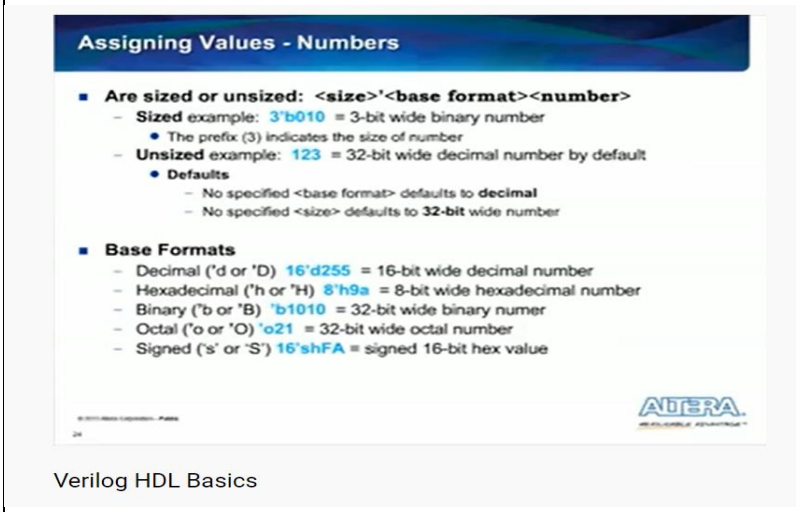
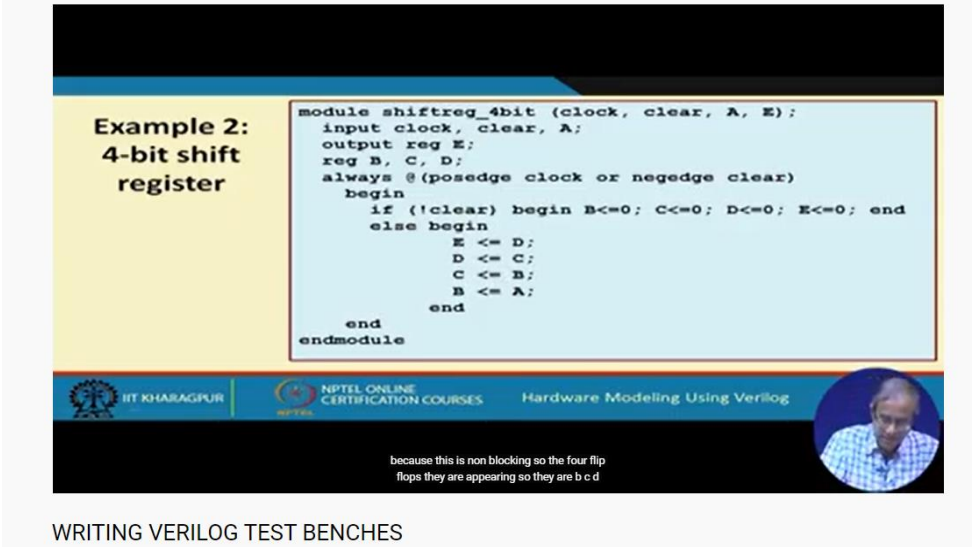
DAILY ASSESSMENT FORMAT

Date:	**02/06/2020**	**Name:**	**Sachin Krishna Moger**
Course:	**DIGITAL DESIGN USING HDL**	**USN:**	**4AL17EC103**
Topic:	**FPGA Basics: Architecture, Applications and Uses** **Verilog HDL Basics by Intel** **Verilog Testbench code to verify the design under test (DUT)**	**Semester & Section:**	**6-B**
Github Repository:	**Sachin-Courses**		

FORENOON SESSION DETAILS



Verilog HDL Basics



WRITING VERILOG TEST BENCHES

Field programmable gate arrays (FPGAs) have established themselves as one of the preferred digital implementation platforms in a plethora of current industrial applications, and extensions and improvements are still continuously being included in the devices. This paper reviews recent advancements in FPGA technology, emphasizing the novel features that may significantly contribute to the development of more efficient digital systems for industrial applications. Special attention is paid to the design paradigm shift caused by the availability of increasingly powerful embedded (and soft) processors, which transformed FPGAs from hardware accelerators to very powerful system-on-chip (SoC) platforms. New analog resources, floating-point operators, and hard memory controllers are also described, because of the great advantages they provide to designers..

What is FPGA?

The **field-programmable gate array (FPGA)** is an integrated circuit that consists of internal hardware blocks with user-programmable interconnects to customize operation for a specific application. The interconnects can readily be reprogrammed, allowing an FPGA to accommodate changes to a design or even support a new application during the lifetime of the part.

The FPGA has its roots in earlier devices such as programmable read-only memories (PROMs) and programmable logic devices (PLDs). These devices could be programmed either at the factory or in the field, but they used fuse technology (hence, the expression “burning a PROM”) and could not be changed once programmed. In contrast, FPGA stores its configuration information in a re-programmable medium such as static RAM (SRAM) or flash memory. FPGA manufacturers include Intel, Xilinx, Lattice Semiconductor, Microchip Technology and Microsemi.

FPGA Architecture

A basic FPGA architecture (Figure 1) consists of thousands of fundamental elements called configurable logic blocks (CLBs) surrounded by a system of programmable interconnects, called a fabric, that routes signals between CLBs. Input/output (I/O) blocks interface between the FPGA and external devices.

Depending on the manufacturer, the CLB may also be referred to as a logic block (LB), a logic element (LE) or a logic cell (LC).

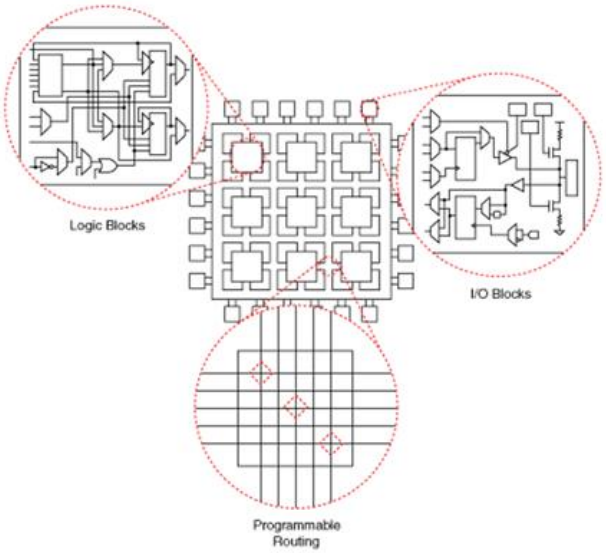


Figure 1: The fundamental FPGA architecture (Image Source: National Instruments)

An individual CLB (Figure 2) is made up of several logic blocks. A lookup table (LUT) is a characteristic feature of an FPGA. An LUT stores a predefined list of logic outputs for any combination of inputs: LUTs with four to six input bits are widely used. Standard logic functions such as multiplexers (mux), full adders (FAs) and flip-flops are also common.

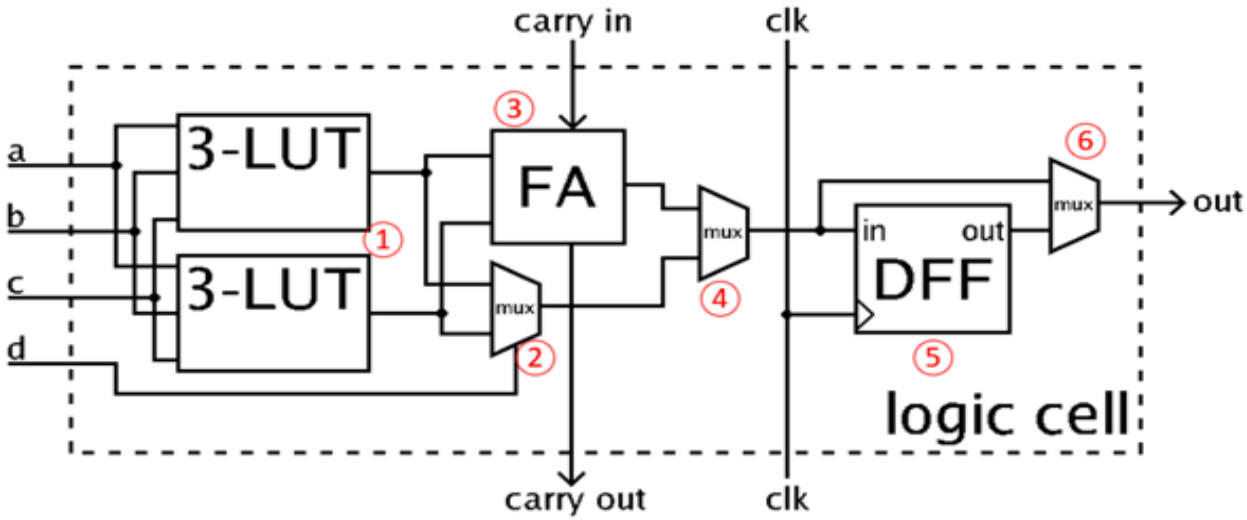


Figure 2: A simplified CLB: The four-input LUT is formed from two three-input units.

The number and arrangement of components in the CLB varies by device; the simplified example in Figure 2 contains two three-input LUTs (1), an FA (3and a D-type flip-flop (5), plus a standard mux (2) and two muxes, (4) and (6), that are configured during FPGA programming.

This simplified CLB has two modes of operation. In normal mode, the LUTs are combined with Mux 2 to form a four-input LUT; in arithmetic mode, the LUT outputs are fed as inputs to the FA together with a carry input from another CLB. Mux 4 selects between the FA output or the LUT output. Mux 6 determines whether the operation is asynchronous or synchronized to the FPGA clock via the D flip-flop.

Current-generation FPGAs include more complex CLBs capable of multiple operations with a single block; CLBs can combine for more complex operations such as multipliers, registers, counters and even digital signal processing (DSP) functions.