Implementation of PIPO REGISTER using VHDL

▪ OBJECTIVE:

To implement **PIPO REGISTER** using VHDL and verify its waveform in Xilinx.

▪ INTRODUCTION:

1. Shift Register:

- A **shift register** is a type of digital memory circuit used in electronics to store and manipulate data. It operates by shifting the data through its registers one bit at a time, either to the left or right. This process is often synchronized with a clock signal.
- Shift Register are used to implement arithmetic operation.
- Basis register used in this register is D ff.

➢ Combined Example

Let’s visualize this with n=4 (binary **0100**):

Left shift by 1 (multiply by 2):
0100 → 1000 **= 4×2=8**

Right shift by 1 (divide by 2):
0100 → 0010 **= 4÷2=2**

Types of Shift Registers

1. **Serial-In Serial-Out (SISO):**
2. **Serial-In Parallel-Out (SIPO):**
3. **Parallel-In Serial-Out (PISO):**
4. **Parallel-In Parallel-Out (PIPO):**

Key Components

- **Flip-Flops:** Basic building blocks, often D-type, which store individual bits.
- **Clock Input:** Synchronizes the shifting of data.
- **Control Inputs:** May include enable, shift direction (left or right), and reset.

Applications

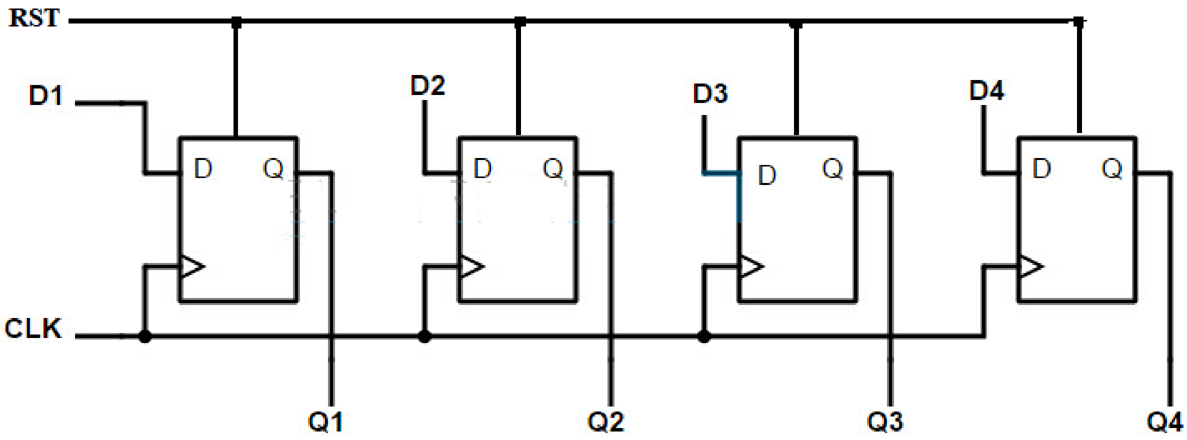
- **Data Conversion:** Serial-to-parallel or parallel-to-serial conversion in communication systems.

- **Data Storage:** Temporary storage of data.
- **Counters:** Acting as ring or Johnson counters.
- **Signal Processing:** Delay lines in digital signal processing.
- **LED Display Control:** Controlling multiple LEDs with fewer I/O pins.

➢ But in this lab we'll implement PIPO Register only.

❖ Pipo Register/Storage Register/Buffer Register:

- Data is loaded and retrieved in parallel.
- Acts like a temporary data storage unit.

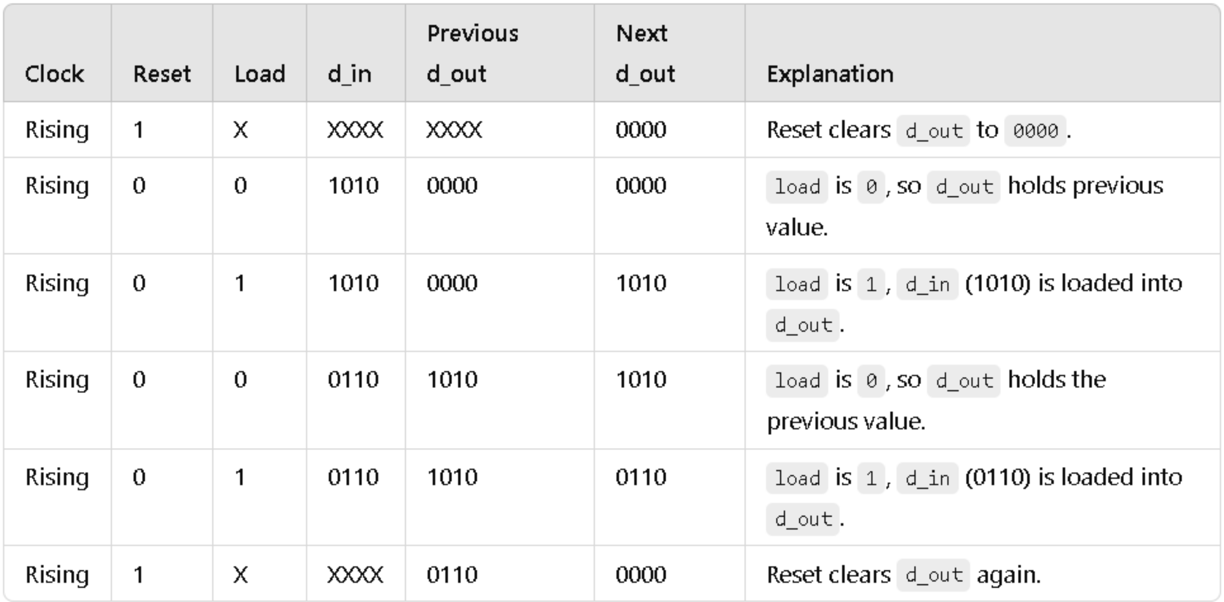


- A reset input (RST) is added to the D-flip flop to reset the flip flop to 0.
- The reset input can be asynchronous, meaning the output responds immediately to the reset input.
- The reset input can also be synchronous, meaning the output changes only at the specified clock edge.

Applications of PIPO Registers

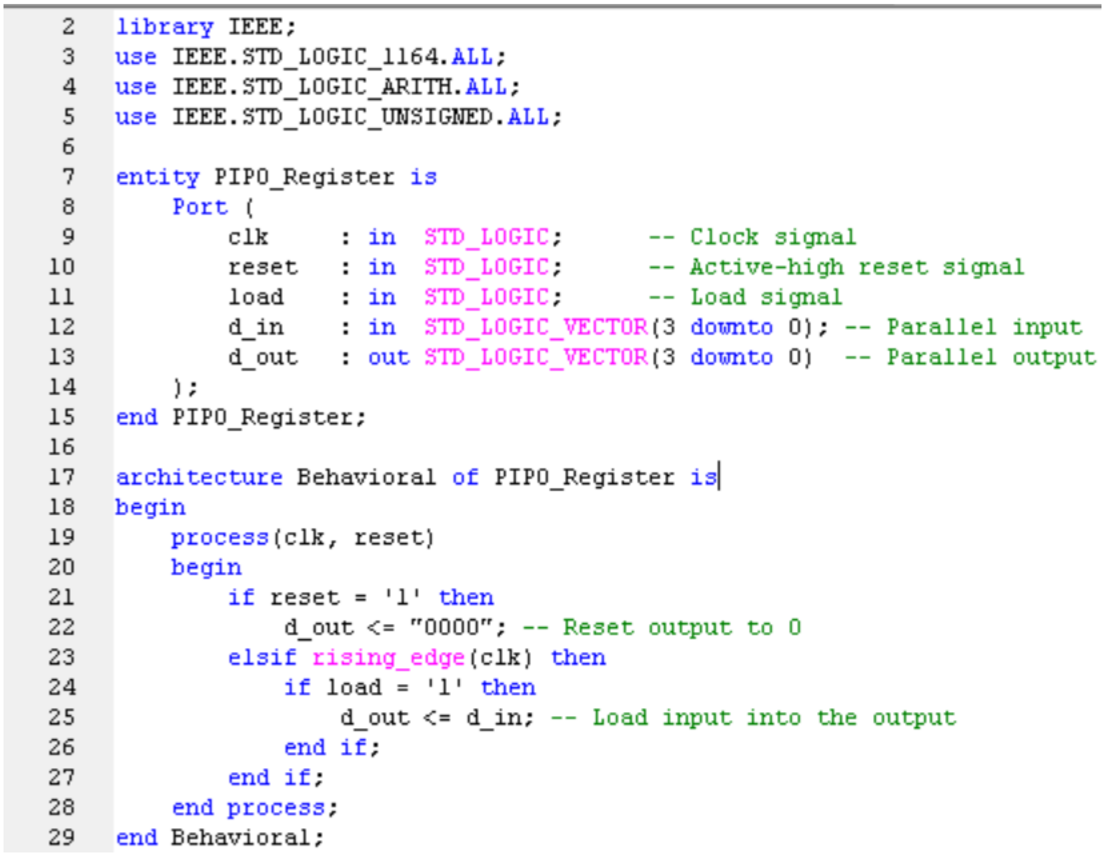
- **High-Speed Data Transfer**: Used in scenarios requiring fast processing and simultaneous data handling.
- **Temporary Storage**: Acts as a buffer to hold multiple bits of data while the system processes them.
- **Signal Processing**: Common in digital circuits to handle signals in parallel form.
- **Microcontroller/Processor Communication**: Often used in embedded systems to interface with peripherals.

▪ Truth Table:



Clock	Reset	Load	d_in	Previous d_out	Next d_out	Explanation
Rising	1	X	XXXX	XXXX	0000	Reset clears `d_out` to `0000`.
Rising	0	0	1010	0000	0000	`load` is `0`, so `d_out` holds previous value.
Rising	0	1	1010	0000	1010	`load` is `1`, `d_in` (1010) is loaded into `d_out`.
Rising	0	0	0110	1010	1010	`load` is `0`, so `d_out` holds the previous value.
Rising	0	1	0110	1010	0110	`load` is `1`, `d_in` (0110) is loaded into `d_out`.
Rising	1	X	XXXX	0110	0000	Reset clears `d_out` again.

▪ Source Code: (Using Implicit D Flip-Flops (4-bit))

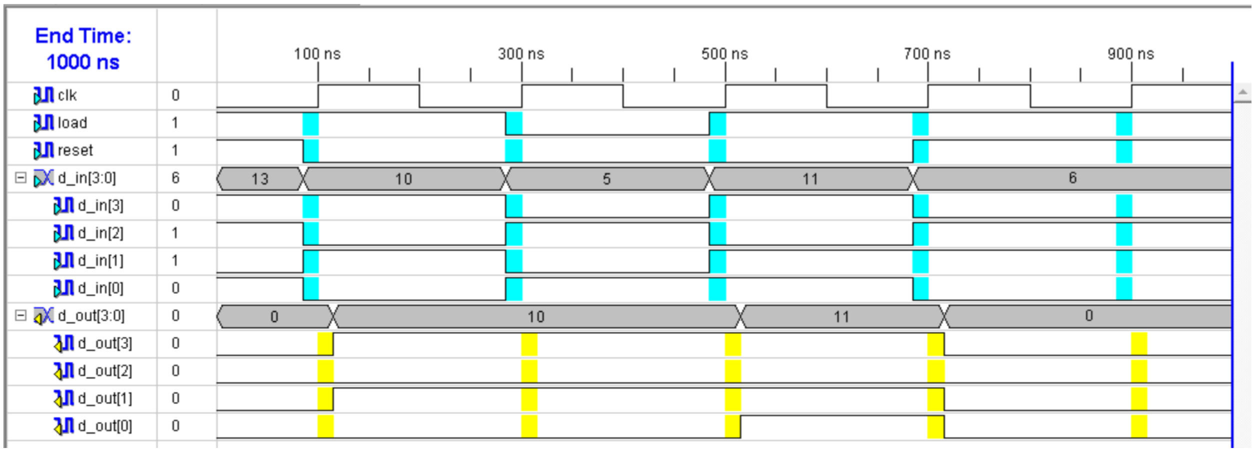


```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity PIPO_Register is
    Port (
        clk     : in  STD_LOGIC;      -- Clock signal
        reset   : in  STD_LOGIC;      -- Active-high reset signal
        load    : in  STD_LOGIC;      -- Load signal
        d_in    : in  STD_LOGIC_VECTOR(3 downto 0); -- Parallel input
        d_out   : out STD_LOGIC_VECTOR(3 downto 0)  -- Parallel output
    );
end PIPO_Register;

architecture Behavioral of PIPO_Register is
begin
    process(clk, reset)
    begin
        if reset = '1' then
            d_out <= "0000"; -- Reset output to 0
        elsif rising_edge(clk) then
            if load = '1' then
                d_out <= d_in; -- Load input into the output
            end if;
        end if;
    end process;
end Behavioral;
```

▪ Test Bench Waveform:



▪ Discussion:

The 4-bit PIPO shift register is implemented using VHDL in Xilinx software. The data loads parallelly and shifts simultaneously in all flip-flops. Simulation results verified correct functionality, indicating proper VHDL coding and synthesis. Timing constraints and propagation delays affect real-time performance, which should be considered in hardware implementation.

▪ Conclusion:

Successfully, the PIPO shift register of 4-bit was designed and simulated. Proper working of the register was confirmed by observing waveform outputs. Future enhancement can include expanding bit-size and optimizing performance parameters.