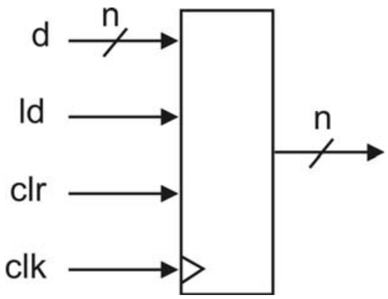
Objectives

- To implement, simulate and test the Register Set required for the 32-bit CPU.
- To implement, simulate and test the 32-bit Program Counter (PC) required for the 32-bit CPU.

1. Register Design

The processor you are designing in this course requires several registers, which are either 32-bit or 1-bit. The symbol for generic n-bit register is illustrated in Figure 1 (below).

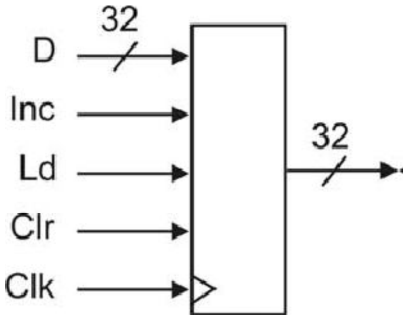


Main inputs of the register include clock (clk), clear (clr), load/enable (ld) signals and an n-bit data (d). The n-bit output is denoted by (q).

You need to implement a 1-bit and a 32-bit register using VHDL. The entity names of these registers should be register1 and register32.

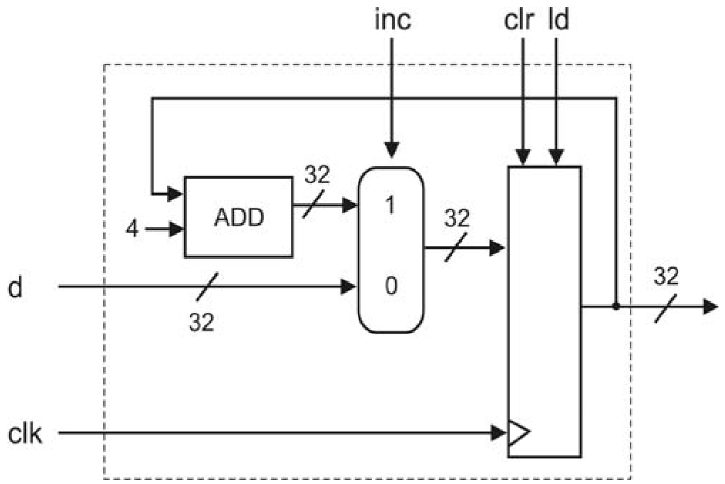
2. Program Counter Design

The program counter **(**PC) is a key component of the processor. The PC is illustrated in Figure 2 (below). The PC has a 32-bit output q and several inputs. Inputs include the clock (clk) and clear (clr) to reset the PC to zero. Also included is a 32-bit input (d) used to set the PC to non-zero values. A load/enable (ld) input load the program counter with input (d) at the rising clock edge. Finally, an increment (inc) input is required for incrementing the PC.



3. Internal design of a 32 bit Program Counter

The internals of the PC are illustrated in Figure 3 (below). The *inc* input is used to select between d or incremented pc. The clr, ld and clk are directly connected to an internal 32-bit register, which is part of the PC comes from. You need to implement a 32-bit PC.



Demonstrate the implementation and simulation results of part 1, 2 & 3 above.