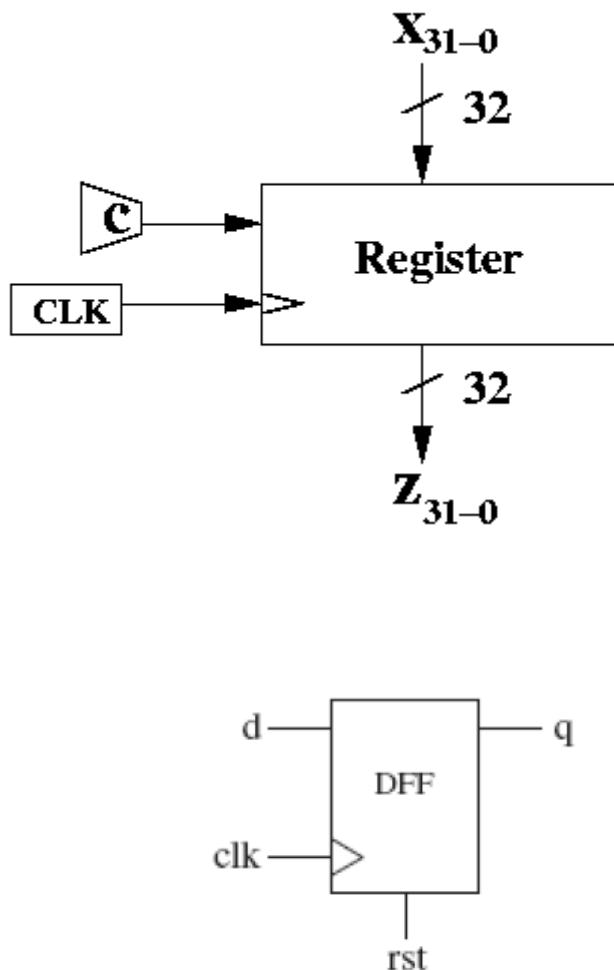


## 32 bit Register

Uses a combination of 32 (1 bit Memory Elements) to store 32 bit data along with 1 bit reset signal.

Block Diagram:



### Explanation:

The Register block shown above contains 32 D flip flop blocks in parallel.

Each of the D flip flop blocks, have a *common Clock* signal and *Reset* signal (for setting output to 32'b0).

Together the register takes 32 bit input and gives out 32 bit output given a positive edge clock and 1 bit reset signal

## Verilog Code:

```
module Reg32b (output [0:31] data_out, input [0:31] data_in, input clk, input res);

    Dff inst1 (data_out[0], data_in[0], clk, res);
    Dff inst2 (data_out[1], data_in[1], clk, res);
    Dff inst3 (data_out[2], data_in[2], clk, res);
    Dff inst4 (data_out[3], data_in[3], clk, res);
    Dff inst5 (data_out[4], data_in[4], clk, res);
    Dff inst6 (data_out[5], data_in[5], clk, res);
    Dff inst7 (data_out[6], data_in[6], clk, res);
    Dff inst8 (data_out[7], data_in[7], clk, res);
    Dff inst9 (data_out[8], data_in[8], clk, res);
    Dff inst10 (data_out[9], data_in[9], clk, res);
    Dff inst11 (data_out[10], data_in[10], clk, res);
    Dff inst12 (data_out[11], data_in[11], clk, res);
    Dff inst13 (data_out[12], data_in[12], clk, res);
    Dff inst14 (data_out[13], data_in[13], clk, res);
    Dff inst15 (data_out[14], data_in[14], clk, res);
    Dff inst16 (data_out[15], data_in[15], clk, res);
    Dff inst17 (data_out[16], data_in[16], clk, res);
    Dff inst18 (data_out[17], data_in[17], clk, res);
    Dff inst19 (data_out[18], data_in[18], clk, res);
    Dff inst20 (data_out[19], data_in[19], clk, res);
    Dff inst21 (data_out[20], data_in[20], clk, res);
    Dff inst22 (data_out[21], data_in[21], clk, res);
    Dff inst23 (data_out[22], data_in[22], clk, res);
    Dff inst24 (data_out[23], data_in[23], clk, res);
    Dff inst25 (data_out[24], data_in[24], clk, res);
    Dff inst26 (data_out[25], data_in[25], clk, res);
    Dff inst27 (data_out[26], data_in[26], clk, res);
    Dff inst28 (data_out[27], data_in[27], clk, res);
    Dff inst29 (data_out[28], data_in[28], clk, res);
    Dff inst30 (data_out[29], data_in[29], clk, res);
    Dff inst31 (data_out[30], data_in[30], clk, res);
    Dff inst32 (data_out[31], data_in[31], clk, res);

endmodule
```