```
module fifo_memory #(parameter mem_width =32, mem_depth = 128, depth= 7 )
(input[mem_width-1:0]temp_data_in,input[depth-1:0] read_addr, write_addr,
input write_enable,read_enable, flush, reset,output reg [mem_width-1:0]
data_out) ;
reg [mem_width-1:0] mem[0:mem_depth-1];
integer i;
always @(*)
begin
      if(!reset)
      begin

            data_out = 32'h0;
            for(i=0;i<128;i=i+1)
                  mem[i] = 32'h0;
      end

      else if(flush)
      begin
            data_out = 32'h0;
            for(i=0;i<128;i=i+1)
                  mem[i] = 32'h0;
      end

      else if(write_enable && read_enable)
      begin
            mem[write_addr] = temp_data_in;
            data_out =  mem[read_addr];
      end

      else if(write_enable )
      begin
            mem[write_addr] = temp_data_in;
      end

      else if(read_enable )
             data_out =  mem[read_addr];


end
endmodule
```