```
use work.interface.all;

entity irqctrl is port (
  clk   : in std_logic;
  rst   : in std_logic;
  sysif : in sysif_type;
  irqo  : out irqctrl_type);
end;

architecture rtl of irqctrl is

  type reg_type is record
    irq    : std_logic;
    pend   : std_logic_vector(0 to 7);
    mask   : std_logic_vector(0 to 7);
  end record;

  signal r, rin : reg_type;

begin

comb : process (sysif, r)
  variable v : reg_type;
begin
  v := r; v.irq := '0';
  for i in r.pend'range loop
    v.pend := r.pend(i) or
       (sysif.irq(i) and r.mask(i));
    v.irq := v.irq or r.pend(i);
  end loop;
  rin <= v;
  irqo.irq <= r.irq;
end process;

reg : process (clk)
begin
  if rising_edge(clk) then
    r <= rin;
  end if;
end process;

end architecture;
```