```
----- gpio.vhd -----
--*****************************************
--
--  GPIO ver 1.0
--  for Sender
--  programmed by kaneko  Dec.20 2004
--
--*****************************************
library ieee;
use ieee.std_logic_1164.all;

--=========================================
--
--  entity
--
--=========================================
entity gpio is
    port(
        RSTn    : in    std_logic;                      -- reset
        CS1n    : in    std_logic;                      -- chip select 1
        CS2n    : in    std_logic;                      -- chip select 2
        AB0     : in    std_logic;                      -- address bus
        RDn     : in    std_logic;                      -- read pulse
        WRn     : in    std_logic;                      -- write pulse
        DB      : inout std_logic_vector(7 downto 0);   -- data bus
        GOUT    : out   std_logic_vector(7 downto 0);   -- output pin
        GIN     : in    std_logic_vector(7 downto 0);   -- input pin
        IRLn    : out   std_logic                       -- IRL
        );
end gpio;

--=========================================
--
--  architecture
--
--=========================================
architecture RTL of gpio is
signal  OREG    : std_logic_vector(7 downto 0);         -- output register
signal  DWRi    : std_logic;                            -- internal write pulse
begin
    ----- make output pin -----
    DWRi <= CS1n or CS2n or WRn or not AB0;
    process(RSTn, DWRi)
    begin
        if(RSTn = '0') then
            OREG <= "11111111";
        elsif(DWRi'event and DWRi = '1') then
            OREG <= DB;
        end if;
    end process;

    GOUT <= OREG;

    ----- make data bus -----
    DB <= GIN   when CS1n = '0' and CS2n = '0' and RDn = '0' and AB0 = '0' else
          OREG  when CS1n = '0' and CS2n = '0' and RDn = '0' and AB0 = '1' else
          (others => 'Z');

    ----- make IRL -----
--  IRLn <= GIN(0) and
--          GIN(1) and
--          GIN(2) and
--          GIN(3) and
--          GIN(4) and
--          GIN(5) and
--          GIN(6) and
--          GIN(7);

    IRLn <= GIN(0) and
            GIN(1) and
            GIN(2) and
            GIN(3) and
            GIN(4) and
            GIN(5) and
            GIN(6);

end RTL;
```