Fulladder

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;


entity fulladder is
port(
  a_i          : in  std_logic;
  b_i          : in  std_logic;
  c_i          : in  std_logic;
--  res_n        : in  std_logic;
--  clk_i        : in  std_logic;
--  enable_i     : in  std_logic;

  c_o          : out  std_logic;
  sum_o        : out  std_logic

);

end entity;

architecture rtl_unbedingt of fulladder is
begin

  c_o <= (a_i and b_i) or (a_i and c_i) or (b_i and c_i);
  sum_o <=  (a_i and (b_i and c_i))
            or ((not a_i) and ((not b_i) and c_i))
            or ((not a_i) and (b_i and (not c_i)))
            or (a_i and ((not b_i) and (not c_i)));
end rtl_unbedingt;

architecture rtl_bedingt of fulladder is
  signal in_vec : std_logic_vector(2 downto 0);
  signal out_vec : std_logic_vector(1 downto 0);
begin
  in_vec <= (c_i, b_i, a_i);
  out_vec <=  "00" when in_vec = "000" else
              "01" when in_vec = "001" else
              "01" when in_vec = "010" else
              "10" when in_vec = "011" else
              "01" when in_vec = "100" else
              "10" when in_vec = "101" else
              "10" when in_vec = "110" else
              "11" when in_vec = "111" else
              "00";
  sum_o <= out_vec(0);
  c_o <= out_vec(1);
```

```vhdl
end rtl_bedingt;

architecture rtl_select of fulladder is
  signal in_vec : std_logic_vector(2 downto 0);
  signal out_vec : std_logic_vector(1 downto 0);
begin
  in_vec <= (c_i, b_i, a_i);
  with in_vec select
  out_vec <= "00" when "000",
             "01" when "001",
             "01" when "010",
             "10" when "011",
             "01" when "100",
             "10" when "101",
             "10" when "110",
             "11" when "111",
             "00" when others;
  sum_o <= out_vec(0);
  c_o <= out_vec(1);
end rtl_select;

configuration fulladder_conf of fulladder is
  for rtl_select
  end for;
end fulladder_conf;
```