

```
-----  
--  
-- Purpose:  
--  
--   Add / Sub  
--  
-- Discussion:  
--  
--  
-- Licensing:  
--  
--   This code is distributed under the GNU LGPL license.  
--  
-- Modified:  
--  
--   2012.03.29  
--  
-- Author:  
--  
--   Young W. Lim  
--  
-- Parameters:  
--  
--   Input:  
--  
--   Output:  
-----
```

```
library STD;  
use STD.textio.all;
```

```
library IEEE;  
use IEEE.std_logic_1164.all;  
use IEEE.numeric_std.all;
```

```
entity addsub is  
  generic (  
    WD      : in natural := 32);  
  
  port (  
    an      : in  std_logic_vector (WD-1 downto 0) := (others=>'0');  
    bn      : in  std_logic_vector (WD-1 downto 0) := (others=>'0');  
    s       : in  std_logic := '0';  
    cn      : out std_logic_vector (WD-1 downto 0) := (others=>'0');  
    co      : out std_logic := '0');  
end addsub;
```

```
architecture rtl of addsub is
```

```
  component adder  
    generic (  
      WD      : in natural );  
    port (  
      an      : in  std_logic_vector (WD-1 downto 0);  
      bn      : in  std_logic_vector (WD-1 downto 0);  
      ci      : in  std_logic := '0';  
      cn      : out std_logic_vector (WD-1 downto 0);  
      co      : out std_logic := '0');  
  end component;
```

```
  signal un : std_logic_vector (WD-1 downto 0) := (others=>'0');  
begin
```

```
  process (an, bn, s)  
  begin -- process  
    if (s='1') then  
      un <= not bn;  
    else
```

```
        un <= bn;  
    end if;  
end process;  
  
A0: adder  
    port map (an => an, bn => un, ci => s, cn => cn, co => co);  
  
end rtl;
```