

```
-----  
--  
-- Purpose:  
--  
-- Barrel Shifter  
--  
-- Discussion:  
--  
--  
-- Licensing:  
--  
-- This code is distributed under the GNU LGPL license.  
--  
-- Modified:  
--  
-- 2012.03.30  
--  
-- 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 bshift is  
  generic (  
    WD      : in natural := 32;  
    SH      : in natural := 5 );  
  
  port (  
    din     : in  std_logic_vector (WD-1 downto 0) := (others=>'0');  
    nbit    : in  std_logic_vector (SH-1 downto 0) := (others=>'0');  
    dout    : out std_logic_vector (WD-1 downto 0) := (others=>'0');  
  
end bshift;
```

```
architecture rtl of bshift is
```

```
begin
```

```
  bshft: process (din, nbit)  
    variable s      : std_logic := '0';  
    variable dinX   : std_logic_vector (2*WD-1 downto 0) := (others=>'0');  
    variable offset : natural := 0;  
    variable result : std_logic_vector (WD-1 downto 0) := (others=>'0');  
  begin -- process bshft  
  
    s := din(WD-1);  
    for i in 2*WD-1 downto WD loop  
      dinX(i) := s;  
    end loop; -- i  
    dinX := dinX(2*WD-1 downto WD) & din (WD-1 downto 0);  
  
    offset := to_integer(unsigned(nbit));  
  
    result := dinX(WD-1+offset downto offset);  
  
    dout <= result;
```

```
end process;
```

```
end rtl;
```