```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL; use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity mod25 is
Port (rst: in STD_LOGIC;
pr: in STD_LOGIC;
clk: in STD_LOGIC;
dir: in STD_LOGIC;
Q: out STD_LOGIC_VECTOR (4 downto 0)); end mod25;
architecture mod25_arch of mod25 is signal Qtemp: STD_LOGIC_VECTOR (4 downto 0) := "00000";
begin
process(rst,pr,clk, dir)
begin
if rst ='1' then Qtemp <= (OTHERS =>'0');
elsif pr='1' then Qtemp <= (OTHERS =>'1');
elsif falling_edge(clk) then if dir = '1' then
if Qtemp < 24 then Qtemp <= Qtemp + 1; Qtemp <= "00000"; else end if;
```

```
else

if Qtemp > 7 then Qtemp <= Qtemp -1; Qtemp <= "11111"; else end if;

end if;

end if;

end process; Q<=Qtemp;

end mod25_arch;

tbb

--Inputs
signal rst std_logic := '0'; signal pr: std_logic := '0'; signal clk: std_logic := '0';

signal dir: std_logic := '0';

--Outputs signal Q: std_logic_vector(4 downto 0);

- Clock period definitions constant clk_period: time := 10 ns;

BEGIN

-- Instantiate the Unit Under Test (UUT) uut: mod25 PORT MAP (

rst => rst,

pr => pr,
clk => clk, dir => dir, QQ );

- Clock process definitions clk_process process

begin

clk <= '0';

wait for clk_period/2;

clk <= '1';

wait for clk_period/2;

end process;

- Stimulus process

stim_proc_dir: process begin

dir <= not(dir);

wait for 320 ns;

end process;

stim_proc_rst: process

begin
```

```
wait for 680 ns;

rst <= '1'

wait for 40 ns;

rst <= '0';

wait;

end process;

stim_proc_pr: process

begin

wait for 750 ns;

pr <= '1';

wait for 40 ns;

pr <= '0';

wait;

end process;

END;
```