POROČILO

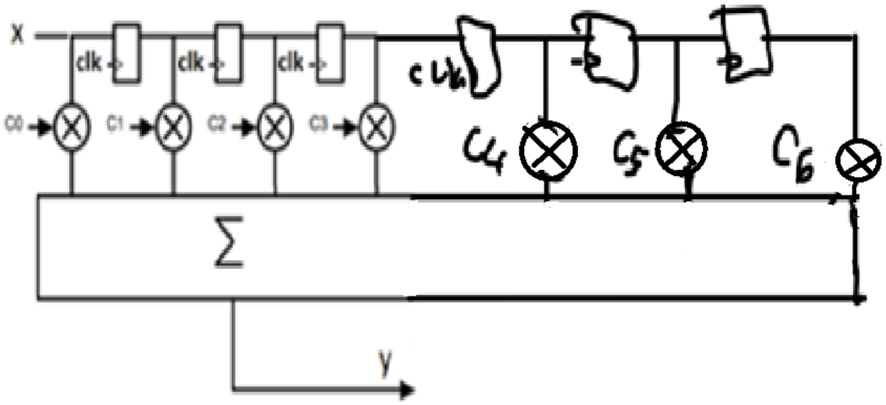
1. Fir.vhdl

Kažem vam kodo modela pasovnega sita 7. Reda (za 7 koeficientov), ki zaduši nizke in zelo visoke frekvence. Sito ima simetrične koeficiente, med katerimi so tokrat tudi negativne vrednosti

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

entity fir is
 port (
   clk : in std_logic;
   x : in signed(7 downto 0);
   y : out signed(7 downto 0) );
end fir;

architecture RTL of fir is
 constant c0 : signed(9 downto 0) := to_signed(52,10);
 constant c1 : signed(9 downto 0) := to_signed(-164,10);
 constant c2 : signed(9 downto 0) := to_signed(-120,10);
 constant c3 : signed(9 downto 0) := to_signed(468,10);
 constant c4 : signed(9 downto 0) := to_signed(-120,10);
 constant c5 : signed(9 downto 0) := to_signed(-164,10);
 constant c6 : signed(9 downto 0) := to_signed(52,10);
 signal x1 : signed(7 downto 0) := "00000000";
 signal x2 : signed(7 downto 0) := "00000000";
 signal x3 : signed(7 downto 0) := "00000000";
 signal x4 : signed(7 downto 0) := "00000000";
 signal x5 : signed(7 downto 0) := "00000000";
 signal x6 : signed(7 downto 0) := "00000000";
 signal sum : signed(17 downto 0);
begin
sum <= ((((((c0*x) + (c1*x1)) + (c2*x2)) + (c3*x3)) + (c4*x4)) +
(c5*x5)) + (c6*x6);
y <= sum(17 downto 10);

process(clk)
begin
 if rising_edge(clk) then
  x6 <= x5;
  x5 <= x4;
  x4 <= x3;
  x3 <= x2;
  x2 <= x1;
  x1 <= x;
 end if;
end process;

end RTL;
```

2. **Enačba je $y[n]$ = (1 /1024)($c0 \cdot x[n]$ + $c1 \cdot x[n-1]$ + $c2 \cdot x[n-2]$ + $c3 \cdot x[n-3]$ + c4*x[n-4] + c5*x[n-5] + c6*x[n-6])**

3. Zgradba pasovnega sita (blokovno shemo)

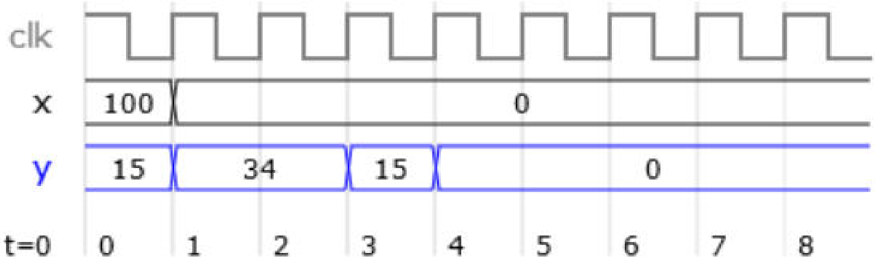


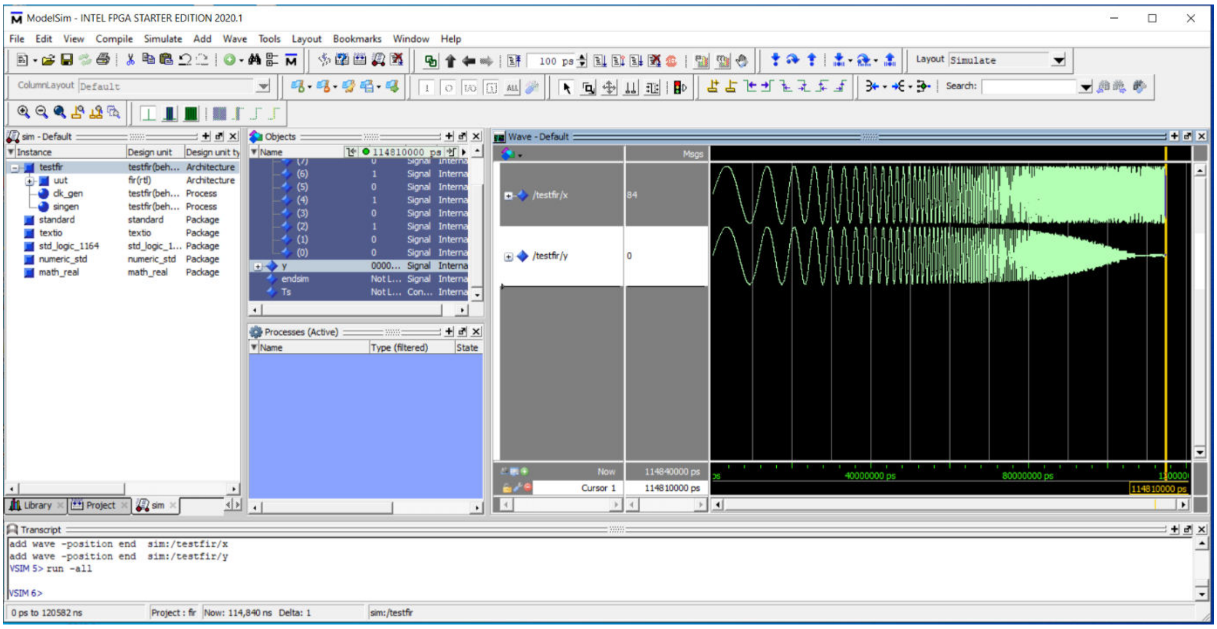
Logični gradniki iz katerih je vezje sestavljeno
Digitalno sito s končnim odzivom FIR naredimo iz zakasnilnih elementov (6 flip flopov), množilnikov (7) in seštevalnika 1. Imamo vhodni signal x, izhodni y in notranji c0-c6. Sliko sem risal z risarjem in z miško, pomembno mi je bilo da se vidi zgradbo.

4. Simulacija odziva na impulz in odziva na sinusni signal pri različnih frekvencah

Opis digitalnega sita v jeziku VHDL smo prenesli v program ModelSim in naredili simulacijo s

testno strukturo, ki generira frekvenčni prelet. Tega v sami spletni strani in modelu shdl ne moremo, zato smo uporabili testno strukturo test fir, ki jo pridajam odspodaj





Instance	Design unit	Design unit ty
testfir	testfir(beh...	Architecture
uut	fir(rtl)	Architecture
clk_gen	testfir(beh...	Process
singen	testfir(beh...	Process
standard	standard	Package
textio	textio	Package
std_logic_1164	std_logic_1...	Package
numeric_std	numeric_std	Package
math_real	math_real	Package

Name	114810000 ps		
(7)	0	Signal	Interna
(6)	1	Signal	Interna
(5)	0	Signal	Interna
(4)	1	Signal	Interna
(3)	0	Signal	Interna
(2)	1	Signal	Interna
(1)	0	Signal	Interna
(0)	0	Signal	Interna
y	0000...	Signal	Interna
endsim	Not L...	Signal	Interna
Ts	Not L...	Con...	Interna

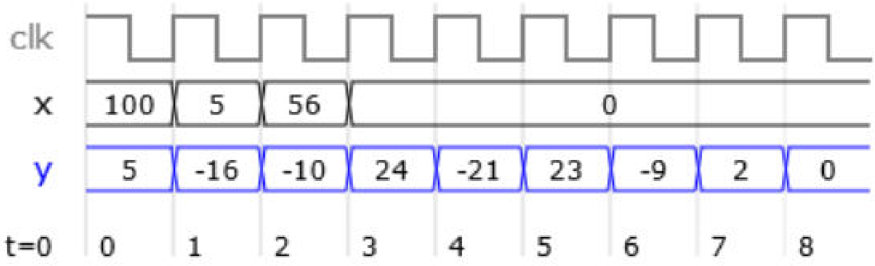
Name	Type (filtered)	State

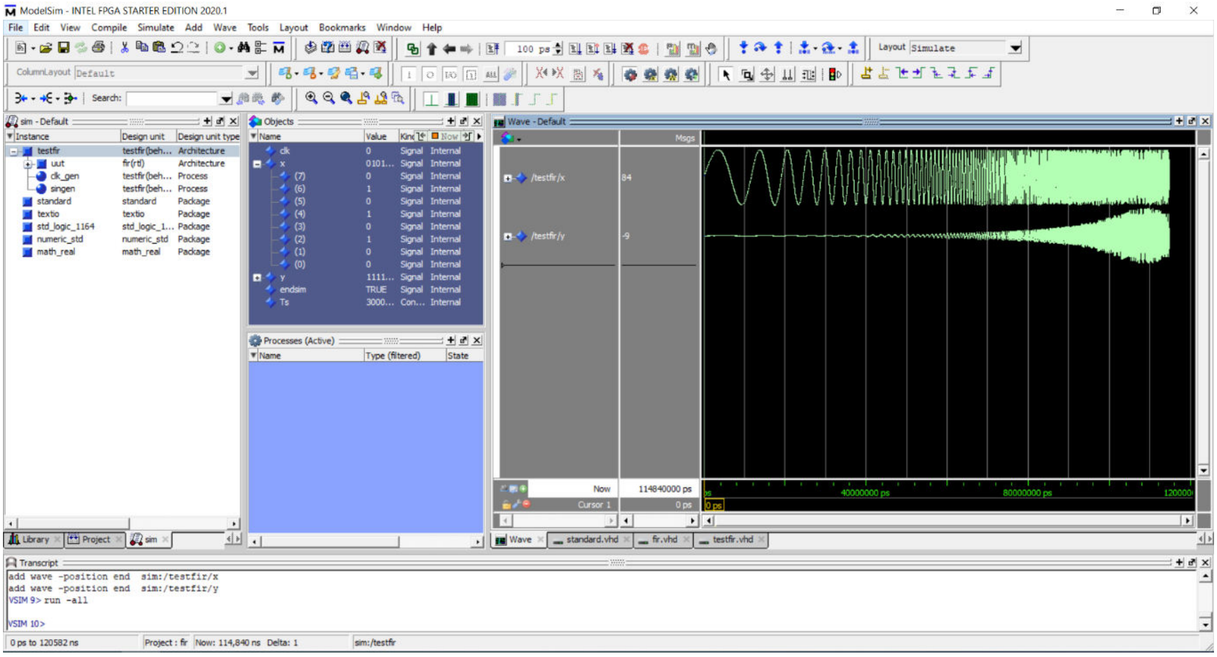
	Msgs
/testfir/x	84
/testfir/y	0
Now	114840000 ps
Cursor 1	114810000 ps

ps 40000000 ps 80000000 ps 1100000 114810000 ps

```
add wave -position end  sim:/testfir/x
add wave -position end  sim:/testfir/y
VSIM 5> run -all

VSIM 6>
```

0 ps to 120582 ns | Project : fir | Now: 114,840 ns Delta: 1 | sim:/testfir





Instance	Design unit	Design unit type
testfir	testfir(beh...	Architecture
uut	fir(rtl)	Architecture
clk_gen	testfir(beh...	Process
singen	testfir(beh...	Process
standard	standard	Package
textio	textio	Package
std_logic_1164	std_logic_1...	Package
numeric_std	numeric_std	Package
math_real	math_real	Package

Name	Value	Kind	Now
clk	0	Signal	Internal
x	0101...	Signal	Internal
(7)	0	Signal	Internal
(6)	1	Signal	Internal
(5)	0	Signal	Internal
(4)	1	Signal	Internal
(3)	0	Signal	Internal
(2)	1	Signal	Internal
(1)	0	Signal	Internal
(0)	0	Signal	Internal
y	1111...	Signal	Internal
endsim	TRUE	Signal	Internal
Ts	3000...	Con...	Internal

Name	Type (filtered)	State

```
add wave -position end  sim:/testfir/x
add wave -position end  sim:/testfir/y
VSIM 9> run -all

VSIM 10>
```

0 ps to 120582 ns | Project : fir | Now: 114,840 ns Delta: 1 | sim:/testfir

Datoteka testfir.vhdl

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.numeric_std.ALL;
use IEEE.math_real.all;

entity TestFIR is
end TestFIR;

architecture Behavioral of TestFIR is

 signal clk : std_logic:= '0';
 signal x : signed(7 downto 0) := (others=>'0');
 signal y : signed(7 downto 0);
 constant Ts : time := 30 ns;
 signal endsim: boolean := false;
begin

uut: entity work.fir port map (
     clk => clk,
     x => x,
     y => y
);

-- Clock generator
clk_gen: process
begin
 if not endsim then
 clk <= '1';  wait for Ts/2;
 clk <= '0';  wait for Ts/2;
 else
   wait;
 end if;
end process;

-- Generiraj prelet sinusnega signala za vhod vezja
singen : process(clk)
 variable ods: integer := 0;  -- stevec odsekov
 variable fi: real:= 0.0;
 variable d: real := 0.002;  -- zacetni korak
 variable si: integer;
begin
 if(rising_edge(clk)) then
  if ods > 64 then   -- po 64 ciklih je nov odsek
   ods := 0;
   d := d * 1.1;
   if d >= 0.5 then
     endsim <= true;
   end if;
  else
   ods := ods + 1;
  end if;

  fi := fi + d;
  si := integer(100.0*sin(MATH_2_PI*fi)+0.49);  -- sinus, amp=100
  x <= to_signed(si, 8);
 end if;
end process;

end Behavioral;
```