

מעבדה לVHDL – מחלק תדר

תחילה כתבנו תוכנית בעלי משתני generic אשר הם יקבעו את הזמן מחזור ואת המחזור פעולה (Duty Cycle).

הרעיון בתוכנית הוא לקחת אות שעון בתדר 50[MHz] ולחשב כמה עליות שעון צריך לספור כדי לעשות השהייה של שנייה אחת (עבור תדר 1[Hz]) והשהייה של 100[msec] (עבור תדר 10[Hz]).

תחילה חישבנו כמה זמן לוקח לclk בעל תדר 50[MHz] לבצע עלייה (זמן מחזור שלם):

$$f_{clk_{in}} = \frac{1}{T_{clk_{in}}} \rightarrow 50[MHz] = \frac{1}{T_{clk_{in}}} \rightarrow T_{clk_{in}} = 20[nsec]$$

כעת חישבנו כמה פעמים עלינו לספור את הזמן הזה כדי להגיע לזמן של שנייה אחת:

$$Tick = \frac{T}{T_{clk_{in}}} = \frac{1[sec]}{20[nsec]} = f_{clk_{in}} = 50,000,000[Ticks]$$

קיבלנו את כמות העליות שעון שעלינו לספור בכדי להגיע להשהייה של שנייה אחת.

כעת עבור D.C של 50[%] נספור חצי מהזמן שעבורו המחזור יהיה במצב '1' לוגי.

תוכנית:

```

library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity Generator_nHz is
generic(N: integer:= 50000000;
        DC: integer:= 25000000);
port(
clk_in: in std_logic;
clk_out: buffer std_logic:= '1');
end;

architecture one of Generator_nHz is
signal count: integer range 0 to N := 0;
begin
process(clk_in)
begin
if (clk_in 'event and clk_in = '1') then
    if (count = N) then
        count <= 0;
        clk_out <= '1';
    elsif (count = DC) then
        count <= count + 1;
        clk_out <= '0';
    else
        count <= count + 1;
    end if;
end if;
end process;
end;

```

הרצויים:

תוכנית:

```

library ieee;
use ieee.std_logic_1164.all;

entity Generator is
port(
sel: in std_logic_vector(1 downto 0);
clk1in: in std_logic;
clkout: buffer std_logic;
x: out std_logic);
end;

architecture one of Generator is

-- / Components \ --
component Generator_nHz
generic(N: integer:= 5000000;
        DC: integer:= 2500000);
port(
clk1in: in std_logic;
clkout: buffer std_logic:='0');
end component;

-- / Components \ --

-- / Signals \ --
signal clk1,clk2,clk3,clk4: std_logic;
-- / Signals \ --

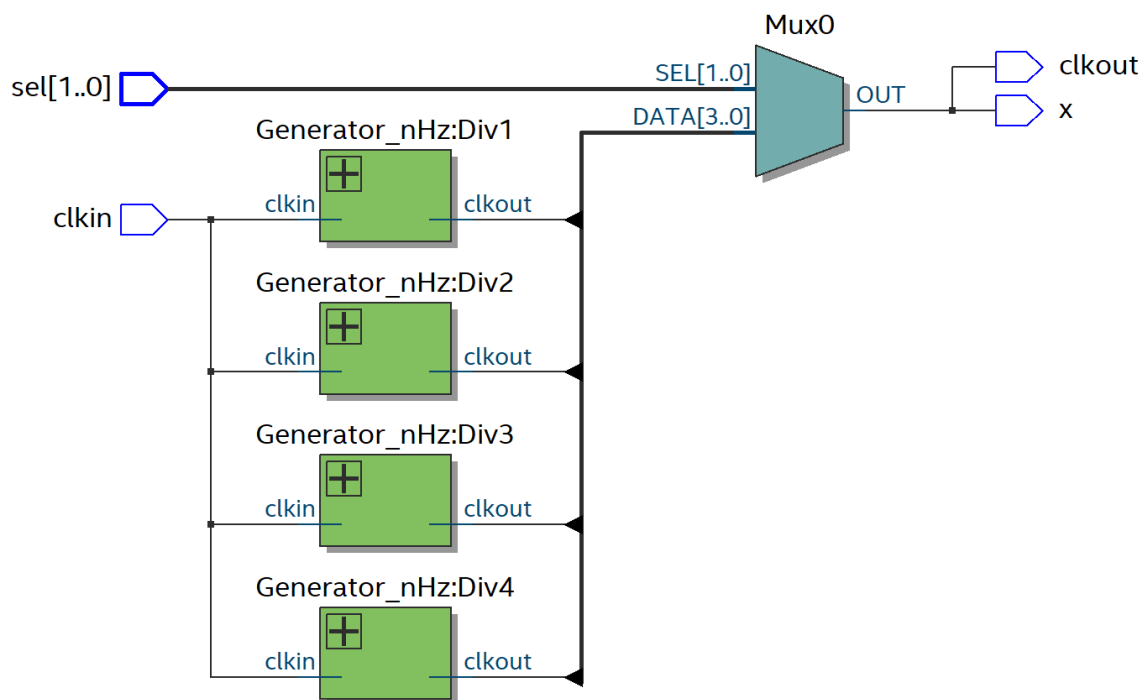
begin
Div1: Generator_nHz generic map (5000000, 2500000) port map (clk1in,clk1); -- freq = 1[Hz] , DC = 50[%]
Div2: Generator_nHz generic map (5000000, 1250000) port map (clk1in,clk2); -- freq = 1[Hz] , DC = 25[%]
Div3: Generator_nHz generic map (5000000, 250000) port map (clk1in,clk3); -- freq = 10[Hz], DC = 50[%]
Div4: Generator_nHz generic map (5000000, 375000) port map (clk1in,clk4); -- freq = 10[Hz], DC = 75[%]

with sel select
clkout <= clk1 when "00",
          clk2 when "01",
          clk3 when "10",
          clk4 when "11";

x <= clkout;
end;

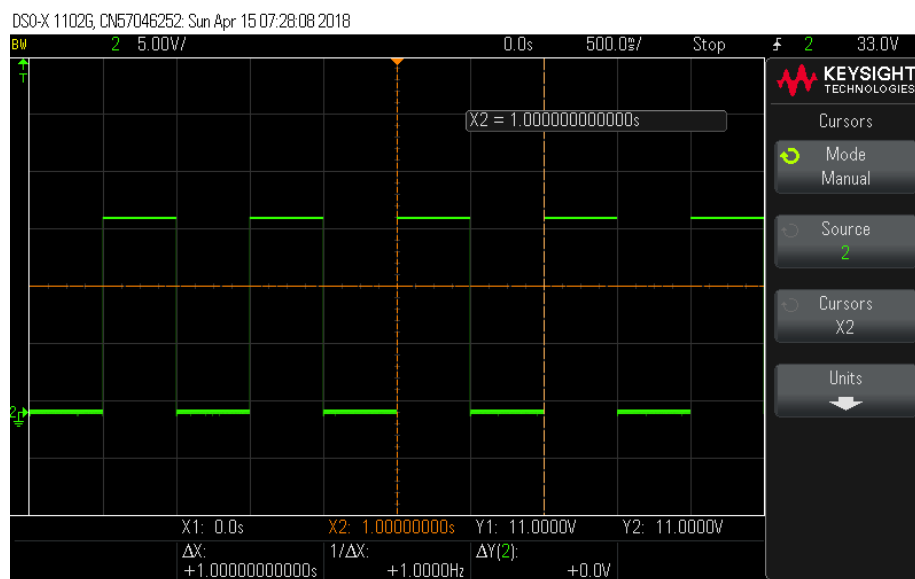
```

:RTL

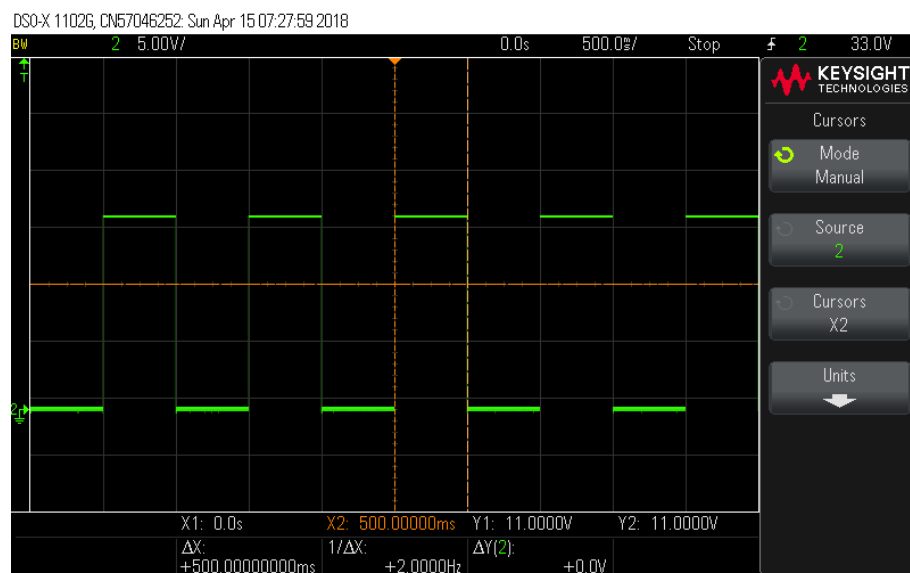


לכידת האותות בScope:

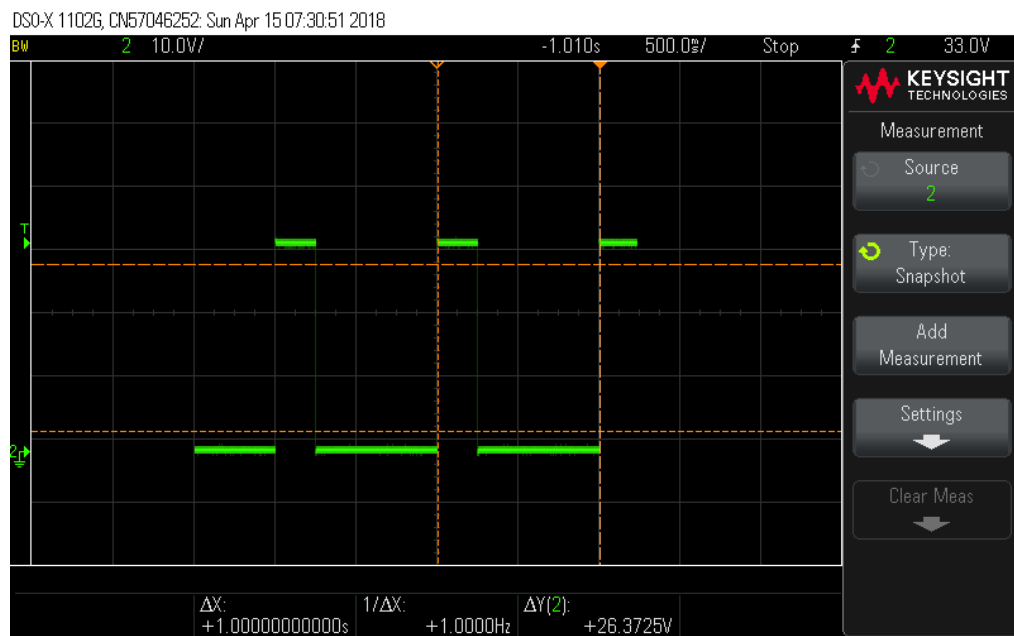
Div1 - 1[Hz] (D.C. = 50[%]) – Full period measure



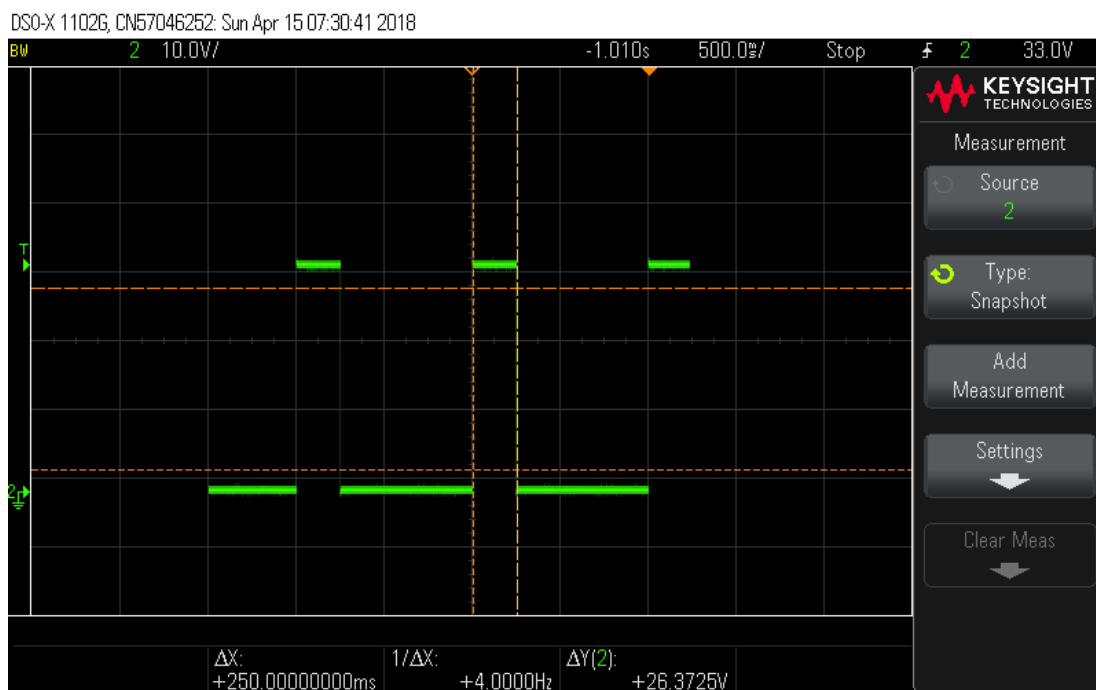
Div1 - 1[Hz] (D.C. = 50[%]) – Pulse measure



Div2 – 1[Hz] (D.C = 25[%]) – Full period measure



Div2 – 1[Hz] (D.C = 25[%]) – Pulse measure



Div3 – 10[Hz] (50[%]) – Full period measure

DSO-X 1102G, CN57046252: Sun Apr 15 07:25:11 2018

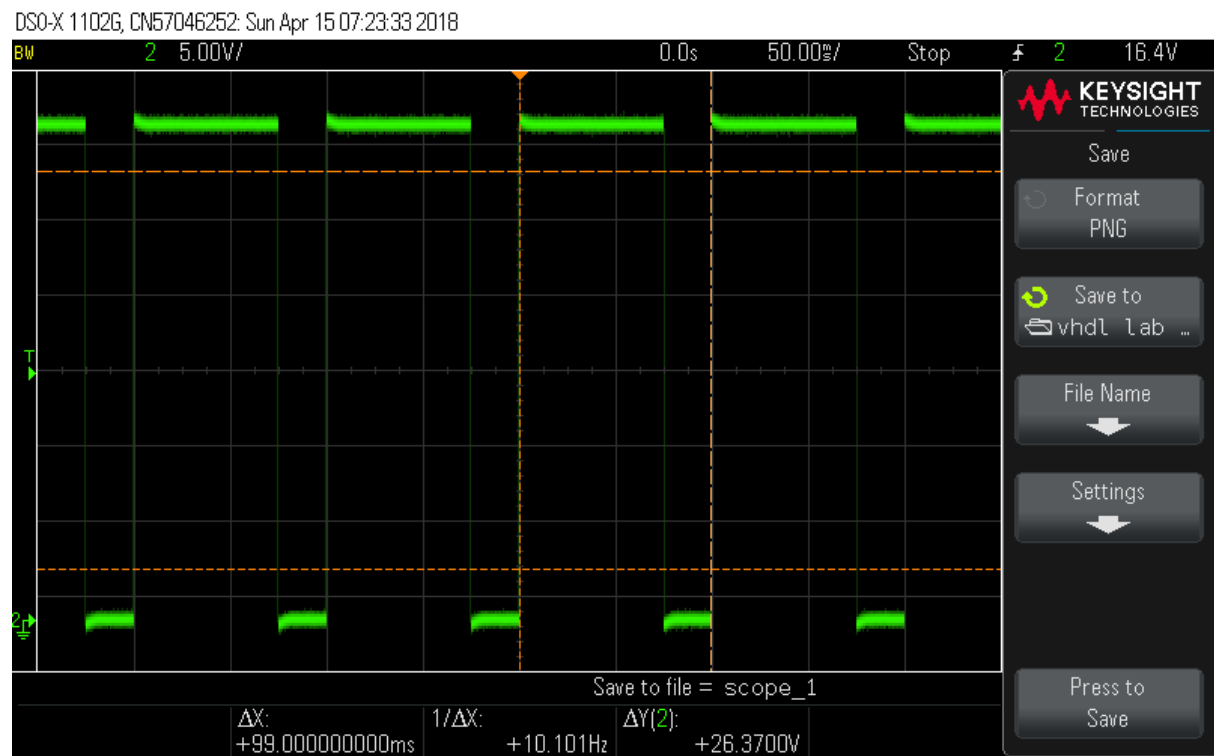


Div3 – 10[Hz] (50[%]) – Pulse measure

DSO-X 1102G, CN57046252: Sun Apr 15 07:24:52 2018



Div4 – 10[Hz] (D.C=75[%]) – Full period measure



Div4 – 10[Hz] (D.C = 75[%]) – Pulse measure

