



UNIVERZITET U TUZLI
FAKULTET ELEKTROTEHNIKE



GHDL

Dr. Sc. Asmir Gogić, vanr. prof.

Tuzla, 2022.

GHDL - uvod

- **GHDL** - G Hardware Design Language
- ...analizator, kompajler i simulator VHDL koda prema IEEE 1076 standardu.
- ...**ne omogućuje sintetiziranje/generiranje** bitstream-a.
- Znatno brži od ostalih kompajlera jer ne koristi posredničke jezike za generiranje rezultat među koraka tokom transliranja VHDL koda u mašinski kod.
- **Ne uključuje grafički preglednik** za talasne oblike signala. Rezultirajući signali *.ghw, *.vcd ili *.fst file-ovi se mogu analizirati koristeći alat kao što je **GtkWave**
- dostupan za **Linux, Windows, MAC iOS** operative sisteme

GHDL - sintaksa poziva

- **ghdl** [-switch] files
- **-a** - sprovodi analizu sintakse nad datim VHDL file-om, kompajlira sve biblioteke i pakete u kod koji se koristi interno.
- **-t** - generira izvršni mašinski kod za top entity.
- **-r** - generira talasne oblike za predefinirani ulaze.
- **GHDL** proces **MORA** imati listu osjetljivih signala! U suprotnom GHDL kompajler neće moći generirati izvršni mašinski kod.
- U cilju redukovanja unosa datih komandi kreiran je generički **Makefile** koji se može koristiti kako bi se cijeli proces automatizirao.
- Za instalaciju ghdl na Linux baziranim OS iskoristiti skriptu
sh install-ghdl.sh
sudo apt-get install gtkwave

GHDL - primjeri

PRIMJER 1

Napisati VHDL kod koji će implementirati 4-bitni binarni brojač. Funkcionalnost modula testirati koristeći GHDL kompajler i GtkWave scope.

GHDL - debugging in simulation

- Ispis stringa izvodimo sa instrukcijom **report** koja ima opštu formu:

```
report "ovo je test string";
```

- Ispis vrijednosti varijable ili signala tipa integer (varijabla fclkcnt u našem slučaju) u decimalnoj notaciji izvodimo na sljedeći način:

```
report "mul op done in: " & integer'image(fclkcnt/2) & " CLK's@500MHz" ;
```

dok ispis signala ili varijable tipa unsigned (utmp signal) izvodimo kao:

```
report "beg T" & integer'image(to_integer(utmp));
```

- Ispis varijable tipa *float16* (S1 + E5 + M10):

```
report "A("&integer'image(i)&","&integer'image(j)&") = "  
& to_string(to_real(to_float(fvar, 5, 10)),"%0.9f");
```

GHDL -generiranje FPGA takt impulsa

- **Generiranje takt impulsa** karakteriziraju sljedeća svojstva:
 - treba biti **nezavisan proces** (clk),
 - generiranje takt impulsa **izvodimo neograničeno dugo!**
 - **zaustavljanje procesa** generiranja takt impulsa izvodimo **nakon sprovođenja svih testova**, u suprotnom simulacija će trajati jako dugo.
- Zaustavljanje procesa generiranja takt impulsa izvodimo kroz globalni signal čije stanje se provjerava u procesu generiranja impulsa.
- U cilju izbjegavanja situacije gdje procesiranje testnih sekvenci u "glavnom dijelu" *TestBench*-a može da traje neograničeno, generiranje impulsa se ograničava na razumno veliki broj (1M ili više ciklusa).

GHDL -generiranje FPGA takt impulsa

```
-- clk generation process for 500 MHz clock
clk: process
  variable fclk_end: integer:= 5000000;
  variable fclk_cnt: integer:=0;

  begin
    fclk <= '1';
    wait for 1 ns;
    fclk <= '0';
    wait for 1 ns;
    g_fclk_cnt <= g_fclk_cnt + 1;

    fclk_cnt := fclk_cnt + 1;
    -- stop clk generation if the simulation process hangs
    if(fclk_cnt = fclk_end) then
      report "fpga clock exhausted";
      endsim2 <= '1';
      wait;
    end if;

    -- stop the clock generation if simulation process is completed
    if(endsim = '1') then
      report "fpga clock shutdown";
      wait;
    end if;

  end process;
```

GHDL - primjeri

PRIMJER 2

Napisati VHDL kod koji će implementirati 3-bitni binarni komparator. Funkcionalnost modula testirati koristeći GHDL kompajler i GtkWave scope.

PRIMJER 3

Napisati VHDL kod koji će implementirati PWM modulator. Funkcionalnost modula testirati za različite vrijednosti popunjenosti impulsa koristeći GHDL kompajler i GtkWave scope.

Literatura

- RTL Hardware Design Using VHDL: Coding for Efficiency, Portability, and Scalability, 1st Edition by Pong P. Chu, 2006.
- Digital Systems Design Using VHDL 2 nd Edition, by Charles H. Roth, Jr. and Lizy Hurian John, Thomson, 2007.
- The Designer's Guide to VHDL, Third Edition, Peter J. Ashenden, 2008.