

Dr. Bob Davidov

Интеграция VHDL (язык описания аппаратных средств) и Simulink моделей

Цель работы: освоение правил моделирования систем с объектами разного типа.

Задача работы: построение динамической модели включающей VHDL (Verilog) и Simulink объекты.

Приборы и принадлежности: Персональный компьютер с компилятором Verilog и Интегрированная среда МатЛАБ с Simulink.

ВВЕДЕНИЕ

Для описания и построения электронных устройств используется множество языков, среди них и язык VHDL. МатЛАБ позволяет объединить “инородные” VHDL модели электронных устройств и собственные средства моделирования Simulink, что позволяет решать задачи моделирования и синтеза интегрированных систем, например, для целей управления.

ОБЩИЕ СВЕДЕНИЯ

Степень интеграции современных СБИС – «Систем на кристалле» (СнК) достигает нескольких десятков миллионов вентилях на кристалле. СБИС содержат программируемые процессорные ядра, специализированные логические блоки, модули памяти, интерфейсные и периферийные устройства, аналоговые и аналого-цифровые схемы.

Для реализации полного цикла проектирования СнК используется определенный набор программных продуктов и языков программирования способных описывать параллельно протекающие во времени процессы, поддерживать множество стилей описания аппаратуры и инженерных приложений. На **Рис. 1** приведены основные этапы проектирования СнК и различные языки и средства разработки, используемые для выполнения этих этапов.

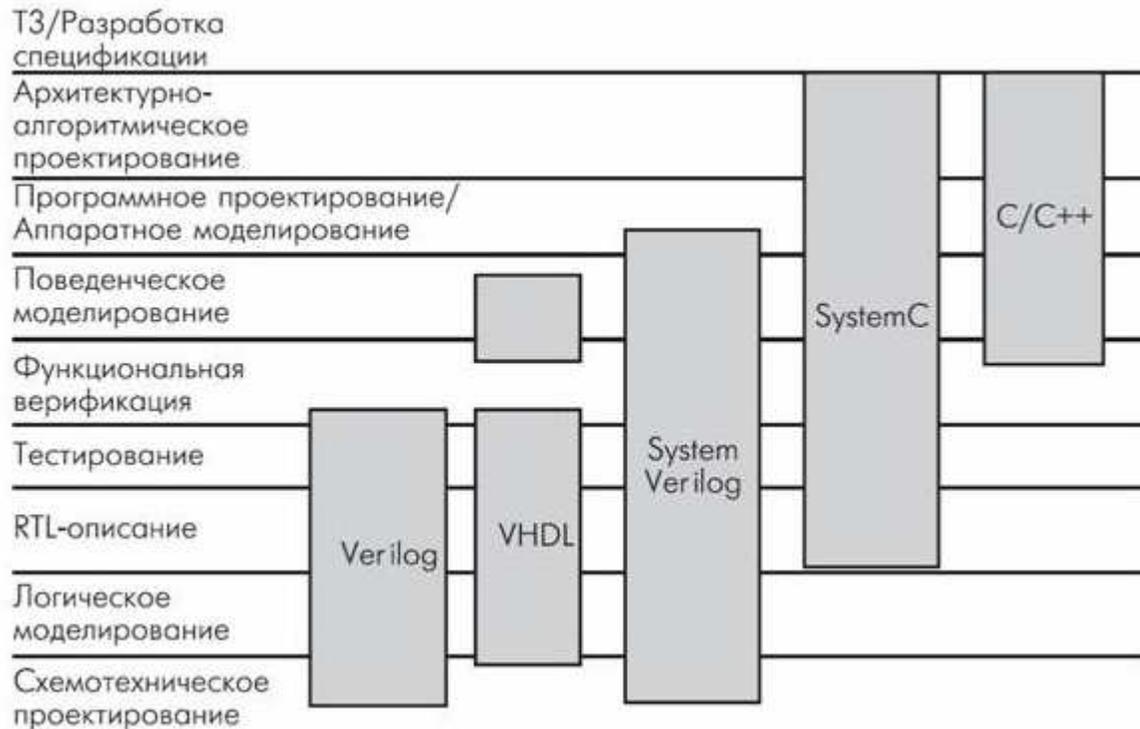


Рис. 1. Этапы проектирования «Систем на кристалле» и области применения современных средств и языков программирования

Языки описания аппаратуры (HDL-языки) имеют две основные разновидности – языки низкого уровня (аналоги языков программирования типа ассемблера) и высокого уровня.

Языки низкого уровня ближе к аппаратным средствам, вследствие чего представляют для компиляторов потенциальные возможности создания проектов с более выигрышными параметрами. Платой за это является обычно жесткая ориентация на определенную аппаратуру и производящую ее фирму. Примерами таких языков могут служить языки PLDASM (фирма Intel), AHDL (Фирма Altera) и ABEL (Фирма Zilinx).

Языки высокого уровня менее связаны с аппаратными платформами и поэтому более универсальны. Среди них наиболее распространены языки VHDL и Verilog.

MatLAB обеспечивает соединение Simulink и Verilog модулей через TCP/IP на одном или удаленных компьютерах.

ПОРЯДОК СОЗДАНИЯ МОДЕЛИ SIMULINK - VERILOG

1. Запустите MatLAB
2. Настройте MatLAB на рабочую папку.
3. В рабочем каталоге откройте (или создайте) модель с ModelSim блоком, например, valve_solenoid_control.mdl

4. Установите ModelSim командой setupmodelsim ('PropertyName', 'PropertyValue'...), например,

```
>> setupmodelsim
Identify the ModelSim installation to be configured for MATLAB and Simulink
Do you want setupmodelsim to locate installed ModelSim executables [y]/n? y
Select a ModelSim installation:
[1] C:\Modeltech_6.0c\win32      ModelSim SE 6.0c
[0] None

>>[1]
Selected Modelsim installation: 1
Previous MATLAB startup file found in this installation of ModelSim:
C:\Modeltech_6.0c\win32\..\tcl\ModelSimTclFunctionsForMATLAB.tcl
Do you want to replace this file [y]/n? y
Modelsिम successfully configured for MATLAB and Simulink

>> [y]
```

5. Для запуска ModelSim наберите следующую команду к окну команд MatLAB:

```
>> vsim('tclstart', 'do plot.do','socketsimulink', 4449)
```

6. Запустите модель Simulink на выполнение

Примечание: Для последующих запусков четвертый пункт (setupmodelsim) выполнять не надо.

РЕДАКТИРОВАНИЯ VERILOG ФАЙЛА:

1. Запустите ModelSim
2. Настройте текущую папку
3. Отредактируйте *.v файл, например, VLV_SC.v
4. Создайте VHDL файл командой ModelSim: wrapverilog VLV_SC

УСТАНОВКА ВРЕМЕННЫХ ПАРАМЕТРОВ

1. Величина шага моделирования (Clock) в ModelSim задается параметром "resolution" в файле modelsim.ini
2. Чтобы установить период моделирования Simulink равный периоду ModelSim генератора (SYS_CLK) щелкните по блоку ModelSim в Simulink модели и выберите "Clocks" > "Period"
3. Ввод параметров моделирования Simulink осуществляется в окне, показанном на следующем рисунке.

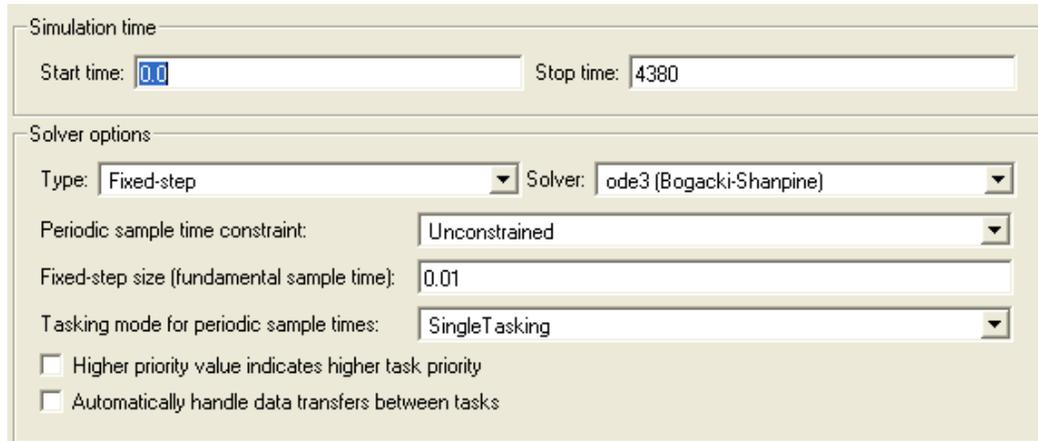


Рис. 2. Параметры моделирования Simulink.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Задание 1. Совместная работа ModelSIM и Simulink модулей

1. В среде ModelSim создайте файл, например, CNS1.v

```
//-----
// Design unit : PRED_O2_VLV_CTRL
// Filename: CNS.v
//
// Description:
//
//      Limitations:
//
// Author: Bob D
//      Ac Ltd.
//
// Simulator: ModelSim 6.0c
// Version 1.0 by 13th April 2006, original release
//-----
// CNS = 100/2^((1600 - PPO2_SNS)/100)
module CNS( S_CLK, PPO2_SNS, CNS_OUT );

input S_CLK;
input [15:0] PPO2_SNS;      // measured PPO2 in mATA
output CNS_OUT;

//parameter PPO2_SNS = 940;    // range is 940 - 1600
parameter PPO2_SAT = 1600;    // max PPO2 in mATA
parameter A = 25600;
reg [7:0] CNS_OUT = 0;

reg [9:0] ER;                // range is 0 - 660
```

```

reg [2:0] ER_INT;           // range is 0 - 7
reg [6:0] ER_FRACT;       // range is 0 - 99
reg [7:0] ER_FRACT_NUM;
reg [14:0] B_SHIFT_1;
reg [13:0] B_SHIFT_2;
reg [14:0] B_SUM;

always @(posedge S_CLK)
//initial
begin
  if (PPO2_SNS >= PPO2_SAT)
    ER = 1;
  else
    begin
      if (PPO2_SNS < 940)
        begin
          ER = 660;
        end
      else
        ER = PPO2_SAT - PPO2_SNS[13:0];
    end

  // $display("ER = %d %b", ER, ER);
  ER_INT = ER/100;
  // $display("ER_INT = %d %b", ER_INT, ER_INT);
  ER_FRACT = ER - 100*ER_INT;
  // $display("ER_FRACT = %d %b", ER_FRACT, ER_FRACT);
  ER_FRACT_NUM = ER_FRACT*100/39;
  // $display("ER_FRACT_NUM = %d %b", ER_FRACT_NUM, ER_FRACT_NUM);
  B_SHIFT_1 = 256 << ER_INT;
  // $display("B_SHIFT_1 = %d %b", B_SHIFT_1, B_SHIFT_1);
  B_SHIFT_2 = ER_FRACT_NUM << ER_INT;
  // $display("B_SHIFT_2 = %d %b", B_SHIFT_2, B_SHIFT_2);
  B_SUM = B_SHIFT_1 + B_SHIFT_2;
  // $display("B_SUM = %d %b", B_SUM, B_SUM);
  CNS_OUT = A/B_SUM;
  // $display("CNS_OUT = %d", CNS_OUT);
end

endmodule

```

2. Командой `wrapverilog < CNS1.v >` создайте оболочку файла.

```

-----
-- Module cns VHDL Wrapper
--
-- Generated by The MathWorks wrapverilog tcl command
--
-- Generated on: 2006-04-13 01:28:46

```

```

--
-----
LIBRARY IEEE;
USE IEEE.std_logic_1164.all;

ENTITY cns_wrap IS

    generic(
        PPO2_SAT    : integer := 1600;
        A           : integer := 25600
    );
    port(
        S_CLK       : in  std_logic;
        PPO2_SNS    : in  std_logic_vector(15 downto 0);
        CNS_OUT     : out std_logic_vector(7 downto 0)
    );

END cns_wrap;

ARCHITECTURE rtl OF cns_wrap IS

    component CNS
        generic(
            PPO2_SAT    : integer := 1600;
            A           : integer := 25600
        );
        port(
            S_CLK       : in  std_logic;
            PPO2_SNS    : in  std_logic_vector(15 downto 0);
            CNS_OUT     : out std_logic_vector(7 downto 0)
        );
    end component;

    FOR ALL : cns
        USE ENTITY work.cns(ignored);

    BEGIN
        u_cns: cns
            PORT MAP (
                S_CLK => S_CLK ,
                PPO2_SNS => PPO2_SNS ,
                CNS_OUT => CNS_OUT
            );

    END rtl;

```

3. Загрузите МатЛАБ.

4. Настройте Simulink как показано ниже.

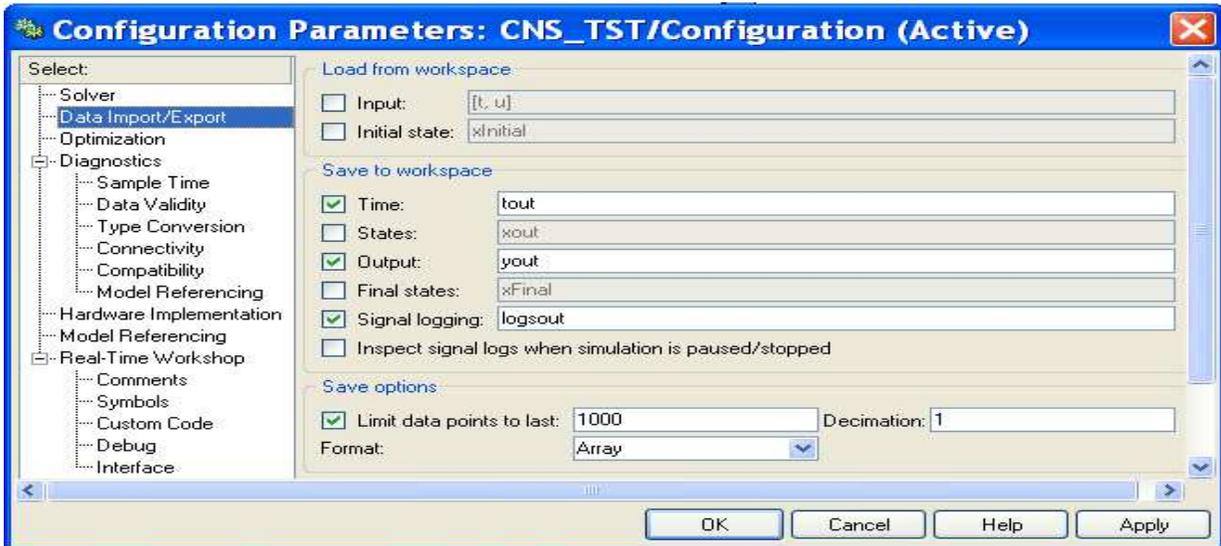
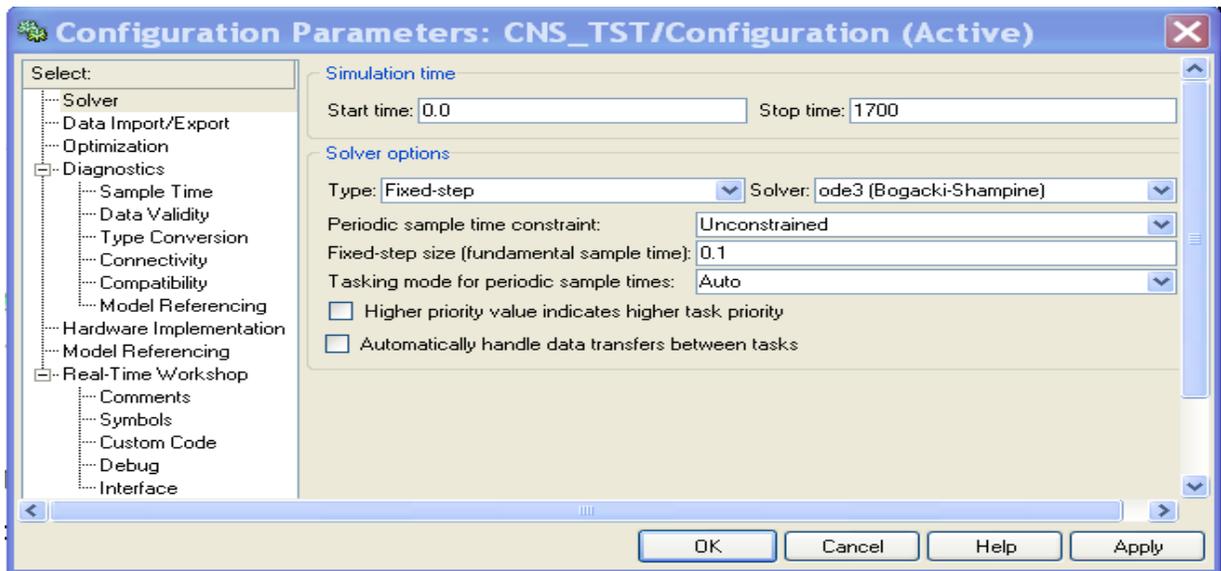


Рис. 3. Параметры конфигурации Simulink. Параметры конфигурации Simulink.

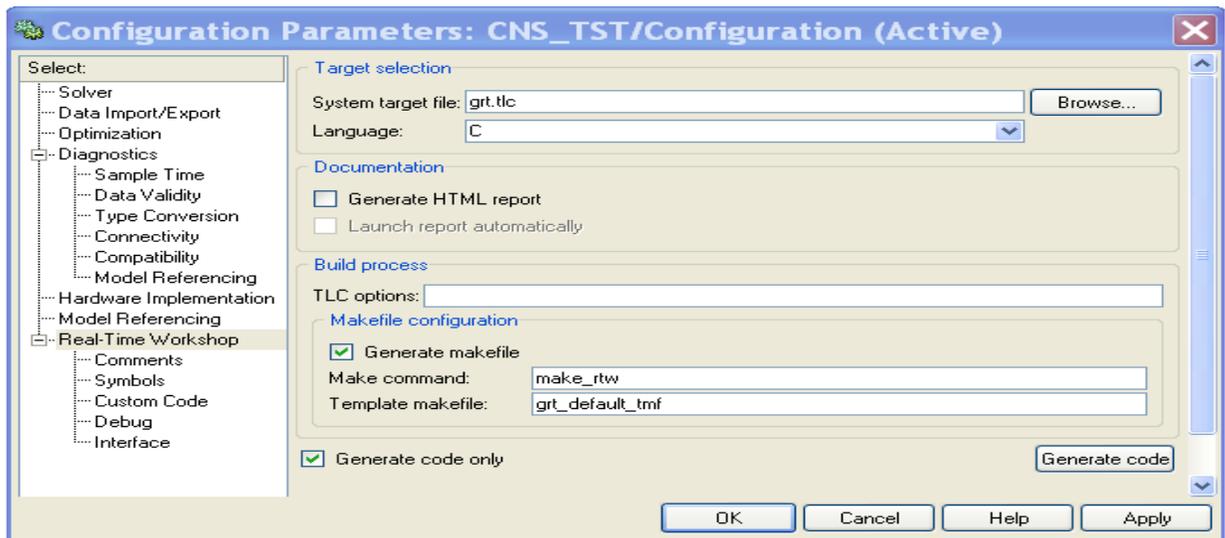
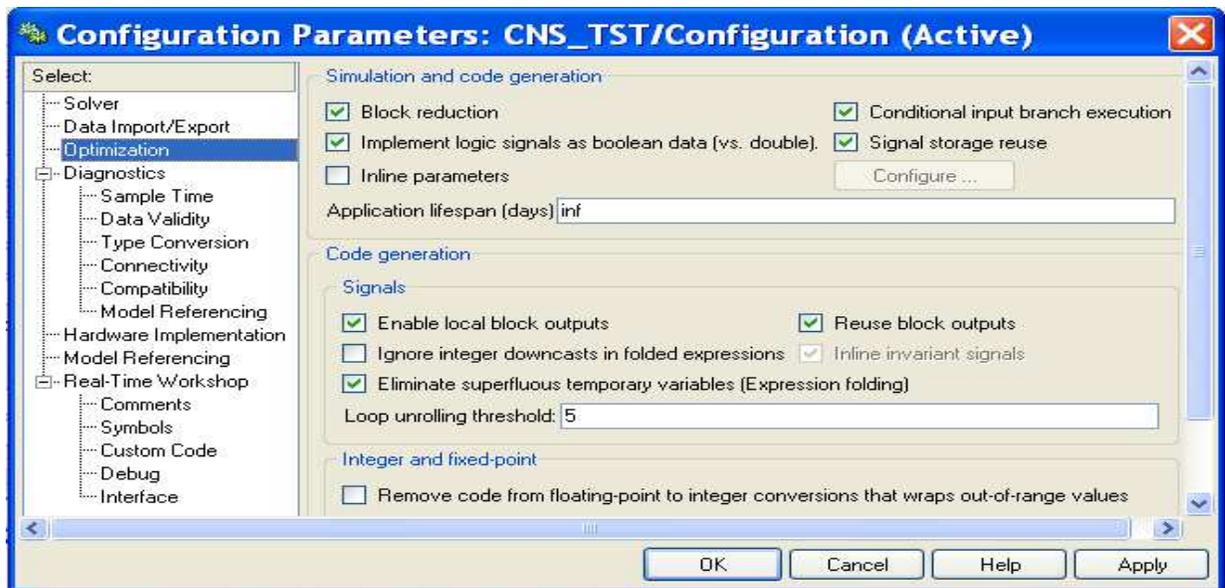


Рис. 4. Параметры конфигурации Simulink.

5. В рабочем каталоге постройте модель CNS_TST.mdl

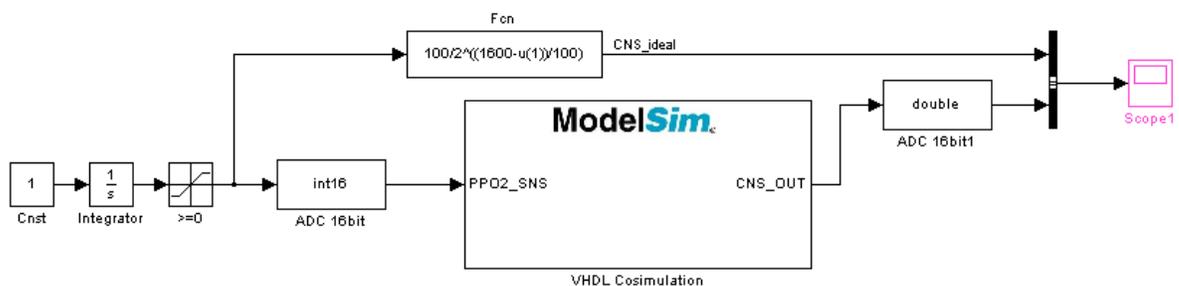
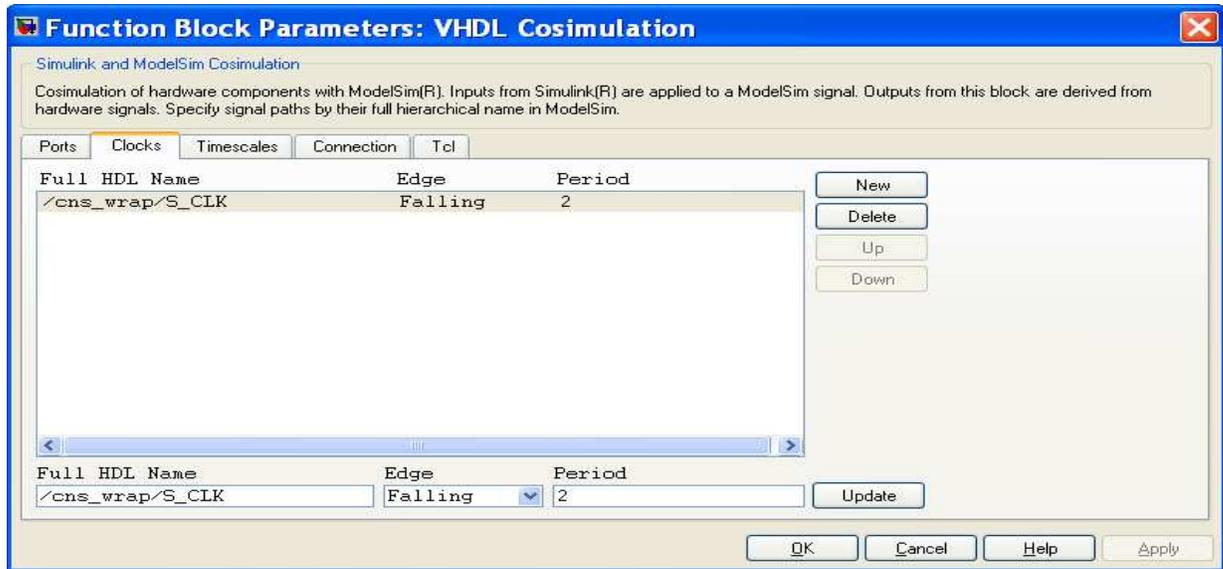
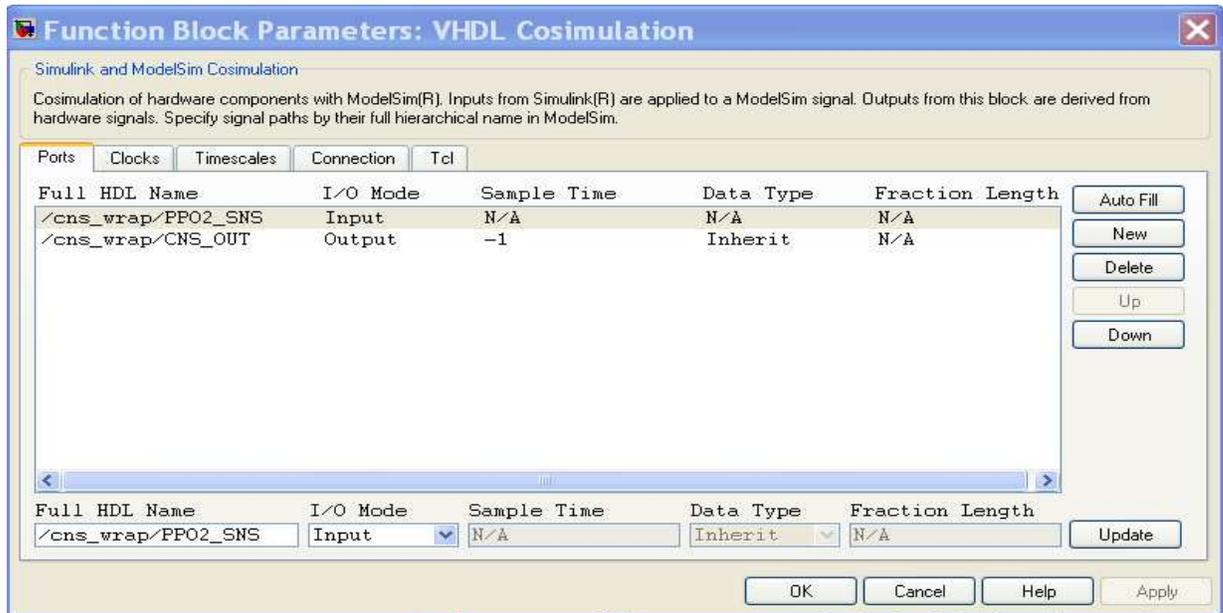
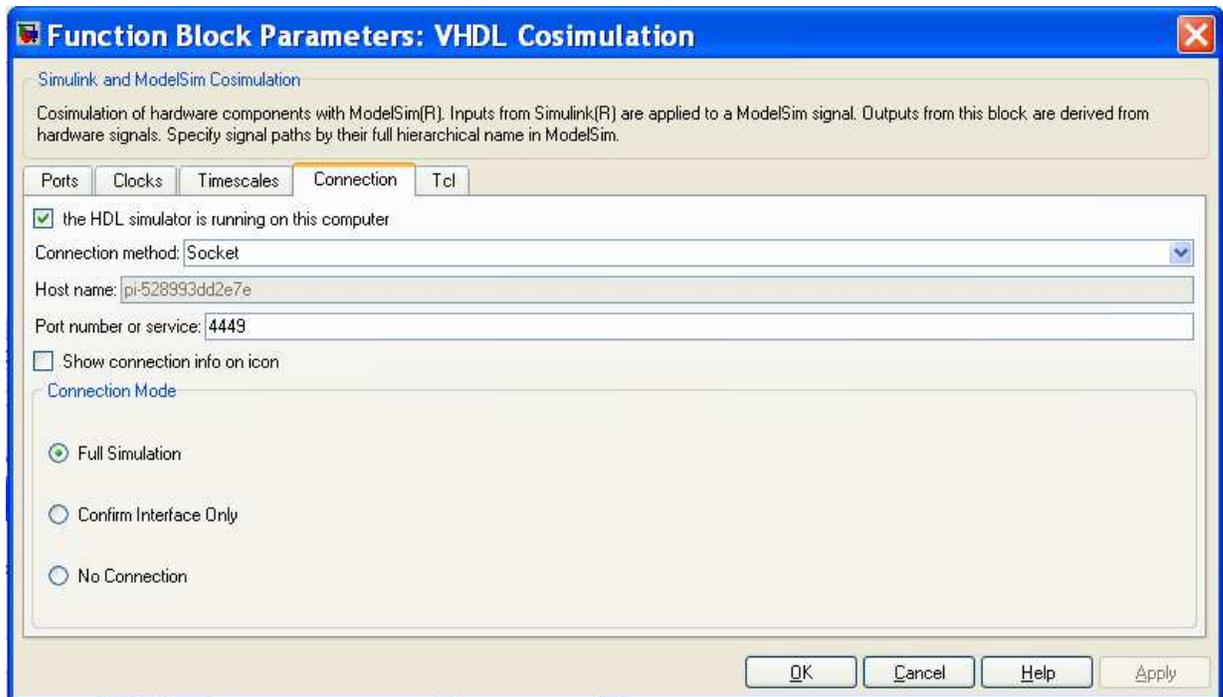
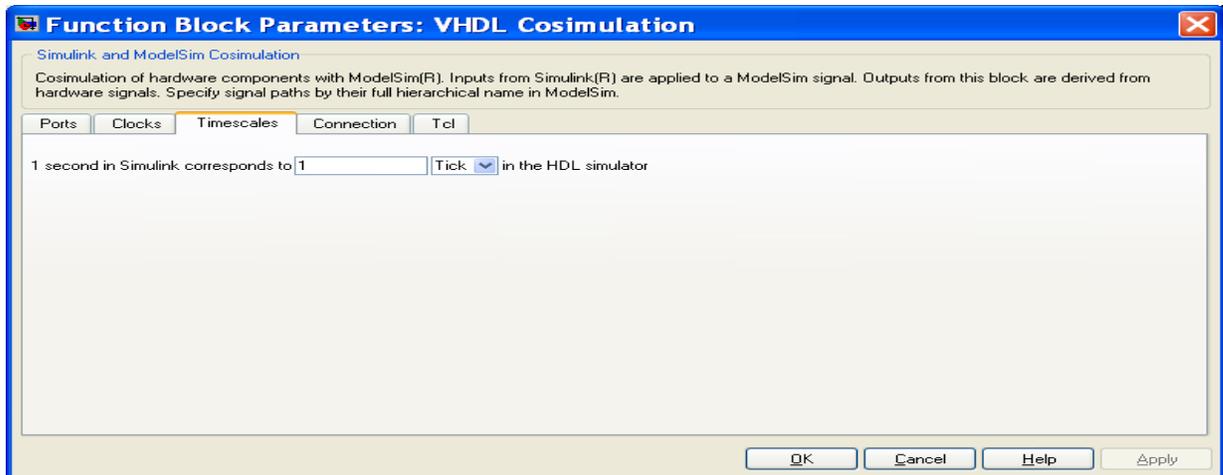


Рис. 5. Модель CNS_TST.mdl

6. Настройте блок VHDL Cosimulation





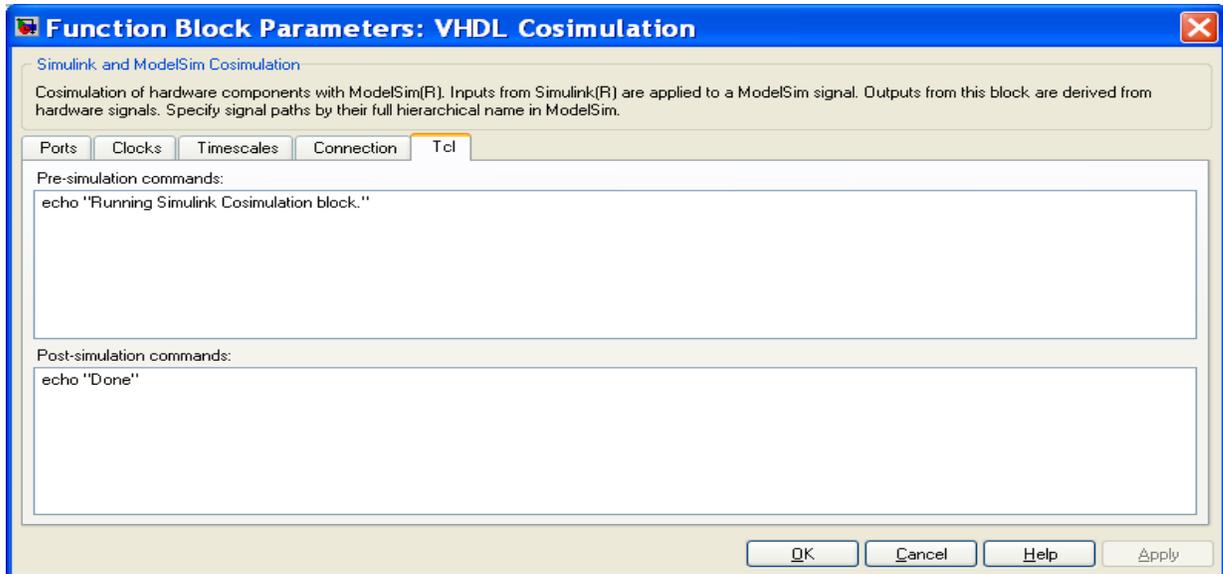


Рис.6. Параметры функционального блока.

7. Инсталлируя ModelSim в Командном окне МатЛАБ для его совместной работы с Simulink выполните следующий ввод.

```
>> setupmodelsim
```

...

Do you want setupmodelsim to locate installed ModelSim executables [y]/n? y

Выберите “y”

Select a ModelSim installation:

[1] C:\Modeltech_6.0c\win32 ModelSim SE 6.0c

[0] None

Выберите “1”

Previous MATLAB startup file found in this installation of ModelSim:

C:\Modeltech_6.0c\win32\..\tcl\ModelSimTclFunctionsForMATLAB.tcl

Do you want to replace this file [y]/n? y

Выберите “y”

8. Запустите ModelSim командой МатЛАБ:

```
>> vsim('tclstart', 'do plot.do', 'socketsimulink', 4449)
```

9. Запустите mdl модель Simulink.

10. Рассмотрите специфику взаимодействия модели Simulink и Verilog модуля.

11. Добавьте канал передачи данных в общую модель

12. Постройте осциллограмму передачи данных

Примечание: После редактирования verilog файла, откомпилируйте его и затем выполните команду: `wrparverilog <имя отредактированного файла>`

Задание 2. Построение Simulink модели управления клапаном с использованием модуля описания аппаратуры

1. Постройте модель управления клапаном в среде Simulink как показано на рисунке **Рис. 7**.

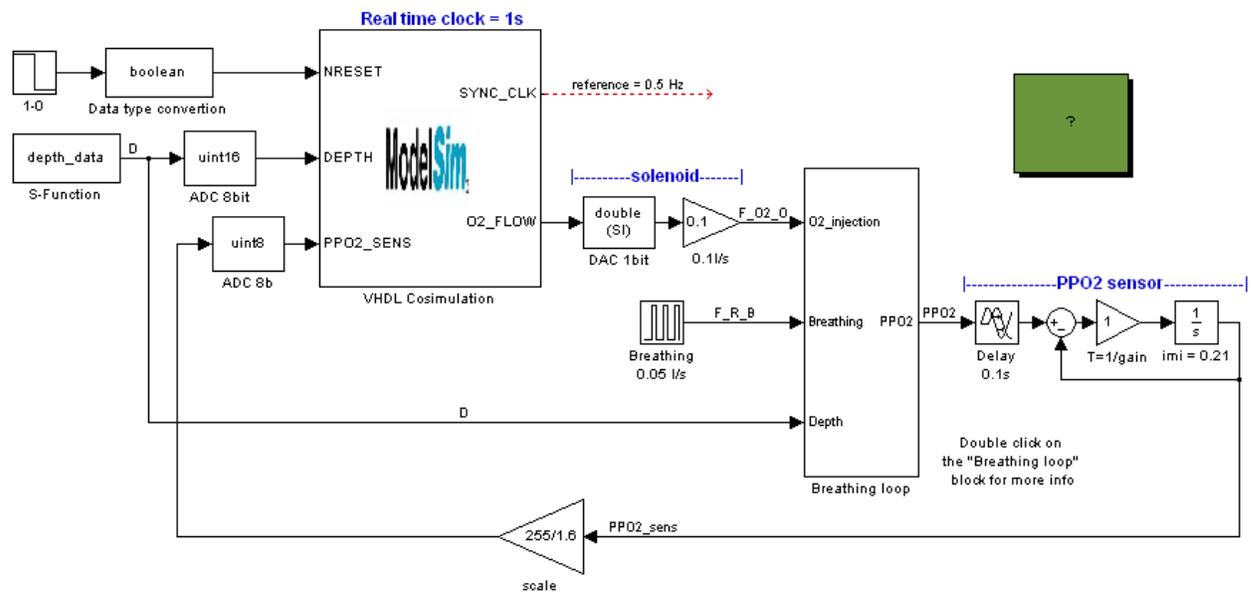


Рис. 7. Пример модели verilog – simulink

2. Откомпилируйте следующий Verilog файл.

Verilog HDL script

```
//-----  
// Design unit : VALVE_CONTROL  
// Filename: VLV_SC.v  
//  
// Description: valve solenoid control  
//  
// Limitations:  
//  
// Author: BD  
//  
// Simulator: ModelSim 6.0c  
// Version 1.1 by 3 October 2005, original release  
//-----
```

```

//timescale 100ms / 10ms

module VLV_SC( SYS_CLK, DEPTH, NRESET, PPO2_SENS, SYNC_CLK, O2_FLOW);

input SYS_CLK, NRESET;
input [15:0] DEPTH;           // 16-bit DAC. Depth resolution = 1m
input [7:0] PPO2_SENS;      // 8-bit ADC of PPO2 sensor

output SYNC_CLK, O2_FLOW;

reg O2_FLOW;                // solenoid valve injector control
reg [7:0] PPO2_GOAL;        // 8-bit PPO2 set point (code 255 means PPO2 is 1.6)
                             // code 112 means PPO2 is 0.7
                             // code 191 means PPO2 is 1.2

assign SYNC_CLK = SYS_CLK; // control of the MatLAB clock

// Real time clock is 1 s
always @SYS_CLK             // Simulink lowest clock period is 2 s
if ( !NRESET )
begin
    if (DEPTH < 10)         // Depth in m
        PPO2_GOAL = 112;   // 0.7 of PPO2
    else
        PPO2_GOAL = 191;   // 1.2 of PPO2

    if (PPO2_SENS < PPO2_GOAL)
        O2_FLOW = 1;
    else
        O2_FLOW = 0;
end
else
    O2_FLOW = 0;           //solenoid is off

endmodule

```

3. Создайте оболочку Verilog файла

VHDL wrap

The Simulink connects the verilog compiled data via the VHDL wrap.

The ModelSim command: wrapverilog **verilog_file_name.v** runs generation of the VHDL file which name is **verilog_file_name_wrap.vhd**

The listing of the used VHDL file is following.

```

-----
-- Module vlv_sc VHDL Wrapper
--
-- Generated by The MathWorks wrapverilog tcl command
--
-- Generated on: 2005-10-03 09:34:16
-----

```

```

LIBRARY IEEE;
USE IEEE.std_logic_1164.all;

ENTITY vlv_sc_wrap IS

    port(
        SYS_CLK      : in  std_logic;
        DEPTH        : in  std_logic_vector(15 downto 0);
        NRESET       : in  std_logic;
        PPO2_SENS    : in  std_logic_vector(7 downto 0);
        SYNC_CLK     : out std_logic;
        O2_FLOW      : out std_logic
    );

END vlv_sc_wrap;

ARCHITECTURE rtl OF vlv_sc_wrap IS

    component VLV_SC
        port(
            SYS_CLK      : in  std_logic;
            DEPTH        : in  std_logic_vector(15 downto 0);
            NRESET       : in  std_logic;
            PPO2_SENS    : in  std_logic_vector(7 downto 0);
            SYNC_CLK     : out std_logic;
            O2_FLOW      : out std_logic
        );
    end component;

    FOR ALL : vlv_sc
        USE ENTITY work.vlv_sc(ignored);

BEGIN
    u_vlv_sc: vlv_sc
        PORT MAP (
            SYS_CLK => SYS_CLK ,
            DEPTH  => DEPTH ,
            NRESET => NRESET ,
            PPO2_SENS => PPO2_SENS ,
            SYNC_CLK => SYNC_CLK ,
            O2_FLOW => O2_FLOW
        );

END rtl;

```

4. Настройте параметры моделирования.

5. Запустите модель.
6. Убедитесь в работоспособности модели.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Для чего предназначен язык описания аппаратных средств Verilog?
2. Для чего предназначен блок “VHDL Cosimulation” mdl модели?
3. Какие задачи можно решать при помощи объединенной модели Verilog – Simulink?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Обзор современных языков описания аппаратуры <http://systemc.dax.ru/book/1.html>
2. Введение в Verilog <http://www.marsohod.org/index.php/ourblog/11/77-veriloglesson2>
3. Краткий курс HDL. Описание языка Verilog. http://www.kit-e.ru/articles/circuit/2008_5_154.php
4. Dr. Bob Davidov. Компьютерные технологии управления в технических системах <http://portalnp.ru/author/bobdavidov>