DS1WM Release Notes

Rev 060507

Introduction:

This release contains Verilog and VHDL source code for the DS1WM RTL, and Verilog source code for the testbench.

The document discusses the following topics:

- DS1WM source code and compile notes
- Testbench architecture and source code
- DS1WM programming model
- DS1WM functional verification testcases

DS1WM source code and compile notes

The DS1WM RTL source code is located under:

```
delivery/design/ds1wm
```

This directory has separate counterpart sub-directories for VHDL and Verilog DS1WM RTL code.

The module name and hierarchy is identical and equivalent between the two languages. A simulation flow is not included, however, the VHDL / Verilog should be
compiled in the following order:

Verilog	**VHDL**
one_wire_io.v	one_wire_io.vhd
clk_prescaler.v	clk_prescaler.vhd
one_wire_interface.v	one_wire_interface.vhd
onewiremaster.v	onewiremaster.vhd
ds1wm.v	ds1wm.vhd

Testbench Architecture and Source Code

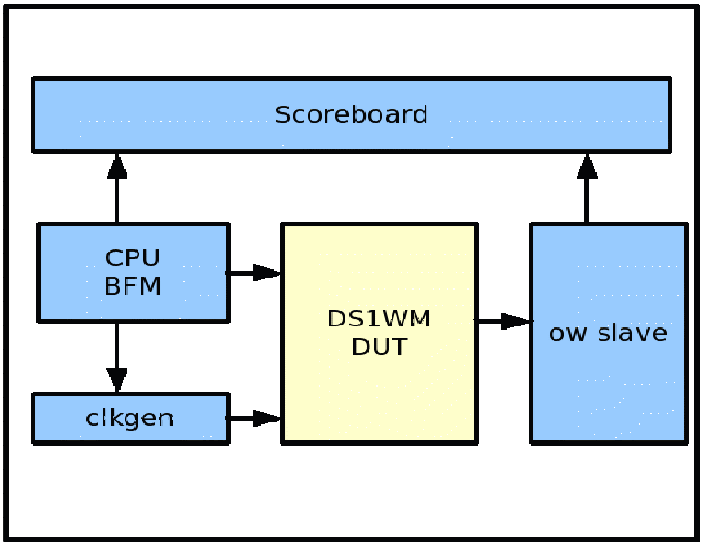

Illustration 1: Testbench Architecture

The testbench presented as Illustration 1 is comprised of the following source code file:

```
delivery/verification/testbench/tb_ds1wm

        tb_ds1wm.v
        tc_ds1wm.v


delivery/verification/testbench/cpu_bfm

        cpu_bfm.v

delivery/verification/testbench/ow_slave

        ow_slave.v
        cmd_ctrl.v
        iox.v

delivery/verification/testbench/clkgen

        clkgen.v

delivery/verification/testbench/scoreboard

        scoreboard.v
```

The **stimulus.inc** file contains the test stimulus. It is included by the tc_ds1wm block, and should not be explicitly compiled.

DS1WM Functional Verification

Each DS1WM transaction consists of the following instruction sequence:

1. 1-Wire Reset
2. A device addressing instruction
3. A data access instruction

A device addressing instruction may one of the following:

- READ_ROM
- SKIP_ROM

A data access instruction may be one of the following:

- WRITE_SP
- READ_SP
- COPY_SP
- READ_MEM

This version of the testbench verifies that the slave correctly recognizes 1-wire commands issued by the testbench. Scratch Pad data integrity is not checked.

There are 3 fundamental 1-Wire transactions:

1. 1-wire reset
2. 1-wire write cycle
3. 1-wire read cyle

The DS1WM sets specific status bits in it's Interrupt Register to indicate that a 1-Wire transaction has completed. Interrupts are raised if the corresponding Interrupt Enable bit has be set.

The testbench uses these interrupts to control instruction sequencing.

Test Cases

Testcases are located under delivery/verification/tests.

cmd_recognition -

Performs 1-wire command recognition tests for all valid Clock Divisor Register settings. It includes scenarios for both active high resets and active low resets.

This comes with a run script for simulation with ncsim.

For usage, please see the README file in the test directory.