Pattern Conversion

上海瀚驰信息科技有限公司

------- Cavin Rui

2014-04-16

EDA Pattern

1. **EVCD**
   Extended Verilog value Change Dump
2. **VCD**
   Verilog value Change Dump
3. **WGL**
   Waveform Generate Language
4. **STIL**
   Standard Test Interface Language

V C D

- $date
- Mar 18, 2011 22:07:16
- $end
- $version
- TOOL: ncsim 05.40-p004
- $end
- $timescale
- 1 ps
- $end
- $scope module Testbenchx2 $end
- $scope module u_chip_topx2 $end
- $var wire 1 ! ROUT1 $end
- $scope module u_d_top $end
- $scope module u_PAD $end
- $var wire 1 . DIN9_oen $end
- $var wire 1 / DIN8_oen $end
- $upscope $end
- $upscope $end
- $upscope $end
- $upscope $end
- $enddefinitions $end
- $dumpvars
- x!
- x.
- 1/
- #10000
- 0!
- 1.
- 0/
- #10000020000

VCD

- File Head

  date, version

- Value

  timescale, signal definition，time edge, data

- time scale

  10fs, 1ps, 100ps, 1ns

- signal definition

  control pins, functional pins

VCD

- Conversion requirements
  - -period
  - -pins mapping
  - -pins type
  - -data function

VCD

- Conversion requirements (option)
  - -prefix
  - -suffix
  - -compare mode
  - -jet process
  - -compress to repeat

V C D

- Cyclizing
- Timing Mask
- Data Capture

W G L

- Key words

  waveform, signal, scanCell, scanChain, scanState, timeplate, pattern, end

- Second Key words

  output, input, bidir, vector, loop, repeat, scan

W G L

❖SIgnal

```
signal
     SDO : output;
     CSB : input;
     resetb : input;
     SCK    : input;
     INT1   : bidir;
     "PS"   : input;
     SDI    : bidir;
     DUMMY_PORB : input;
     DUMMY_CLOCK_DIG : input;
end
```

Timeplate

```
timeplate _default_WFT_ period 500nS
    SDO := output[0pS:X, 245nS:Q, 255nS:X];
    CSB := input[0pS:S];
    resetb := input[0pS:U, 225nS:S, 275nS:U];
    SCK    := input[0pS:S];
    INT1   := input[0pS:D, 255nS:S, 495nS:D];
    INT1   := output[0pS:X, 245nS:Q'edge, 250nS:X];
    "PS"   := input[0pS:S];
    SDI    := input[0pS:P, 55nS:S];
    SDI    := output[0pS:X, 245nS:Q'edge, 250nS:X];
    DUMMY_PORB := input[0pS:S];
    DUMMY_CLOCK_DIG := input[0pS:S];
end
```

W G L

❖Pattern

```
pattern group_ALL (SDO,CSB,resetb,SCK,INT1:I,INT1:O,"PS",SDI:I,SDI:O,DUMMY_PORB,DUMMY_CLOCK_DIG)
    vector(0, 0pS, _default_WFT_) := [ X 1 0 1 - X 0 0 - 0 0 ];
     repeat 50 vector(1, 500nS, _default_WFT_) := [ X 1 0 1 - X 0 0 - 0 0 ];
    vector(2, 1uS, _default_WFT_)   := [X 1 0 1 - X 0 0 - 0 1 ];
    vector(3, 1.5uS, _default_WFT_) := [X 1 0 1 - X 0 0 - 0 1 ];
    vector(4, 2uS, _default_WFT_)   := [X 1 0 1 - X 0 0 - 0 0 ];
    vector(5, 2.5uS, _default_WFT_) := [X 1 0 1 - X 0 0 - 0 0 ];
    vector(6, 3uS, _default_WFT_)   := [X 1 0 1 - X 0 0 - 0 1 ];
    vector(7, 3.5uS, _default_WFT_) := [X 1 0 1 - X 0 0 - 0 1 ];
    vector(8, 4uS, _default_WFT_)   := [X 1 0 1 - X 0 0 - 0 0 ];
```

WGL

Scan

scancell, scanchain, scanstate

```
scanCell
    chip_core_i_dig_i_adcd_top_i_adcd_i_adcd_dp_i1_tstrobe_d_reg ;
    chip_core_i_dig_i_interface_i_isi_i_isi_core_i_addr_valid_reg ;
    chip_core_i_dig_i_interface_i_isi_i_isi_core_i_byte_ack_reg ;
    chip_core_i_dig_i_interface_i_isi_i_isi_core_i_data_o_reg_reg_0_ ;
    chip_core_i_dig_i_interface_i_isi_i_isi_core_i_data_o_reg_reg_1_ ;

scanChain
    c0 [SDI, chip_core_i_dig_i_adcd_top_i_adcd_i_adcd_dp_i1_tstrobe_d_reg,
    chip_core_i_dig_i_interface_i_isi_i_isi_core_i_addr_valid_reg, !,
    chip_core_i_dig_i_interface_i_isi_i_isi_core_i_byte_ack_reg, !,
     chip_core_i_dig_i_interface_i_isi_i_isi_core_i_data_o_reg_reg_0_,
     chip_core_i_dig_i_interface_i_isi_i_isi_core_i_data_o_reg_reg_1_,

scanState

    c0L0 := ALLSCAN(10111001100111110010110010001101001100110000011110110110
    01110100111100100011001100110011001001110011001100110011001100110011001
    c0U1 := ALLSCAN(10111001100111110010110010001101001100110000011110110110
```

WGL

Scan

```
vector(1371, 685.5uS, _default_WFT_) := [X 1 1 1 0 - 0 - X 1 0 ];
scan(1372, 686uS, _default_WFT_)     := [- 1 1 1 1 - 0 - - 1 0 ],
output[c0:c0U0], input[c0:c0L0];    { capture }
vector(2503, 1.2515mS, _default_WFT_) := [1 1 1 0 0 - 1 - X 1 0 ];
{ load_unload }
vector(2504, 1.252mS, _default_WFT_)  := [X 1 1 0 0 - 1 - X 1 0 ];
scan(2505, 1.2525mS, _default_WFT_)   := [- 1 1 0 1 - 1 - - 1 0 ],
output[c0:c0U1], input[c0:c0L1];    { capture_int_dig }
vector(3636, 1.818mS, _default_WFT_)  := [X 0 1 1 0 - 1 - X 1 0 ];
vector(3637, 1.8185mS, _default_WFT_) := [0 0 1 1 0 - 1 - X 1 0 ];
vector(3638, 1.819mS, _default_WFT_)  := [X 0 1 1 1 - 1 - X 1 0 ];
```

W G L

- Conversion options
  - -Flatten scan chain
  - -loop expanding
  - -save comment
  - -reserve scanstate
  - -output data “Z”

Conversion Option

- 自己写script，要求pattern format简单
- 找vendor
- 买转换工具

MegaVector & MegaWaver

- 将VCD/EVCD的文件转换成 WGL 的文件，再将WGL 的文件转换成ATE 向量。并且可以支持显示转换后的向量波形(支持VCD/EVCD/WGL)。
- 目前支持的ATE平台有
  - J750/UltraFlex/Flex
  - 83K/93K/V50
  - Duo/Quartet/D10
  - T2000/LTX VX4/Eagle ETS
  - VTT/3360

提问讨论

Thank You !

Q & A