AGNISYS
SYSTEM DEVELOPMENT WITH CERTAINTY

Correct by construction SV UVM code with DVinsight - a smart editor

Devender Khari
(Presenter)

Neena Chandawale
(Host)

Agenda

- Challenges in SV/UVM Development
- Solution along with Demo
- Editing Features
  - Edit inline in the same context (Quick Edit)
  - Multiple edit allowing multiple lines edited together
  - Hinting & suggestions
  - Auto code completion
  - Easy code move
  - Easy commenting
  - Quick search
- Pre-Compiling Features
  - Syntax checking while writing code
  - Built-in UVM linter
  - Automatically checks for past practices
- GUI features & Command interface
  - DVi Console window
  - Links with all popular simulators
  - Multiple split windows to work on multiple files together
  - Vi & Emacs mode
- Benefits

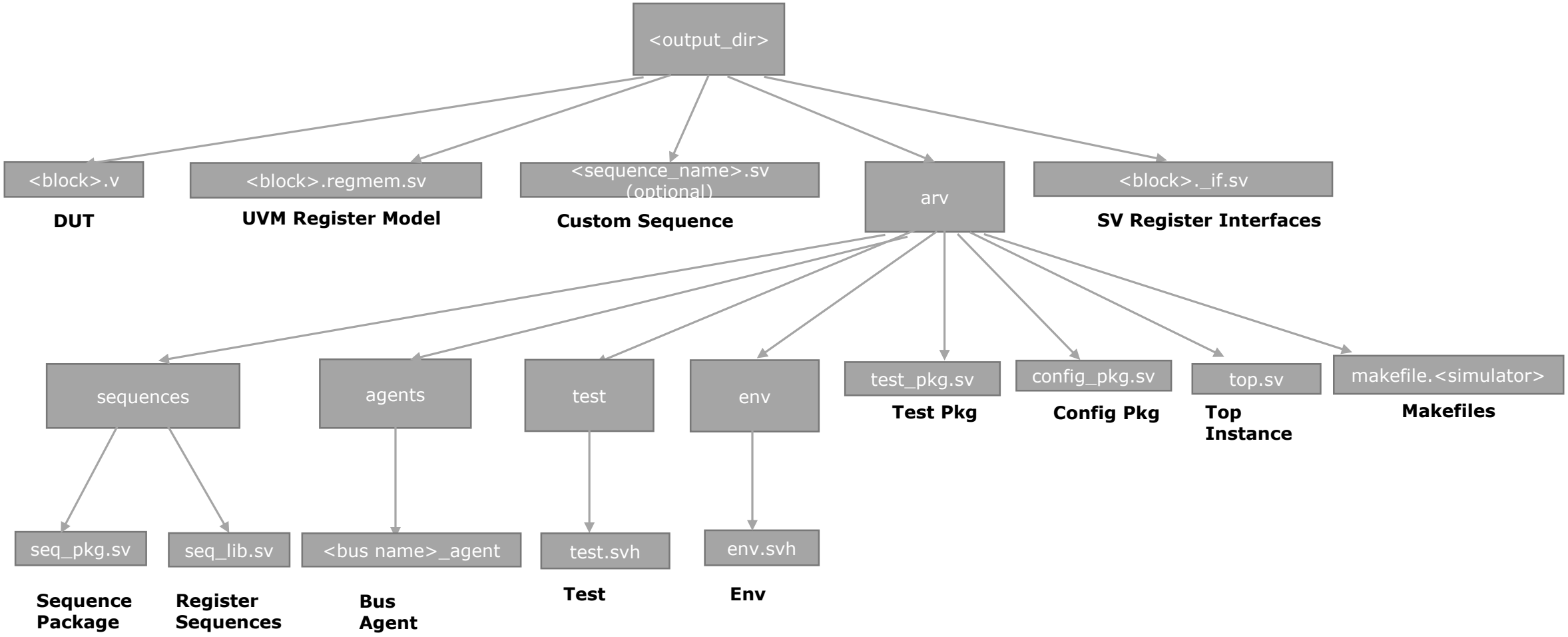
AGNISYS
SYSTEM DEVELOPMENT WITH CERTAINTY

Challenges faced in SV/UVM Development

- Hierarchical directory structure of Verification environment
- Large number of files scattered all over the hierarchy of directories
- Conventional editing is Time Consuming and Confusing
- Limited number of simulator licenses
- Longer learning curve of new DV engineers
- Error free code development besides keeping the development cycle short

AGNISYS
SYSTEM DEVELOPMENT WITH CERTAINTY

Directory hierarchy of a typical verification environment



AGNISYS
SYSTEM DEVELOPMENT WITH CERTAINTY

Editing Features

Edit inline in the same context (Quick Edit)

- Edit different files quickly by opening an insight of files within the same editor window.
- These insights are editable so no chance of getting lost in files.
- Use key shortcut (Ctrl-E) or right click on the scope whose insight is needed and select "Quick Edit" from pop-up menu to see the editable insight.

```
1  class pic_ctrl_test extends uvm_test;
2      `uvm_component_utils(pic_ctrl_test)
3
4      pic_ctrl_env env;
5      pic_ctrl_block modelinst;
6      config_object cfg;
7      uvm_reg_sequence seq;
8      uvm_arv_reset_seq resetseq;
9      my_catcher catch = new("catch");
10
11     function new(string name,uvm_component parent);
12         super.new(name,parent);
13     endfunction
14
15     function void build_phase(uvm_phase phase);
16
17         super.build_phase(phase);
18
19         cfg = config_object::type_id::create("cfg");
20         cfg.check(strName);
21         if(cfg.model)
22             begin
23             end
24         else
```

```
3
4       pic_ctrl_env env;
5       pic_ctrl_block modelinst;
6       config_object cfg;
```

config_pkg.sv:8

```
8       class config_object extends uvm_object;
9           `uvm_object_utils(config_object)
10          virtual ambaapb_if ambaapbif;
11          pic_ctrl_block model1;
12          virtual pic_ctrl_hw_if pic_ctrl_hif;
13          pic_ctrl_block model;
14
15          function new(string name="");
16              super.new(name);
17          endfunction
18          function check(string strName);
19          endfunction
20      endclass
21      string replacements[string] = '{ "<" : "&lt;",
```

```
7       uvm_reg_sequence seq;
8       uvm_arv_reset_seq resetseq;
```

Hinting & Auto completion

- DVi-Hint
  - Provide context sensitive hints and guidance to the user
  - No need to open different files to see name of any member or method.
- Auto code completion
  - Apply code templates when selected from the hint menu.
  - Using code hints like "if else" instead of writing "if" construct, it will do auto completion of the template

```
 3
 4       pic_ctrl_env env;
 5       pic_ctrl_block modelinst;
 6       config_object cfg;
 7       uvm_reg_sequence seq;
 8       uvm_arv_reset_seq resetseq;
 9       my_catcher catch = new("catch");
10
11       function new(string name,uvm_component parent);
12           super.new(name,parent);
13       endfunction
14
15       function void build_phase(uvm_phase phase);
16
17           super.build_phase(phase);
18
19           cfg = config_object::type_id::create("cfg");
20           cf
21                 cfg
22           if (!uvm_config_db #(virtual ambaapb_if)::get(this,"","AMBAAPB_IF",cfg.ambaapbif
23
24               `uvm_fatal("BUILD_PHASE", "cannot get ambaapb_if from config_db")
25           end
26           if(!uvm_config_db #(virtual pic_ctrl_hw_if)::get(this,"","pic_ctrl_hif",cfg.pic_
27               `uvm_fatal("BUILD_PHASE", "cannot get pic_ctrl_hif from config_db")
28           end
```

```
12          super.new(name,parent);
13      endfunction
14
15      function void build_phase(uvm_phase phase);
16
17          super.build_phase(phase);
18
19          cfg = config_object::type_id::create("cfg");
20          cfg.
21
22          if                                  al ambaapb_if)::get(this,"","AMBAAPB_IF",cfg.ambaapbif)) begin
23
24                                              E", "cannot get ambaapb_if from config_db")
25          end
26          if(                                 l pic_ctrl_hw_if)::get(this,"","pic_ctrl_hif",cfg.pic_ctrl_hif)
27                                              E", "cannot get pic_ctrl_hif from config_db")
28          end
29
30          uvm_reg::include_coverage("*",UVM_CVR_ALL);
31          modelinst = pic_ctrl_block::type_id::create("pic_ctrl");
32          modelinst.build();
33
34          cfg.model = modelinst;
35          uvm_config_db #(config_object)::set(null,"uvm_test_top*","cfg",cfg);
36
37          env = pic_ctrl_env::type_id::create("env",this);
```

ambaapbif
check(strName)
model
model1
new(name)
pic_ctrl_hif

```
12          super.new(name,parent);
13      endfunction
14
15      function void build_phase(uvm_phase phase);
16
17          super.build_phase(phase);
18
19          cfg = config_object::type_id::create("cfg");
20          if
21
22          i                                  tual ambaapb_if)::get(this,"","AMBAAPB_IF",cfg.ambaapbif)) be
23
24                                             ASE", "cannot get ambaapb_if from config_db")
25          e
26          i                                  ual pic_ctrl_hw_if)::get(this,"","pic_ctrl_hif",cfg.pic_ctrl_
27                                             ASE", "cannot get pic_ctrl_hif from config_db")
28          end
29
30          uvm_reg::include_coverage("*",UVM_CVR_ALL);
31          modelinst = pic_ctrl_block::type_id::create("pic_ctrl");
32          modelinst.build();
33
34          cfg.model = modelinst;
35          uvm_config_db #(config_object)::set(null,"uvm_test_top*","cfg",cfg);
```

```
if
if else
if else if
iff
ifnone
```

```
12          super.new(name,parent);
13      endfunction
14
15      function void build_phase(uvm_phase phase);
16
17          super.build_phase(phase);
18
19          cfg = config_object::type_id::create("cfg");
20          if()
21            begin
22            end
23          else
24            begin
25            end
26
27          if (!uvm_config_db #(virtual ambaapb_if)::get(this,"","AMBAAPB_IF",cfg.ambaapbif)) begin
28
29              `uvm_fatal("BUILD_PHASE", "cannot get ambaapb_if from config_db")
30          end
31          if(!uvm_config_db #(virtual pic_ctrl_hw_if)::get(this,"","pic_ctrl_hif",cfg.pic_ctrl_hif))
32              `uvm_fatal("BUILD_PHASE", "cannot get pic_ctrl_hif from config_db")
33          end
34
35          uvm_reg::include_coverage("*",UVM_CVR_ALL);
36          modelinst = pic_ctrl_block::type_id::create("pic_ctrl");
37          modelinst.build();
```

Multiple edit & Easy code move

- Multiple edit
  - Helps writing same text at multiple lines at the same time
  - Hold "Ctrl" and place cursors on all the locations where editing is needed
  - Release "Ctrl" and start writing

- Easy code move
  - Helps in moving code within the same file without cut+paste
  - Select line to be moved and use (Ctrl + Shift + up/down arrow).

- Indentation
  - Use (Ctrl + ‘[‘) to left indent code
  - Use (Ctrl + ‘]’) to right indent code

AGNISYS
SYSTEM DEVELOPMENT WITH CERTAINTY

```
1  class pic_ctrl_test extends uvm_test;
2      `uvm_component_utils(pic_ctrl_test)
3
4      pic_ctrl_env env;
5      pic_ctrl_block modelinst;
6      config_object cfg;
7      uvm_reg_sequence seq;
8      uvm_arv_reset_seq resetseq;
9      my_catcher catch = new("catch");
10
11     function new(string name,uvm_component parent);
12         super.new(name,parent);
13     endfunction
14
15     function void build_phase(uvm_phase phase);
16
17         super.build_phase(phase);
18
19         cfg = config_object::type_id::create("cfg");
20         if()
21           begin
22           end
23         else
24           begin
```

```
1  class pic_ctrl_test extends uvm_test;
2      `uvm_component_utils(pic_ctrl_test)
3
4      local pic_ctrl_env env;
5      pic_ctrl_block modelinst;
6      local config_object cfg;
7      uvm_reg_sequence seq;
8      local uvm_arv_reset_seq resetseq;
9      local my_catcher catch = new("catch");
10
11     function new(string name,uvm_component parent);
12         super.new(name,parent);
13     endfunction
14
15     function void build_phase(uvm_phase phase);
16
17         super.build_phase(phase);
18
19         cfg = config_object::type_id::create("cfg");
20         if()
21           begin
22           end
23         else
24           begin
```

```
 6      local config_object cfg;
 7      uvm_reg_sequence seq;
 8      local uvm_arv_reset_seq resetseq;
 9      local my_catcher catch = new("catch");
10
11      function new(string name,uvm_component parent);
12          super.new(name,parent);
13      endfunction
14
15      function void build_phase(uvm_phase phase);
16
17          super.build_phase(phase);
18
19          cfg = config_object::type_id::create("cfg");
20          if()
21            begin
22            end
23          else
24            begin
25            end
26
27          if (!uvm_config_db #(virtual ambaapb_if)::get(this,"","AMBAAPB_IF",cfg.ambaapbif)) begin
28
29              `uvm_fatal("BUILD_PHASE", "cannot get ambaapb_if from config_db")
```

```
 6      local config_object cfg;
 7      uvm_reg_sequence seq;
 8      local uvm_arv_reset_seq resetseq;
 9      local my_catcher catch = new("catch");
10
11      function new(string name,uvm_component parent);
12          super.new(name,parent);
13      endfunction
14
15      function void build_phase(uvm_phase phase);
16
17          super.build_phase(phase);
18
19          cfg = config_object::type_id::create("cfg");
20
21          if (!uvm_config_db #(virtual ambaapb_if)::get(this,"","AMBAAPB_IF",cfg.ambaapbif)) begin
22
23              `uvm_fatal("BUILD_PHASE", "cannot get ambaapb_if from config_db")
24          end
25          if()
26            begin
27            end
28          else
29            begin
30            end
31          if(!uvm_config_db #(virtual pic_ctrl_hw_if)::get(this,"","pic_ctrl_hif",cfg.pic_ctrl_hif)) begin
32              `uvm_fatal("BUILD_PHASE", "cannot get pic_ctrl_hif from config_db")
33          end
```

Comments & Quick search

- Easy commenting
  - Helps in commenting code with simple key combination
  - Use (Ctrl + "/") for single line comment on current/selected line
  - Use (Ctrl + Shift + "/") for block line comment on current/selected line

- Quick search
  - Helps in finding words within the file.
  - Just select the word to be searched and it will highlight all the matching words in file
  - Also it will set markers on the scroll bar for positions of search results

```
1  class pic_ctrl_test extends uvm_test;
2      `uvm_component_utils(pic_ctrl_test)
3
4      pic_ctrl_env env;
5      pic_ctrl_block modelinst;
6      config_object cfg;
7      uvm_reg_sequence seq;
8      uvm_arv_reset_seq resetseq;
9      my_catcher catch = new("catch");
10
11     function new(string name,uvm_component parent);
12         super.new(name,parent);
13     endfunction
14
15     function void build_phase(uvm_phase phase);
16
17         super.build_phase(phase);
18
19         cfg = config_object::type_id::create("cfg");
20
21         if (!uvm_config_db #(virtual ambaapb_if)::get(this,"","AMBAAPB_IF",cfg.ambaapbif)) begin
22
23             `uvm_fatal("BUILD_PHASE", "cannot get ambaapb_if from config_db")
24         end
```

Pre-Compiling Features

Pre-Compiling Features

- Built-in UVM Linter
  - Helps in writing DV code correct-by-construction
  - Checks UVM rules-based errors early in the development cycle
  - Helps avoiding potential debug challenges saving huge costs
  - Analyze code on save in the context of the environment and report mistakes
  - Automatically checks for compliance with UVM best practices

- Syntax checking on the fly
  - Syntax errors are reported while writing code itself

```
1  class pic_ctrl_test extends uvm_test;
   DVi-WARN-04 :: All classes in UVM tree should be registered with UVM Factory.
2      //`uvm_component_utils(pic_ctrl_test)
3
4      pic_ctrl_env env;
5      pic_ctrl_block modelinst;
6      config_object cfg;
7      uvm_reg_sequence seq;
8      uvm_arv_reset_seq resetseq;
9      my_catcher catch = new("catch");
10
11     function new(string name,uvm_component parent);
12         super.new(name,parent);
13     endfunction
14
15     function void build_phase(uvm_phase phase);
16
17         super.build_phase(phase);
18
19         cfg = config_object::type_id::create("cfg");
20
21         if (!uvm_config_db #(virtual ambaapb_if)::get(this,"","AMBAAPB_IF",cfg.ambaapbif)) begin
22
23             `uvm_fatal("BUILD_PHASE", "cannot get ambaapb_if from config_db")
24         end
```

```
38          resetseq = uvm_arv_reset_seq::type_id::create("resetseq",this);
39          resetseq.model = modelinst;
40          begin
41              string seq_name;
42              if ($value$plusargs("UVM_SEQUENCE=%s",seq_name)) begin
43                  seq = uvm_utils #(uvm_reg_sequence)::create_type_by_name(seq_name,"");
44                  if (seq == null) begin
45                      `uvm_fatal("NO_SEQUENCE","This env requires you to specify the sequence t
46                  end
47              end
48          end
49          seq.model = modelinst;
50      endfunction
51      function void end_of_elaboration_phase(uvm_phase phase);
  DVi-WARN-08 :: It is suggested to call super.end_of_elaboration_phase() on the first line of en
52          uvm_report_cb::add(null,catch);
53          uvm_top.print_topology();
54      endfunction
55
56      task main_phase(uvm_phase phase);
  DVi-WARN-08 :: It is suggested to call super.main_phase() on the first line of main_phase
57          phase.raise_objection(this);
58          resetseq.start(env.v_seqr);
59          seq.start(env.v_seqr);
60          phase.drop_objection(this);
61      endtask
```

GUI features & Command interface

GUI features & Command interface

- DVi console window
  - Click on the Console button on right to open console window
  - Use console commands for simulators

- Simulator Link
  - Links with popular simulators, cross reference compile errors in DVinsight
  - Then click on the file names with errors
  - Avoid back and forth between editor & simulator

- Multiple split windows to work on multiple files together

- Vim/Emacs Mode
  - Enables to use vim/emacs key bindings

```
 "cannot get ambaapb_if from config_db")

ic_ctrl_hw_if)::get(this,"","pic_ctrl_hif",cfg.pic_ctrl_hif)) begin
 "cannot get pic_ctrl_hif from config_db")


,UVM_CVR_ALL);
ype_id::create("pic_ctrl");



)::set(null,"uvm_test_top*","cfg",cfg);

create("env",this);

:type_id::create("resetseq",this);



SEQUENCE=%s",seq_name)) begin
_reg_sequence)::create_type_by_name(seq_name,"");
n
QUENCE","This env requires you to specify the sequence to run using UVM_SEQUENCE=seq_name");
```

```
36 ▼      function void build_phase(uvm_phase phase);
37            super.build_phase(phase);
38
39            if(!uvm_config_db #(config_object)::get(this,"","cfg",cfg)) begin
40                `uvm_fatal("CONFIG", "config object not found");
```

DVi Console

```
/home/sumeet/DVI/test cases/dvcon_example/ids/arv/$ make -f Makefile.questa run
```

```
12      function new(string name,uvm_component parent);
13          super.new(parent, name);
14          $display("constructor call here");//dvi ignore
15      endfunction
16
17      function void build_phase(uvm_phase phase);
18          super.build_phase(phase);
19          con_obj.length=10;
20          con_obj.print(str);
21          cfg = config_object::type_id::create("cfg");
22          if (!uvm_config_db #(virtual prop_if)::get(this,"","PROP_IF",cfg.propif)) b
23
24              `uvm_fatal("BUILD_PHASE", "cannot get prop_if from config_db")
25          end
26          if(!uvm_config_db #(virtual Eth_chip_hw_if)::get(this,"","Eth_chip_hif",cfg
27              `uvm_fatal("BUILD_PHASE", "cannot get Eth_chip_hif from config_db")
28          end
```

DVi Console

```
  -- Compiling package test_pkg
** Error: test/test.svh(13): Arg. 'name' of 'new': Illegal assignment to type 'string' from type 'class
Types are not assignment compatible.
** Error: test/test.svh(13): Arg. 'parent' of 'new': Illegal assignment to type 'class uvm_pkg.uvm_compo
Types are not assignment compatible.
** Error: test/test.svh(20): (vlog-2730) Undefined variable: 'str'.
```

top.sv (arv) - DVinsight

top.sv

```
1  //Agnisys, Inc. ***** Copyright 2019 All Rights Reserved.
   *****
2  //
3  //*** This file is auto generated by IDesignSpec
   (http://www.agnisys.com) . Please do not edit this file. ***
4  // created on        : 2020-04-06T21:25:00.199+05:30
5  // created by        :
6  // generated by      : agnis
7  // generated from    : C:\Users\agnis\Desktop\DV-
   insight\interrupt.docx
8  // IDesignSpec rev   : idsbatch v6.34.0.0
9
10 //*** This code is generated with following settings ***
11 // Reg Width                  : 32
12 // Address Unit               : 8
13 // C++ Types int              : hwint
14 // Bus Type                   : APB
15 // BigEndian                  : true
16 // LittleEndian               : true
17 // Dist. Decode and Readback  : false
18 //-------------------------------------------------------
   -----------------------------------------------
19 module top;
20     import uvm_pkg::*;
21     import test_pkg::*;
22
23     parameter bus_width    = 32;
24     parameter addr_width   = 4;
```

top_1.sv

```
1  //Agnisys, Inc. ***** Copyright 2019 All Rights Reserved.
   *****
2  //
3  //*** This file is auto generated by IDesignSpec
   (http://www.agnisys.com) . Please do not edit this file. ***
4  // created on        : 2020-04-06T21:25:00.199+05:30
5  // created by        :
6  // generated by      : agnis
7  // generated from    : C:\Users\agnis\Desktop\DV-
   insight\interrupt.docx
8  // IDesignSpec rev   : idsbatch v6.34.0.0
9
10 //*** This code is generated with following settings ***
11 // Reg Width                  : 32
12 // Address Unit               : 8
13 // C++ Types int              : hwint
14 // Bus Type                   : APB
15 // BigEndian                  : true
16 // LittleEndian               : true
17 // Dist. Decode and Readback  : false
18 //-------------------------------------------------------
   -----------------------------------------------
19 module top;
20     import uvm_pkg::*;
21     import test_pkg::*;
22
23     parameter bus_width    = 32;
24     parameter addr_width   = 4;
```

```
60      // User can add the external logic inside the external ids
61
62
63
64
65      bit clk;
66      assign pic_ctrl_ambaapb.pclk = clk;
67      assign pic_ctrl_hw.clk=clk;
68      always
69          #5 clk = ~clk;
70
71      initial
72          begin
73              clk = 1'b0;
74              uvm_config_db #(virtual ambaapb_if)::set(null,"uvr
75              uvm_config_db #(virtual pic_ctrl_hw_if)::set(null,
76              run_test("pic_ctrl_test");
77          end
78
79      //-------Wave Dump----------//
80
81      `ifdef INCA_WAVE_ON
82      initial
83          $recordvars();
84      `endif
85
```

```
62
63
64
65      bit clk;
66      assign pic_ctrl_ambaapb.pclk = clk;
67      assign pic_ctrl_hw.clk=clk;
68      always
69          #5 clk = ~clk;
70
71      initial
72          begin
73              clk = 1'b0;
74              uvm_config_db #(virtual ambaapb_if)::set(null,'
75              uvm_config_db #(virtual pic_ctrl_hw_if)::set(nu
76              run_test("pic_ctrl_test");
77          end
78
79      if()
80          begin
81          end
82      //-------Wave Dump----------//
83
84      `ifdef INCA_WAVE_ON
85      initial
86          $recordvars();
87      `endif
```

Benefits of DVinsight

- Accelerating error-free code development
- Helps shorten the learning curve of new DV engineers
- Ensures compliance with UVM best practices and established standards
- Saves huge cost by early identification of potential debug challenges
- Optimize simulator license usage
- Avoid back and forth between editor & simulator
- **And, its FREE! Download today!**
- If you need support, you can pay for it (Red Hat Model)

AGNISYS
SYSTEM DEVELOPMENT WITH CERTAINTY

About Agnisys

- The EDA leader in solving complex design and verification problems associated with HW/SW interface
- Formed in 2007
- Privately held and Profitable
- Headquarters in Boston, MA
  - ~1000 users worldwide
  - ~50 customer companies
- Customer retention rate ~90%
- R&D centers (US and India)
- Support centers - Committed to ensure comprehensive support
  - Email : support@agnisys.com
  - Phone : 1-855-VERIFYY
  - Response time within one day; within hours in many cases
  - Time Zones (Boston MA, San Jose CA and Noida India)

IDESIGNSPEC™ (IDS) EXECUTABLE REGISTER SPECIFICATION

Automatically generate UVM models, Verilog/VHDL models, Coverage Model, Software model etc.

AUTOMATIC REGISTER VERIFICATION (ARV)

ARV-Sim™ : Create UVM Test Environment, Sequences, Verification Plans and instantly know the status of the verification project.

ARV-Formal™ : Create Formal Properties and Assertions, and Coverage Model from the specification.

ISEQUENCESPEC™ (ISS)

Create UVM sequences and Firmware routines from the specification.

DVinsight™ (DVi)

Smart Editor for SystemVerilog and UVM projects.

IDS – Next Generation™ (IDS-NG)

Comprehensive SoC/IP Spec Creation and Code Generation Tool

AGNISYS
SYSTEM DEVELOPMENT WITH CERTAINTY

AGNISYS
SYSTEM DEVELOPMENT WITH CERTAINTY

THANK YOU