JasperGold Tutorial

Instructor: Lih-Yih Chiou

Speaker: Chun-Wen Yang

Date: 2022/10/05

Outline

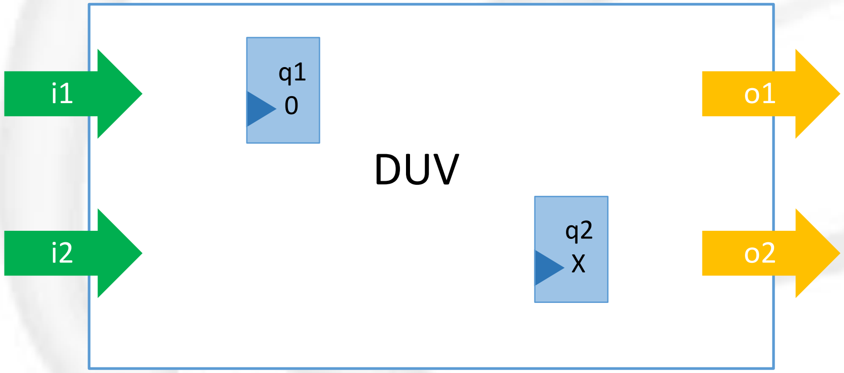
- □ Concept of Formal Verification
- □ JasperGold Tool Guide
- □ Exercise

Outline

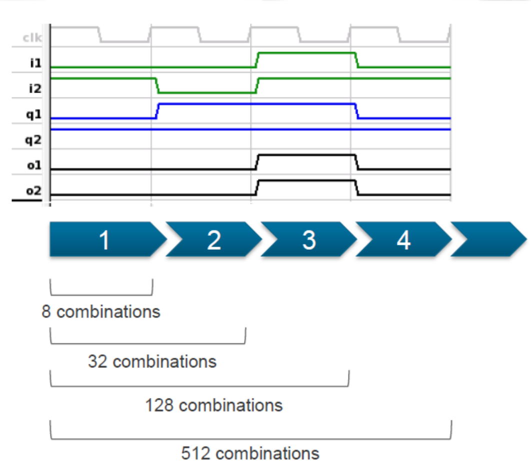
- □ Concept of Formal Verification
- □ JasperGold Tool Guide
- □ Exercise

Formal analysis

- The analysis requires to generate all possible stimulus every cycle
  - All value combinations on **inputs** and **undriven** wires at every cycle. For example, i1 and i2 of the DUV below.
  - All value combinations on **uninitialized** registers(ex:q2) at first cycle. For example, q2 of the DUV below.
  - Similar to simulating $random at every input/register for all possible random values



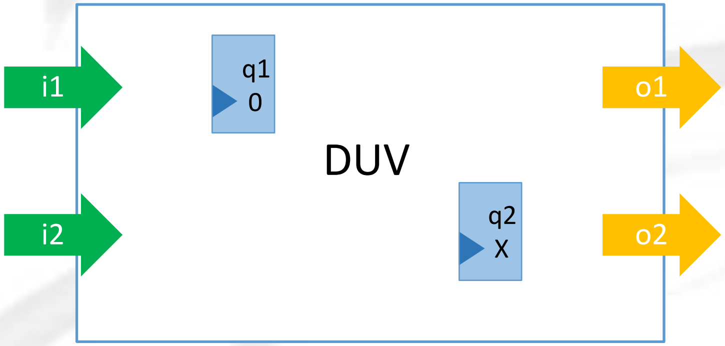
DUV = design under verification

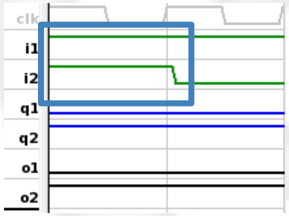


Constraints

- Not all value combinations are legal, so use ***assume*** to tell formal what's legal
  - Only scenarios where assumptions are true will be considered
  - Formal throws away any stimulus that violates assumptions

assume (i1 |-> !i2)





Found scenario
that violates assumptions !

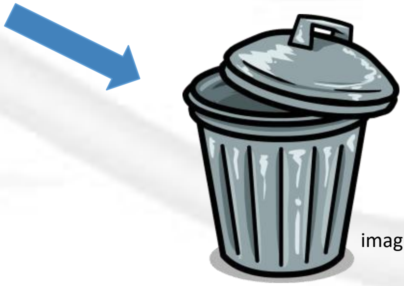
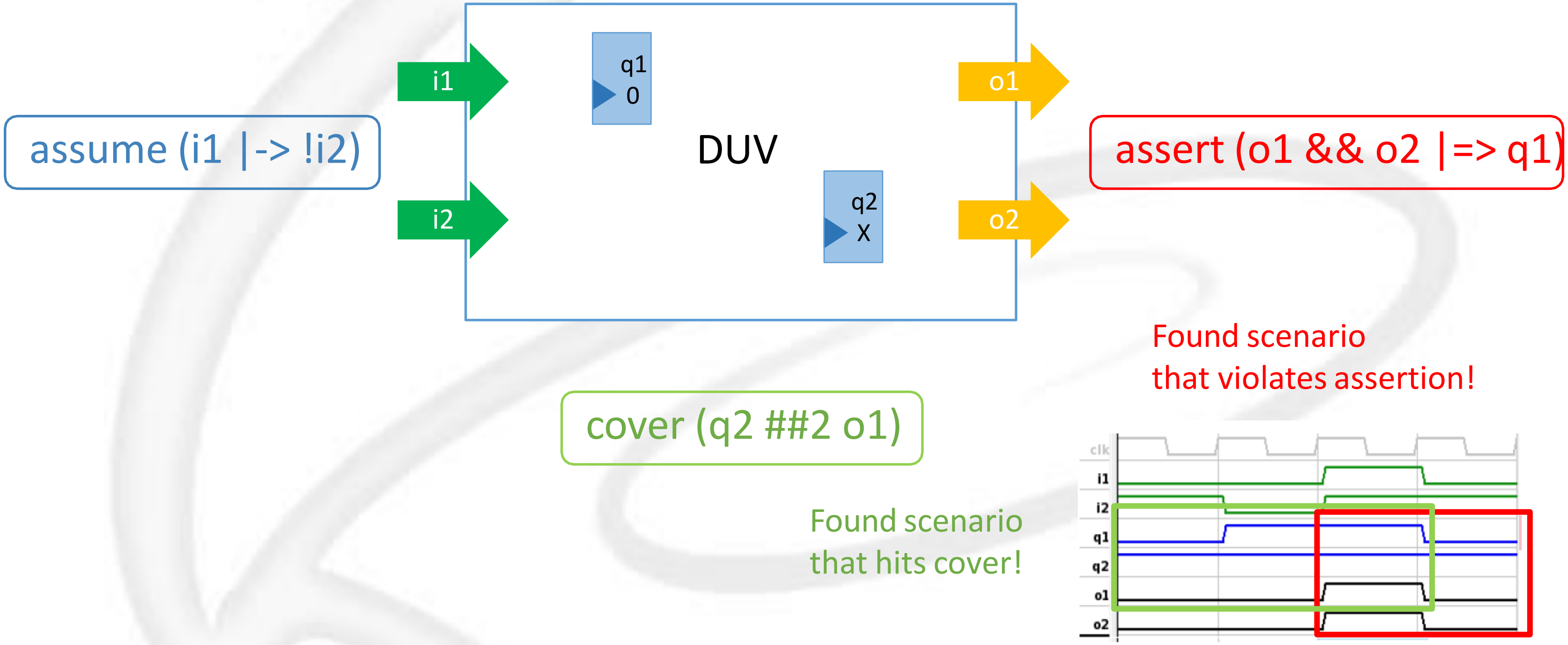


image from: https://melbournechapter.net/explore/open-trash-can-clipart/

Check

- □ Report when ***assert*** properties are violated
- □ Report when ***cover*** properties are hit



assume (i1 |-> !i2)

assert (o1 && o2 |=> q1)

cover (q2 ##2 o1)

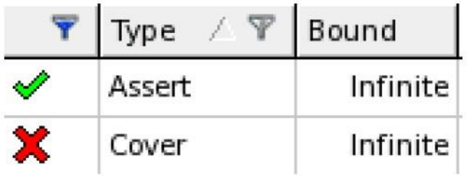
Found scenario that violates assertion!

Found scenario that hits cover!

Proof Results

Proven / Unreachable

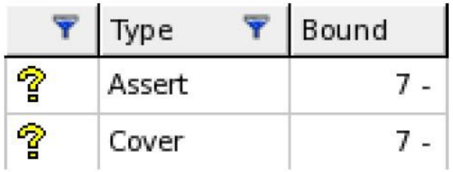
(Full Proof)



	Type	Bound
✓	Assert	Infinite
✗	Cover	Infinite

- Formal analyzed all reachable states
- Impossible to violate assertion
- Impossible to hit cover

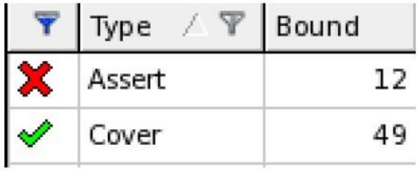
Undetermined



	Type	Bound
?	Assert	7 -
?	Cover	7 -

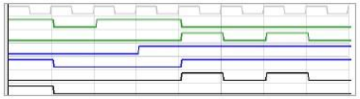
- Formal analyzed a subset of all reachable states
- Assertion does not fail in these states
- Cover is not hit in these states

CEX / Covered



	Type	Bound
✗	Assert	12
✓	Cover	49

- Formal found a state that hits a property
- Assertion failure
- Cover hit
- Waveform available



symbol	meaning
●	Assumption
🟨	Unprocessed property
⚡	Processing property
✗	Assert: Counterexample(CEX) found Cover: Proven as unreachable
✓	Assert: Proven as always true Cover: Cover trace found
?	Undetermined property

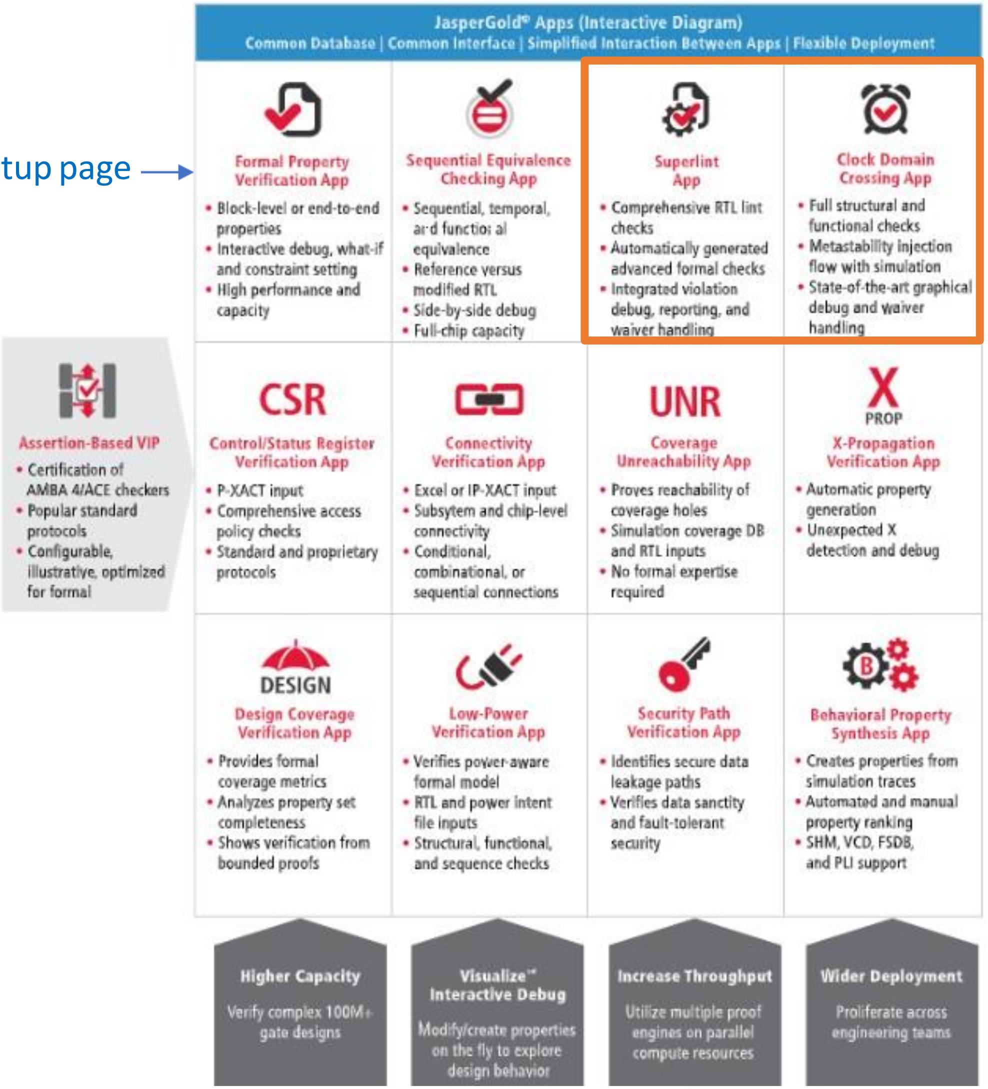
Outline

- □ Concept of Formal Verification
- □ JasperGold Tool Guide
- □ Exercise

JasperGold

- JasperGold is a platform developed by Cadence

Startup page →



JasperGold® Apps (Interactive Diagram)

Common Database | Common Interface | Simplified Interaction Between Apps | Flexible Deployment

Formal Property Verification App
- Block-level or end-to-end properties
- Interactive debug, what-if and constraint setting
- High performance and capacity

Sequential Equivalence Checking App
- Sequential, temporal, and functional equivalence
- Reference versus modified RTL
- Side-by-side debug
- Full-chip capacity

Superlint App
- Comprehensive RTL lint checks
- Automatically generated advanced formal checks
- Integrated violation debug, reporting, and waiver handling

Clock Domain Crossing App
- Full structural and functional checks
- Metastability injection flow with simulation
- State-of-the-art graphical debug and waiver handling

For designer

Assertion-Based VIP
- Certification of AMBA 4/ACE checkers
- Popular standard protocols
- Configurable, illustrative, optimized for formal

CSR

Control/Status Register Verification App
- P-XACT input
- Comprehensive access policy checks
- Standard and proprietary protocols

Connectivity Verification App
- Excel or IP-XACT input
- Subsytem and chip-level connectivity
- Conditional, combinational, or sequential connections

UNR

Coverage Unreachability App
- Proves reachability of coverage holes
- Simulation coverage DB and RTL inputs
- No formal expertise required

X PROP

X-Propagation Verification App
- Automatic property generation
- Unexpected X detection and debug

For verification

DESIGN

Design Coverage Verification App
- Provides formal coverage metrics
- Analyzes property set completeness
- Shows verification from bounded proofs

Low-Power Verification App
- Verifies power-aware formal model
- RTL and power intent file inputs
- Structural, functional, and sequence checks

Security Path Verification App
- Identifies secure data leakage paths
- Verifies data sanctity and fault-tolerant security

Behavioral Property Synthesis App
- Creates properties from simulation traces
- Automated and manual property ranking
- SHM, VCD, FSDB, and PLI support

Higher Capacity

Verify complex 100M+ gate designs

Visualize™ Interactive Debug

Modify/create properties on the fly to explore design behavior

Increase Throughput

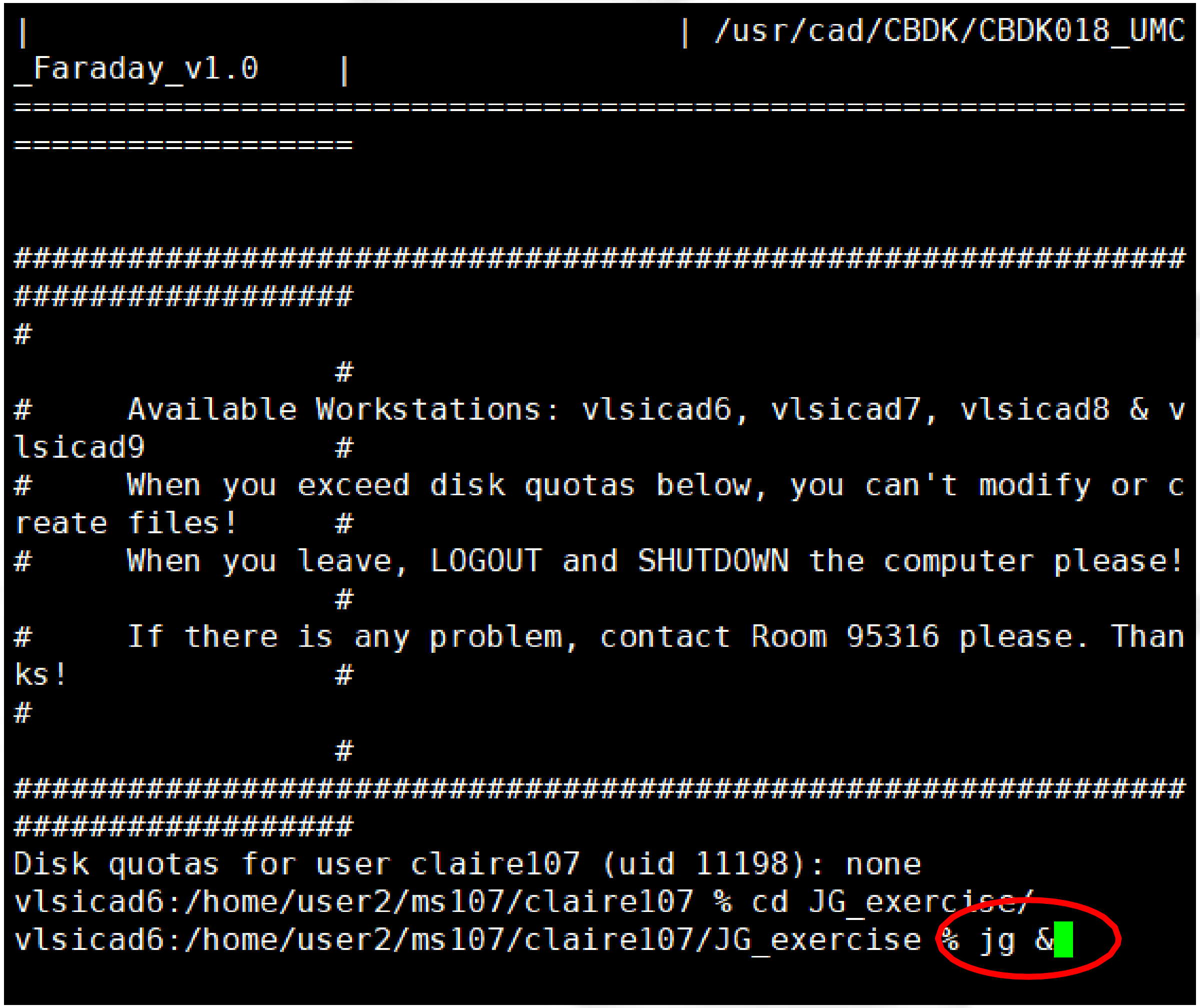
Utilize multiple proof engines on parallel compute resources

Wider Deployment

Proliferate across engineering teams

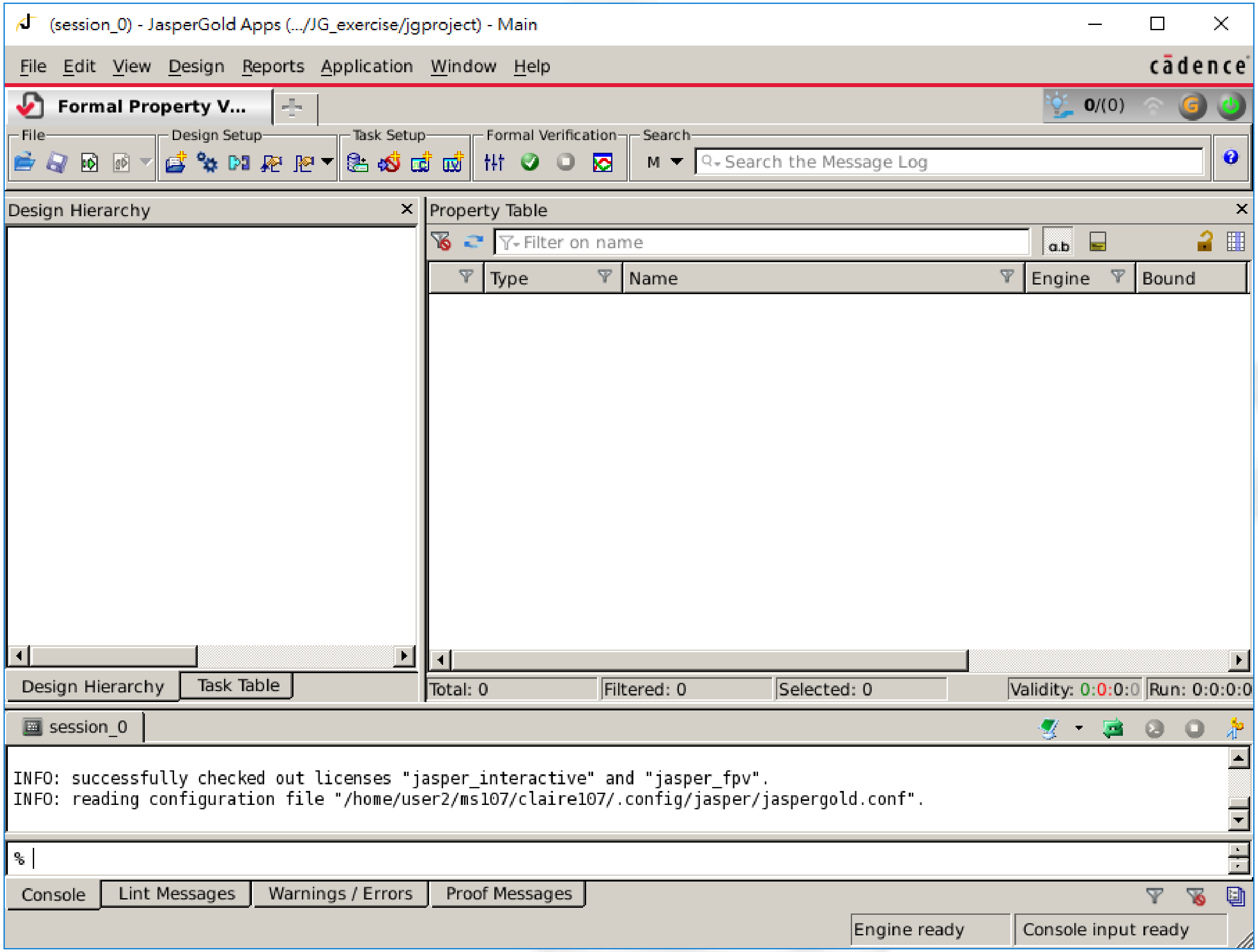
Start Jasper Gold(1/2)

- Command: jg &(run in background)



```
|                                  | /usr/cad/CBDK/CBDK018_UMC
_Faraday_v1.0     |
==============================================================
==================



##############################################################
##################
#
                 #
#     Available Workstations: vlsicad6, vlsicad7, vlsicad8 & v
lsicad9          #
#     When you exceed disk quotas below, you can't modify or c
reate files!     #
#     When you leave, LOGOUT and SHUTDOWN the computer please!
                 #
#     If there is any problem, contact Room 95316 please. Than
ks!              #
#
                 #
##############################################################
##################
Disk quotas for user claire107 (uid 11198): none
vlsicad6:/home/user2/ms107/claire107 % cd JG_exercise/
vlsicad6:/home/user2/ms107/claire107/JG_exercise % jg &
```

Start Jasper Gold(2/2)

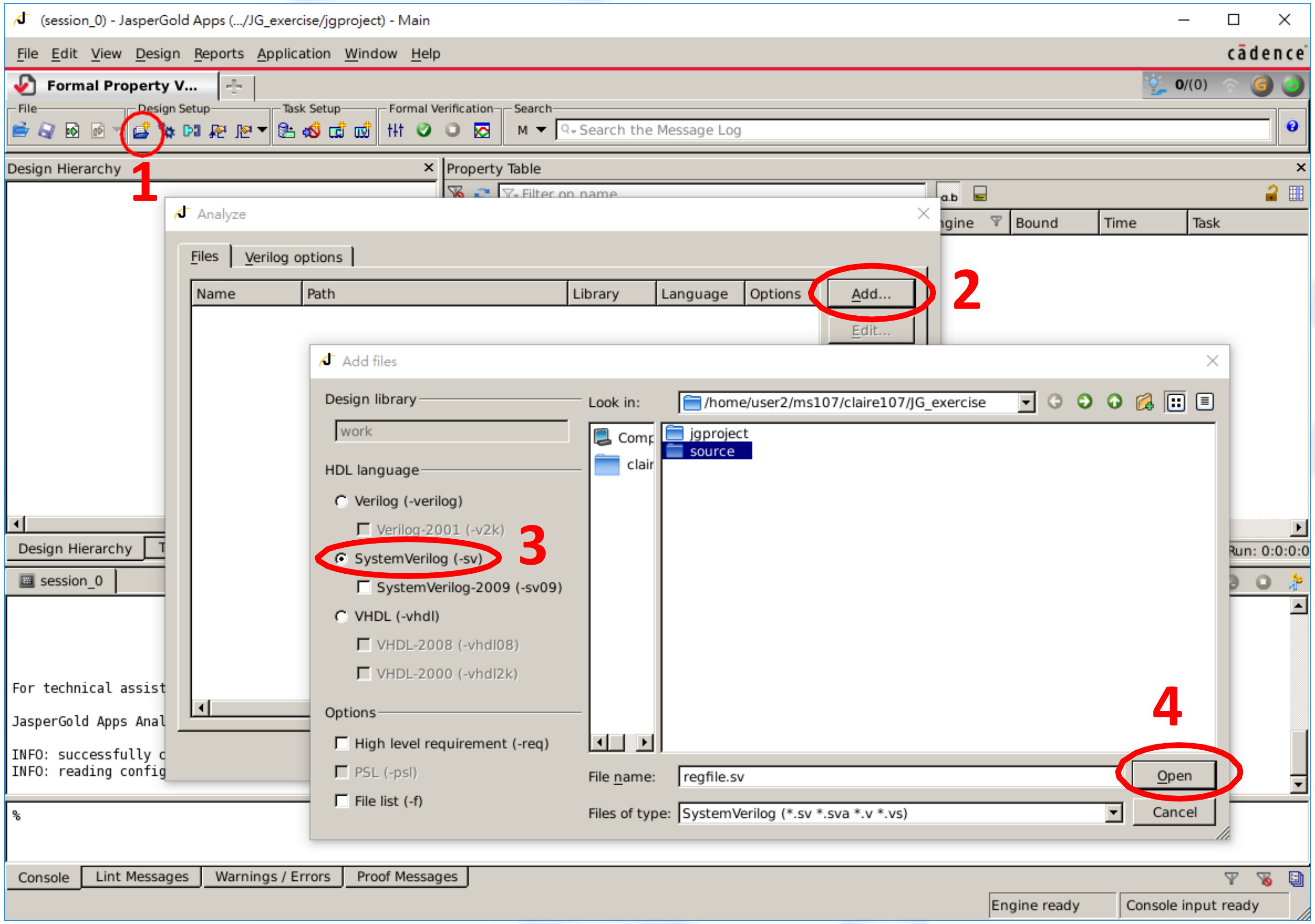


```
INFO: successfully checked out licenses "jasper_interactive" and "jasper_fpv".
INFO: reading configuration file "/home/user2/ms107/claire107/.config/jasper/jaspergold.conf".
```

```
% |
```

Analyzing Design and SVA Files (1/4)

- Click the ***Analyze*** wizard button.



Design library: work

HDL language: ● SystemVerilog (-sv)

Look in: /home/user2/ms107/claire107/JG_exercise

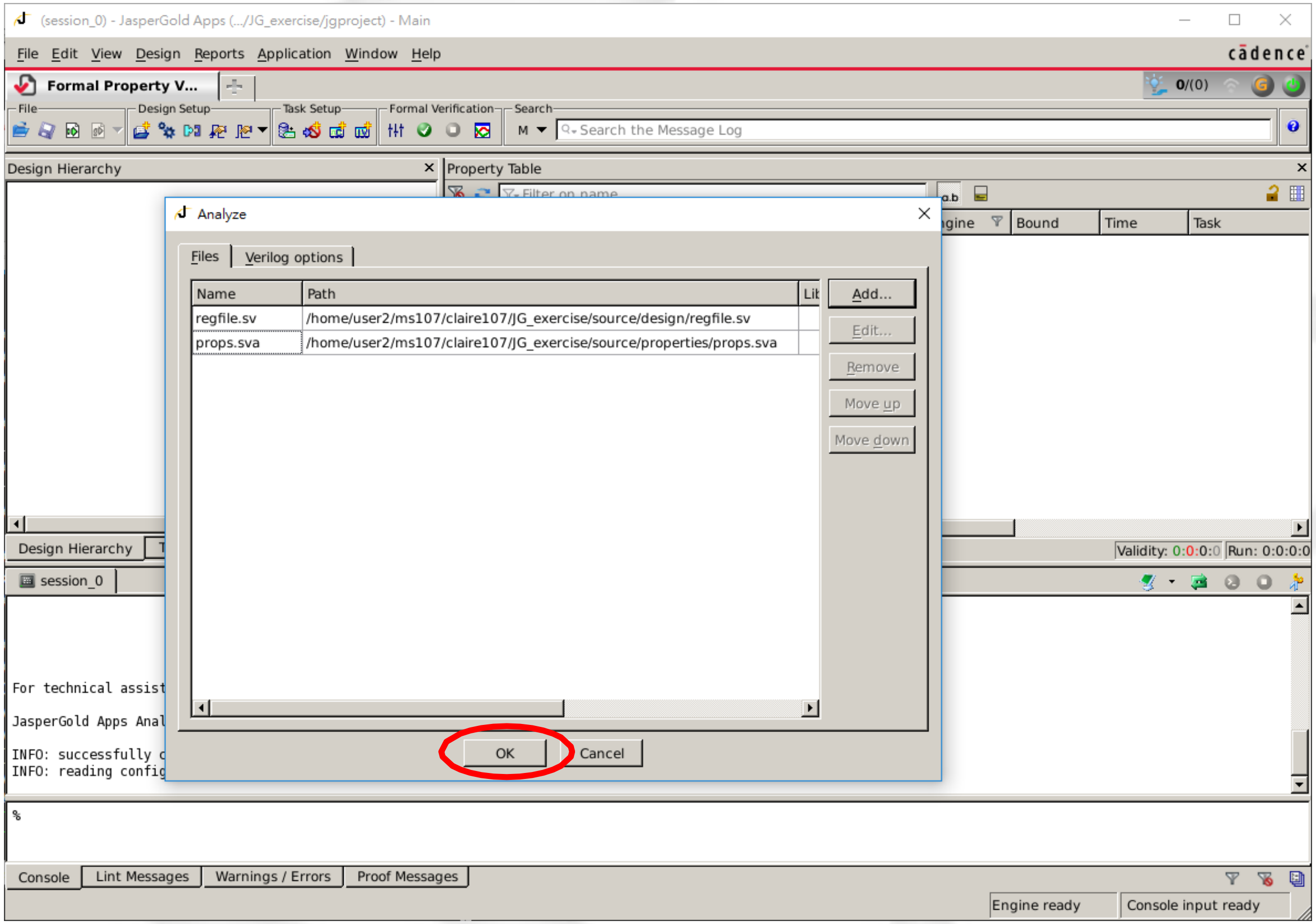
File name: regfile.sv

Files of type: SystemVerilog (*.sv *.sva *.v *.vs)

```
For technical assist

JasperGold Apps Anal

INFO: successfully c
INFO: reading config
%
```

Analyzing Design and SVA Files (2/4)

- Open design and SVA file



(session_0) - JasperGold Apps (.../JG_exercise/jgproject) - Main

Analyze

Name	Path	Lit
regfile.sv	/home/user2/ms107/claire107/JG_exercise/source/design/regfile.sv	
props.sva	/home/user2/ms107/claire107/JG_exercise/source/properties/props.sva	

```
For technical assist
JasperGold Apps Anal
INFO: successfully c
INFO: reading config
```

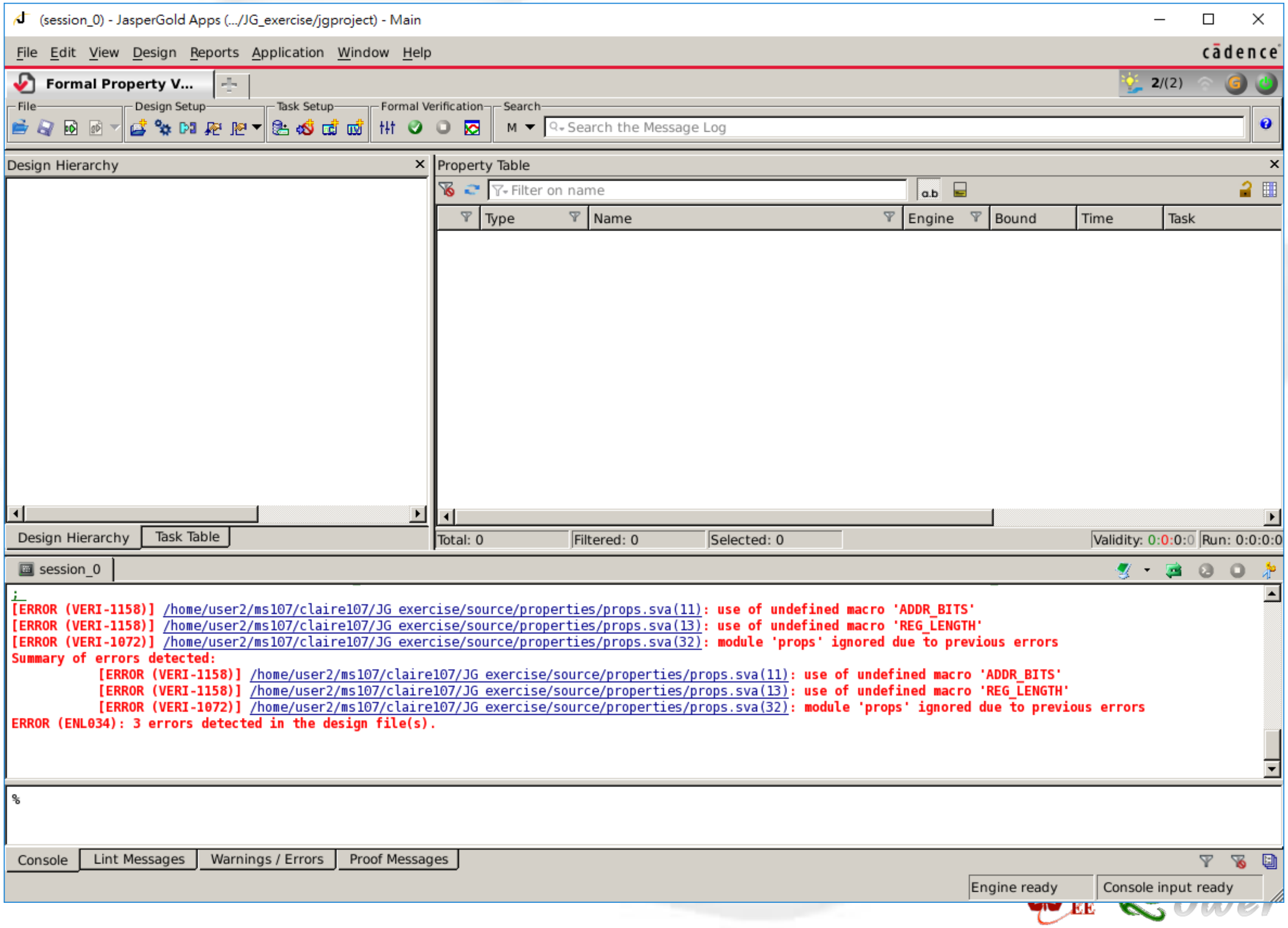
Tcl:

analyze -sv ./source/design/regfile.sv

analyze -sv ./source/properties/props.sva

Analyzing Design and SVA Files (3/4)

- If any errors or warnings show in the terminal window, you should modify your codes to avoid them.

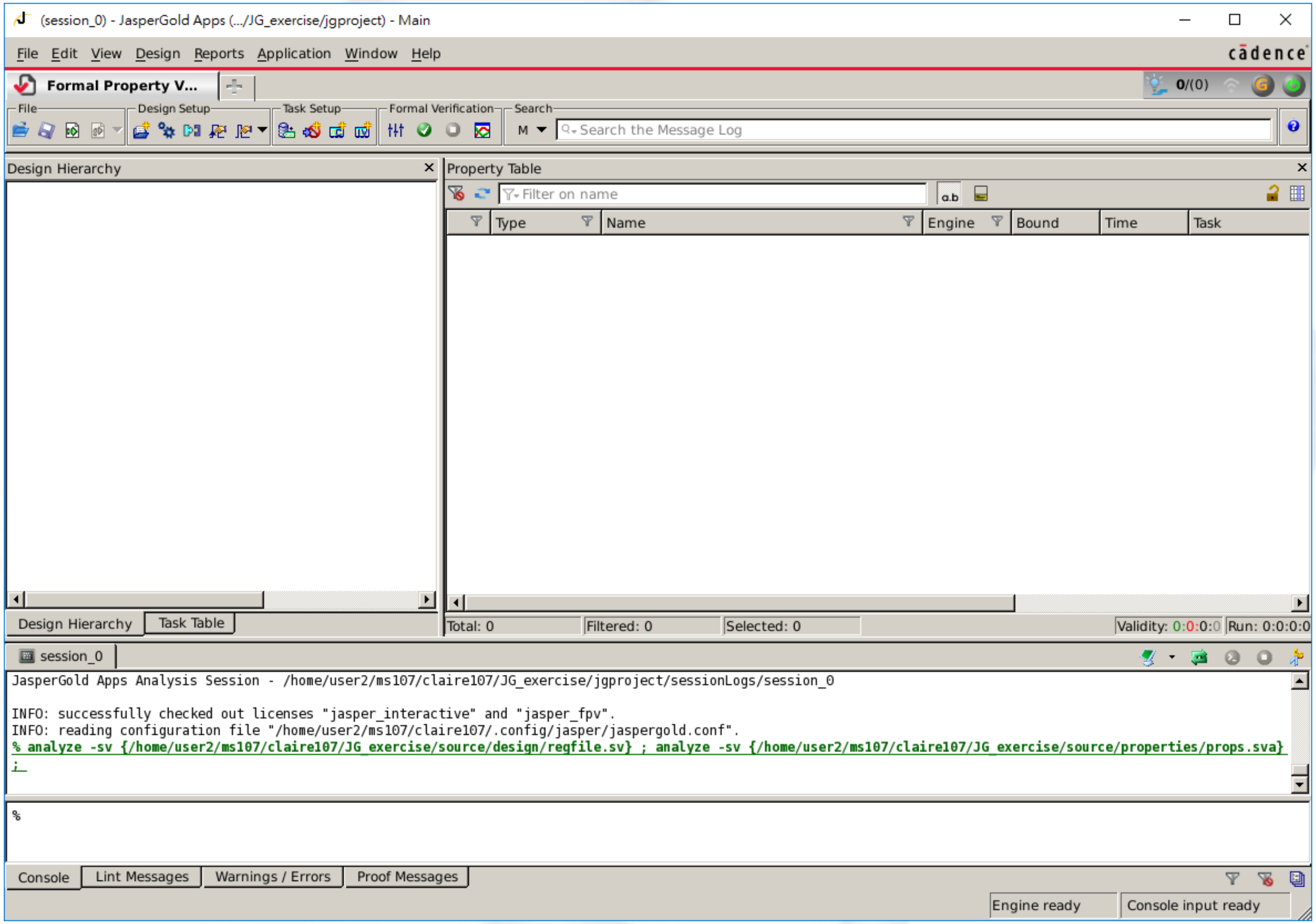


```
;
[ERROR (VERI-1158)] /home/user2/ms107/claire107/JG_exercise/source/properties/props.sva(11): use of undefined macro 'ADDR_BITS'
[ERROR (VERI-1158)] /home/user2/ms107/claire107/JG_exercise/source/properties/props.sva(13): use of undefined macro 'REG_LENGTH'
[ERROR (VERI-1072)] /home/user2/ms107/claire107/JG_exercise/source/properties/props.sva(32): module 'props' ignored due to previous errors
Summary of errors detected:
          [ERROR (VERI-1158)] /home/user2/ms107/claire107/JG_exercise/source/properties/props.sva(11): use of undefined macro 'ADDR_BITS'
          [ERROR (VERI-1158)] /home/user2/ms107/claire107/JG_exercise/source/properties/props.sva(13): use of undefined macro 'REG_LENGTH'
          [ERROR (VERI-1072)] /home/user2/ms107/claire107/JG_exercise/source/properties/props.sva(32): module 'props' ignored due to previous errors
ERROR (ENL034): 3 errors detected in the design file(s).
```

```
%
```

Analyzing Design and SVA Files (4/4)

- syntax check pass

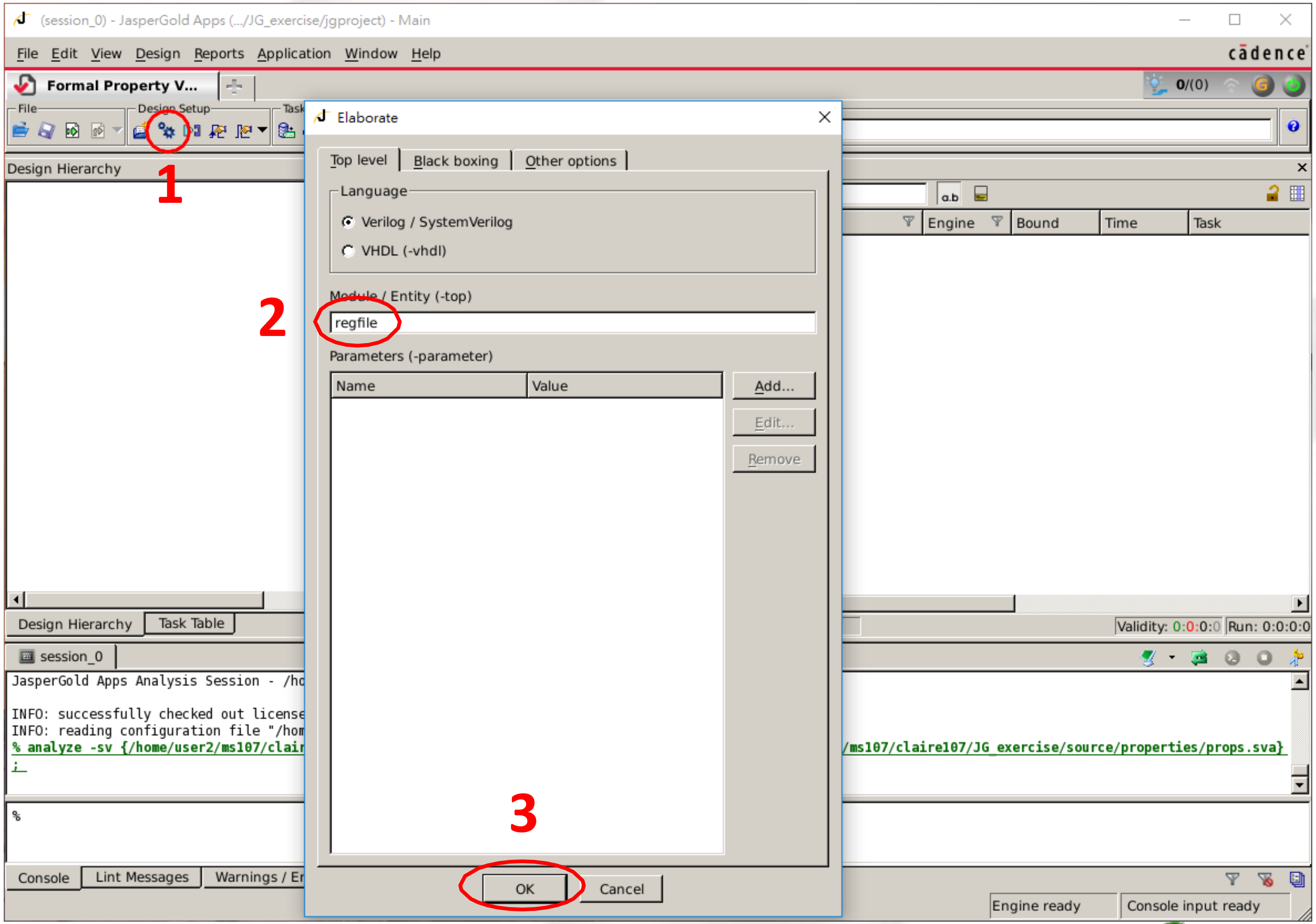


```
JasperGold Apps Analysis Session - /home/user2/ms107/claire107/JG_exercise/jgproject/sessionLogs/session_0

INFO: successfully checked out licenses "jasper_interactive" and "jasper_fpv".
INFO: reading configuration file "/home/user2/ms107/claire107/.config/jasper/jaspergold.conf".
% analyze -sv {/home/user2/ms107/claire107/JG_exercise/source/design/regfile.sv} ; analyze -sv {/home/user2/ms107/claire107/JG_exercise/source/properties/props.sva}
;
```

```
%
```

Elaborate Design Hierarchy (1/2)

- Click the ***Elaborate*** wizard button



Language
● Verilog / SystemVerilog
○ VHDL (-vhdl)

Module / Entity (-top): regfile

Parameters (-parameter)

Name	Value

```
JasperGold Apps Analysis Session - /ho
INFO: successfully checked out license
INFO: reading configuration file "/hom
% analyze -sv {/home/user2/ms107/clair    /ms107/claire107/JG_exercise/source/properties/props.sva}
;
%
```

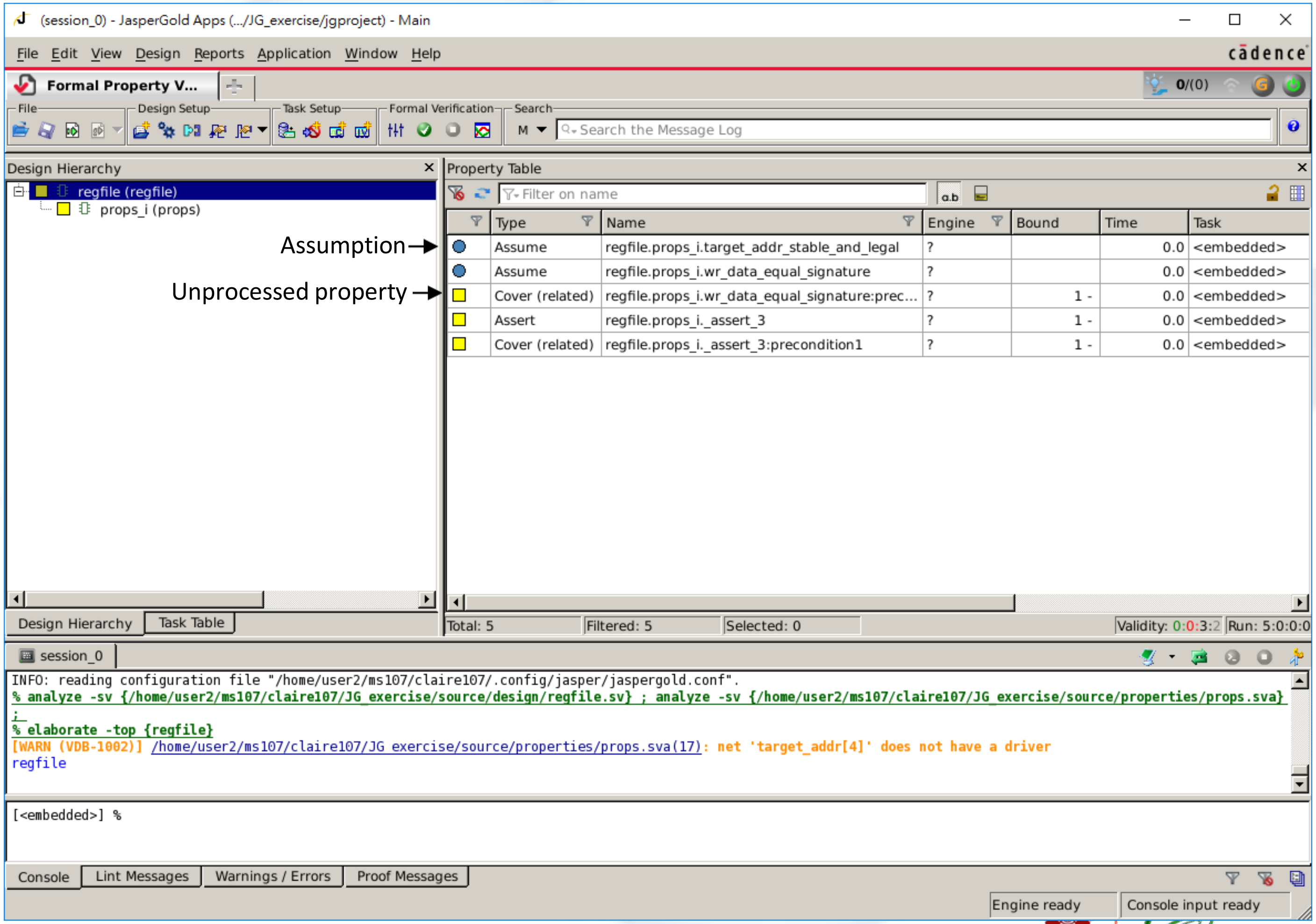
Validity: 0:0:0:0 Run: 0:0:0:0

Engine ready | Console input ready

Tcl: elaborate -top regfile

Elaborate Design Hierarchy (2/2)

- Elaborates the design hierarchy, synthesizes the netlist, and ex-tracts the embedded properties.



Assumption

Unprocessed property

	Type	Name	Engine	Bound	Time	Task
●	Assume	regfile.props_i.target_addr_stable_and_legal	?		0.0	<embedded>
●	Assume	regfile.props_i.wr_data_equal_signature	?		0.0	<embedded>
□	Cover (related)	regfile.props_i.wr_data_equal_signature:prec...	?	1 -	0.0	<embedded>
□	Assert	regfile.props_i._assert_3	?	1 -	0.0	<embedded>
□	Cover (related)	regfile.props_i._assert_3:precondition1	?	1 -	0.0	<embedded>

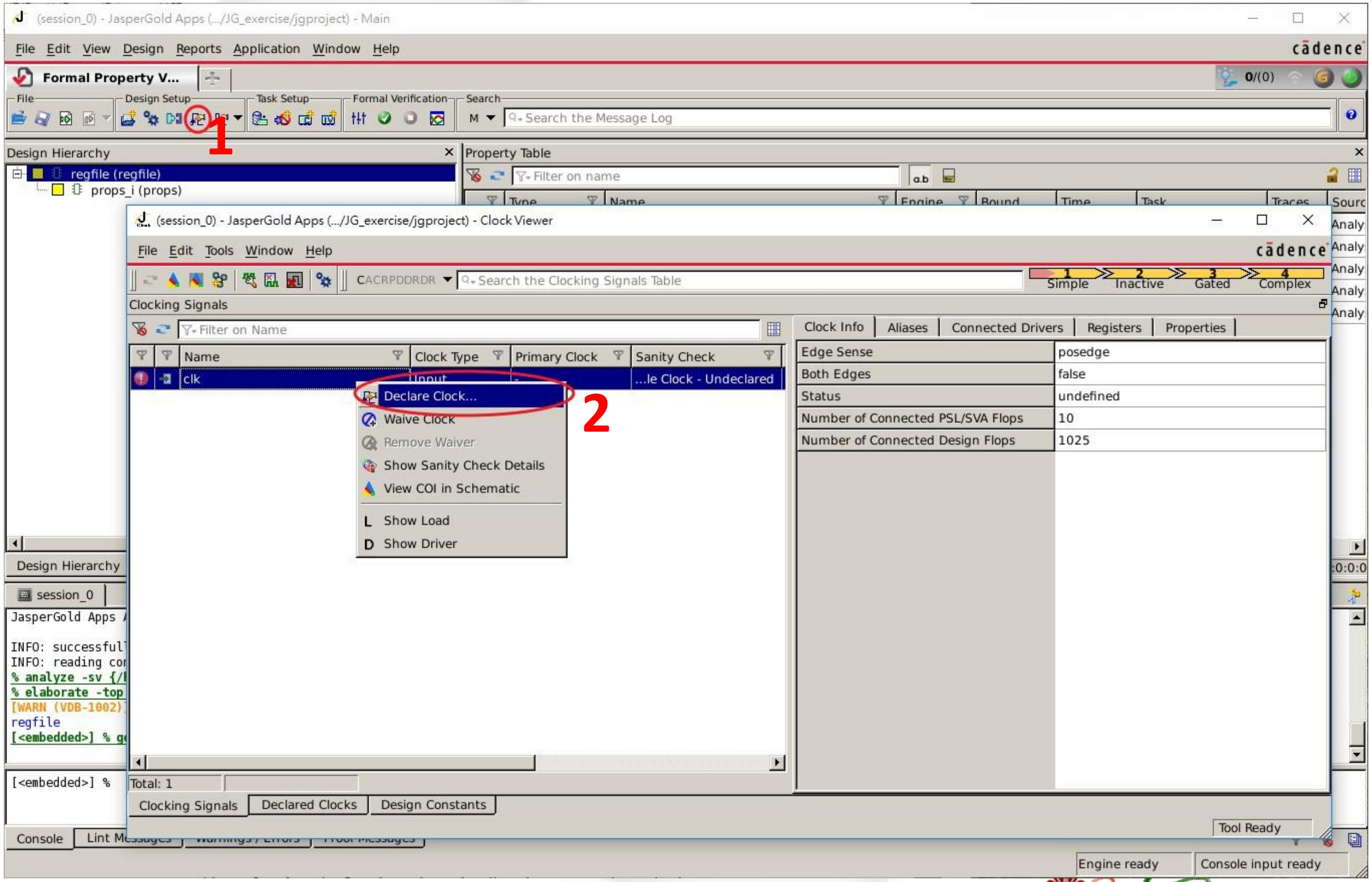
Total: 5 Filtered: 5 Selected: 0 Validity: 0:0:3:2 Run: 5:0:0:0

```
INFO: reading configuration file "/home/user2/ms107/claire107/.config/jasper/jaspergold.conf".
% analyze -sv {/home/user2/ms107/claire107/JG_exercise/source/design/regfile.sv} ; analyze -sv {/home/user2/ms107/claire107/JG_exercise/source/properties/props.sva}
;
% elaborate -top {regfile}
[WARN (VDB-1002)] /home/user2/ms107/claire107/JG_exercise/source/properties/props.sva(17): net 'target_addr[4]' does not have a driver
regfile
```

```
[<embedded>] %
```

Specifying the Clock (1/2)

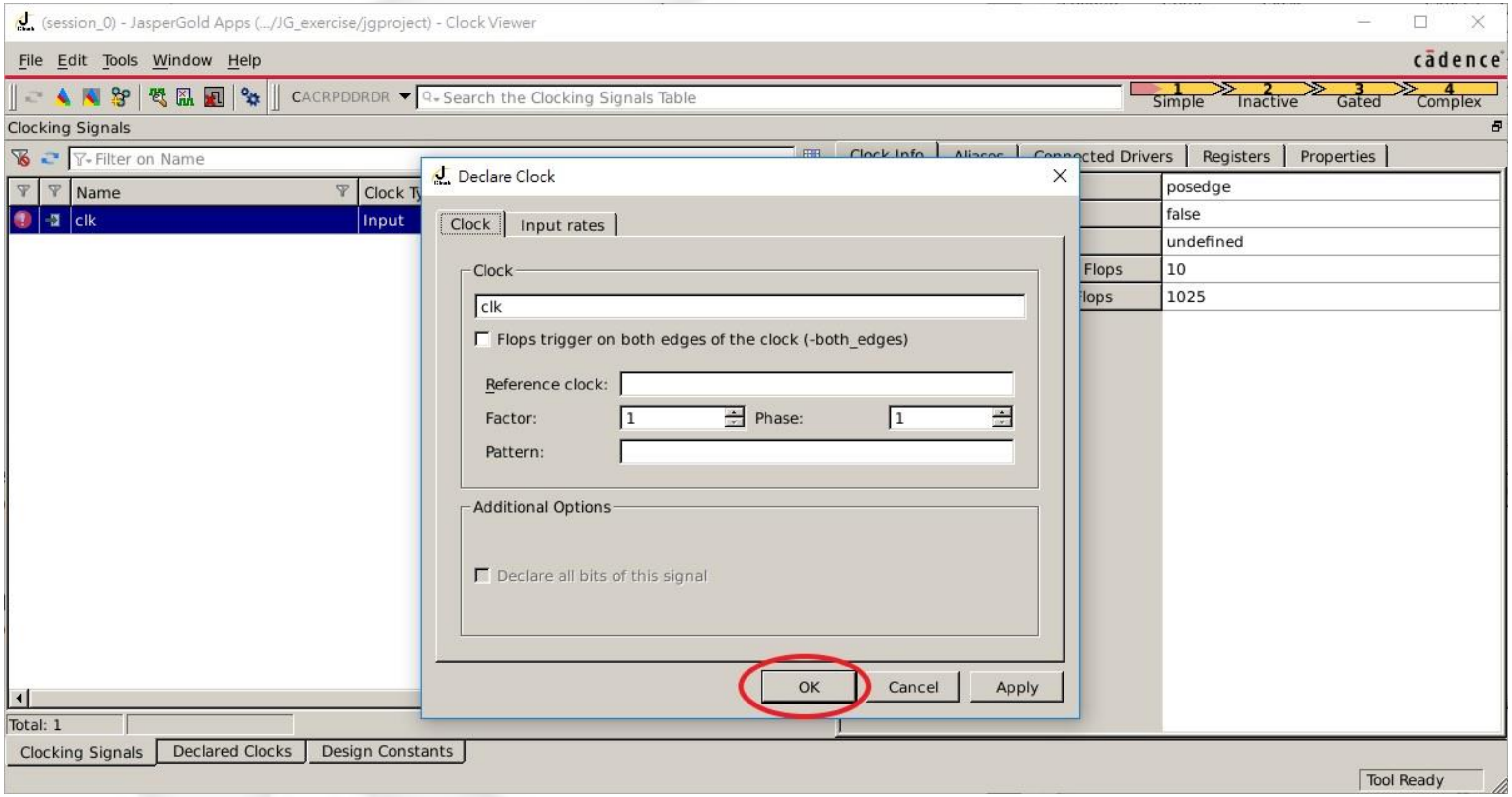
- Click the ***Clock Viewer*** button
- Right-click clk and select Declare Clock...



Edge Sense	posedge
Both Edges	false
Status	undefined
Number of Connected PSL/SVA Flops	10
Number of Connected Design Flops	1025

```
JasperGold Apps A

INFO: successful
INFO: reading cor
% analyze -sv {/
% elaborate -top
[WARN (VDB-1002)]
regfile
[<embedded>] % ge

[<embedded>] %
```

Specifying the Clock (2/2)



(session_0) - JasperGold Apps (.../JG_exercise/jgproject) - Clock Viewer

File Edit Tools Window Help

Search the Clocking Signals Table

Clocking Signals

Name	Clock Ty
clk	Input

Total: 1

	posedge
	false
	undefined
Flops	10
lops	1025

Declare Clock

Clock

clk

□ Flops trigger on both edges of the clock (-both_edges)

Reference clock:

Factor: 1 Phase: 1

Pattern:

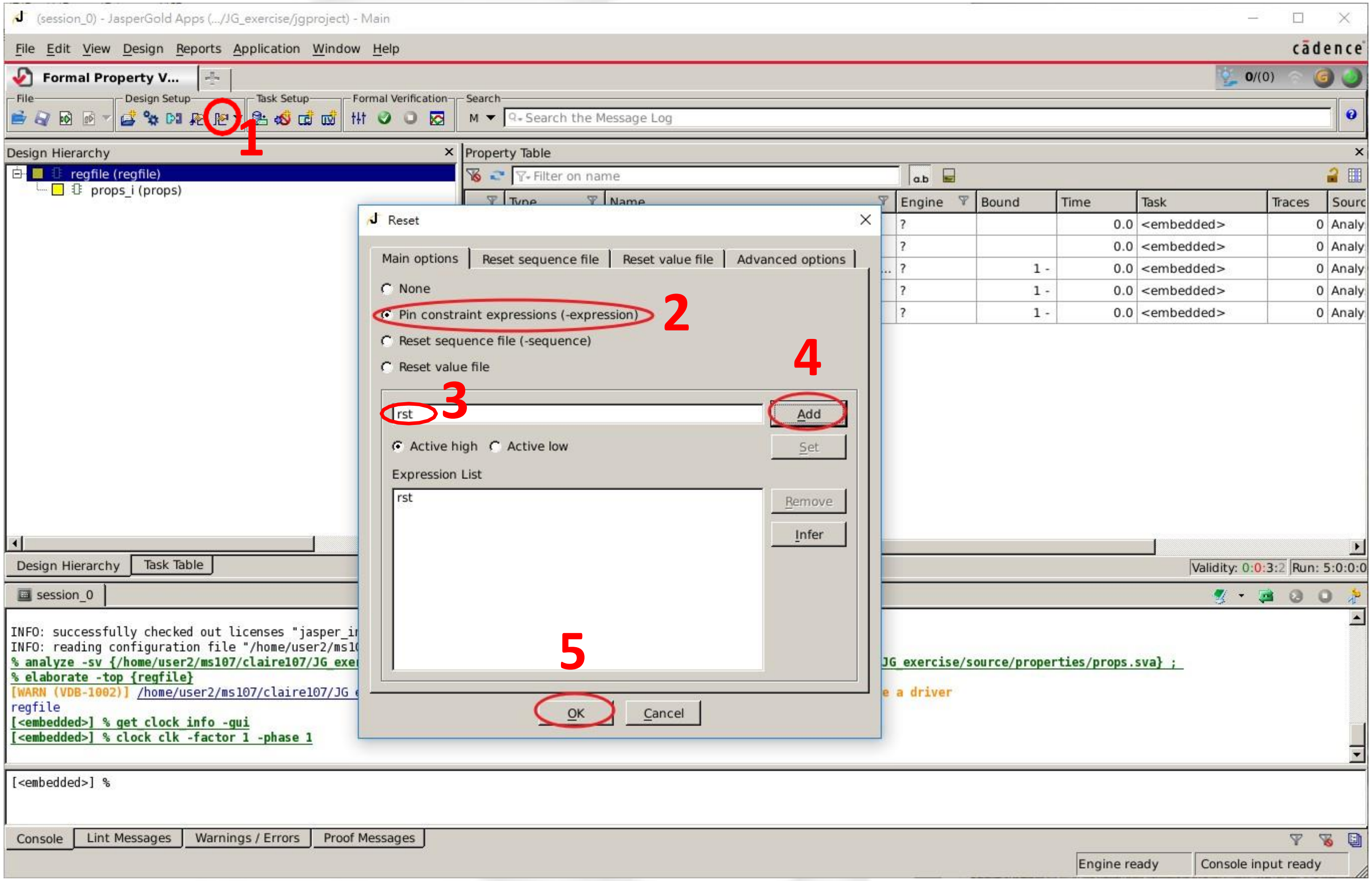
Additional Options

□ Declare all bits of this signal

Tcl: clock clk

Specifying the Reset

- Click the ***Reset*** wizard button.



(session_0) - JasperGold Apps (.../JG_exercise/jgproject) - Main

Type	Name	Engine	Bound	Time	Task	Traces	Sourc
		?		0.0	<embedded>	0	Analy
		?		0.0	<embedded>	0	Analy
	...	?	1 -	0.0	<embedded>	0	Analy
		?	1 -	0.0	<embedded>	0	Analy
		?	1 -	0.0	<embedded>	0	Analy

Validity: 0:0:3:2 Run: 5:0:0:0

Reset

Main options | Reset sequence file | Reset value file | Advanced options

○ None

● Pin constraint expressions (-expression) **2**

○ Reset sequence file (-sequence)

○ Reset value file

4

3 rst [Add]

● Active high ○ Active low [Set]

Expression List

rst

[Remove] [Infer]

5

[OK] [Cancel]

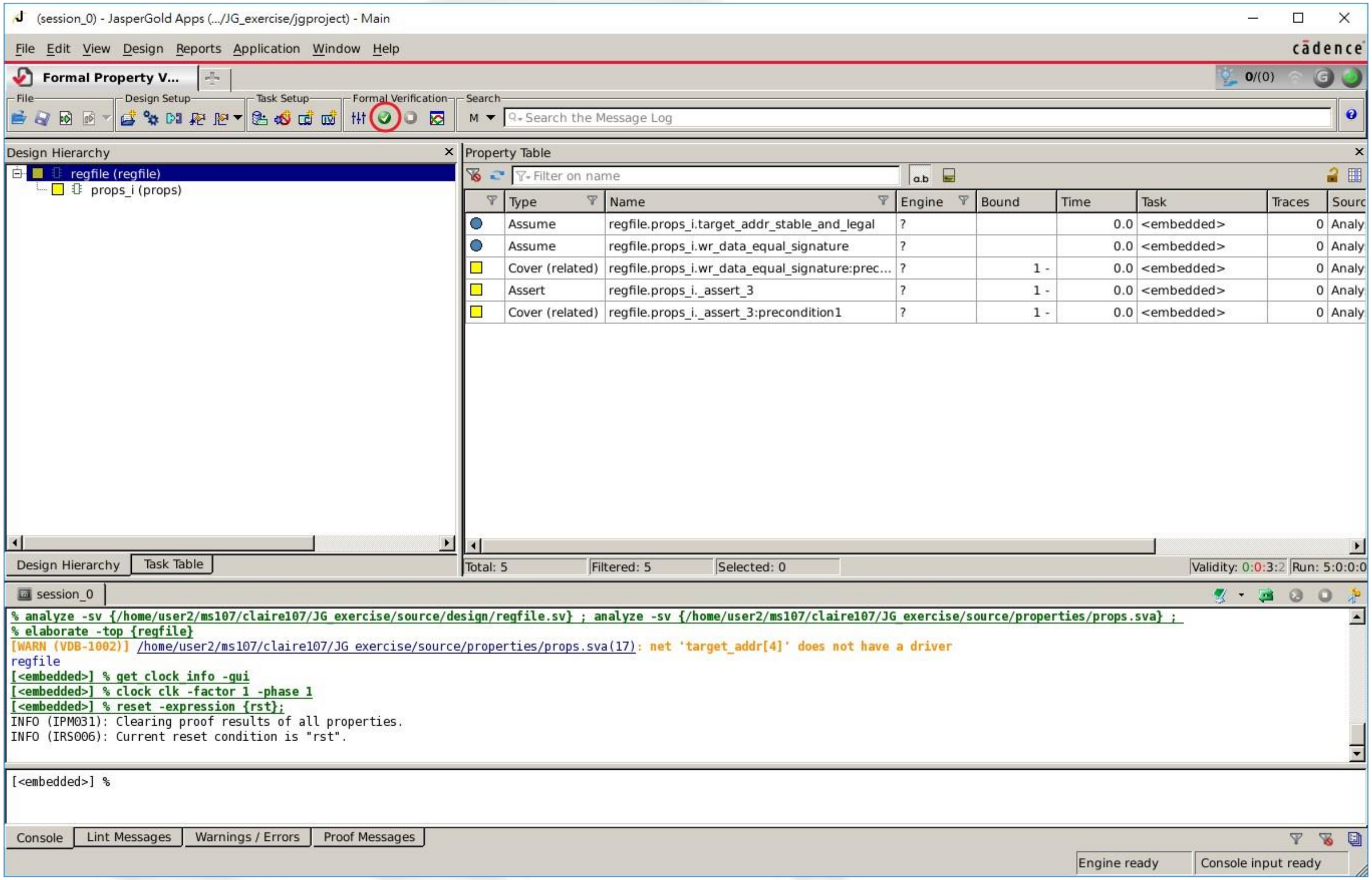
```
INFO: successfully checked out licenses "jasper_i
INFO: reading configuration file "/home/user2/ms1
% analyze -sv {/home/user2/ms107/claire107/JG_exe     JG_exercise/source/properties/props.sva} ;
% elaborate -top {regfile}
[WARN (VDB-1002)] /home/user2/ms107/claire107/JG      e a driver
regfile
[<embedded>] % get_clock_info -gui
[<embedded>] % clock clk -factor 1 -phase 1
```

```
[<embedded>] %
```

Engine ready Console input ready

Tcl: reset -expression {rst}

Prove All (1/2)



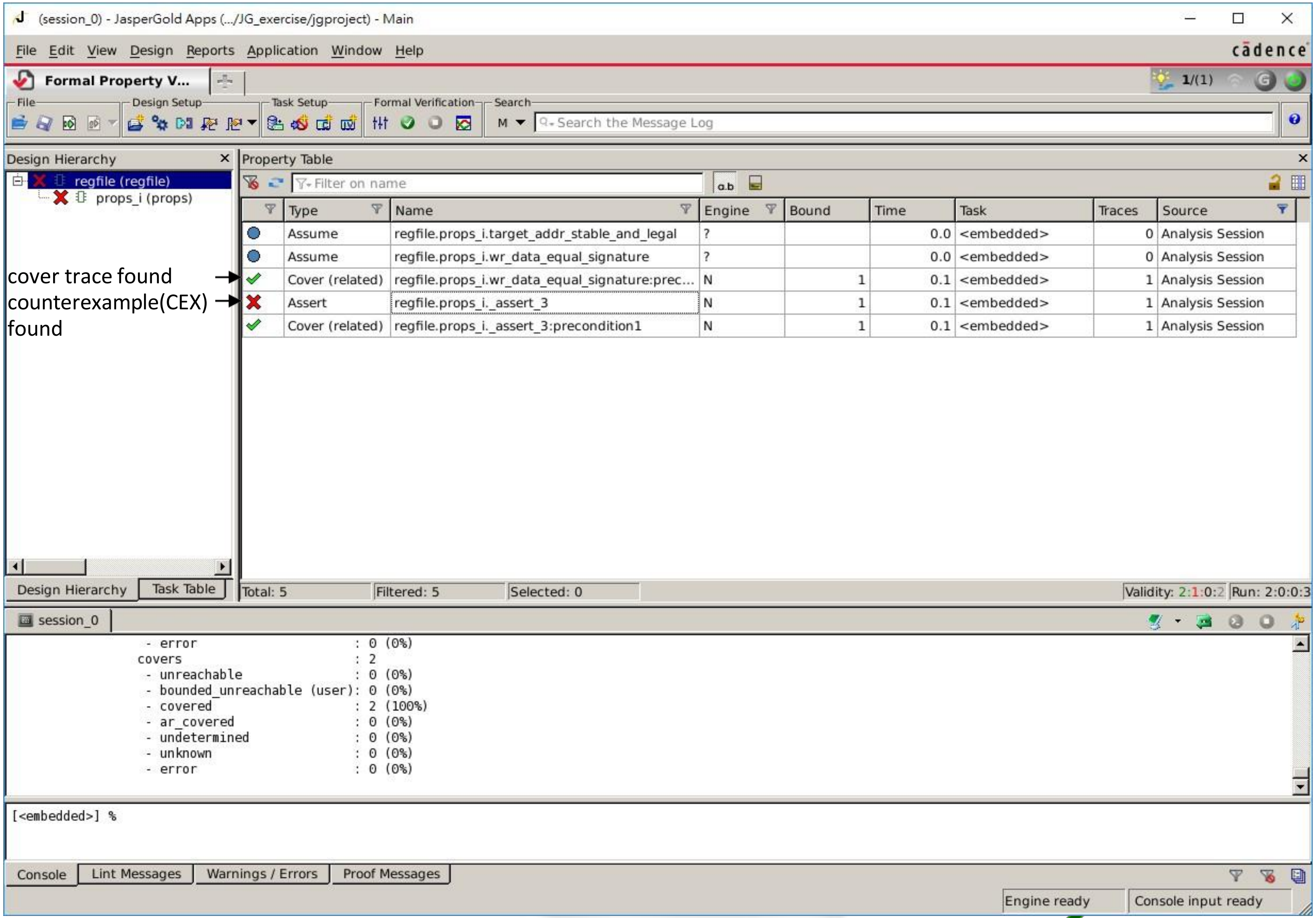
	Type	Name	Engine	Bound	Time	Task	Traces	Sourc
●	Assume	regfile.props_i.target_addr_stable_and_legal	?		0.0	<embedded>	0	Analy
●	Assume	regfile.props_i.wr_data_equal_signature	?		0.0	<embedded>	0	Analy
□	Cover (related)	regfile.props_i.wr_data_equal_signature:prec...	?	1 -	0.0	<embedded>	0	Analy
□	Assert	regfile.props_i._assert_3	?	1 -	0.0	<embedded>	0	Analy
□	Cover (related)	regfile.props_i._assert_3:precondition1	?	1 -	0.0	<embedded>	0	Analy

```
% analyze -sv {/home/user2/ms107/claire107/JG_exercise/source/design/regfile.sv} ; analyze -sv {/home/user2/ms107/claire107/JG_exercise/source/properties/props.sva} ;
% elaborate -top {regfile}
[WARN (VDB-1002)] /home/user2/ms107/claire107/JG_exercise/source/properties/props.sva(17): net 'target_addr[4]' does not have a driver
regfile
[<embedded>] % get_clock_info -gui
[<embedded>] % clock clk -factor 1 -phase 1
[<embedded>] % reset -expression {rst};
INFO (IPM031): Clearing proof results of all properties.
INFO (IRS006): Current reset condition is "rst".
```

```
[<embedded>] %
```

Tcl: prove -all

Prove All (2/2)



cover trace found

counterexample(CEX) found

	Type	Name	Engine	Bound	Time	Task	Traces	Source
●	Assume	regfile.props_i.target_addr_stable_and_legal	?		0.0	<embedded>	0	Analysis Session
●	Assume	regfile.props_i.wr_data_equal_signature	?		0.0	<embedded>	0	Analysis Session
✓	Cover (related)	regfile.props_i.wr_data_equal_signature:prec...	N	1	0.1	<embedded>	1	Analysis Session
✗	Assert	regfile.props_i._assert_3	N	1	0.1	<embedded>	1	Analysis Session
✓	Cover (related)	regfile.props_i._assert_3:precondition1	N	1	0.1	<embedded>	1	Analysis Session

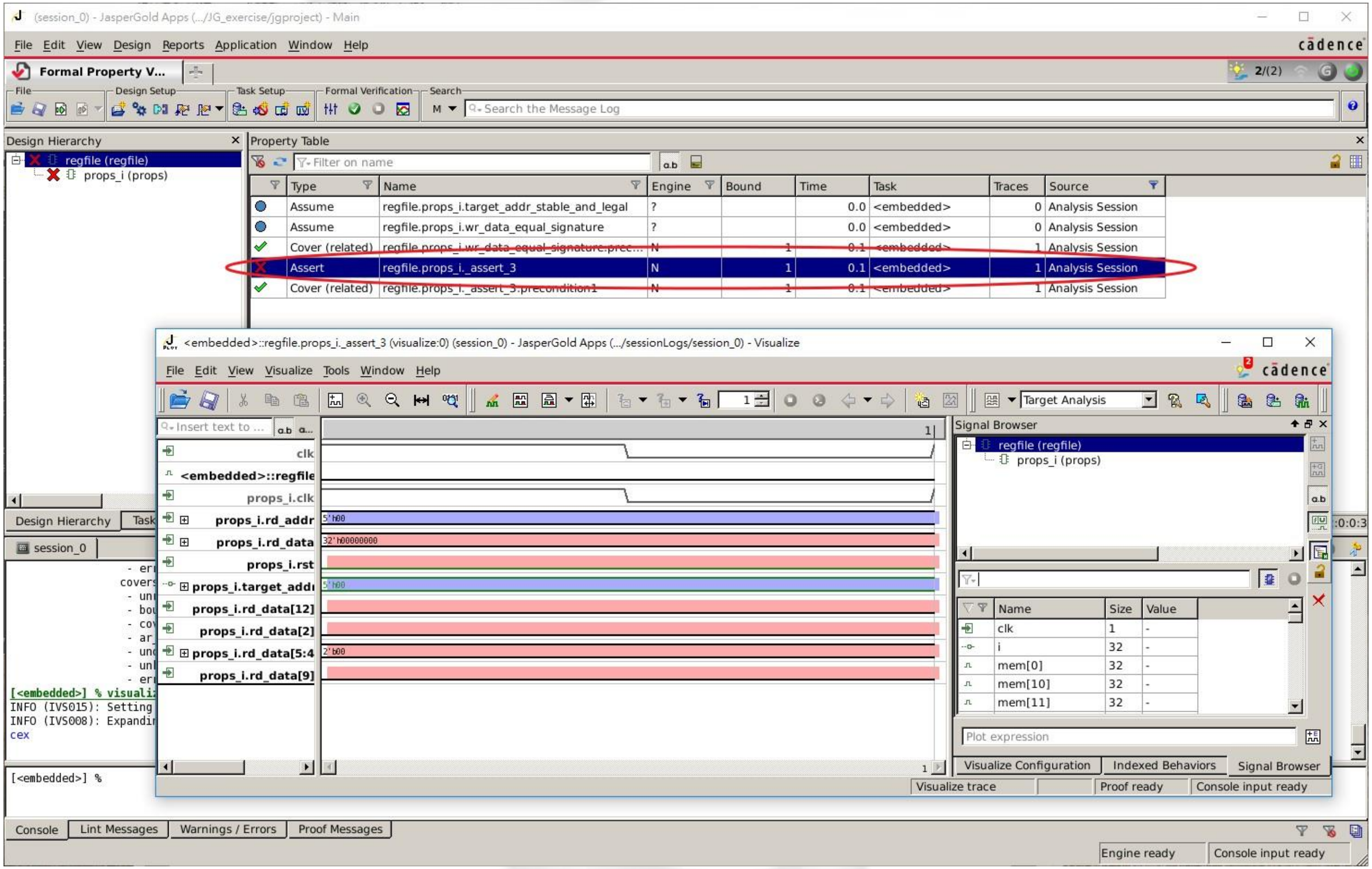
Total: 5 Filtered: 5 Selected: 0 Validity: 2:1:0:2 Run: 2:0:0:3

```
    - error                    : 0 (0%)
  covers                       : 2
    - unreachable              : 0 (0%)
    - bounded_unreachable (user): 0 (0%)
    - covered                  : 2 (100%)
    - ar_covered               : 0 (0%)
    - undetermined             : 0 (0%)
    - unknown                  : 0 (0%)
    - error                    : 0 (0%)
```

```
[<embedded>] %
```

Visualize (1/2)

- Double click the CEX to activate visualize window



(session_0) - JasperGold Apps (.../JG_exercise/jgproject) - Main

Property Table

	Type	Name	Engine	Bound	Time	Task	Traces	Source
●	Assume	regfile.props_i.target_addr_stable_and_legal	?		0.0	<embedded>	0	Analysis Session
●	Assume	regfile.props_i.wr_data_equal_signature	?		0.0	<embedded>	0	Analysis Session
✓	Cover (related)	regfile.props_i.wr_data_equal_signature.prec...	N	1	0.1	<embedded>	1	Analysis Session
✗	Assert	regfile.props_i._assert_3	N	1	0.1	<embedded>	1	Analysis Session
✓	Cover (related)	regfile.props_i._assert_3.precondition1	N	1	0.1	<embedded>	1	Analysis Session

<embedded>::regfile.props_i._assert_3 (visualize:0) (session_0) - JasperGold Apps (.../sessionLogs/session_0) - Visualize

```
clk
<embedded>::regfile
props_i.clk
props_i.rd_addr            5'h00
props_i.rd_data            32'h00000000
props_i.rst
props_i.target_addr        5'h00
props_i.rd_data[12]
props_i.rd_data[2]
props_i.rd_data[5:4        2'b00
props_i.rd_data[9]
```

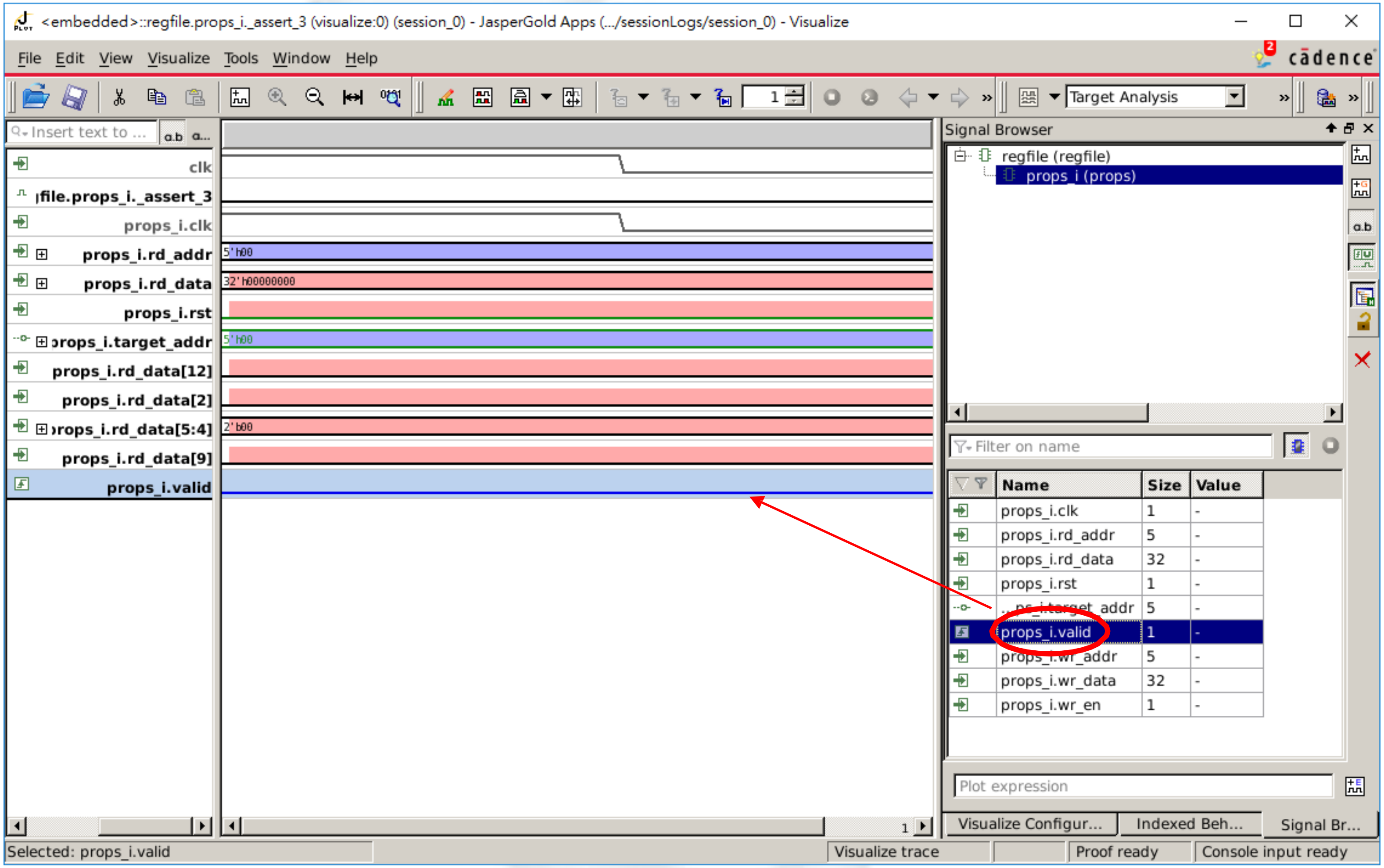
Signal Browser

Name	Size	Value
clk	1	-
i	32	-
mem[0]	32	-
mem[10]	32	-
mem[11]	32	-

```
  - err
covers
  - unr
  - bou
  - cov
  - ar_
  - unc
  - unl
  - err
[<embedded>] % visuali
INFO (IVS015): Setting
INFO (IVS008): Expandin
cex

[<embedded>] %
```

Visualize (2/2)

- Drag signal from signal browser to check the value



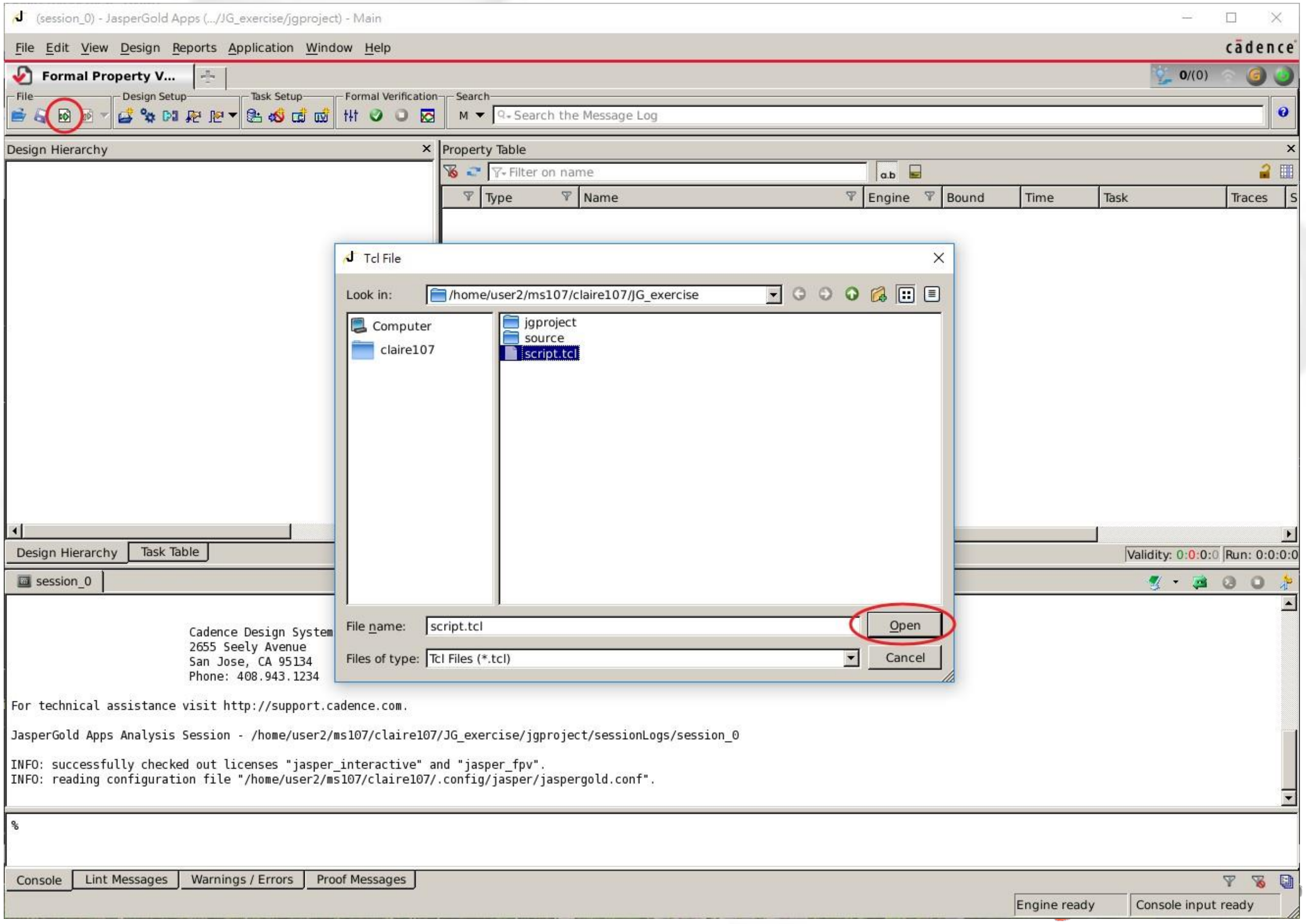
<embedded>::regfile.props_i._assert_3 (visualize:0) (session_0) - JasperGold Apps (.../sessionLogs/session_0) - Visualize

Name	Size	Value
props_i.clk	1	-
props_i.rd_addr	5	-
props_i.rd_data	32	-
props_i.rst	1	-
...ps_i.target_addr	5	-
props_i.valid	1	-
props_i.wr_addr	5	-
props_i.wr_data	32	-
props_i.wr_en	1	-

Selected: props_i.valid | Visualize trace | Proof ready | Console input ready

Load Tcl file

- In Terminal : jg <scriptfile> &
- In JasperGold GUI : click ***Source*** button



Look in: /home/user2/ms107/claire107/JG_exercise

File name: script.tcl

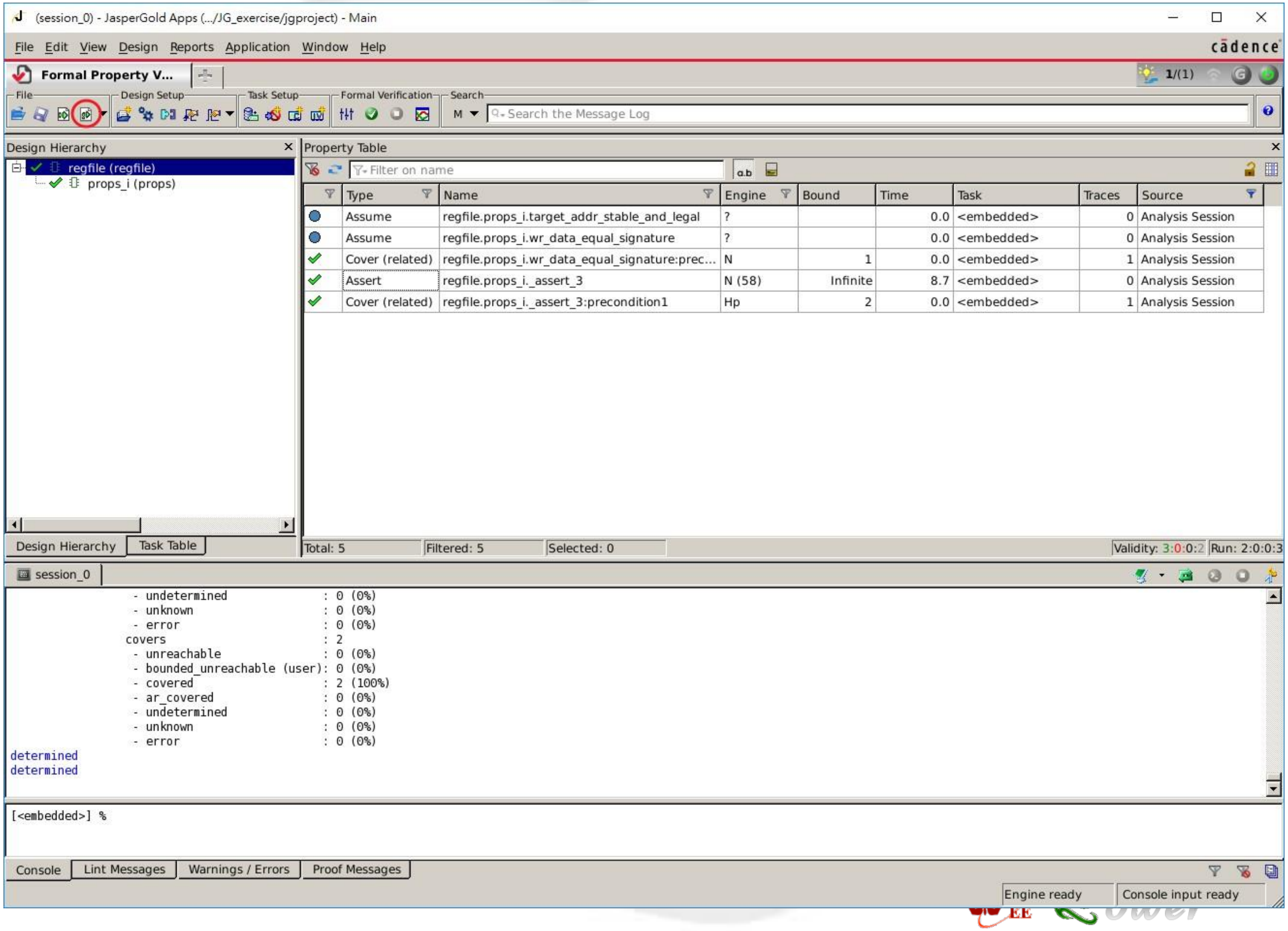
Files of type: Tcl Files (*.tcl)

```
                    Cadence Design System
                    2655 Seely Avenue
                    San Jose, CA 95134
                    Phone: 408.943.1234

For technical assistance visit http://support.cadence.com.

JasperGold Apps Analysis Session - /home/user2/ms107/claire107/JG_exercise/jgproject/sessionLogs/session_0

INFO: successfully checked out licenses "jasper_interactive" and "jasper_fpv".
INFO: reading configuration file "/home/user2/ms107/claire107/.config/jasper/jaspergold.conf".
```

```
%
```

Reload Tcl file

- Click ***Source Recent/Database Script*** button



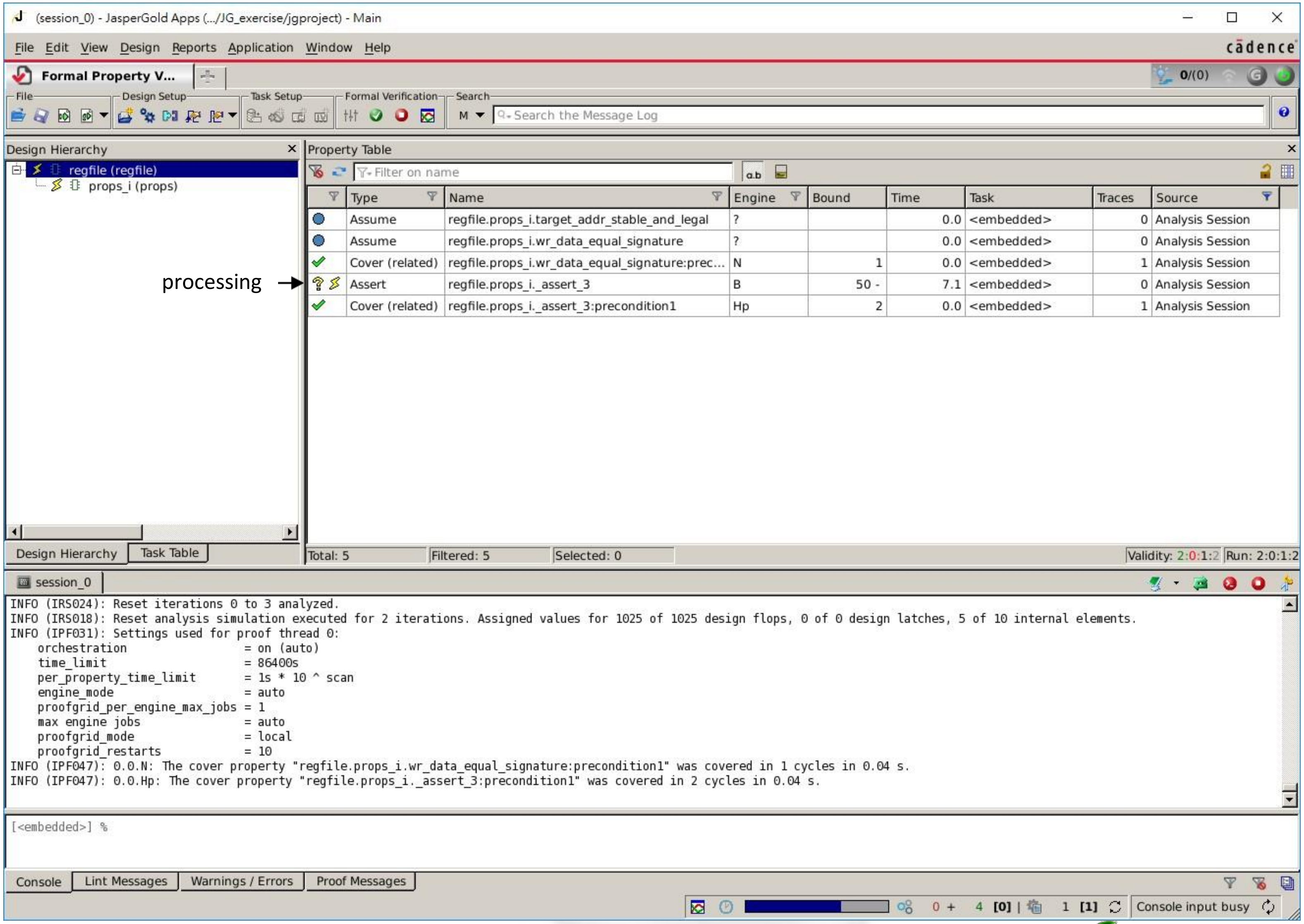
	Type	Name	Engine	Bound	Time	Task	Traces	Source
●	Assume	regfile.props_i.target_addr_stable_and_legal	?		0.0	<embedded>	0	Analysis Session
●	Assume	regfile.props_i.wr_data_equal_signature	?		0.0	<embedded>	0	Analysis Session
✓	Cover (related)	regfile.props_i.wr_data_equal_signature:prec...	N	1	0.0	<embedded>	1	Analysis Session
✓	Assert	regfile.props_i._assert_3	N (58)	Infinite	8.7	<embedded>	0	Analysis Session
✓	Cover (related)	regfile.props_i._assert_3:precondition1	Hp	2	0.0	<embedded>	1	Analysis Session

Total: 5 Filtered: 5 Selected: 0 Validity: 3:0:0:2 Run: 2:0:0:3

```
                  - undetermined              : 0 (0%)
                  - unknown                   : 0 (0%)
                  - error                     : 0 (0%)
                covers                        : 2
                  - unreachable               : 0 (0%)
                  - bounded_unreachable (user): 0 (0%)
                  - covered                   : 2 (100%)
                  - ar_covered                : 0 (0%)
                  - undetermined              : 0 (0%)
                  - unknown                   : 0 (0%)
                  - error                     : 0 (0%)
determined
determined
```

```
[<embedded>] %
```

Prove all again



processing →

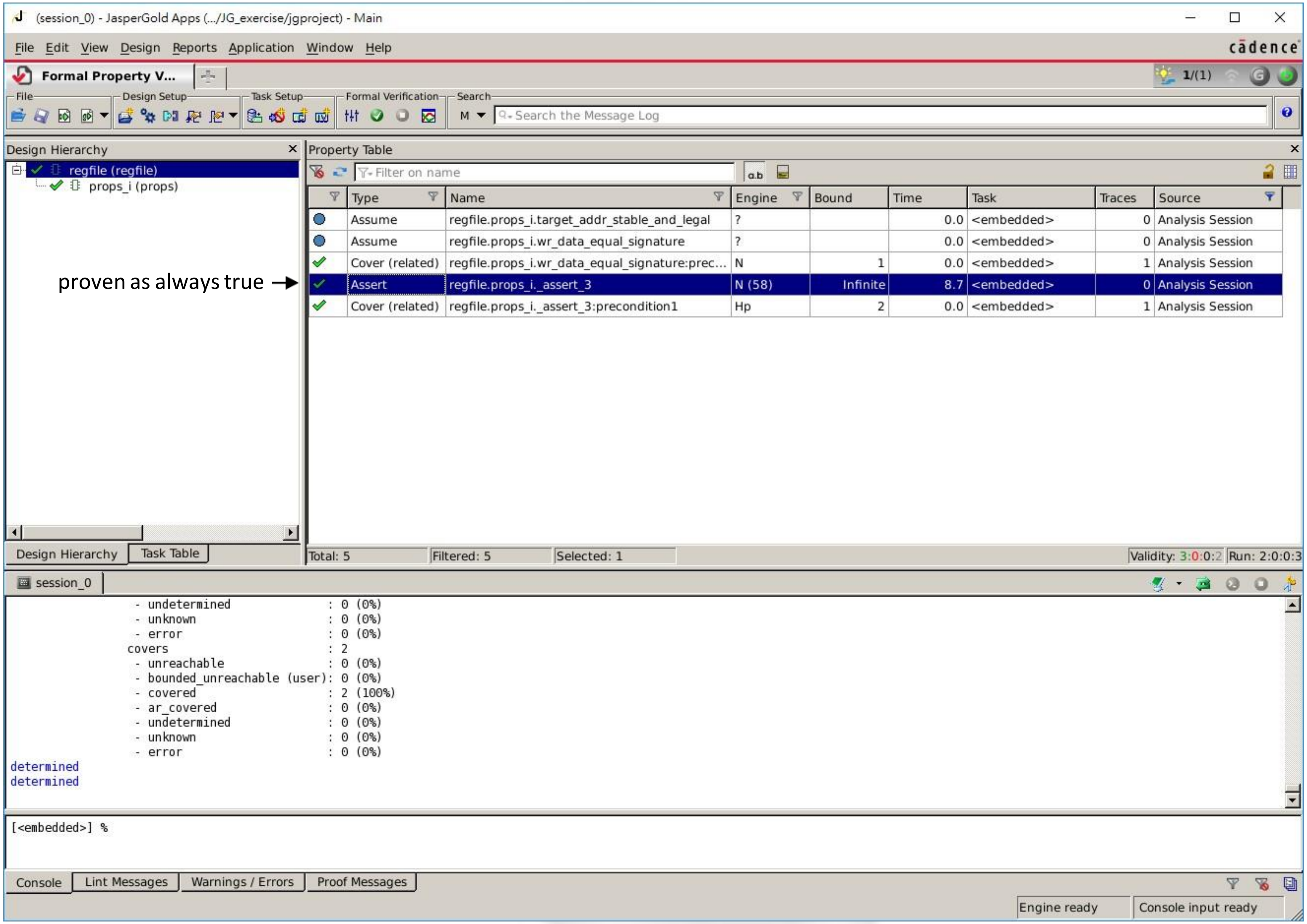
Property Table

	Type	Name	Engine	Bound	Time	Task	Traces	Source
●	Assume	regfile.props_i.target_addr_stable_and_legal	?		0.0	<embedded>	0	Analysis Session
●	Assume	regfile.props_i.wr_data_equal_signature	?		0.0	<embedded>	0	Analysis Session
✓	Cover (related)	regfile.props_i.wr_data_equal_signature:prec...	N	1	0.0	<embedded>	1	Analysis Session
? ⚡	Assert	regfile.props_i._assert_3	B	50 -	7.1	<embedded>	0	Analysis Session
✓	Cover (related)	regfile.props_i._assert_3:precondition1	Hp	2	0.0	<embedded>	1	Analysis Session

```
INFO (IRS024): Reset iterations 0 to 3 analyzed.
INFO (IRS018): Reset analysis simulation executed for 2 iterations. Assigned values for 1025 of 1025 design flops, 0 of 0 design latches, 5 of 10 internal elements.
INFO (IPF031): Settings used for proof thread 0:
    orchestration              = on (auto)
    time_limit                 = 86400s
    per_property_time_limit    = 1s * 10 ^ scan
    engine_mode                = auto
    proofgrid_per_engine_max_jobs = 1
    max engine jobs            = auto
    proofgrid_mode             = local
    proofgrid_restarts         = 10
INFO (IPF047): 0.0.N: The cover property "regfile.props_i.wr_data_equal_signature:precondition1" was covered in 1 cycles in 0.04 s.
INFO (IPF047): 0.0.Hp: The cover property "regfile.props_i._assert_3:precondition1" was covered in 2 cycles in 0.04 s.
```

```
[<embedded>] %
```

Full prove



proven as always true →

	Type	Name	Engine	Bound	Time	Task	Traces	Source
●	Assume	regfile.props_i.target_addr_stable_and_legal	?		0.0	<embedded>	0	Analysis Session
●	Assume	regfile.props_i.wr_data_equal_signature	?		0.0	<embedded>	0	Analysis Session
✓	Cover (related)	regfile.props_i.wr_data_equal_signature:prec...	N	1	0.0	<embedded>	1	Analysis Session
✓	Assert	regfile.props_i._assert_3	N (58)	Infinite	8.7	<embedded>	0	Analysis Session
✓	Cover (related)	regfile.props_i._assert_3:precondition1	Hp	2	0.0	<embedded>	1	Analysis Session

Total: 5 Filtered: 5 Selected: 1 Validity: 3:0:0:2 Run: 2:0:0:3

```
                - undetermined               : 0 (0%)
                - unknown                    : 0 (0%)
                - error                      : 0 (0%)
               covers                        : 2
                - unreachable                : 0 (0%)
                - bounded_unreachable (user): 0 (0%)
                - covered                    : 2 (100%)
                - ar_covered                 : 0 (0%)
                - undetermined               : 0 (0%)
                - unknown                    : 0 (0%)
                - error                      : 0 (0%)
determined
determined
```

```
[<embedded>] %
```

Engine ready Console input ready

Outline

□ Concept of Formal Verification

□ JasperGold Tool Guide

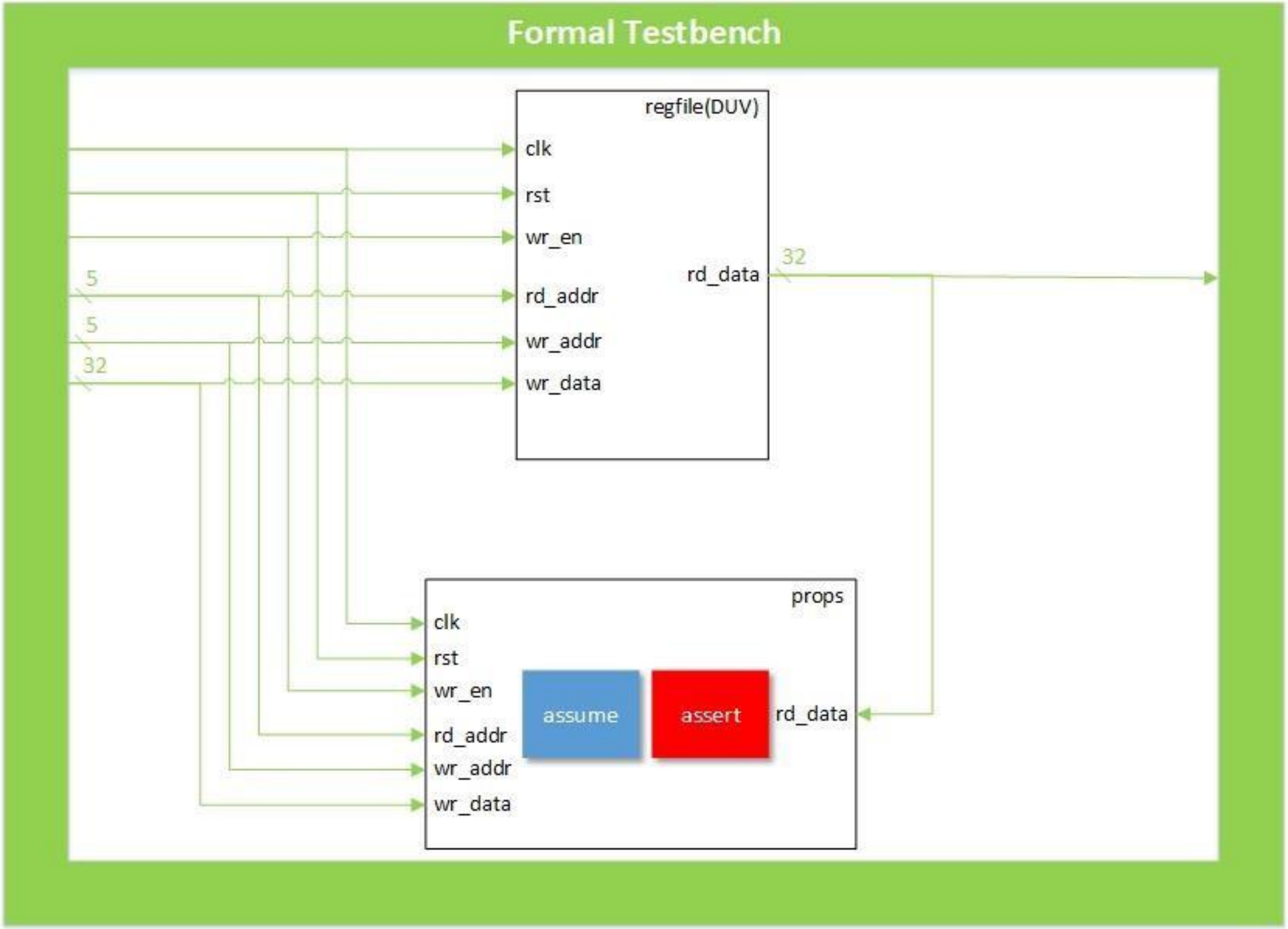
□ **Exercise**

Exercise (1/3)

- Complete property in SVA file to get full proof of Register File (design under verification, DUV)
- Step:
  - Download and *untar* JG_exercise.tar from Moodle
  - Follow steps in the previous slides(JasperGold tool guide) to verify the design
  - Modify props.sva of the DUV for “full proof”

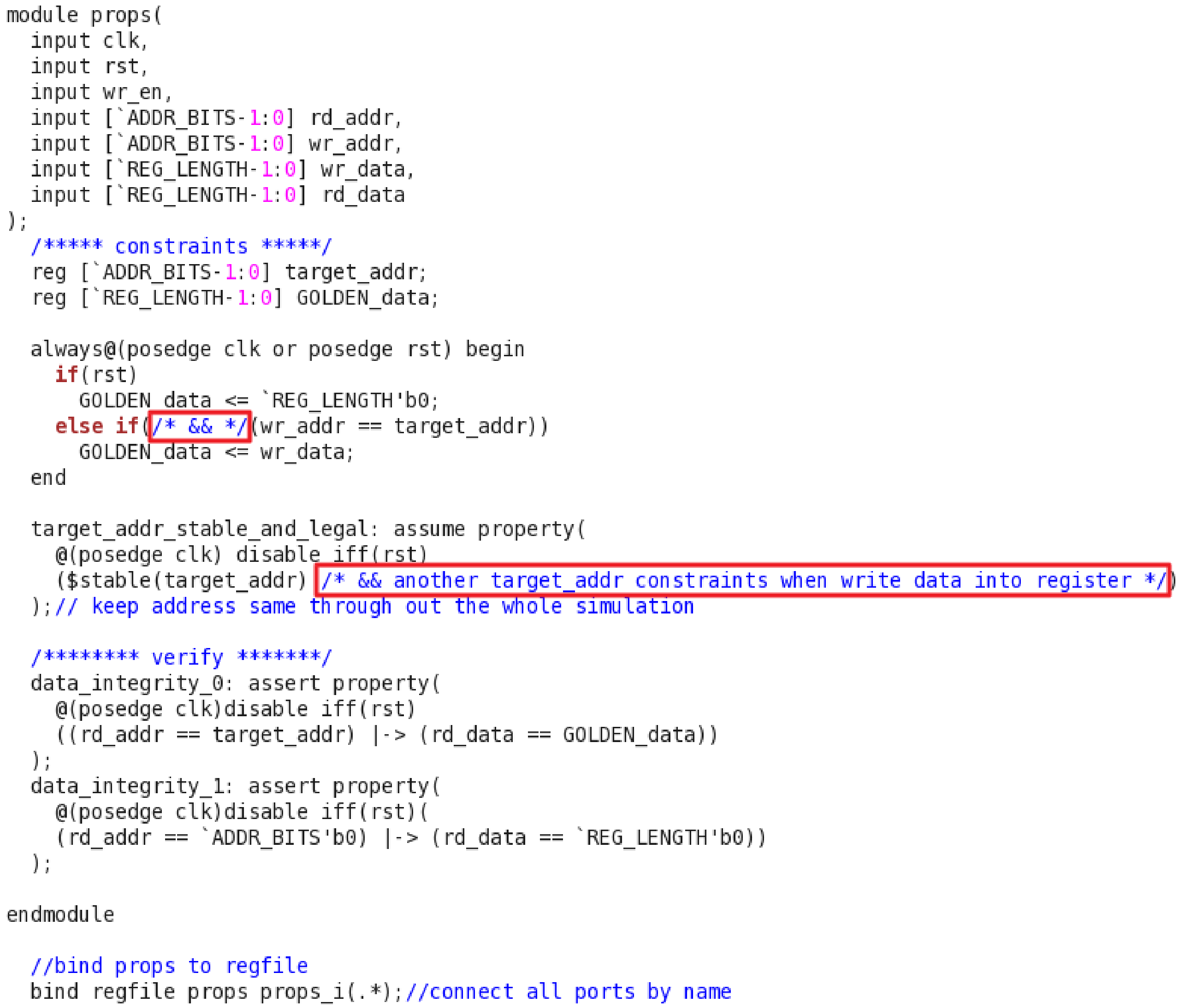
Exercise (2/3)

- System structure



Note: mem[0] of “regfile” is hardwire to zero

Exercise (3/3)



```
module props(
  input clk,
  input rst,
  input wr_en,
  input [`ADDR_BITS-1:0] rd_addr,
  input [`ADDR_BITS-1:0] wr_addr,
  input [`REG_LENGTH-1:0] wr_data,
  input [`REG_LENGTH-1:0] rd_data
);
  /***** constraints *****/
  reg [`ADDR_BITS-1:0] target_addr;
  reg [`REG_LENGTH-1:0] GOLDEN_data;

  always@(posedge clk or posedge rst) begin
    if(rst)
      GOLDEN_data <= `REG_LENGTH'b0;
    else if(/* && */(wr_addr == target_addr))
      GOLDEN_data <= wr_data;
  end

  target_addr_stable_and_legal: assume property(
    @(posedge clk) disable iff(rst)
    ($stable(target_addr) /* && another target_addr constraints when write data into register */)
  );// keep address same through out the whole simulation

  /******** verify *******/
  data_integrity_0: assert property(
    @(posedge clk)disable iff(rst)
    ((rd_addr == target_addr) |-> (rd_data == GOLDEN_data))
  );
  data_integrity_1: assert property(
    @(posedge clk)disable iff(rst)(
    (rd_addr == `ADDR_BITS'b0) |-> (rd_data == `REG_LENGTH'b0))
  );

endmodule

  //bind props to regfile
  bind regfile props props_i(.*);//connect all ports by name
```

Thanks for your participation and attendance !

Reference

- □ JasperGold Formal Verification Platform (Apps)