

VLSI System Design (Graduate Level)

Fall 2020

HOMEWORK II

REPORT

Must do self-checking before submission:

- ☐ Compress all files described in the problem into one tar
- ☐ All SystemVerilog files can be compiled under SoC Lab environment
- ☐ All port declarations comply with I/O port specifications
- ☐ Organize files according to File Hierarchy Requirement
- ☐ No any waveform files in deliverables

Student name: 周昱佑 杜冠勳

Student ID: N26090180 N26094883

● Summary :

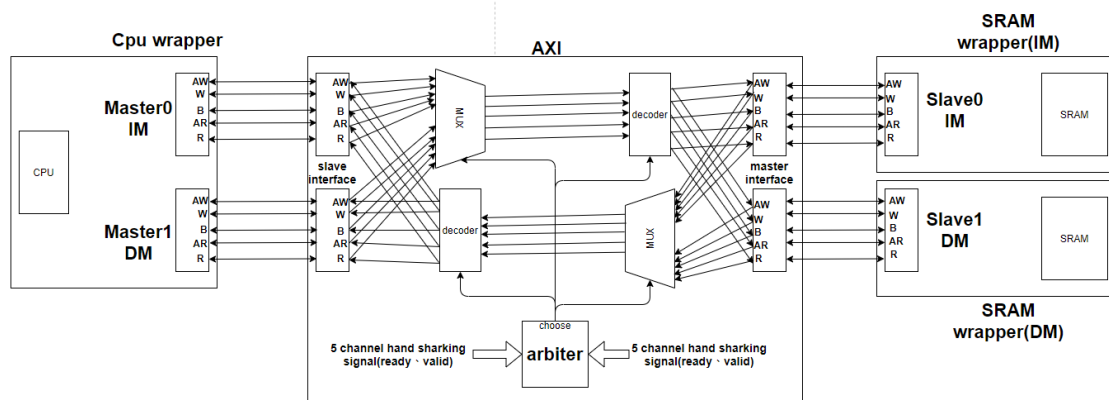
1.	AXI.sv verification(jaspergold)	DONE
2.	CPU_wrapper.sv verification(jaspergold)	DONE
3.	SRAM_wrapper.sv verification(jaspergold)	DONE
3.1	IM=slave1(ID=0)	DONE
3.2	DM=slave2(ID=1)	DONE
4.	Outstanding	1 (In order)
5.	Burst operation	Single transfer only
6.	Transfer type:	No burst transfer
7.	Synthesis	DONE
7.1	Prog0 before and after Synthesis	PASS
7.2	Prog1 before and after Synthesis	PASS
7.3	Prog2 before and after Synthesis	PASS
7.4	Prog3 before and after Synthesis	PASS
8.	superlint	99.24%

● 分工：

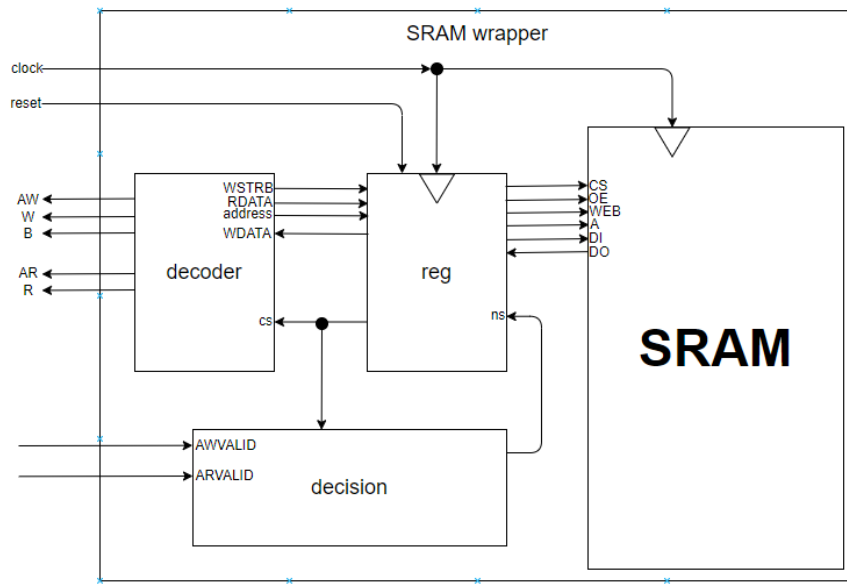
姓名	周昱佑 (50%)	杜冠勳 (50%)
負責工作	AXI SRAM Wrapper	AXI CPU Wrapper

● Architecture :

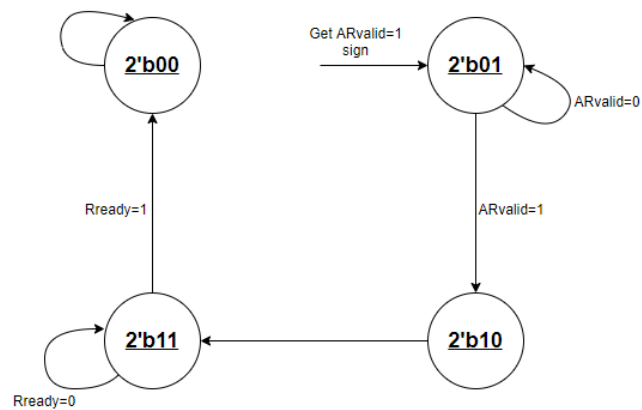
此次 AXI 的設計架構以狀態機來做訊號線的控制與讀存資料，以這樣設計的原因為較易於做判定是否要維持訊號與 cpu 的 stall，同時在 debug 能較清楚的觀察到是哪個狀態的判定上出了問題，狀態的切換依照 spec 上描述，以 valid 與 ready 為主，在 slave 端會驗證 ID 與 ADDRESS 的正確性給予 RESP 為 OK 或 DECERR，中間 AXI 的部分以 arbiter 來決定優先執行的部分，設計上以 master_dm 優先於 master_im，write 優先於 read。



➤ Slave :



1. Read FSM



hand_shaking_signal

2'b00 : finish (default state)

ARREADY=1'b0

RLAST=1'b0

RVALID=1'b0

2'b01 : AR (get ID or address)

ARREADY=1'b1

2'b10 : access memory (read "Data" from memory)

2'b11 : R (return respon of write)

RRESP=((ID is self_ID)&&(addr is in range))?2'b00:2'b11

RID=ID

RLAST=1'b1

RVALID=1'b1

memory_data

2'b00 : keep signal

2'b01 : get "address" "ID" from master

addr<=(read or write)?ARADDR:AWADDR

ID<=(read or write)?ARID:AWID

2'b10 : memory control and "write" data to memory

CS<=1'b1

OE<=(read or not)?1'b1:1'b0

WEB<=(write or not)?WSTRB:4'b1111

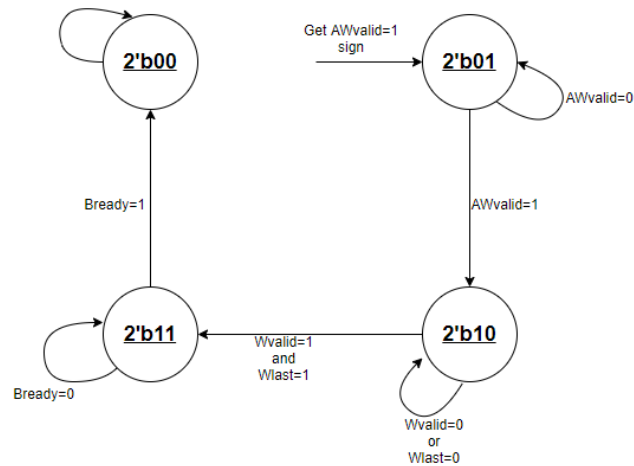
A<=addr[13:0]

DI<=(write valid)?WDATA:32'd0

2'b11 : get "read" data

RDATA=(read or not)?DO:RDATA

2. Write FSM



hand_shaking_signal

2'b00 : finish (default state)

AWREADY=1'b0
WREADY=1'b0
BVALID=1'b0;

2'b01 : AW (get ID or address)

AWREADY=1'b1

2'b10 : W (write "Data" to memory)

WREADY=1'b1

2'b11 : B (return respon of write)

BRESP=((ID is self_ID)&&(addr is in range))?2'b00:2'b11
BID=ID
BVALID=1'b1

memory_data

2'b00 : keep signal

2'b01 : get "address" "ID" from master

addr<=(read or write)?ARADDR:AWADDR
ID<=(read or write)?ARID:AWID

2'b10 : memory control and "write" data to memory

CS<=1'b1
OE<=(read or not)?1'b1:1'b0
WE<=(write or not)?WSTRB:4'b1111
A<=addr[13:0]

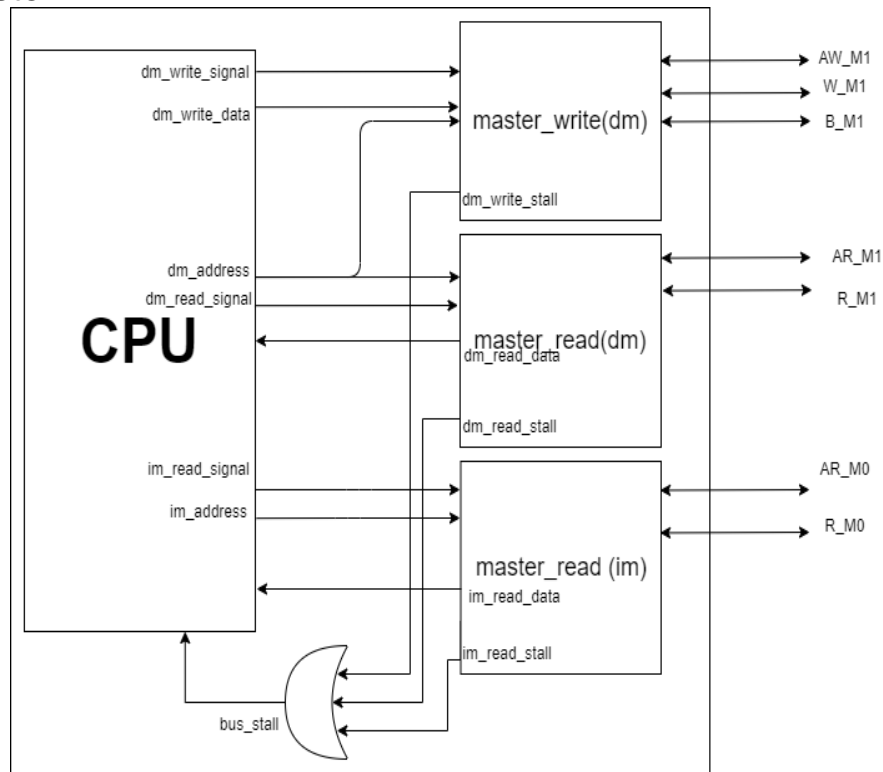
DI<=(write valid)?WDATA:32'd0

2'b11 : get "read" data

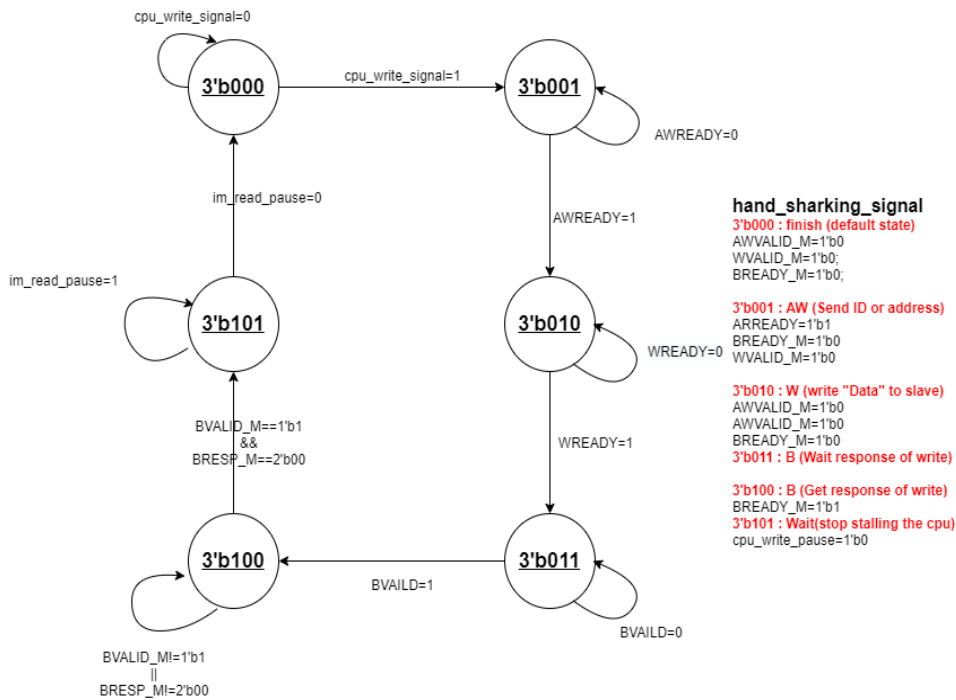
RDATA=(read or not)?DO:RDATA

在設計上，有 hand shaking 訊號才會進入下一個 state，抓取 master 端送過來的 data 則也是以達到 ready 與 valid 同時拉為 1 時才抓取，並存入 reg 中保持穩定，state 01 為 AW 或 AR 抓取 data，state 10 為 W 或 R 抓取 master 送過來要寫入 memory 的 data(LW、LB)與訪問 memory 取得正確 address 上的 data(SW、SB)，當為 read 時，則會在 state 11 時把 state 10 取得的 data 送出並拉起 RREADY 訊號，並回傳 RRESP，在 write 時，state 10 則會寫入 data，並直接進入 state 11 傳送 BVALID 訊號與 BRESP。

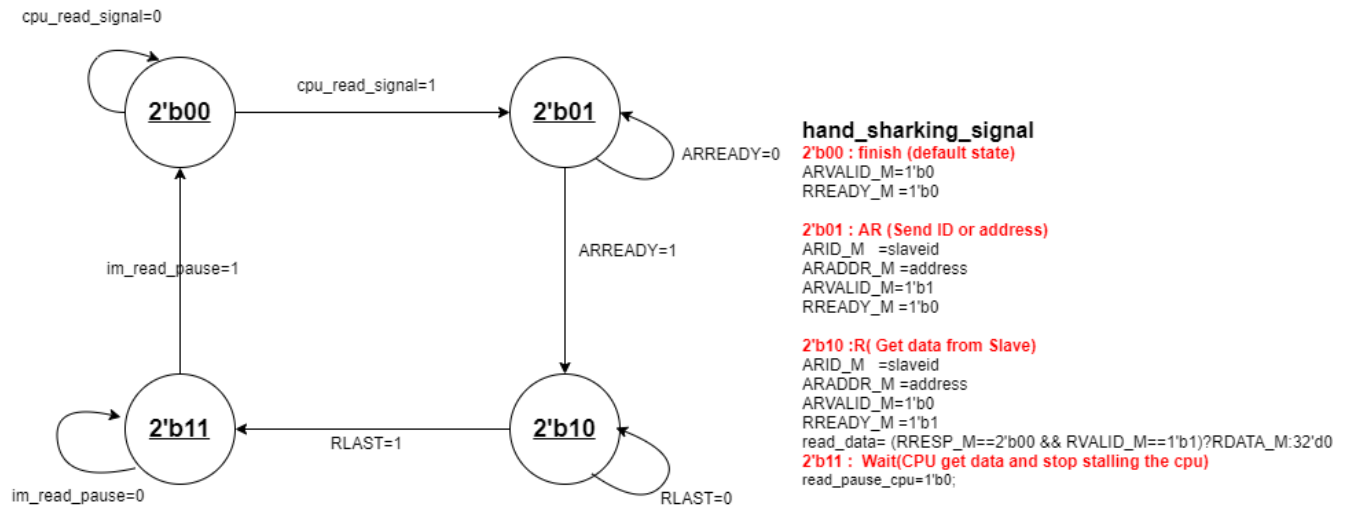
➤ Master:



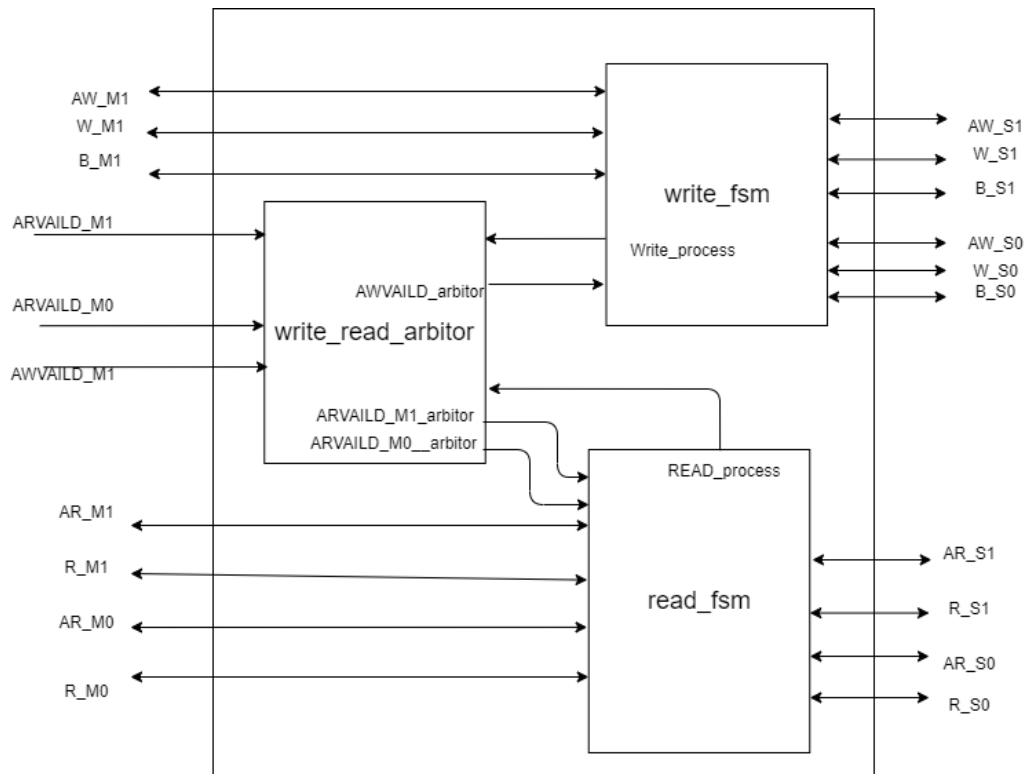
1. Master_write



2. Master_read



➤ AXI:



1. write_read_arbitor

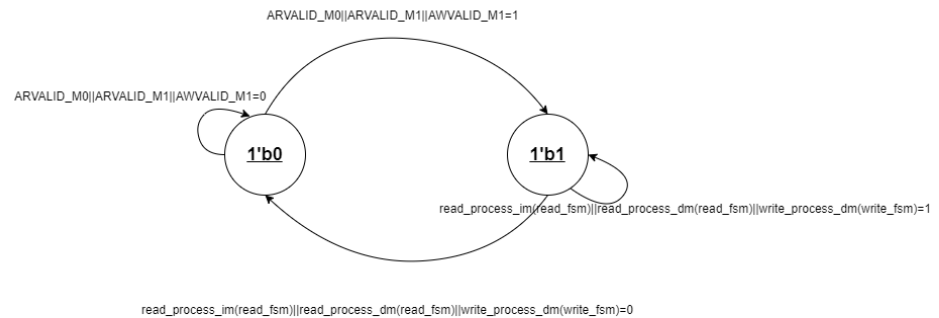
Arbitor_control_signal

1'b0 : Arbitration

```

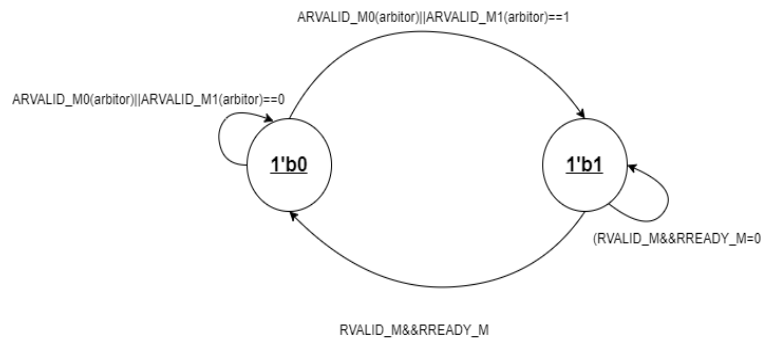
situation=(ARVALID_M0,
ARVALID_M1,
AWVALID_M1
);
case(situation)
3'b000:
begin
ARVALID_M_0stage1=1'b0;
ARVALID_M_1stage1=1'b0;
AWVALID_M_1stage1=1'b0;
end
3'b001:
begin
ARVALID_M_0stage1=1'b0;
ARVALID_M_1stage1=1'b0;
AWVALID_M_1stage1=AWVALID_M1;
end
.
.
.
.
.

```



1'b1 : Transfer Processing
keep the signal

2. read_fsm



Read_control_signal

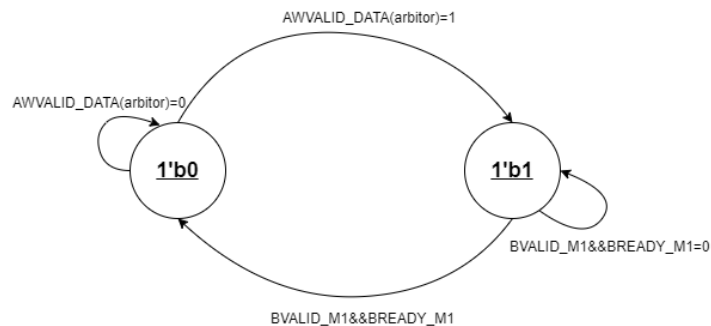
1'b0 : Initializing

signal all set to 0

1'b1 : Transfer Processing

slave_sel=(situation_decode=2'b01)?ARADDR_M1[31:16]:ARADDR_M0[31:0]
Master connect the signal with corresponding slave

3. write_fsm



Write_control_signal

1'b0 : Initializing

100 : Initializing
signal all set to 0

1'b1 : Transfer Processing

```
slave_select=AWADDR_M1[31:16]
```

Master connect the signal with corresponding slave

● Major problem & resolutions :

➤ Slave :

1. RVALID、ARREADY、AWREADY等訊號線為unstable?
<sol.>等待master端送出相對應的hand shaking訊號才進入下一個狀態。
2. RID、RDATA、BID等資料為unstable?
<sol.>在轉換與接收data的state時才做改變並將資料栓鎖進reg中，其他時候則輸出栓鎖的數值。
3. 輸入訊號線出現亂跳的現象，port與線的數值不相同?
<sol.>檢查top.sv的接線，發現有誤，下次接完後要再細心的檢查。
4. 當訊號線讀跟寫同時送入slave端?
<sol.>多設定一個flag訊號來做優先度的判定，以write為優先再來才是read，達到防止狀態機亂跳狀態而送錯訊號。
5. Read data時，資料沒有在cpu stall時回傳，反而多延後一個stall區間才收到?
<sol.>當memory在讀出資料時，address送入為1個clock，送出資料(RDATA)為1個clock，所以在狀態機state10時，多停留1個clock讓資料被讀出並在state11時將資料送出。
6. 收到的address在某些時候讀錯data?
<sol.>由於我們原先處理address為cpu做完除以4後才傳出給slave，但是在做prog0時發現master_dm會去讀slave_im的數值，造成我們事先處理的地址會使AXI的abiter開錯通道，使資料讀錯並回傳，再改成由slave端統一處理時已經解決。

➤ AXI :

1. Output訊號unstable(aw、arvalid與aw、arready)?
<sol.>由於在沒有選擇到要開啟通道的slave與master，jaspergold仍然會給予訊號，所以需要將其輸出的訊號做栓鎖，若有出現hand shaking的時候才能改變其訊號。

2. valid與ready訊號的先後順序？

<sol.>測試 jaspergold時，valid與ready訊號並沒有指定先後，若valid訊號先送出，則要保持與其一同送出的資料，所以加上一個flip-flop與狀態機來做資料的栓鎖。

3. AR、R與AW、W的先後順序問題？

<sol.>由於AW和W可以同時送出，但是AR與R有先後送出的問題，由於我們在設計通道上，但是jaspergold會提前拉起B與R的hand shaking訊號，如此，則會出現通道過早關閉，所以將read與write的hand shaking訊號放入arbiter內來做通道的開關判定。

4. Decoder條件沒有寫滿？

<sol.>造成latch產生與訊號線的值可能亂跳，在問題1.有提到的栓鎖問題有解決但是仍要把沒用到的線補上，防止產生latch。

● Results of verification :

➤ Max pending number : 1

✧ Jaspergold for AXI 4.0 protocol :

1. Master wrapper :

The screenshot shows the Cadence JasperGold verification interface. The top panel displays the Design Hierarchy with the Master wrapper selected. The middle panel shows the Property Table with various properties and their values. The bottom panel displays the Summary of the verification results.

Type	Engine	Bound	Time	Task	Traces	Source
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	2	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	2	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	B	5	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	B	5	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	2	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	2	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	2	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	2	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	B	5	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	2	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	2	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	B	5	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	1	0.0	<embedded>	1 Analysis Session
Cover (related)	top_axi_monitor_0.genStableChks.genStable...	N	1	0.0	<embedded>	1 Analysis Session

Summary:

Properties Considered	Count
assertions	230
- proven	98 (34 disabled)
- bounded_proven (user)	65 (65 disabled)
- bounded_proven (auto)	0 (0%)
- axi4_proven	0 (0%)
- cex	0 (0%)
- ar_cex	0 (0%)
- undetermined	0 (0%)
- unknown	34 (34 disabled)
- error	0 (0%)
covers	121 (41 disabled)
- unreachable	0 (0%)
- bounded_unreachable (user)	0 (0%)
- covered	98 (68.023%)
- ar_covered	0 (0%)
- undetermined	0 (0%)
- unknown	41 (41 disabled)
- error	0 (0%)

Validity: 155.0 75.0 Run: 118.0 0.155

2. AXI :

The screenshot displays the Cadence JasperGold Formal Verification tool interface for the 'jg_bridge.tcl' project. The Design Hierarchy on the left shows the 'axi_slave_bridge' component. The Property Table on the right lists 11 properties, all of which are 'Cover (related)' and 'Embedded'. The Summary section at the bottom provides a detailed breakdown of the verification results:

Properties Considered	Count	Percentage
assertions	105	
- proven	133 (47 disabled)	
- bounded_proven (user)	146 (75.6477%)	
- bounded_proven (auto)	0 (0%)	
- marked_proven	0 (0%)	
- cex	0 (0%)	
- ar_cex	0 (0%)	
- undetermined	0 (0%)	
- unknown	47 (47 disabled)	(24.3523%)
- error	0 (0%)	
covers	112 (56 disabled)	
- unreachable	0 (0%)	
- bounded_unreachable (user)	0 (0%)	
- covered	256 (82.0512%)	
- ar_covered	0 (0%)	
- undetermined	0 (0%)	
- unknown	56 (56 disabled)	(17.9487%)
- error	0 (0%)	

The console output shows the completion of the proof on task 'embedded'.

3. SRAM wrapper :

The screenshot displays the Cadence JasperGold Formal Verification tool interface for the 'jg_slave.tcl' project. The Design Hierarchy on the left shows the 'axi_slave' component. The Property Table on the right lists 11 properties, all of which are 'Cover (related)' and 'Embedded'. The Summary section at the bottom provides a detailed breakdown of the verification results:

Properties Considered	Count	Percentage
assertions	128	
- proven	130 (12.0812%)	
- bounded_proven (user)	0 (0%)	
- bounded_proven (auto)	0 (0%)	
- marked_proven	0 (0%)	
- cex	0 (0%)	
- ar_cex	0 (0%)	
- undetermined	7 (7 disabled)	(18.9189%)
- unknown	0 (0%)	
- error	0 (0%)	
covers	81 (11 disabled)	
- unreachable	0 (0%)	
- bounded_unreachable (user)	0 (0%)	
- covered	80 (87.9122%)	
- ar_covered	0 (0%)	
- undetermined	0 (0%)	
- unknown	10 (11 disabled)	(12.0879%)
- error	0 (0%)	

The console output shows the completion of the proof on task 'embedded'.

Superlint :

Total line of warning : 55

Total line of code : 7330

Total line of code : 7330

Total line of warning : 55 (47+7+1)

※Correct rate of code : $(7330-55)/7330=0.992496$ (99.24%)

修正錯誤與 warning:

1. Master 端的送出 hand shaking 訊號有 latch 產生，有 53 條 warning，補齊訊號線在 always_comb 後，已被修正。

2. 上圖所示的 47 條 warning，為 file format 上的問題，檔案的命名上為該 warning 出現的原因，由於檔案命名不影響，故按照其設計命名沒有做修正。

❖ Test for Prog0~3

1. Prog0

```
File Edit View Search Terminal Help
DM[ 10] = cccccccc, pass
DM[ 11] = fffffffc, pass
DM[ 12] = cccccccc, pass
DM[ 13] = fffffffc, pass
DM[ 14] = cccccccc, pass
DM[ 15] = 00000d9d, pass
DM[ 16] = 00000004, pass
DM[ 17] = 00000003, pass
DM[ 18] = 000001a6, pass
DM[ 19] = 00000ec6, pass
DM[ 20] = 2468b7a8, pass
DM[ 21] = 5dbf9f00, pass
DM[ 22] = 00012b38, pass
DM[ 23] = fa2817b7, pass
DM[ 24] = ff000000, pass
DM[ 25] = 12345678, pass
DM[ 26] = 0000f000, pass
DM[ 27] = 00000f00, pass
DM[ 28] = 000000f0, pass
DM[ 29] = 0000000f, pass
DM[ 30] = 12345678, pass
DM[ 31] = 78000000, pass
DM[ 32] = 12345678, pass
DM[ 33] = 00000078, pass
DM[ 34] = 12345678, pass
DM[ 35] = 8a345678, pass
DM[ 36] = fffff000, pass
DM[ 37] = fffff000, pass
DM[ 38] = fffff000, pass
DM[ 39] = fffff000, pass
DM[ 40] = fffff000, pass
DM[ 41] = fffff000, pass
DM[ 42] = 1357a070, pass
DM[ 43] = 13578000, pass
DM[ 44] = fffff004, pass

make rtl0

*****
**                                     **
** Congratulations !!                **
**                                     **
** Simulation PASS!!                 **
**                                     **
*****
                                     |__|
                                     / 0.0 |
                                     /-----\
                                     |^ ^ ^ ^ |w|
                                     \m__m__|_||
```

```
檔案(F) 編輯(E) 檢視(V) 搜尋(S) 終端機(T) 求助(H)
DM[ 0] = fffffff0, pass
DM[ 1] = fffffff8, pass
DM[ 2] = 00000008, pass
DM[ 3] = 00000001, pass
DM[ 4] = 00000001, pass
DM[ 5] = 78787878, pass
DM[ 6] = 000091a2, pass
DM[ 7] = 00000003, pass
DM[ 8] = fefcfefd, pass
DM[ 9] = 10305070, pass
DM[ 10] = cccccccc, pass
DM[ 11] = fffffffc, pass
DM[ 12] = cccccccc, pass
DM[ 13] = fffffffc, pass
DM[ 14] = cccccccc, pass
DM[ 15] = 00000d8d, pass
DM[ 16] = 00000004, pass
DM[ 17] = 00000003, pass
DM[ 18] = 000001a6, pass
DM[ 19] = 00000ec6, pass
DM[ 20] = 2468b7a8, pass
DM[ 21] = 5dbf9f00, pass
DM[ 22] = 00012b38, pass
DM[ 23] = fa2817b7, pass
DM[ 24] = ff000000, pass
DM[ 25] = 12345678, pass
DM[ 26] = 0000f000, pass
DM[ 27] = 00000f00, pass
DM[ 28] = 000000f0, pass
DM[ 29] = 0000000f, pass
DM[ 30] = 12345678, pass
DM[ 31] = 78000000, pass
DM[ 32] = 12345678, pass
DM[ 33] = 00000078, pass
DM[ 34] = 12345678, pass
DM[ 35] = 8a345678, pass
DM[ 36] = fffff000, pass
DM[ 37] = fffff000, pass
DM[ 38] = fffff000, pass
DM[ 39] = fffff000, pass
DM[ 40] = fffff000, pass
DM[ 41] = fffff000, pass
DM[ 42] = 1357a070, pass
DM[ 43] = 13578000, pass
DM[ 44] = fffff004, pass

make syn0

*****
**                                     **
** Congratulations !!                **
**                                     **
** Simulation PASS!!                 **
**                                     **
*****
                                     |__|
                                     / 0.0 |
                                     /-----\
                                     |^ ^ ^ ^ |w|
                                     \m__m__|_||
```

2. Prog1

```
DM[ 56] = 4db52726, pass
DM[ 57] = 514c9dfd, pass
DM[ 58] = 55507b33, pass
DM[ 59] = 55f54ad2, pass
DM[ 60] = 5d2b5714, pass
DM[ 61] = 5fc5592e, pass
DM[ 62] = 5fd797ec, pass
DM[ 63] = 75fef64b, pass
```

make rtl1

```
*****
**                                     **
**  Congratulations !!               **
**                                     **
**  Simulation PASS!!                **
**                                     **
*****
```



```
Simulation complete via $finish(1) at time 15204455 NS + 2
./sim/top_tb.sv:88      $finish;
ncsim> exit
[khduh@ncku N260XXXXX]$
```

Applications Places System

Terminal

File Edit View Search Terminal Help

```
DM[ 24] = e022a79, pass
DM[ 25] = e170f380, pass
DM[ 26] = e383c4f6, pass
DM[ 27] = e567e6fa, pass
DM[ 28] = e700b309, pass
DM[ 29] = e9272eff, pass
DM[ 30] = ec6ae05d, pass
DM[ 31] = ef2f22d0, pass
DM[ 32] = f2c1b005, pass
DM[ 33] = fe3bc3c8, pass
DM[ 34] = 00000000, pass
DM[ 35] = 00ec1335, pass
DM[ 36] = 0338c9ff, pass
DM[ 37] = 0436621b, pass
DM[ 38] = 06653805, pass
DM[ 39] = 13b18826, pass
DM[ 40] = 19833740, pass
DM[ 41] = 19b169b6, pass
DM[ 42] = 1f19454a, pass
DM[ 43] = 1ffeb1ab, pass
DM[ 44] = 24177e3f, pass
DM[ 45] = 261e160a, pass
DM[ 46] = 2c80844b, pass
DM[ 47] = 2c808ffb, pass
DM[ 48] = 2fa4086c, pass
DM[ 49] = 30ef160d, pass
DM[ 50] = 36eae26, pass
DM[ 51] = 37ff1409, pass
DM[ 52] = 3af5a03c, pass
DM[ 53] = 308248a2, pass
DM[ 54] = 3ac99409, pass
DM[ 55] = 44f80a53, pass
DM[ 56] = 4db52726, pass
DM[ 57] = 514c9dfd, pass
DM[ 58] = 55507b33, pass
DM[ 59] = 55f54ad2, pass
DM[ 60] = 5d2b5714, pass
DM[ 61] = 5fc5592e, pass
DM[ 62] = 5fd797ec, pass
DM[ 63] = 75fef64b, pass
```

```
**                                     **
**  Congratulations !!               **
**                                     **
**  Simulation PASS!!                **
**                                     **
*****
```

Simulation complete via \$finish(1) at time 26464210 NS + 1
./sim/top_tb.sv:88 \$finish;
ncsim> exit
vlsicad8:/home/user2/vsd20/vsd2055/N26094883

New Volume vsd2055 Terminal

make syn1

3. Prog2

```
DM[ 0] = 371ee447, pass
DM[ 1] = fffff739, pass
```

make rtl2

```
*****
**                                     **
**  Congratulations !!              **
**                                     **
**  Simulation PASS!!              **
**                                     **
*****
```



```

pullup_down_keep_10_03us
pullup_down_keep_10_16
pullup_down_keep_10_16
pullup_inv_10
top.t0
ncelab: *% DSEMC009: This SystemVerilog design will be simulated as per IEEE 1800-2009 SystemVerilog simulation semantics. Use -disable_sen2009 option for turning off SV 2009 simulation semantics.
ncelab: *% CUSINT: Illegal Keyword Type for argument 2. Argument Skipped.
initial $sf_emulate top_syn.sdf, "top_syn.v";
ncelab: *% CUSST2 (./sim/top.th.sv,81(22)): This SDF System Task will be ignored.
Building instance overlay tables: ..... Done
Building instance specific data structures.
Loading native compiled code: ..... Done
Design hierarchy summary:
Instances Unique
Modules: 10896 444
BOPs: 2572 11
Primitives: 21015 14
Timing outputs: 11197 294
Registers: 2443 365
Scalar wires: 15415 -
Expanded wires: 299 21
Named events: 47 42
Always blocks: 125 112
Initial blocks: 8 7
Cont. assignments: 43 356
Timing checks: 16211 1869
Delayed tcheck signals: 544 1943
Simulation timescale: 1ps
Writing initial simulation snapshot: worklib.ANCBI: Lib
Loading snapshot worklib.ANCBI: Lib ..... Done
ncsim: *% DSEMC009: This SystemVerilog design is simulated as per IEEE 1800-2009 SystemVerilog simulation semantics. Use -disable_sen2009 option for turning off SV 2009 simulation semantics.
Verilog Loading libscope: bus162_50
ncsim: source /usr/cad/cadence/INCISIV/cur/tools/inca/files/ncsimrc
ncsim: run
Done
DM[ 0] = 371ee447, pass
DM[ 1] = fffff739, pass

*****
**                                     **
**  Congratulations !!              **
**                                     **
**  Simulation PASS!!              **
**                                     **
*****

Simulation complete via $finish() at time 1455330 NS + 1
./sim/top.th.sv:87 $finish
ncsim: exit
visicad8: /home/user2/vsd20/vsd2055/ND6094883 X
```

make syn2

4. Prog3

```
DM[ 0] = 00000003, pass
```

make rtl3

```
*****
**                                     **
**  Congratulations !!               **
**                                     **
**  Simulation PASS!!                **
**                                     **
*****
```



```
Simulation complete via $finish(1) at time 1158105 NS + 2
../sim/top.tb:sv:87 $finish;
ncsim> exit
```

```

  檔案(F) 編輯(E) 功能(V) 搜索(S) 結構視圖(T) 波數(W)
  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1041 1042 1043 1044 1045 1046 1047 1048 1049 1050 1051 1052 1053 1054 1055 1056 1057 1058 1059 1060 1061 1062 1063 1064 1065 1066 1067 1068 1069 1070 1071 1072 1073 1074 1075 1076 1077 1078 1079 1080 1081 1082 1083 1084 1085 1086 1087 1088 1089 1090 1091 1092 1093 1094 1095 1096 1097 1098 1099 1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 1110 1111 1112 1113 1114 1115 1116 1117 1118 1119 1120 1121 1122 1123 1124 1125 1126 1127 1128 1129 1130 1131 1132 1133 1134 1135 1136 1137 1138 1139 1140 1141 1142 1143 1144 1145 1146 1147 1148 1149 1150 1151 1152 1153 1154 1155 1156 1157 1158 1159 1160 1161 1162 1163 1164 1165 1166 1167 1168 1169 1170 1171 1172 1173 1174 1175 1176 1177 1178 1179 1180 1181 1182 1183 1184 1185 1186 1187 1188 1189 1190 1191 1192 1193 1194 1195 1196 1197 1198 1199 1200 1201 1202 1203 1204 1205 1206 1207 1208 1209 1210 1211 1212 1213 1214 1215 1216 1217 1218 1219 1220 1221 1222 1223 1224 1225 1226 1227 1228 1229 1230 1231 1232 1233 1234 1235 1236 1237 1238 1239 1240 1241 1242 1243 1244 1245 1246 1247 1248 1249 1250 1251 1252 1253 1254 1255 1256 1257 1258 1259 1260 1261 1262 1263 1264 1265 1266 1267 1268 1269 1270 1271 1272 1273 1274 1275 1276 1277 1278 1279 1280 1281 1282 1283 1284 1285 1286 1287 1288 1289 1290 1291 1292 1293 1294 1295 1296 1297 1298 1299 1300 1301 1302 1303 1304 1305 1306 1307 1308 1309 1310 1311 1312 1313 1314 1315 1316 1317 1318 1319 1320 1321 1322 1323 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 1334 1335 1336 1337 1338 1339 1340 1341 1342 1343 1344 1345 1346 1347 1348 1349 1350 1351 1352 1353 1354 1355 1356 1357 1358 1359 1360 1361 1362 1363 1364 1365 1366 1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403 1404 1405 1406 1407 1408 1409 1410 1411 1412 1413 1414 1415 1416 1417 1418 1419 1420 1421 1422 1423 1424 1425 1426 1427 1428 1429 1430 1431 1432 1433 1434 1435 1436 1437 1438 1439 1440 1441 1442 1443 1444 1445 1446 1447 1448 1449 1450 1451 1452 1453 1454 1455 1456 1457 1458 1459 1460 1461 1462 1463 1464 1465 1466 1467 1468 1469 1470 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 1481 1482 1483 1484 1485 1486 1487 1488 1489 1490 1491 1492 1493 1494 1495 1496 1497 1498 1499 1500 1501 1502 1503 1504 1505 1506 1507 1508 1509 1510 1511 1512 1513 1514 1515 1516 1517 1518 1519 1520 1521 1522 1523 1524 1525 1526 1527 1528 1529 1530 1531 1532 1533 1534 1535 1536 1537 1538 1539 1540 1541 1542 1543 1544 1545 1546 1547 1548 1549 1550 1551 1552 1553 1554 1555 1556 1557 1558 1559 1560 1561 1562 1563 1564 1565 1566 1567 1568 1569 1570 1571 1572 1573 1574 1575 1576 1577 1578 1579 1580 1581 1582 1583 1584 1585 1586 1587 1588 1589 1590 1591 1592 1593 1594 1595 1596 1597 1598 1599 1600 1601 1602 1603 1604 1605 1606 1607 1608 1609 1610 1611 1612 1613 1614 1615 1616 1617 1618 1619 1620 1621 1622 1623 1624 1625 1626 1627 1628 1629 1630 1631 1632 1633 1634 1635 1636 1637 1638 1639 1640 1641 1642 1643 1644 1645 1646 1647 1648 1649 1650 1651 1652 1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 1674 1675 1676 1677 1678 1679 1680 1681 1682 1683 1684 1685 1686 1687 1688 1689 1690 1691 1692 1693 1694 1695 1696 1697 1698 1699 1700 1701 1702 1703 1704 1705 1706 1707 1708 1709 1710 1711 1712 1713 1714 1715 1716 1717 1718 1719 1720 1721 1722 1723 1724 1725 1726 1727 1728 1729 1730 1731 1732 1733 1734 1735 1736 1737 1738 1739 1740 1741 1742 1743 1744 1745 1746 1747 1748 1749 1750 1751 1752 1753 1754 1755 1756 1757 1758 1759 1760 1761 1762 1763 1764 1765 1766 1767 1768 1769 1770 1771 1772 1773 1774 1775 1776 1777 1778 1779 1780 1781 1782 1783 1784 1785 1786 1787 1788 1789 1790 1791 1792 1793 1794 1795 1796 1797 1798 1799 1800 1801 1802 1803 1804 1805 1806 1807 1808 1809 1810 1811 1812 1813 1814 1815 1816 1817 1818 1819 1820 1821 1822 1823 1824 1825 1826 1827 1828 1829 1830 1831 1832 1833 1834 1835 1836 1837 1838 1839 1840 1841 1842 1843 1844 1845 1846 1847 1848 1849 1850 1851 1852 1853 1854 1855 1856 1857 1858 1859 1860 1861 1862 1863 1864 1865 1866 1867 1868 1869 1870 1871 1872 1873 1874 1875 1876 1877 1878 1879 1880 1881 1882 1883 1884 1885 1886 1887 1888 1889 1890 1891 1892 1893 1894 1895 1896 1897 1898 1899 1900 1901 1902 1903 1904 1905 1906 1907 1908 1909 1910 1911 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921 1922 1923 1924 1925 1926 1927 1928 1929 1930 1931 1932 1933 1934 1935 1936 1937 1938 1939 1940 1941 1942 1943 1944 1945 1946 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023 2024 2025 2026 2027 2028 2029 2030 2031 2032 2033 2034 2035 2036 2037 2038 2039 2040 2041 2042 2043 2044 2045 2046 2047 2048 2049 2050 2051 2052 2053 2054 2055 2056 2057 2058 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 2069 2070 2071 2072 2073 2074 2075 2076 2077 2078 2079 2080 2081 2082 2083 2084 2085 2086 2087 2088 2089 2090 2091 2092 2093 2094 2095 2096 2097 2098 2099 2100 2101 2102 2103 2104 2105 2106 2107 2108 2109 2110 2111 2112 2113 2114 2115 2116 2117 2118 2119 2120 2121 2122 2123 2124 2125 2126 2127 2128 2129 2130 2131 2132 2133 2134 2135 2136 2137 2138 2139 2140 2141 2142 2143 2144 2145 2146 2147 2148 2149 2150 2151 2152 2153 2154 2155 2156 2157 2158 2159 2160 2161 2162 2163 2164 2165 2166 2167 2168 2169 2170 2171 2172 2173 2174 2175 2176 2177 2178 2179 2180 2181 2182 2183 2184 2185 2186 2187 2188 2189 2190 2191 2192 2193 2194 2195 2196 2197 2198 2199 2200 2201 2202 2203 2204 2205 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 2216 2217 2218 2219 2220 2221 2222 2223 2224 2225 2226 2227 2228 2229 2230 2231 2232 2233 2234 2235 2236 2237 2238 2239 2240 2241 2242 2243 2244 2245 2246 2247 2248 2249 2250 2251 2252 2253 2254 2255 2256 2257 2258 2259 2260 2261 2262 2263 2264 2265 2266 2267 2268 2269 2270 2271 2272 2273 2274 2275 2276 2277 2278 2279 2280 2281 2282 2283 2284 2285 2286 2287 2288 2289 2290 2291 2292 2293 2294 2295 2296 2297 2298 2299 2300 2301 2302 2303 2304 2305 2306 2307 2308 2309 2310 2311 2312 2313 2314 2315 2316 2317 2318 2319 2320 2321 2322 2323 2324 2325 2326 2327 2328 2329 2330 2331 2332 2333 2334 2335 2336 2337 2338 2339 2340 2341 2342 2343 2344 2345 2346 2347 2348 2349 2350 2351 2352 2353 2354 2355 2356 2357 2358 2359 2360 2361 2362 2363 2364 2365 2366 2367 2368 2369 2370 2371 2372 2373 2374 2375 2376 2377 2378 2379 2380 2381 2382 2383 2384 2385 2386 2387 2388 2389 2390 2391 2392 2393 2394 2395 2396 2397 2398 2399 2400 2401 2402 2403 2404 2405 2406 2407 2408 2409 2410 2411 2412 2413 2414 2415 2416 2417 2418 2419 2420 2421 2422 2423 2424 2425 2426 2427 2428 2429 2430 2431 2432 2433 2434 2435 2436 2437 2438 2439 2440 2441 2442 2443 2444 2445 2446 2447 2448 2449 2450 2451 2452 2453 2454 2455 2456 2457 2458 2459 2460 2461 2462 2463 2464 2465 2466 2467 2468 2469 2470 2471 2472 2473 2474 2475 2476 2477 2478 2479 2480 2481 2482 2483 2484 2485 2486 2487 2488 2489 2490 2491 2492 2493 2494 2495 2496 2497 2498 2499 2500 2501 2502 2503 2504 2505 2506 2507 2508 2509 2510 2511 2512 2513 2514 2515 2516 2517 2518 2519 2520 2521 2522 2523 2524 2525 2526 2527 2528 2529 2530 2531 2532 2533 2534 2535 2536 2537 2538 2539 2540 2541 2542 2543 2544 2545 2546 2547 2548 2549 2550 2551 2552 2553 2554 2555 2556 2557 2558 2559 2560 2561 2562 2563 2564 2565 2566 2567 2568 2569 2570 2571 2572 2573 2574 2575 2576 2577 2578 2579 2580 2581 2582 2583 2584 2585 2586 2587 2588 2589 2590 2591 2592 2593 2594 2595 2596 2597 2598 2599 2600 2601 2602 2603 2604 2605 2606 2607 2608 2609 2610 2611 2612
```


✧ Synthesis :

合成上我們使用的 clock cycle 為 20ns，結果如下附圖

*****				cpu/CPUI/ais/U28/O (BUFICK)	0.21	11.52 r
Report : timing				cpu/CPUI/ais/U55/O (A0222)	0.19	11.71 r
-path full				cpu/CPUI/ais/src2_data[1] (alu_in_selector)	0.00	11.71 r
-delay max				cpu/CPUI/U140/O (BUFICK)	0.10	11.81 r
-max_paths 1				cpu/CPUI/ard/src2[1] (alu_rd)	0.00	11.81 r
-sort_by group				cpu/CPUI/ard/U530/O (INVIS)	0.17	11.99 f
Design : top				cpu/CPUI/ard/U529/O (BUFICK)	0.15	12.13 f
Version: 0-2018.06				cpu/CPUI/ard/U159/O (INV2)	0.41	12.55 r
Date : Tue Nov 10 21:23:54 2020				cpu/CPUI/ard/sub_38/B[1] (alu_rd_DW01_sub_0)	0.00	12.55 r
*****				cpu/CPUI/ard/sub_38/U4/O (INVIS)	0.10	12.65 f
# A fanout number of 1000 was used for high fanout net computations.				cpu/CPUI/ard/sub_38/U2_1/CO (FAIS)	0.25	12.90 f
Operating Conditions: BCCOM Library: fsa0m_a_generic_core_fflp98vm40c				cpu/CPUI/ard/sub_38/U2_2/CO (FAIS)	0.21	13.11 f
Wire Load Model Mode: enclosed				cpu/CPUI/ard/sub_38/U2_3/CO (FAIS)	0.21	13.33 f
Startpoint: cpu/CPUI/stage3_register_out_reg[130]				cpu/CPUI/ard/sub_38/U2_4/CO (FAIS)	0.21	13.54 f
(rising edge-triggered flip-flop clocked by clk)				cpu/CPUI/ard/sub_38/U2_5/CO (FAIS)	0.21	13.75 f
Endpoint: cpu/CPUI/stage3_register_out_reg[95]				cpu/CPUI/ard/sub_38/U2_6/CO (FAIS)	0.21	13.97 f
(rising edge-triggered flip-flop clocked by clk)				cpu/CPUI/ard/sub_38/U2_7/CO (FAIS)	0.21	14.18 f
Path Group: clk				cpu/CPUI/ard/sub_38/U2_8/CO (FAIS)	0.21	14.39 f
Path Type: max				cpu/CPUI/ard/sub_38/U2_9/CO (FAIS)	0.21	14.61 f
Des/Clust/Port	Wire Load Model	Library		cpu/CPUI/ard/sub_38/U2_10/CO (FAIS)	0.21	14.82 f
top	enG500K	fsa0m_a_t33_generic_io_fflp98vm40c		cpu/CPUI/ard/sub_38/U2_11/CO (FAIS)	0.21	15.03 f
CPU	enG30K	fsa0m_a_generic_core_sslp62v125c		cpu/CPUI/ard/sub_38/U2_12/CO (FAIS)	0.21	15.24 f
forwarding_unit	enG5K	fsa0m_a_generic_core_sslp62v125c		cpu/CPUI/ard/sub_38/U2_13/CO (FAIS)	0.21	15.45 f
alu_in_selector	enG5K	fsa0m_a_generic_core_sslp62v125c		cpu/CPUI/ard/sub_38/U2_14/CO (FAIS)	0.21	15.67 f
alu_rd	enG10K	fsa0m_a_generic_core_sslp62v125c		cpu/CPUI/ard/sub_38/U2_15/CO (FAIS)	0.21	15.88 f
alu_rd_DW01_sub_0	enG5K	fsa0m_a_generic_core_sslp62v125c		cpu/CPUI/ard/sub_38/U2_16/CO (FAIS)	0.21	16.09 f
Point	Incr	Path		cpu/CPUI/ard/sub_38/U2_17/CO (FAIS)	0.21	16.30 f
clock clk (rise edge)	10.00	10.00		cpu/CPUI/ard/sub_38/U2_18/CO (FAIS)	0.21	16.51 f
clock network delay (ideal)	0.00	10.00		cpu/CPUI/ard/sub_38/U2_19/CO (FAIS)	0.21	16.72 f
cpu/CPUI/stage3_register_out_reg[130]/CK (QDFFRBN)	0.00 #	10.00 r		cpu/CPUI/ard/sub_38/U2_20/CO (FAIS)	0.21	16.94 f
cpu/CPUI/stage3_register_out_reg[130]/Q (QDFFRBN)	0.30	10.30 f		cpu/CPUI/ard/sub_38/U2_21/CO (FAIS)	0.21	17.15 f
cpu/CPUI/fwu/exe_mem_rd_addr[2] (forwarding_unit)	0.00	10.30 f		cpu/CPUI/ard/sub_38/U2_22/CO (FAIS)	0.21	17.36 f
cpu/CPUI/fwu/U4/O (NR3)	0.11	10.40 r		cpu/CPUI/ard/sub_38/U2_23/CO (FAIS)	0.21	17.58 f
cpu/CPUI/fwu/U3/O (AN3B2S)	0.13	10.54 r		cpu/CPUI/ard/sub_38/U2_24/CO (FAIS)	0.21	17.79 f
cpu/CPUI/fwu/U24/O (AN4B1S)	0.07	10.60 f		cpu/CPUI/ard/sub_38/U2_25/CO (FAIS)	0.21	18.00 f
cpu/CPUI/fwu/U15/O (AN4B1S)	0.15	10.75 f		cpu/CPUI/ard/sub_38/U2_26/CO (FAIS)	0.21	18.22 f
cpu/CPUI/fwu/rs2_exe_hazard (forwarding_unit)	0.00	10.75 f		cpu/CPUI/ard/sub_38/U2_27/CO (FAIS)	0.21	18.43 f
cpu/CPUI/ais/rs2_exe_hazard (alu_in_selector)	0.00	10.75 f		cpu/CPUI/ard/sub_38/U2_28/CO (FAIS)	0.21	18.65 f
cpu/CPUI/ais/U44/O (BUFICK)	0.38	11.13 f		cpu/CPUI/ard/sub_38/U2_29/CO (FAIS)	0.21	18.86 f
cpu/CPUI/ais/U43/O (AN2B1S)	0.18	11.31 r		cpu/CPUI/ard/sub_38/U2_30/CO (FAIS)	0.20	19.06 f
cpu/CPUI/ais/U28/O (BUFICK)	0.21	11.52 r		cpu/CPUI/ard/sub_38/U2_31/O (XOR3)	0.14	19.20 f
cpu/CPUI/ais/U55/O (A0222)	0.19	11.71 r		cpu/CPUI/ard/sub_38/D1FF[31] (alu_rd_DW01_sub_0)	0.00	19.20 f
cpu/CPUI/ais/src2_data[1] (alu_in_selector)	0.00	11.71 r		cpu/CPUI/ard/U926/O (A0222)	0.21	19.41 f
cpu/CPUI/U140/O (BUFICK)	0.10	11.81 r		cpu/CPUI/ard/U1231/O (OR3)	0.12	19.53 f
cpu/CPUI/ard/src2[1] (alu_rd)	0.00	11.81 r		cpu/CPUI/ard/alu_rd_data[31] (alu_rd)	0.00	19.53 f
cpu/CPUI/ard/U530/O (INVIS)	0.17	11.99 f		cpu/CPUI/U123/O (A022)	0.13	19.66 f
cpu/CPUI/ard/U529/O (BUFICK)	0.15	12.13 f		cpu/CPUI/stage3_register_out_reg[95]/D (QDFFRBN)	0.00	19.66 f
cpu/CPUI/ard/U159/O (INV2)	0.41	12.55 r		data arrival time		19.66
cpu/CPUI/ard/sub_38/B[1] (alu_rd_DW01_sub_0)	0.00	12.55 r		clock clk (rise edge)	30.00	30.00
cpu/CPUI/ard/sub_38/U4/O (INVIS)	0.10	12.65 f		clock network delay (ideal)	0.00	30.00
cpu/CPUI/ard/sub_38/U2_1/CO (FAIS)	0.25	12.90 f		cpu/CPUI/stage3_register_out_reg[95]/CK (QDFFRBN)	0.00	30.00 r
cpu/CPUI/ard/sub_38/U2_2/CO (FAIS)	0.21	13.11 f		library setup time	-0.11	29.89
cpu/CPUI/ard/sub_38/U2_3/CO (FAIS)	0.21	13.33 f		data required time		29.89
				data required time		29.89
				data arrival time		-19.66
				slack (MET)		10.23

※於報告的附檔中有截錄 time、power、area 的 txt 檔報告

在執行合成時會看到一些 warning 或 error 的出現，代表有些無法合成的部份或多餘的 code，例如：原先在 master wrapper 內 always_ff(posedge clk)begin end，這種原先在寫 code 時沒有刪除掉的 code，進行修正。

● Lesson learned :

➤ 周昱佑：

在這次的作業中對於 slave 與 master 之間的溝通介面有了更多的了解，這次我們所學的為 AXI4.0 的協定，在這之中從 hand shaking 的了解是最主要的設計要點，當 valid 與 ready 同時為 1 時才能將資料當作正確的讀入或送出，所以我以這兩種訊號作為狀態機進下一個 state 的條件，這個條件設計原本我並沒有考慮到，是因為在測試 jaspergold 時，發現到這兩個訊號如果其中一者沒有拉為 1，則另一者不能放下原本拉起的訊號，由於原本由狀態機的設計是依照 valid 來在給一個 clock time 的 ready，所以在這裡修正進下一個 state 的狀態，並依照 valid 來判定資料抓取的確認訊號，並用 address 與 AW、ARID 來判定訪問正確的 slave，由於我負責設計 slave 端的部分，同時考慮到要留一個 clock time 給 memory 作讀寫的動作，所以考慮到之後合成不能有 multi-edge clock 所以給一個 stage 作 memory 訪問用，另外一個修正點在 AW、ARvalid 同時送入時要先執行哪一個，所以多用了一個變數 flag 來作判斷，總結來說最後我將送至 slave 的訊號分為 hand shaking(控制、確認)與 data(要以 register 栓鎖)，由於一開始對這個協定不是很了解，所以花了不少時間跟實驗室的同學一起研究了一下 spec 並討論中間 AXI 的設計，並了解到其中並不能用純組合電路的方式來做設計，需要考慮到 master 訪問的優先度、valid 與 ready 訊號先後問題、未被選擇的 interface 訊號保持的問題等，基本上都要加上 register 來作資料得儲存，在 AXI 最後的設計上才知道 AW 與 W 是有可能同時送出資料的，所以在 arbiter 上判斷條件又作了點修正。

在做 prog0 的 debug 時也發現很多沒注意到的，例如:address，若是在 cpu 優先處理好除以 4 再送出，會使 AXI 有判定通道開啟上的錯誤，進而讀錯資料回傳給 master，另外在 master 端送資料過來時，我的 slave 能否在我所設計的 state 栓鎖住 data 並且保持穩定，同時在 default 時也保持住直到下一次訪問才能更新栓鎖的 data，這些設計上的小缺漏在整合之後才會看到的。

同時也感謝助教的協助解答了我不少關於 AXI protocol 的問題，讓我對這次陌生的作業學習到不少。

➤ 杜冠勳:

此次作業學習到了 AMBA AXI 的溝通架構 是以 handshake 的基礎上完成的
而這次作業中我設計了 AXI 架構 以及 CPUWrapper 架構
在設計 AXI 架構時,使用了三個狀態機互相送出訊號去做溝通,三個狀態機分別為
arbitor,讀取,以及寫入,arbiter 收到來自 master 的訊號時,會仲裁出通道使用權,
而沒被選中的通道則是被控制在讀 0, 讀取,以及寫入狀態機 接收來自 arbitor 狀態
機的訊號,線決定 slave 端口是否接上。

在設計 AXI 架構時因為過去不好的習慣導致在使用狀態機時產生了 latch 以及
一些 recursive 的電路 經過這次的練習修正了許多觀念 也成功設計出來。

在設計 CPUWrapper 架構時 因為要考慮到 cpu stall 住的問題 所以一開始就
有將 CPU 放入 wrapper 中 設計 只是沒想到我把 reset 接為 0 導致等於沒有接上,
但當時只測 wrapper 時 jg 是有全過的,如附圖,但當跑完 4 個 prog 回去測 jg 時
發現不會過了,原因可能如下

我們的 CPU 設計與 Wrapper 溝通 除了 read write 訊號 還有來自 wrapper 的
bus_stall 訊號線,而 bus_stall 訊號線的設計是 當 read write 訊號來時就會拉起,而
dm 優先做的關係 導致在設計 讀寫 dm 的狀態機必須多一個 state 來做為判斷是
否回狀態 0(等候 read write 訊號狀態)的依據

因為如果回到狀態 0 由於 IM 在讀取指令中 造成 pipeline register 會 stall 住
而造成那段時間的 dm_read 或者是 dm_write 為 1 會讓 dm 端口 一直送出
request 造成無窮迴圈 故在這次的設計中 兩個 master 端口有互傳控制訊號的情
形出現導致某一些 state 無法被測到(uncover)像是 dm 端口先拉 ARVaild
AWVaild 的情形 就不會有

- 原因一 CPU 會一直讀取指令 故 im_read 當初設計為恆拉為一
- 原因 2 儘管後來有改成在 load_hazard 避免再向 IM 讀取指令 但是考量
到設計的架構,我們將 bus stall 統一在 wrapper 做處理, wrapper 僅會送
一個控制訊號給 cpu 使用不用多作處理 才是符合 wrapper 設計 這導致
我們不用在 cpu 去協調 送出 IM DM Request 的優先順序

另一種寫法則是使用控制 cpu 讀寫的順序進 wrapper,但控制流程較為複雜,而
wrapper 還是要送訊號回來給 cpu 只是狀態機不用多一個 state 判斷 優先順序,而

是 CPU 端口處理,由於我們的設計是選前者,故會讓 CPU 只會有 IM DM 端口同時送出訊號的情形出現,或是單獨 IM 送出請求訊號的情形,造成 jg 有 uncover。

在與 CPU 整合上 須注意 bus_stall 情形必須優先於 其他影響 pipeline register 的情形出現 像是 load word 或是 jump 的情況

另外 write_data 回 register 時為了確保資料安全性,亦可在 stall 後一刻再拉起訊號 下一個 posedge clk 即可寫回。