2018

2. (15%) What are the advantages of using code coverage in simulation-based functional verification? If you got 100% statement coverage in the simulation, does it guarantee that the design is correct? Please explain the reasons of your answer.

1. 速度快，可在大电路使用，可测试驗證是否全面也可以減少不必要的 code statement。

2. 不会，如 statement coverage 100% 只代表有盖的 statement 有测試到，而無法驗證其餘未盖到的地方。

3. (15%) What are the advantages and disadvantages of formal verification? What are the keys to improve the efficiency of formal verification??

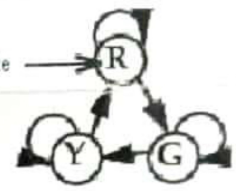
adv：能覆蓋完整的設計狀態，速度快。

dis：需要 memory 存 BDD，太大的 design 可能会導致 state explosion。

improve：使用 partial 或 bounded model checking 可減少狀態空間，或限制 cycle 數降低 memory 使用。

4. (10%) Do the CTL formulas below satisfy the STG shown right ? Please answer "TRUE" or "FALSE".

(a) **EG**(RED) T
(b) **AF**(GREEN) F
(c) **E**(RED **U** GREEN) T
(d) **A**(RED **U** GREEN) F
(e) **AG**(RED → **EX**(GREEN)) T
　　代 red 都在一條下次一定 green



5. (10%) Using the CTL formulas, how can we verify the design functionality? What are the issues of CTL-based verification?

1. CTL 提供了所有可能的路徑，藉由以 logic operator, temporal operators

1. path quantifiers構成的CTL formulas可驗證特定的條件屬性是否發生。

2. issue: state explosion，使用tree記錄若state太多 memory會不夠

6. (10%) Why do we need analog behavioral models for mixed-signal system verification? Compared to the original Verilog language, what are the major extensions of Verilog-AMS?

out : in = 1 : 0.4

In = 0.4out

1. 利用數學model去模擬behavioral level比hspice花費較少時間。

2. 支援一些analog function：微分積分... Ch10 P.55
contribution operator：<+。
包含單位的宣告：electrical
等analog behavior discription

In = 0.4out

7. (10%) Given a 4-phase sin-wave generator, please show the analog section of the behavioral model for this circuit based on Verilog-A. Its output voltage swing is 1.0V with 0.4V DC shift, and the phase difference between consecutive outputs is 90°. The signal frequency is 250MHz with maximum slew rate ±80MV/sec.

```
module sinwave (out1) out2 out3 out4
`define PI 3.14159
inout out;
real phase;
parameter real fc = 2.5e6;
electrical t1, t2, t3, t4, t5, out;
analog begin
    phase = 2.0 * PI * fc * $realtime();
  V(t1) = 0.5 sin (phase);
  V(t2) = 0.5 sin (phase + `PI/2);
  V(t3) = 0.5 sin (phase + `PI);
  V(t4) = 0.5 sin (phase + 3*PI/2);
```

V(out1) <+ slew(V(t1), 8e7, -8e7) + 0.4
V(out2) <+ slew(V(t2), 8e7, -8e7) + 0.4
V(out3) <+ slew(V(t3), 8e7, -8e7) + 0.4
V(out4) <+ slew(V(t4), 8e7, -8e7) + 0.4
end
endmodule

8. (15%) What are the benefits and limitations of simulation-based approach and equation-based approach for analog design automation? Why does the difference between synthesis results and post-layout simulation often exist?

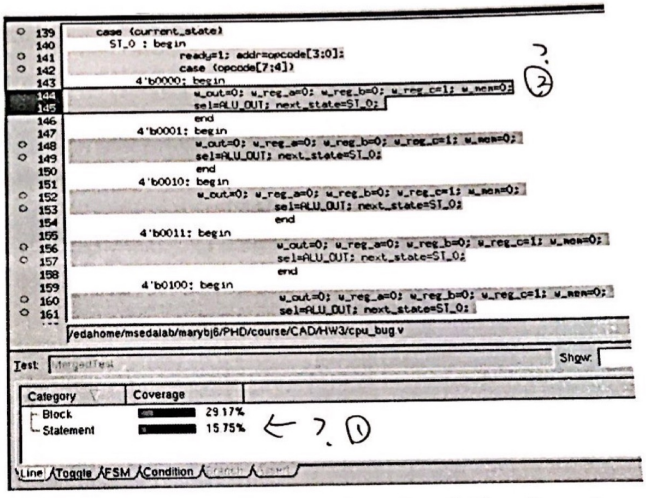
1. simu: 正確率高，耗時 → general. ch 13.
equ = 用GP較快，但簡化 non-linear equation 正確率下降.

Pareto-front-based: They can provide a tradeoff between different design objectives, allowing designers to choose a design point that best meets their needs / may not be able to find the optimal solution, as the optimal solution may lie outside of the search space, may require a large number of design simulations in order to generate the Pareto front, which can be computationally intensive.

2. synthesis 不会考慮 non-real effect (parasitics effect and physical effect) 因此 post-layout 後的結果可能不符合 spec.

2019

2. (15%) While you are running code coverage analysis, you got a coverage report as shown right. What are the meanings about the numbers in the bottom window and the red lines in the upper window? In your opinions, is the result good enough for functional verification? What can we do for the users, who are not satisfied with the results?



1. upper: 沒有被執行到的 statement
bottom: Block, statement coverage 的比例.

2. 不夠，statement coverage 沒100%代表有 case 沒測到.
儘管100%也無法代表 design 正確，但到需100%，verification 才有可信度

3. design 沒錯的情況下，增加 random pattern 數量或手刻 corner case

3. (15%) What are the benefits and drawbacks of using the simulation-based approach and formal-based approach for functional verification?? What are the keys to improve the efficiency of simulation-based verification??

sim：
- adv：可以簡單方法驗證複雜 design。
- dis：需要大量 pattern 驗證，很難完全測出所有可能。

formal：
- adv：能覆蓋完整的設計狀態，速度快。 Ch6. p36
- dis：需要 memory 存 BDD，太大的 design 可能會導致 state explosion。

improve：faster simulator, testbench tools

To speed up logic simulation by mapping some gate-level netlist into specific hardware

4. (15%) What is "equivalent checking" verification technique? What are the benefits and limitations of this technique?? Is it possible to have false alert in real cases with this technique??? Please explain your reasons.

1. 檢查兩 gate-level circuit 或 HDL 和 gate-level design 有無匹配 (mismatch) Ch8 p10. 檢查兩個design功能是否相同
2.
   - adv：檢查所有 input 可能 (solve SAT problem)。
   - dis：只檢查的出 implementation errors 非 design errors

   如果比較的design本身就是錯的，便無法被偵測
3. 有可能，可能因為建模過程有元件遺漏或人為失誤都有可能造成 false alert。

5. (15%) What are the differences of using top-down and bottom-up approaches for analog behavioral modeling in mixed-signal system designs? How can they help designers in the design flow??

1. Top-down：可簡略驗證AMS system 而未考慮 non-ideal effect

Bottom-up：加入 non-ideal effect 便驗證更精確。 Ch9 P4

2. Top-down 可幫助新 design 有個雛形，再加以修改。

Bottom-up 適合用於 Ip-based SOC designs 也更精確驗證.

6. (10%) Given a 2-bit DAC circuit, please show the analog section of the behavioral model for this circuit based on Verilog-A. The maximum input voltage is 1.0V, and the signal frequency is 250MHz. The output signal has a 0.5ns rising time and a 0.4ns falling time, and its input-to-output delay time is 1ns.

```
module DAC (in, out);
input [0:1] in;
output out;
voltage in, out;
parameter vth=1;
real code;
integer pow2[0:dac_size];
analog begin
    @(initial_step)
        for (i=0; i<=dac_size; i=i+1)
            pow2[i]= pow(2,i);
ations.

code=0;
for(i=0, i<dac_size; i=i+1)
    code = code+(V(in[i])>vth)?
                pow2[i]:0;

V(out) <+ transition (code/pow2[dac_size
        , 1, 0.5, 0.4);

end
endmodule.
```

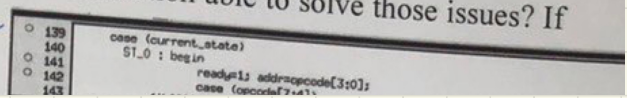
以下為3-bit的範例，比較清楚

```
`timescale 1ns/1ps

module dac_3bit (
    input [2:0] digital_in,  // 3-bit digital input
    output real analog_out   // Analog output
);

    // 定義參數
    parameter real V_MAX = 1.0;  // 最大輸出電壓
    parameter real DELAY = 1.0;  // 輸入到輸出的延遲時間 (ns)
    parameter real RISE_TIME = 0.5;  // 上升時間 (ns)
    parameter real FALL_TIME = 0.4;  // 下降時間 (ns)

    // 中間變量
    real v_out;

    // 行為模型，n-bit input就有2^n種case
    always @* begin
        case (digital_in)
            3'b000: v_out = 0.0;
            3'b001: v_out = V_MAX / 7.0;
            3'b010: v_out = 2 * V_MAX / 7.0;
            3'b011: v_out = 3 * V_MAX / 7.0;
            3'b100: v_out = 4 * V_MAX / 7.0;
            3'b101: v_out = 5 * V_MAX / 7.0;
            3'b110: v_out = 6 * V_MAX / 7.0;
            3'b111: v_out = V_MAX;
            default: v_out = 0.0;
        endcase
    end

    // 延遲和上升/下降時間處理
    analog begin
        analog_out <+ transition(v_out, DELAY, RISE_TIME, FALL_TIME);
    end

endmodule
```

2021

...Aided Design (IEE6661)

1. (15%) Why simulation-based functional verification often takes a lot of time for large designs? Is formal-based functional verification able to solve those issues? If so, why formal verification is not so popular in current design flow?



2. (10%) Wh...

1. 要完整的驗證所有可能的case需要大量pattern，且產生testbench也需額外時間。
2. formal不需input pattern，使用理論推導而可達100% coverage
3. 電路大，過於複雜，formal會state explosion.

5. (15%) Compared to the original Verilog language, what are the major extensions of Verilog-AMS? If we use the original Verilog language to model analog circuits, what issues should be solved??

1. 支援一些analog function：微分積分...
contribution operator：<+。
包含單位的宣告：electrical
等analog behavior discription

ch10 p.55

2. 缺少許多analog modeling的功能函語法，無法做微分積分等function，只能用近似導致正確率下降.

ch11 p10 12/12

Ch12. p21

✓ 6. (15%) What are the major non-ideal effects in an OPA circuit? How can we model those effects accurately in behavioral models??

1. noise, finite OP gain, slew rate, setting response,

② Extract 上述的 non-ideal effects 的數值 from simulation results.

- DC Gain: Measure the input/output voltages to get the real value.
- Output slew Rate: Measure the output slew rate of two different inputs to find the slew rate equation.
- Settling Response: 把標準公式的變因換成可量測的, 再代入取得值
- Noise: Obtained by Hspice command (.noise)

⇒由以上的數值建起的 behavioral model 較精準。

P12

7. (15%) How to consider layout-induced effects in analog circuit sizing? What are the key issues to be solved while considering the layout effects??

Ch14. p19. 12.

① 用 fast performance model for fast estimation of layout-induced effect during sizing。Rough P&R: extract the layout template of the original design. rough estimation before layout.

② for parastic effects: 要避免 sizing-layout loop.

for process variations: avoid simulating lot of samples iteratively. 要 fast & 準確 analysis for variation effects.

for lifetime yeild: Simulate at every time step → cost 太高. considered with fresh yeild simultaneously.

8. (15%) What are the benefits and limitations of simulation-based approach and equation-based approach for analog design automation? What are the possible bottlenecks for those circuit synthesis techniques to be applied on real cases?

1. simu: 正確率高，耗時 → general. ch 13.
   equ = 用GP較快，但簡化 non-linear equation 正確率下降.

2. synthesis 不会考慮 non-real effect (parasitics effect and physical effect) 因此 post-layout 後的結果可能不符合 spec.