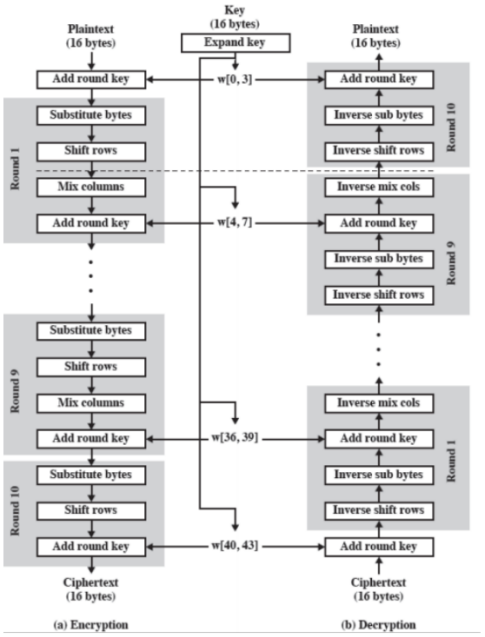
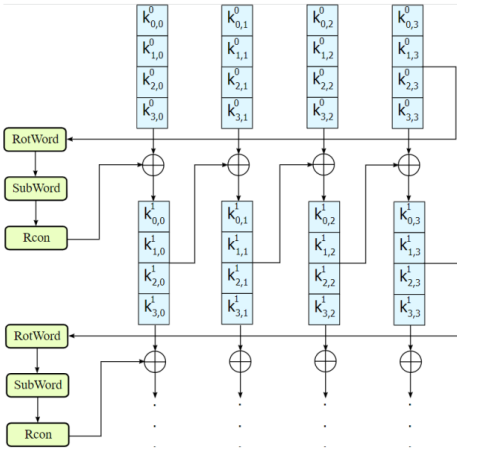
Team1_Final_project

(1)AES-128

Block diagram:





(ref. https://en.wikipedia.org/wiki/AES_key_schedule)

Source Code:

```
addroundkey(state_grid,0,sub,ekey);
#pragma hls_pipeline_init_interval 1  //Catapult Constraint
#pragma hls_unroll yes //catapult C Constraint
loop_main : for(iteration = 1; iteration < nr; iteration++) {
   subbytes(sub,shift);
   shift_row_enc(shift,mix);
   mixcolumn(mix,round);
   addroundkey(round,iteration,sub,ekey);

}
  subbytes(sub,shift);
  shift_row_enc(shift,round);
  addroundkey(round,nr,result,ekey);
```

圖為 aes-128 運作的邏輯。

(2-1)HLS code :

此次 catapult hls 因為我是找到 catapult 所公開的 github 內的 cpp code 以及 aes 流程滿固定的，所以整個結構基本上沒什麼修改，主要是語法改變還有原先 keyexpansion 他們是沒有要合成的，當要在 testbench 使用才呼叫並且放在 testbench 內，我們則把他放回來我們的主程式內改變大致如下:

```
void aes_enc(uint8_t state[16],uint8_t cipher[16],uint8_t ekey[240])
```

```
#pragma hls_design interface top
void CCS_BLOCK(run)(ac_channel<ac_int<128, false>> &state, ac_channel<ac_int<128, false>> &cipher, ac_channel<ac_int<128, false>>
```

上圖為原先 source code，下圖則為我改為使用 catapult 語法的方式，原先他提供的是 aes-256 而我改為 aes-128。

```
state_grid[0][0] = state[0]; cipher[0] = result[0][0];
state_grid[0][1] = state[1]; cipher[1] = result[0][1];
state_grid[0][2] = state[2]; cipher[2] = result[0][2];
state_grid[0][3] = state[3]; cipher[3] = result[0][3];
state_grid[1][0] = state[4]; cipher[4] = result[1][0];
state_grid[1][1] = state[5]; cipher[5] = result[1][1];
state_grid[1][2] = state[6]; cipher[6] = result[1][2];
state_grid[1][3] = state[7]; cipher[7] = result[1][3];
state_grid[2][0] = state[8]; cipher[8] = result[2][0];
state_grid[2][1] = state[9]; cipher[9] = result[2][1];
state_grid[2][2] = state[10];cipher[10] = result[2][2];
state_grid[2][3] = state[11];cipher[11] = result[2][3];
state_grid[3][0] = state[12];cipher[12] = result[3][0];
state_grid[3][1] = state[13];cipher[13] = result[3][1];
state_grid[3][2] = state[14];cipher[14] = result[3][2];
state_grid[3][3] = state[15];cipher[15] = result[3][3];
```

```
state_in = state.read();
for (x = 0; x < 4; x++) {
    for (y = 0; y < 4; y++) {
        state_grid.mat[x][y] = state_in.slc<8>(120 - (x + 4 * y) * 8).to_uint();
    }
}
```

```
for (x = 0; x < 4; x++) {
    for (y = 0; y < 4; y++) {
        cipher_out.set_slc(120 - (x + 4 * y) * 8, ac_int<8, false>(result.mat[x][y]));
    }
}

cipher.write(cipher_out);
```

值的讀入/讀出也需要因應 catapult 的語法而做修改，上圖為原先提供的寫法，我把它改成 for loop 讀取並且加上.read/.write，

```
addroundkey(state_grid,0,sub,ekey);
#pragma hls_pipeline_init_interval 1  //Catapult Constraint
#pragma hls_unroll yes //catapult C Constraint
loop_main : for(iteration = 1; iteration < nr; iteration++)
   subbytes(sub,shift);
   shift_row_enc(shift,mix);
   mixcolumn(mix,round);
   addroundkey(round,iteration,sub,ekey);

}
  subbytes(sub,shift);
  shift_row_enc(shift,round);
  addroundkey(round,nr,result,ekey);
```

```
keyexpansion(key, rkey);

addroundkey(state_grid, 0, sub, rkey);

#pragma hls_pipeline_init_interval 1  // Catapult Constrain
#pragma hls_unroll yes // Catapult C Constraint
loop_main: for (iteration = 1; iteration < 10; iteration++)
    subbytes(sub, shift);
    shift_row_enc(shift, mix);
    mixcolumn(mix, round);
    addroundkey(round, iteration, sub, rkey);
}
subbytes(sub, shift);
shift_row_enc(shift, round);
addroundkey(round, 10, result, rkey); // assuming nr is 10
```

上圖為原先提供的運作流程，我的運作流程基本上一樣但是原先的 source code 並沒有加上 keyexpansion 等到在 testbench 使用才在 testbench(原先寫在 testbench 內)呼叫，所以我有在主程式加上 keyexpansion，但邏輯基本上一致。

```
void shift_row_enc(uint8_t state[4][4], uint8_t result[4][4])
```

```
typedef struct {
    unsigned char mat[4][4];
} Mat;
```

```
void shift_row_enc(Mat &state, Mat &result)
```

上圖為原先的 sub-function 的使用，原為[4][4]的矩陣，我把他另外定義在 header file 並定義叫 Mat，但 sub-function 內的邏輯基本上沒做修改。

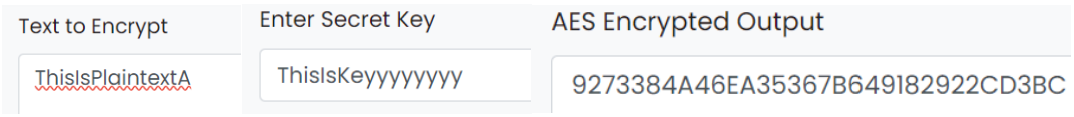
(2-2)Catapult hls

這次 catapult 我們有生出兩種架構，分別是想辦法讓他面積盡可能小和有最低的 latency 的架構，我們會去比較此兩種架構在 catapult 下的 area 和 timing，再與我們原先用 verilog 寫好的 aes 在 90nm 之下的 area 和 timing 去做比較。

下圖為我們驗證功能正確性的圖，上圖是我們 catapult 裡面模擬跑出來的結果，可以對上下圖產生出來的 aes output，表示我們的 catapult 產生的 aes 功能是正確的。

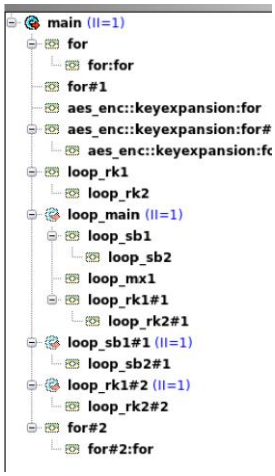
```
const char* state_str = "ThisIsPlaintextA";
const char* key_str = "ThisIsKeyyyyyyyy";
```

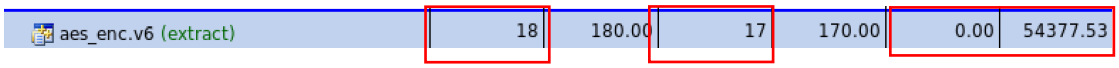
```
Message
make: Warning: File `scverify/orig_cxx_osci/scverify_top' has modificat
==========================================
Simulating design
cd ../..; ./Catapult_2/aes_enc.v4/scverify/orig_cxx_osci/scverify_top
 92 73 38 4a 46 ea 35 36 7b 64 91 82 92 2c d3 bc
finish
```



Aes_enc.v6(latency 最低)

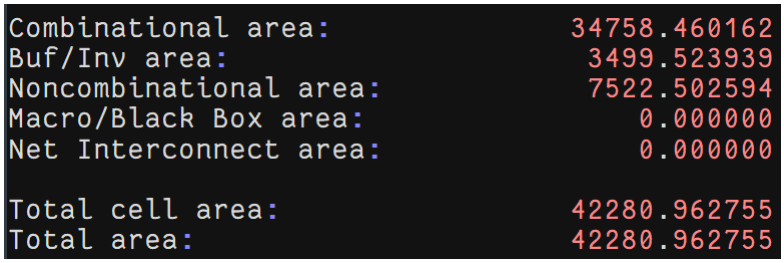
catapult hls 45nm

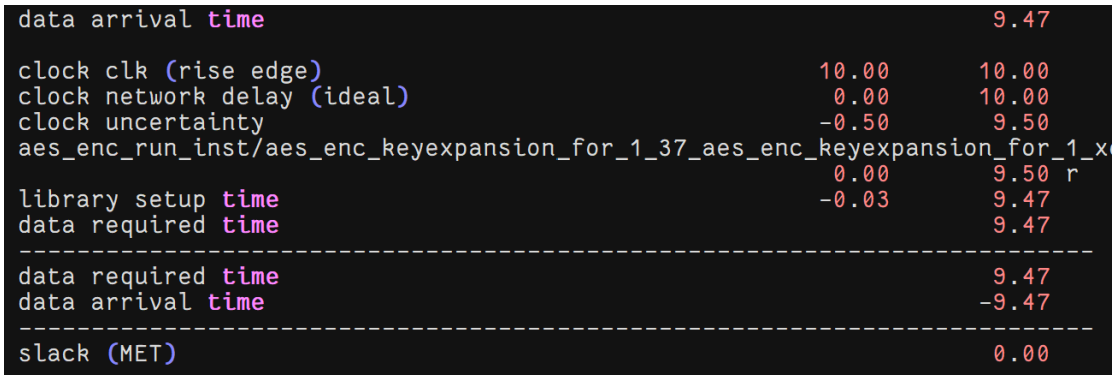




紅圈由左到右分別是 latency、throughput、slack 和 area，分別是 18、17、0 和 54377，此 architecture 為我們想盡可能讓其 latency 低，最後確實也得到最低的 latency 18T。

90nm

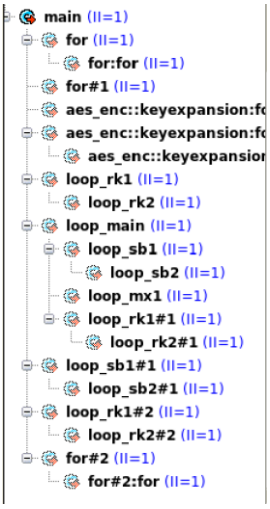




以上為我們在 90nm 的製程檔下進行合成的結果，其面積約為 42280um^2，slack 為 0.00。

Aes_enc.v8(area 小)

catapult hls 45nm



aes_enc.v8 (extract)	583	5830.00	584	5840.00	8.16	29917.95

此 architecture 為我們想盡可能讓其面積小，最後得出的面積約為 29917.95um^2，較上面的面積少了約 15000um^2，但其 latency 為 583T 大相當多。

90nm

```
Combinational area:                12403.237970
Buf/Inv area:                       1840.255165
Noncombinational area:             11758.446326
Macro/Black Box area:                  0.000000
Net Interconnect area:                 0.000000

Total cell area:                   24161.684296
Total area:                        24161.684296
```

```
data arrival time                                                   6.26

clock clk (rise edge)                                 10.00        10.00
clock network delay (ideal)                            0.00        10.00
clock uncertainty                                     -0.50         9.50
aes_enc_run_inst/aes_enc_keyexpansion_temp_0_lpi_1_reg[6]/CK (DFFQ_X0P5M_A9TR)
                                                       0.00         9.50 r
library setup time                                    -0.04         9.46
data required time                                                  9.46
-------------------------------------------------------------------------
data required time                                                  9.46
data arrival time                                                  -6.26
-------------------------------------------------------------------------
slack (MET)                                                         3.20
```

而在我們 90nm 的製程檔去合成的情況下，其面積為 24161um^2，slack 為 3.20，因為我們原先 verilog 的目的為盡可能縮減 area，所以我們後面會將此 aes_enc 去和我們的 aes 去做比較。

(3)Compare with Our's verilog aes

```
    sc9_cln40g_base_rvt_tt_typical_max_0p90v_25c (
/SynopsysDC/db/sc9_base_rvt/sc9_cln40g_base_rvt_tt

Number of ports:                         1344
Number of nets:                          5352
Number of cells:                         4256
Number of combinational cells:           3516
Number of sequential cells:               673
Number of macros/black boxes:               0
Number of buf/inv:                        622
Number of references:                      61

Combinational area:            3664.407518
Buf/Inv area:                   294.386394
Noncombinational area:         2599.808477
Macro/Black Box area:             0.000000
Net Interconnect area:            0.000000

Total cell area:               6264.215995
Total area:                    6264.215995
```

```
Combinational area:            12403.237970
Buf/Inv area:                   1840.255165
Noncombinational area:         11758.446326
Macro/Black Box area:              0.000000
Net Interconnect area:             0.000000

Total cell area:               24161.684296
Total area:                    24161.684296
```

左圖為我們原先 verilog 在 90nm 的製程下所合成出來的結果，其 area 為 6264um^2，表現優於使用 catapult hls 所生出來的 rtl 所合成出來的結果 24161um^2。

```
Des/Clust/Port     Wire Load Model       Library
------------------------------------------------
aes                Small                 sc9_cln40g_base_rvt

Point                                    Incr       Path
-----------------------------------------------------------
clock clk (rise edge)                    0.00       0.00
clock network delay (ideal)              0.00       0.00
input external delay                     4.00       4.00 f
dout_rdy (in)                            0.00       4.00 f
U1330/Y (AND2_X1M_A9TR)                  0.05       4.05 f
U1510/Y (INV_X3M_A9TR)                   0.21       4.26 r
U1649/Y (NOR2_X3B_A9TR)                  0.29       4.56 f
dout[39] (out)                           0.00       4.56 f
data arrival time                                   4.56

clock clk (rise edge)                   10.00      10.00
clock network delay (ideal)              0.00      10.00
clock uncertainty                       -0.50       9.50
output external delay                   -4.00       5.50
data required time                                  5.50
-----------------------------------------------------------
data required time                                  5.50
data arrival time                                  -4.56
-----------------------------------------------------------
slack (MET)                                         0.94
```

```
data arrival time                                                                  6.26

clock clk (rise edge)                                          10.00      10.00
clock network delay (ideal)                                     0.00      10.00
clock uncertainty                                              -0.50       9.50
aes_enc_run_inst/aes_enc_keyexpansion_temp_0_lpi_1_reg[6]/CK (DFFQ_X0P5M_A9TR)
                                                                0.00       9.50 r
library setup time                                             -0.04       9.46
data required time                                                         9.46
--------------------------------------------------------------------------------
data required time                                                         9.46
data arrival time                                                         -6.26
--------------------------------------------------------------------------------
slack (MET)                                                                3.20
```

Slack 的部分，左圖為我們 verilog 的 slack 約為 0.94，對比使用 catapult hls 所生出來的 rtl 所合成出來的結果 3.20。

而我們原先 verilog 的 Latency 表現約 400T，其表現也優於 catapult hls 所生出來的 latency 表現 583T，因此我們推測 catapult 或許對於時間最佳化的表現較好，所以我們後續進行 integrate 的 rtl 為 **latency 表現較好的 aes_enc.v6**。

(4)Integrate into FSIC

此次將 final project integrate into FSIC 的流程我們基本上照著 lab2-2 的實作過程，首先第一張圖是我們所使用的 filelist，concat_rtl_less_lat.v 是我們這次所使用的有最小 latency 的 rtl，其餘檔案基本上和 lab2-2 的一樣。

```
../fpga.v
../../rtl/axi_ctrl_logic.sv
../../rtl/axil_axis.sv
../../rtl/axilite_master.sv
../../rtl/axilite_slave.sv
../../rtl/axis_master.sv
../../rtl/axis_slave.sv
// ../../rtl/axis_switch.v
../../rtl/sw_caravel.v
../../rtl/config_ctrl.v
../../rtl/fsic_clkrst.v
../../rtl/fsic_clock.v
../../rtl/fsic_coreclk_phase_cnt.v
../../rtl/fsic_io_serdes_rx.v
../../rtl/fsic.v
../../rtl/io_serdes.v
../../rtl/logic_anlz.dummy_io.v
../../rtl/logic_anlz.dummy_io.vd
// ../../rtl/mprj_io.dummy_io.v
// ../../rtl/mprj_io.dummy_io.vd
../../rtl/mprj_io.sv
../../rtl/user_subsys.all.v
../../rtl/user_prj0.v
../../rtl/user_prj1.v
../../rtl/user_prj2.v
../../rtl/user_prj3.v
../../rtl/spram.v
../../rtl/concat_rtl_less_lat.v
```

更新 ：Input /output 透過 AXI 傳輸，input 由 ss_tdata 接收，並儲存在 SRAM 內，並等待與儲存於 register 內的 key 做 aes 並將做完的 cipher 存於 SRAM 等待輸出，最後再透過 sm_tdata 輸出。

我們這次的將我們的 aes 放在 user_prj0，透過給 protocol 來和 fsic 溝通，以下是操作原理與 port information。

Port Information

Port Name	Port Address	Width (bit)	Port Functionality
reg_rst	0x00	1	reset IP when reg_rst = 1
key0	0x10	32	32-bit key for encryption
key1	0x14	32	32-bit key for encryption
key2	0x18	32	32-bit key for encryption
key3	0x1C	32	32-bit key for encryption
dlen	0x08	32	length of data to encrypt
state (ss_tdata)	Dynamic	128	state input through ss_tdata
cipher (sm_tdata)	Dynamic	128	cipher output through sm_tdata

1. 系統初始化
2. 當有 awvalid/wvalid 時，根據 awaddr 寫入不同的 register (key)
3. 當有 arvalid 時表示有 read request，當 arready 時，即根據 araddr 進行讀取不同的 register
4. 當 ss_tready & ss_tvalid → 接收 state
5. 將接收的 state 和 key 進行 aes
6. 當 sm_tready & sm_tvalid → 輸出 cipher

最後跑出來的結果如下兩圖，我們總共測兩種 pattern，兩者都能夠通過，final result 最後也通過。

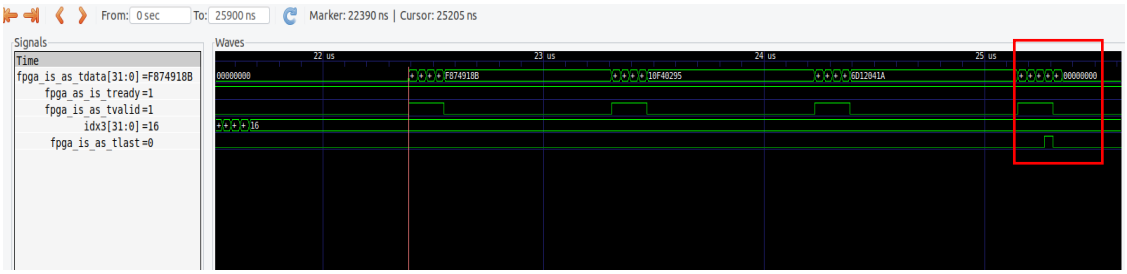
Input 64 筆 ：input pattern 透過 readmemh 接收，一次 pattern 有 64 個。

```
61 41
62 00
63 00
64 00
```

```
15925=> test002 [PASS] soc_to_fpga_axis_expect_count =         16, soc_to_fpga_axis_captured_count =         16
15925=> test002 [PASS] idx=          0, soc_to_fpga_axis_expect_value[          0] = 000049bdb9b6, soc_to_fpga_axis_captured[          0]  = 000049bdb9b6
15925=> test002 [PASS] idx=          1, soc_to_fpga_axis_expect_value[          1] = 000081d9576c, soc_to_fpga_axis_captured[          1]  = 000081d9576c
15925=> test002 [PASS] idx=          2, soc_to_fpga_axis_expect_value[          2] = 0000b131a0b5, soc_to_fpga_axis_captured[          2]  = 0000b131a0b5
15925=> test002 [PASS] idx=          3, soc_to_fpga_axis_expect_value[          3] = 00003ba9432f, soc_to_fpga_axis_captured[          3]  = 00003ba9432f
15925=> test002 [PASS] idx=          4, soc_to_fpga_axis_expect_value[          4] = 000049bdb9b6, soc_to_fpga_axis_captured[          4]  = 000049bdb9b6
15925=> test002 [PASS] idx=          5, soc_to_fpga_axis_expect_value[          5] = 000081d9576c, soc_to_fpga_axis_captured[          5]  = 000081d9576c
15925=> test002 [PASS] idx=          6, soc_to_fpga_axis_expect_value[          6] = 0000b131a0b5, soc_to_fpga_axis_captured[          6]  = 0000b131a0b5
15925=> test002 [PASS] idx=          7, soc_to_fpga_axis_expect_value[          7] = 00003ba9432f, soc_to_fpga_axis_captured[          7]  = 00003ba9432f
15925=> test002 [PASS] idx=          8, soc_to_fpga_axis_expect_value[          8] = 000049bdb9b6, soc_to_fpga_axis_captured[          8]  = 000049bdb9b6
15925=> test002 [PASS] idx=          9, soc_to_fpga_axis_expect_value[          9] = 000081d9576c, soc_to_fpga_axis_captured[          9]  = 000081d9576c
15925=> test002 [PASS] idx=         10, soc_to_fpga_axis_expect_value[         10] = 0000b131a0b5, soc_to_fpga_axis_captured[         10]  = 0000b131a0b5
15925=> test002 [PASS] idx=         11, soc_to_fpga_axis_expect_value[         11] = 00003ba9432f, soc_to_fpga_axis_captured[         11]  = 00003ba9432f
15925=> test002 [PASS] idx=         12, soc_to_fpga_axis_expect_value[         12] = 000049bdb9b6, soc_to_fpga_axis_captured[         12]  = 000049bdb9b6
15925=> test002 [PASS] idx=         13, soc_to_fpga_axis_expect_value[         13] = 000081d9576c, soc_to_fpga_axis_captured[         13]  = 000081d9576c
15925=> test002 [PASS] idx=         14, soc_to_fpga_axis_expect_value[         14] = 0000b131a0b5, soc_to_fpga_axis_captured[         14]  = 0000b131a0b5
15925=> test002 [PASS] idx=         15, soc_to_fpga_axis_expect_value[         15] = 00003ba9432f, soc_to_fpga_axis_captured[         15]  = 00013ba9432f
```

```
25305=> test002 [PASS] soc_to_fpga_axis_expect_count =         16, soc_to_fpga_axis_captured_count =         16
25305=> test002 [PASS] idx=          0, soc_to_fpga_axis_expect_value[          0] = 0000f874918b, soc_to_fpga_axis_captured[          0]  = 0000f874918b
25305=> test002 [PASS] idx=          1, soc_to_fpga_axis_expect_value[          1] = 0000e84a3016, soc_to_fpga_axis_captured[          1]  = 0000e84a3016
25305=> test002 [PASS] idx=          2, soc_to_fpga_axis_expect_value[          2] = 00002f260218, soc_to_fpga_axis_captured[          2]  = 00002f260218
25305=> test002 [PASS] idx=          3, soc_to_fpga_axis_expect_value[          3] = 00003975cf8b, soc_to_fpga_axis_captured[          3]  = 00003975cf8b
25305=> test002 [PASS] idx=          4, soc_to_fpga_axis_expect_value[          4] = 000010f40295, soc_to_fpga_axis_captured[          4]  = 000010f40295
25305=> test002 [PASS] idx=          5, soc_to_fpga_axis_expect_value[          5] = 000039c11671, soc_to_fpga_axis_captured[          5]  = 000039c11671
25305=> test002 [PASS] idx=          6, soc_to_fpga_axis_expect_value[          6] = 00008330d4e8, soc_to_fpga_axis_captured[          6]  = 00008330d4e8
25305=> test002 [PASS] idx=          7, soc_to_fpga_axis_expect_value[          7] = 0000a4bc2f87, soc_to_fpga_axis_captured[          7]  = 0000a4bc2f87
25305=> test002 [PASS] idx=          8, soc_to_fpga_axis_expect_value[          8] = 00006d12041a, soc_to_fpga_axis_captured[          8]  = 00006d12041a
25305=> test002 [PASS] idx=          9, soc_to_fpga_axis_expect_value[          9] = 00005c4823a6, soc_to_fpga_axis_captured[          9]  = 00005c4823a6
25305=> test002 [PASS] idx=         10, soc_to_fpga_axis_expect_value[         10] = 00007e22d51e, soc_to_fpga_axis_captured[         10]  = 00007e22d51e
25305=> test002 [PASS] idx=         11, soc_to_fpga_axis_expect_value[         11] = 000026e64ac7, soc_to_fpga_axis_captured[         11]  = 000026e64ac7
25305=> test002 [PASS] idx=         12, soc_to_fpga_axis_expect_value[         12] = 00003e87a927, soc_to_fpga_axis_captured[         12]  = 00003e87a927
25305=> test002 [PASS] idx=         13, soc_to_fpga_axis_expect_value[         13] = 000088c09ac9, soc_to_fpga_axis_captured[         13]  = 000088c09ac9
25305=> test002 [PASS] idx=         14, soc_to_fpga_axis_expect_value[         14] = 00000543d9cf, soc_to_fpga_axis_captured[         14]  = 00000543d9cf
25305=> test002 [PASS] idx=         15, soc_to_fpga_axis_expect_value[         15] = 0000e4008bf7, soc_to_fpga_axis_captured[         15]  = 0001e4008bf7
================================================================================
================================================================================
================================================================================
            25905=> Final result [PASS], check_cnt = 0059, error_cnt = 0000
================================================================================
================================================================================
================================================================================
$finish called at time : 25905 ns : File "/home/ubuntu/soc2_final/fsic/testbench/tb_fsic.v" Line 440
## quit
INFO: [Common 17-206] Exiting xsim at Sun Jun 16 22:23:16 2024...
ubuntu@ubuntu2004:~/soc2_final/fsic/testbench/tc$
```

波型圖如下，測一次會給 16 筆 data，每 4 筆要傳輸時會拉 tvalid，當最後 4 筆傳輸出去後會拉 tlast，而我們做完 16 筆大約花 25205 – 22390 = 2815ns。



(5)Synopsys flow

Synopsys flow 的部分我們主要也是跟著 lab3 來做，不過要更改 common.tcl 把 design name 改成我們的 aes_enc，clk 的部分則從.sdc 修改我們最初由 2 開始，不過合成的時候 slack 會有 violation，之後我們改成 4，合成 slack 可以通過但是後續跑 CTS 等等後面的動作會有 violation，因此最後我們將 clk 改為 8，則能順利通過合成並且後續 CTS 也不會有 violation，後續的部分幾乎都是照著流程走完，但是在 lab4-finishing 的部分 tcl 檔需要稍作修改才能順利打完晶片。

以下是我們各 lab 的運作結果以及該階段的 Layout 圖:

1. Lab Synthesis:

Area: 37975um^2

```
Library(s) Used:

    saed14rvt_tt0p8v25c (File: /home/course/ee5252/

Number of ports:                        4357
Number of nets:                        40972
Number of cells:                       36701
Number of combinational cells:         34655
Number of sequential cells:             1943
Number of macros/black boxes:              0
Number of buf/inv:                      5615
Number of references:                      1

Combinational area:             10770.285600
Buf/Inv area:                    1199.998809
Noncombinational area:           1666.731556
Macro/Black Box area:               0.000000
Net Interconnect area:          25538.404033

Total cell area:                12437.017156
Total area:                     37975.421189
```

Timing:

Clk4 : Slack : 0.04

```
clock clk (rise edge)                                  4.00       4.00
clock network delay (ideal)                            0.00       4.00
clock uncertainty                                     -0.30       3.70
aes_enc_run_inst/aes_enc_keyexpansion_for_1_37_aes_enc_keyexpansion_for_1_xor_7_ncse_lpi_1_dfm_reg[2]/CK (SAEDRVT14_FDP_V2LP_1)
                                                       0.00       3.70 r
library setup time                                     0.00       3.70
data required time                                                3.70
--------------------------------------------------------------------------
data required time                                                3.70
data arrival time                                                -3.66
--------------------------------------------------------------------------
slack (MET)                                                       0.04
```

Clk 8 : Slack : 3.82

```
data arrival time                                                 3.88

clock clk (rise edge)                                  8.00       8.00
clock network delay (ideal)                            0.00       8.00
clock uncertainty                                     -0.30       7.70
aes_enc_run_inst/aes_enc_keyexpansion_for_1_37_aes_enc_keyexpansion_for_
                                                       0.00       7.70
library setup time                                     0.00       7.70
data required time                                                7.70
--------------------------------------------------------------------------
data required time                                                7.70
data arrival time                                                -3.88
--------------------------------------------------------------------------
slack (MET)                                                       3.82
```

Power:

Clk4 :

```
--------------------------------------------------------------------------------
                                    Switch   Int      Leak     Total
Hierarchy                           Power    Power    Power    Power    %
--------------------------------------------------------------------------------
aes_enc                              556.007  951.519 5.23e+06 1.51e+03 100.0
  aes_enc_run_inst (aes_enc_run)     556.007  951.510 5.23e+06 1.51e+03 100.0
    aes_enc_run_run_fsm_inst (aes_enc_run_run_fsm)
                                    2.33e-02    0.384  767.993    0.408   0.0
```

Clk 8 :

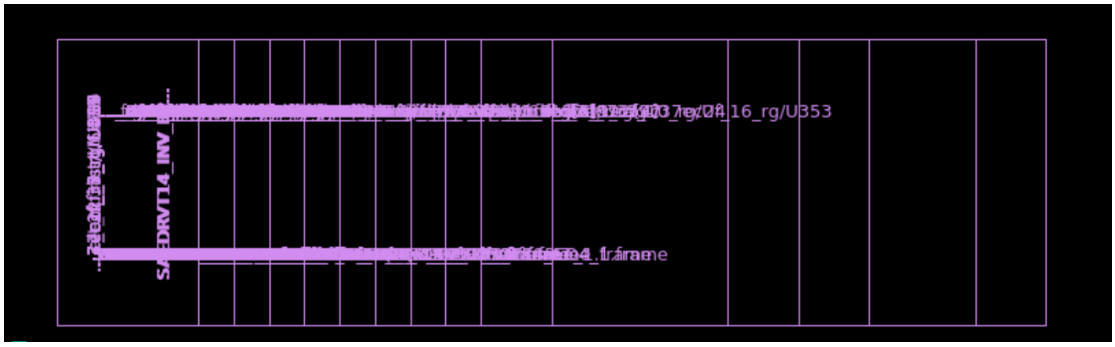
```
                                    Switch   Int      Leak     Total
Hierarchy                           Power    Power    Power    Power    %
--------------------------------------------------------------------------------
aes_enc                              294.702  482.837 5.40e+06  782.935 100.0
  aes_enc_run_inst (aes_enc_run)     294.702  482.836 5.40e+06  782.934 100.0
```

2. Lab_1_planning:

Step 1 data_setup:

```
################################################################################Import_Design
read_verilog "$Core_compile"
Loading verilog file '/home/m112/m112061601/HLS/Final_syn/input/aes_enc.v'
Information: Reading Verilog into new design 'aes_enc' in library 'aes_enc'. (VR-012)
Number of modules read: 104
Top level ports: 395
Total ports in all modules: 4357
Total nets in all modules: 41026
Total instances in all modules: 36701
Elapsed = 00:00:00.30, CPU = 00:00:00.29
1
current_design ${DESIGN_NAME}
{aes_enc:aes_enc.design}
```

Layout:



Step 2 floorplan:

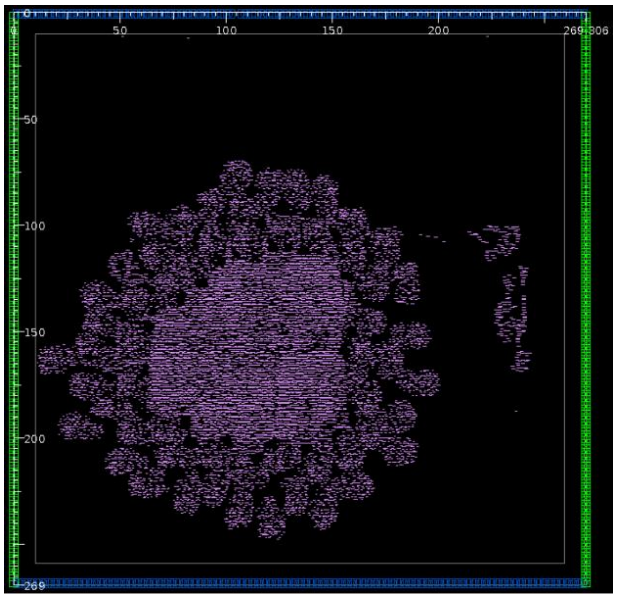
Source used:

```
phase4. Total Wire Length = 391652.07
phase4. Layer M1 wire length = 6236.86
phase4. Layer M2 wire length = 119249.86
phase4. Layer M3 wire length = 141502.48
phase4. Layer M4 wire length = 66439.81
phase4. Layer M5 wire length = 49039.23
phase4. Layer M6 wire length = 6137.72
phase4. Layer M7 wire length = 3046.12
phase4. Layer M8 wire length = 0.00
phase4. Layer M9 wire length = 0.00
phase4. Layer MRDL wire length = 0.00
phase4. Total Number of Contacts = 271513
phase4. Via VIA12SQ_C count = 102788
phase4. Via VIA23SQ_C count = 137313
phase4. Via VIA34SQ_C count = 20342
phase4. Via VIA45SQ count = 10297
phase4. Via VIA56SQ count = 561
phase4. Via VIA67SQ_C count = 212
phase4. Via VIA78SQ_C count = 0
phase4. Via VIA89_C count = 0
phase4. Via VIA9RDL count = 0
phase4. completed.
```

No violations:

```
980  Wire length report (all)
981  ========================
982  wire length in design temp_data_setup: 350874.150 microns.
983  wire length in design temp_data_setup (see through blk pins): 350874.150 microns.
984
985  Physical hierarchy violations report
986  ====================================
987  Violations in design temp_data_setup:
988     0 cells have placement violation.
989
990  Voltage area violations report
991  ==================================
992  Voltage area placement violations in design temp_data_setup:
993     0 cells placed outside the voltage area which they belong to.
994
995 Information: Default error view temp_data_setup_dpplace.err is created in GUI error browser. (DPP-054)
```

Layout:



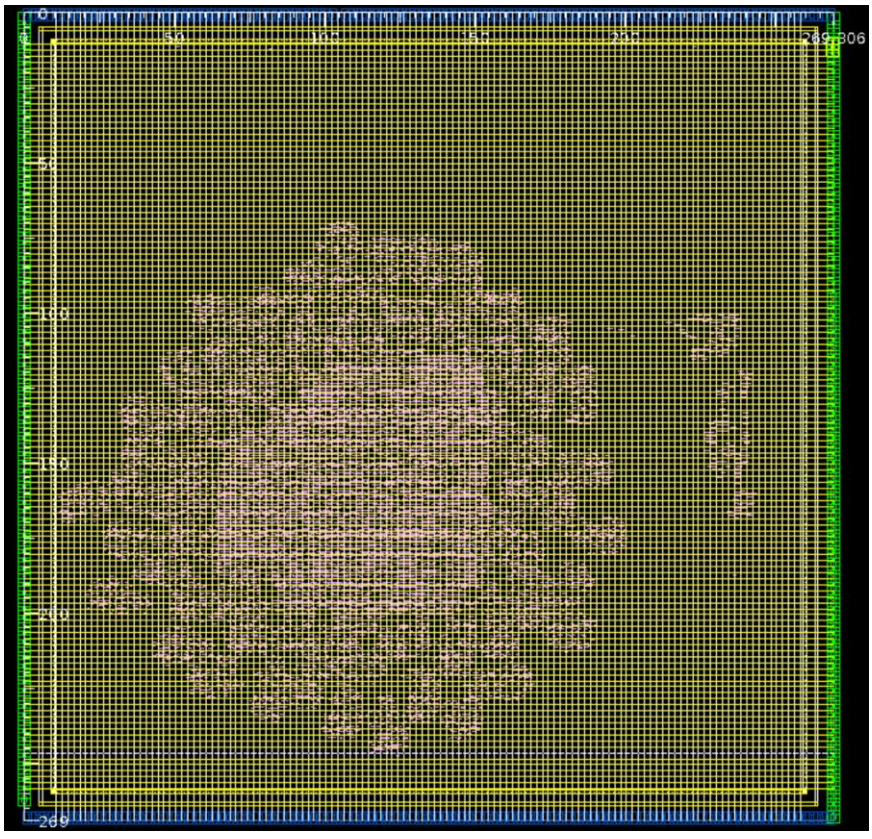
刻度為 269.806um * 269 um

```
Congestion utilization per direction:
Average vertical track utilization   =  8.99 %
Peak    vertical track utilization   = 78.95 %
Average horizontal track utilization =  7.98 %
Peak    horizontal track utilization = 64.00 %
```

Step 3 powerplan:

```
check_pg_connectivity -nets "VDD VSS"
Loading cell instances...
Number of Standard Cells: 36598
Number of Macro Cells: 0
Number of IO Pad Cells: 0
Number of Blocks: 0
Loading P/G wires and vias...
Number of VDD Wires: 336
Number of VDD Vias: 4096
Number of VDD Terminals: 249
Number of VSS Wires: 338
Number of VSS Vias: 4225
Number of VSS Terminals: 253
**************Verify net VDD connectivity*****************
  Number of floating wires: 208
  Number of floating vias: 0
  Number of floating std cells: 36598
  Number of floating hard macros: 0
  Number of floating I/O pads: 0
  Number of floating terminals: 1
  Number of floating hierarchical blocks: 0
************************************************************
**************Verify net VSS connectivity*****************
  Number of floating wires: 208
  Number of floating vias: 0
  Number of floating std cells: 36598
  Number of floating hard macros: 0
  Number of floating I/O pads: 0
  Number of floating terminals: 1
  Number of floating hierarchical blocks: 0
************************************************************
Overall runtime: 0 seconds.
```

Layout:

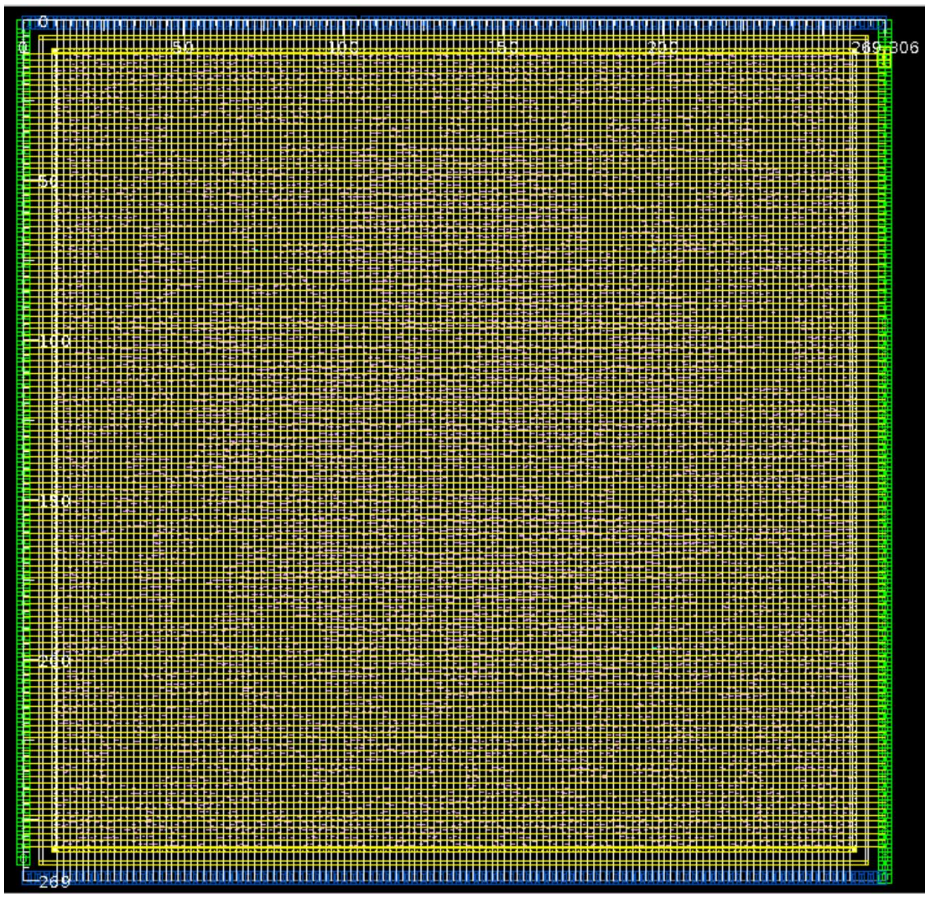


3. Lab2_pnr:

Step 4 place:

```
3564 ------------------------------------------------------------------------------
3565 WNS of each timing group:                    s1        s2
3566 ------------------------------------------------------------------------------
3567 clk                                      0.1842    0.0019
3568 ------------------------------------------------------------------------------
3569 Setup WNS:                               0.1842    0.0019       0.0019
3570 Setup TNS:                               0.0000    0.0000       0.0000
3571 Number of setup violations:                   0         0            0
3572 Hold WNS:                               0.0259    0.0261       0.0259
3573 Hold TNS:                               0.0000    0.0000       0.0000
3574 Number of hold violations:                    0         0            0
3575 Number of max trans violations:               0         0            0
3576 Number of max cap violations:                 0         0            0
3577 Number of min pulse width violations:         0         0            0
3578 ------------------------------------------------------------------------------
3579 Area:                                                          12207.469
3580 Cell count:                                                       35042
3581 Buf/inv cell count:                                                4059
3582 Std cell utilization:                                            0.1966
3583 CPU(s):                                                             456
3584 Mem(Mb):                                                            914
3585 Host name:                                                         ws24
3586 ------------------------------------------------------------------------------
3587 Histogram:             s1   s2
3588 ------------------------------------------------------------------------------
```

Layout:



Step 5 Clock Tree Synthesis:

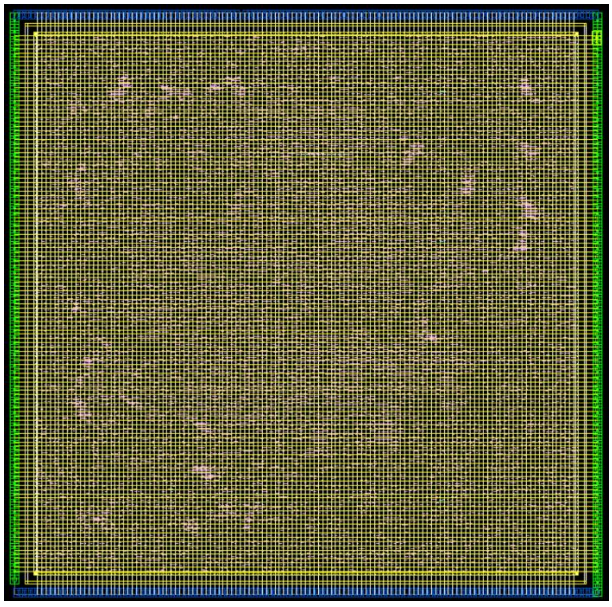
WNS of each timing group:	s1	s2	
clk	-	-0.0103	
Setup WNS:	0.2126	-0.0103	-0.0103
Setup TNS:	0.0000	-0.1283	-0.1283
Number of setup violations:	0	28	28
Hold WNS:	0.0211	0.0224	0.0211
Hold TNS:	0.0000	0.0000	0.0000
Number of hold violations:	0	0	0
Number of max trans violations:	0	0	0
Number of max cap violations:	3	12	12
Number of min pulse width violations:	0	0	0

更新: clock 8

WNS of each timing group:	s1	s2	
clk	1.0740	0.4295	
Setup WNS:	1.0740	0.4295	0.4295
Setup TNS:	0.0000	0.0000	0.0000
Number of setup violations:	0	0	0
Hold WNS:	0.0214	0.0226	0.0214
Hold TNS:	0.0000	0.0000	0.0000
Number of hold violations:	0	0	0
Number of max trans violations:	0	0	0
Number of max cap violations:	3	20	20
Number of min pulse width violations:	0	0	0
Area:			12437.017
Cell count:			35962
Buf/inv cell count:			3794
Std cell utilization:			0.2011
CPU(s):			942
Mem(Mb):			1685
Host name:			ws24
Histogram:	s1	s2	

因為原先上圖的 clk 使用 4 會有 violation，因此改用下圖的 clk 8。

Layout:



Step 6 Route:

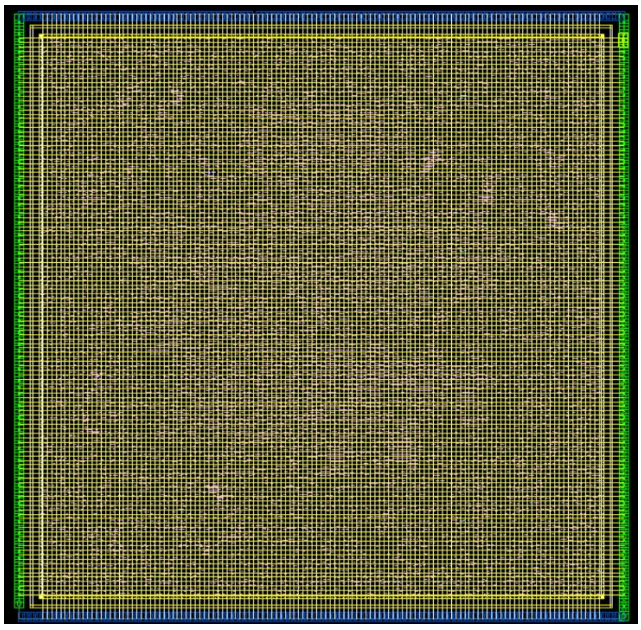
WNS of each timing group:	s1	s2	
clk	-	-0.0033	
Setup WNS:	0.2250	-0.0033	-0.0033
Setup TNS:	0.0000	-0.0072	-0.0072
Number of setup violations:	0	5	5
Hold WNS:	0.0264	0.0267	0.0264
Hold TNS:	0.0000	0.0000	0.0000
Number of hold violations:	0	0	0
Number of max trans violations:	0	0	0
Number of max cap violations:	0	4	4
Number of min pulse width violations:	0	0	0
Area:			12999.565
Cell count:			36794
Buf/inv cell count:			4177
Std cell utilization:			0.2094
CPU(s):			1447
Mem(Mb):			1045
Host name:			ws24

更:

WNS of each timing group:	s1	s2	
clk	1.8783	1.3453	
Setup WNS:	1.8783	1.3453	1.3453
Setup TNS:	0.0000	0.0000	0.0000
Number of setup violations:	0	0	0
Hold WNS:	0.0264	0.0267	0.0264
Hold TNS:	0.0000	0.0000	0.0000
Number of hold violations:	0	0	0
Number of max trans violations:	0	0	0
Number of max cap violations:	0	3	3
Number of min pulse width violations:	0	0	0
Area:			12416.016
Cell count:			35951
Buf/inv cell count:			3783
Std cell utilization:			0.2008
CPU(s):			1690
Mem(Mb):			1075
Host name:			ws24

同樣 clk 從 4→8 避免 violation。

Layout:

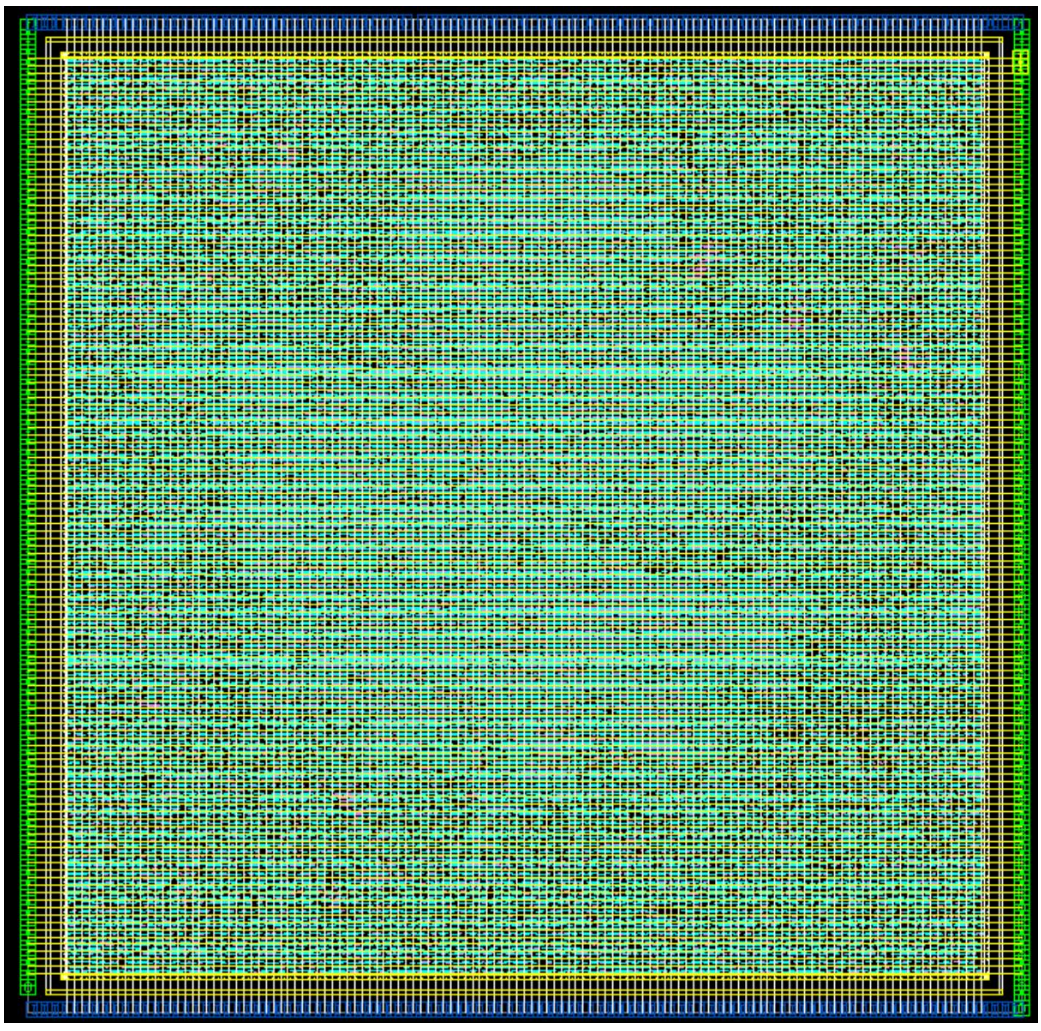


4. Lab4_finishing:

Step 7 finishing:

```
----------------------------------------
VERSION                          : 1
DIVIDERCHAR                      : 1
BUSBITCHARS                      : 1
DESIGN                           : 1
UNITS                            : 1
PROPERTYDEFINITIONS              : 1
DIEAREA                          : 1
ROW                              : 415
TRACKS                           : 20
VIAS                             : 38
NONDEFAULTRULES                  : 1
COMPONENTS                       : 200863
PINS                             : 395
PINPROPERTIES                    : 395
SPECIALNETS                      : 4237
NETS                             : 37230
1
```

Layout:



5. Lab_formal:

```
********************************* Verification Results *********************************
Verification SUCCEEDED
---------------------
 Reference design: r:/WORK/aes_enc
 Implementation design: i:/WORK/aes_enc
 2074 Passing compare points
-----------------------------------------------------------------------------------------
Matched Compare Points     BBPin    Loop   BBNet     Cut     Port     DFF      LAT    TOTAL
-----------------------------------------------------------------------------------------
Passing (equivalent)           0       0       0       0      131    1943        0     2074
Failing (not equivalent)       0       0       0       0        0       0        0        0
*****************************************************************************************
1
quit

Maximum memory usage for this session: 731 MB
CPU usage for this session: 23.2 seconds ( 0.01 hours )
Current time: Sun Jun 16 15:46:26 2024
Elapsed time: 128 seconds ( 0.04 hours )

Thank you for using Formality (R)!
```

Verification SUCCEEDED

6. lab_StarRC:

```
Setup                   Elp=00:00:35 Cpu=00:00:01 Usr=1.6    Sys=0.0    Mem=454.5
Layers_part1            Elp=00:00:01 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=454.6
Layers_part2            Elp=00:00:00 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=456.3
HN_partHN               Elp=00:00:00 Cpu=00:00:00 Usr=0.4    Sys=0.0    Mem=469.1
HN_partSplitTopDef1     Elp=00:00:01 Cpu=00:00:00 Usr=0.2    Sys=0.0    Mem=399.9
HN_partSplitTopDef2     Elp=00:00:01 Cpu=00:00:00 Usr=0.2    Sys=0.0    Mem=399.8
HN_partSplitTopDef3     Elp=00:00:00 Cpu=00:00:00 Usr=0.3    Sys=0.0    Mem=399.8
Cells_partStdCells1     Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=463.5
Cells_partTopCell1      Elp=00:00:01 Cpu=00:00:00 Usr=0.5    Sys=0.0    Mem=471.3
Cells_partTopCell2      Elp=00:00:01 Cpu=00:00:00 Usr=0.4    Sys=0.0    Mem=467.2
Cells_partTopCell3      Elp=00:00:00 Cpu=00:00:00 Usr=0.3    Sys=0.0    Mem=467.2
MergeCells              Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=456.6
Translate_part1         Elp=00:00:01 Cpu=00:00:00 Usr=0.8    Sys=0.0    Mem=456.6
Translate_part2         Elp=00:00:01 Cpu=00:00:00 Usr=0.5    Sys=0.0    Mem=456.6
XinCombine              Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=394.9
PartitionPP             Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=456.6
MergeIndexPins          Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=456.6
MergeIndex              Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=456.6
MergeGpdInstPins        Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=456.6
GPD_XtractSetup         Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=457.9
NetlistSetup            Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=397.8
NetlistAssemblySetup    Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=394.9
CleanDirectory          Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=394.9
GPD_NameMap             Elp=00:00:01 Cpu=00:00:00 Usr=0.2    Sys=0.0    Mem=397.9
StitchPreProcess1       Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=0.0
xTract_part1            Elp=00:00:41 Cpu=00:00:38 Usr=35.9   Sys=2.2    Mem=1389.0
xTract_part2            Elp=00:00:00 Cpu=00:00:00 Usr=0.4    Sys=0.1    Mem=470.0
xTract_part3            Elp=00:00:01 Cpu=00:00:00 Usr=0.5    Sys=0.1    Mem=470.0
xTract_part4            Elp=00:00:00 Cpu=00:00:00 Usr=0.5    Sys=0.1    Mem=470.0
StitchPreProcess2       Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=271.6
Stitch_part1            Elp=00:00:00 Cpu=00:00:00 Usr=0.2    Sys=0.0    Mem=278.0
Stitch_part2            Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=278.0
Stitch_part3            Elp=00:00:01 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=278.0
xTractPP                Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=278.4
ReportViolations        Elp=00:00:00 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=399.6
ReportOpens             Elp=00:00:01 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=394.9
Netlist_assembly1       Elp=00:00:00 Cpu=00:00:00 Usr=0.2    Sys=0.0    Mem=275.3
Netlist_assembly2       Elp=00:00:00 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=274.6
Netlist_assembly3       Elp=00:00:01 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=274.9
Netlist_assembly4       Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=273.8
GPD_PostProcess         Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=395.3
GPD_Converter1          Elp=00:00:01 Cpu=00:00:00 Usr=0.7    Sys=0.0    Mem=281.8
GPD_Converter2          Elp=00:00:01 Cpu=00:00:00 Usr=0.2    Sys=0.0    Mem=274.9
GPD_Converter3          Elp=00:00:00 Cpu=00:00:00 Usr=0.4    Sys=0.0    Mem=277.7
GPD_Converter4          Elp=00:00:01 Cpu=00:00:00 Usr=0.3    Sys=0.0    Mem=277.2
GPD_Converter5          Elp=00:00:00 Cpu=00:00:00 Usr=0.2    Sys=0.0    Mem=273.1
GPD_Converter6          Elp=00:00:00 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=272.5
GPD_Converter7          Elp=00:00:00 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=272.7
```

No violations:

```
GPD Converter Part 4:  Mon Jun 17 00:08:23 2024
  Warnings: 0    Errors: 0
  GPD_Converter4 Elp=00:00:01 Cpu=00:00:00 Usr=0.3    Sys=0.0    Mem=277.2
Done

GPD Converter Part 5:  Mon Jun 17 00:08:27 2024
  Warnings: 0    Errors: 0
  GPD_Converter5 Elp=00:00:00 Cpu=00:00:00 Usr=0.2    Sys=0.0    Mem=273.1
Done

GPD Converter Part 6:  Mon Jun 17 00:08:30 2024
  Warnings: 0    Errors: 0
  GPD_Converter6 Elp=00:00:00 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=272.5
Done

GPD Converter Part 7:  Mon Jun 17 00:08:32 2024
  Warnings: 0    Errors: 0
  GPD_Converter7 Elp=00:00:00 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=272.7
Done

GPD Converter Part 8:  Mon Jun 17 00:08:35 2024
  Warnings: 0    Errors: 0
  GPD_Converter8 Elp=00:00:00 Cpu=00:00:00 Usr=0.1    Sys=0.0    Mem=272.2
Done

GPD Converter Merge Corner 1:  Mon Jun 17 00:08:37 2024
  Warnings: 0    Errors: 0
  GPD_Converter_merge_c1 Elp=00:00:01 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=271.6
Done

GPD Converter Merge Corner 2:  Mon Jun 17 00:08:41 2024
  Warnings: 0    Errors: 0
  GPD_Converter_merge_c2 Elp=00:00:00 Cpu=00:00:00 Usr=0.0    Sys=0.0    Mem=271.6
Done
```

(6)FPGA simulation & validation

FPGA simulation & validation 的部分我們也是大致上照著 lab4 走，主要是把我們的 concat_rtl 和 user_prj0 丟到 vivado_src 裡面，並且把 tcl 改成吃到我們的 concat_rtl 和 user_prj0，還有修改 fsic_tb.sv 以完成 vivado_fsic_sim，最後再把 vivado_fsic 跑出來的.bit 和.hwh 丟到 jupyter notebook 上看結果。

1.FPGA simulation

```
          62001258=> AXI4LITE_WRITE_BURST 00000020, value: 0002, resp: 00
xecuting Axi4 End Of Simulation checks
arning: AXI4_ERRS_RLAST_ALL_DONE_EOS: All outstanding read bursts must have com
leted a the end of the simulation.
ime: 62009858 ns  Iteration: 0  Process: /fsic_tb/DUT/design_1_i/axi_vip_0/inst
IF/PC/Always3109_1895  Scope: fsic_tb.DUT.design_1_i.axi_vip_0.inst.IF.PC.arm_a
ba4_pc_msg_err  File: /tools/Xilinx/Vivado/2022.1/data/xilinx_vip/hdl/axi_vip_a
i4pc.sv Line: 725
xecuting Axi4 End Of Simulation checks
xecuting Axi4 End Of Simulation checks
xecuting Axi4 End Of Simulation checks
finish called at time : 62009858 ns : File "/home/ubuntu/final_fpga/vivado/fsic
tb.sv" Line 182
un: Time (s): cpu = 00:12:34 ; elapsed = 00:17:37 . Memory (MB): peak = 3006.18
 ; gain = 0.000 ; free physical = 126 ; free virtual = 3768
 exit
NFO: xsimkernel Simulation Memory Usage: 286492 KB (Peak: 335636 KB), Simulatio
 CPU Usage: 1045690 ms
NFO: [Common 17-206] Exiting Vivado at Tue Jun 18 03:59:02 2024...
==========================================================================
                          vivado complete
==========================================================================
buntu@ubuntu2004:~/final_fpga/vivado$
```

2.FPGA validation

```
open_run: Time (s): cpu = 00:00:32 ; elapsed = 00:00:34 . Memory (MB): peak = 33
29.922 ; gain = 297.105 ; free physical = 5458 ; free virtual = 7407
# report_timing_summary -file timing_report.log
INFO: [Timing 38-91] UpdateTimingParams: Speed grade: -1, Delay Type: min_max.
INFO: [Timing 38-191] Multithreading enabled for timing update using a maximum o
f 2 CPUs
report_timing_summary: Time (s): cpu = 00:00:12 ; elapsed = 00:00:08 . Memory (M
B): peak = 3431.430 ; gain = 101.508 ; free physical = 5329 ; free virtual = 728
1
# exit
INFO: [Common 17-206] Exiting Vivado at Thu Jun 20 00:18:12 2024...
==========================================================================
                          vivado complete
==========================================================================
ubuntu@ubuntu2004:~/final_fpga/vivado$
```

```
print("waiting userdma transfer done...")
while True:
    if mmio.read(PL_USERDMA+0x10) == 0x01:
        break
print("userdma finished!!!")
```

```
waiting userdma transfer done...
userdma finished!!!
```