2023 IC Design Contest

Cell-Based IC Design Category for Graduate Level

Laser Treatment

1.Problem description

In medicine, lasers are often used for diagnosis, prevention and treatment of diseases. Using a single wavelength, in-phase and high-intensity laser beam to directly irradiate biological tissue, it can achieve functions such as thinning, vaporization, and devascularization. Laser itself contains powerful energy. Excessive use will stimulate normal cells, so it must be used with caution. This question assumes that a fixed Only two lasers can be used on a large area. Please find the positions of these two lasers to achieve the best treatment effect.

The detailed specifications of this question will be described later. Table 1 shows the function description of each input and output signal of this circuit (LASER). Each participating team Wu must complete design verification in accordance with the design specifications specified in the next section.

The competition time of this IC design competition is from 08:30 am to 20:30 pm. When the IC design competition ends, this question will be scored according to the scoring criteria in Section 3. For the convenience of grading, each participating team should refer to the requirements listed in Appendix C and attach the files required for grading.

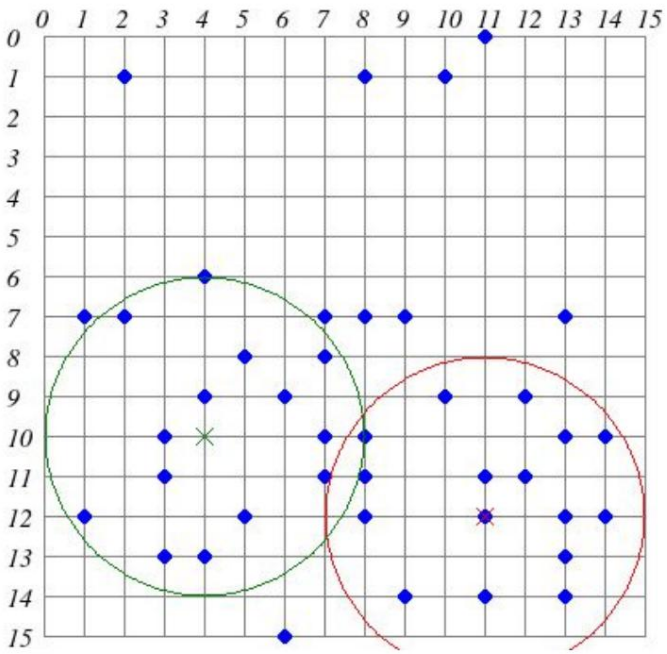


Figure 1. Schematic diagram of laser treatment

2.Design specifications

2.1 System block diagram

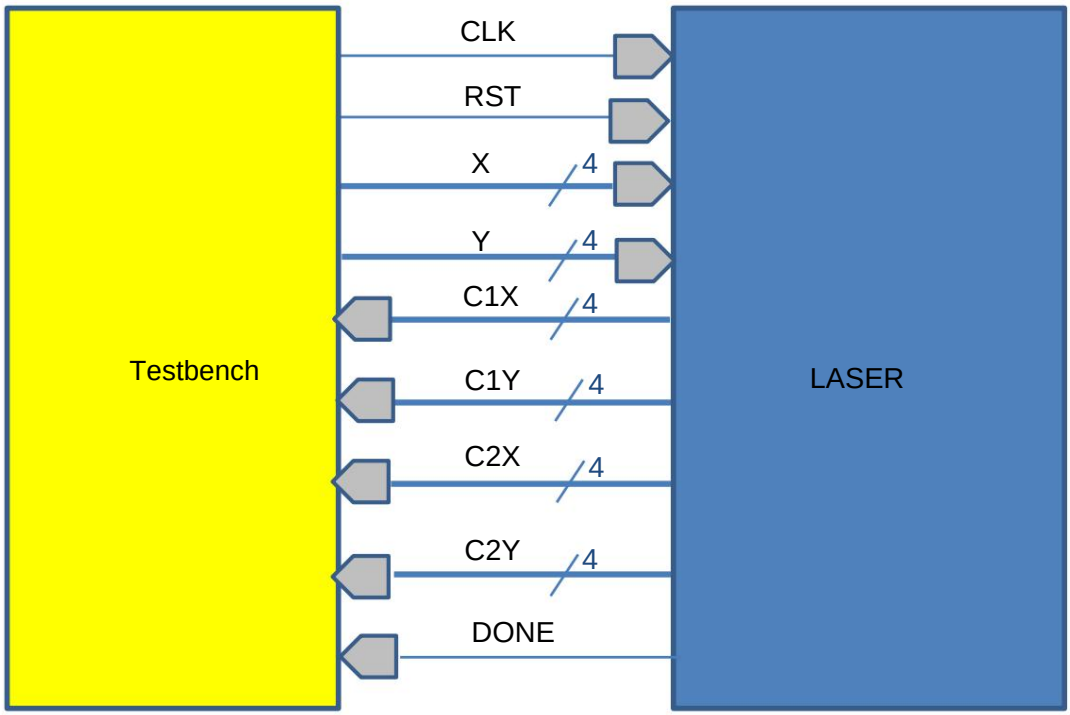


Figure 2. System block diagram

2.2 Input/output interface

Table 1. Input/output signals

Signal Name	I/O	Width	Simple Description
CLK	I	1	Clock Signal (positive edge trigger)
RST	I	1	Synchronous reset signal (active high). Provided by testbench Supply, pull high for 2 cycles and then return to low.
X	I	4	X coordinate of the target object, unsigned binary integer.
Y	I	4	Y coordinate of the target object, unsigned binary integer.
C1X	O	4	Outputs the X coordinate of the first power transmission, an unsigned binary integer.
C1Y	O	4	Outputs the Y coordinate of the first transmission, an unsigned binary integer.
C2X	O	4	Outputs the X coordinate of the second transmission, an unsigned binary integer.
C2Y	O	4	Outputs the Y coordinate of the second transmission, an unsigned binary integer.
DONE	O	1	Complete signal, testbench will capture two rounds after receiving the DONE signal The coordinates of the laser and calculate the number of objects covered.

2.3 System description

This question assumes that there are 40 fixed objects in an area of 16x16. Only two lasers can be used in this area, and the laser shape is For a circle with a radius of 4, please find the center position of the two circles so that the two circles can achieve the maximum coverage of the target object. (Figure 1)

There are 6 sets of patterns in this question. When the completion signal (DONE) is received, the next set of patterns will start to be sent.

2.3.1 Input of LASER circuit object data

The number of objects in this title is fixed at 40, which are input from port 40 coordinates are recorded for subsequent calculations.

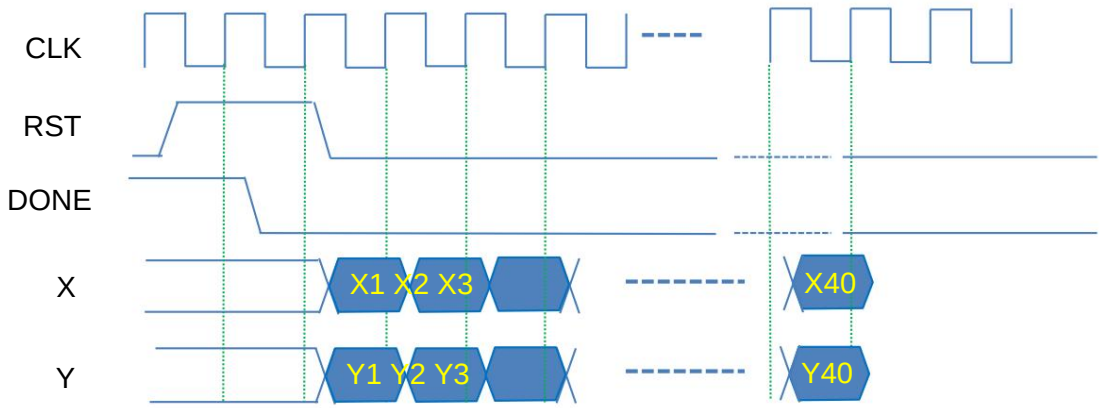


Figure 3. Pattern input waveform after reset

After LASER completes the first set of pattern calculations, it pulls the DONE signal high to indicate completion. When the DONE signal is pulled back to low Then, the second set of patterns begins to be sent.

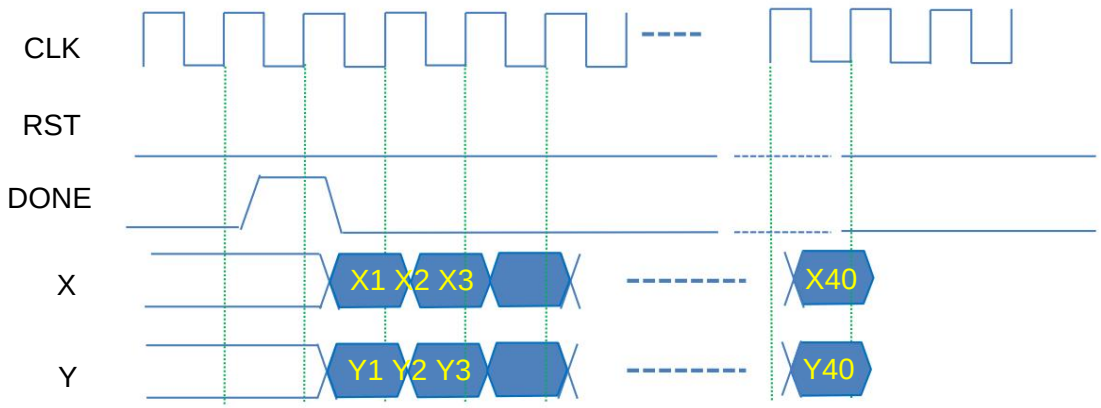


Figure 4. Pattern input waveform after DONE is pulled back to low.

2.3.2 Output of LASER circuit calculation results

The LASER circuit calculates the positions of the two circles so that the two circles can cover the maximum amount of objects. The calculation results are sent through ports C1X, C1Y, C2X, and C2Y. At the same time, the DONE signal is pulled high. Testbench will start when it receives the DONE signal. Calculate the amount of subject matter covered.

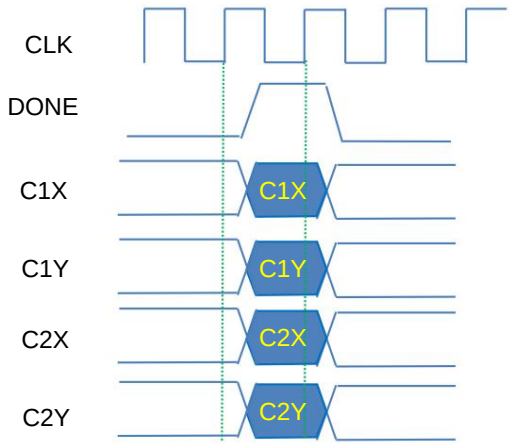


Figure 5. Output signal waveform

2.3.3 Output within a limited time

In conjunction with the radio resonance stabilization time, this question limits the maximum number of calculation cycles for each set of test samples to 50,000 cycles. If the DONE signal is not raised after this time, testbench will automatically capture the current C1X, C1Y, C2X, and C2Y contents as output. , and resend RST, and then send the next set of patterns (Figure 6).

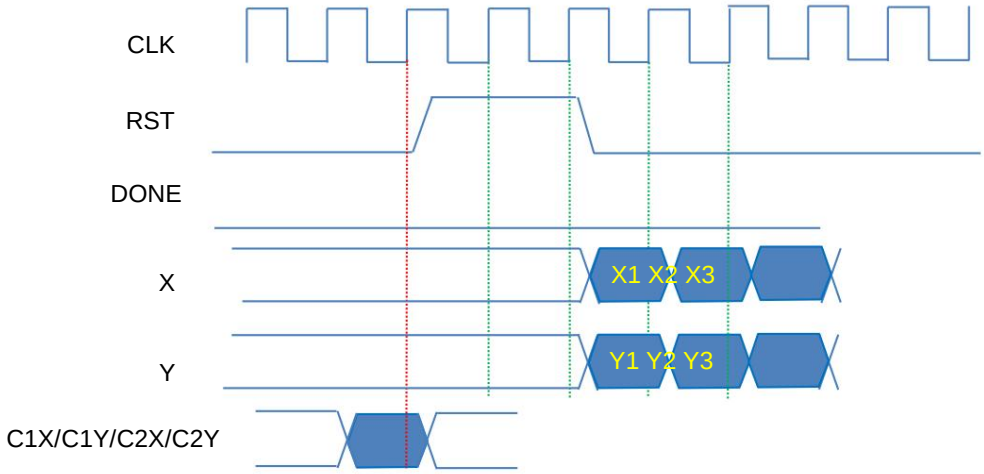


Figure 6. Forced output signal waveform when time is up

2.3.4 Limited design area

Due to system area planning, this question limits the maximum available area to 37,000um2. Please design a fast and High coverage circuitry. If the design area exceeds this size, the rating will be reduced by one level.

2.3.5 Judgment within a circle

The radius of the laser light circle is fixed at 4. Please judge whether the target is within the circle according to the distance between the target and the center of the circle. If the distance is exactly equal to 4, it is considered within the circle. If two circles cover the same target at the same time, only one object is counted.

2.3.6 Calculate the positions of two circles

The goal of this question is to pursue the maximum coverage of the subject within the two circles, and the scoring is also based on the coverage. It is too time-consuming to exhaustively enumerate the positions of two circles at the same time, but the problem can be solved iteratively. 1. Find the center position of the maximum coverage when there is only one circle. We call this circle circle one. 2. Fix the circle 1 position and find the circle 2 position to maximize the coverage. 3. Fix the circle 2 position and readjust the circle 1 position to maximize the coverage. 4. Repeat steps 2 and 3 until the results converge.

Note 1: In steps 2 and 3, when selecting the center position of the circle, there may be multiple choices, and different selection strategies will affect The speed of convergence.

Note 2. The problem with the iterative solution method is that it may fall to a local stable point, resulting in the inability to find the best solution. This situation has been eliminated in this question. All the patterns in this question can be iterated to find the best solution. Note 3: In order to avoid cheating, when performing step 1, the initial value of the uncalculated circle center can only be limited to (0,0) or three other corners. Direct initialization at other points is prohibited. Note 4:

The same set of patterns may have multiple combinations that can reach the maximum coverage, and any combination is the best solution.

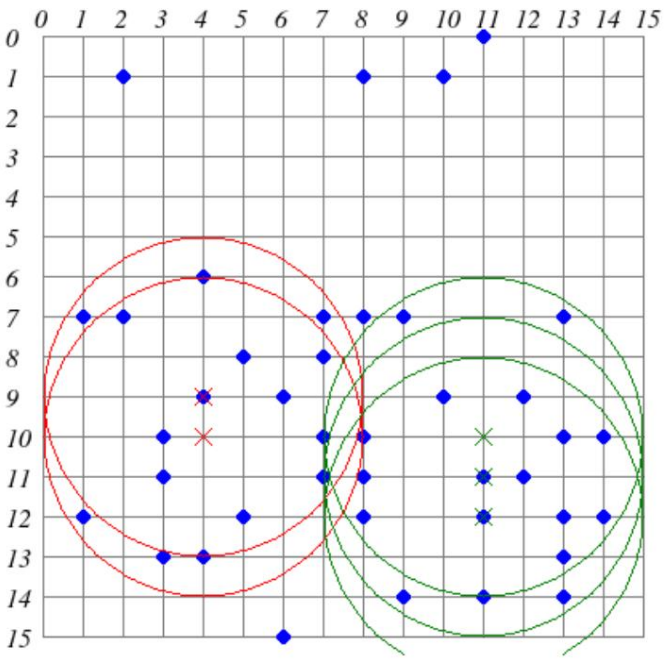


Figure 7. Different combinations can achieve optimal coverage.

3. Scoring criteria

There are three design goals.

Goal 1. A single test sample must be completed within 50,000 cycles. Goal 2. The area after synthesis must be less than 37,000um2. Goal 3. Reach the maximum total number of targets covered by six groups of patterns.

Area calculation:

Take Design compile report area as an example: dc_shell> report_area

```
Combinational area:                23327.368050
Buf/Inv area:                       2714.142565
Noncombinational area:             11067.048170
Macro/Black Box area:                  0.000000
Net Interconnect area:            423082.135437

Total cell area:                   34394.416220
Total area:                       457476.551657
```

Number of cycles and total coverage:

At the end of the simulation, the total number of simulation cycles, total coverage and optimal coverage will be listed.

```
*********************************
**    Finish Simulation        **
**    RUN CYCLE =      #####   **
**    Cover total = 170/170    **
*********************************
```

The scoring method will be divided into three levels: A, B, and C according to the degree of design completion. The ranking order is A>B>C.

This question sets the clock cycle time to 8ns, and contestants cannot adjust the clock cycle time.

ÿLevel A: Level A **conditions :** a. The area after

synthesis is less than 37,000um2 b. Under the clock cycle of

8ns, Gate-Level and RTL simulation can be completed, and the simulation results are the same.

Scoring method for Level **A :** Sort by

the total coverage of the six groups of pattern objects. Those with the same total are sorted by the number of completed simulation cycles. If the number of cycles is also the same,

they are sorted by the synthetic area.

ÿLevel B: Level B **conditions :** a. The area after

synthesis is greater than 37,000um2 b. In an environment

with a clock cycle of 8ns, Gate-Level and RTL simulations can be completed, and the simulation results are the same. Scoring method for Level **B :** Sort by the total

coverage amount of the six groups of

pattern objects. Those with the same total are sorted by the number of completed simulation cycles. If the number of cycles is also the same, they are sorted by the synthetic area.

ÿGrade C : Grade **C** conditions:

a. The synthesis has not been completed, or the Gate-Level simulation failed or the simulation cannot be completed.

Scoring method for Level **C :** RTL

simulation scoring, ranking according to the total coverage of the six groups of patterns. If the output circle center information is unknown, the

Circles are not included in the calculation.

Appendix A. Design file description

1. The table below shows the design files of each contestant provided by the organizer.

Table 2. Design file description

File name	illustrate
LASER.v	The design files used by the contestants already contain system input/output port declarations.
tb.sv	Test Bench Archives.
	Test Pattern information
img*.pattern .synopsys_dc.setup synopsys_dc.setup	Use Design Compiler to create the initial configuration file for synthesis. Participate Please modify the Search Path according to the actual placement of the Library. settings. Note: Please use worst case library when synthesizing.
LASER.sdc	Constraint files synthesized by Design Compiler. Please do not modify the contents of this file.
report.000	report file format, see Appendix C.
dc_syn.tcl	dc synthesis reference command
xrun.cmd	xrun simulation reference instructions
xrun.tcl	xrun probe command
verisium.cmd	verisium waveform debug reference instructions (with xrun)
vcs.cmd	vcs simulation reference instructions
verdi.cmd	verdi waveform debug reference instructions (with vcs)
vsim.cmd	Modelsim simulation reference instructions

2. Please use LASER.v to design the circuit in this question. Its Verilog module name and output/input port declaration are as follows:

If necessary, contestants can remove the reg declaration of the output by themselves.

```
module LASER (
input CLK,
input RST,
input [3:0] X,
input [3:0] Y,
output reg [3:0] C1X,
output reg [3:0] C1Y,
output reg [3:0] C2X,
output reg [3:0] C2Y,
output reg DONE);

endmodule
```

3. The testbench file provided in this question has several lines of define added as follows:

```
`define sdf_file " ./LASER_syn.sdf "
`ifdef SDF
    initial $sdf_annotate(`sdf_file , u_LASER) ;
`endif
```

3.1 Please modify the SDF file name according to the actual SDF file name and path before simulating. 3.2 The `ifdef SDF description in testbench allows the testbench to be suitable for both RTL simulation and post-synthesis gate-level simulation. When performing gate-level simulation, contestants need to add an extra +define+SDF parameter to the simulation command to simulate smoothly. The example is as follows: xrun tb.v LASER_syn.v -v

```
tsmc13_neg.v +define+P1 +define+SDF
```

4. The organizer will provide six sets of test samples for contestants to verify the correctness of the design.

Please use the +define+P1, +define+P2, +define+P3... parameters to switch by yourself. If definitions such as +define+P1 are not used, all test samples will be simulated.

5. When simulating a single sample, the resulting image will be drawn. The meanings of each symbol are as follows:

Symbol meaning - No
target position + target, not
covered x target, covered
X Circle center position, also the target position* Circle center
position, non-target position

Use +define+USECOLOR during simulation to add additional color display. If your simulation environment cannot accept ANSI color display, it may mess up the layout. Do not use the +USECOLOR function at this time.

Example: xrun tb.v LASER.v +define+P1+USECOLOR

```
== PATTERN img1.pattern
---- Used Cycle:      42148
---- Get Return: C1(11,12),C2( 4,10)
---- cover =  30, optimum =  30
   0 1 2 3 4 5 6 7 8 9 a b c d e f
 0 - - - - - - - - - - - + - - - -
 1 - - + - - - - - + - + - - - - -
 2 - - - - - - - - - - - - - - - -
 3 - - - - - - - - - - - - - - - -
 4 - - - - - - - - - - - - - - - -
 5 - - - - - - - - - - - - - - - -
 6 - - - - X - - - - - - - - - - -
 7 - + X - - - - + + + - - - + - -
 8 - - - - - X - X - - - - - - - -
 9 - - - - X - X - - - X - X - - -
 a - - - X * - - X X - - - - X X -
 b - - - X - - - X X - - X X - - -
 c - X - - - X - - X - - X - X X -
 d - - - X X - - - - - - - - X - -
 e - - - - - - - - - X - X - X - -
 f - - - - - - + - - - - - - - - -
```

When simulating all samples (+define+P* is not used), this figure will not be displayed.

6. Please do not design based on the content of these six groups of test samples, such as judging the pattern to be a fixed value in the design , or judging the nth pattern to directly set the output result, etc. If found, no points will be awarded. When scoring, the scores of the six groups of test samples will be adjusted

The design of filtering out tricks has come one after another.

7. When executing the first step of the iteration (Chapter 2.3.6), the initial value of the uncalculated circle center can only be limited to (0,0) or three other corners.
Stop directly initializing at other points.

8. The solution provided in the question is not the only solution. As long as it can complete the function, there is no restriction on using the method of the question.

9. The complete instructions for RTL and Gate-
level simulation are as follows:

Instructions for RTL simulation using P1 test sample

ÿUse xrun simulation instructions (xrun.cmd):

```
xrun tb.sv LASER.v +define+P1+ USECOLOR +access+r -clean -createdebugdb \ -input xrun.tcl
```

xrun.tcl content:

```
ida_probe -log
ida_probe -wave -wave_probe_args="[scope -tops] -all -depth all -memories"
run
exit
```

ÿ Use vcs simulation command (vcs.cmd):

```
vcs -R -sverilog tb.sv LASER.v +define+P1+USECOLOR +access+r +vcs+fsdbon \
+fsdb+mda +fsdbfile+LASER.fsdb
```

ÿ To use modelsim simulation, please refer to the instructions in vsim.cmd :

```
vsim -c -do vsim.cmd
```

vsim.cmd content:

```
vlib work
vlog tb.sv LASER.v +define+P1+USECOLOR
vsim work.testfixture
run-All
```

The Gate-level simulation instructions are as

follows ÿUse the xrun simulation instruction (xrun.cmd):

```
xrun tb.sv LASER_syn.v +define+ SDF +access+r -clean -createdebugdb \
-input xrun.tcl -v tsmc13_neg.v +ncmaxdelays
```

ÿUse vcs simulation command (vcs.cmd):

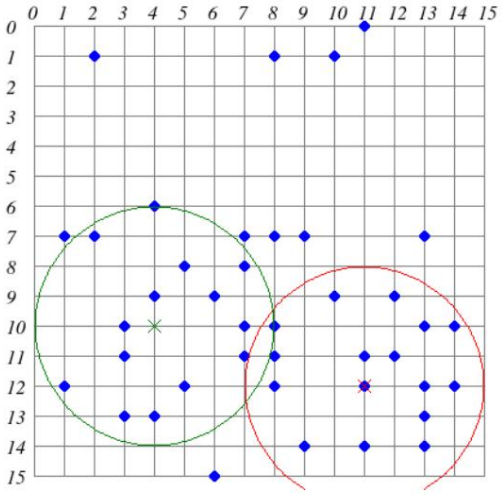
```
vcs -R -sverilog tb.sv LASER_syn.v +define+SDF +access+r +vcs+fsdbon \
+fsdb+mda +fsdbfile+LASER.fsdb -v tsmc13_neg.v +maxdelays
```

Modelsim users, please use the built-in waveforms directly for debugging.

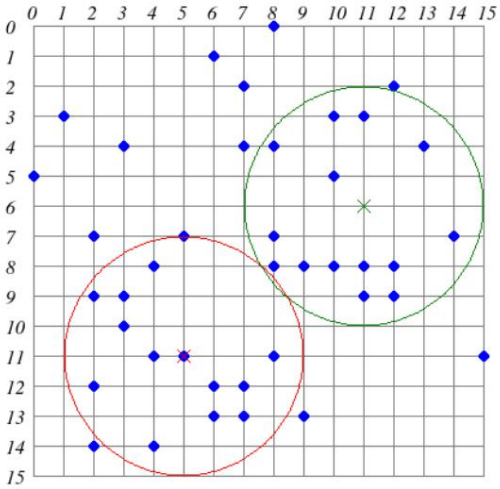
Appendix **B. Test Pattern**

The following lists 6 sets of test pattern contents: there may be multiple sets of optimal solutions, and only one set is shown in the figure.

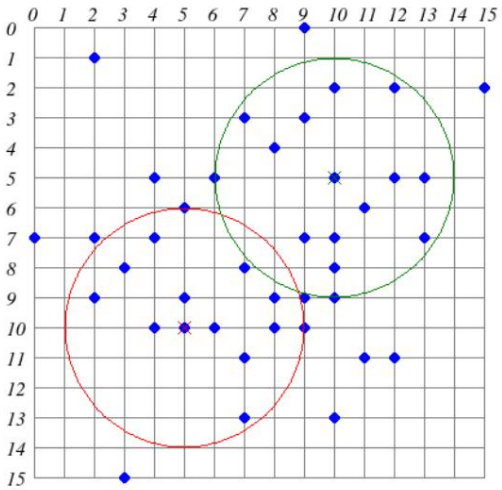
Pattern1:



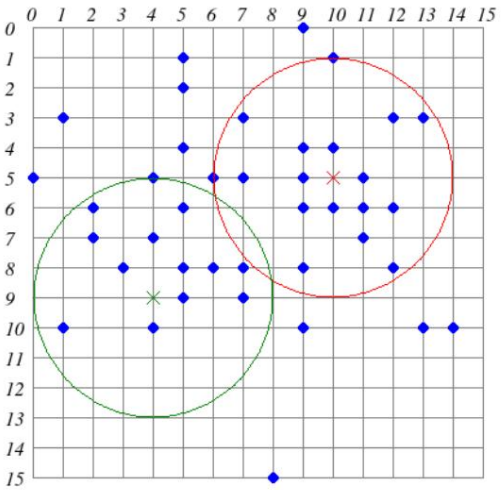
Pattern2:



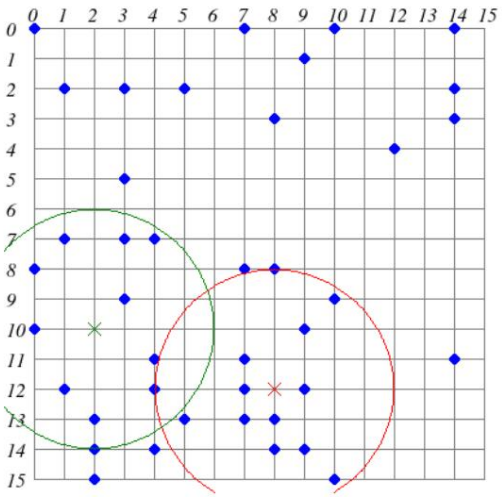
Pattern3:



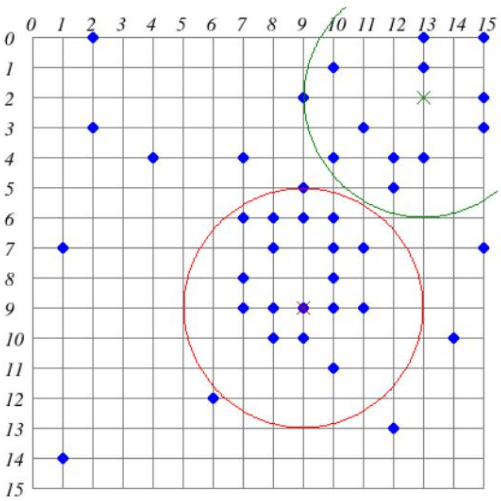
Pattern4:



Pattern5:



Pattern6:



Appendix C. Grading files

The files required for scoring can be divided into several parts:

(1) RTL design, which is the RTL code designed by each participating team for this competition . If the design is modular and has multiple design files, Please be sure to submit it together to avoid being unable to perform simulations when the judges are grading.

(2) Gate-Level design, which is the gate-level netlist generated by the synthesis software and the corresponding SDF file.

(3) report file. The participating teams must write a report.000 file according to their own design content to facilitate the organizer. Rating, the format of report.000 is as shown below. (The three serial numbers after the report file indicate the version. If the file is submitted, it will be updated. version, the file name of the new version of the report file is report.001, and so on)

Table 3. Submission of files

RTL category		
Design stage	*File*	*Description*
N/A	report.00* Design Report Form	
RTL Simulation *.v or *.vhd Verilog (or VHDL)		synthesizable RTL code
Gate-Level category		
Design stage	*File*	*Description*
Pre-layout Gate-level Simulation	*_syn.v	Verilog gate-level netlist
	*_syn.sdf Pre-layout	gate-level sdf

report file

```
FTP account:                    B23xxx, FTP account

Level:                          A/B/C Design Completion Level
cycle:                          40000, total number of simulation cycles
Synthesis area: /               30000, the cell area of the synthesized report

---RTL category---
HDL simulator:                  xrun/vcs/vsim, the name of the HDL simulator used
RTL filename:                   LASER.v, RTL file name and submodule files used...

--- Pre-layout gate-level ---
gate_level filename:            LASER_syn.v, gate-level file name
gate-level sdf filename: -----  LASER_syn.sdf, sdf file name
(annotation)---------- ----------
```

(The rest of the notes should be filled in according to the needs of each participating team)

Appendix D. File compression and organization steps

When all documents are prepared as listed in Table 3, they need to be submitted to TSRI. Please follow the steps below to submit relevant design files. Copy all files to the same folder and compress them. The steps are as follows:

1. Create a result_xxx folder. Among them, "xxx" means the submitted version. For example, "000" means the first upload; "001" means the second upload; 002 means the third upload, and so on...

   > ***mkdir result_000***

2. Copy the files required in Appendix C to the result_xxx directory. For example: > ***cp LASER.v result_000*** > ***cp LASER_syn.v result_000*** > ***cp LASER_syn.sdf result_000*** > ***cp report.000 result_000*** Please put all files in the result_000 directory and do not add other subdirectories.

3. Execute the tar command to package the result_xxx folder. The tar command example is as follows:

   > ***tar cvf result_000.tar*** After result_000 is executed, you should get the result_000.tar file. If you use the Windows system, you can compress it in zip format, except for tar and zip formats . In addition, please do not use other compression formats.

4. Use ftp to upload result_xxx.tar to the ftp server provided by TSRI. The review will be based on the last uploaded design file number. Graded assignments.

   The FTP upload needs to be switched to binary mode, and the transmission port must be set to 21 (port:21).

   The ftp account number and password have been emailed to each participant before the game. If you have any questions, please contact TSRI

   FTP site1 (Hsinchu Semiconductor Center): iccftp.tsri.org.tw (140.126.24.18)

   FTP site2 (Southern District Semiconductor Center): iccftp2.tsri.org.tw(140.110.117.9)

   EDA Cloud: 140.126.24.18

5. If you need to submit an updated version, please repeat the above steps and remember to modify the version number of the tar/zip file because you cannot modify it. or delete or overwrite previously uploaded information