## DSD Exercise

# Gauss-Seidel Iteration Machine

Speaker：Alex

Advisor：Prof. An-Yeu Wu

Date：2025/04/17

# Background

- Solving large systems of linear equations is required in many engineering simulations and scientific computing applications
- Several iterative methods are used to accelerate the computing due to their simplicity, such as Jacobi Iteration, **Gauss-Seidel Iteration**

$$\begin{bmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{N1} & \cdots & a_{NN} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_N \end{bmatrix} \Longrightarrow$$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1N}x_N &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2N}x_N &= b_2 \\ &\vdots \\ a_{N1}x_1 + a_{N2}x_2 + \cdots + a_{NN}x_N &= b_N \end{aligned}$$

# Gauss-Seidel Iteration Machine (GSIM)

- **Iterative method** to solve a linear system of equations

- **Ax = b** $\Longrightarrow$

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \qquad (1)$$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1N}x_N &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2N}x_N &= b_2 \\ &\vdots \\ a_{N1}x_1 + a_{N2}x_2 + \cdots + a_{NN}x_N &= b_N \end{aligned}$$

- Expand eq (1)

$$\begin{aligned} x_1^1 &= \frac{1}{a_{11}}(b_1 - a_{12}x_2^0 - \cdots - a_{1N}x_N^0) \\ x_2^1 &= \frac{1}{a_{22}}(b_2 - a_{21}x_1^1 - a_{23}x_3^0 - \cdots - a_{2N}x_N^0) \\ &\vdots \\ x_N^1 &= \frac{1}{a_{NN}}(b_N - a_{N1}x_1^1 - a_{N2}x_2^1 - \cdots - a_{NN-1}x_{N-1}^1) \end{aligned}$$

- Change the order

$x_3^0$: initial value of $x_3$

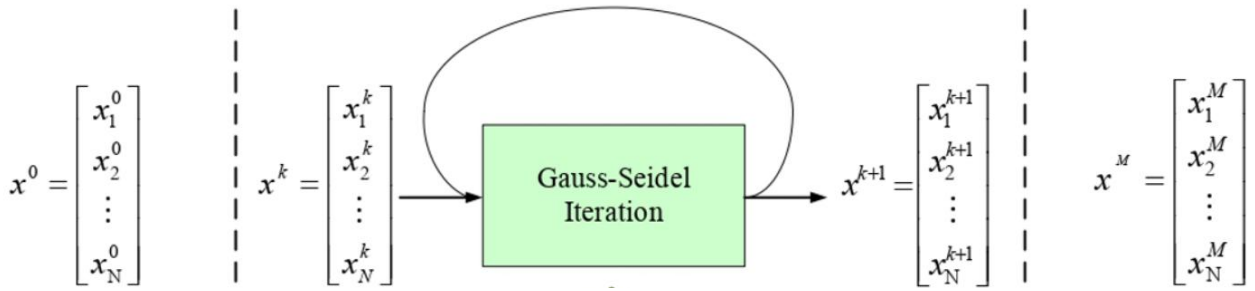
$x_2^1$: first iteration result of $x_2$

$$x_i^{k+1} = \frac{1}{a_{ii}}\left[ b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i+1}^{N} a_{ij}x_j^k \right]$$

- Final equation

# Project Description

Initialization $M$ iterations (define by yourself!) Output result **x**




- Given a fixed $16 \times 16$ matrix **A**

$$A = \begin{bmatrix}
20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 & 6 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 6 & -13 & 20 & -13 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 6 & -13 & 20
\end{bmatrix}$$

By the previous equation:

$$x_1^1 = \frac{1}{20}[b_1 + 13(x_2^0 + 0) - 6(x_3^0 + 0) + (x_4^0 + 0)]$$
$$x_2^1 = \frac{1}{20}[b_2 + 13(x_3^0 + x_1^1) - 6(x_4^0 + 0) + (x_5^0 + 0)]$$
$$x_3^1 = \frac{1}{20}[b_3 + 13(x_4^0 + x_2^1) - 6(x_5^0 + x_1^1) + (x_6^0 + 0)]$$
$$x_4^1 = \frac{1}{20}[b_4 + 13(x_5^0 + x_3^1) - 6(x_6^0 + x_2^1) + (x_7^0 + x_1^1)]$$
$$\vdots$$
$$x_{16}^1 = \frac{1}{20}[b_{16} + 13(0 + x_{15}^1) - 6(0 + x_{14}^1) + (0 + x_{13}^1)]$$

At most 7 non-zero terms one time

Only divided by 20

# Score Criteria

- 評分一：Error rate $E^2$
  - $E^2 = \sum_{i=1}^{16} \sum_{j=1}^{16} (a_{ij} x_j^k - b_i)^2$
  - Achieve Level A

| | | | | |
|---|---|---|---|---|
| A 級： | | | $E^2$ | < 0.000001 |
| B 級： | 0.000001 | ≦ | $E^2$ | < 0.000005 |
| C 級： | 0.000005 | ≦ | $E^2$ | < 0.000010 |
| D 級： | 0.000010 | ≦ | $E^2$ | < 0.000050 |
| E 級： | 0.000050 | ≦ | $E^2$ | < 0.000100 |
| F 級： | 0.000100 | ≦ | $E^2$ | < 0.001000 |
| G 級： | 0.001000 | ≦ | $E^2$ | < 0.005000 |
| H 級： | 0.005000 | ≦ | $E^2$ | < 0.010000 |
| I 級： | 0.010000 | ≦ | $E^2$ | < 0.100000 |
| J 級： | 0.100000 | ≦ | $E^2$ | < 0.300000 |
| K 級： | 0.300000 | ≦ | $E^2$ | |

- 評分二：AT score
  - $AT = area \times total\ timing$
  - Area: synthesis cell area
  - Timing: total execution time (tb1+tb2+…+tb5)

```
--------------------------------------------------

Your Score Level: A

Congratulations! GSIM's Function Successfully!

-----------------------PASS-----------------------

Simulation complete via $finish(1) at time 3734500 PS + 0
./testfixture5.v:213        #(`CYCLE/2); $finish;
ncsim> exit
```

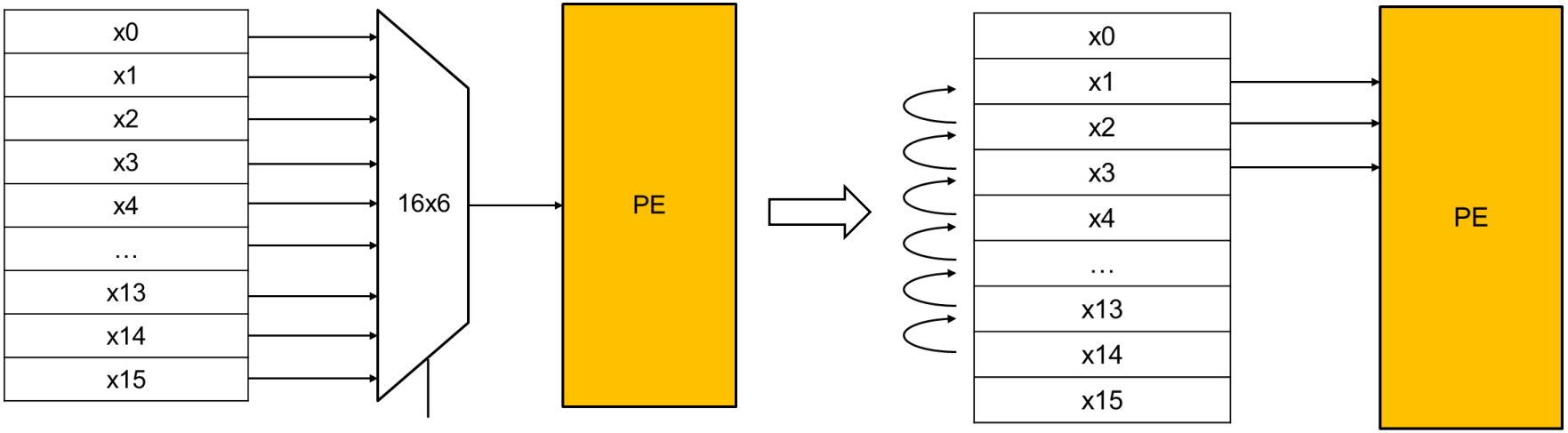
| | |
|---|---|
| Combinational area: | 3875.164193 |
| Buf/Inv area: | 434.534396 |
| Noncombinational area: | 1147.442383 |
| Macro/Black Box area: | 0.000000 |
| Net Interconnect area: | 48580.242432 |
| Total cell area: | 5022.606576 |
| Total area: | 53602.849008 |

# Design Guidelines

- Architecture level
  - Data path scheduling
  - Parallel processing (unfolding)
  - Pipelining
- Computation unit level
  - Constant multiplier, constant divider
  - Decimal analysis

# Architecture Level Optimization (1/3)

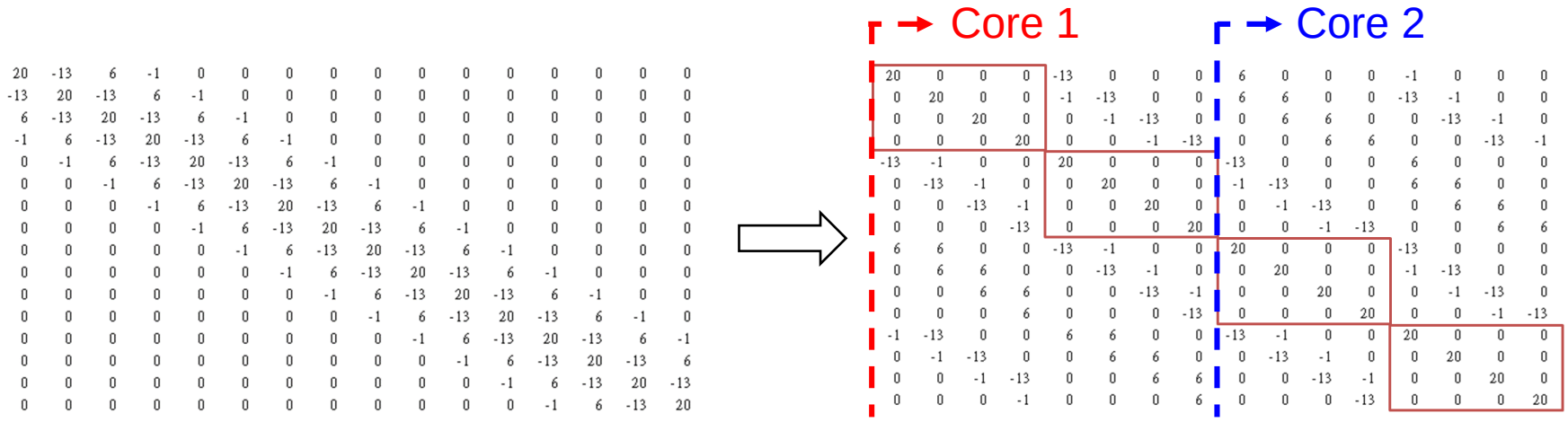
- Reading data
  - Arbitrary reading: Using several MUXs to load data
  - Structural reading: Similar computation dataflow




Scheduling by removing 6 MUXs

# Architecture Level Optimization (2/3)

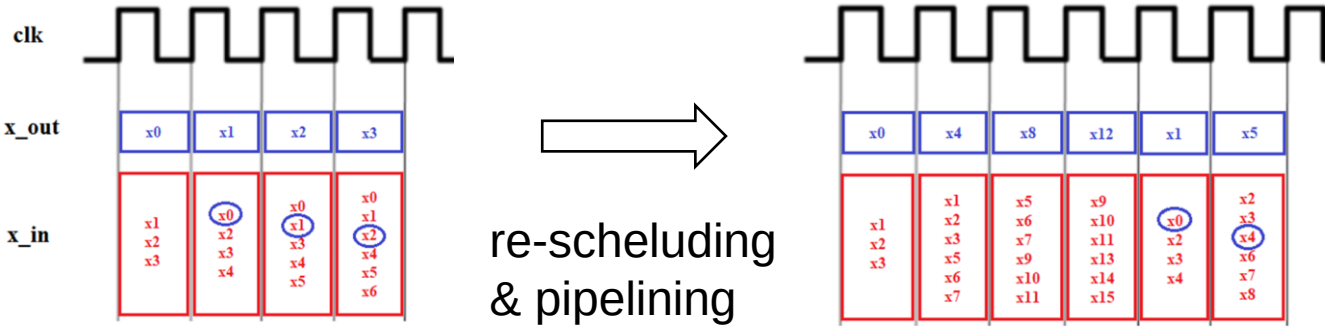
- Parallel processing (unfolding)
  - Using multi-core to compute
  - Necessity: No data dependency
- Reordering computation
  - Processing elements: 1, 5, 9, 13, …



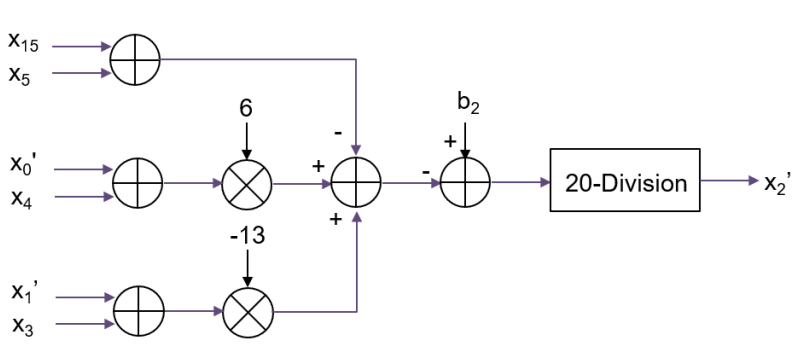

rescheduling

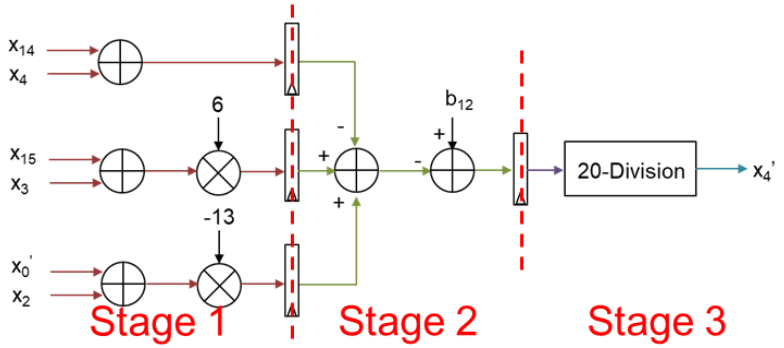
# Architecture Level Optimization (3/3)

- Pipelining
  - Divide computation into several cycles




e.g. 3-stages pipeling



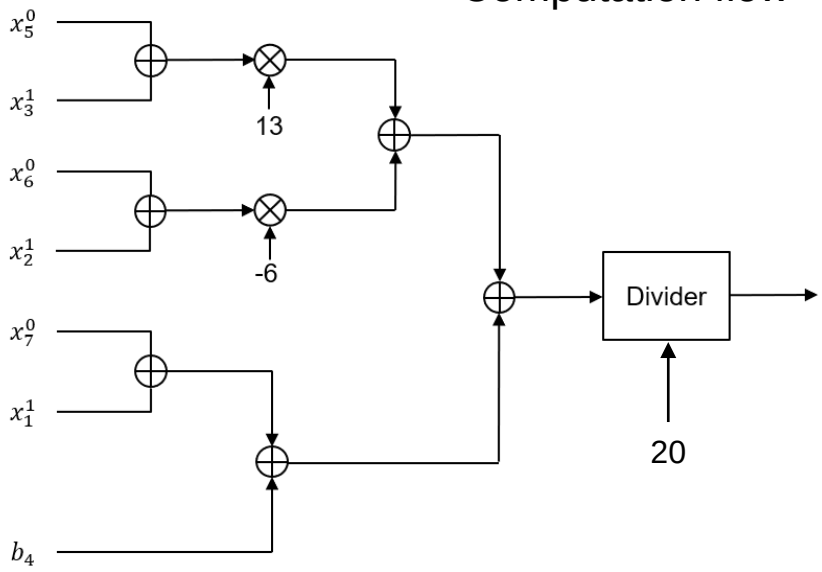



# Computation Unit Level (1/2)

1 Division  2 Multiplication

$$Ex:\ x_4^1 = \frac{1}{20}\left[b_4+13\times\left(x_5^0+x_3^1\right)+(-6)\times\left(x_6^0+x_2^1\right)+\left(x_7^0+x_1^1\right)\right]$$

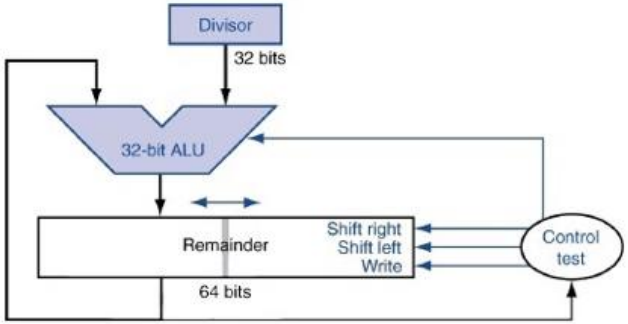
Computation flow




Division and multiplication is complicated.
It need large area and long computation time.

# Computation Unit Level (2/2)

- Multiplier
  - Power-of-two method (e.g. $6 = 110_2$, $13 = 1101_2$)
- Divider
  - Conventional (for arbitrary input): requires 32 cycles
  - Constant divider: requires 3 cycles
    - Canonic Signed Digit (CSD) code
    - For each odd integer $d$, there exists an odd integer $m$ such that $d \times m = 2^n - 1$




Need: 1 Adder, 1 shifter
Time: 32 cycles

$$\frac{1}{d} = \frac{m}{2^n - 1} = \frac{m}{2^n(1 - 2^{-n})}$$

$$= \frac{m}{2^n}(1 + 2^{-n})(1 + 2^{-2n})(1 + 2^{-4n})\ldots$$

Need: 1 Adder, 1 shifter
Time: 3 cycles

# Simulation & Synthesis

- Check “doc.pdf”
- 3 Major Things
  - RTL coding & Simulation
  - Synthesizable coding & Logic Synthesis
  - Gate-level simulation & Debugging/refinement
- Files needed for simulation
  - RTL code: GSIM.v
  - Gate-level code: GSIM _syn.v,
  - Timing info (SDF file): GSIM _syn.sdf,
  - Design library (DDC file): GSIM _syn.ddc

# Notice

- Do not fit the given test pattern, there will be hidden cases!
- Latches are not allowed in gate level code after synthesis, use Flip-flop instead.
- Negative Slack and Timing Violations are not allowed after synthesis.
- The tsmc13_neg.v file is not allowed to be downloaded! Or you may offend the copyright protected by NTU & TSRI!

# Grading Policy

- RTL (40%): Function correctness, Rank A
  - 10% for hidden tb
- Synthesis (40%):
  - 10% for hidden tb
  - 20% for passing weak baseline AT score: $1.0 \times 10^8$
  - 10% for passing strong baseline AT score: $1.0 \times 10^7$
- Ranking (15%): AT ranking. Get 0% if not passing the hidden tb
- Report (5%)

# report.pdf

1. **Simulated cycle time (ns)**
   - Gate-level simulation clock cycle

   (i.e. The cycle you passed testbench after synthesis)
2. **Area (um^2)**
   - report_area
3. **AT score**
   - $AT = area \times timing$
   - Area: total cell area
   - Timing: total execution time (tb1 + tb2 + … + tb5)
4. **Screenshot**
   - Inferred memory devices in process (※No latch should be inferred!)
5. **Difference between 0.13$\mu m$ technology and 16nm ADFP**
   - Area comparison
   - Timing comparison
   - Design methodology

| | |
|---|---|
| Combinational area: | 3875.164193 |
| Buf/Inv area: | 434.534396 |
| Noncombinational area: | 1147.442383 |
| Macro/Black Box area: | 0.000000 |
| Net Interconnect area: | 48580.242432 |
| Total cell area: | 5022.606576 |
| Total area: | 53602.849008 |

# performance.txt

1. **Simulated cycle time (ns)**
   - ❖ Gate-level simulation clock cycle

   (i.e. The cycle you passed testbench after synthesis)
2. **Area (um^2)**
   - ❖ report_area


| | |
|---|---|
| Combinational area: | 3875.164193 |
| Buf/Inv area: | 434.534396 |
| Noncombinational area: | 1147.442383 |
| Macro/Black Box area: | 0.000000 |
| Net Interconnect area: | 48580.242432 |
| Total cell area: | 5022.606576 |
| Total area: | 53602.849008 |

# Submission – NTUCOOL

- *dsdxxx-hw4/*
  - *GSIM.v*
  - *report.pdf*
- Compress all the files into one **tar.gz** file
  - File name: dsdxxx-hw4.tar.gz
  - tar -zcvf dsdxxx-hw4.tar.gz [path to dsdxxx-hw4]
  - EX: dsd001-hw4.tar.gz
- Upload the file to NTUCOOL
- Deadline: 2025/05/01 23:59 ※Late submission is not allowed

# Submission – 16nm ADFP

- *dsdxxx-hw4/*
  - *GSIM.v*
  - *GSIM _syn.v*
  - *GSIM _syn.sdf*
  - *GSIM _syn.ddc*
  - *performance.txt*
- Compress all the files into one **tar.gz** file
  - File name: dsdxxx-hw4.tar.gz
  - EX: dsd001-hw4.tar.gz
- Put the tar.gz file in your home directory
  - EX path: /nfshome/dsd029/dsd029-hw4.tar.gz
- Deadline: 2025/05/01 23:59 ※Late submission is not allowed