VLSI Physical Design HW1 Report　111062678 卓榮祥

- **Comparison Table (** core utilization：0.7, clock period：10 **)**

  The red text represents the optimal outcome.

	(congestion-driven, timing-driven)					
	(L, off)	**(L, on)**	**(M, off)**	**(M, on)**	**(H, off)**	**(H, on)**
Slack (ns)	0.051	1.016	0.114	1.246	0.114	1.246
total wire length (um)	108272.2300	103605.9850	108212.0650	102711.9700	108212.0650	102711.9700

(L, M, and H stand for Low, Medium, and High congestion effort.)

- **Explanation of the result**

Fixing congestion-driven mode allows us to observe the difference between enabling and disabling the timing-driven mode. It is evident that when the timing-driven mode is enabled, it improves slack values and reduces the total wire length, in other words, it enhances timing.

Similarly, fixing the timing-driven mode enables us to observe the difference when setting different level (Low, Medium or High) of the congestion-driven mode. It is clear that increasing the congestion-driven mode also improves slack values and reduces the total wire length, thereby enhancing timing.

- **The difference(s) between the congestion-driven placement and timing-driven placement**

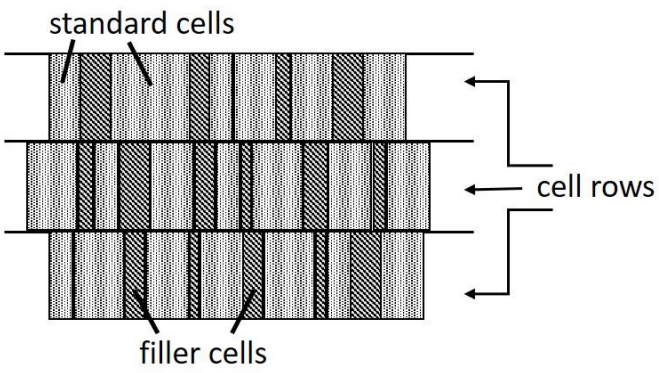
If the number of routing tracks available for routing in one particular area is less than the required routing tracks then the area said to be congested. Congestion-Driven

Placement primarily focuses on addressing congestion, which refers to areas on the circuit board with high component density or high interconnection density. During congestion driven placement, the cells (Higher cell density) which caused for congestion are spread apart. Aims to reduce or minimize congestion areas to ensure proper allocation of routing resources.

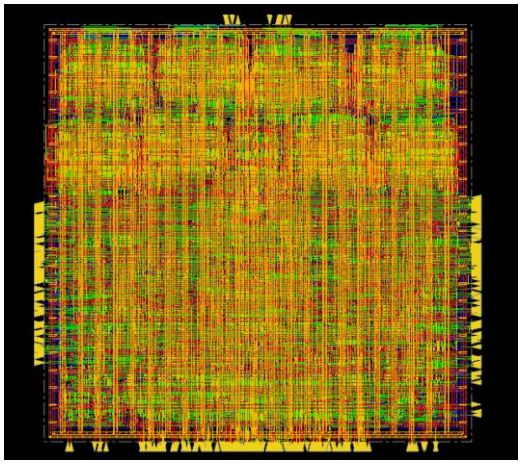
Timing-driven placement (TDP) is designed specifically targeting wires on timing critical paths. The cell in a timing critical path are placed together to reduced routing related delays. Timing-Driven Placement mainly concerned with ensuring that all components in the circuit meet predefined timing constraints. Also aims to optimize the layout to ensure that signals reach their destinations in the shortest possible time, achieving high-performance circuits.

➢ Why we insert filler cells ?

After P&R is complete, there are some open areas between logic cells that must be filled before physical verification can begin. P&R engineers typically use P&R tools to add filler cells—mostly non-functional cells that are used to fill any spaces between regular library cells to avoid planarity problems and connect power rail. They are need when the density of the required metal or layer has not meet the foundry or fab requirement.



➢ Initial setting (Baseline)

clock period	10
core utilization	0.7
timing-driven option	on
congestion-driven effort setting	Auto
Total wire length	102719.2700 um
Total area of chip	14697.883 um^2
Arrival time	8.668
slack	1.246
Final chip layout	

➢ Best result (a non-negative slack and no DRC violation).

clock period	6
core utilization	0.6
timing-driven option	On
congestion-driven effort setting	High
Total wire length	113266.4600 um
Total area of chip	16958.458 um^2

Arrival time	5.828
slack	0.001

- **Final chip layout of the best result generated by Innovus**

```
Other End Arrival Time                0.000
- Setup                               0.171
+ Phase Shift                         6.000
= Required Time                       5.829
- Arrival Time                        5.828
= Slack Time                          0.001
     Clock Rise Edge                            0.000
     + Clock Network Latency (Ideal)            0.000
     = Beginpoint Arrival Time                  0.000
```

```
#-check_same_via_cell true              # bool, default=false, user setting
 *** Starting Verify DRC (MEM: 2522.8) ***

  VERIFY DRC ..... Starting Verification
  VERIFY DRC ..... Initializing
  VERIFY DRC ..... Deleting Existing Violations
  VERIFY DRC ..... Creating Sub-Areas
  VERIFY DRC ..... Using new threading
  VERIFY DRC ..... Sub-Area: {0.000 0.000 65.880 65.880} 1 of 4
  VERIFY DRC ..... Sub-Area : 1 complete 0 Viols.
  VERIFY DRC ..... Sub-Area: {65.880 0.000 130.530 65.880} 2 of 4
  VERIFY DRC ..... Sub-Area : 2 complete 0 Viols.
  VERIFY DRC ..... Sub-Area: {0.000 65.880 65.880 129.920} 3 of 4
  VERIFY DRC ..... Sub-Area : 3 complete 0 Viols.
  VERIFY DRC ..... Sub-Area: {65.880 65.880 130.530 129.920} 4 of 4
  VERIFY DRC ..... Sub-Area : 4 complete 0 Viols.

  Verification Complete : 0 Viols.
```