ISSN 0002-306X. Proc. of the RA NAS and NPUA Ser. of tech. sc. 2020. V. LXXIII, N1

UDC 621.382.13 MICROELECTRONICS

S.S. ABAZYAN, V.A. JANPOLADOV, N.E. MAMIKONYAN

STANDARD CELL PIN ACCESS CHECKING INTEGRATION INTO TEST DESIGN VERIFICATION

An algorithm for standard cell pin access checking integration into test design verification is presented, which covers almost 65% more placement and routing cases and is almost twice slower compared to test design verification without the proposed algorithm because of checking abutment and routing cases for all cells from the standard cell library. Compared with the pin access checking flow, the algorithm is faster ~9 times but covers almost 8% less cases of placement and routing.

Keywords: pin access, test design, physical design, verification.

Introduction. In current manufacture of Integrated Circuits (ICs), standard cell libraries are most commonly used to design foundation libraries to create the digital circuit`s Place and Route. Digital IC quality, area, manufacturability, etc. are strongly affected by standard cell library cell layout. Hence library cells need to be verified and optimized to meet IC production challenges [1,2].

Standard cell library is validated and verified in different stages from cell design to silicon verification. Standard cell design flow level (Fig. 1) includes test design creation and pin access check flows for standard cell library verification [3].

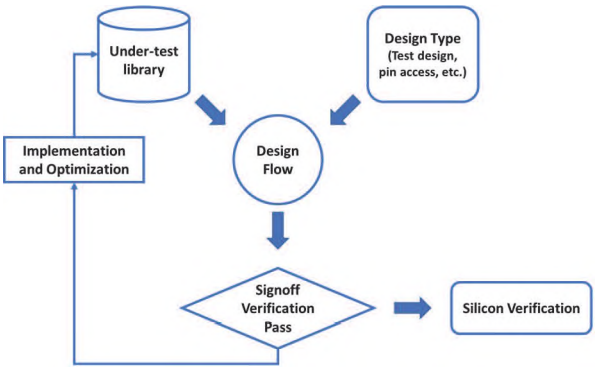

Fig. 1. The Standard cell design level general flow

Despite the fact that the above-mentioned methods cover almost all standard cell placement and routing cases and create good test environments and designs, there are several issues to be considered. Some of the issues are:

- Pin Access check run time. A big standard cell list in the pin access check can bring to big tool runtime like hours and even days and also cause huge usage of machine resources [4].
- Separate designs for each library cell for pin access check. As all cells need to be checked by themselves and with other library cells, too many designs need to be taken into consideration. This makes hard to do signoff checks and verifications on one design and requires many runs.
- Less case coverage for test design. Sometimes, to create any circuit design, not all standard cells are needed from the library. Hence some cells will be unused and unverified. Also, not all abutment and routing cases can be used for design as placement and routes are in line with the gate level netlist [5].

Standard cell test design. A simple way of standard cell library validation is the test design creation (Fig. 2). This can be done with any RTL (Register Transfer Level) description. At first RTL description of design is synthesized into Gate level description. This description involves standard library cells which the synthesizer tool decides to use, according to the design constraints that have been given. Later, the Place and Route tool uses gate level description and library cells to create a manufacturable IC layout [5].

In the next phase, DRC (Design Rule Check) and LVS (Layout Versus Schematic) checks are applied, and if there are problems coming from standard cell library cells, those cells might be updated to meet the problems' solutions.

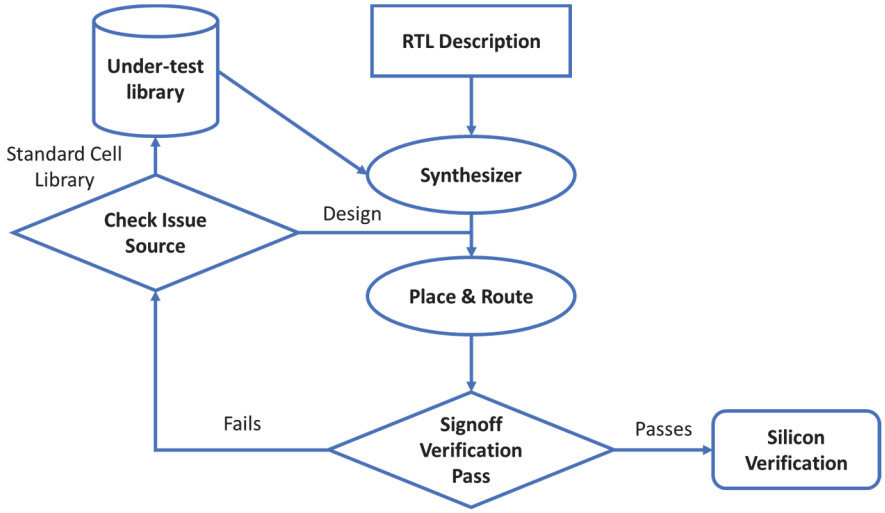

Fig. 2. RTL to the Layout design flow

Standard cell pin access checker. For latest nanotechnology processes, accessing to I/O pins of standard cells is becoming an important problem. With

more transistors used and more complex routing and placing rules considered (the same net spacing, metal min width, end-to-end spacing etc.) mentioned above challenges begin to require more attention.

It is important to consider standard cells' pin access points and ensure the maximum number of accessibility while in the development phase - to have optimal placement on the chip.

A standard cell can have different layout configuration and sizes (Fig. 3). Different sizes allow different count of tracks to go through the standard cell, which allows different count of metals to pass through the cell [5]. With technology scaling, the cell size is becoming smaller and smaller, thus making the tracks fewer. This causes challenges for the router while connecting to standard cell pins [6].

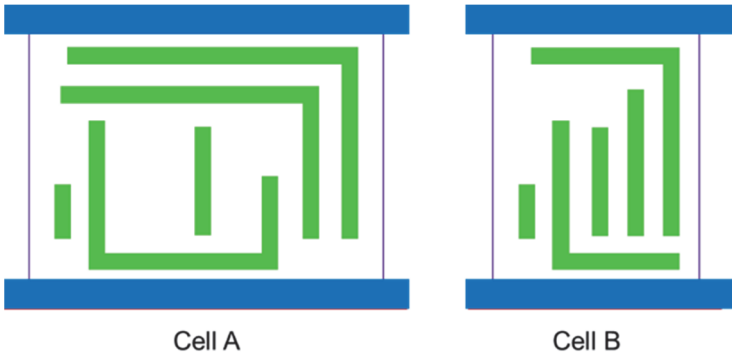

Fig. 3. Two ways of the same cell layout

For the given example (Fig. 3), cell B has better routability for pin access. Cell A has pins that are blocking each other and routing one of them can cause problems in routing the other.

The pin access checker creates a design with possible standard cell orientations of placement. In this way, a limited amount of routing space and lithography constraints makes the intercell pin access interference unavoidable, hereby causing DRC violations [7].

The main DRC issues (Fig. 4) are described below:

- DRC issue 1 – Line to line spacing issue because of the placed Via
- DRC issue 2 – Via center to center spacing issue
- DRC issue 3 – Short

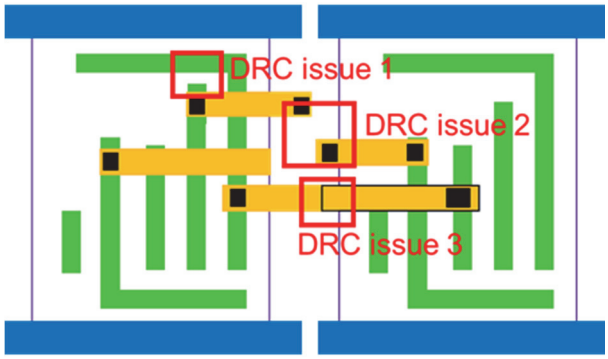

Fig. 4. The Pin access caused DRC issues

Standard cell pin access checker integration in test design. Standard cell pin access check integration in test design flow (Fig. 5) is a different approach of standard cell-based test design creation. It is meant to address runtime issues that are coming from pin access checker flow and add more cell check test cases than that in test design. Along with this, the flow allows to have one relatively big design, still having the opportunity to run Signoff DRC/LVS, etc. on it. While having one more simple RTL description in its structure, this check also combines all other unused cells. In this way, it can cover 100% cell usage from the library, hence it can cover more cell abutment and routing cases.

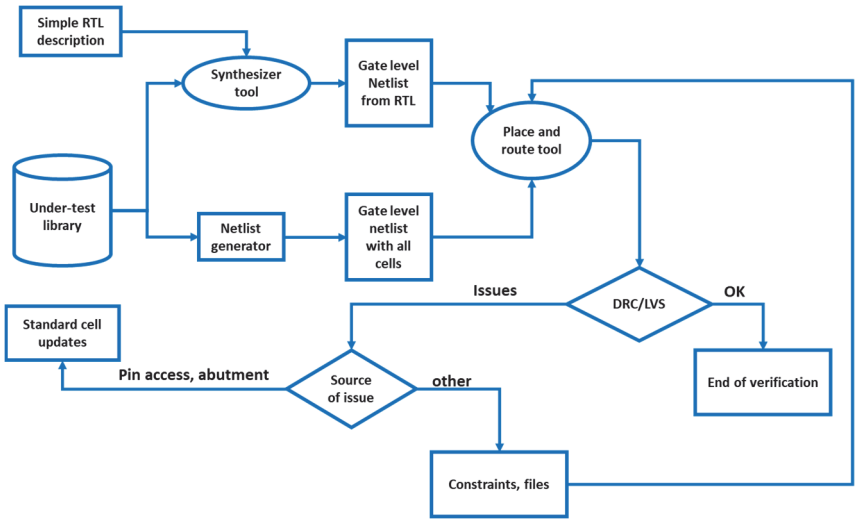

Fig. 5. Standard cell pin access checker integrated test design flow

Flow inputs are Standard Cell Library, simple RTL description netlist and technology-specific files (example: technology file).

The simple RTL description netlist is used to create a small real design which will be combined with the second netlist. This will give a chance to have a design on which the LVS check can be run.

The netlist generator is a script, that takes all the cells from the library, excluding the cells that are physical only, randomly connects half of the pins of the cell to similar nets on other cells and half of the pins connects randomly to other nets. Meanwhile, the netlist generator also marks cells as “left”, “right”, “center”, “up”, “down”, “flipped” and “last” which will be used later on place and route step to correctly place the cells and maximally duplicate the pin access checker function (Fig. 6).

```
CELLNAME_1 inst_CELLNAME_1_cntr ( .A(In[43]),
.B(In[63]), .C(Out[178]) );
 CELLNAME_2 inst_CELLNAME_2_cntr ( .A(In[40]),
.B(In[47]), .C(In[50]), .D(Out[188]));
 CELLNAME_3 inst_CELLNAME_3_cntr ( .A(In[41]),
.B(Out[198]));
 CELLNAME_4 inst_CELLNAME_4_cntr ( .A(In[48]),
.B(Out[32]));
 CELLNAME_5 inst_CELLNAME_5_cntr ( .A(In[60]),
.B(Out[78]));
 CELLNAME_6 inst_CELLNAME_6_cntr ( .A(In[100]),
.B(Out[19]), .C(Out[55]), .D(Out[44]),);
 CELLNAME_7 inst_CELLNAME_7_cntr ( .A(In[15]),
.B(Out[18]));
```

Fig. 6. Generated netlist example

All cells that are used in physical designs can have 4 placement orientations (Fig. 7).

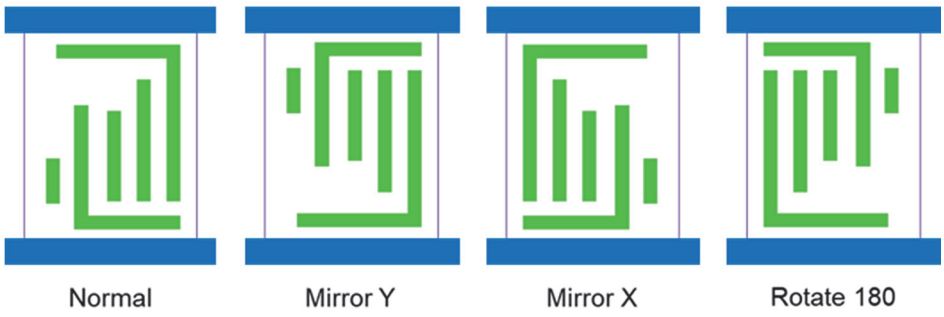

Fig. 7. Placement orientations

The proposed flow can use the placement constraints to place cells with strait and flipped orientations. This causes more case coverage (Fig. 8).

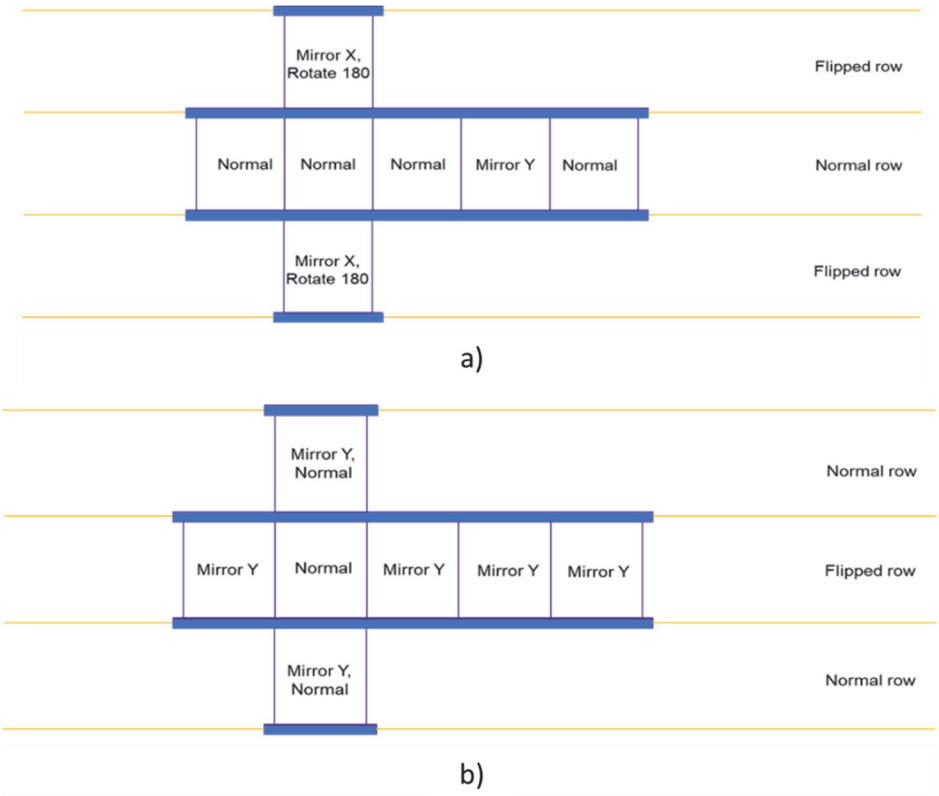

Fig. 8. The two possible placement examples: a) and b)

After successful placement, the pins of standard cells are routed as in the netlist file. While routing the cells from the generated netlist from RTL description, synthesized netlist cells also pass the routing stage making the router run process more challenging and giving more cases to check the routing.

To demonstrate the benefits of the proposed algorithm, 3 test runs were done with the standard test design flow, a separate pin access check and pin access integration in test design in case of 3 different standard library cell sets. The results were compared and analyzed to show the run time and standard cell abutment case coverage (Table).

Table

Cell count	Comparison	Design Type		
		Test design	Pin access	Demonstrated algorithm
50 Cells	Runtime	337,8 sec	5267,6 sec	675 sec
	Case Coverage	10 abut. case	48 abut. case	40 abut. case
100 Cells	Runtime	364 sec	6212,8 sec	685 sec
	Case Coverage	27 abut. case	97 abut. case	93 abut. case
200 Cells	Runtime	390 sec	7937,5 sec	705 sec
	Case Coverage	52 abut. case	195 abut case	187 abut. case

Comparison result for three types of runs

Conclusion. An algorithm with pin-access integration in test design creation flow is demonstrated for the standard cell verification process. It validates all cells of the standard cell library with real design and gives an opportunity to run signoff DRC/LVS, etc. checks on the results. The technique significantly reduces the runtime of the design creation (nearly 10X) compared with pin-access check. However, it reduces also the coverage cases. Cells are placed with normal and flipped orientations and routing is done with random priority, challenging the router and creating more ways of routing/pin tests.

REFERENCES

1. **Ricci, I. De Munari and Ciampolini P.** An evolutionary approach for standard-cell library reduction // Proc. of ACM Great Lakes Symposium on VLSI (GLSVLSI).- 2007.- P. 305-310.
2. **Agatstein W., McFaul K., and Themins P.** Validating an ASIC standard cell library// Proc. of IEEE ASIC Seminar and Exhibit. -1990.- P. 12/(6,1-6,5) .
3. Self-aligned double patterning aware pin access and standard cell layout co-optimization/**X. Xu, B. Cline, G. Yeric, et al**// ACM International Symposium on Physical Design (ISPD).- 2014.- P. 101–108.
4. **S. A. A. B. Olivier Aupoix**. Optimizing standard cell pin accessibility in 14nm FDSOI with Synopsys pin access checker/ Synopsys Users Group (SNUG).- France, 2014. - P. 1-13.
5. Self-aligned double patterning aware pin access and standard cell layout co-optimization/**X. Xu, B. Cline, G. Yeric, B. Yu, and D. Z. Pan**// IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD). – 2015. - 34(5).- P. 699–712.
6. PARR: Pin access planning and regular routing for self-aligned double patterning/ **X. Xu, B. Yu, J.-R. Gao, et al** //ACM IEEE Design Automation Conference (DAC). – 2015. - P. 28:1–28:6.

7. Standard cell layout regularity and pin access optimization considering middle-of-line / **W.Ye, B.Yu, D.Z. Pan, et al** // ACM Great Lakes Symposium on VLSI (GLSVLSI). – 2015. - P. 289–294.

Yerevan State University, Russion–Armenian University. The material is received on 02.03.2020.

Ս.Ս. ԱԲԱԶՅԱՆ, Վ.Ա. ՋԱՆՓՈԼԱԴՈՎ, Ն.Է. ՄԱՄԻԿՈՆՅԱՆ

ՍՏԱՆԴԱՐՏ ԲՋԻՋՆԵՐԻ ԵԼՈՒՍՏՆԵՐԻ ՀԱՍԱՆԵԼԻՈՒԹՅԱՆ ՍԱՏՈՒԳՄԱՆ ԻՆՏԵԳՐՈՒՄԸ ՓՈՐՁՆԱԿԱՆ ՆԱԽԱԳԾՈՒՄ

Ներկայացված է ստանդարտ բջիջների ելուստների հասանելիության ստուգման ալգորիթմի՝ փորձնական նախագծում ինտեգրման մեթոդը, որի կիրառման արդյունքում դիտարկվում են մոտ 65%-ով ավելի շատ տեղակայման և ծրագծման դեպքեր՝ համեմատած փորձնական նախագծի, բայց ընթացքի ժամանակը դանդաղում է մոտ 2 անգամ՝ բջիջների հատուկ համադրման և ծրագծման պատճառով: Համեմատած ստանդարտ բջիջների ելուստների հասանելիության ստուգման ալգորիթմի հետ՝ մշակված ալգորիթմը արագ է մոտ 9 անգամ, բայց տեղակայման և ծրագծման դեպքերի ծածկողականությունը պակաս է մոտ 8%-ով: Մշակված ալգորիթմը հնարավորություն է տալիս նախագծման ավարտին ամբողջական ստուգումներ կատարել՝ ի տարբերություն ելուստների հասանելիության ստուգման ալգորիթմի:

Առանցքային բառեր. ելուստների հասանելիություն, փորձնական նախագիծ, ֆիզիկական նախագիծ:

С.С. АБАЗЯН, В.А. ДЖАНПОЛАДОВ, Н.Э. МАМИКОНЯН

ИНТЕГРАЦИЯ ПРОВЕРКИ ДОСТУПНОСТИ КОНТАКТОВ СТАНДАРТНЫХ ЯЧЕЕК В ВЕРИФИКАЦИЮ УСТРОЙСТВА ДЛЯ ТЕСТИРОВКИ

Представлен алгоритм интеграции проверки доступности контактов стандартных ячеек в верификацию устройства для тестировки, который покрывает приблизительно на 65% больше вариантов размещения и трассировки, но примерно в два раза медленнее в сравнении с верификацией устройства для тестировки без разработанного алгоритма из-за проверки вариантов размещения и трассировки стандартных ячеек между собой. В сравнении с маршрутом проверки доступности контактов, алгоритм быстрее примерно в 9 раз и покрывает примерно на 8% меньше случаев размещения и трассировки. Разработанный алгоритм делает возможной проверку всех стандартных ячеек из используемой библиотеки.

Ключевые слова: доступность контактов, проект для тестировки, физическое проектирование.