“CuQA Tool Development”

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

RGUKT

Rajiv Gandhi University of Knowledge Technologies

R.K. VALLEY

Submitted by

PUNEETH NAKKA -- R170182

Under the Esteemed guidance
of S Shabana
RGUKT RK Valley.

DECLARATION

We hereby declare that the report of the B. Tech Major Project Work entitled

"**CuQA Tool Development**" which is being submitted to Rajiv Gandhi University of Knowledge Technologies, RK Valley, in partial fulfilment of the requirements for the award of Degree of Bachelor of Technology in Computer Science and Engineering, is a bonafide report of the work carried out by us. The material contained in this report has not been submitted to any university or institution for award of any degree.

PUNEETH NAKKA– R170182

Dept. of Computer Science and Engineering.

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

RGUKT

(A.P. Government Act 18 of 2008)
RGUKT, RK VALLEY
Department of Computer Science and Engineering

CERTIFICATE FOR PROJECT COMPLETION

This is to certify that the project entitled with **"CuQA Tool Development"** in **Intel** submitted by **PUNEETH NAKKA(R170182),** under our guidance and supervision for the partial fulfilment for the degree Bachelor of Technology in Computer Science and Engineering during the academic year 2022-2023 at RGUKT, RK VALLEY. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any University or Institute for the award of any degree or diploma.

Project Internal Guide	**Head of the Department**
Mrs. Shabana S	Mr. Satyanandaram N
Assistant Professor	HOD Of CSE
RGUKT, RK Valley	RGUKT, RK Valley

ABSTRACT

This report presents a summary of activities I was involved in during an internship at **Intel Technology India Pvt. Ltd.** from Aug 1th 2022 to May 26 2023. The activities that I was involved in **Layout Automation for Standard Cells**. **I developed CuQA tool for Lib wise and DP wise with multiprocessing Approach to improve the runtime** and I was also able to support to perform some standard testing for verification of Cells/Layout by **Coding Stand Alone Scripts like pg_track, pac_ mo, pac_m1, check_vt, qa(Quality Analysis) and few other script**. I developed **QA Dashboards** for nearly 22+ checks/Scripts, I did Enhancements for previous Scripts . I developed **Machine Learning model To predict Runtime of CuQA tool we Developed**. This internship was a perfect opportunity to apply some of the knowledge that I have learned in previous course concepts like **Multiprocessing and Multithreading** which is giving run-time Improvement for the scripts. Previously we are using **Netbatch** which is a type of Scheduling Algorithm for jobs/tasks in different Machines for running above checks for the complete Library of cells , It takes nearly or sometimes 1 day or 1 and half day to run for finising 22+ scripts, here **Disadvantage is we using maximum machines and not utilising the power of CPU, but by using multiprocessing in python I can Catch the same run-time in one Machine and by including netbatch+multiprocessing, It reduced the run-time a lot**. In addition, I was able to develop some additional soft skills like communication, team work, and flexibility. I was taught about safety with the point of interest being fire prevention & firefighting; hence, I was able to understand the different types of fire and methods for firefighting. Finally, I learned how the depot works, in different departments, and how persons from the different departments work together to achieve the depot mission.

INDEX

Table of Contents:

INDUSTRIAL WORK EXPERIENCE REPORT

Introduction:

Custom QA Tool is a cockpit which is developed to validate OA views (layout, Schematics, symbol) developed Internally . As of now 22 checks are developed, to be executed on custom libraries to ensure their quality before delivering to customers.

Helps to validate the Layout, Developed by Layout Team. Custom QA tool can support both for DP flow and Library Wise Flow, Here QA Dashboards are generated at the end of the Flow for validating the Developed Layouts.

Purpose:

The purpose of this document is to show what I did in my Internship Period at **Intel Technology India Pvt. Ltd**. It also focuses on various key features, the Tools, Scripts, QA Dashboard and ML Model to predict Runtime of CuQA Tool.

CuQA Vision:

The CuQA vision is to Facilitate Layout Team to verify the Layouts what they developed and to check whether layout are following certain DRC rules and regulations are not.

Technologies:

- Machine Learning for Prediction CuQA Runtime
- Python
- Scikit-learn
- Multiprocessing and Multithreading Concepts
- Teras
- TensorFlow
- PyTorch

CuQA Tool

What is CuQA:

- Custom QA Tool is a cockpit which is developed to validate OA views (layout, Schematics, symbol) developed Internally .
- As of now 22 checks are developed, to be executed on custom libraries to ensure their quality before delivering to customers.

Versions Of CuQA Flow:

1. **DP Flow**
2. **Library Flow**

DP Flow:

It is a collection of Libraries.

It launches multiple libraries parallelly using xterms.

Before Releasing the Dp to customers we use it.

Library Flow:

It launches a single libraries at a time.

Useful in Developing Phase.

Overview Of Layout Automation:

Layout automation for standard cells refers to the use of software tools like **Virtuoso and Calibredrv** to automate the creation and modification of standard cell layouts for integrated circuits (ICs). Standard cells are pre-designed and pre-verified functional blocks that are used to build complex digital circuits quickly and efficiently.

Multiprocessing and Multithreading Approach in CuQA Tool:

Previously we are using Net batch Version of CuQA Flow

Disadvantages:

- Number of machine usage is high.
- Net Batch Crunch happens if machines are not available.
- Proper usage of Resources like Ram, Memory and Cpu are not taking Properly

To Overcome above issue we Developed Multiprocessing and Multithreading Approach including CalibreDrv as a Backend Tool.

Methodology

Design of CuQA Tool:

CuQA is a tool designed to automate the testing of libraries Developed By Layout Team. The CuQA tool is designed to run a set of checks defined in Runfile on the libraries and report any errors present in the Cell.

There are Three Ways to run the script

- Cell Wise
- Cell List Wise
- Parallel Mode

The tool is built using Python and includes a suite of test cases that cover a wide range of DRC Rules to be followed when Developing Layouts. The CuQA tool is also designed to be extensible, allowing user to add their own checks in Runfile and customize the tool for their specific needs.

Checks are Generic there are adaptable according to config file required for the script. They support Various Technologies like

- **N3E**
- **N3**
- **N5(**synopsis**)**
- **N6**
- **Samsung C14(ARM)**

Implementation of CuQA Tool:

The CuQA tool is implemented using Python and VLSI Development tools like Calibre Virtuoso and ML framework for Runtime Prediction .The tool requires a set of input files like

- Environmental file
  - Technology Dependent data will be present
- Runfile contain list of script and the mode which they need to run
- Cell List file
- Library name
- Check names
- Type of Flow(netbatch/multiprocess).

The CuQA tool is also designed to generate detailed reports on the results of each test, making it easy for Layout Team to identify and diagnose any issues that arise.

Standard Testing and QA Dashboards:

After Running the CuQA Flow and QA dashboards (Excel Sheet) will be generated when using -m option in CuQA Flow . It is used to track the progress and results of Layout and quality assurance activities. These dashboards typically include PASS/FAIL/ABORTED/NA as status for checks, and test results, and bug reports can be seen in separate sheets for each checks. The dashboards are designed to provide Testing and QA teams with real-time visibility into the status of Layout activities and help identify issues and trends quickly.

Machine Learning Model:

A machine learning model is used to Predict the Runtime of an CuQA tool. We used neural network to Predict Runtime of an CuQA Tool.

Machine learning models is typically trained using large datasets and require significant computational resources. The output of a machine learning model is a of prediction of Runtime based on the input data like

- Number of Checks
- Type of Checks
- Number of cells
- Number of Cores used
- Memory Given
- Total Checks = Number of cells * number of checks given at runtime ,

These are some input given to predict the Runtime Time of ongoing Flow by using the Model.

Checks Developed by me:

- **QA**
  - **QA**
  - **MJ_Prop**
  - **M2_Pin**
  - **QA1**
  - **Label Extraction check**
- **Pgt**
- **Pac_m0**
- **Pac_m1**
- **Check_vt**

Results and Analysis

.

Runtime Improvement:

The runtime improvement is the measure of the decrease in the time required to complete a particular task or process as a result of implementing a new tool or method. For example, the CuQA tool implemented using Multiprocessing Approach Given an Improved. This runtime improvement is achieved by running checks parallelly and by using Light Wight Tools like Calibredrv other than Virtuoso.

Finally ,it reduced the amount of amount of time taken to Comple the Flow.

Effectiveness of QA Dashboards:

The effectiveness of QA dashboards can be measured in several ways. One key metric is the ability of the dashboard to provide real-time visibility into the status of Layout Testing and quality assurance activities. This visibility allows Layout Validating Team and QA teams to identify issues and trends quickly and take action to address them. Another metric is the accuracy of the data presented in the dashboard. The data should be accurate, up-to-date, and presented in a clear and actionable format. Additionally, the dashboard should provide insights into the root causes of issues and enable QA teams to take corrective action to prevent similar issues from occurring in the future.

Machine Learning Model Accuracy:

The accuracy of a machine learning model is a measure of its ability to make correct predictions or decisions based on input data. This accuracy is typically measured as a percentage, with higher percentages indicating greater accuracyThe accuracy of a machine learning model can be influenced by a variety of factors, including the quality and quantity of training data, the complexity of the model, and the performance of the underlying hardware.

In our CuQA tool we have High accuracy for predicting Runtime Even such datapoint is not at all present in the training data.

Challenges and Solutions

NetBatch Scheduling Algorithm:

The Netbatch scheduling algorithm is a type of job scheduling algorithm that is used to optimize the use of resources in a computer system. One of the challenges associated with the Netbatch scheduling algorithm is that it can be difficult to determine the optimal settings for the algorithm for a given workload. This can lead to suboptimal resource allocation and longer processing times. One solution to this challenge is to use machine learning algorithms to automatically optimize the settings of the NetBatch scheduling algorithm based on the characteristics of the workload.

To over Netbatch issue we developed Multiprocessing version of CuQA Flow which given better Runtime.

Integration with Existing Technologies:

Integrating new Technologies like N3E, N3, S14RF etc into an existing workflow or toolchain can be challenging, particularly when the new technologies requires significant changes to the existing system.

With Generic Version of CuQA flow we are able to integrate with any Technology.

Debugging and Troubleshooting:

Debugging and troubleshooting are critical tasks in any Software development project, and they can be particularly challenging in complex systems such as layout automation tools. One solution to this challenge is to use a combination of automated and manual debugging and troubleshooting techniques. Automated techniques such as log analysis and error reporting can help identify issues quickly, while manual techniques such as code inspection and unit testing can help diagnose and fix more complex issues. Additionally, building in robust error handling and reporting capabilities can make it easier to diagnose and fix issues that arise during the development process.

Appendices

Screenshots of CuQA Tool and QA Dashboards:

Custom QA Tool is a cockpit which is developed to validate OA views (layout, Schematics, symbol).

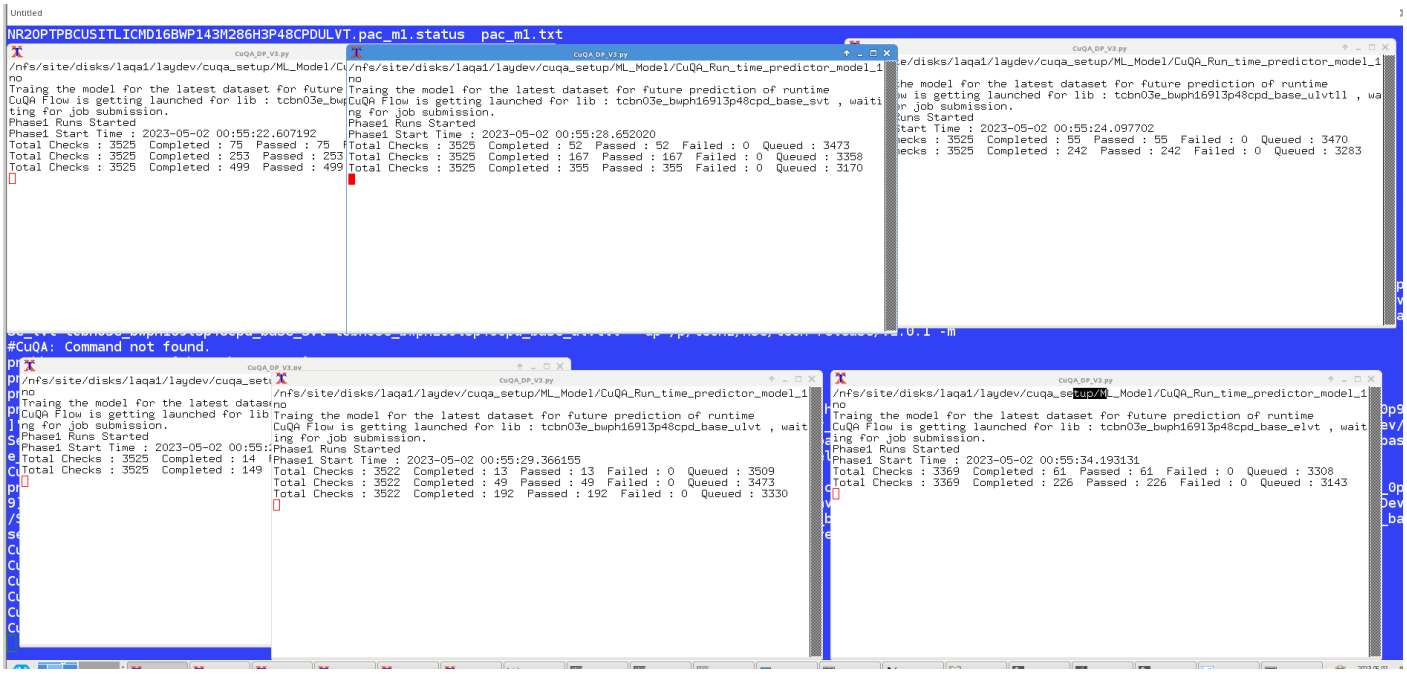
Library Flow:

```
pnakka@scce093104 : [cheetah 2022.09]> $CuQA -chk "[gds cdl oaout oasc qa antdrc tc_drc sad  pac_m0 pac_m1 pgt check_vt arch_drc_chk boundarydrc lvs plvs arraydrc idrc bou
ndarydrc_0p9 gvc gve]" -e /nfs/site/disks/laqa1/laydev/cuqa_setup/BE_SETUP_143.env -gr /nfs/site/disks/laqa1/laydev/cuqa_setup/N3E_143_runfile -mpEnable -lib TSMC_H286_Bas
e_Rev -l /nfs/site/disks/laqa1/laydev/n3e_143/layout/pnakka/test6/cells
Predicted Runtime for Completing Given Checks: 348.44955
CuQA Flow is getting launched , waiting for job submission.
Phase1 Runs Started
Phase1 Start Time : 2023-04-28 00:53:28.743207
Total Checks : 6  Completed : 2  Passed : 2  Failed : 0  Queued : 4
Total Checks : 6  Completed : 4  Passed : 4  Failed : 0  Queued : 2
Total Checks : 6  Completed : 6  Passed : 6  Failed : 0  Queued : 0
Total checks  : 6 Total checks Passed : 6 Total checks Failed : 0  Total checks aborted : 0

Run data moved to temp_dir : /tmp/pnakka/Layout_Checker/2023-04-2800:53:25.436240
Phase1 data generation completed
Phase1 Elapsed time:141.18
```

```
Preparing the setup for Phase 2 runs . Wait.............

Calculating the Licenses Available.
Available Licences : 1500
Number of Process Acquired :42
Phase2 runs are started
Phase2 Start Time :  2023-04-28 00:42:58.827775
Total Checks : 36  Completed : 2  Passed : 1  Failed : 1  Queued : 34
Total Checks : 36  Completed : 4  Passed : 3  Failed : 1  Queued : 32
Total Checks : 36  Completed : 6  Passed : 5  Failed : 1  Queued : 30
Total Checks : 36  Completed : 8  Passed : 5  Failed : 3  Queued : 28
Total Checks : 36  Completed : 10  Passed : 7  Failed : 3  Queued : 26
Total Checks : 36  Completed : 12  Passed : 7  Failed : 5  Queued : 24
Total Checks : 36  Completed : 14  Passed : 9  Failed : 5  Queued : 22
Total Checks : 36  Completed : 15  Passed : 10  Failed : 5  Queued : 21
Total Checks : 36  Completed : 16  Passed : 10  Failed : 6  Queued : 20
Total Checks : 36  Completed : 18  Passed : 11  Failed : 7  Queued : 18
Total Checks : 36  Completed : 19  Passed : 12  Failed : 7  Queued : 17
Total Checks : 36  Completed : 20  Passed : 12  Failed : 8  Queued : 16
Total Checks : 36  Completed : 21  Passed : 13  Failed : 8  Queued : 15
Total Checks : 36  Completed : 22  Passed : 14  Failed : 8  Queued : 14
Total Checks : 36  Completed : 23  Passed : 15  Failed : 8  Queued : 13
Total Checks : 36  Completed : 24  Passed : 16  Failed : 8  Queued : 12
Total Checks : 36  Completed : 26  Passed : 18  Failed : 8  Queued : 10
Total Checks : 36  Completed : 28  Passed : 20  Failed : 8  Queued : 8
Total Checks : 36  Completed : 30  Passed : 22  Failed : 8  Queued : 6
Total Checks : 36  Completed : 32  Passed : 24  Failed : 8  Queued : 4
Total Checks : 36  Completed : 34  Passed : 26  Failed : 8  Queued : 2
Total Checks : 36  Completed : 35  Passed : 27  Failed : 8  Queued : 1
Total Checks : 36  Completed : 36  Passed : 28  Failed : 8  Queued : 0
Phase2 Data generation is completed
Phase2 Run End time :  2023-04-28 00:46:54.973133
Phase2 Elapsed time:238.64
Runs completed
```

DP Flow:

Here all libraries present in a DP will get executed parallely.




```
onakka@scce093104 : [cheetah 2022.09]> $CuQA -chk "[gds cdl oaout oasc qa antdrc tc_drc sad cac pgt check
9]" -e /nfs/site/disks/laqa1/laydev/n3e_143/layout/adhamimo/CuQA_PV/CuQA_Lib_Enable/Test/BE_SETUP.env_lib
/Setup/N3E_143_runfile_v3 -mpEnable -lib "tcbn03e_bwph169l3p48cpd_base_elvt tcbn03e_bwph169l3p48cpd_base_
se_lvt tcbn03e_bwph169l3p48cpd_base_svt tcbn03e_bwph169l3p48cpd_base_ulvtll" -dp /p/tech1/n3e/tech-releas
CuQA Flow is getting launched for Lib name : tcbn03e_bwph169l3p48cpd_base_elvt
CuQA Flow is getting launched for Lib name : tcbn03e_bwph169l3p48cpd_base_lvtll
CuQA Flow is getting launched for Lib name : tcbn03e_bwph169l3p48cpd_base_ulvt
CuQA Flow is getting launched for Lib name : tcbn03e_bwph169l3p48cpd_base_lvt
CuQA Flow is getting launched for Lib name : tcbn03e_bwph169l3p48cpd_base_svt
CuQA Flow is getting launched for Lib name : tcbn03e_bwph169l3p48cpd_base_ulvtll
CuQA Flow is completed for Lib name : tcbn03e_bwph169l3p48cpd_base_svt
CuQA Flow is completed for Lib name : tcbn03e_bwph169l3p48cpd_base_ulvtll
CuQA Flow is completed for Lib name : tcbn03e_bwph169l3p48cpd_base_elvt
CuQA Flow is completed for Lib name : tcbn03e_bwph169l3p48cpd_base_lvtll
CuQA Flow is completed for Lib name : tcbn03e_bwph169l3p48cpd_base_ulvt
CuQA Flow is completed for Lib name : tcbn03e_bwph169l3p48cpd_base_lvt
CuQA Flow for all libraries are completed
Elapsed Time :185.43
```

Scripts I Developed Reports:

PAC_M0:

- Used to check whether M0 pins are accessible by Routers are not for **power supply.**

```
#fileName: NR2OPTPBCUSITLICMD16BWP143M286H3P48CPDULVT.pac_m0.rep
#version: v1.0
#runTime: 2023-04-27 05:48:04.940819
#cell name: NR2OPTPBCUSITLICMD16BWP143M286H3P48CPDULVT
#check: Cell Accessibility Check

Total Tracks: 28
Total Tracks Count Having M1 Layer = 7
Track Positions Having M1 Layer = 2 10 15 18 20 23 26
Total Tracks Count Having M1 Blocakges = 0
Track Positions Having M1 Blockages = 0

Initial Pin information :
        ZN,A2 pins are in M1 Drawing
        A1 pins are in M0 Drawing

Detail track information of each pin :
        A1 pin is spread from 1 to 26 tracks

After Post Processing of VIAO Blockages
        A1 pin is spread from 1 to 4 tracks

After Post Processing of M1 Layer &  Blockages (M1 and VIA0)
        A1 is having 3 track   1,3,4

All pins are having more than 1 track ---

PG :1-9-17-25 : A1 = 2 (3 4); ---->ACCESSIBLE
PG :2-10-18-26 : --->NOT ACCESSIBLE
PG :3-11-19-27 : A1 = 2 (1 4); ---->ACCESSIBLE
PG :4-12-20-28 : --->NOT ACCESSIBLE
PG :5-13-21 : A1 = 3 (1 3 4); ---->ACCESSIBLE
PG :6-14-22 : A1 = 3 (1 3 4); ---->ACCESSIBLE
PG :7-15-23 : --->NOT ACCESSIBLE
PG :8-16-24 : A1 = 3 (1 3 4); ---->ACCESSIBLE

Out of 8 PG 5 is accessible.
Out of 8 PG 5 is completely accessible with all pins having atleast 2 tracks accessible.

PASS
```

Check vt:

- Check whether Layout is following VT rules or not.

```
#fileName: AOI21OPT3FD9BWP143M286H3P48CPDULVT.check_vt.rep
#version: v1.0
#runTime: 2023-02-09 21:09:06.639692
#cell name: AOI21OPT3FD9BWP143M286H3P48CPDULVT
#check: To check proper vt layers present or not

INFO:Cell AOI21OPT3FD9BWP143M286H3P48CPDULVT , VT layer defined in config VTUL_N is present in cellview
INFO:Cell AOI21OPT3FD9BWP143M286H3P48CPDULVT , VT layer defined in config VTUL_P is present in cellview


PASS
```

```
#fileName: CKLVLHLBUFFSNKCWBALOPT1V8ITLICMD8BWP143M286H3P48CPDULVT.check_vt.rep
#version: v1.0
#runTime: 2023-02-13 05:18:47.363959
#cell name: CKLVLHLBUFFSNKCWBALOPT1V8ITLICMD8BWP143M286H3P48CPDULVT
#check: To check proper vt layers present or not

INFO:Cell CKLVLHLBUFFSNKCWBALOPT1V8ITLICMD8BWP143M286H3P48CPDULVT , VT layer defined in config VTUL_N is present in cellview
INFO:Cell CKLVLHLBUFFSNKCWBALOPT1V8ITLICMD8BWP143M286H3P48CPDULVT , VT layer defined in config VTUL_P is present in cellview
ERROR:Cell CKLVLHLBUFFSNKCWBALOPT1V8ITLICMD8BWP143M286H3P48CPDULVT , VT layer not defined in config VTL_N but present in cellview


FAIL
```

PAC M1:

- Used to check whether M1 pins are accessible by Routers are not for power supply.

```
#fileName: NR2OPTPBCUSITLICMD16BWP143M286H3P48CPDULVT.pac_m1.rep
#version: v1.0
#runTime: 2023-04-27 05:47:51.253140
#cell name: NR2OPTPBCUSITLICMD16BWP143M286H3P48CPDULVT
#check: M2_Pin Accessibility Check

Height of cell : 0.286
Total M2 Tracks: 11
Total M2 Tracks Available: 9
M2 Tracks Available Positions: 1 2 3 4 6 7 8 9 10
Total Tracks Count Having M2 Layer : 0
Track Positions Having M2 Layer : nil

Initial Pin information :
        A2 ZN ZN ZN ZN are in M1 Drawing
        VDD A1 VSS are in M0 Drawing

Extending the Heights of M1 Pins to its max possible limits :

Modified Heights of M1 Pins:

        ZN-1.21-0.005-1.238-0.271
        ZN-0.442-0.005-0.47-0.271
        ZN-0.826-0.005-0.854-0.271
        A2-1.066-0.005-1.094-0.271
        ZN-0.058-0.005-0.086-0.271


Detail track information of each M1 pin :
        ZN pin is spread from 1 to 10 tracks
        ZN pin is spread from 1 to 10 tracks
        ZN pin is spread from 1 to 10 tracks
        A2 pin is spread from 1 to 10 tracks
        ZN pin is spread from 1 to 10 tracks

After Post Processing of M2 Layer :
        ZN is having 9 track : Track 1-2,3,4,6,7,8,9,10
        ZN is having 9 track : Track 1-2,3,4,6,7,8,9,10
        ZN is having 8 track : Track 2-1,3,4,6,7,8,9,10
        A2 is having 8 track : Track 2-1,3,4,6,7,8,9,10
        ZN is having 7 track : Track 3-1,2,4,6,7,8,9,10

Refer the modified gds with extended pin heigths :NR2OPTPBCUSITLICMD16BWP143M286H3P48CPDULVT/pac_m1/NR2OPTPBCUSITLICMD16BWP143M286H3P48CPDULVT_mod.gds
A2 = 7(1 4 6 7 8 9 10) ZN = 7(1 4 6 7 8 9 10) ZN = 7(2 4 6 7 8 9 10) ZN = 7(2 4 6 7 8 9 10) ZN = 7(3 4 6 7 8 9 10) ---->ACCESSIBLE


PASS

```

PGT:

- Pgt check is used to verify pg tracks accessible.

```
#fileName: NR2OPTPBCUSITLICMD16BWP143M286H3P48CPDULVT.pgt.rep
#version: v1.0
#runTime: 2023-04-27 05:47:41.163836
#cell name: NR2OPTPBCUSITLICMD16BWP143M286H3P48CPDULVT
#check: PG Track Accessibility Check

Total Tracks: 28
Total Tracks Count Having M1 Layer = 7
Track Positions Having M1 Layer = 2 10 15 18 20 23 26

PG :1-9-17-25 : --->ACCESSIBLE
PG :2-10-18-26 : --->NOT ACCESSIBLE
PG :3-11-19-27 : --->ACCESSIBLE
PG :4-12-20-28 : --->NOT ACCESSIBLE
PG :5-13-21 : --->ACCESSIBLE
PG :6-14-22 : --->ACCESSIBLE
PG :7-15-23 : --->NOT ACCESSIBLE
PG :8-16-24 : --->ACCESSIBLE

Out of 8 PG 5 is completely accessible

PASS
```

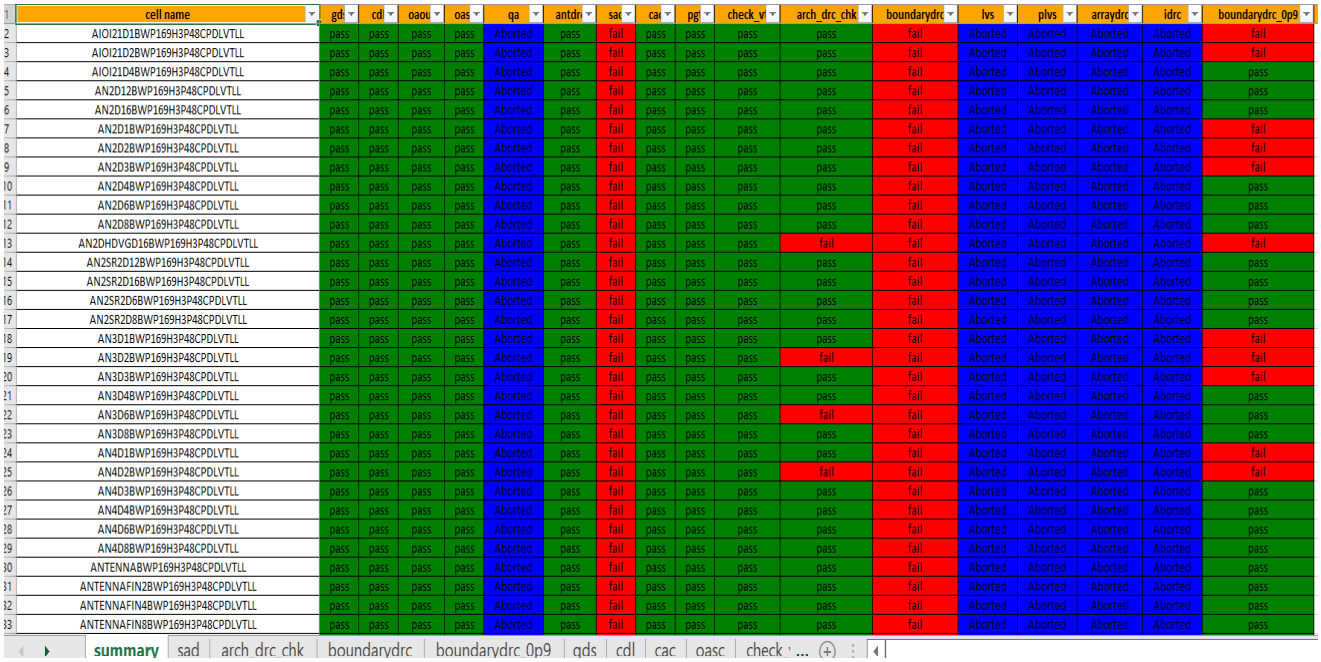
QA:

It is Quality Assurance Check, helps to check whether it is following DRC rules or not.

```
#fileName: AOI21OPT3FD9BWP143M286H3P48CPDULVT.qa.rep
#version: v1.0
#runTime: 2023-02-09 21:08:25.932987
#cell name: AOI21OPT3FD9BWP143M286H3P48CPDULVT
#check: QA Check

INFO: pr_boundary object is present
INFO: pr_boundary shape is  present
INFO: prBoundary shape and object is aligned.
INFO: prBoundary is on the origin
WARN: Aspect ratio W/L: 3.18881  Should be less than 1.5.
INFO: Text label to represent cell name is matching with cellname
INFO: Text label to represent cell name is on the origin.
INFO: Height of cell  is integral multiple of height defined in cellname (286)
INFO: No pin is having extra spaces in their names.
INFO: Pins present in cdl :A1 A2 B ZN VBB VDD VPP VSS
INFO: Pins present in layout :VBB VPP VPP VDD VSS VDD B ZN A1 A2 ZN
INFO: pins btw layout and cdl is same
INFO: VBB label is present on layer PSUB
INFO: VPP label is present on layer NW
INFO: VDD label is present on layer M0
INFO: VSS label is present on layer M0
INFO: VDD label is present on layer M0
INFO: B label is present on layer M1
INFO: ZN label is present on layer M1
INFO: A1 label is present on layer M1
INFO: A2 label is present on layer M1
INFO: ZN label is present on layer M1

INFO: The specified VDD pin layer is on M0
INFO: The (M0, drawing) layer is exactly overlaping with VDD pin layer

INFO: The specified VSS pin layer is on M0
INFO: The (M0, drawing) layer is exactly overlaping with VSS pin layer

INFO: The specified VDD pin layer is on M0
INFO: The (M0, drawing) layer is exactly overlaping with VDD pin layer

INFO: The specified B pin layer is on M1
INFO: The (M1, drawing) layer is exactly overlaping with B pin layer
INFO: The (Pin_Mrk, M1) layer is exactly overlaping with B pin layer

INFO: The specified ZN pin layer is on M1
INFO: The (M1, drawing) layer is exactly overlaping with ZN pin layer
INFO: The (Pin_Mrk, M1) layer is exactly overlaping with ZN pin layer
```

```
INFO: The specified A1 pin layer is on M1
INFO: The (M1, drawing) layer is exactly overlaping with A1 pin layer
INFO: The (Pin_Mrk, M1) layer is exactly overlaping with A1 pin layer

INFO: The specified A2 pin layer is on M1
INFO: The (M1, drawing) layer is exactly overlaping with A2 pin layer
INFO: The (Pin_Mrk, M1) layer is exactly overlaping with A2 pin layer

INFO: The specified ZN pin layer is on M1
INFO: The (M1, drawing) layer is exactly overlaping with ZN pin layer
INFO: The (Pin_Mrk, M1) layer is exactly overlaping with ZN pin layer

###################### MJ_PROP CHECK  ##################
INFO: The specified ZN  pin is having output direction
INFO: The ZN has 2 pins
INFO: The "Y" cordinates of all Metal pins for ZN are aligned properly--
INFO: The "Y" cordinates of all MJ pins for ZN are aligned properly
INFO: The "Y" cordinates of all Metals for ZN are aligned properly
INFO: The "MJ group layer" is drawn perfectly on all pins ZN pins
INFO: Pin of ZN (0.634 0.005 0.662 0.2685 ) aligned with MJ pins
INFO: Pin of ZN (0.394 0.005 0.422 0.2685 ) aligned with MJ pins
WARN: The required metal strip_count is 4.5 , used for ZN is 2

###################### M2_PIN CHECK  ##################
INFO: M2_pins are not present
INFO: M_Group is not present in M2

##################################    QA1      ##################################
INFO: Cell : AOI21OPT3FD9BWP143M286H3P48CPDULVT is having normal cell configration (M0 pattern : VDD VSS VDD).
INFO: Label VDD X:0.024 Y:-0.006 is present M0_E2  is covering the label.
INFO: Label VSS X:0.024 Y:0.123 is present M0_E1  is covering the label.
INFO: Label VDD X:0.024 Y:0.28 is present M0_E1  is covering the label.
INFO: Label VBB is prsent  (Ignore case of error)
INFO: Label VPP X:0.024 Y:0 is prsent NW  is covering the label.
INFO: Label VPP X:0.024 Y:0.286 is prsent NW  is covering the label.
################################################ labels for extraction check ###############################
INFO : All pin labels : VDD VSS B ZN A1 A2 are not present on higher metals.
PASS
```

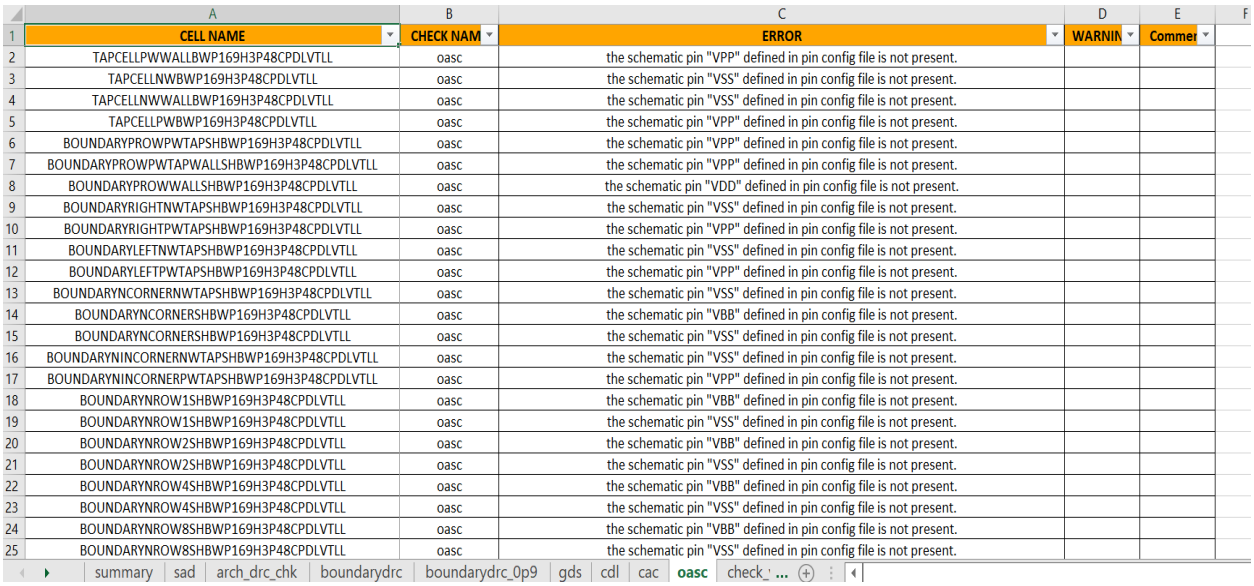
QA Dashboards For Verification



cell name	gd	cd	oaou	oas	qa	antdr	sa	ca	pg	check_v	arch_drc_chk	boundarydrc	lvs	plvs	arraydrc	idrc	boundarydrc_0p9
AIOI21D1BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	fail
AIOI21D2BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	fail
AIOI21D4BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2D12BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2D16BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2D1BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	fail
AN2D2BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	fail
AN2D3BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	fail
AN2D4BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2D6BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2D8BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2DHDVGD16BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	fail	fail	Aborted	Aborted	Aborted	Aborted	fail
AN2SR2D12BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2SR2D16BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2SR2D6BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN2SR2D8BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN3D1BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	fail
AN3D2BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	fail	fail	Aborted	Aborted	Aborted	Aborted	fail
AN3D3BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	fail
AN3D4BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN3D6BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	fail	fail	Aborted	Aborted	Aborted	Aborted	pass
AN3D8BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN4D1BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	fail
AN4D2BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	fail	fail	Aborted	Aborted	Aborted	Aborted	fail
AN4D3BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN4D4BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN4D6BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
AN4D8BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
ANTENNABWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
ANTENNAFIN2BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
ANTENNAFIN4BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass
ANTENNAFIN8BWP169H3P48CPDLVTLL	pass	pass	pass	pass	Aborted	pass	fail	pass	pass	pass	pass	fail	Aborted	Aborted	Aborted	Aborted	pass

summary | sad | arch_drc_chk | boundarydrc | boundarydrc_0p9 | gds | cdl | cac | oasc | check_' ...

Details of Failed Checks Info



CELL NAME	CHECK NAM	ERROR	WARNIN	Commer
TAPCELLPWWALLBWP169H3P48CPDLVTLL	oasc	the schematic pin "VPP" defined in pin config file is not present.		
TAPCELLNWBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
TAPCELLNWWALLBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
TAPCELLPWBWP169H3P48CPDLVTLL	oasc	the schematic pin "VPP" defined in pin config file is not present.		
BOUNDARYPROWPWTAPSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VPP" defined in pin config file is not present.		
BOUNDARYPROWPWTAPWALLSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VPP" defined in pin config file is not present.		
BOUNDARYPROWWALLSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VDD" defined in pin config file is not present.		
BOUNDARYRIGHTNWTAPSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
BOUNDARYRIGHTPWTAPSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VPP" defined in pin config file is not present.		
BOUNDARYLEFTNWTAPSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
BOUNDARYLEFTPWTAPSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VPP" defined in pin config file is not present.		
BOUNDARYNCORNERNWTAPSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
BOUNDARYNCORNERSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VBB" defined in pin config file is not present.		
BOUNDARYNCORNERSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
BOUNDARYNINCORNERNWTAPSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
BOUNDARYNINCORNERPWTAPSHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VPP" defined in pin config file is not present.		
BOUNDARYNROW1SHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VBB" defined in pin config file is not present.		
BOUNDARYNROW1SHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
BOUNDARYNROW2SHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VBB" defined in pin config file is not present.		
BOUNDARYNROW2SHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
BOUNDARYNROW4SHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VBB" defined in pin config file is not present.		
BOUNDARYNROW4SHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		
BOUNDARYNROW8SHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VBB" defined in pin config file is not present.		
BOUNDARYNROW8SHBWP169H3P48CPDLVTLL	oasc	the schematic pin "VSS" defined in pin config file is not present.		

summary | sad | arch_drc_chk | boundarydrc | boundarydrc_0p9 | gds | cdl | cac | oasc | check_' ...

Potential Impact

The potential impact of layout automation tools for standard cells is significant, as they have the potential to significantly reduce the time and cost required to design and manufacture complex chips. By automating tasks such as design rule checking and verification, these tools can help designers identify and address issues more quickly, reducing the time required to bring a chip to market. Additionally, the use of machine learning algorithms and other advanced techniques can help improve the accuracy and efficiency of the design process, ultimately leading to higher-quality chips and more satisfied customers.

Future Work

There are several areas of future work that could further improve the efficiency and accuracy of layout automation tools for standard cells. One area of focus could be the development of more advanced machine learning algorithms that can identify and predict issues in the chip design process before they occur. Additionally, the integration of artificial intelligence and machine learning into the layout automation process could help improve the accuracy and efficiency of the process even further.

Our next approach is to develop Netbatch + Multiprocessing approach of CuQA Flow to archieve improved Runtime and to apply Machine Learning models on Layout.

CONCLUSION

In conclusion, the development of layout automation tools for standard cells has the potential to significantly improve the efficiency and accuracy of the chip design process. The implementation of tools such as CuQA Tool by LAQA team and NetBatch scheduling algorithms has been shown to improve the runtime of deep learning model training, optimize resource allocation, and improve the accuracy of machine learning models. Additionally, the development of effective QA dashboards and debugging and troubleshooting techniques can help identify and address issues quickly, improving the overall quality of the chip design process.

REFERENECE

[1] Scikit Learn :

https://scikit-learn.org/stable/modules/neural_networks_supervised.html

[2] Wikipedia :

https://en.wikipedia.org/wiki/Standard_cell

[3] Multiprocessing :

https://docs.python.org/3/library/multiprocessing.html

[4] Multithreading :

https://docs.python.org/3/library/threading.html