

**Ревазян Д.В.**

студент 2 курса магистратуры НПУА,

г. Ереван, Армения

## **СПЕЦИФИКА СОЗДАНИЯ КОМАНДНОГО ФАЙЛА DRC ДЛЯ ПРОВЕРКИ НЕСКОЛЬКИХ ПРАВИЛ**

**Аннотация.** Этапы проектирования могут сопровождаться ошибками, которые не соответствуют определенным ограничениям. Процесс проверки правил проектирования (Design Rule Checking или DRC) осуществляется в соответствии с большим количеством правил, обеспечивающих функционал автоматизированного характера для оценки целостности проекта на физическом и логическом уровне. В статье представлен процесс разработки командного файла DRC для проверки ряда правил на основе типового примера, в соответствии с которым проходит оценка правильности проверки.

**Ключевые слова:** специфика, создание, командный файл, DRC, проверка, несколько, правила.

## **THE SPECIFICS OF CREATING A DRC COMMAND FILE FOR CHECKING MULTIPLE RULES**

**Annotation.** The design stages may be accompanied by errors that do not meet certain constraints. The process Design Rule Checking (DRC) is carried out in accordance with a large number of rules that provide automated functionality to assess the integrity of the project at the physical and logical level. The article presents the process of developing a DRC command file for checking a number of

rules based on a sample example, according to which the correctness of the check is evaluated.

**Keywords:** specifics, creation, batch file, DRC, verification, multiple, rules.

На основе комплекса правил проектирования (DRC) выделяют ограничения платы, что позволяет ограничить нахождение целевых объектов в рамках конструктивных допусков и требований. Проверка направлена на все правила проектирования или только некоторые из них. Она может проходить как в онлайн режиме, так и оффлайн одновременно с самим процессом проектирования. Также возможна организация проверки в пакетном варианте, когда результаты проверки отображаются на системной панели Messages, либо в виде сформированного по результатам проверки отчета. Наиболее часто такой вариант встречается при организации проверок при проектировании в пакетном варианте. Оптимизация итоговой стоимости готовой продукции способствует реализации проектирования с использованием технологических возможностей самого производителя. DRC – это автоматизированный программный модуль с высокой производительностью, который оценивает целостность проекта печатных плат на физическом и логическом уровне. Включение функционала DRC является обязательным на этапе трассировки, так как помогает контролировать соответствие на уровне минимальных зазоров и выявлять другие нарушения или их отсутствие. По результатам проверки, проведенной данным модулем, формируется отчет о выявленных ошибках [3].

Таким образом, проверка правил проектирования DRC представляет собой процесс физического проектирования, позволяющий определить, соответствует ли компоновка микросхемы ряду правил, определенных производителем полупроводников. Каждый полупроводниковый процесс будет иметь свой собственный набор правил и обеспечивать достаточные пределы, чтобы нормальная изменчивость производственного процесса не привела к выходу из строя микросхемы. При этом, правило вполне может

быть таким же простым, как минимальная ширина провода и расстояние между двумя проводами.

Количество правил, связанных с более продвинутыми узлами процесса, быстро увеличивается, и сложность этих правил также возрастает. Многие из этих правил распространились от однослойных до многослойных правил. Конструктивные ограничения превратились в сложный, взаимозависимый, многомерный набор переменных. Ограничения создают постоянно расширяющийся набор требований, что приводит к резкому увеличению размера и сложности колоды правил. Там, где когда-то было достаточно простых одномерных проверок, многомерные проверки, которые изучают взаимосвязь нескольких геометрий в больших диапазонах, теперь необходимы для обеспечения технологичности.

Часто имеются конструктивные соображения, которые бывает довольно трудно реализовать в текстовых правилах проектирования. Этот всплеск сложности означает, что многие желаемые расширенные проверки DRC трудно (если не невозможно) точно закодировать. Увеличение размера и сложности колоды также приводит к увеличению времени выполнения проверки и увеличению времени отладки [5].

В качестве примера специфики создания командного файла DRC для проверки нескольких правил рассмотрим Diva DRC. Программный модуль, осуществляет проверку соответствия между технологическими проектными правилами и топологией. Программа способна проверять разное количество слоев от одного слоя и проведения одномерного формата проверки до многомерного формата для большого количества слоев. Итогом действия команды DRC является либо выявление ошибки, либо формирование результирующего слоя, который в дальнейшем становится совокупностью входных данных для следующей команды Diva.

DIVA позволяет осуществлять пространственную интерполяцию данных (анализ) оптимальным способом, сопоставимым с оптимальной интерполяцией (OI). Предоставляются инструменты для создания сетки

конечных элементов, а также инструменты для оптимизации параметров анализа. Можно выполнить контроль качества данных и рассчитать поля ошибок. Кроме того, возможно изменение тренда данных.

Diva команды базируются на 5 инструментах и представляют собой комплекс элементов, осуществляющих топологическую проверку:

1. Элементы, осуществляющие контроль соответствий между установленными технологическим нормам и топологией - Design Rule Checker (iDRC).

2. Элементы, осуществляющие выделения паразитных частей, обнаруженных в топологии - Layout Parasitic Extractor (iLPE).

3. Элементы, осуществляющие выделение паразитных резисторов и емкостей (RC-цепей) - Parasitic Resistance Extractor (iPRE).

4. Элементы контроля электрических параметров и соответствий Electrical Rules Checker (iERC).

5. Элементы, осуществляющие контроль соответствий между принципиальной электрической схемой и топологией - Layout Versus Schematic program (iLVS).

Элементы iDrc применяются при оценке корректности топологических правил и технологических норм. Элементы iLPE привлекаются, когда необходимо выделить паразитные устройства, обнаруженные в топологии. Действие элементов iPRE направлено на выделение резисторов и емкостей с целью дальнейшей трансформации принципиальной электрической схемы в полную схему, на базе созданных в процессе RC-цепей.

Элементы iERC контролируют процессы в электрических соединениях (аномальные варианты подключений в схеме и топологии, устройства и цепи «висящего» типа). Схемы на транзисторном уровне трансформируются в схемы, приводимые в действие на уровне вентилях (включая процессы моделирования в зоне вентилях). Элементы iLVS реализуют контрольные функции по установлению соответствий между принципиальной электрической схемой и топологией.

Проверки могут быть запущены в трех доступных вариантах:

1. Путем активации команды «Verify» в окне дизайна.
2. Путем использования языка SKILL.
3. Путем активации опции «-nograph» без включения графического интерфейса.

Чаще всего результаты интерактивных процессов DRC отражаются в соответствующем CIW окне. Относительно данных выводимых результатов необходимо отметить следующее:

1. Правила могут быть объединены в группы на основании процессов оптимизации и изменения порядка следования элементов. В результате проверок, осуществляемых в групповом формате, формируется сводный отчет, состоящий из сообщений единичных проверок.

2. При активации иерархической формы проверок, результаты не отражаются на CIW экране. Это происходит для того, чтобы не перегружать отчет большим количеством однотипных сообщений, получаемых в результате проверки каждой ячейки.

3. Невозможность запуска интерактивной формы DIVA в среде UNIX, так как отсутствует потребность в активации графического интерфейса. Результаты проверок, с выявленными ошибками оформляются в виде стандартного файла.

При создании проверочного командного файла DRC должны соблюдаться определенные условия. Команды верификационного файла должны быть сгруппированы в соответствии с тремя функциями языка SKILL. Каждая функция соотносится с соответствующей проверочной программой Diva (DRC/Extract, ERC и LVS).

Преимущества пакетного формата применения DRC перед онлайн форматом заключается в возможности активации проверочных процессов в любое заданное время в процессе формирования платы, а не ограничиваться контролем только вновь обнаруженных ошибок. Для проектировщиков с

опытом преимущества онлайн формата DRC являются безоговорочными, однако именно пакетный вариант используется ими для начальных и финальных этапов конструирования плат.

При внимательной и поэтапной проверке правил в процессе проектирования, в результате финального запуска пакетной DRC для завершенной, полностью растрассированной платы может произойти только несколько нарушений, либо их может не быть совсем. Однако может быть отмечено и достаточно большое количество нарушений, и тогда задача сводится к устранению этих нарушений.

В процессе DRC всегда происходит обнаружение всех нарушений. Система отслеживает, какие нарушения были отклонены, и повторно применяет это состояние после выполнения проверки правил. Кроме того, отклоненные нарушения удаляются из сообщений об ошибках в панели Messages, но они продолжают помечаться визуально в проектной области [2].

Отчеты о DRC, формируемые пакетной DRC, могут показаться довольно пугающими для начинающих конструкторов. Для управляемого процесса необходимо определить стратегию. Одна из стратегий – ограничить количество сообщаемых нарушений. При задании опций отчета в диалоговом окне Design Rule Checker указывается меньшее значение для Stop When Found. Еще один вариант – это многоэтапное применение DRC. При выявлении многократных ошибок в конструкции необходимо активировать правила одно за другим. Такой подход позволяет найти наиболее оптимальный вариант осуществления проверки правил проектирования.

Проведенный анализ данных показывает, что именно своевременно активированная программа проверки правил проектирования дает возможность выявить максимальное количество ошибок и несоответствий, и выработать наиболее оптимальный вариант для их исключения. Анализ теоретических источников по данному вопросу и практического опыта выявления DRC ошибок позволил определить три ключевых варианта, позволяющих скорректировать несоответствия [4].

1. Корректировка объектов по модели 1111 путем изменения положения объектов, а также через корректировку полигонов и проводников:

1.1 Изменение положения, добавление объектов, удаление, корректировка слоев и размеров.

1.2 Корректировка положения отверстий, посадочных мест и т.д.

2. Корректировка правил проектирования и соответствующих ограничений.

3. Корректировка установленных настроек DRC-проверки. В отчете будут отражаться только ошибки тех правил, которые включены в проверку через настройку DRC. Ошибки отключенных правил, входить в отчет не будут.

Последний вариант будет эффективным на промежуточных этапах проектирования для ограничений, ошибки которых невозможно скорректировать другими методами, в частности, это касается ограничений целостности сигнала.

Таким образом, выбор правил должен осуществляться, исходя из того, что в приоритете правила, которые не могут быть нарушены. Любое правила, попадающее под проверку, может быть выключено и активировано через настройки. DRC-режим может быть активирован только при условии внесения всех требуемых корректировок в правила. Результаты проведенной проверки в виде обнаруженных несоответствий и ошибок представляются в окне просмотра. Все выявленные несоответствия и нарушения правил имеют комментарий и обозначение соответствующего слоя [1]. Итоги проведенной проверки правил проектирования представляют собой комплекс ошибок и несоответствий, которые должны быть скорректированы до финальных этапов проекта. Командный файл DRC является эффективным инструментом для выявления, анализа и корректировки ошибок в правилах проектирования, что в конечном итоге будет способствовать уменьшению общих временных затрат на проектирование правил и повышения их качества.

## СПИСОК ИСТОЧНИКОВ

1. Контроль правил разработки (DRC). URL: <http://radio-hobby.org/modules/instruction/sprint-layout-6/drc> (дата обращения: 12.02.2022).
2. Проверка правил проектирования AD. URL: <https://www.altium.com/ru/documentation/altium-designer/design-rule-checking-ad> (дата обращения: 12.02.2022).
3. Романова Е.Б. Разработка и апробация алгоритма коррекции ошибок в системах автоматизации проектирования печатных плат // Научно-технический вестник информационных технологий, механики и оптики. 2016. №2. URL: <https://cyberleninka.ru/article/n/razrabotka-i-aprobatsiya-algoritma-korreksii-oshibok-v-sistemah-avtomatizatsii-proektirovaniya-pechatnyh-plat> (дата обращения: 12.02.2022).
4. Романова Е.Б., Сумцов А.В. Анализ и коррекция DRC-ошибок в САПР печатных плат // Приборостроение. 2015. №10. URL: <https://cyberleninka.ru/article/n/analiz-i-korreksiya-drc-oshibok-v-sapr-pechatnyh-plat> (дата обращения: 12.02.2022).
5. Design Rule Checking (DRC). URL: [https://semiengineering.com/knowledge\\_centers/eda-design/verification/design-rule-checking-drc/](https://semiengineering.com/knowledge_centers/eda-design/verification/design-rule-checking-drc/) (дата обращения: 12.02.2022).

© Ревазян Д.В. 2022