

## МОДИФИКАЦИЯ МОДУЛЯ ОТОБРАЖЕНИЯ ИНФОРМАЦИИ О МОДЕЛЯХ ЭЛЕКТРИЧЕСКИХ ЭЛЕМЕНТОВ В САПР «CIRCUIT STUDIO»

Лубов Г.П.

Научный руководитель – кандидат технических наук, доцент Горяинов А.Е.  
Томский Государственный Университет Систем Управления и Радиоэлектроники  
(ТУСУР)

### Введение

В компании ООО «50ohm Technologies» ведётся разработка системы автоматизированного проектирования (САПР) «Circuit Studio», предназначенной для автоматизированного синтеза принципиальных схем.

Для осуществления синтеза пользователь программы в лице проектировщика сверхвысокочастотных монолитных интегральных схем (СВЧ МИС) выбирает тип синтезируемого устройства и загружает файл дубликата комплексного инструмента проектирования (КИП). В дубликате КИП находятся табличные модели электрических элементов, учитывающие особенности конкретного технологического процесса.

В процессе проектирования пользователю может потребоваться информация о математических моделях. На вкладке «Техпроцесс» пользователь может просмотреть информацию о загруженных моделях электрических элементов в виде нумерованного списка, в котором для каждой модели отображено название модели, количество портов и количество вариаций данной модели. Однако пользователю не доступна такая информация о моделях как: условно-графическое обозначение (УГО), топология, параметры и состояния, теги модели с метаинформацией. Это осложняет использование моделей при задании требований для синтеза.

Таким образом, целью работы является расширение возможностей интерфейса отображения информации о загруженных моделях характеристик в САПР «Circuit Studio».

### Основная часть

Программа «Circuit Studio» реализована на платформе для создания настольных приложений WPF [1], а также использует архитектурный шаблон проектирования MVVM [2], подразумевающий разбиение программы на 3 уровня: Вид, Представление Вида и Модель. Для достижения цели необходимо модифицировать существующую вкладку «Техпроцесс»: добавить канву для отрисовки УГО, канву для отрисовки топологии, таблицу параметров модели и таблицу состояний модели (должна отображаться только в случае, если выбранная пользователем модель имеет состояния).

Для реализации требуемой функциональности добавлен класс представления вида *SelectedModelVM*, который отвечает за загрузку и хранение данных о выбранной модели. В момент выбора пользователем модели, данный класс загружает данные по этой модели из файла дубликата КИП, используя существующие классы *ProjectManager* и *ModelsManager*.

Алгоритмы отрисовки УГО и топологии имеют сходства, т.к. оба работают с графическими примитивами, однако на вход принимают различные типы данных, поэтому добавлены два класса: *SymbolDrawer* и *LayoutDrawer* для отрисовки УГО и топологии, соответственно. УГО и топология могут иметь различные размеры, а элементы управления для отрисовки могут масштабироваться при изменении размеров окна, поэтому первым этапом алгоритма отрисовки является масштабирование графических примитивов, из которых состоят УГО и топология. Сперва рассчитывается коэффициент масштабирования так, чтобы элемент поместился на канву отрисовки с

сохранением исходных пропорций. Затем в классах *SymbolDrawer* и *LayoutDrawer* вызываются методы *Scale*, в которых координаты каждого примитива, координаты расположения текстовых обозначений и размер шрифта умножаются на коэффициент масштабирования. Методы *Scale* возвращают УГО и топологию с корректным масштабом, которые поступают на вход методов *Draw* тех же классов. Отрисовка производится с использованием библиотеки Skia [3]. Методы *Draw* возвращают растровые изображения УГО и топологии, которые присваиваются в свойства зависимости *Symbol* и *Layout* класса *SelectedModelVM*, а затем с помощью механизма привязки данных помещаются в элементы управления *Image* графического интерфейса.

Параметр модели имеет следующий вид: название параметра; название зависимого параметра (при наличии); размерность; строковое или числовое значение; минимальное, максимальное значение и значение шага – для числовых параметров. Для хранения отображаемых данных используется свойство зависимости *Parameters* в классе *SelectedModelVM*, содержащее коллекцию параметров, представленных классом *ModelParameterVM*. Для отображения параметров на пользовательском интерфейсе добавлена таблица *DataGrid*.

Значения некоторых параметров могут варьироваться в зависимости от состояния модели. Это актуально для моделей переключательных элементов, например, коммутационных транзисторов. Для хранения состояний в классе *SelectedModelVM* добавлено свойство зависимости *States* типа *ModelStatesVM*. Внутри *ModelStatesVM* добавлена наблюдаемая коллекция *ModelStateVM*. *ModelStateVM* является представлением вида для параметра в определённом состоянии модели.

Таблицы параметров и состояний размещены в элементе управления *TabControl* на двух разных вкладках. Вкладка с таблицей состояний доступна пользователю только, если у выбранной модели есть состояния.

Для модифицированного модуля проведено ручное тестирование. Выявленные ошибки исправлены. Модифицированный модуль успешно прошёл тестирование и интегрирован в САПР «Circuit Studio».

### Выводы

В результате модификации модуля отображения информации о моделях электрических элементов, пользователи программы могут просмотреть подробную информацию о каждой загруженной модели. Данная модификация ускоряет взаимодействие пользователя с программой и удобство использования. Как следствие, сокращается время проектирования принципиальной схемы устройства.

### Литература

1. Общие сведения о WPF [Электронный ресурс]. URL: <https://learn.microsoft.com/ru-ru/dotnet/desktop/wpf/overview/> (дата обращения 19.02.2026)
2. R. Vice, MVVM Survival Guide for Enterprise Architectures in Silverlight and WPF. – Birmingham: Packt Publishing, 2012. – 491 с.
3. Документация Skia [Электронный ресурс]. URL: <https://skia.org/docs/> (дата обращения 19.02.2026)

Автор \_\_\_\_\_ Лубов Г.П.

Научный руководитель \_\_\_\_\_ Горяинов А.Е.