

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ WEBGL ДЛЯ ОТОБРАЖЕНИЯ РЕЗУЛЬТАТОВ РАБОТЫ СИСТЕМ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ УЗЛОВ ЭВМ

Яковлев Д.С.

Научный руководитель – к.т.н., профессор Борде Б.И.

Сибирский федеральный университет

Для вывода результатов автоматического проектирования любая САПР должна использовать самые передовые технологии, иначе она неизбежно начнет отставать от своих конкурентов. Одной из таких технологий и является WebGL, которая позволяет выводить результаты в браузере клиента в трехмерном виде без установки дополнительных программ на клиенте.

WebGL – это технология отображения трехмерной графики в браузере, использующая в своей работе аппаратное ускорение видеоадаптера, установленного на клиентской машине. Аппаратное ускорение реализуется, используя аппаратно-программный интерфейс OpenGL ES через привязку скриптов на языке JavaScript. Таким образом, для отображения трехмерной графики с помощью технологии WebGL достаточно наличия html-страницы, содержащей JavaScript-скрипты, которые вызывают функции из библиотеки OpenGL ES. В будущем, при широком распространении данной технологии, мы все чаще будем видеть трехмерные элементы на страницах и даже целые сайты в трехмерном исполнении.

Поддержка трехмерной графики имела и в более ранних версиях браузеров, но такая поддержка требовала установки на компьютер пользователя дополнительного программного обеспечения. Примерами таких систем являются: VRML (Virtual Reality Modeling Language - язык моделирования виртуальной реальности) и X3D (стандарт ISO, предназначенный для работы с трёхмерной графикой в реальном времени).

Целью данной работы была разработка модуля отображения результатов проектирования для учебно-исследовательской САПР аналого-цифровых вычислительных устройств и систем – УИ САПР COD (Conceptual Object Design).

Интерфейс УИ САПР COD с другими САПР может выполняться с использованием функций различных уровней. Функции нижних уровней работают со структурами данных конкретной САПР нижнего уровня, например PCAD. Функции верхних уровней не зависят от конкретной САПР нижнего уровня и от способа реализации интерфейса. На рис. 1 приведена многоуровневая структура табличного интерфейса с САПР PCAD.

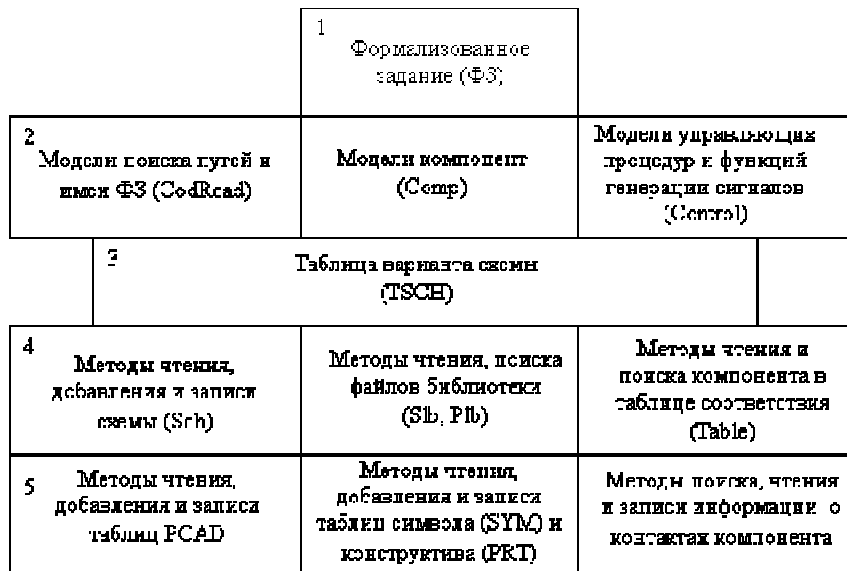


Рис. 1. Многоуровневая структура табличного интерфейса УИ САПР с САПР PCAD

Реализованный в ходе разработки модуль состоит из двух функциональных блоков:

1. Формирование xml-файла, содержащего описание микросхем, входящих в состав схемы, с указанием корпуса микросхемы и размеров отдельных деталей корпуса;
2. Html-страницы, содержащей трехмерную схему, с подключаемым внешним файлом скриптов на языке JavaScript.

Работа блока формирования xml-файла заключается в преобразовании таблицы варианта схемы (TSCH) в xml-файл, содержащий компоненты, располагающиеся на схеме. Xml-файл содержит тип корпуса для каждого компонента и все необходимые размеры и характеристики для отображения данного компонента. Для формирования данного файла, кроме самой таблицы вариантов схемы, используются дополнительные библиотеки данных, информация из которых загружается в специализированную утилиту на языке C#, которая производит сопоставление информации из библиотек в иерархическую внутреннюю структуру. После загрузки внешних библиотек, происходит чтение формализованного задания, из которого извлекаются используемые компоненты и в выходной список компонентов добавляются данные элементы, содержащие необходимую дополнительную информацию, загруженную из библиотек. Далее утилита формирует выходной файл, экспортируя созданный список компонентов в xml-файл.

Html-страница содержит только контейнер для вывода трехмерной модели на странице, вся логика работы сосредоточена во внешнем файле скриптов, который обеспечивает асинхронное чтение ранее сформированного xml-файла и преобразования его в графические примитивы WebGL, которые и отображаются в браузере клиента.

В ходе разработки первоначально был реализован вариант прямой обработки таблицы вариантов схемы скриптами, включаемыми в отображаемую клиенту html-страницу. Отказ от этого варианта реализации имел несколько причин:

1. Тесное связывание САПР с реализацией клиентской части, что приводит к практически полному изменению кода при использовании другого источника;
2. Скорость работы скриптов в браузере намного ниже скорости работы компилируемых языков, даже тех, которые компилируются в промежуточные языки;

3. Разработка на языках с поддержкой контроля типов и другими встроенными возможностями позволяет разрабатывать более надежные приложения;
4. Отсутствие необходимости загружать и сопоставлять данных из библиотек, рассчитывать размеры каждого компонента при каждой загрузке страницы.

Вариант реализации, использованный в данной работе, позволяет разделить обработку входных данных от отображения, что в некоторой степени реализует архитектуру MVC (Model-view-controller - Модель-представление-поведение). Разделение производится с помощью введения унифицированного формата xml-файла, загружаемого в отображаемую пользователем страницу, что добавляет возможность использования данного модуля в других системах, внося изменения только в блок формирования xml-файла, либо используя стороннюю утилиту, вплоть до ручного создания xml-файла. Так же данное разделение позволяет вносить изменения в отображение трехмерной модели, не изменяя используемую модель данных, т.е. нет необходимости вносить изменения в блок формирования xml-файла, что, учитывая возможность использования модуля в разных системах с использованием различных блоков формирования xml-файла, является большим преимуществом.

В данной работе использован ряд прогрессивных технологий:

- Технологии linq и linq to xml при загрузке и сопоставлении данных из библиотек;
- Технологии сериализации и десериализации объектов, при загрузке данных из библиотек и формировании xml-файла;
- Технология AJAX, позволяющая загружать сформированный xml-файл асинхронно, уже после загрузки html-страницы пользователем.

В результате работы был реализован модуль отображения результатов проектирования для УИ САПР COD. Данный модуль дает возможность подключения его к другим САПР, изменяя только один функциональный блок, а так же возможность вывода результатов, независимо от платформы клиента. К клиенту имеется 2 требования: наличие браузера, поддерживающего технологию WebGL и аппаратной возможности выводить трехмерную графику.

На данном этапе реализации сформированная html-страница находится на том же компьютере, где и производилось проектирование. В дальнейшем, чтобы раскрыть полностью возможности WebGL технологии, необходимо помещать данную страницу на веб-сервер, чтобы любой клиент мог иметь к ней доступ из локальной сети или из сети интернет. В этом и заключается отличительная черта технологии WebGL – возможность отображения трехмерной графики в сети без использования дополнительного программного обеспечения.