

ПРОГРАММНАЯ ПОДДЕРЖКА БАЗЫ ЗНАНИЙ В СИСТЕМЕ ЛОГИЧЕСКОГО ПРОЕКТИРОВАНИЯ FLC-2

Романов В. И.

Объединённый институт проблем информатики Национальной академии наук Беларуси
Минск, Республика Беларусь
E-mail: rom@newman.bas-net.by

Предлагается краткое описание подсистемы ведения базы знаний, используемой для системы логического проектирования дискретных устройств FLC-2. Данная подсистема позволяет эксперту подготавливать разнообразные сценарии проектирования без изменения самой САПР.

ВВЕДЕНИЕ

На современном этапе развития САПР логического проектирования дискретных устройств, реализуемых на базе заказных цифровых СБИС, масштабы решаемых ими задач таковы, что интеллектуальная поддержка процессов проектирования становится необходимой. В работе [1] был предложен подход к организации логического проектирования дискретных устройств на основе продукционно-фреймовой модели представления знаний. Там же была описана и программная система FLC, реализующая указанный подход.

В настоящее время эта система получила дальнейшее развитие и, выступая под наименованием FLC-2, отличается от своей предшественницы кроссплатформенной реализацией, основанной на использовании языка программирования C++ и библиотеки Qt [2]. Вместе с тем, существенно поменялся и состав входящих в ядро САПР программных модулей, реализующих проектные процедуры.

I. ПРОЦЕСС ПРОЕКТИРОВАНИЯ И ОБЪЕКТЫ В БАЗЕ ЗНАНИЙ

Собственно процесс логического проектирования, реализуемый в рамках системы FLC-2, можно трактовать как процесс изменения описания схемы, представленного на некотором языке проектирования. Переход из одного состояния к другому осуществляется под воздействием проектных процедур, реализуемых заранее подготовленными программными модулями. Обычно функционально-структурные описания схемы носят иерархический характер, листовые элементы иерархии на заключительном этапе проектирования соответствуют элементам используемой технологической библиотеки, на которой схема реализуется. Проектная процедура может применяться как ко всей схеме, так и к отдельным ветвям дерева (блокам иерархии). В силу указанного, на каждом шаге проектирования решается вопрос о том, какую из доступных проектных процедур применить в текущий момент времени и к какому из существующих блоков иерархии. В некоторых случаях ответ на поставленный вопрос может быть получен на основании автома-

тизированного анализа текущего состояния проектирования. Фактически речь идет о подготовленном сценарии обработки проекта (*стратегии* в терминах системы FLC-2). Сценарий представляется цепочкой команд (*продукций*), задающих требуемое для срабатывания продукции условие и описывающих действие, которое будет выполнено при срабатывании. Условие продукции представляется предикатом, в качестве аргументов которого выступают *атрибуты*, определяющие существенные свойства проектируемой схемы или самого процесса проектирования. В заключении продукции устанавливается выполняемое действие, чаще всего *программный модуль*, реализующий заказанную проектную процедуру.

Таким образом, в базе знаний (БЗ) определены такие типы объектов, как стратегии, продукции, программные модули и атрибуты. Кроме них оказывается удобным размещать в БЗ *меню*, определяющее возможности обслуживания пользователя с точки зрения доступных проектных процедур и целых стратегий, а также *фреймы* и *слоты*, представляющие собой средства классификации и группирования атрибутов, используемые в процессе эксплуатации FLC-2.

II. ПОДСИСТЕМА ОБСЛУЖИВАНИЯ БАЗЫ ЗНАНИЙ

Ведение БЗ в системе FLC-2 возлагается на эксперта – специалиста в рассматриваемой проблемной области, с одной стороны, и знакомого с организацией применяемого программного обеспечения, с другой. Для автоматизации процессов управления БЗ разработана отдельная подсистема ESS-2 (Expert SubSystem), являющаяся развитием ранее используемой подсистемы, описанной в [1].

Используемая в FLC-2 БЗ представлена одним файлом в формате JSON [3]. Этого формата оказывается вполне достаточно для представления информации, обладающей древовидной структурой, где на первом уровне иерархии располагаются типы хранимых объектов, а далее все сводится либо к парам «имя»–«значение», либо к векторному представлению «имя»–«перечень значений». Применение фор-

мата JSON позволяет воспользоваться готовым программным обеспечением, реализующем его обслуживание.

Эксплуатация подсистемы ESS-2 осуществляется в диалоговом режиме. Основное рабочее окно ESS-2 после открытия файла БЗ представлено на рис.1. Пользователь в меню выбирает интересующий его тип данных, в соответствии с чем открывается специальное диалоговое окно определения характеристик объекта и его значений, используемых по умолчанию. Пример такого диалога для работы с программными модулями представлен на рис.2. Часть функционала подсистемы может быть использована путем нажатия кнопок, размещенных на ее основном окне возле представленных на нем списков объектов.

Таким образом можно заказать добавление нового, изменение или удаление текущего объекта, а также изменение порядка следования объектов (продукций отдельной стратегии).

СПИСОК ЛИТЕРАТУРЫ

1. Бибило, П. Н. Логическое проектирование дискретных устройств с использованием продукционно-фреймовой модели представления знаний / П. Н. Бибило, В. И. Романов / – Минск: Беларус. навука, 2011. – 279 с.
2. Шлее, М. Qt 5.3. Профессиональное программирование на C++ / М. Шлее // – СПб.: БХВ-Петербург, 2015. – 928 с.
3. JSON [Electronic resource] / Mode of access: <https://ru.wikipedia.org/wiki/JSON>. Date of access: 09.10.2019.

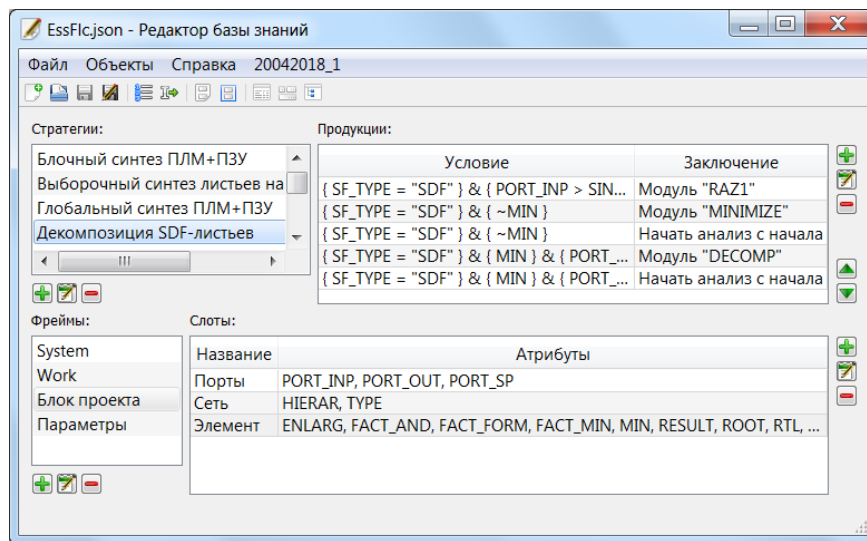


Рис. 1 – Основное рабочее окно подсистемы ESS-2

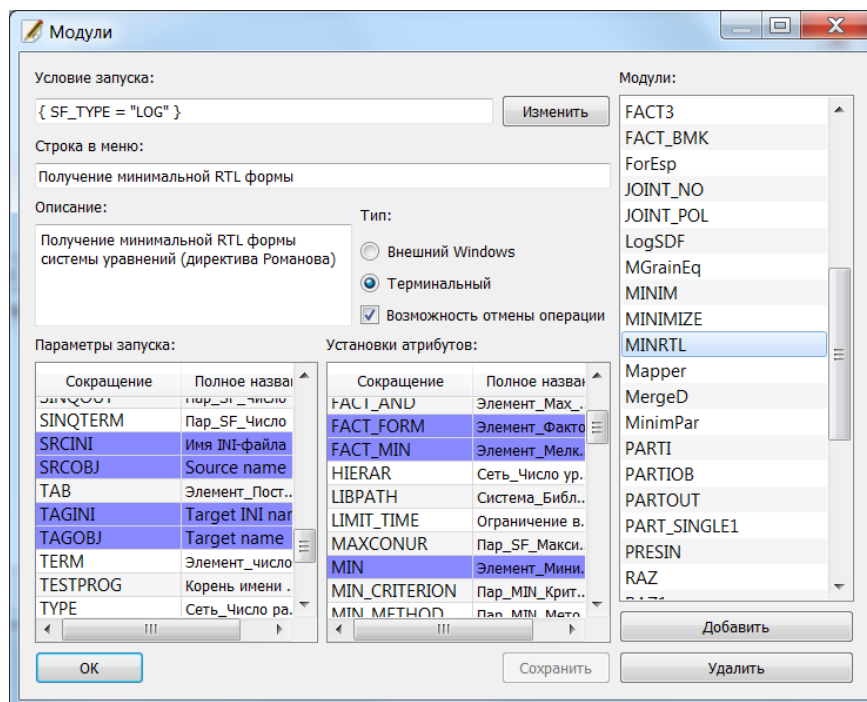


Рис. 2 – Диалоговая панель для установки свойств программного модуля