

МАЛЕНЬКИЙ КОНТРОЛЛЕР ДЛЯ БОЛЬШИХ ЗАДАЧ ИЛИ ПРОГРАММИРОВАНИЕ И СОСТАВЛЕНИЕ ПРОГРАММЫ ПЛК

М.В. Зайцев (ООО "Компэл-СПб")

Продолжаем серию статей, посвященных особенностям программирования контроллеров Phoenix Contact в универсальной среде PC WORX. Вторая статья посвящена составлению на языке функционально-блоковых диаграмм FBD программы для управления светофором согласно техническому заданию.

Ключевые слова: среда программирования, контроллер, имитатор.

В первой статье¹ были приведены основные возможности среды разработки PC WORX, для изучения которых начато написание простейшей программы "СВЕТОФОР". На первом шаге был выбран контроллер и настроено ПО для работы в режиме симулятора. Теперь составим на языке функционально-блоковых диаграмм FBD программу для управления светофором согласно техническому заданию. В качестве ПЛК используется контроллер Phoenix Contact серии InLine.

Техническое задание

1. Цель разработки программы: управление движением транспорта на Т-образном перекрестке.

2. Программа должна управлять включением соответствующих цветов транспортного светофора. Транспортный светофор оснащен красным (RED), желтым (YELLOW), зеленым (GREEN) огнями.

3. Для исключения возникновения аварийных ситуаций должны быть выполнены условия:

- предварительная (начальная) установка светофоров: горит желтый свет;
- требования к таймерам по длительности "горения" соответствующих цветов светофора – $t=10$ с;
- служба управления движением должна иметь возможность оперативно менять интервалы срабатывания светофоров.

Определим требуемые входные / выходные переменные (таблица).

Таблица

Имя переменной	Тип переменной	Описание
TimeInterval	TIME	Время горения комбинации сигналов светофора
AmberLight	BOOL	Сигнал "Внимание"
GreenLight	BOOL	Сигнал "Движение"
RedLight	BOOL	Сигнал "Стоп"
AmberLightSide	BOOL	Сигнал "Внимание" для бокового направления
GreenLightSide	BOOL	Сигнал "Движение" для бокового направления
RedLightSide	BOOL	Сигнал "Стоп" для бокового направления
RedLightTimer	TON	Время горения красного света
AmberLightTimer	TON	Время горения желтого света в комбинации с другим
GreenLightTimer	TON	Время горения зеленого света
AmberOnlyTimer	TON	Время горения только желтого света

¹ Зайцев М.В. Маленький контроллер для больших задач или программирование и составление программы ПЛК // Автоматизация в промышленности. 2010. №12.

Алгоритм функционирования конечного автомата представим в виде последовательности (рис. 1):

- предустановка: желтый сигнал на момент включения системы;
- запрещающий сигнал для прямого направления движения и разрешающий – для бокового (10 с);
- желто-красный сигнал для прямого направления движения и желтый для бокового (10 с);
- зеленый сигнал для прямого направления движения и красный для бокового (10 с).

После того как было определено техническое задание можно приступить к программированию контроллера.

Открываем ранее сохраненный проект и в окне Project Tree Window раскрываем закладку MainV. В этом окне необходимо создать те переменные, которые были определены в техническом задании ранее (рис. 2).

В менеджере проекта Project Tree Window раскрываем закладку Main, так как мы пишем программу на языке FBD, то на иконке схематично изображен значок функционального блока. В любой момент после компиляции программу можно конвертировать в другой язык LD, IL. В открывшемся окне начинаем создавать программу. Необходимо будет из существующего набора библиотечных функциональных блоков и функций выбрать требуемые функции. Для этого в меню View выбираем пункт Edit Wizard. Этот блок содержит все подключенные к проекту библиотеки функциональных блоков. Из существующих понадобятся блоки таймеров с задержкой на включение

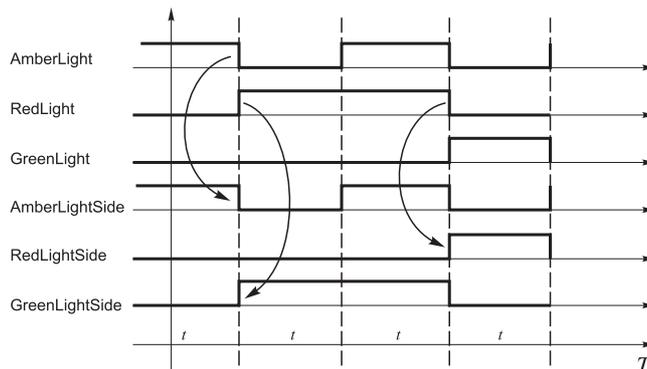


Рис. 1. Векторная диаграмма работы светофора

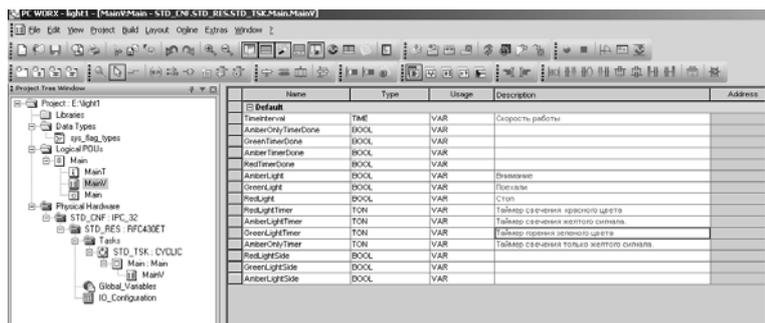


Рис. 2. Описание локальных переменных

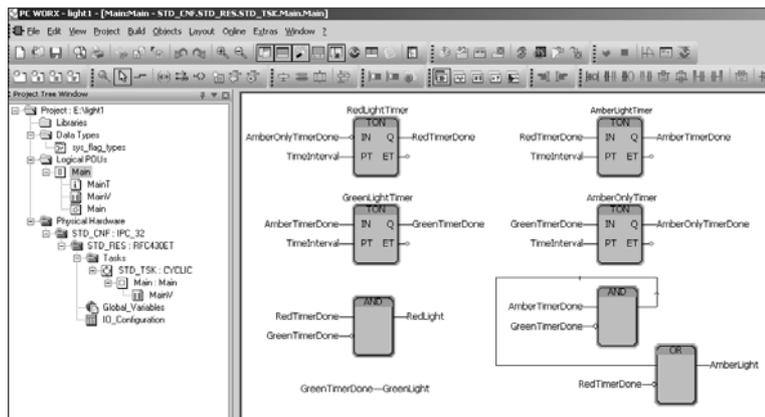


Рис. 3. Текст программы на языке FBD



Рис. 4. Окно отладки проекта

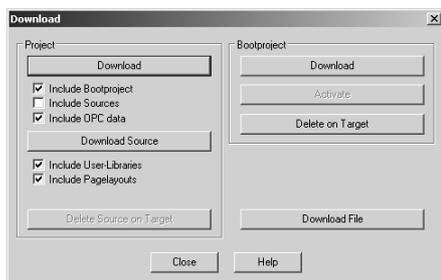


Рис. 5. Окно загрузки программы

(TON), функции логического сравнения "И" (AND) и "ИЛИ" (OR). В проекте вставляем четыре блока TON, два AND и один OR. Присваиваем им имена и создаем связи с переменными, которые описаны ранее. Окно редактирования должно выглядеть, как показано на рис. 3.

Для компиляции программы выбираем меню Build – Rebuild Project. Далее проверим правильность результатов компиляции, открыв пункт меню View – Message Window с сообщениями системы о результатах компиляции. Если в проекте обнаружены ошибки, то, кликнув дважды на соответствующей ошибке, откроется текст программы или описание переменных, где обнаружена данная ошибка.

Зайцев Михаил Вячеславович – ведущий инженер группы промавтоматика ООО "Компэл-СПб".

Контактный телефон (812)327-94-04 доб. 4270.

E-mail: zm.spb@compel.ru Http:// www.compel.ru

После устранения ошибок проект можно загрузить в контроллер на исполнение и проверить правильность составленного алгоритма: открываем меню OnLine – Project Control и организуем коммуникацию с контроллером. В примере выбрана система симулятора, поэтому подключать контроллер нет необходимости. Достаточно в открывшемся окне (рис. 3) нажать кнопку Download, которая загружает программу в контроллер в соответствии с коммуникациями, определенными в конфигурации контроллера. Перед загрузкой программы система предложит остановить контроллер, если он находится в режиме исполнения, и определить состав загружаемых в контроллер документов (рис. 5): откомпилированный проект (Include Bootproject); исходный код программы (Include Source); переменные, помеченные как OPC тэги (Include OPC data); пользовательские библиотеки (в проекте кроме использования стандартных библиотек FBD можно создавать собственные) (Include user-library); комментарии, структуру программы и другую технологическую информацию (Include pagelayouts).

Таким образом, можно сохранить полный проект со всеми комментариями и пояснениями в контроллер, а на объекте просто считать эту информацию из контроллера, не используя других накопителей. Кроме того, интересна секция Download File для создания образа проекта и сохранения его в файл, чтобы впоследствии можно было скопировать проект на контроллер через встроенный FTP-сервер.

Работая в режиме симулятора, достаточно выбрать только опцию загрузки откомпилированного проекта и нажать Download.

После загрузки в окне отладки проекта (рис. 4) нажимаем кнопку Cold и программа будет запущена на исполнение в режиме симулятора. Теперь можно посмотреть состояние сигналов светофора в проекте, выбрав режим монитора в меню Online – Debug. Просматривать переменные можно как в теле программы в режиме отладки, так и переместив их в окно монитора в меню View – Watch Windows. В мониторе для удобства сортировки и просмотра предусмотрены несколько листов (как в MS Excel), где можно перетаскивать контролируемые переменные, группируя их по определенному признаку.

В следующей статье завершим отладку проекта, расскажем о некоторых возможностях контроллера и ПО.