

МОДУЛЬНЫЕ ИЗМЕРИТЕЛЬНЫЕ КОНТРОЛЛЕРЫ СЕРИИ «К15» И ИХ ПРИМЕНЕНИЕ

Д.А. ГРИШИН (ООО «Эй энд Ти Текнолоджис»)



В статье на основе анализа рынка программируемых логических контроллеров разработана собственная концепция построения аппаратно-программных средств измерительных контроллеров (ИК «К15»). Рассмотрены особенности измерительных контроллеров серии «К15». Описаны функциональные возможности центральных процессорных устройств ИК «К15»: K15.MCU.F1, K15.CPU.F4, K15.CPU.H7 и примеры их использования. Представлены технические характеристики модулей ввода/вывода измерительных контроллеров «К15» и рассмотрены ключевые особенности их программирования. Приведены два примера практического программирования контроллеров, один с использованием языка промышленного программирования (Codesys), а другой на основе языка СИ.

Ключевые слова: ПЛК; измерительные контроллеры «К15»; K15.MCU.F1; K15.CPU.F4; K15.CPU.H7; датчики; горячая замена модулей; ЦПУ; FRAM; SD карта; шина T-Bus; CAN; Modbus; CubelDE; CodeSys.

ВВЕДЕНИЕ

Сегодня рынок автоматизации предлагает обширный перечень модульных и моноблочных программируемых контроллеров для решения широкого спектра задач, отличающихся как по сложности, так и по ценовому порядку. И каждый для себя выделяет более приоритетные изделия, отвечающие требуемым характеристикам, бюджету проекта либо же просто давно знакомые и проверенные, подтвердившие свою надежность за годы эксплуатации. И очень часто предпочтение отдается иностранному производителю¹.

Современные текущие тенденции диктуют новые условия, и российский потребитель вынужден все больше склоняться не к импортным решениям, а присматриваться к отечественным разработкам. Одной из таких разработок и является серия модульных программируемых измерительных контроллеров «К15» (ИК «К15»). Эти контроллеры российского производства предприятия ООО «Эй энд Ти Текнолоджис», г. Уфа предназначены для создания «легких» и «средних»

АСУ ТП, а также могут применяться в составе больших и сложных систем промышленной автоматизации.

Важно отметить, что измерительные контроллеры К15 внесены в Госреестр средств измерений РФ под номером 75449-19.

ОСОБЕННОСТИ ИЗМЕРИТЕЛЬНЫХ КОНТРОЛЛЕРОВ СЕРИИ «К15»

Модульные контроллеры этой линейки реализуют классическую концепцию ПЛК: центральное процессорное устройство (ЦПУ) плюс «корзина» модулей ввода/вывода. Такой подход, в отличие от моноблочных вариаций либо смешанных решений, дает возможность создать гибкую, масштабируемую локальную систему управления ровно под те задачи, которые необходимо решить в данный момент. Это позволяет не переплачивать за лишнее «железо», но в то же время иметь возможность практически бесшовно – просто докупив необходимые модули – расширить и усложнить систему при необходимости.

Еще один несомненный плюс такого решения – простота и дешевизна эксплуатации. Не нужно менять дорогостоящий ПЛК ради пары каналов ввода/вывода, вышедших из строя. Всего лишь меняем неисправный модуль – и система снова в работе.

¹ А.А. Егоров (Компания «АВИАТЭКС») «Индустрия 4.0», гиперавтоматизация и PLC – умное производство будущего. Проблемы и решения. Части 1, 2, 3, Журнал «Автоматизация и ИТ в энергетике», № 1(138), № 2(139), № 3(140), 2021 г.

Таблица 1. Сравнительный анализ по основным функциональным характеристикам процессорных модулей ИК серии “К15”

Характеристика	Процессорный модуль К15.MCU.F1	Процессорный модуль К15.CPU.F4	Процессорный модуль К15.CPU.H7
Центральный процессор	ЦП ARM® 32-bit Cortex®-M3, 72 MHz	ЦП ARM® 32-bit Cortex®-M4, 168 MHz	ЦП p ARM® 32-bit Cortex®-M7, 480 MHz
WEB-интерфейс	Нет	Есть	Есть
Ввод дискретных сигналов DIN	8	3	3
Вывод дискретных сигналов DOUT	4	2	2
Ввод аналоговых сигналов AIN	3	Нет	Нет
Интерфейс RS-485	1 шт.	3 шт.	2 шт.
Интерфейс Ethernet	Нет	Ethernet 10/100 Base-T	Ethernet 10/100 Base-T
Интерфейс CAN	Нет	1 шт.	1 шт.

ИК “К15” отличаются от других подобных решений хорошей эргономичностью: классическое крепление на DIN рейку вкупе с малой шириной модулей. Все модули, включая ЦПУ, имеют один форм-фактор, что существенно упрощает компоновку при проектировании шкафного оборудования. Также следует отметить удобное расположение интерфейсной шины, соединяющей модули с ЦПУ: она уложена непосредственно в DIN рейку и фиксируется в ней от выпадения. Это дает возможность менять модули, не разбирая “корзину” и даже не отключая питания.

Контроллеры измерительные К15 предназначены для измерений, преобразования, обработки, анализа входных аналоговых унифицированных измерительных сигналов (сила и напряжение постоянного тока, частота и количество электрических импульсов, сопротивление – выходной сигнал от термопреобразователя сопротивления, ТЭДС – выходной сигнал от термопар) и формирования выходных аналоговых сигналов (сила и напряжение постоянного тока, частота и количество электрических импульсов).

Контроллеры ИК “К15” оснащены индикаторным устройством для отображения измерительной информации, а также осуществляют прием дискретных электрических сигналов и обмен данными по цифровым интерфейсам.

К преимуществам ИК “К15” следует отнести следующие его характеристики:

- возможность горячей замены и высокая производительность;
- компактность – малогабаритные решения;
- низкая потребляемая мощность;
- ресурс не менее 50 000 часов;
- срок службы не менее 10 лет.

Очень важно отметить, что ИК “К15” выполнены в промышленном исполнении. Условия эксплуатации: температура – от –40 °С до +60 °С, влажность от 10 % до 90 %.

ЦЕНТРАЛЬНЫЕ ПРОЦЕССОРНЫЕ УСТРОЙСТВА ИК “К15”

Центральным и основным элементом измерительных контроллеров (ИК) серии “К15” безусловно является центральное процессорное устройство (ЦПУ). В настоящее время в линейке измерительных контроллеров серии “К15” присутствует три модели: F1, F4 и H7. Эти измерительные контроллеры отличаются в основном функциональными характеристиками, в первую очередь быстродействием и памятью и соответственно стоимостью.

Все ИК серии “К15” имеют возможность подключения дополнительных модулей ввода/вывода. Сравнительный анализ по основным функциональным характеристикам процессорных модулей ИК серии “К15” представлен в таблице 1.

Измерительные контроллеры серии “К15” призваны покрыть достаточно широкий спектр потребностей промышленной автоматизации.

Для интеграции с системами верхнего уровня, а также для построения распределенных систем все ИК серии “К15” реализуют широко распространенный протокол обмена Modbus, доступный через все имеющиеся интерфейсы. Важным дополнением к этому является возможность реализации нестандартных протоколов обмена.



Рис. 1. Процессорный модуль K15.MCU.F1

Таблица 2. Технические характеристики процессорного модуля K15.MCU.F1

Основные технические характеристики	
Центральный процессор ARM® 32-bit	
Cortex®-M3, 72 MHz	✓
Часы реального времени	✓
Возможность подключения дополнительных модулей ввода/вывода	✓
Электрические характеристики	
Напряжение питания	24 В ±20 %
Потребляемая мощность, не более	5 Вт
Электрическая прочность изоляции цепей	500 В
Защита входного напряжения	ограничение тока
Количество дискретных выходов	4 шт.
Количество дискретных входов	8 шт.
Количество аналоговых входов	3 шт.
Коммуникационные характеристики	
Изолированный порт RS-485	✓
Индикатор передачи данных по RS-485	✓
Индикаторы состояния (Status, Run, Fault)	✓
Индикаторы дискретных сигналов	✓
Механические характеристики	
Размеры (Д × Ш × В)	107 × 22,5 × 136 мм
Масса, грамм	400
Степень защиты корпуса	IP20
Крепление	DIN-рейка 35 мм
Условия эксплуатации	
Температура	от -40 до +60 °С
Влажность	от 10 до 90 %

Процессорный модуль K15.MCU.F1 имеет достаточно скромные характеристики и подойдет для небольших задач, где требуется изящное недорогое решение с сохранением всех преимуществ модульной схемы (рис. 1, табл. 2).

Например, процессорный модуль K15.MCU.F1 успешно используется для построения систем управления исполнительными механизмами с трехфазным асинхронным электроприводом. Применяется также в качестве управляющего контроллера блока управления переключателем потока БУ-ПП. Этот блок предназначен для управления мотор-редуктором переключателя потока. Допускается применение этого блока на других видах трубопроводной арматуры с трехфазным асинхронным электроприводом мощностью до 2 кВт. Имеет взрывозащищенное исполнение и может эксплуатироваться во взрывоопасных зонах. БУПП обеспечивает удаленное (RS-485) и ручное управление посредством кнопок по месту использования.

Процессорный модуль K15.CPU.F4 – более мощный, имеющий на борту Ethernet интерфейс. Помимо увеличенных тактовой частоты процессора, RAM и Flash, данное ЦПУ имеет встроенный веб-интерфейс. Он позволяет в свою очередь следить за состоянием корзины, ее составом, производить ее мониторинг и диагностику в реальном времени. Также веб-интерфейс дает возможность производить загрузку проекта в ЦПУ без применения программатора и обновлять программное обеспечение подключенных модулей ввода/вывода (рис. 2, табл. 3).

Модуль K15.CPU.F4 успешно используется для построения систем управления слива/налива нефтепродуктов. Построения блока дозирования реагентов (БДР). Применяется в качестве управляющего контроллера установки поверки поточных влагомеров нефти “Астра 0-100-1”.

Процессорный модуль K15.CPU.H7 это наиболее производительное ЦПУ из представленных. Имея хорошие характеристики процессора, а также возможность применения внешнего Flash-накопителя в виде SD карты, он способен реализовывать достаточно сложные алгоритмы с хорошим быстродействием. Также стоит отметить применение FRAM вместо EEPROM в качестве энергонезависимой памяти. Веб-интерфейс, конечно, также присутствует (рис. 3, табл. 4).

Например, процессорный модуль K15.CPU.H7 успешно используется для построения систем управления автоматизированной групповой замерной установкой (АГЗУ), а также малогабаритной сепарационной измерительной установкой (МСИУ).



Рис. 2. Процессорный модуль K15.CPU.F4



Рис. 3. Процессорный модуль K15.CPU.H7

Таблица 3. Технические характеристики процессорного модуля K15.CPU.F4

Основные технические характеристики	
Центральный процессор ARM® 32-bit	
Cortex®-M4, 168 MHz	✓
Web-интерфейс	✓
Часы реального времени	✓
Возможность подключения дополнительных модулей ввода/вывода	✓
Электрические характеристики	
Напряжение питания	24 В ± 20%
Потребляемая мощность, не более	5 Вт
Электрическая прочность изоляции цепей	500 В
Защита входного напряжения	ограничение тока
Количество дискретных выходов	2 шт.
Количество дискретных входов	3 шт.
Коммуникационные характеристики	
Порт Ethernet 10/100 Base-T	1 шт.
Количество изолированных портов RS-485	1 шт.
Количество не изолированных портов RS-485	2 шт.
Поддерживаемые протоколы обмена	ModBus RTU/TCP
Интерфейс обмена данными с модулями	CAN
Индикатор передачи данных по RS-485	3 шт.
Индикаторы состояния (Status, Run, Fault)	✓
Индикаторы дискретных сигналов	✓
Механические характеристики	
Размеры (Д × Ш × В)	107 × 22,5 × 136 мм
Масса, грамм	400
Степень защиты корпуса	IP20
Крепление	DIN-рейка 35 мм
Условия эксплуатации	
Температура	от -40 до +60 °С
Влажность	от 10 до 90%

Таблица 4. Технические характеристики процессорного модуля K15.CPU.H7

Основные технические характеристики	
Центральный процессор ARM® 32-bit	
Cortex®-M7, 480 MHz	✓
Web-интерфейс	✓
Поддержка MicroSD	✓
Часы реального времени	✓
Возможность подключения дополнительных модулей ввода/вывода	✓
Электрические характеристики	
Напряжение питания	24 В ± 20%
Потребляемая мощность, не более	5 Вт
Электрическая прочность изоляции цепей	500 В
Защита входного напряжения	ограничение тока
Количество дискретных выходов	2 шт.
Количество дискретных входов	3 шт.
Коммуникационные характеристики	
Порт Ethernet 10/100 Base-T	1 шт.
Количество изолированных портов RS-485	2 шт.
Количество не изолированных портов RS-485	1 шт.
Поддерживаемые протоколы обмена	ModBus RTU/TCP
Интерфейс обмена данными с модулями	CAN
Индикатор передачи данных по RS-485	3 шт.
Индикаторы состояния (Status, Run, Fault)	✓
Индикаторы дискретных сигналов	✓
Механические характеристики	
Размеры (Д × Ш × В)	107 × 22,5 × 136 мм
Масса, грамм	400
Степень защиты корпуса	IP20
Крепление	DIN-рейка 35 мм
Условия эксплуатации	
Температура	от -40 до +60 °С
Влажность	от 10 до 90%

Все ЦПУ работают под управлением ОСРВ, что повышает надежность системы, ее быстродействие и позволяет использовать аппаратные возможности процессоров на все сто процентов.

МОДУЛИ ВВОДА/ВЫВОДА ИЗМЕРИТЕЛЬНЫХ КОНТРОЛЛЕРОВ “К15”

Вот тут открывается простор для фантазии – линейка “К15” может похвастаться обширной номенклатурой различных модулей ввода/вывода. От банальных и простых дискретных модулей ввода и вывода, до гальванически развязанных аналоговых модулей вывода, счетных модулей, PWM модулей (ШИМ, англ. *pulse-width modulation*) и т.д.

Комбинируя модули в “корзине”, можно создавать абсолютно произвольные по сложности системы. Причем это могут быть как системы совсем без модулей, с одним модулем либо сложные вариации до 8 модулей с одним ЦПУ.

Ключевая особенность всех этих модулей – механизм взаимодействия с ЦПУ. Тут стоит остановиться немного подробнее. Как было сказано выше, между ЦПУ и модулями предусмотрена интерфейсная системная шина. Она выполнена на базе CAN интерфейса и физически представляет собой 5-контактную шину T-Bus, на которую устанавливаются ЦПУ и модули в процессе монтажа на DIN рейку. Так как шина расположена в задней части корпуса модулей со стороны рейки, появляется возможность без труда снимать и устанавливать модули без демонтажа всей “корзины”.

Но главное состоит в том, что модули ввода/вывода можно заменять даже на работающей системе, не отключая питания (так называемая “горячая замена”). Это бывает весьма критично там, где отключение и перезапуск локальной системы управления связаны с нарушением технологического процесса или серьезными производственными издержками.

В ближайшей перспективе планируется выпуск специальной серии модулей ввода/вывода “К15”, оснащенных интерфейсом RS-485, на котором будет реализован все тот же популярный протокол Modbus. Это позволит произвести буквально помодульное дооснащение имеющихся ЛСУ и РСУ без применения ЦПУ “К15” в тех проектах, где уже имеются ЦПУ, но не хватает каналов ввода/вывода.

КЛЮЧЕВЫЕ ОСОБЕННОСТИ ПРОГРАММИРОВАНИЯ ИЗМЕРИТЕЛЬНЫХ КОНТРОЛЛЕРОВ “К15”

В отличие от классических языков программирования МЭК[IEC], которые позволяют без глубоких знаний программирования создавать управляющие проекты для таких систем, “К15” предлагает создание проектов на широко распространенных языках высокого уровня C/C++. Стоит согласиться, что это не совсем популярный метод работы с модульными системами в сфере автоматизации. Хотя тут можно вспомнить, что в качестве скриптовых языков C, C++, C#, VBA используются в контроллерах Siemens, B&R, Allen Bradley, Schneider и других изделиях. Какие же преимущества дает создание проектов таким способом?

Во-первых, не стоит пугаться формулировки “язык высокого уровня”. В конце концов упомянутые C и C++ очень похожи на один из языков МЭК – ST (Structured Text). По сути это все те же переменные, циклы, условия, переходы. Но если ST позволяет реализовать лишь то, что заложено в Runtime-оболочке контроллера, которая является как бы исполняемой средой для нашего проекта, то язык Си позволяет заглянуть за рамки этой среды. Нам доступен весь синтаксический инструментарий этого мощного орудия программирования. К тому же многие алгоритмические приемы на C/C++ реализуются проще и понятнее, а создание структур, приведение типов, callback функции, перегруженные функции и прочие приемы программирования – приятный бонус использования языка.

Во-вторых, немаловажное значение играет инструментарий разработки. Далеко не все среды разработки проектов под те или иные модульные системы легки и понятны в освоении. Например, описание работы со средой TIA Portal занимает более 1000 страниц руководства программиста. Не менее сложна в освоении и Studio 5000 для контроллеров Allen Bradley серий Control и CompactLogix. А чтобы уверенно работать в CodeSys, нужен не один месяц освоения среды.

Семейство “К15” программируется в среде разработки CubeIDE – это официальная свободно распространяемая IDE от компании STMicroelectronics, чьи процессоры и являются главным элементом контроллеров на данный момент. В комплекте с контроллерами

идет стартовый проект и подробная инструкция с описанием работы. Просто открываем проект в CubeIDE, компилируем его и “отправляем” в контроллер. Система уже будет работать, отображать веб-интерфейс, опрашивать модули. Ну, а затем, используя описанные структуры и функции, можно реализовать задуманное (рис. 4, рис. 5).

В-третьих, доступность среды разработки и ценовой аспект. Если кто-то приобрел лицензии для работы с контроллерами Emerson, Siemens, V&R, Yokogawa и многих других – знают, что это порой немалая сумма. Да, возможно, читатель возразит: а как же CodeSys, Veremiz, OwenLogic и прочие системы? Да, они не требуют приобретения лицензий, но и далеко не все модульные системы ими поддерживаются. Как правило, каждый производитель старается разработать свой инструментарий как платформозависимый программный продукт либо как свою экосистему. Это, например, DeltaV, TIA Portal, FactoryTalk и прочие. И если до недавнего времени все эти программные продукты можно было хотя бы свободно приобрести, то, учитывая повальный уход с рынка зарубежных игроков, это уже будет сделать весьма затруднительно. В противовес этому тем более привлекательно решение “К15” – использовать доступную среду CubeIDE, не требующую лишних затрат на приобретение лицензий.



Рис. 4

Ну, и, наконец, процесс отладки проекта. Это то, что непосредственно влияет на скорость разработки. В чем одна из ключевых особенностей языков МЭК – механизм он-лайн трассировки и отладки проекта. “К15” также может похвастать этим функционалом, который изначально заложен в самой IDE. Но отладка здесь, как и подобает любому языку высокого уровня, гораздо более многосторонняя и глубокая. Тут и классический вывод текущих переменных, и точки останова, и принудительная запись значений, и пошаговое исполнение кода. Также есть возможность вернуться на шаг назад либо выполнить остановку исполнения кода по условию. Дополнительным удобством для разработчика можно считать контекстную подсветку значений переменных при наведении курсора.

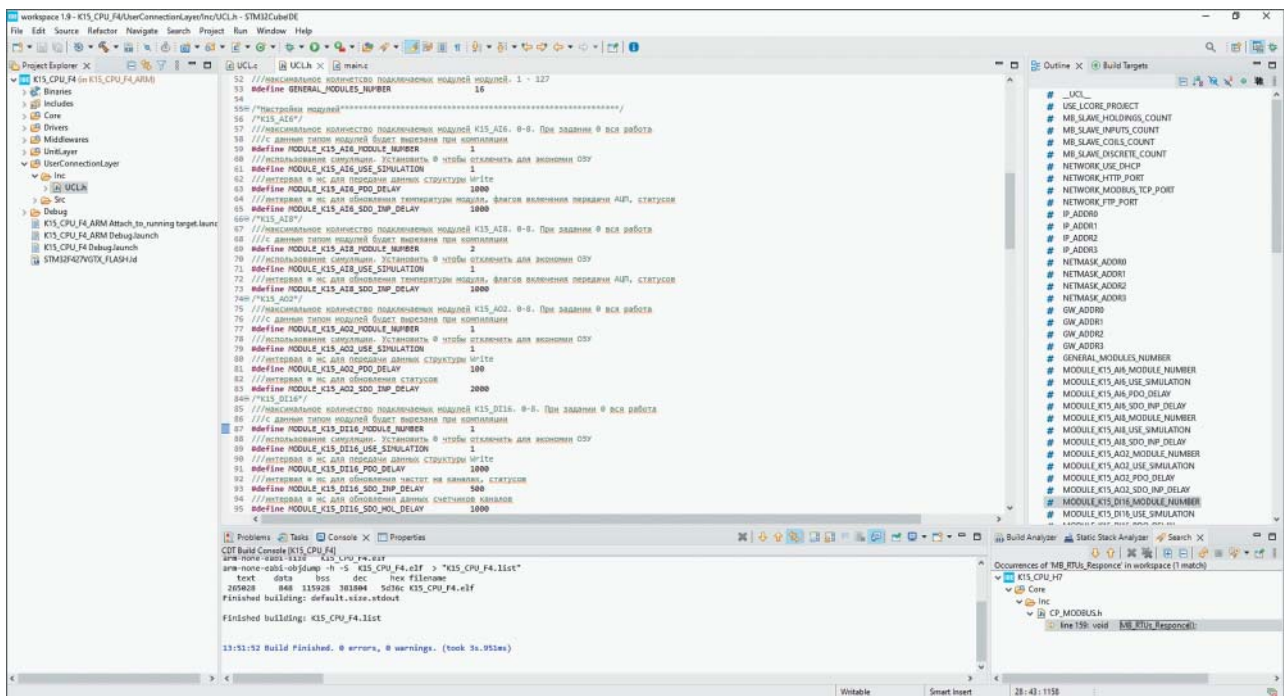


Рис. 5

Рис. 6.

start_DI – сигнал пуска системы (например, кнопка)

stop_DI – сигнал остановки системы

foto_DI – фотодатчик наличия пламени

iskra_DO – управление подачей искры

valve_DO – управление клапаном

```

PLC_PRG (PRG-FBD)
0001 PROGRAM PLC_PRG
0002 VAR
0003 (*сигналы ввода-вывода*)
0004 start_DI: BOOL;
0005 stop_DI: BOOL;
0006 foto_DI: BOOL;
0007 iskra_DO: BOOL;
0008 valve_DO: BOOL;
0009
0010 (*флаги*)
0011 iskra: BOOL;
0012 try: BOOL;
0013 try_out: BOOL;
0014
0015 (*функциональные блоки*)
0016 rtrig1: R_TRIG;
0017 rtrig2: R_TRIG;
0018 rs1: RS;
0019 ton1: TON;
0020 ctu1: CTU;
0021 ftrig1: F_TRIG;
0022 rtrig3: R_TRIG;
0023 tp1: TP;
0024
0025 END_VAR
    
```

Одним словом, вся мощь языков C/C++, к Вашим услугам. Это делает разработку быстрее и качественнее, а дальнейшее сопровождение и рефакторинг проекта – дешевле для конечного потребителя. Если Вы – убежденный сторонник языков программирования МЭК, то, конечно, переход на такой способ разработки нельзя назвать совсем простым. Но некоторые усилия, потраченные на изучение языков Си, по крайней мере в рамках поставленных задач, с лихвой окупятся в дальнейшем.

ПРИМЕР РЕАЛИЗАЦИИ ЗАДАЧИ: ОТ СЛОВ – К ДЕЛУ

Рассмотрим, как решаются прикладные задачи средствами разработки измерительных контроллеров “К15”. Возьмем вполне конкретный пример. Допустим, у нас есть факельная установка с электроискровым розжигом дежурной горелки и фотодатчиком наличия пламени. Наша задача: реализовать режим автоматического розжига дежурной горелки.

При подаче сигнала пуска должен открываться клапан топливного газа, а розжиг должен происходить циклично до тех пор, пока либо не загорится пламя, либо не будет сделано 3 попытки розжига. Также алгоритм должен обеспечивать автоматический перерозжиг горелки в случае погасания пламени. Что ж, перейдем к реализации.

Сначала попытаемся реализовать задуманное средствами Codesys. Пускай это будет всеми любимый язык программирования МЭК – FBD. Создадим необходимые переменные и функциональные блоки (рис. 6).

Затем создаем функциональную схему, реализующую алгоритм (рис. 7).

При подаче сигнала пуска через детектор переднего фронта срабатывает триггер **rs1**, принимая логическое состояние 1. Открывается клапан, далее сигнал следует на блок **AND**. Так как флаг совершения попытки розжига **try** и сигнал с фотодатчика имеют логический сигнал 0, то инверсивный сигнал на входе блока имеет логический сигнал 1, и происходит подача искры через таймер импульсного сигнала

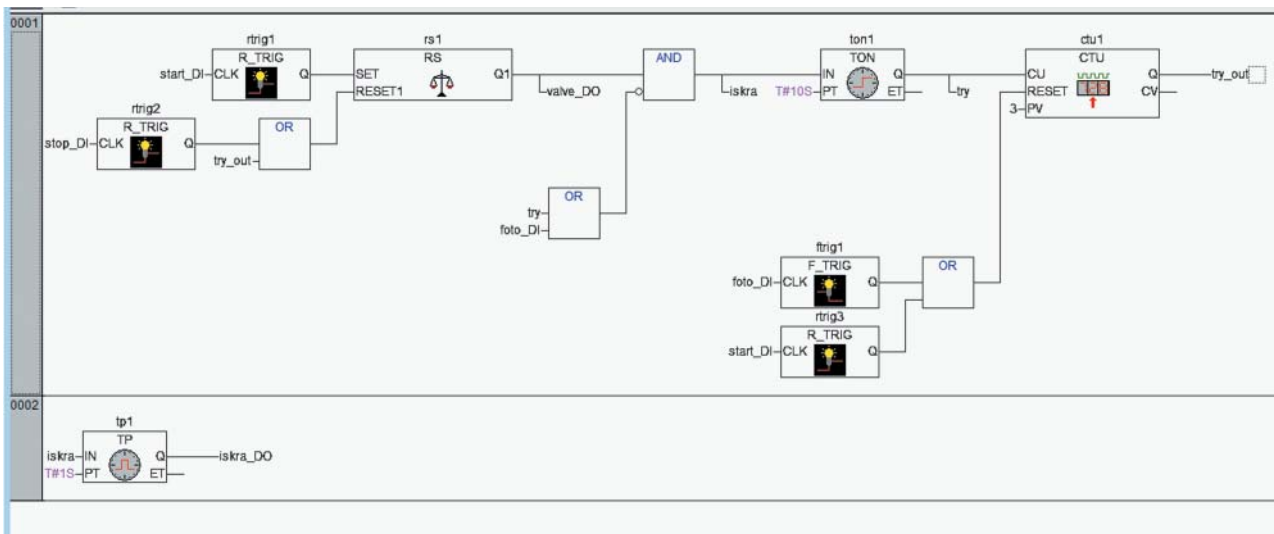


Рис. 7. Функциональная схема, реализующая алгоритм

ла **tp1** во второй строке схемы. Одновременно с этим запускается таймер ожидания пламени **ton1**. По завершению его счета, спустя 10 секунд выставляется флаг совершения попытки. Вместе с этим счетчик **ctu1** производит инкрементирование числа попыток на 1, сравнивая полученное число с уставкой, равной трем. При достижении трех попыток взводится флаг **try_out**, который вызывает сброс триггера **rs1**. Также его сброс вызывает и подача сигнала остановки системы розжига.

При установке флага **try** происходит изменение значения флага **iskra** сначала на 0, затем вновь на 1. Тем самым запускается новый цикл розжига. Здесь следует иметь в виду, что переменная **try** должна быть объявлена как глобальная и не сбрасываться при каждом цикле скана используемого **POU**.

В случае горения флаг **foto_DI** принимает логическое состояние 1. Благодаря этому через инверсию блок **AND** на выходе принимает логическое состояние 0, блокируя цикл розжига. Если пламя гаснет, блок **AND** снова принимает состояние 1, и цикл розжига начинается снова.

```
bool start_DI;
bool stop_DI;
bool foto_DI;
bool iskra_DO;
bool valve_DO;

uint16_t state, count, try_count;
```

▲ Рис. 8

Следовательно, для реализации проекта потребовалось создать, помимо простых переменных, еще 8 функциональных блоков, что для такой простой задачи немало.

Теперь попробуем решить ее в рамках программного функционала “K15”. Весь код размещаем в файле **UCL.c**. Здесь также не обойтись без объявления переменных (рис. 8).

Для единообразия, сигналы имеют схожее обозначение. Дополнительно добавились переменные **state** (стадия работы системы), **count** (аккумулятор счетчика времени) и **try_count** (аккумулятор числа попыток розжига). Далее обратимся к самому коду (рис. 9).

```
67
68
69
70 switch(state){
71 case 0: //стадия СТОП
72     valve_DO=false;
73     iskra_DO=false;
74     count=0;
75     try_count=0;
76     if(start_DI){state=1;};
77     break;
78 case 1: //стадия РОЗЖИГ
79     valve_DO=true;
80     if(count<1000){iskra_DO=true; //подача искры 1 сек
81         osDelay(1);
82         count++;
83         }else{iskra_DO=false;};
84     if(count>=10000){count=0; //ожидание пламени 10 сек
85         try_count++;
86         };
87     if(foto_DI){state=2;};
88     if(stop_DI || (try_count==3)){state=0;}; //3 попытки
89     break;
90 case 2: //стадия ГОРЕНИЕ
91     valve_DO=true;
92     iskra_DO=false;
93     if(!foto_DI){count=0;
94         try_count=0; //сбросить число попыток
95         state=1;
96         };
97     if(stop_DI){state=0;};
98     break;
99     };
100
```

◀ Рис. 9

Алгоритм выполняется в основном цикле файла. Работа системы разбита на стадии: **СТОП**, **РОЗЖИГ** и **ГОРЕНИЕ**. Код достаточно компактный и читаемый. Создание дополнительных функциональных блоков не требуется. Следует обратить внимание, что число стадий может быть легко увеличено, например, можно добавить стадию **АВАРИЯ**.

В целом, конечно, проект небольшой, задачи весьма тривиальны, но даже тут читатель сможет заметить хороший потенциал использования “К15” в плане программной реализации, простоту сопровождения и масштабирования проекта.

КТО ПРОТЯНЕТ РУКУ ПОМОЩИ?

Любая современная разработка не может обходиться без постоянного совершенствования, а также без исправления неизбежно существующих текущих недостатков. Если продукт “живой”, он нуждается в неусыпном надзоре его непосредственных создателей. “К15” полностью соответствует этому принципу. Более того, постоянно собирается обратная связь от клиентов, разрабатывающих свои проекты на “К15”, а также конечных потребителей, эксплуатирующих эти изделия у себя на объектах. Затем, путем коллективного анализа пожеланий наиболее критические из них сразу же идут “под карандаш” разработчиков. Менее важные перманентно внедряются в последующих релизах. Это касается как программной части, так и аппаратного оснащения. Например, увеличение скорости работы FRAM, асинхронное чтение регистров Modbus, реализация пользовательских протоколов и многое другое – это то, что хотели видеть потребители в наших контроллерах.

Как сказано выше, бывают и неполадки. Но, как говорят, их нет лишь у того, кто ничего не делает. Тут важна оперативная реакция и своевременное их устранение. С этим тоже все в порядке. Любая неполадка или возникший вопрос обрабатываются в течение 24 часов квалифицированными инженерами, которые без проблем

проконсультируют клиента и предложат варианты решения проблемы. Ну, и, разумеется, не останутся без внимания и те, кто столкнулся с трудностями создания проектов – программисты также вникнут в проблематику и поделятся своим опытом решения.

ЗАКЛЮЧЕНИЕ

Измерительные контроллеры “К15” реализуют не только классическую модель ПЛК. Здесь мы видим несколько иной, альтернативный подход к реализации модульных систем. Перекрывая большинство задач, которые решают обычные модульные ПЛК, данная линейка дает разработчику нечто большее, позволяет вывести свои проекты на качественно иной уровень.

Линейка “К15” – отличная отечественная альтернатива в сфере автоматизации как промышленного, так и других сегментов, требующих надежных и экономичных решений. Это постоянно развивающийся динамичный продукт, способный составить достойную конкуренцию ушедшим вендорам, является экономически эффективным по стоимости владения, эргономичным и надежным решением отвечающим необходимым стандартом качества. Срок производства серийных моделей “К15” составляет 30 дней. Стандартный гарантийный период 24 месяца с момента поставки. Расширенный гарантийный период – до 48 месяцев с момента поставки.

Дополнительную техническую информацию можно получить по E-mail: grishin@a-t-tech.ru

По вопросам сотрудничества обращайтесь по телефону 8 (800) 775-74-70 или по E-mail: info@custom-eng.ru

<https://custom-eng.ru/>

Гришин Дмитрий Анатольевич – инженер-разработчик компании ООО “Эй энд Ти Текнолоджис”.