



УПРОЩЕННЫЕ ИНСТРУМЕНТЫ МЭК ПРОГРАММИРОВАНИЯ – УДЕЛ ПОЛЬЗОВАТЕЛЯ ПЛК?

И.В. Петров (Компания ПРОЛОГ)

Программист ПЛК любой ценовой категории вправе требовать от изготовителя предоставления ему инструментов программирования высшего класса, поддерживающих передовые достижения компьютерных технологий. Рассматриваются особенности использования существующих инструментов программирования контроллеров и перспективы их развития.

С каждым годом при выборе ПЛК пользователи все большее внимание уделяют возможностям среды программирования. Наряду с надежностью и ценой контроллера этот фактор неизменно включается в тройку важнейших. Объяснить это несложно. С ростом вычислительной мощности ПЛК расширяется круг решаемых им задач. 10 лет назад поддержка языка релейно-контактной логики (LD) или функциональных блокковых диаграмм (FBD) с подобранным изготовителем набором стандартных программных компонентов считалась вполне достаточной. Сейчас это уже не может удовлетворить пользователей. Контроллерные программы, включающие многомерную интерполяцию, адаптивное регулирование, операторский Web-интерфейс – это повседневная реальность.

Несколько особняком стоят так называемые программируемые реле (например, Siemens Logo!, Moeller Easy). Это миниатюрные контроллеры, имеющие автономную панель программирования и индикации. Упрощенные средства программирования считаются здесь плюсом. Однако, если на предприятии собирается десяток типов ПЛК разных классов, то возникает естественное желание иметь одну стандартную систему программирования вместо десяти. Показательно, что в современных классификациях ПЛК появилось понятие "нано-контроллер" [1]. Такие крошечные контроллеры снабжаются стандартными средствами программирования, отличаясь от старших собратьев только размерами и числом входов/выходов.

С ПЛК происходит то, что уже случилось с ПК. На заре их производства каждый изготовитель самостоятельно разрабатывал ОС и собственные инструментальные средства. Можно представить, что получится, если современный изготовитель материнских плат или корпусов возьмется за разработку ПО. К сожалению, на рынке ПЛК это пока возможно.

Разработать и поддерживать современную среду программирования для ПЛК – это очень большая и сложная работа, это отдельный вид бизнеса. Изготовителей высококлассных компиляторов и интегрированных инструментальных комплексов для компьютеров можно пере-

считать по пальцам и это неудивительно. Такая же ситуация формируется и на рынке ПО для контроллеров. Ведущие изготовители ПЛК понимают эту тенденцию. Большинство из них активно реализуют поддержку лидирующих комплексов, обеспечивая тем самым мощное конкурентное преимущество своей продукции.

Итак, сформулируем тезис: современный инструмент программирования ПЛК не должен уступать по функциональным возможностям, уровню сервиса, надежности и качеству создаваемого кода продуктам мировых лидеров рынка ПО для ПК.

Рассмотрим данные требования подробнее, используя в качестве примера комплекс МЭК 61131-3 программирования CoDeSys (рис. 1). Взяв его за точку отсчета, пользователи смогут оценить уровень других инструментов.

Языки программирования МЭК 61131-3

Несмотря на более чем 10-летнюю историю, языки стандарта МЭК 61131-3 продолжают вызывать оживленные дискуссии. С одной стороны, стандарт позитивен.

Пользователям ПЛК он снижает затраты на обучение специалистов, приобретение инструментов, существенно облегчает переход на технику разных производителей. Изготовителю ПЛК стандарт позволяет снять огромнейшую проблему разработки и продвижения на рынке собственного языка программирования. Используя известный универсальный комплекс МЭК 61131-3, молодая динамичная компания может реально конкурировать с крупными изготовителями.

Сосредоточившись на развитии уникальных качеств своих ПЛК, она может строить успешный бизнес.

С другой стороны, рекомендательный характер стандарта МЭК привел к появлению достаточно большого числа систем программирования начального уровня. Если мы сделаем графический редактор, позволяющий соединять функциональные блоки, разместим в памяти контроллера исполняемый код стандартных МЭК блоков, то такой инструмент вполне можно назвать системой МЭК программирования. Три профессиональных программис-

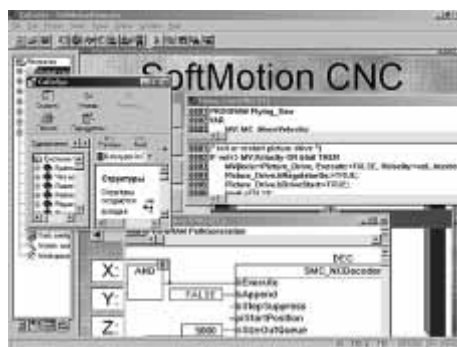


Рис. 1. Среда программирования CoDeSys

та способны сделать его за год. Вполне возможно, что для большинства несложных проектов этого инструмента будет достаточно. Но вскоре программист ПЛК столкнется с необходимостью ввести новые расширенные блоки. Тут ему предстоит узнать о необходимости применять внешний С компилятор, который нужно купить (и освоить) вместе со специальным дополнительным комплектом разработчика. Это означает, что его инструмент МЭК программирования неполноценен. Шаг влево, шаг вправо, шаг вперед — нужно использовать язык С. В итоге оказывается, что удобнее просто полностью перейти на язык С, обладающий самодостаточностью. Естественно, это поражает соответствующее отношение к стандарту МЭК вообще, и в первую очередь среди профессиональных программистов.

Ситуацию усугубляет несколько устаревшая точка зрения, что ПЛК — это устройство, программируемое исключительно непрофессионалами. Исходя из этого, в инструментах МЭК программирования делается упор на поддержку языков FBD и LD (как наиболее простых для начинающих). Наши исследования показали, что в 2005 г. до 70% пользователей CoDeSys использовали язык ST [2]. Причем их число постоянно растет. Как правило, это люди, имеющие высшее образование.

В целом, стандарт МЭК дает те же плюсы, что и стандартизация органов управления в автомобиле. Если машина не едет быстрее 40 км/ч, то это не означает, что все машины таковы. Имеющиеся у водителя навыки позволяют выбросить эту машину и купить новую. Стандарт МЭК не делает сами по себе инструменты программирования ПЛК лучше. Они по-прежнему очень разные. Оценивать возможности МЭК систем в целом нельзя.

Комплекс CoDeSys выделяется тем, что он изначально задумывался как инструмент для профессионалов. Он содержит встроенный компилятор быстрого машинного кода. Языки программирования полностью реализуют требования стандарта МЭК и сверх того имеют ряд расширений. Стандарт МЭК требует однозначного исполнения описанных в нем конструкций. Но запрета на расширение их числа нет. Практически все, что можно написать на языке С, в CoDeSys можно написать на МЭК языках. Это касается работы с указателями, доступа к аппаратным ресурсам, обработки прерываний, управления процессами, работы с файлами, потоками и сокетами. Изготовитель ПЛК может закрыть использование части этих возможностей по различным соображениям, но в самой системе ограничений нет. Все эти расширения опциональны и не мешают работе начинающих программистов.

Переносимость

Модель языков МЭК предполагает отсутствие прямых обращений к аппаратным ресурсам. При-

кладные программы работают с образом входов/выходов, который расположен в оперативной памяти и обслуживается системой исполнения контроллера. В итоге МЭК программы не могут работать без поддержки системного ПО. В ряде комплексов МЭК программирования система исполнения включает интерпретатор некоторого промежуточного языка. CoDeSys непосредственно генерирует машинный код. Этот код не включает начального загрузчика, планировщика задач, драйверов ввода/вывода, отладочных функций, протоколов связи и прочих аппаратно зависимых функций. Система исполнения представляет собой специализированную переносимую ОС. В результате, генерируемый код пригоден для исполнения на любых одноплатных процессорах.

В МЭК программах можно напрямую обращаться к образу входов/выходов по прямым адресам. Но удобнее использовать символические имена переменных, соответствующие входам/выходам. Их привязку к аппаратным ресурсам задает конфигуратор ПЛК (конфигуратор сети — для распределенных систем). Это визуальный ин-

струмент, опирающийся на конфигурационные файлы целевой платформы, поставляемые изготовителем ПЛК (рис. 2). Программист ПЛК пишет программу, не задумываясь об этих деталях. Перенос проекта на другой контроллер приведет только к необходимости перенастроить конфигурацию ПЛК. Конечно, это касается только переноса проекта CoDeSys на другой тип ПЛК. Но даже здесь возможны затруднения. Любой оригинальный

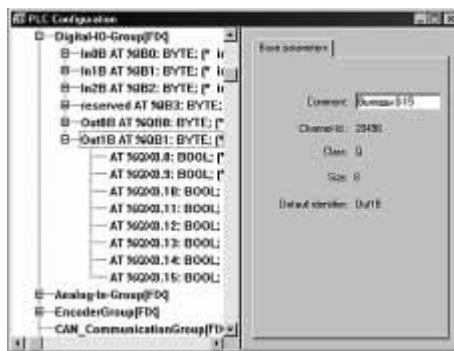


Рис. 2. Конфигуратор ПЛК

ПЛК имеет некоторые особенности, и программист старается их использовать. Однако на практике замена ПЛК, как правило, вызывается необходимостью проведения более значительных доработок.

Перенос проектов из одной среды МЭК программирования в другую существенно сложнее. Единого стандарта на формат файлов проектов нет. Впрочем, его нет и в интегрированных средах программирования на С. В простейшем случае, текстовые программы можно копировать через буфер обмена Windows. В CoDeSys программы на языках IL, ST и SFC можно экспортировать и импортировать в Common Elements format. В апреле 2004 г. PLCopen подготовлен документ XML Formats for IEC 61131-3 [3]. Данный формат позволяет сохранять программы на текстовых (IL, ST) и графических (LD, FBD) языках, включая информацию о расположении и соединениях графических элементов; комментарии; описания программ, функций и функциональных блоков; типов данных; структуру и конфигурацию проекта. В CoDeSys предусмотрено опциональное использование системы управления версиями прикладных проектов (ENI). В ней уже используется XML. Насколько широко данный формат будет поддерживаться другими изготовителями систем МЭК программирования пока прогнозировать трудно.

Надежность

Это наиболее сложный параметр при выборе системы МЭК программирования. Всегда есть опасения, что помимо собственных программных ошибок среда программирования внесет свою лепту. Особенно это касается компиляторов, которые относятся к сложнейшим программным продуктам. Несмотря ни на какие рекламные заявления, специалисты всегда ориентируются на авторитет компании, ее историю, известность и список практических применений. Что касается CoDeSys, то на сегодняшний день он используется 190 компаниями, большинство которых — производители систем промышленной автоматизации. Из изготовителей контроллеров, в которых штатно используется CoDeSys, в России наиболее известны ABB, Beck IPC, Beckhoff, ifm, Elrest, Kontron, Lenze, Moeller, WAGO, Fastwel и Овен.

Диапазон применений

Очевидно, диапазон применений инструмента МЭК программирования для разных ПЛК определяется его способностью к адаптации. Немалую роль играет и лицензионная политика.

В CoDeSys среда программирования универсальна. Чтобы она смогла генерировать код для конкретного ПЛК, к ней необходимо подключить специальные файлы конфигурации целевой платформы (TSP). В них заданы тип процессора, распределение памяти и другие сведения об аппаратуре. Такие файлы формируются изготовителем ПЛК. На сегодняшний день встроенный компилятор CoDeSys V2.3 имеет генераторы кода микропроцессоров: Intel 8051, Intel 80x86/80186/Pentium, ARM, MIPS, Motorola MC68000/MC68332/ColdFire, PowerPC, Hitachi SH 2/3/4, H8, Infineon C16x, Infineon TriCore, Texas Instruments TMS32028x.

Система исполнения поставляется изготовителям в виде исходных текстов. Существует несколько вариантов, оптимизированных для разных платформ. Наиболее простые системы могут работать в ПЛК без ОС. Самая совершенная система исполнения CSP32F функционирует только с многозадачными ОС и обеспечивает вытесняющую многозадачность в МЭК проектах.

Для PC-совместимых контроллеров и промышленных компьютеров существуют готовые SoftPLC, которые поставляются в виде исполняемых файлов с комплектом драйверов для распространенных плат ввода/вывода и fieldbus. Драйверы для новых устройств можно создать с помощью SDK, включенного в комплект поставки. Особое место занимает CoDeSys SP RTE. Это система исполнения с собственным ядром жесткого PB под Windows NT/2000/XP. Она гарантирует детерминированное время реакции с точностью в пределах микросекундной области без изменения ОС и дополнительной аппаратуры.

Среда программирования CoDeSys (в оригинальном виде) поставляется бесплатно. Лицензируются только системы исполнения. Многие изготовители сразу включают лицензию в комплект поставки ПЛК. Таким образом, покупатель ПЛК просто получает CD с CoDeSys и не должен ничего более приобретать.

Интегрированный комплекс

Современная система МЭК программирования — это всегда интегрированный комплекс, включающий редакторы языков, загрузчик, отладчик, средства архивации, документирования и др.

CoDeSys включает все общепринятые в известных комплексах программирования инструменты отладки [2]. Отдельно стоит выделить менеджер рецептов и графическую трассировку. Рецепты позволяют запоминать и восстанавливать состояние проекта. Таким образом, можно детально исследовать сложные дефекты, имеющие неустойчивые проявления. Графическая трассировка CoDeSys очень похожа на исторические тренды в SCADA-системах. Основное отличие состоит в накоплении данных непосредственно в памяти контроллера. Это дает возможность исследовать не только медленные ТП, но и любые изменения переменных синхронно с циклом работы прикладных программ.

Большой интерес вызывает встроенный в CoDeSys инструмент визуализации (HMI). Он позволяет размещать на экранной форме геометрические фигуры, растровые картинки, стрелочные и столбчатые индикаторы, гистограммы, тренды и таблицы тревог (alarm). Параметры элементов связываются со значениями переменных проекта. Этот инструмент широко применяется не только для операторского управления, но и при моделировании объектов управления. Встроенный эмулятор ПЛК позволяет построить визуальное представление объекта, моделируемого отдельной задачей, и управляющую задачу на единственном компьютере. Готовую визуализацию можно отображать: 1) на компьютере (CoDeSys HMI), 2) через Web-браузер в сетях TCP/IP, 3) на панели ПЛК (если она есть).

Из ряда дополнительных компонентов CoDeSys наиболее интересен пакет SoftMotion. Это интегрируемый в среду CoDeSys функциональный набор средств управления движением от простейших перемещений по одной оси до сложной многомерной интерполяции для систем ЧПУ. Обычно в подобных задачах применяются специализированные контроллеры. CoDeSys SoftMotion работает в обычных ПЛК (32-х разрядных) и промышленных PC.

Перспективы развития

Принято считать, что программисты ПЛК консервативны. Но это не совсем так. Посмотрите любую Internet конференцию, посвященную АСУТП и программированию ПЛК. Здесь ведутся живейшие обсуждения новейших методологий программирования. В первую очередь это касается объектно-ориентированного программирования (ООП). Многие используют язык C++ для программирования контроллеров, даже несмотря на существенные сложности поддержки аппаратуры и сопровождения проектов в сравнении с МЭК средами. Другие пытаются использовать приемы ООП в МЭК системах, преодолевая отсутствие необходимых свойств в языках ПЛК. Единственное, что сдерживает применение ООП в ПЛК — это отсутствие удобных инструментов [4].

Заключение

С одной стороны, ООП расширения в МЭК языках необходимы. С другой стороны, необходимо сохранить возможность работы со "старыми" проектами. Кроме того, нельзя навязывать программистам какую парадигму программирования им использовать. Компания 3S первой решилась на попытку решения столь противоречивой задачи. Путем "косметической" доработки CoDeSys этого сделать нельзя. В результате было решено разрабатывать совершенно новый инструмент, получивший наименование CoDeSys 3.0. Он уже поддержан несколькими европейскими изготовителями ПЛК, и осенью 2006 г. ожидается презентации новых оригинальных контроллеров.

CoDeSys 3.0 включает большое число интересных новшеств. При этом полностью сохранена совместимость на уровне прикладных проектов. Все расширения языков опциональны. Стандартные МЭК программы будут работать в этой среде обычным образом. Сверх того, новый инструмент поддерживает профили версий. Это означает, что все обновления компонентов комплекса устанавливаются не взамен старых, а параллельно с ними. То есть, если открыть проект, сделанный несколько лет назад, то CoDeSys спросит, должен ли он использовать в точности те версии компилятора и библиотек, в которых он создавался или можно использовать новейшие компоненты. Это существенно упрощает модификацию ПЛК программ, которые, как правило, используются годами.

Критика языков стандарта МЭК вызвана в первую очередь низким качеством инструментов программирования. Упрощенные ограниченные инструменты программирования ПЛК являются следствием недооценки изготовителями ПЛК уровня квалификации программистов контроллеров и сложности задач, которые им приходится решать. Стандартные инструменты программирования ПЛК высшего класса становятся доступны для простых контроллеров низшей ценовой категории. Лучшие современные системы программирования ПЛК превышают по сложности системы программирования офисных ПК. Очевидно, специализация компаний на изготовителях ПЛК и инструментов программирования будет усиливаться. Стандартные языки программирования МЭК могут и должны развиваться. Можно надеяться, что наиболее удачные расширения будут со временем стандартизированы.

Список литературы

1. Хоске М.Т. Микро-ПЛК // Control engineering Россия. 2006. № 1.
2. Петров И., Вагнер Р. Отладка прикладных ПЛК программ в CoDeSys (часть 3) // Промышленные АСУ и контроллеры. 2006. № 4.
3. PLCopen Technical Committee 6. XML Formats for IEC 61131-3. Released for Comments.
4. Хесс Д. Объектно-ориентированные расширения МЭК 61131-3 // Современные технологии автоматизации. 2006. № 2.

Петров Игорь Викторович — технический директор компании ПРОЛОГ.

E-mail: i.petrov@prolog-plc.ru

Http:// www.prolog-plc.ru

Протоколирование данных с помощью микро-ПЛК Simatic S7-200

Х. Шиндлер (Компания Siemens)

Простая и гибкая регистрация данных о состоянии оборудования, их хранение, опрос и дальнейшее использование в стандартных программах обработки электронных таблиц, например Microsoft Excel, — функция, часто необходимая для станков и агрегатов. Рассматривается возможность протоколирования данных с помощью микро-ПЛК Simatic S7-200.

Производственные данные (сообщения, нарушения граничных значений, время работы установки и др.) должны регистрироваться, сохраняться и передаваться для дальнейшей обработки. Это требование часто означает отказ от использования систем микро-ПЛК из-за их слишком малой памяти. На помощь в этом случае приходят новые продукты семейства микро-ПЛК Simatic S7-200 фирмы Сименс.

Больше памяти благодаря плате памяти

Хотя пользовательская память и память данных в CPU микро-ПЛК все еще малы по сравнению с более крупными системами ПЛК, однако новые CPU и две новые платы памяти для микро-ПЛК Simatic S7-200 предоставляют возможность записывать необходимые объемы данных. Процесс сохранения данных предлагается выполнять в следующем порядке:

- определение данных, подлежащих регистрации;
- добавление метки даты и времени в случае необходимости;

- создание пользовательской программы для регистрации данных в модуле памяти;
- вставка дополнительной платы памяти в CPU;
- передача проекта в CPU;
- сохранение данных с помощью нового ПО, разработанного фирмой Сименс.

Определение данных, подлежащих регистрации

Новое ПО для программирования микро-ПЛК Step7 Micro/WIN V4 содержит наряду со многими другими новшествами также инструмент под названием "Мастер регистрации данных" (Data-Log-Assistant). При его запуске в первую очередь проверяется наличие сохраняемого проекта в уже занятых областях памяти, что препятствует повторной перезаписи данных в микро-ПЛК. Сразу после этого может быть начато определение структуры данных, подлежащих регистрации, с помощью "Мастера регистрации данных", включая метку времени и дату" (рис. 1).