

**ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ЛОГИЧЕСКОГО УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ ОБОРУДОВАНИЕМ****Фурман И. А.¹, Тимчук С. А.¹, Малиновский М. Л.², Аллашев А. Ю.³**¹*Харьковский национальный технический университет сельского хозяйства имени Петра Василенко,*²*Публичная компания Thales group (Франция),*³*Производственно-торговое предприятие УкрТрейд (г. Харьков)*

В статье рассматривается подход к разработке программ логического управления с помощью табличного проблемно-ориентированного технологического языка циклограмм. Предложена концепция технологического визуального программирования, позволяющая привлечь непосредственно инженеров-технологов к созданию программ логического управления

Постановка проблемы. В последние годы наблюдается повышенный интерес со стороны различных специалистов к языкам и технологиям программирования систем логического управления промышленными объектами. С ростом уровня автоматизации промышленных объектов становятся более высокими требования к качеству программного обеспечения систем логического управления. В качестве основного элемента управления в современных системах используется программируемый логический контроллер (ПЛК), который является специализированным микропроцессорным устройством, работающим в промышленных условиях в режиме реального времени. На рынке присутствует большое количество моделей ПЛК от различных мировых производителей.

Для стандартизации большого количества вариантов языков в 1993 году Международная электротехническая комиссия выпустила стандарт МЭК 61131-3, который описывает языковые средства программирования промышленных контроллеров. В стандарт вошли три графических и два текстовых языка [1]: 1) SFC (Sequential Function Chart) – графический язык, позволяющий описать алгоритм управления в виде набора пар шаг и переход; 2) LD (Ladder Diagram) – язык лестничных диаграмм, графический язык на принципах релейно-контактных схем; 3) FBD (Functional Block Diagram) – графический язык функциональных блоков; 4) ST (Structured Text) – текстовый язык высокого уровня с паскалеподобным синтаксисом; 5) IL (Instruction List) – текстовый язык низкого уровня, представляющий собой упрощенный ассемблер.

Несмотря на то, что при разработке стандарта МЭК 61131-3 одной из задач было предоставить разработчикам унифицированные языки программирования, которые были бы просты, наглядны и доступны широкому кругу инженеров, не обладающими специальными знаниями в области информатики, на данный момент разработкой управляющих программ занимаются профессиональные программисты [2].

С другой стороны, в некоторых случаях (высокие требования к быстродействию, повышенные требования к надежности системы управления и т.д.) используют специализированные контроллеры логического управления с нетрадиционной архитектурой. В настоящее время наблюдается все более широкое применение методов и средств создания устройств логического управления на основе микроэлектронных компонен-

тов параллельного действия с применением такой современной элементной базы как программируемые логические интегральные схемы (ПЛИС) [3]. Такие промышленные ПЛИС-контроллеры логического управления программируются на текстовых специализированных языках описания аппаратуры (HDL-языках). Разработка ПЛИС-контроллера для конкретной задачи логического управления является трудоемкой задачей и требует от разработчика специальных знаний микроэлектроники и языков описания аппаратуры.

Сегодня при выборе в качестве аппаратного средства управления традиционного микропроцессорного ПЛК или специализированного ПЛИС-контроллера с параллельной архитектурой созданием и модификацией программного обеспечения занимаются профессиональные программисты. Созданный программный код остается "закрытым" для понимания и верификации инженеру-технологу, который является наиболее компетентным экспертом в предметной области автоматизируемого технологического процесса, но не обладает знаниями в области языков программирования. Таким образом, для привлечения в процесс создания программ логического управления самих инженеров-технологов, как наиболее компетентных экспертов в области автоматизируемого технологического процесса, сегодня актуальной задачей является разработка и исследование новых технологических языковых средств и нетрадиционных подходов к созданию программ логического управления.

Анализ основных исследований и публикаций. При создании программ профессиональные программисты в большинстве случаев используют частные подходы к написанию кода, зависящие от субъективного опыта и предпочтений конкретного разработчика. Следует отметить, что, учитывая специфическую предметную область, разработка и программная реализация алгоритмов логического управления технологическими объектами представляет собой отдельное направление современного программирования.

Каждый из пяти языков стандарта МЭК 61131-3 ориентирован на одну конкретную область задач логического управления. При описании сложных алгоритмов управления программисты вынуждены в одном проекте использовать несколько языков стандарта [4]. В такой программе часть алгоритма описывается на одном языке программирования, а остальная

часть программы описывается на другом языке программирования. Многие специалисты указывают на явные недостатки использования языков стандарта, "закрытость" для понимания и использования этих языков для инженеров-технологов. Поэтому появляются различные работы, описывающие альтернативные стандарту языки программирования ПЛК. Описываются идеи использования универсального визуального языка моделирования UML для создания программ логического управления [5].

Другим направлением, является адаптация универсальных текстовых языков программирования к задачам логического управления [6]. Однако, использование универсальных языков программирования и моделирования доступно только профессиональным программистам. Так же предлагается способ автоматической генерации программного кода на стандартных языках программирования ПЛК по математическим моделям, например из математического пакета MathCAD [7].

Однако программная ошибка в логике управления сложным технологическим объектом может привести не только к серьезным материальным потерям, но создать аварийную ситуацию, угрожающую здоровью и жизни людей. На данный момент для привлечения специалистов из разных предметных областей к решению прикладных задач развивается подход, опирающийся на создание и использование при создании программ проблемно-ориентированных языков [8]. Сегодня уже широко используется на практике множество проблемно-ориентированных языков для создания программ в области управления базами данных, финансов, математических пакетов и других задач. Для привлечения инженеров-технологов к разработке систем управления на базе стандартных ПЛК так же предлагается использовать различные предметно-ориентированные языки описания технологических алгоритмов.

Целью данной статьи является показать возможность и эффективность использования непосредственно в качестве языка программирования современных микроэлектронных средств логического управления табличного языка технологических циклограмм (как для различных микропроцессорных ПЛК, так и для ПЛИС-контроллеров). Что позволит улучшить качество создаваемого программного обеспечения и ускорить процесс создания (модернизации) системы логического управления.

Основные материалы исследования. С момента появления первых промышленных контроллеров различные специалисты указывали на необходимость создания наглядных проблемно-ориентированных языков программирования. Эти языки должны быть простыми и должны позволять описать алгоритм логического управления только терминами технологического процесса. Такой подход позволит инженерам-технологам самим создавать и редактировать программы управления.

Так еще в 90-х годах в харьковском институте ВНИИТЭлектромаш были разработаны и внедрены в производство на многих предприятиях бывшего СССР логические управляющие автоматы параллельного действия (ЛУАПД), при использовании которых

для формализации алгоритмов управления предлагалось использовать табличные технологические циклограммы. Этот язык получил название "язык параллельных логических контроллеров" (ЯПЛК). Описание алгоритма управления в виде таблиц циклограммы было простым, наглядным и доступным для чтения инженерам-технологам. Таким образом, непосредственно инженеры-технологи на этом табличном языке составляли программу управления, проверяли созданный алгоритм управления и устраняли логические ошибки в программе. При создании табличного языка ЯПЛК также основным требованием было понимание табличного представления алгоритма управления в виде циклограммы зрительного восприятия инженером-технологом.

С развитием микропроцессорной техники и прикладных информационных технологий в качестве программатора современных ПЛК стал использоваться персональный компьютер. В результате чего авторами был предложен новый подход к разработке программ логического управления для промышленных контроллеров, получивший название технологическое визуальное программирование (ТВП).

Предложенный подход подразумевает формализацию логики управления в виде таблиц технологической циклограммы в специализированном редакторе таблиц. По созданной технологической циклограмме специализированный транслятор автоматически генерирует программный код на стандартном языке программирования для выбранной аппаратной платформы (микропроцессорный ПЛК или ПЛИС-контроллер).

Базовым понятием программирования с помощью языка циклограмм есть "состояние" управляющего автомата. Реакция управляющего автомата на входное воздействие будет зависеть от того, на какой строке циклограммы находится управление (рис. 1).

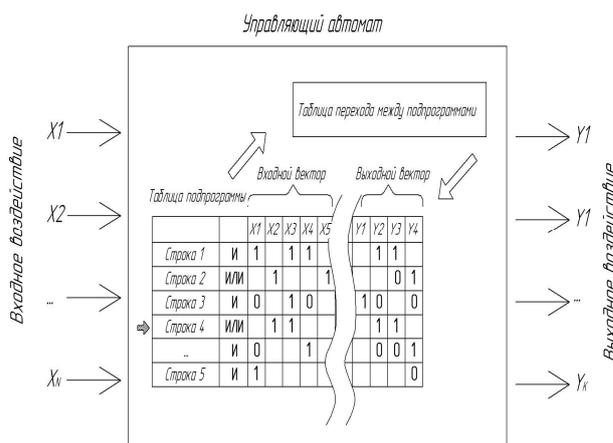


Рисунок 1 – Формализация логики управляющего автомата в виде таблиц языка циклограмм

Реализация технологического визуального программирования на практике достигается путем:

- 1) создания наглядного предметно-ориентированного языка формализации алгоритма логического управления (например, разные вариации табличного языка технологических циклограмм);

2) разработкой специализированной среды программирования для проблемно-ориентированного языка с мощными визуальными возможностями (табличный редактор циклограмм и т.д.);

3) разработкой анализатора программы на созданном проблемно-ориентированном языке;

4) разработкой правил трансляции (генерации программного кода) табличной циклограммы в программный код на стандартном языке выбранной аппаратной платформы.

Для реализации на практике предложенного подхода программирования средств логического управления была создана специализированная среда программирования, которая включила в себя следующие подсистемы:

- подсистема ввода переменных проекта (входные, выходные, внутренние переменные, таймеры, счетчики и т.д.), задания списка подпрограмм циклограммы в зависимости от реализуемого алгоритма;

- табличный редактор таблиц циклограммы, позволяющий, оперируя перечнем переменных проекта, описывать логику управления в виде строк таблиц циклограммы;

- подсистема проверки программы управления в виде таблиц циклограммы на наличие ошибок;

- "виртуальный логический контроллер";

- набор специализированных трансляторов на стандартные языки стандарта МЭК 61131-3, позволяющие логику управления в виде таблиц отлаженной циклограммы автоматически преобразовать в программный код на стандартном языке ПЛК;

- специализированный транслятор, который по таблицам циклограммы и выбранному HDL-шаблону генерирует описание управляющего цифрового устройства на языке описания аппаратуры (например, полное HDL-описание логического управляющего автомата параллельного действия).

В предложенном технологическом визуальном программировании непосредственно инженер-технолог в табличном виде описывает алгоритм логического управления в специализированной среде программирования с табличным редактором (рис 2).

Номер	Операция	Автоматика	Отладка	Пуск	Стоп	Аварийная остановка	Сигнал блокировки	Датчик 1_1	Датчик 1_2	Датчик парам	Датчик-счетчик	Механизм 1	Механизм 2	Механизм 3	Механизм 4
1	И	1					0		1			1			
2	И	1			1	0				1		0	1		
3	И		1					1				1	0	1	1
4	ИЛИ				1						1	0	0	0	
5	ИЛИ				1	1		1							0

Рисунок 2 – Табличный редактор циклограммы

При формализации алгоритма управления в виде таблиц циклограммы в информационной среде разработчик оперирует исключительно технологическими терминами предметной области.

ТВП-среда позволяет созданную программу управления загрузить в модуль "виртуальный контроллер" и проверить логику ее работы. "Виртуальный контроллер" программно моделирует работу управляющего автомата в зависимости от таблиц циклограммы, что позволяет в режиме эмуляции проверить, какие выходные воздействия будут сформированы в зависимости от тестовой последовательности входящих векторов (рис 3).

История состояний	Вход	Выход	Механизм
<input checked="" type="checkbox"/>	I1 - Автоматика	●	Q1 - Механизм 1
<input type="checkbox"/>	I2 - Отладка	○	Q2 - Механизм 2
<input checked="" type="checkbox"/>	I3 - Пуск	●	Q3 - Механизм 3
<input type="checkbox"/>	I4 - Стоп	○	Q4 - Механизм 4
<input type="checkbox"/>	I5 - Аварийная остановка	○	
<input type="checkbox"/>	I6 - Сигнал блокировки	○	
<input type="checkbox"/>	I7 - Датчик 1_1	○	
<input checked="" type="checkbox"/>	I8 - Датчик 1_2	●	

Рисунок 3 – Модуль "Виртуальный контроллер"

При эмуляции табличной циклограммы и нахождении ошибки в логике управления или возникновении коллизий в последовательности сформированных выходных сигналов, проводится корректировка табличной программы и повторная отладка на виртуальной модели.

Созданная и отлаженная программа управления в виде циклограммы автоматически транслируется в программный код на стандартном языке для выбранного аппаратного средства логического управления. Например, по циклограмме транслятор генерирует программный код на одном из языков стандарта МЭК 61131-3 для микропроцессорных ПЛК.

По этой же циклограмме специализированный транслятор генерирует текстовое описание на HDL-языке логического управляющего автомата параллельного действия. Автоматически сгенерированный программный код без изменений загружается разработчиком в стандартную среду программирования выбранного аппаратного средства, откуда непосредственно и происходит программирование подключенного ПЛК.

В рамках ТВП-технологии был разработан язык ЯПЛК-М, табличный редактор языка и правила генерации программного кода на языке описания аппаратуры AHDL. Логика управления описывается в виде таблиц на языке ЯПЛК-М, по которым с использованием HDL-шаблона управляющего автомата параллельного действия генерируется описание микроэлектронного устройства на базе ПЛИС-контроллера.

Также был создан редактор простых циклограмм, правила анализа и трансляции данных циклограмм в программный код на текстовом языке ST для ПЛК Festo. Разработан табличный язык CycloGraf, представляющий собой набор таблиц состояний автомата управления и таблиц переходов между этими состояниями. В данном языке каждое состояние это про-

граммный микроцикл, который обрабатывается автоматом управления.

Выводы. Проанализированы стандартные языки программирования ПЛК и технология их использования. Сегодня разработкой программ управления занимаются профессиональные программисты, а созданный ими программный код недоступен для чтения и понимания инженерам-технологам. Одной из распространенных визуальной форм описания алгоритма является технологическая циклограмма.

Проблемно-ориентированный табличный язык циклограмм позволяет на высоком уровне абстракции описывать технологическими терминами модель логического управления, на основе которой специализированный транслятор автоматически генерирует программный код на стандартном языке для выбранной аппаратной базы.

Язык технологических циклограмм является аппаратно-независимым. При задании логики управления в виде таблиц циклограммы пользователь оперирует исключительно терминами технологического процесса и не берет во внимание, какое микроэлектронное устройство управления будет использовано (ПЛК или ПЛИС-контроллер).

Для использования языка циклограмм должна быть создана специализированная среда программирования. В состав среды программирования должен входить редактор таблиц циклограммы, модуль проверки логической правильности составленной циклограммы и отладки логики управления, набор трансляторов в стандартные языки программирования. По одной и той же циклограмме автоматически может быть получен программный код как на языках стандарта МЭК 61131-3, так и код на нестандартных языках программирования ПЛК.

По этой же циклограмме автоматически может быть сгенерировано полное описание логического управляющего автомата параллельного действия повышенной надежности на HDL-языке описания аппаратуры.

Программа в виде технологической циклограммы может широко использоваться на этапах пусконаладочных работ, обслуживания и модернизации системы управления, в качестве технической документации. В дальнейшем необходимо произвести сравнительную оценку качества программного кода на традиционных языках ПЛК и на табличном языке циклограмм.

Методы нечеткой логики позволяют осуществить сравнительную оценку показателей качества программного кода в условиях неопределенности, когда отсутствует сколько-нибудь полноценная статистика, а также в числе исследуемых показателей присутствуют лингвистические экспертные оценки.

Список использованных источников

1. John Karl-Heinz, Tiegelkamp M. IEC 61131-3: Programming Industrial Automation Systems. Concepts and Programming Languages, Requirement for Programming Systems, Decision-Making Tools. Springer-Verlag Berlin Heidelberg, 2001.

2. Татарчевский В. А. Проблемы применения языков стандарта IEC 61131-3 и возможные пути решения. *Информационно-математические технологии в экономике, технике и образовании*. Екатеринбург : УГТУ-УПИ, 2007. С. 239-241.

3. Фурман И. О., Бовчалюк С. Я., Аллашев О. Ю. Промисловий ПЛІС-контролер та технологія його програмування. *Проблеми енергозабезпечення та енергозбереження в АПК України: Вісник ХНТУСГ імені Петра Василенка*. 2016. Вип. 176. С. 68-69.

4. Парр Э. Программируемые контроллеры: руководство для инженера / Э. Парр; пер. 3-го англ. изд. Москва : БИНОМ. Лаборатория знаний, 2007. 516 с.

5. Dubinin V, Vyatkin V, Pfeiffer T. Engineering of validatable automation systems based on an extension of UML combined with function blocks of IEC 61499. In robotics and automation (ICRA); 2005. P. 3996-4001.

6. Зюбин В. Е. "Си с процессами": язык программирования логических контроллеров. *Мехатроника*. № 12 2006. С. 31-35.

7. Hasdemir I. T., Kurtulan S. Automatic PLC code generation using MathLab. In discrete-event system design (DESDes); 2006.

8. Петров А. И. Переход от проблемно-ориентированного языка к автоматному при проектировании аналитических приборов. *Научное приборостроение*. 2016. №2(26). С. 82-92.

Анотація

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЛОГІЧНОГО КЕРУВАННЯ ТЕХНОЛОГІЧНИМ ОБЛАДНАННЯМ

Фурман І. О., Тимчук С. О., Малиновський М. Л.,
Аллашев О. Ю.

У статті розглядається підхід до розробки програм логічного керування за допомогою табличної проблемно-орієнтованої технологічної мови циклограм. Запропоновано концепцію технологічного візуального програмування, що дозволяє залучити безпосередньо інженерів-технологів до створення програм логічного управління

Abstract

INFORMATION TECHNOLOGY OF DESIGNING LOGIC CONTROL PROGRAMS OF TECHNOLOGICAL EQUIPMENT

I. Furman, S. Tymchuk, M. Malinovsky, A. Allashev

The article discusses the approach to the development of logical control programs using a tabular problem-oriented technological language of cyclograms. The concept of technological visual programming is proposed, which allows directly involving process engineers to create logical control programs.