

ИНФОРМАЦИОННЫЕ И ТЕЛЕКОММУНИКАЦИОННЫЕ СЕТИ И СИСТЕМЫ

УДК 004.057.2

О СТАНДАРТЕ ИЕС 61499

Акматабеков Рысбек Актаевич, к.т.н., профессор КГТУ, проф. каф. автоматического управления, директор Кыргызско-корейского центра информационного доступа, КГТУ им. И.Раззакова, Кыргызстан, 720044, г. Бишкек, пр. Ч. Айтматова, 66, e-mail: nark@mail.kg

Современные цифровые системы управления реализуются с применением программируемых логических контроллеров (ПЛК). ПЛК также широко используются при создании пространственно-распределенных систем управления. В последние годы для проектирования распределенных систем управления в промышленности разработан международный стандарт ИЕС 61499 (МЭК 61499), который широкого распространения среди проектировщиков и исследователей еще не получил. В связи с этим возникла необходимость в данной обзорно-аналитической статье, где представлены краткая история разработки, основное содержание, особенности и средства проектирования в стандарте ИЕС 61499. Литературный материал комментируется, даются объяснения некоторых терминов, предлагается расширение понятий и анализ материала.

Ключевые слова: ИЕС, МЭК, стандарт, функциональные блоки, Processand Instrumentaion Diagrams, распределенные системы автоматизации, программируемые логические контроллеры, история, ИЕС61499, ИЕС61131, ИЕС61158, программирование, программное обеспечение, модель, XML, язык программирования, ЕСС, IL, LD, FB, SFC, ST, система программирования.

ABOUT THE IEC 61499 STANDARD

Akmatbekov Aktaevich Rysbek, candidate of technical Sciences, professor of KSTU, Professor, department of automatic control, director of the Kyrgyz-Korean Center for Information Access, KSTU. I.Razzakova, Kyrgyzstan, 720044 Bishkek, Ch.Aitmatov 66, e-mail: nark@mail.kg

Modern digital control systems implemented using programmable logic controllers (PLC). PLC is also widely used to create a spatially distributed control systems. In recent years, for the design of distributed control systems in the industry develop an international standard IEC 61499 (IEC 61499), which is widespread among designers and researchers have not yet received. In this regard there was a need in the art surveillance and analytical article, which presents a brief history of the development, the basic content, features and design tools in the IEC 61499 standard. Commented Literary material, explanations of some of the terms proposed extension of the concepts and analysis of the material.

Keywords: IEC, the International Electrotechnical Commission (IEC) standard, P&ID (Processand Instrumentaion Diagrams), programmable logic controllers (PLC), distributed automation systems (DAS).

Введение

В данной работе выполнен обзор и анализ стандарта международной электротехнической комиссии (МЭК, ИЕС – International Electrotechnical Commission) ИЕС61499 с целью ознакомления специалистов Кыргызстана в области исследования, проектирования и программирования цифровых распределенных систем управления [2-4, 7,

8, 10, 12-14, 16]. Статья продиктована тем, что стандарт является относительно новым, широкого распространения пока не получила и отсутствует доступная литература. Вместе с тем в настоящее время распределенные системы автоматизации (РСА), для проектирования которых и разработан стандарт, находят все большее применение. По современной концепции РСА строятся с применением средств автоматизации, вычислительной техники и телекоммуникаций. В таких системах информационная сеть считается частью системы автоматизации. При таком подходе не только большие системы, но простые системы автоматизации становятся распределенными.

В работе рассмотрены краткая история, содержание, структура, особенности стандарта и средства разработки проектов в стандарте IEC 61499.

История разработки стандарта

Стандарт IEC61499 стал разрабатываться Международной электротехнической комиссией (МЭК – IEC – International Electrotechnical Commission) в 90-х годах двадцатого века в рабочей группе WG63 технического комитета TC65 [2, 3, 7, 8, 10, 12, 13]. В рабочую группу вошли производители промышленных систем управления, поставщики оборудования из США, Японии, Англии, Европы. Обсуждение стандарта IEC61499 началось в октябре 1990 года, а активная работа над ним - в марте 1992 года. Апробация подготовленного проекта стандарта состоялась в марте 2001 года, завершение разработки - в 2005 г. Стандарт состоит из четырех условных частей и посвящен использованию программных модулей, называемых функциональными блоками, при создании распределенной системе автоматизации (РСА) техническими объектами.

Стандарт IEC61499 задает распределенную, управляемую событиями архитектуру и требования к программному инструментарию для инкапсуляции, встраивания, развертывания и интеграции программного обеспечения в интеллектуальных устройствах, машинах и системах [12]. В основу проекта были положены стандарты IEC61131-3 и IEC61158 (Fieldbus).

В объектно-ориентированном программировании инкапсуляция представляет собой сокрытие внутренней структуры данных и реализации методов объекта от остальной программы, т. е. включение в объект всей необходимой ему информации, таким образом, чтобы другим объектам не требовалось знаний об его внутренней структуре. Доступен только интерфейс объекта, через который осуществляется все взаимодействие с ним

Стандарт IEC61499 определяет распределенную модель как разбиение различных частей промышленного процесса автоматизации и сложной системы управления на функциональные модули, называемые функциональными блоками. Эти функциональные блоки могут распределяться и взаимодействовать по коммуникационной сети через множество контроллеров.

Приложение становится распределенным путем размещения экземпляров функциональных блоков на различных ресурсах в одном или более устройствах. Функциональные блоки являются простейшими элементами распределения. Приложение со многими функциональными блоками отображается как один элемент, хотя экземпляры функциональных блоков распределяются по ресурсам и устройствам.

Работа над стандартом велась достаточно долго. Это объясняется масштабностью проекта, т.к. стандарт рассчитан не только на производителей контроллеров и разработчиков программного обеспечения, но для производителей промышленного оборудования. Поэтому преимущества стандарта в полной мере могут проявиться только в том случае, если также будут выпускаться стандартные промышленные устройства, свойств которых обеспечивают совместимость с требованиями стандарта IEC61499. Стандарт состоит из следующих компонент:

Часть 1: Архитектура функциональные блоки.

Часть 2: Требования к программным средствам.

Часть 3: Учебное пособие.

Часть 4: Правила для профилей соответствия.

Функциональные блоки

Первая часть стандарта «Архитектура и функциональные блоки» дает описание архитектуры функциональных блоков (ФБ), подходы к их созданию и применению на практике и содержит описание следующих ключевых позиций:

- правила описания ФБ;
- правила поведения событий в ФБ;
- правила использования ФБ в конфигурации РСА;
- правила взаимодействия ФБ по различным каналам связи;
- правила использования ФБ при управлении приложениями, ресурсами и устройствами в распределенных управляющих системах;
- требования соответствия стандартам.

В первой части стандарта дано лишь описание ФБ без деталей о его реализации.

Вторая часть стандарта «Требования к программным средствам» содержит реализацию ФБ, которая требует указания:

- внешних интерфейсов, которые обеспечивает ФБ;
- генерируемые ФБ события;
- ожидаемые ФБ события и реакции на них;
- входные и выходные сигналы ФБ;
- графическое отображение ФБ.

Для описания указанных выше атрибутов ФБ используется XML (Extensible Markup Language – расширяемый язык разметки [гипертекста]) – средство, позволяющее структурно описать любые типы данных в текстовом виде для дальнейшего хранения и транспортировки [9]. Применением XML обеспечивает свойство кроссплатформенности создаваемых программ. Данные, закодированные в XML, будут одинаково интерпретированы везде. ФБ содержит набор алгоритмов или функций, которые придают объекту необходимые свойства. Для программирования ФБ можно использовать языки стандарта IEC61131 (SFC, FB, LD, ST, IL), которые будут расшифрованы далее.

Язык XML, предложенный W3C метаязык форматирования документов World-Wide Web, является подмножеством языка SGML. W3C (World Wide Web Consortium) – международный исследовательский консорциум во главе с Тимоти Джон Бернес-Ли, разрабатывающий и внедряющий стандарты Всемирной сети. Разработка XML велась под руководством Джона Босака. Первая версия спецификации опубликована в феврале 1998 г. XML позволяет создавать языки разметки, учитывающие специфику документов заданной предметной области, например, MathML, XLink, SMIL, XSL и другие. XML документ состоит из двух частей: из определения используемого языка и собственно документа. Определение языка задаётся с помощью шаблонов DTD и может храниться как в файле с документом, так и отдельно.

Проектирование в стандарте IEC61499

Объектное отображение процессов успешно используется при разработке больших программных комплексов. Объекты можно использовать для описания любых физических явлений и процессов, поэтому они представляют собой гибкий описательный механизм. В стандарте IEC61499 и реализуется объектно-ориентированный подход с ФБ. ФБ призваны инкапсулировать в себе реализацию тех или иных функций и предоставлять интерфейсы для взаимодействия с другими блоками. Так происходит переход на более высокий уровень абстракции, где оперируют понятиями объектов производства.

Стандарт IEC61499 определяет общую модель и методологию проектирования, применимую для описания ФБ в независимом от реализации формате. Система, описанная в рамках логически соединенных блоков, может быть выполнена на различных вычислительных ресурсах. Условно можно выделить три стадии создания проекта в терминах функциональных блоков: 1) первичный анализ на физическом уровне; 2) выделение областей функциональной зависимости и их взаимодействия; 3) отображение функциональности на физические устройства.

Первая стадия предполагает применение P&ID (Process and Instrumentation Diagrams) структурных схем, отражающих суперпозицию физического процесса или объекта и применяемых в нем технических средств. На создаваемой схеме наносятся различные технологические установки, указывается расположение датчиков и других технических средств. Этот документ является началом проекта и отражает уровень профессионализма как разработчика системы, так и заказчика проекта. От успешности этой стадии зависит конечный успех проекта. Для создания P&ID схем можно назвать программы AutoCAD, CADWorx, SmartPlant и им подобные, которых достаточно много.

После разработки P&ID схемы, можно приступать к построению зависимостей между различными элементами. Этот шаг подразумевает связывание потоков данных, проходящих между всеми логическими единицами, определенными на предыдущем этапе.

Этап отображения представляет собой процесс перевода созданной архитектуры в термины ФБ, а затем перенос на аппаратные средства, подстройку средств ввода/вывода, согласование коммуникационных каналов и некоторые другие действия.

Так как ФБ могут создаваться (программироваться) с помощью языков программирования контроллеров, определенных стандартом IEC61131, то тем самым обеспечивается преимущество решений, когда разработчики могут легко перейти к созданию программ по новой, объектно-ориентированной методологии. Кроме стандартных языков IEC 61499 (IEC61131), новые ФБ можно также создавать и на стандартных языках программирования, таких как C++ или Delphi.

Предложенная давно идея объектного программирования оказалась жизнеспособной, потому что человеку легче оперировать понятными ему объектами реального мира, обладающими определенными свойствами, чем с набором разрозненных абстрактных переменных. Объекты отражая элементы реального мира обладают свойствами стабильности и пластичности. Эти свойства проявляются в том, что объекты без оказания внешнего воздействия не изменяются. Эти свойства присущи и для ФБ, которые изменяют свое состояние лишь в результате определенного внешнего воздействия.

Для того чтобы определить ФБ, нужно указать его следующие свойства: внешний интерфейс (входные и выходные переменные), схему управления и вызываемые ею алгоритмы.

Схема управления представляет собой карту реакций на внешние воздействия, в которой расписывается, что именно нужно сделать, если произошло конкретное событие. Кроме внешних воздействий при необходимости в карте указывается, какие события будут генерироваться при окончании выполнения функций обработчиков. Наиболее удобным языком для создания схем управления является SFC, в котором можно достаточно легко сделать привязку вызова определенных функций в ответ на изменения некоторых переменных.

Алгоритмы обработки событий можно создавать на любом из языков IEC 61131 или на любом другом, способном воспринимать и обрабатывать входные, выходные и внутренние переменные. Можно предполагать, что со временем появятся библиотеки, позволяющих задавать алгоритмы ФБ на стандартных языках программирования типа C или Delphi.

ФБ также поддерживают интерфейсы обслуживания. Этот механизм позволяет указать, какие именно последовательности действий должны быть проведены, чтобы получить адекватный результат. С его помощью можно легко описать как операции ввода/вывода для различных технических средств, так и взаимодействия со все возможными службами запуска по времени или таймерами.

Согласно второй части стандарта IEC61499 все описания функциональных блоков сохраняется в формате XML. Чтобы обеспечить стандартизацию библиотек, были разработаны соответствующие DTD (Document Type Definitions – определители типа документа). Механизмы DTD являются единственным стандартным способом описания грамматики документа в формате XML, в нем фиксируются допустимые для применения типы данных, физические устройства, дополнительные ресурсы и файлы конфигурации.

В задачах построения сложных информационно-управляющих систем, в том числе РСА, одной из сложных проблем является обмен данными между различными подсистемами [5]. Часто задачи импорта/экспорта данных из одной системы в другую приводит к необходимости разработки дополнительных программных модулей для согласования подсистем. Задача существенно облегчается, если данные определенного класса будут перемещаться между подсистемами, при условии, что в этих подсистемах будет заложена технологически реализованная возможность воспринимать извне и отдавать наружу данные в стандартном формате импорта/экспорта. Данный подход является основой для разработки метаданных и интерфейсов для обмена регулярными данными для различных унаследованных разноформатных систем. Для создания спецификаций протокола взаимодействия разноформатных систем используются технологии XML.

При создании систем автоматизации и управления на основе web технологий совокупность ФБ и XML также обеспечивает выгодную позицию, т. к. уже сейчас имеются браузеры, способные корректно отображать поступающую из сети информацию в этом формате.

Диаграмма управления выполнением

Для определения ФБ необходимо задать (разработать) диаграмму управления выполнением – ЕСС (execution control chart). Стандарт IEC 61499 определяет ЕСС как «графическое или текстовое представление причинных связей среди событий в событийных входах и событийных выходах функционального блока и выполнения алгоритмов функционального блока, с использованием состояний управления выполнением (ЕС состояний), переходов управления выполнением (ЕС переходов) и действий управления выполнением (ЕС действий)». Таким образом, ЕСС задает действия ФБ на внешние возможные воздействия (события). Входные события инициируют и управляют работой ФБ. Считается, наиболее удобный язык для создания ЕСС – язык SFC. Для описания алгоритмов преобразования входных данных ФБ в выходные используются языки и типы данных стандарта IEC 61131-3. Алгоритмы обрабатывают входные, внутренние и выходные переменные.

Поведение в ЕСС описывается следующим образом:

- инициализация и вызов экземпляра функционального блока для проверки входного события;
- проверка на появление события;
- выполнение алгоритма функционального блока;
- генерация выходного события.

Пример диаграммы управления выполнением (ЕСС) представлен на рис. 1. Индивидуальным элементам диаграммы на рис. 1 соответствуют элементы SFC:

- окно с двойным контуром указывает начальный шаг;
- стрелки указывают переходы;
- блоки с одиночным контуром указывают шаги;
- двойные блоки указывают генерирующие (создающие) событие выходы. Место слева указывает имя алгоритма (блока действий) при его определении.

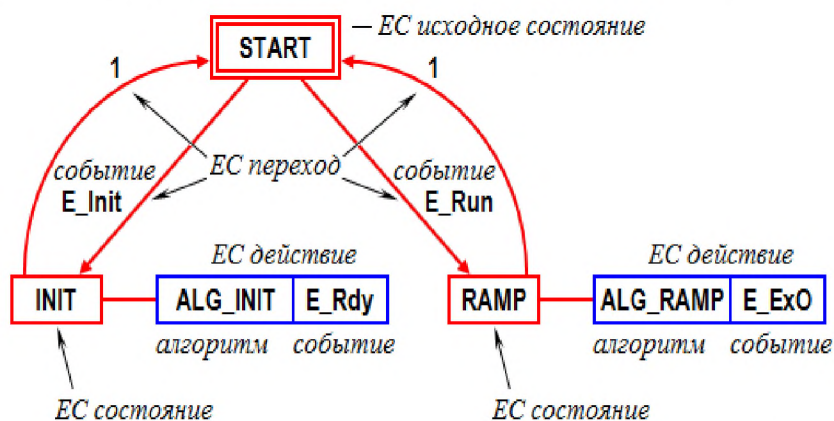


Рисунок 1 Диаграмма управления выполнением (ЕСС).

Функциональные блоки ИЕС 61499

В программируемых системах управления прием, хранение, преобразование, обработка информации и принятие решения возлагается на программируемый логический контроллер (ПЛК) или промышленный компьютер. Эти действия, согласно стандарту ИЕС 61499 выполняют функциональные блоки.

ФБ – это самостоятельный компонент вычислительного процесса, который преобразует входные данные и вырабатывает выходное воздействие (событие) за конечное время. ФБ имеет информационные (входы данных) и управляющие (событийные входы). Событие на управляющем входе активизирует ФБ, который читает данные с информационных входов. Считанная информация обрабатывается заложенным в ФБ алгоритмом, в результате чего создается выходное событие, которое запускает процедуру съема результата на информационных выходах, т.е. вывод информации из ФБ. ФБ может не иметь информационных входов и/или выходов, но должен иметь хотя бы один управляющий вход.

Стандарт ИЕС 61499 различает базисные и композиционные ФБ. Композиционный ФБ – это набор из нескольких базисных ФБ. Базисный ФБ определяется путем задания входных и выходные переменных, входных и выходных события, ЕСС и алгоритма работы ФБ по обработке данных (рис. 2).

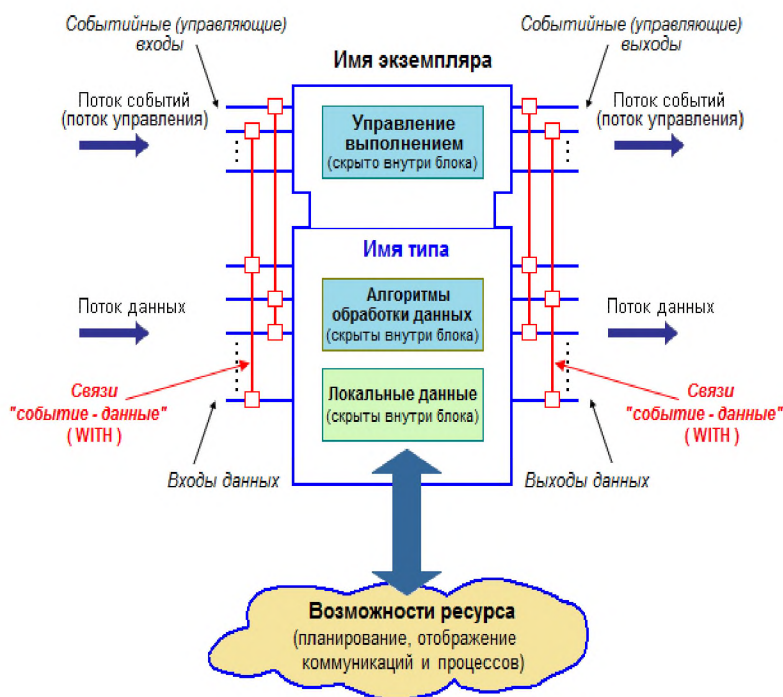


Рисунок 2. Базисный функциональный блок.

Базисный ФБ состоит из двух частей.

1. Верхняя часть содержит ЕСС. Эта часть должна быть запрограммирована как конечный автомат, который рекомендуется программировать с помощью языка программирования ПЛК SFC.

2. Нижняя часть определяет алгоритмы обработки данных и может программироваться с помощью любого языка стандарта ИЕС 61131-3.

В стандарте ИЕС 61499 определено 18 типовых ФБ. Любые физические элементы системы автоматизации должны быть представлены в виде базисного или композиционного ФБ. ФБ являются не просто частью языка FBD, а элементом систем автоматизации при моделировании и проектировании. ФБ могут быть использованы также для поддержания всего жизненного цикла системы автоматизации.

Описанию и применению функциональных блоков посвящены стандарты МЭК 61499, МЭК 61131-3 и МЭК 61804.

Средства разработки проектов в IEC 61499

Для создания (программирования) функционального блока используются языки программирования ПЛК, а в некоторых случаях – алгоритмические языки высокого уровня (АЯП) C⁺⁺ и Delphi. Для программирования ПЛК используются пять языков программирования (ЯП): IL, LD, FB, SFC, ST. Стандарт IEC 61499 предусматривает разработку ЕСС в графической, текстовой, табличной форме, а должна быть она запрограммирована как конечный автомат на ЯП SFC. В то же время нижняя часть ФБ определяет алгоритмы его работы и программируется с помощью любого ЯП ПЛК.

Для унификации инженерных технологических и программных решений, стандартизации и автоматизации программирования ПЛК был разработан стандарт ИЭК 61131-3, описывающий языки программирования для ПЛК [8,10,12], который включает пять ЯП (табл.1).

Таблица 1

Обозначение	Краткая характеристика
IL Instruction List	Текстовый язык. Аппаратно-независимый низкоуровневый ассемблероподобный язык.
LD Ladder Diagram	Графический язык. Представляет собой программную реализацию электрических схем на базе электромагнитных реле.
FBD Function Block Diagram	Графический язык. Функциональный блок (ФБ) выражает некую подпрограмму. Каждый ФБ имеет входы (слева) и выходы (справа). Программа создается путем соединения множества ФБ.
SFC Sequential Function Chart	Графический высокоуровневый язык. Создан на базе математического аппарата сетей Петри. Описывает последовательность состояний и условий переходов.
ST Structured Text	Текстовый Паскалеподобный язык программирования

IL (Instruction List – список инструкций). По синтаксису напоминает ассемблер. Ориентирован на профессиональных программистов и разработчиков контроллеров и программного обеспечения (ПО) для них. Является вместе с языком LD одним из самых распространенных при программировании ПЛК.

LD (Ladder Diagram – релейно-контактные схемы). Язык релейной (лестничной) логики. Применяются также названия:

- язык релейно-контактной логики (РКС);
- релейные диаграммы;
- релейно-контактные схемы;
- язык программирования релейно-лестничной логики.

Предназначен для программирования (ПЛК). Синтаксис языка удобен для замены логических схем, выполненных на релейной технологии. Ориентирован на инженеров по автоматизации, работающих на промышленных предприятиях. Обеспечивает наглядный интерфейс логики работы контроллера, облегчающий не только задачи собственно программирования и ввода в эксплуатацию, но и быстрый поиск неполадок в подключаемом к контроллеру оборудовании.

FBD (Function Block Diagram – диаграмма функциональных блоков).

Предназначен для программирования (ПЛК). Программа образуется из списка цепей, выполняемых последовательно сверху вниз. Цепи могут иметь метки. Инструкция перехода на метку позволяет изменять последовательность выполнения цепей для программирования условий и циклов.

При программировании используются наборы библиотечных функциональных блоков и собственные блоки программиста, также написанные на FBD или других языках МЭК 61131-3. Функциональный блок (ФБ) - это подпрограмма, функция, которые представляют собой логические функции И, ИЛИ, НЕ, триггеры, таймеры, счётчики, блоки обработки сигналов, математические операции и т.д.

Каждая отдельная цепь представляет собой выражение, составленное графически из отдельных элементов. К выходу ФБ подключается следующий блок, образуя цепь. Внутри цепи блоки выполняются строго в порядке их соединения. Результат вычисления цепи записывается во внутреннюю переменную либо подается на выход ПЛК.

SFC (Sequential Function Chart – последовательные функциональных схем).

Предназначен для программирования промышленных контроллеров. Широко используется в SCADA системах. SFC - графический язык, предназначенный для написания программ последовательного управления технологическим процессом, описывающий его в форме близкой к диаграмме состояний. Аналогом может служить сеть Петри с разноцветными фишками. В каждом состоянии система выполняет действия (подпрограммы) с определенными модификаторами. Например, модификатор N - исполнять, пока состояние активно.

ST (Structured Text - структурированный текст). Этот язык является одним из вариантов языка программирования ЭВМ Паскаль. Предназначен для программирования ПЛК и операторских станций. Удобен для написания больших программ и работы с аналоговыми сигналами и числами с плавающей точкой. Широко используется в SCADA системах.

Основой ST-программы служат выражения. Выражения состоят из операторов и операндов (констант и переменных). Операторы должны заканчиваться точкой с запятой. Одна строка может содержать несколько операторов.

Результат вычисления присваивается переменной при помощи оператора присваивания (:=).

ЯП ПЛК, как правило, реализуются в составе системы программирования ПЛК, которая представляет собой программный комплекс, чаще всего, инструментального типа. Программное обеспечение этого класса называются SoftPLC или SoftLogic. Среди них наиболее соответствующими стандарту IEC 61499 являются ISaGRAF и CoDeSys.

ISaGRAF - мощная программная технология для разработки встраиваемых приложений, разработчиком и производителем которой является ICS Triplex. Основано на концепции открытой автоматизации и обеспечивает существенное сокращение времени создания изделий и ускоряет их выход на рынок. Это технология программирования ПЛК, которая позволяет создавать как локальные, так и распределенные системы управления. Основа технологии - среда разработки приложений ISaGRAF Workbench и адаптируемая под различные аппаратно-программные платформы исполнительная система ISaGRAF Runtime.

В многозадачной системе одновременно могут выполняться несколько исполнительных систем ISaGRAF PRO с различными циклами опроса. Механизм связывания переменных обеспечивает передачу данных между исполнительными системами, как в локальном, так и в распределенном проекте, причем аппаратные платформы могут работать под управлением различных операционных систем.

В декабре 2005 года появился первый в мире программный продукт, позволяющий создавать проекты, отвечающие требованиям стандарта IEC61499 – ICS Triplex ISaGRAF 5.0, который объединяет в себе уже известные возможности IEC 61131 и особенности реализации ФБ стандарта IEC61499. Эти возможности позволят создавать приложения, объединяющие многие ресурсы, распределенные на различных ПЛК, создавать сложную вычислительную

систему из готовых ФБ или, создавать собственные ФБ. Необходимо отметить, что сохранилась совместимость с исполняемыми модулями 4-й версии, что позволит с ее помощью создавать проекты и для более ранних версий ISaGRAF 4.xx.

CoDeSys - инструментальный программный комплекс промышленной автоматизации. Производится и распространяется компанией 3S-Smart Software Solutions GmbH (Кемптен, Германия). Название CoDeSys является акронимом от Controller Development System. Версия 1.0 была выпущена в 1994 году. В CoDeSys для программирования доступны все пять ЯП стандарта IEC 61131-3.

Кроме этих систем известностью в мире пользуются такие системы проектирования ПСА и программирования ПЛК как MULTIPROG, SIMATIC STEP 7, GXDeveloper, OpenPCS, MasterPLC Designer, Concept, Unity Pro, Visilogic (Unitronics PLC), TwinCAT, B&R Automation Studio, ITool (Carel).

Simatic Step 7 - программное обеспечение фирмы Siemens AG для разработки систем автоматизации на основе программируемых логических контроллеров Simatic S7-300/S7-400/M7/C7 и WinAC. Программное обеспечение выпускается с интерфейсом на английском, немецком, французском, итальянском и испанском языках. Специальные версии обеспечивают работу на японском и китайском языках. Предшественником данного ПО является более ранняя версия Simatic Step 5, работающая в ОС DOS или в DOS окне ОС Windows.

OpenPCS компании Infoteam Software GmbH имеет ту особенность, которая заключается в использовании языка IL в качестве промежуточного кода. Элементы программы, выполненные на любом ЯП ПЛК, можно копировать в буфер обмена Windows и вставлять в программу на другом языке с автоматическим перекодированием. Для достижения высокого быстродействия в составе комплекса присутствуют компиляторы машинного кода для ряда распространенных процессоров. Симулятор ПЛК SmartSIM позволяет проводить обучение и отладку без внешней аппаратуры.

Заключение

Стандарт МЭК 61499, состоящий из четырех частей, был опубликован в 2005 г. Он устанавливает обобщенную архитектуру функциональных блоков и предоставляет руководство для их применения в распределенных системах промышленной автоматизации. В таких системах программное обеспечение распределено между несколькими физическими устройствами (ПЛК) и несколькими функциональными блоками (ФБ), а промышленная сеть рассматривается как составная часть системы.

Стандарт IEC 61499 определяет распределенную, управляемую событиями архитектуру и требования к программному инструментарию для инкапсуляции, встраивания, развертывания и интеграции программного обеспечения в интеллектуальных устройствах, машинах и системах.

Особенностью ФБ в МЭК 61499 является возможность управления событиями и большая степень обобщения функциональных блоков. Стандарт МЭК 61499 используется совместно с МЭК 61131-3 как средство описания базовых типов и внутреннего описания ФБ для программирования ПЛК.

Технология IEC61499 может быть успешно применена в системах программирования, которые не поддерживают IEC61499

Список литературы

1. Андреев Н.А., Кунцевич О.В., Синенко Е.Б. SCADA-системы: взгляд изнутри. – М.: Издательство «РСТ Софт», 2004. – 176 с.
2. Гулько С. В., Николас Джоврей. Обзор стандарта IEC 61499 // Промышленные измерения, контроль, автоматизация, диагностика. – 2005, №4. – с. 8-12.
3. Денисенко В. В. Компьютерное управление технологическим процессом, экспериментом, оборудованием. – М.: Горячая линия – Телеком, 2009. – 608 с.

4. Дубинин, В. Н., Вяткин В. В. Формализованное описание и моделирование систем функциональных блоков IEC 61499 // Известия высших учебных заведений. Поволжский регион. – 2005. – № 5 (20). – с. 76–89. – (Технические науки).

5. Дубинин, В. Н., Вяткин В. В. Семантический анализ описаний систем управления промышленными процессами на основе стандарта IEC 61499 с использованием онтологий. // Известия высших учебных заведений. Поволжский регион. – 2010. – № 3 (15). – С. 3–15 – (Технические науки).

6. Руководство пользователя по программированию ПЛК в CoDeSys 2.3. – Kempten: 3S - Smart Software Solutions GmbH, 2008. – 454 S. [www.3s-software.com].

7. Сальников С. Н. Моделирование базисных функциональных блоков IEC61499 в интегрированной инструментальной среде CODESYS (3S) 2.3 // Доклады X научно-практическая конференция ППС ВПИ. – Волгоград: ВолГТУ, 2011.

8. Создание распределенных приложений на основе стандарта IEC61499 и среды программирования контроллеров ISaGRAF [Электронный ресурс]. – URL: <http://isagraf.ru/tehnologiya/isagraf-i-iec61499>.

9. Старых В. А., Дунаев С. Б., Коровкин С. Д. Спецификация и форматы обмена данными в разнородных информационных системах на базе XML-технологий [Электронный ресурс]. – URL: <http://citforum.ru/internet/xml/xmltech/>.

10. Энциклопедия АСУ ТП [Электронный ресурс]. – URL: <http://www.bookasutp.ru/>.

11. IEC International Standard 61131-3: Programmable Controllers -Part 3. Programming Languages/ Ed.2. – Geneva, Switzerland: The International Electrotechnical Commission, 2003.

12. International Standard IEC 61499. Function blocks for industrial-process measurement and control systems. Part 1: Architecture / International Electrotechnical Commission. – Geneva, 2005. – 111 p.

13. International Standard IEC 61499. Function blocks for industrial-process measurement and control systems. Part 2: Software tool requirements / International Electrotechnical Commission. – Geneva, 2005. – 41 p.

14. IEC 61499 [Электронный ресурс]. – URL : https://en.wikipedia.org/wiki/IEC_61499.

15. Lewis R.W. Programming industrial control systems using IEC 1131-3/ Revised ed. - London, United Kingdom: The Institution of Electrical Engineers, 1998. –329 p.

16. Lewis R.W. Modelling control systems using IEC 61499. Applying function blocks to distributed systems/ Reprinted. -London, United Kingdom: The Institution of Electrical Engineers, 2008. –192 p

17. OpenPCS Infoteam Software GmbH [Электронный ресурс]. – URL: <http://vywww.infoteam.de/>.

18. Petry, Jochen. Modicon Micro Programmierung mit ConCept. – Seligenstadt: AEG Schneider Automation, 1996. – 420 S.

УДК 303.447.3:528.8.04-022.322:528.931.3

МЕТОД ПОСТКЛАССИФИКАЦИОННОГО АНАЛИЗА ДАННЫХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ДЛЯ ВЫЯВЛЕНИЯ ИЗМЕНЕНИЙ ПОЧВЕННО-РАСТИТЕЛЬНОГО ПОКРОВА

Каримова Гульмира Токтомуратовна, ст.преподаватель кафедры «ИСТТ», Институт Электроники и Телекоммуникаций КГТУ им. И.Раззакова, Кыргызстан, 720044, г.Бишкек, пр. Мира 66, e-mail: k.gulpeace@gmail.com

Аннотация. В данной статье обсуждается пиксел-ориентированный метод выявления изменений почвенно-растительного покрова. Представлены и проанализированы результаты, полученные на основе пост-классификационного метода