

УДК 004.4'2

**О ПРЕИМУЩЕСТВАХ ПРИМЕНЕНИЯ
СТАНДАРТА МЭК 61499 ПРИ
ПРОГРАММИРОВАНИИ АСУ
СОРТИРОВКИ БАГАЖА**

Ю. Н. Поздеев, В. Х. Аль-Тибби

Донской государственный технический
университет, Ростов-на-Дону, Российская
Федерация

pozdeevjura@rambler.ru
nb1979@mail.ru

Рассмотренная в исследовании проблема применения стандарта МЭК 61499 при программировании АСУ сортировки багажа является актуальной, поскольку в настоящее время автоматизированные системы сортировки багажа в аэропорту зависят от централизованных систем управления и от стандартных методов автоматизации. Практическая значимость работы заключается в том, что методы автоматизации обработки материалов или производства устаревают, ограничивают растущий спрос на системы, которые более устойчивы к ошибкам, просты в обслуживании и могут реконфигурироваться. Научная новизна работы состоит в создании полностью автоматизированной системы сортировки багажа в аэропорту на основе функциональных блоков стандарта МЭК 61499.

Ключевые слова: МЭК 61499, распределенные системы, АСУ сортировки багажа, *ISaGRAF*, *BHS*.

Введение. В настоящее время автоматизированные системы сортировки багажа в аэропорту зависят от централизованных систем управления и от стандартных методов автоматизации, что делает рассматриваемую в работе задачу особенно актуальной. Стандартные методы автоматизации устаревают, стимулируя растущий спрос на системы, которые более устойчивы к ошибкам, просты в обслуживании, а также могут реконфигурироваться. Научная новизна предложенного авторами подхода состоит в использовании функциональных блоков стандарта МЭК 61499 — новой архитектурной основы для проектирования распределённых систем промышленной автоматизации и их повторно используемых элементов.

UDC 004.4'2

**ON THE ADVANTAGES OF APPLICATION
OF IEC 61499 STANDARD WHEN
PROGRAMMING THE BAGGAGE
SORTING SYSTEM**

Y. N. Pozdeev, W. H. Al-Tibbi

Don State Technological university, Rostov on Don,
Russian Federation

pozdeevjura@rambler.ru
nb1979@mail.ru

The investigated problem of IEC 61499 standard application at airport baggage handling programming is highly relevant as it is currently dependent on centralized control systems and conventional automation programming techniques. The practical significance of the work is in the fact that in this and other areas of manufacturing and materials handling, automation technologies are increasingly limiting for the growing demand for systems that are reconfigurable, fault tolerant and easy to maintain. The scientific novelty of the work lies in the described possibility of creating a fully automated baggage handling system at the airport, based on the functional blocks of IEC 61499 standard.

Keywords: IEC 61499, distributed system, IEC 61499, DCS Baggage Handling, *ISaGRAF*, *BHS*.

Данная проблема рассматривалась многими исследователями [1]. Существует несколько архитектур для многоагентных систем [2], которые включают в себя функциональные блоки (ФБ). Целью исследования является создание полностью автоматизированной системы сортировки багажа в аэропорту на основе функциональных блоков стандарта МЭК 61499. Предметом исследования выступает объектно-ориентированный подход для создания ФБ, представляющих собой одну секцию конвейера. Такой компонент автономен и совместим, таким образом структура и поведение системы может быть определено объединением вышеупомянутых компонентов в среде разработки ФБ [3]. Для визуализации использовалось графическое представление структуры системы. Решение было разработано в среде *ISaGRAF* — первой реализации МЭК 61499. Такой подход к проектированию объединяет имитационные модели в жизненный цикл проекта, обеспечивая готовность к использованию моделирования в рамках проектирования.

Стандарт IEC 61499 (МЭК 61499). Стандарт МЭК 61499 [4] определяет распределенную, управляемую событиями архитектуру и требования к программному инструментарию для инкапсуляции, встраивания, развертывания и интеграции программного обеспечения в интеллектуальные устройства, машины и системы. В его основу были положены стандарты *IEC61131-3* [5] и *IEC61158 (Fieldbus)* [6]. Для программируемых логических контроллеров (ПЛК) используются первая и вторая части стандарта. Первая часть «Архитектура и функциональные блоки» включает:

- основные требования;
- функциональные блоки и классификация подтипов приложений;
- функциональные блоки интерфейсов служб;
- конфигурацию функциональных единиц и систем;
- совместимость.

Вторая часть определяет «Требования к программным средствам». В соответствии со стандартом *IEC61499* распределенная модель — это функциональные блоки (ФБ) различных частей промышленного процесса автоматизации, которые могут распределяться и взаимодействовать через множество контроллеров (рис. 1, 2). В связи с этим в стандарте *IEC61499* представлены следующие понятия:

- *System* (Система): набор устройств, связанных и взаимодействующих друг с другом посредством коммуникационной сети;
- *Device* (Устройство): независимая физическая единица, способная выполнять определенные функции в конкретном контексте и ограниченная интерфейсами устройства;
- *Resource* (Ресурс): функциональная единица, имеющая независимое управление работой устройства и обеспечивающая различные сервисы для приложений, включая планирование и выполнение алгоритмов;
- *Application* (Приложение): программная функциональная единица, направленная на решение проблемы в системе управления и измерения. Приложения могут быть распределены между устройствами и могут взаимодействовать с другими приложениями;
- *Function block* (Функциональный блок, ФБ): программная функциональная единица, которая является наименьшим элементом в распределенной системе управления. ФБ использует машину состояний (*state machine*) с диаграммой алгоритма работы ФБ.

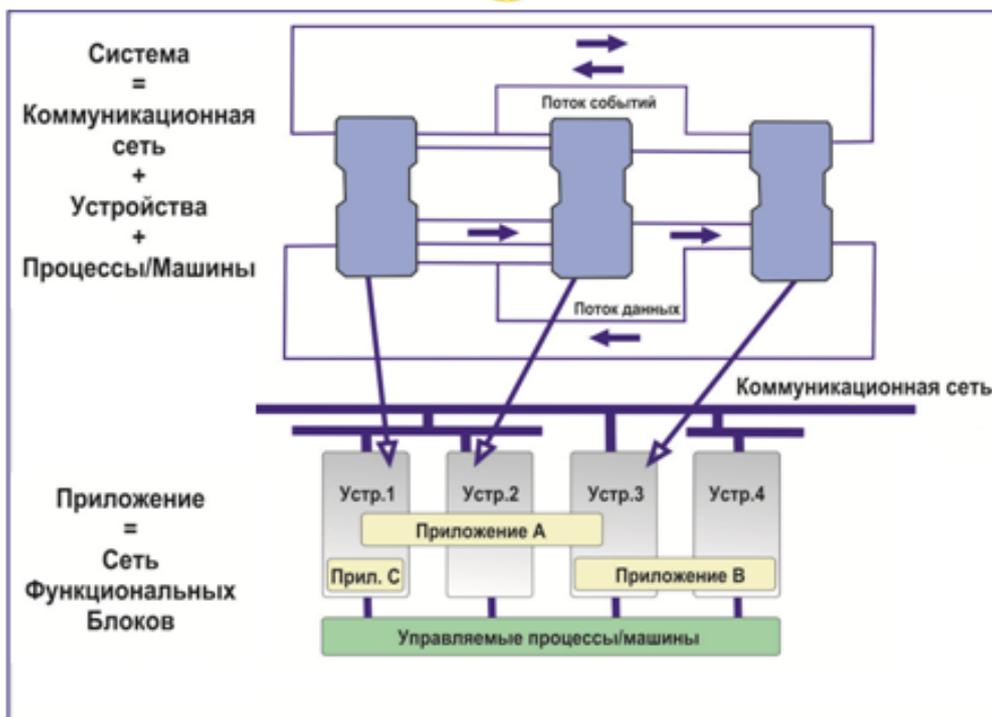


Рис. 1. Архитектура и функциональные блоки



Рис. 2. Редактор функционального блока

Для определения базового ФБ необходимо задать следующие элементы:

- входные и выходные переменные;
- входные и выходные события;
- диаграмму управления (*ECC — Execution Control Chart*);
- алгоритм блока.

Процесс разработки функционального блока IEC61499 описывается на основе анализа диаграммы управления и включает следующие этапы:

- инициализация и вызов экземпляра для проверки входного события;
- проверка на появление события;

- выполнение алгоритма функционального блока;
- генерация выходного события.

Процесс разбивается на стадии (этапы) в соответствии с технологией. Каждый этап представляет собой укрупненный функциональный блок и имеет свои ресурсы (механизмы, датчики). Управление ресурсами осуществляется при помощи программируемых логических контроллеров (ПЛК). Управляющие программы (приложения) могут работать с несколькими устройствами и использовать ресурсы друг друга при помощи коммуникационной сети.

Выделим преимущества стандарта *IEC61499*:

- регулирование потока управляющих сигналов для связанной распределенной системы управления;
- гарантия целостности и непротиворечивости данных в системе;
- обеспечение надежного синхронного взаимодействия между устройствами;
- устранение потребности в отдельных схемах синхронизации алгоритмов;
- упрощение разработки трудоемкой системы управления и сопровождения распределенных систем управления.

Постановка задачи. Разработка комплексных систем автоматизации, например, систем обработки багажа в аэропорту или *Baggage Handling Systems (BHS)*, требует решения основных задач проектирования [7]. Ввиду того, что традиционные системы автоматизации инженерных технологий, ориентированные на программируемые логические контроллеры (ПЛК), имеют низкий уровень абстракции, они не обеспечивают должного уровня контроля, а также сильно зависят от платформы.

В настоящее время основной принцип создания автоматизированных систем раздельный, то есть физические системы проектируются отдельно от программного обеспечения. Таким образом, в *BHS* механические и электрические части обычно проектируются первыми, а затем осуществляется проектирование программного обеспечения. Однако сложные инженерные системы, включая *BHS*, теперь объединяют мехатронные компоненты, компьютерное оборудование и программное обеспечение таким образом, что реинтеграция процессов их проектирования является очень полезной. Широкое признание нового стандарта МЭК 61499 пока не достигнуто из-за первоначальных затрат на переоснащение и обучение сотрудников, привыкших к старым стандартам. Инкапсулировать логику ПЛК в ФБ возможно, существует способ перехода от архитектуры ПЛК МЭК 61131-3 к МЭК 61499 [3–8], который может обеспечить отличную поддерживающую архитектуру в отношении новой концепции дизайна *Cyber Physical Systems (CPS)* [9]. Концепция *CPS* основана на тесной связи и координации между физическими и вычислительными элементами. Ранняя реализация системного распределения включала разделение управляющей программы и написание присоединяющегося кода для объединения меньших компонентов в полную систему. Более современные методы описывают понятие интеллектуального, автономного, целевого агента в многоагентной системе [10].

В данной работе авторы рассматривают расширение проектирования *CPS*, применяемого в сложных системах автоматизации, таких как *BHS*. Схема установки служит основным источником для описания архитектуры управляющего кода, который адаптируется за счет использования концепций агентов для модуляции каждого контроллера. Каждый агент выполняется непосредственно в физическом устройстве. Однако реализация этой концепции устанавливает ряд исследовательских задач, включая создание интеллектуальных агентов с использованием стандарта МЭК 61499, а также интеграцию типичных функций управления *BHS*.

Ниже авторы предлагают концепцию модульного проектирования *BHS* и новый подход к разработке системы сортировки багажа. Авторами также представлены архитектура основных базовых блоков системы и их дальнейшее разделение на модель *Model-View-Control*, автоматический формат макета и разработка приложения визуализации.

Модульная конструкция системы сортировки багажа. Использование МЭК 61499 для распределенного мультиагентного управления *BHS* было предложено в работе [11]. В процессе проектирования, который рассмотрен в работе [12], применяется шаблон проектирования *Model-View-Controller* и обеспечивается дополнительная модульность по уже присущей модульности МЭК 61499. В контексте *BHS* компонент модели представляет собой моделирование физического конвейера и может использоваться для тестирования контроллера в отсутствие физической установки. Компонент вида объединяет полезную информацию из модели или завода и отображает ее на удаленном дисплее. Проектный поток может начинаться с физического плана установки, который будет описывать архитектуру каждого компонента в триаде *MVC*. Автономное моделирование применяется до развертывания управляющего кода на заводе.

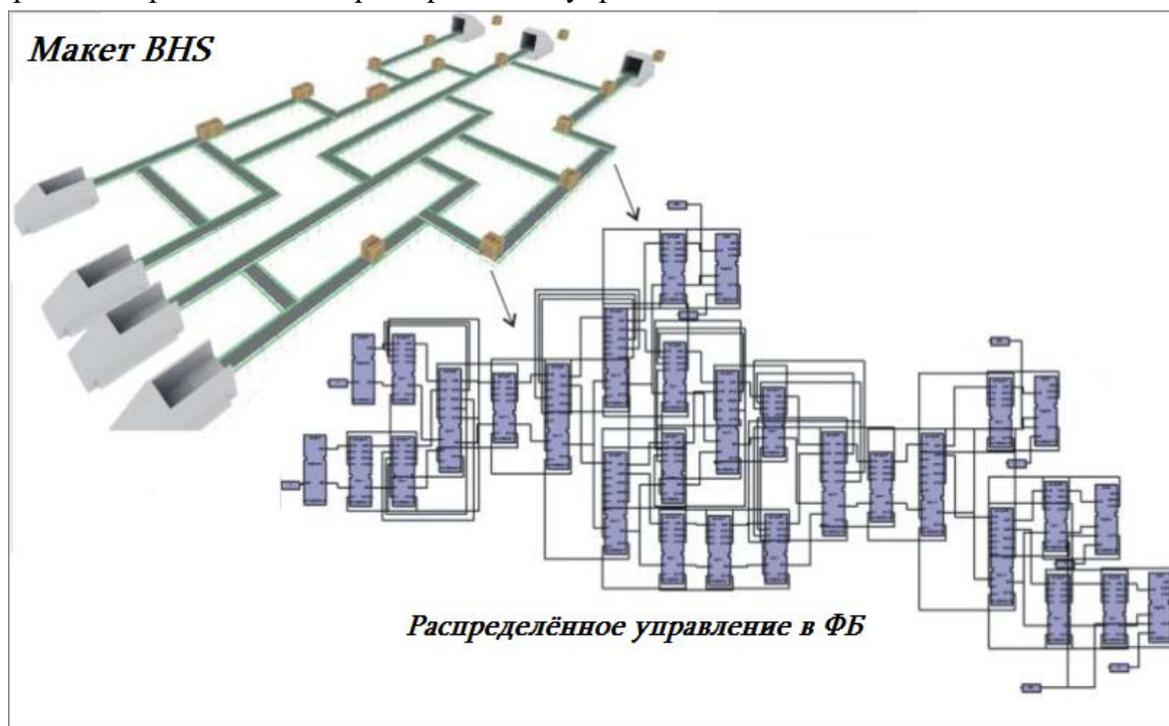


Рис. 3. Концепция дизайна *BHS*: от макета до децентрализованного управления и распределенного развертывания

Цель настоящего исследования — предложить эффективный метод автоматизации *BHS* с меньшей трудоемкостью при проектировании и проверке, который наиболее устойчив и адаптируем по сравнению с современными методами автоматизации. Авторами предложен метод проектирования, сущностью которого является получение управляющего кода автоматически, с помощью воспроизведения структуры системы (рис. 3).

Распределенный управляющий код будет состоять из экземпляров интеллектуальных агентов, явного события и потока данных между ними. Каждый тип агента будет реализован как тип ФБ в соответствии с МЭК 61499. Интеллектуальный агент структурирован внутренне и реализован как функциональный блок. *ISaGRAF* — коммерческая готовая реализация стандарта МЭК 61499. Он поддерживает стандарт МЭК 61131-3 для программирования ПЛК, а также

стандарт МЭК 61499. По сравнению с другими стандартными средами разработки МЭК 61499 (*IDE*) *ISaGRAF* имеет несколько отличительных особенностей, однако основным преимуществом является возможность беспрепятственного распространения приложений. Разработчик может указать, на каких устройствах будет находиться каждый ФБ, тогда *ISaGRAF* автоматически создает необходимую связь. Аппаратную часть можно составить из нескольких контроллеров, где каждый ПЛК будет работать в режиме реального времени под управлением операционной системы вместе со средой исполнения *ISaGRAF* для выполнения традиционных распределенных программ ПЛК и программ стандарта *IEC 61499*. Данные контроллеры представляют прототипный пример интеллектуального модуля, который может быть интегрирован в каждый конвейер или привод.

Мультиагентный дизайн *BHS* на основе МЭК 61499. *BHS* в аэропорту обычно состоит из комбинации конвейеров, диверторов, маяков, рентгеновских и других сканеров с конвейерами, составляющими большинство функциональных возможностей. Конвейеры расположены в макетах в соответствии с требованиями аэропорта, однако макет может подвергнуться значительным изменениям в течение жизненного цикла системы для учета новых маршрутов багажа, изменений в оборудовании или неисправностей. В предлагаемой архитектуре агент ФБ конвейера служит основным строительным блоком для создания кода управления *BHS* на основе макета. Каждый конвейер ФБ содержит код управления и код имитационной модели, инкапсулированный в единый модуль, структурированный в соответствии с архитектурой *MVC*. Часть *View* агента конвейера взаимодействует с автономным приложением визуализации, которое визуализирует *BHS*.

А. Компонент конвейерной модели. Компонент конвейерной модели представляет точную поведенческую модель установки (включая конвейерную ленту, датчики и исполнительные механизмы) или интерфейс с его датчиками или исполнительными механизмами. Модель отражает низкоуровневые механические функции, такие как движение конвейерной ленты и движение сумки. Это позволяет использовать и перемещать мешки, извлекать пакеты. Соответствующие фотоэлементы срабатывают в соответствии с положением мешков на конвейере. Управляющие сигналы, запускающие работу двигателя, передаются в виде булевых сигналов, представляющих состояние фотоэлемента.

Б. Компонент конвейерного контроллера. Контроллер конвейера обрабатывает низкоуровневые механические функции, а также обеспечивает интеллектуальное поведение: слияние, отведение и отслеживание багажа. Распределенная система состоит из многоуровневых программных модулей, поэтому при сортировке должны использоваться распределенные методы отслеживания. Когда багаж передается с одного конвейера на другой, вместе с ним передается и вся его информация. Высокоуровневая функциональность контроллера была дополнительно разделена логически с использованием понятия виртуального конвейера. В работе [9] для каждого конвейерного модуля использовался один параметризованный контроллер. Однако в данном исследовании контроль высокого уровня был разделен на отдельные виртуальные модули, называемые виртуальными конвейерами. Каждый виртуальный конвейер представляет собой элемент управления для простой конвейерной секции, состоящей не более чем из одной точки слияния, одной точки переадресации и одного фоточувствительного датчика. Контроллер для более сложной секции конвейера будет представлять собой совокупность нескольких контроллеров виртуального конвейера. Конвейерная секция, изображенная на рис. 4, представляет

собой пример с двумя фоточувствительными датчиками и диверторами, следующими за каждым фотоэлементом.

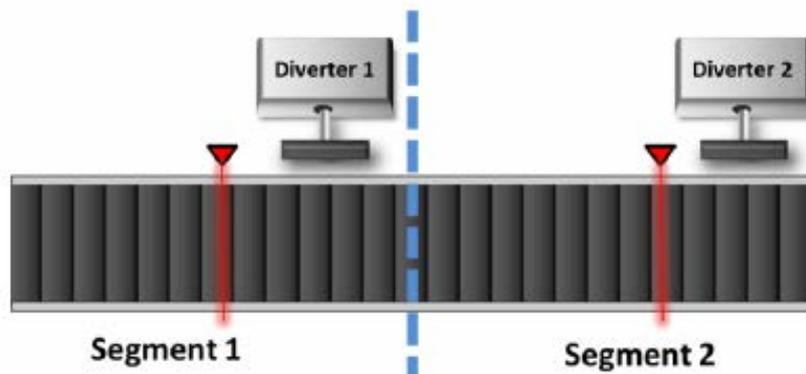


Рис. 4. Пример секции транспортера с двумя фоточувствительными датчиками и двумя диверторами

На рис. 4 управляющий код для каждого раздела *Segment1* и *Segment2* практически идентичен, причем каждый обрабатывает пакеты на своих сегментах. Логическое разбиение выполняется таким образом, чтобы первый и второй сегменты содержали один фотоэлемент и дивертор. Структура кода состоит из виртуального контроллера для каждого сегмента, который обрабатывает пакеты только в соответствующем сегменте. Это продемонстрировано на диаграмме ФБ на рис. 5. Контроллер виртуального конвейера создается столько раз, сколько необходимо в составном ФБ, чтобы написать управляющий код для всей секции транспортера.

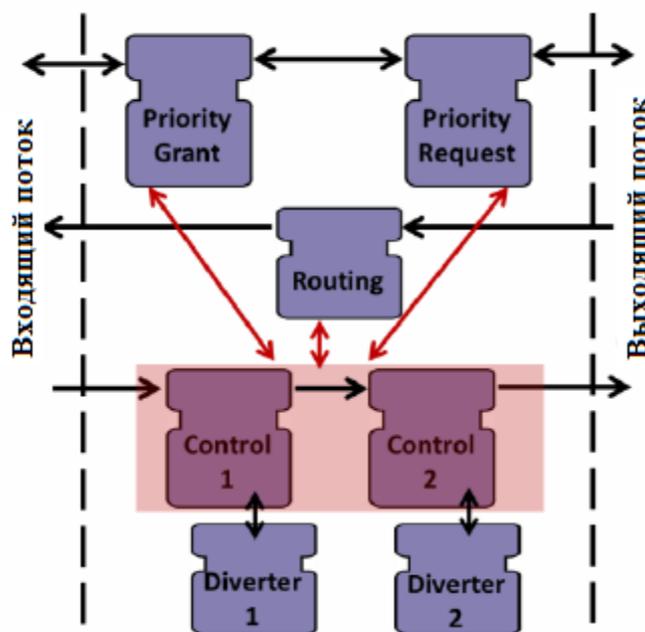


Рис. 5. Внутренняя структура соответствующего управляющего кода *FB* для конвейера с 2-мя фотоэлементами и 2-мя диверторами

С. Компонент вид конвейера. Компонент *MVC triad* отвечает за обеспечение точного представления состояния системы для внешнего приложения визуализации. Авторы разработали данное приложение для отображения состояния *BHS* на локальном или удаленном компьютере. При этом компонент вида предоставлял данные для визуализации, передавая их на OPC-сервер.

Д. Планирование динамического маршрута. Маршрутизация багажа для правильного направления — главная цель любой *BHS*. Планирование маршрута сделано в отдельном приложении, в котором содержится информация о топологии *BHS*. Каждый конвейер связан только со своими ближайшими соседями. Кроме того, авторы предполагают, что из-за возможных неисправностей набор доступных активных конвейеров будет меняться с течением времени. Разработанная распределенная система *BHS* ФБ нацелена на то, чтобы обеспечить возможность маршрутизации багажа эквивалентно централизованной системе.

Основой решения задачи маршрутизации является распределенная система *Bellman-Ford* (*DBF*). Алгоритм решения простой задачи представлен на рис. 6. В момент времени $t = 0$ таблица маршрутизации для точки *A* содержит только метрики расстояния для своих прямых соседей: точки *B* и точки *C*. На следующей итерации точка *A* будет получать таблицы маршрутизации от точки *B* и точки *C*. Новая информация, полученная в ходе этой итерации, выделяется, следовательно, точка *A* обладает сведениями о пересылке к точке *D* и точке *E* через промежуточную точку. Система транслирует положение точки относительно ее соседей, обрабатывая поступающую информацию о кратчайших путях.

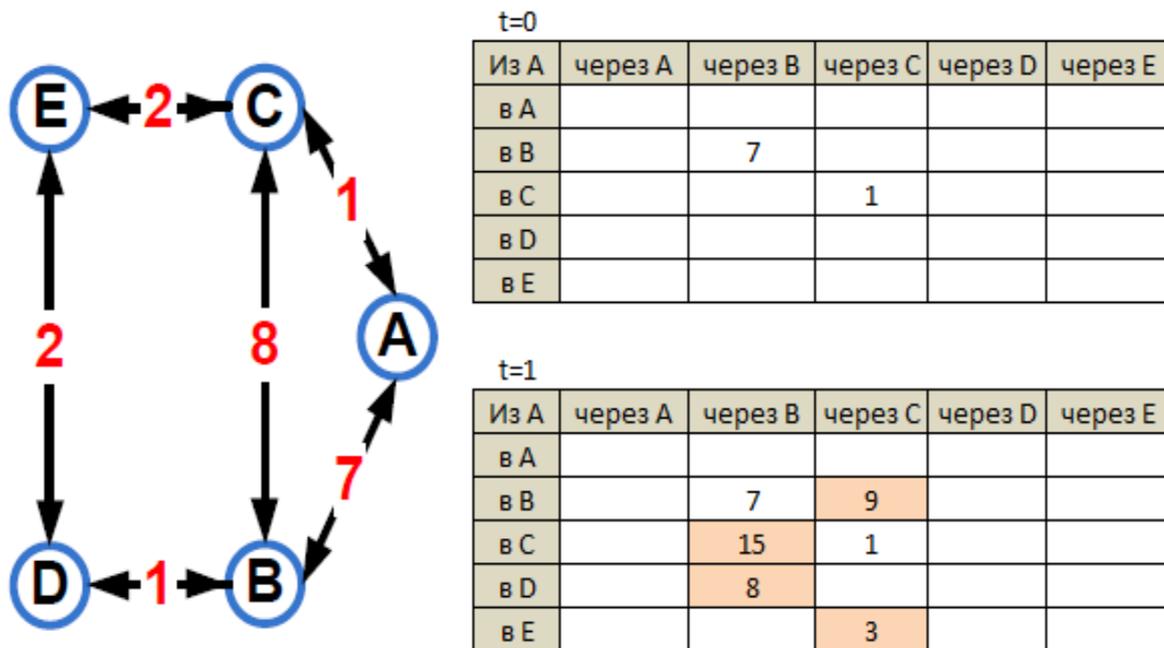


Рис. 6. Сеть из 5 маршрутизаторов с метриками расстояний и таблицами маршрутизации для узла *A* в момент времени $t = 0$ и $t = 1$

Заключение. Авторы рассмотрели возможность реализации *BHS*-контроля в распределенном виде. Подход предусматривал разработку ФБ управления конвейером, который инкапсулировал типовые функции конвейера в автономный модуль. Этот подход продвигает проект *BHS* к реконфигурации путем связывания вычислительных элементов (ФБ) с физическими элементами (конвейерами).

Проектирование системы включало в себя соединение конвейерных ФБ с зеркалом схемы физической системы *BHS*. Интеграция форматов обмена отраслевыми стандартами дизайна, таких как *CAD*, представляется перспективным шагом в упрощении процесса проектирования для крупных распределенных систем управления.

Реализация стандарта *IEC 61499* в *ISaGRAF* позволит ФБ работать на целевом оборудовании с простой пользовательской спецификацией. Один или несколько ФБ могут быть

назначены для работы на одном контроллере. Если управляющий код уже существует в виде модульных ФБ, то он будет скрыт от разработчика системы. Таким образом, только программирование с помощью визуальных методов станет идеальным решением поставленной задачи. Авторы планируют более подробное исследование с целью точного сравнения распределенного решения управления *BHS* с традиционными централизованными решениями.

Библиографический список.

1. Valeriy Vyatkin. Distributed Execution and Cyber-Physical Design of Baggage Handling Automation with IEC 61499 [Электронный ресурс] // IEEE Conference on Industrial Informatics, Lisbon, Portugal, 2011. URL: http://www.vyatkin.org/publ/INDIN11_YV.pdf (дата обращения: 16.01.2017).
2. Портал искусственного интеллекта [Электронный ресурс] / Многоагентные системы. — Режим доступа : <http://www.aiportal.ru/articles/multiagent-systems/multiagent-systems.html> (дата обращения : 01.02.17).
3. P. Valckenaers and H. Van Brussel, "Holonc Manufacturing Execution Systems," *CIRP Annals*, pp. 427-432, 2005.
4. Передовые технологии и технические решения [Электронный ресурс] / Обзор стандарта IEC 61499. — Режим доступа : <http://www.picad.com.ua/0405/pdf/8-12.pdf> (дата обращения : 01.02.17).
5. IEC 61131-3 [Электронный ресурс] / Национальная библиотека им. Н. Э. Баумана. — Режим доступа : http://ru.bmstu.wiki/IEC_61131-3 (дата обращения : 03.02.17).
6. Промышленная сеть [Электронный ресурс] / Википедия. — Режим доступа : https://ru.wikipedia.org/wiki/Промышленная_сеть/ (дата обращения : 03.02.17).
7. J. H. Christensen, "Design patterns for system engineering with IEC 61499," in Conference Verteilte Automatisierung, Magdeburg, Germany, 2000, pp. 63-71.
8. J. Christensen, "HMS/FB Architecture and its Implementation," in Agent Based Manufacturing. Advances in the Holonic Approach, M. Deen, Ed., ed: Springer, 2003.
9. E. A. Lee, "Cyber Physical Systems: Design Challenges," in Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on, 2008, pp. 363-369.
10. HOLOBLOC., "Function Block Development Kit," ed, 2008. <http://www.holobloc.com/>.
11. V. Vyatkin, et al., "Information Infrastructure of Intelligent Machines based on IEC61499 Architecture," *IEEE Industrial Electronics Magazine*, vol. 1, pp. 17-29, 2007.
12. G. Black and D. V. Vyatkin, "Intelligent Component-based Automation of Baggage Handling Systems with IEC 61499," *IEEE Transactions on Automation Science and Engineering*, vol. 6, p. in print, 2009.