

Рис. 5. Генетический алгоритм.

УДК 621.3

Майкив И.М., Кочан Р.В., Кочан В.В., Саченко А.О., Турченко И.В.

## СЕТЕВОЙ ПРИКЛАДНОЙ ПРОЦЕССОР, РЕАЛИЗОВАННЫЙ НА ПРОГРАММИРУЕМОЙ ЛОГИЧЕСКОЙ МАТРИЦЕ

### ВВЕДЕНИЕ

Усложнение алгоритмов обработки данных в системах управления технологическими процессами требует использования многопроцессорных иерархических систем. Поэтому, современные измерительно-управляющие системы строятся в виде локальных сетей с распределенными вычислительными ресурсами. Базовым элементом таких сетей является сетевой процессор (СП), выполняющий ряд заданных функций, связанных с обработкой сенсорных данных и взаимодействием с другими узлами сети.

Области использования таких систем и решаемые задачи весьма разнообразны. Поэтому СП должны быть многофункциональными элементами, способными обеспечить обработку данных в реальном времени и прозрачный обмен результатами в сети. Поскольку оборудование автоматизации технологических процессов выпускает множество фирм, на первый план выступают вопросы унификации интерфейсов и протоколов взаимодействия на всех уровнях сети. Их решение обеспечит открытость, гибкость, масштабируемость сетей, возможность их адаптации к различным реальным задачам, долгий жизненный цикл оборудования.

Стандарты, унифицирующие требования к интерфейсам

на аппаратном уровне, известны уже давно и постоянно совершенствуются [1]. В тоже время стандарты, устанавливающие требования к программному обеспечению и протоколам организации взаимодействия в таких сетях, появились значительно позже, и их внедрение вызывает определенные трудности. В результате, СП различных производителей зачастую не совместимы между собой на программном уровне.

Для решения этой проблемы была разработана серия стандартов IEEE-1451 [2-4], регламентирующая требования к обеспечению аппаратной и программной совместимости элементов распределенных измерительно-управляющих сетей. Структура такой сети, удовлетворяющая требованиям IEEE-1451, включает (рис. 1):

1. Интерфейсные модули преобразователей (ИМП) – узлы нижнего уровня сети, к которым непосредственно подключаются сенсоры и исполнительные механизмы;
2. Сетевые прикладные процессоры (СПП) – промежуточный уровень сети, управляют работой ИМП, и обрабатывают текущие данные, поступившие из ИМП;
3. Центральный сервер – верхний уровень сети, обеспечивает ее функционирование. Сюда же включены и другие потребители информации.

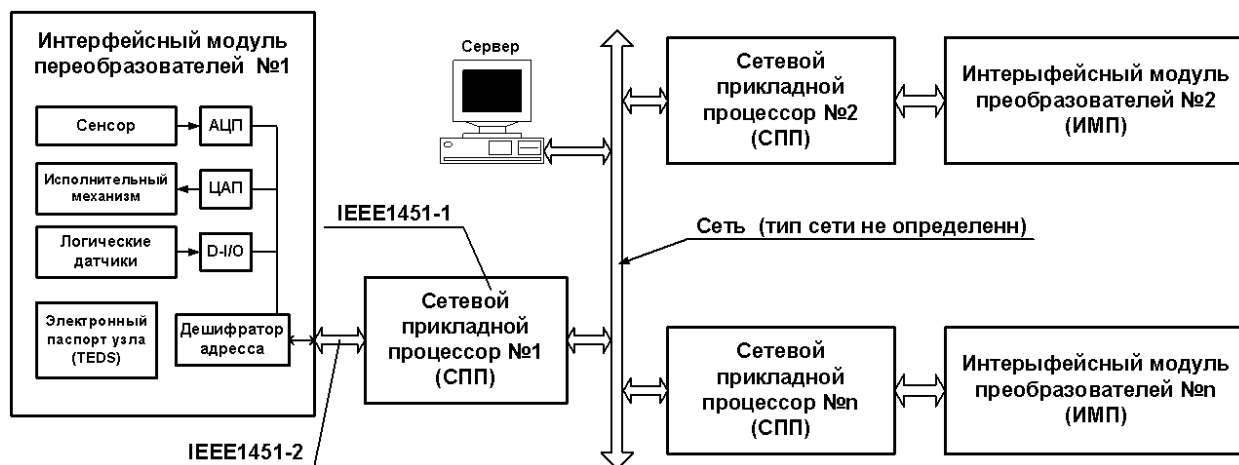


Рис. 1. Структура сети в соответствии с требованиями IEEE-1451.

Майкив И.М., Кочан Р.В., Кочан В.В., Саченко А.О., Турченко И.В., Научно-исследовательский институт Интеллектуальных компьютерных систем.

Украина, 46000, г. Тернополь, ул. Львовская, 11. [mim@tanet.edu.te.ua](mailto:mim@tanet.edu.te.ua)

Важной особенностью сети является использование в ИМП электронного паспорта (TEDS - Transducer Electronic Data Sheet), что позволяет применить в сети технологию "plug-and-play". Однако требования к связанному интерфейсу между СПП и ИМП жестко регламентированы стандартами IEEE-1451.2 и IEEE-1451.4, а тип сети верхнего уровня не задан. Поэтому, известные реализации СПП [5-7] ориентированы на определенный тип сети верхнего уровня, что значительно снижает их универсальность. Кроме того, стандарт IEEE-1451.2 определяет новый тип интерфейса ТИ (Transducer Independent Interface, модифицированная версия SPI), которым устанавливается "point-to-point" соединение между СПП и ИМП, ограничивая количество ИМП, подключаемых к одному СПП. Для решения этой проблемы был разработан стандарт IEEE-1451.4, в котором предполагается организация локальных подсетей нижнего уровня с использованием нового интерфейса - MMI (Mode Mixed Interface). Однако, ТИ и MMI не совместимы с широко распространенными интерфейсами (RS-232, RS-485, 1-Wire, LIN, CANbus и др.).

### ПРЕДЛАГАЕМАЯ СТРУКТУРА СЕТИ

Исходя из проведенного анализа, предложена [8] усовершенствованная структура распределенной информационно-управляющей сети (рис.2). В ней СПП должен обеспечить поддержку набора различных коммуникационных интерфейсов (RS-232, RS-485, SPI, I<sup>2</sup>C, 1-Wire, LIN, CANbus, ТИ, Ethernet и др.), каждый из которых может свободно использоваться в сети как верхнего, так и нижнего уровня.

В этом случае предложенная структура сможет обеспечить ряд преимуществ:

1. Совместимость с компонентами реализованных систем и используемыми в них модулями сбора сенсорных данных.
2. Уменьшение затрат на обновление и модернизацию системы.
3. Уменьшение нагрузки на сеть верхнего уровня.

Возможны несколько путей решения поставленной задачи. При аппаратной реализации интерфейсов используется набор специализированных микросхем, каждая из которых поддерживает один из необходимых интерфейсов, что ведет к аппаратной избыточности и высокой цене изделия. Более эффективна программная реализация набора требуемых интерфейсов. Этот подход реализован в СПП [9], в котором обеспечена поддержка интерфейсов RS-232, RS-485, SPI, LPT и ISA-8, первые три из которых двойные и могут использоваться в сети и верхнего и нижнего уровня. СПП использует двухпроцессорную архитектуру (микроконтроллеры серии MCS-51), программную реализацию интерфейсов выполняет отдельный процессор. Этот СПП обеспечивает возможность дистанционной замены программы обработки данных в режиме on-line и ориентирован на системы с традиционными промышленными датчиками, которые не нуждаются в интерфейсах с большой пропускной способностью. Главные его преимущества – минимум аппаратного обеспечения и гибкость структуры.

Однако одновременная реализация двух (как минимум) интерфейсов сильно загружает интерфейсный контроллер, что, даже при использовании высокопроизводительных процессоров, не позволяет обеспечить поддержку интерфейсов с большой скоростью передачи данных (CANbus, USB, Ethernet и др.). Это требует разработки универсального СПП, который должен обеспечить:

1. Поддержку набора периферийных интерфейсов (включая скоростные), которые можно произвольно использовать как в сети верхнего, так и нижнего уровней.
2. Возможность реконфигурации типов и параметров интерфейсов обеих уровней, в соответствии с требованиями пользователя.
3. Возможность дистанционной замены программного обеспечения в режиме on-line.

### ПРЕДЛАГАЕМАЯ СТРУКТУРА СЕТЕВОГО ПРИКЛАДНОГО ПРОЦЕССОРА

Предлагается реализовать СПП в виде двухпроцессорной архитектуры, в которой те узлы интерфейсного контроллера, от которых требуется высокая производительность, реализуются аппаратно с помощью программируемой логической матрицы (ПЛИМ). Обобщенная структурная схема предложенного СПП представлена на рис. 3.

Она включает главный контроллер (ГК), который производит обработку данных, интерфейсный контроллер (ИК), который обеспечивает поддержку интерфейсов сетей верхнего и нижнего уровней, а также блоки памяти программ и данных. Интерфейсный контроллер состоит из конфигурационного контроллера (КК), ПЛИМ и аппаратного драйвера, который обеспечивает согласование уровней напряжений и токов между ПЛИМ и линиями связи поддерживаемых интерфейсов. Кроме того, в состав КК входит библиотека интерфейсов – постоянная память, в которой хранится набор программ, необходимых для конфигурации ПЛИМ. КК обеспечивает:

1. начальную конфигурацию ПЛИМ (путем загрузки из библиотеки интерфейсов соответствующего файла);
2. последующую реконфигурацию ПЛИМ согласно требованиям главного контроллера;
3. программную поддержку интерфейсов верхнего и нижнего уровня;
4. дистанционное перепрограммирование главного контроллера (после того, как из сети верхнего уровня поступила соответствующая команда).

При этом использовано так называемое "тесное взаимодействие" КК и ПЛИМ – на базе последней реализованы только высокочастотные узлы интерфейсов – в основном регистр сдвига и программируемый делитель частоты. Остальные функции интерфейса реализует КК программно. Это резко снижает требуемый объем ПЛИМ. Такая структура СПП совместима с предложенной структурой распределенной информационно-управляющей сети (рис.2), полностью отвечает ее требованиям и обеспечивает:

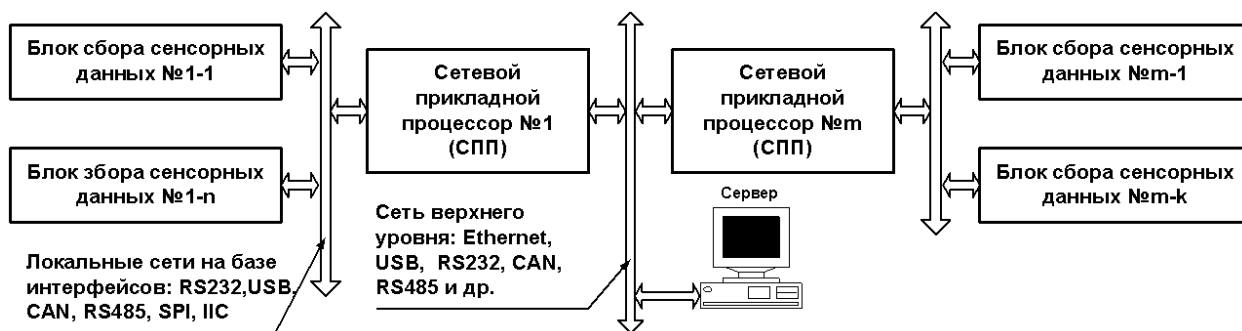


Рис. 2. Новая структура распределенной информационно-управляющей системы.

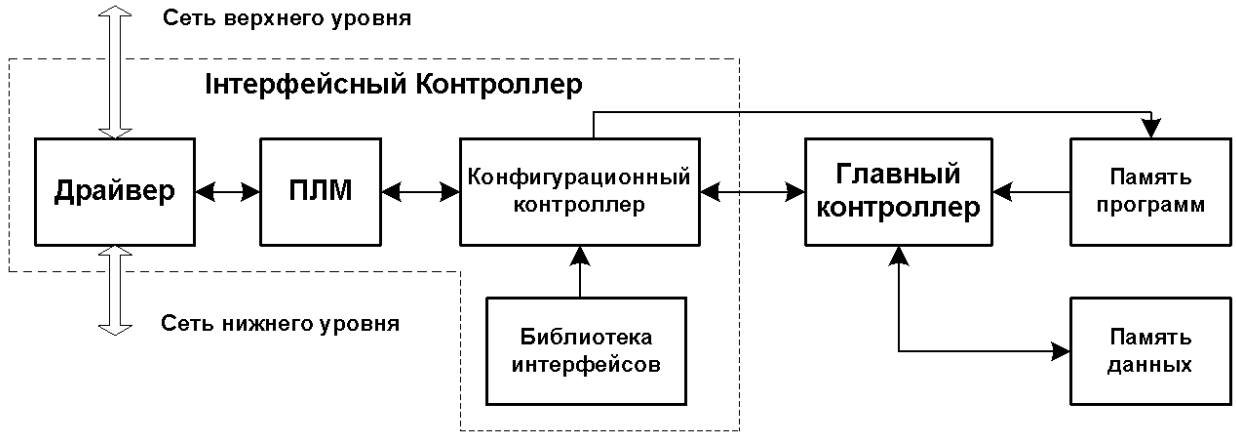


Рис. 3. Обобщенная структура предложенного СПП.

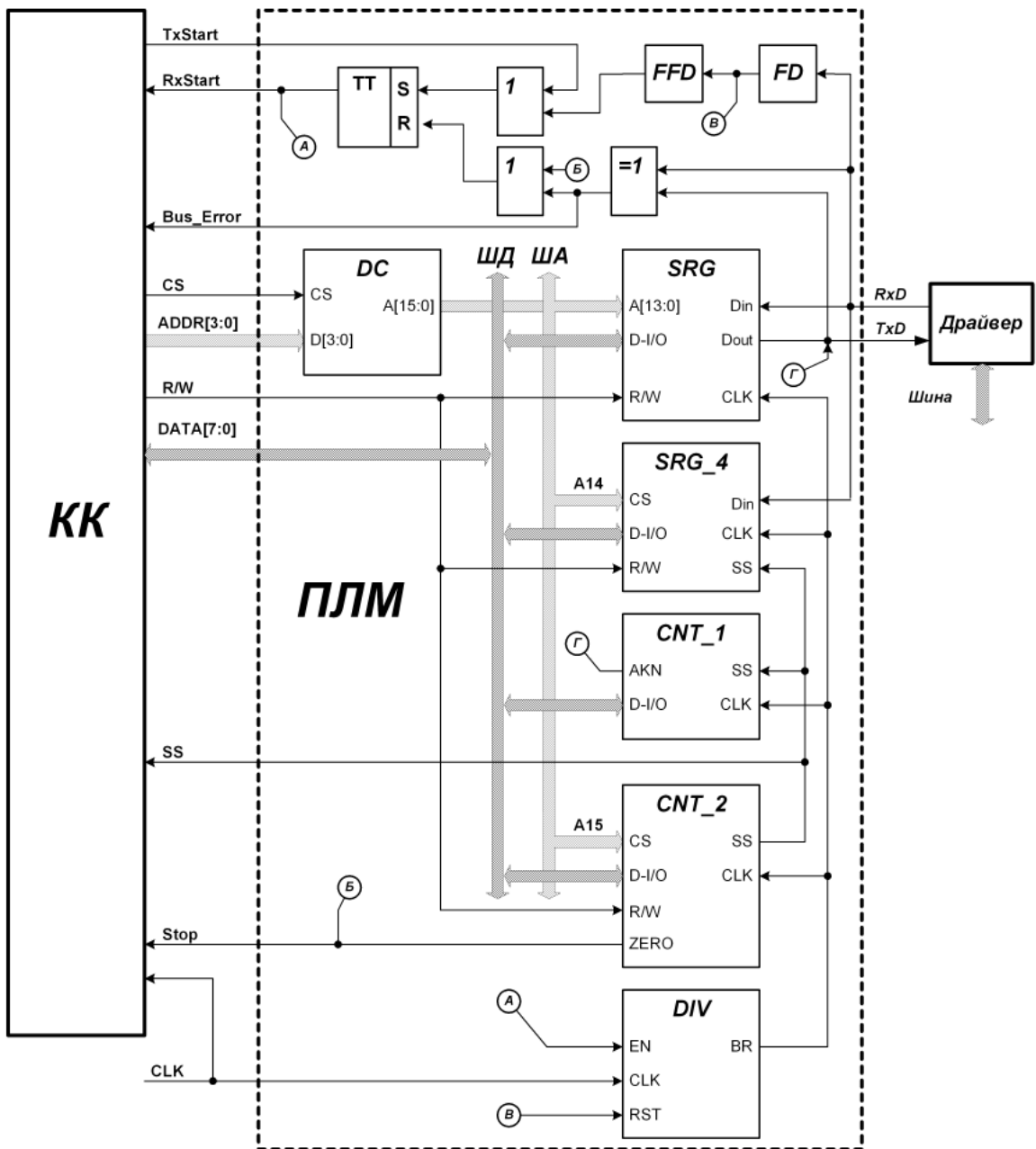


Рис. 4. Обобщенная структура интерфейсного контроллера CANbus.

1. поддержку широкого набора периферийных интерфейсов (RS-232, SPI, I<sup>2</sup>C, 1-Wire, LIN, CANbus, TII, Ethernet), которые можно использовать в сетях как верхнего, так и нижнего уровней в соответствии с требованиями программы обработки данных;
2. Возможность дистанционной замены программного обеспечения в режиме on-line.

Рассмотрим структуру ПЛИМ (рис. 4) при реализации наиболее сложного из упомянутых интерфейсов – CANbus – согласно спецификации [1, 10]. Ее основными компонентами являются регистр сдвига SRG и реверсивный счетчик битов CNT-2 премо-передатчика, делитель частоты тактовых импульсов DIV и дешифратор DC. Кроме того, в схему входят дополнительные регистр сдвига SRG-4 счетчик битов CNT-1, идентификаторы фронтов FD и первого бита сообщения FFD и триггер режима ТТ.

КК имеет возможность побайтного чтения и записи содержимого SRG. Тип операции чтения или записи определяется управляющими сигналами CS и R/W. Для сокращения количества используемых выводов, адресация байтов SRG выполняется с помощью дешифратора DC. Синхронизация сдвига битов SRG производится стробирующими импульсами, формируемыми делителем частоты DIV, на вход которого поступают импульсы синхронизации (они поступают и на КК). Подсчет количества стробирующих импульсов в одном сообщении выполняется CNT-2.

В режиме передачи КК с помощью DC записывает побайтно в регистр SRG данные (согласно [10]), а в CNT-2 – информацию об общей длине сообщения (включая все служебные биты). Для начала передачи КК, установив на выходе TxStart лог.1, переводит триггер режима ТТ в активное состояние (лог.1), что разрешает работу делителя DIV – он начинает формировать стробирующие импульсы, которые сдвигают данные, записанные в SRG, на аппаратный драйвер CAN (он приводит уровни сигналов к требованиям CAN шины). Стробирующие импульсы поступают также на вход счетчика CNT-2 (в режиме вычитания). В момент, когда достигается записанное КК в CNT-2 значение длины сообщения, CNT-2 формирует сигнал Stop на входе КК. Одновременно ТТ переводится в пассивное состояние. Для реализации неразрушающего арбитража в схему введен элемент “исключающее ИЛИ” – при их не совпадении он принудительно переводит ТТ в пассивное состояние. Информация об этом сигналом Bus\_Error передается на КК.

В режиме приема селектор первого импульса FFD перево-

дит ТТ в активное состояние. При этом включается DIV, начинающий формировать стробирующие импульсы, по которым происходит сдвиг принимаемых битов в SRG (прием сообщения). Одновременно CNT-2 формирует дополнительный строб-импульс SS, соответствующий 16-19 битам сообщения, которые определяют его длину. Они записываются в дополнительный регистр сдвига SRG-4. Сигнал SS своим спадающим фронтом информирует КК о наличии в SRG-4 данных о длине сообщения. Одновременно длина сообщения записывается в CNT-1 для определения момента формирования сигнала подтверждения приема AKN (лог. 0 на шине). После задержки, соответствующей длине сообщения, КК считывает (с помощью DC) содержание SRG и выполняет проверку достоверности принятого сообщения. В процессе приема частота стробирующих импульсов приемника автоматически синхронизируется с частотой передатчика путем периодического сброса DIV по фронтам импульсов принимаемого сообщения, которые идентифицируются FD.

Интерфейсный контроллер реализован на ПЛИМ фирмы Xilinx типа XC3S200-4 и микроконтроллере AT89C52, работающем на частоте 24 МГц.

### РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ

Проведем оценку эффективности предложенного метода реализации интерфейсных контроллеров. В таблицах 1-3 представлены примеры [11] реализованных полностью на ПЛИМ, контроллеров широко распространенных последовательных интерфейсов – I<sup>2</sup>C, SPI, CAN (рассматриваются версии контроллеров, для реализации которых использовано минимальное количество макроячеек ПЛИМ). В первой строке таблиц указаны названия фирм-разработчиков, а в последующих строках приводятся: (i) тип микросхем, на которых реализован контроллер; (ii) поддерживаемый режим работы; (iii) количество макроячеек, используемых для реализации; (iv) максимальная рабочая частота. В последней колонке каждой таблицы представлены параметры контроллера, разработанного авторами согласно изложенной концепции “тесного взаимодействия” конфигурационного контроллера и ПЛИМ.

Таким образом, из представленных результатов следует, что предложенный метод реализации интерфейсных контроллеров позволяет резко, в 1,5-2 раза, уменьшить требования к количеству макроячеек ПЛИМ (объему), необходимому для его реализации. Для универсального интерфейсного контроллера необходимый объем ПЛИМ определяется наиболее сложным

Таблица 1. Контроллеры интерфейса I<sup>2</sup>C, реализованные на ПЛИМ (FPGA и CPLD)

Разработчик	CAST	Memec Design		DigitalCore Design		XILINX	
Тип м-с	XC2S100-6	XC2S30-5	XC2S30-5	XC2S50-6	XC2S50-6	XCR3256-7	XC3S200-4
Раб. режим	Master/Slave	Master	Slave	Master	Slave	Master/Slave	Master/Slave
Кол. м.я.	252	151	123	149	47	123	77
Частота, МГц	103	50	50	86	104	62	62

Таблица 2. Контроллеры интерфейса SPI, реализованные на ПЛИМ (FPGA и CPLD)

Разработчик	CAST	CAST	XILINX	
Тип м-с	XC2S200-6	XC2S200-6	XCR3256-7	XC3S200-4
Раб. режим	Master/Slave	Slave	Master	Master/Slave
Кол. м.я.	143	111	125	60
Частота, МГц	38	80	48	48

Таблица 3. Контроллеры интерфейса CANbus, реализованные на ПЛИМ (FPGA)

Разработчик	CAST	MemecDesign	Intelliga	XYLONX	
Тип м-с	XC2S100-6	XC2S100-5	XC2S50E-7	XC2S50E-7	XC3S200-4
Раб. режим	CAN 2.0B	CAN 2.0B	CAN 2.0B	CAN 2.0B	CAN 2.0A
Кол. м.я.	1009	374	346	431	198
Частота, МГц	26	18	16	24	12

реализуемым интерфейсом, в данном случае CANbus. Характеристики реализованных интерфейсных контроллеров полностью отвечают требованиям их стандартов и спецификаций. Предложенный метод является экономически эффективным, особенно при использовании в качестве конфигурационного контроллера не дорогого микроконтроллера.

#### СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. "Протоколы прикладного уровня CAN-сетей": Современные технологии автоматизации №3/99 ст.6-15.
2. "IEEE Std 1451.2-1997, Standard for a Smart Transducer Interface for Sensors and Actuators - Transducer to Microprocessor Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats", Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey 08855, September 26, 1997.
3. "IEEE Std 1451.1-1999, Standard for a Smart Transducer Interface for Sensors and Actuators - Network Capable Application Processor (NCAP) Information Model", Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey 08855, June 25, 1999.
4. "IEEE Std. 1451.4-2004, Standard for a Smart Transducer Interface for Sensors and Actuators - Mixed-Mode Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats", 15 December, 2004.
5. D. Wobshall, An Implementation of IEEE 1451 NCAP for Internet Access of Serial Port-Based Sensors, Proceedings of second Sensor for Industry Conference SIcon/02, 19-21 November 2002, Houston, Texas, ISBN 1-55617-834-4, pp.157-160.
6. Depari, P. Ferrari, A. Flammini, D. Marrioli, E. Sissini, A. Tarroni, IEEE1451 Smart Sensors Supporting USB Connectivity. Proceedings of the ISA/IEEE 2004 Sensors for Industry Conference (Sicon/04). New Orleans, Louisiana, USA, 27-29 January 2004. pp. 177-182
7. T. Brooks, S. Chen, K. Lee, IEEE 1451 Smart Wireless Machinery Monitoring and Control for Naval Vessels. Presented at the Thirteenth International Ship Control Systems Symposium (SCSS) in Orlando, Florida, on April 7-9, 2003.
8. V.Golovko, L.Grandinetti, V.Kochan, T.Laopoulos, A.Sachenko, V.Turchenko, V.Tymchyshyn. Approach of an Intelligent Sensing Instrumentation Structure Development. Proceedings of the IEEE International Workshop on Intelligent Signal Processing, Budapest, Hungary, 4-7 September, 1999, pp.336-341.
9. R. Kochan, K. Lee, V. Kochan, A. Sachenko, Development of a Dynamically Reprogrammable NCAP, Proceedings of the IEEE Instrumentation and Measurement Technology Conference IMTC/2004, May 18-20, pp. 1188-1193.
10. Robert Bosch GmbH. CAN Specification 2.0 Part A+B, September 1991.
11. "<http://www.xilinx.com/xlnx/xebiz/search/searchresult.jsp?sGlobalNavPick=PRODUCTS&sSecondaryNavPick=IntellectualXilinx : Locator Results for IP Solutions+Property&category=-538081132&iLanguageID=1&ResultsView=Standard&IPCategory=Bus+Interface+and+IO&IPProducts=Core>."

УДК 004.8

*Якимов А.И., Захарченков К.В.*

## БИБЛИОТЕКА ИМИТАЦИОННЫХ МОДЕЛЕЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПРОМЫШЛЕННОГО ПРЕДПРИЯТИЯ

**Введение.** Информационная система (ИС) предприятия является объектом со сложным составом и структурой компонентов. Ряд параметров ИС может изменяться случайным образом, например, количество и объем заказов клиентов, появление сбоев и отказов в работе оборудования, время пополнения запасов ресурсов. Это обуславливает необходимость создания имитационных моделей (ИМ) для оценки параметров и выбора состава и структуры ИС предприятия [1].

Моделирование ИС предприятия предполагает реализацию бизнес-процессов корпоративной информационной системой (КИС) в соответствии с функциями МRP II [2] на основе имитационного моделирования. Такой подход к моделированию ИС позволяет оценивать эффективность функционирования базового и проектного вариантов ИС предприятия и принимать решение по рациональному выбору состава, структуры и параметров ИС [3, 4].

**Состав и структура библиотеки имитационных моделей.** Для моделирования ИС предприятия разработана библиотека ИМ (рис. 1), являющаяся предметным расширением программно-технологического комплекса имитации сложных систем (ПТКИ) BelSim [5] при исследовании ИС. Библиотека ИМ ИС промышленного предприятия включает ИМ бизнес-процесса планирования продаж и производства, ИМ бизнес-процесса управления запасами ресурсов, ИМ ИС производ-

ственного процесса, а также ИМ функционирования ИС в корпоративной сети.

Все ИМ, входящие в состав библиотеки ИМ ИС предприятия реализованы на основе процессного способа имитации [6] и предназначены для рационального выбора параметров подсистем ИС [7]. Каждая ИМ, кроме процессов, реализованных в ИС, включает процессы финансово-хозяйственной деятельности промышленного предприятия, непосредственно связанных с процессами ИС.

**Имитационная модель бизнес-процесса планирования продаж и производства.** ИМ бизнес-процесса планирования продаж обеспечивает возможность анализа эффективности различных методов планирования продаж, реализованных в КИС, с точки зрения экономических показателей работы предприятия. Структуры данных ИМ предназначены для хранения параметров процессов модели, которые, в свою очередь, реализуют бизнес-логику ее работы.

В структуре *TOrder* (рис. 2) хранятся параметры заказов: даты поступления; виды продукции; объемы заказов; даты их выполнения; максимальное время ожидания выполнения заказов после истечения срока поставки; распределение объема заказов по видам продукции; распределение отгрузок продукции по датам; интервал времени между поступлениями заказов.

*Якимов Анатолий Иванович, к.т.н., доцент каф. «Автоматизированные системы управления» Государственного учреждения высшего профессионального образования «Белорусско-Российский университет».*

*Захарченков Константин Васильевич, аспирант каф. «Автоматизированные системы управления» Государственного учреждения высшего профессионального образования «Белорусско-Российский университет».*

*Беларусь, БРУ, 212005, г.Могилев, пр.Мира, 43.*