

№ 4

2013

апрель

Научно-практический
журнал для специалистов

Республика Беларусь, г. Минск

ЭЛЕКТРОИНФОРМАТИКА

Тема номера:

*«Интеллектуальные системы
и искусственный интеллект»*

Комплектация для систем защиты кабельной продукции в автомобильной и других отраслях промышленности



Schlemmer
CABLE PROTECTION SYSTEMS

Fek
company

ООО «ФЭК» РБ, 220015, г. Минск, пр. Пушкина, 29 Б,
тел./факс: +375 (17) 210-21-89,
+375 (29) 370-90-92, +375 (29) 274-17-13,
e-mail: info@fek.by, www.fek.by

ПОДПИСКА В БЕЛАРУСИ: «Белпочта» – 00822
ПОДПИСКА В РОССИИ: «Роспечать» – 00822
«АРЗУ – Почта России» – 91654

ISSN 1999-7515



ЭЛЕКТРОНИКА, инфо апрель №4

Строительно-монтажные клеммы WAGO



- надежное подключение одножильных/маложильных медных и/или одножильных алюминиевых проводников, а также **смешанный электромонтаж**

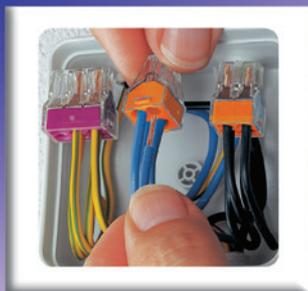
- быстрое и надежное подсоединение без использования инструмента

- надежная защита от прикосновения к выводам проводника

- гарантированная надежность контактов, исключающая короткое замыкание

- возможность измерения электрических параметров цепи без разбора и нарушения изолированности соединения

- качество подключения не зависит от аккуратности электромонтажника



ООО «ФЭК» —

единственный официальный дистрибьютор компании WAGO на территории Республики Беларусь, 220015, г. Минск, пр-т Пушкина, 29 Б, тел./факс: 210-21-89, +375 (29) 274-17-13, +375 (29) 370-90-92, www.fek.by e-mail: wago@fek.by



С В Е Т О Д И О Д Н А Я П Р О Д У К Ц И Я

Светодиодные экраны • Светодиодные табло • Светодиодные вывески
Табло "бегущая строка" • Информационные табло
Светодиодная продукция для освещения

ИЗГОТОВЛЕНИЕ ПОД ЗАКАЗ

для помещений и уличного исполнения

СОБСТВЕННОЕ ПРОИЗВОДСТВО • ГАРАНТИЯ КАЧЕСТВА

ДОСТУПНОСТЬ • НАДЕЖНОСТЬ • ОПЕРАТИВНОСТЬ



ЭЛЕКТРОКОНТИНЕНТ

КОНТРАКТНОЕ ПРОИЗВОДСТВО
Э Л Е К Т Р О Н И К И

www.elcontinent.by

Республика Беларусь, 220026, г. Минск, пер. Бехтерева, 8, офис 35

тел. (+375 17) 205 06 94, 296 31 61, Velcom (+375 29) 115 35 75

e-mail: info@elcontinent.com

**ЖУРНАЛ ИЗДАЕТСЯ ПРИ ИНФОРМАЦИОННОЙ ПОДДЕРЖКЕ
ФАКУЛЬТЕТА РАДИОФИЗИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ БЕЛГОСУНИВЕРСИТЕТА.
ЖУРНАЛ ВКЛЮЧЕН В СПИСОК НАУЧНО-ТЕХНИЧЕСКИХ ИЗДАНИЙ ВАК РЕСПУБЛИКИ БЕЛАРУСЬ**



**International magazine
of amateur and professional electronics
№4 (94) апрель 2013**

Зарегистрирован
Министерством информации
Республики Беларусь

Регистрационный №71
от 05 марта 2009 года

Главный редактор:
Раковец Леонид Иванович
e-mail: electronica@nsys.by

Заместитель главного редактора
Асмоловская Ирина Михайловна
i.asmalouskaya@electronica.by

Технический редактор:
Бензарь Валентин Кузьмич, д.т.н.

Редакционная коллегия:

Председатель:
Чернявский Александр Федорович
академик НАН Беларуси, д.т.н.

Секретарь:
Садов Василий Сергеевич, к.т.н.
e-mail: sadov@bsu.by

Члены редакционной коллегии:

Беляев Борис Илларионович, д.ф.-м.н.

Борздов Владимир Михайлович, д.ф.-м.н.

Голенков Владимир Васильевич, д.т.н.

Гончаров Виктор Константинович, д.ф.-м.н.

Есман Александр Константинович, д.ф.-м.н.

Ильин Виктор Николаевич, д.т.н.

Кугейко Михаил Михайлович, д.ф.-м.н.

Кучинский Петр Васильевич, д.ф.-м.н.

Мулярич Степан Григорьевич, д.т.н.

Петровский Александр Александрович, д.т.н.

Попечиц Владимир Иванович, д.ф.-м.н.

Рудницкий Антон Сергеевич, д.ф.-м.н.

Отдел рекламы и распространения:

Антоневич Светлана Геннадьевна
тел./факс. +375 (17) 204-40-00
e-mail: asg@nsys.by

Учредитель:

ТЧУП «Белэлектронконтракт»
220015, Республика Беларусь,
г. Минск, пр. Пушкина, 29 Б,
тел./факс: +375 (17) 210-21-89,
+ 375 (17) 204-40-00
www.electronica.by

© Перепечатка материалов, опубликованных
в журнале «Электроника инфо», допускается
с разрешения редакции

За содержание рекламных материалов редакция
ответственности не несет

Подписной индекс в РБ:
00822 (индивидуальная),
008222 (ведомственная)

Цена свободная

Подготовка, печать:

299 экз. отпечатано
тип. ООО «Полиграф»
г. Минск, ул. Кнорина, 50/4-401А
Лицензия №02330/0494199 от 03.04.2009 г.
Подписано в печать 04.04.2013 г.
Заказ №

СЛАВНЫЕ ИМЕНА «ОТЕЦ» ИСКУССТВЕННОГО ИНТЕЛЛЕКТА.....	2
НЕМНОГО ИСТОРИИ ИСТОРИЯ РАЗВИТИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ЗА РУБЕЖОМ.....	4
КОНФЕРЕНЦИИ ОТЧЕТ О ПРОВЕДЕНИИ МЕЖДУНАРОДНОЙ НАУЧНО-ТЕХНИЧЕСКОЙ КОНФЕРЕНЦИИ OSTIS-2013.....	6
ВЫСТАВКИ «АВТОМАТИЗАЦИЯ. ЭЛЕКТРОНИКА-2013», «ЭЛЕКТРОТЕХ. СВЕТ-2013».....	11
НАШИ ИНТЕРВЬЮ ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ – ВНЕ СФЕРЫ ИНТЕРЕСОВ ГОСУДАРСТВА В. Карпов.....	13
ПАРАДОКС ИСКУССТВЕННОГО ИНТЕЛЛЕКТА А. Панов.....	16
НАУКА ПОЧТИ РАЗУМНЫЙ СУПЕРКОМПЬЮТЕР В. Калинин.....	17
ПРИНЦИПИАЛЬНЫЕ ОСНОВЫ ОТКРЫТОГО ПРОЕКТА, НАПРАВЛЕННОГО НА СОЗДАНИЕ ТЕХНОЛОГИИ КОМПЬЮТЕРНОГО ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ В. Голенков, Н. Гулякина.....	18
РЕЦЕНЗИРУЕМЫЕ СТАТЬИ	28-45
ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ ИНТЕЛЛЕКТУАЛЬНАЯ ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНАЯ СИСТЕМА НА ОСНОВЕ LABVIEW ДЛЯ ЛАБОРАТОРНЫХ ПРАКТИКУМОВ ПО РАДИОЭЛЕКТРОНИКЕ С. Здоровцев, И. Круглов, А. Петрович.....	28
КОМПОНЕНТ УПРАВЛЕНИЯ ДОСТУПОМ СИСТЕМЫ УПРАВЛЕНИЯ ДОКУМЕНТАМИ М. Стержанов.....	31
ГЕНЕРИРОВАНИЕ И ФУНКЦИОНАЛЬНО-СТРУКТУРНАЯ ОПТИМИЗАЦИЯ БАЗОВОГО КОМПЛЕКТА ТАБЛИЦ ДЛЯ МУЛЬТИПЛИКАТИВНОЙ МИМА-СХЕМЫ МОНТГОМЕРИ А. Коляда, А. Чернявский, Е. Шабинская.....	35
ПОСТРОЕНИЕ ОТКАЗОУСТОЙЧИВЫХ ПРОГРАММИРУЕМЫХ ЦИФРОВЫХ УСТРОЙСТВ НА ОСНОВЕ ПРОГРАММНЫХ И АППАРАТУРНО-ПРОГРАММНЫХ СРЕДСТВ САМОВОССТАНОВЛЕНИЯ Н. Коротаев, В. Попечиц.....	42
ТЕХНОЛОГИИ HUD: ПРОЕКЦИЯ БУДУЩЕГО С. Сысоева.....	46
СВЕТОТЕХНИКА ВКЛЮЧЕНИЕ СВЕТОДИОДНОГО ОСВЕЩЕНИЯ НИЗКОЙ МОЩНОСТИ: МЕТОД УПРАВЛЕНИЯ ПИТАНИЕМ E. Haile.....	58
ПРАЙС-ЛИСТ	64

СПИСОК РЕКЛАМОДАТЕЛЕЙ

«Алнар».....	64	«Тиком».....	41
«АльфаЛидер групп».....	64	«ФЭК».....	64
«БелПлата».....	45	«Чип электроникс».....	64
«Вектор-Систем».....	41	«Электроконтинент».....	63
«ГорнТрейд».....	45		
«Минский часовой завод».....	5		
«МПОВТ».....	34		
«Нанотех».....	5	Обложки, цветные вставки	
«Полдень Плюс».....	63	Microchip.....	II вст.
«Приборостроительная компания».....	64	«Альфасофт».....	III обл.
«Промтехсервисснаб».....	45	«ФЭК».....	I, II обл., I, вст.
«СветЛед решения».....	64	«Электроконтинент».....	II, IV обл.
		«Элтикон».....	IV обл.



«ОТЕЦ» ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Джон Маккарти (4 сентября 1927 – 24 октября 2011) – выдающийся американский информатик, автор термина «искусственный интеллект», изобретатель языка Лисп, основоположник функционального программирования, лауреат Премии Тьюринга за огромный вклад в область исследований искусственного интеллекта.

Джон Маккарти родился 4 сентября 1927 года в Бостоне, штат Массачусетс, в семье иммигрантов, Джона Патрика и Иды Глатт Маккарти. Во время Великой депрессии семья была вынуждена часто менять место жительства, пока, наконец, отец Маккарти не нашел работу в Лос-Анджелесе, Калифорния.

У Маккарти рано проявились склонности к математике, в юности он самостоятельно изучал ее по университетским учебникам, которые использовались в расположенном неподалеку Калифорнийском технологическом институте. В результате, поступив в университет, он смог сразу пропустить два первых года по математике. В 1948 году получил степень бакалавра математики в Калифорнийском технологическом институте и степень доктора философии по математике в Принстоне в 1951 году. После кратковременных должностей в Принстонском и Стэнфордском университетах, Дартмутском колледже и Массачусетском технологическом институте, в 1962 году он стал профессором в Стэнфорде, где и оставался до своего ухода на пенсию в конце 2000-го.

В мае 1997 года суперкомпьютер компании IBM – Deep Blue одержал победу над чемпионом мира Гарри Каспаровым в знаменитом шахматном матче-реванше. Эта победа стала триумфом искусственного интеллекта. Но Джон Маккарти (John McCarthy) – человек, который придумал термин «искусственный интеллект», и стал пионером в области исследований искусственного интеллекта – не так ее себе представлял.

Еще в середине 60-х годов, шахматы называли «дрозофила искусственного интеллекта». Дрозophilы –



это плодовые мушки. Их использовали биологи для раскрытия секретов генетики. Маккарти верил, что его последователи – исследователи искусственного интеллекта – продвигнутся далеко вперед с помощью этой аналогии. «Компьютерные шахматы развились так же, как это могло бы случиться с генетикой, если бы еще с 1910 года усилия ученых сосредоточились только на выведении быстрой дрозофилы», – написал Маккарти, комментируя победу Deep Blue. «У нас была бы наука, но в основном мы получили очень быстрых плодовых мушек».

Дафна Колер (Daphne Koller), профессор Стэнфордской лаборатории искусственного интеллекта, продолжающая нести факел, заженный ортодоксальным видением искусственного интеллекта Маккарти, считает, что это высказывание характеризует Джона Маккарти и его работу. «Слово, которое лучше всего характеризует Маккарти – «бескомпромиссный», – сказала она. «Он считал, что искусственный интеллект мог бы действительно воссоздать уровень интеллекта человека, и поэтому нам очень не нравится то, что происходит с искусственным интеллектом сегодня. В основном создаются различные полезные приложения, но нет упора на машинное обучение».

На научно-исследовательской конференции Dartmouth Summer Research Conference по искусственному интеллекту в 1956 году Маккарти не только внес термин в популярный

лексикон, но и основал совершенно новую область исследований, наряду с другими первопроходцами Марвином Мински (Marvin Minsky), Натаниэлем Рочестером (Nathaniel Rochester) и Клодом Шенноном (Claude Shannon). Вскоре, он изобретает LISP – один из самых влиятельных в мире языков программирования, который сыграет важную роль в развитии систем с разделением времени.

«Без разделения времени, не было бы современного Интернета», – говорит Лестер Еарнест (Lester Earnest). Он работал с Маккарти в Массачусетском технологическом институте (MIT) в конце пятидесятых годов, а затем в Стэнфордской лаборатории искусственного интеллекта (Stanford Artificial Intelligence Lab-SAIL). Эту лабораторию Маккарти помог основать в 1962 году.

«Но, несмотря на все оказанное влияние на современный мир, Маккарти преуспел нечто гораздо большее. В области искусственного интеллекта Маккарти был философом», – говорит Себастьян Трун, возродивший SAIL в 2003 году. В настоящее время Трун работает в компании Google, где он создал беспилотный автомобиль.

Лестер Еарнест впервые встретился с Маккарти в Массачусетском технологическом институте, в то время он работал над оборонной наземной системой автоматического наведения на цель (Semi-Automatic Ground Environment – SAGE). SAGE – первая компьютерная сеть, которая позволяла нескольким пользователям одновременно получать доступ к системе и, по словам Еарнеста, именно SAGE вдохновила Маккарти на работу над системой разделения времени. Он был первым, кому пришла идея создания системы разделения времени.

Научный труд Маккарти дал толчок развитию совместимой системы разделения времени (Compatible Time-Sharing System – CTSS), которая была продемонстрирована в Массачусетском технологическом институте в 1961 году. Над похожей системой разделения времени работала компания

Bolt, Beranek and Newman, которая, впоследствии, сыграла значительную роль в основании Интернета.

Но для Маккарти система разделения времени отступила на второй план по сравнению с искусственным интеллектом. В Массачусетском технологическом институте Маккарти совместно с другим первопроходцем искусственного интеллекта Марвином Минским (Marvin Minsky) создали «группу искусственного интеллекта» (AI group), позднее эта группа вошла в университетский проект Мас. Финансировало этот проект Агентство передовых оборонных исследовательских проектов (DARPA-Defense Advanced Research Projects Agency).

В Массачусетском технологическом институте Маккарти разработал язык LISP, ставший стандартным языком программирования сообщества искусственного интеллекта, а также проникнул в компьютерный мир в целом. Сегодня это второй, старейший, все еще используемый язык программирования высокого

уровня после Fortran. Но в отличие от Fortran, он по-прежнему является основой для новых языков.

В 1962 году Маккарти перешел в Стэнфорд, где основал лабораторию искусственного интеллекта – SAIL. Лестер Еарнест позже присоединился к лаборатории, и считает ее местом рождения многих современных технологий мира. «Начиная от робототехники, созданной в SAIL, до интерфейса пользователя», – приводит примеры Себастьян Трун (Sebastian Thrun), продолживший дело SAIL.

Еарнест говорит, что примером влияния лаборатории может служить Алан Кей, который работал в лаборатории, прежде чем перешел в научно-исследовательский центр в Пало-Альто (PARC), он изобрел объектно-ориентированный язык программирования Smalltalk. Основное влияние на создание Smalltalk оказал Маккарти и Lisp. Кей позднее назовет Lisp «величайшим языком программирования когда-либо разрабатываемым».

В 80–90-х годах лаборатория попала в немилость. «Искусственный интеллект пережил «зимний период», – рассказывает сотрудник Google, Себастьян Трун. «Из-за возникшего несоответствия между обещаниями и реальностью люди поняли, что мы не могли воссоздать человеческий интеллект». Но в 2003 году Трун возродил SAIL и в какой-то степени дух Маккарти.

Как и Дафна Коллер, Трун утверждает, что работа, которая сегодня ведется в области искусственного интеллекта, не всегда совпадает с замыслом Маккарти. «На мой взгляд, Google – это и есть искусственный интеллект», – говорит он. «Но Маккарти не был согласен с этим. Он частенько приходил ко мне, и критиковал меня». Несмотря на это, Себастьян Трун считает, что однажды задумка Маккарти осуществится. «Это наступит тогда, когда мы постигнем человеческий разум».

golos.am

НОВОСТИ

RAPYUTA: ИНТЕРНЕТ ДЛЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

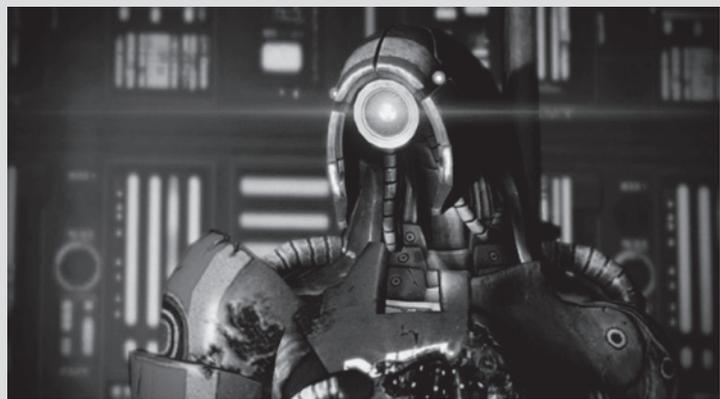
Для роботов в рамках проекта RoboEarth запустили «облако» – единый ресурс, на котором отдельные машины смогут производить нужные им трудоемкие вычисления.

Консорциум из шести европейских исследовательских учреждений запустил первую фазу своего проекта RoboEarth. Суть самого проекта его участники описывают как «интернет для роботов», а смысл заключается в том, чтобы позволить любым роботам получать доступ к общей базе данных, в которой содержится информация о программном обеспечении любого отдельно взятого устройства. Это позволит им в теории учиться друг у друга выполнению новых задач.

Первая фаза всего проекта – запуск «облака» под названием Rapyuta: The RoboEarth Cloud Engine, используя мощности которого роботы смогут выполнять нужные трудоемкие вычисления. Таким образом, отдельных роботов не потребуются оснащать собственными мощными компьютерами (а соответственно, производителям можно будет сэкономить на размерах и массе этих самых машин и больше внимания уделить их надежности).

В «облаке» каждому роботу будет отведен свой «аккаунт», область, в которую он сможет загружать нужные данные и производить вычисления, а затем скачивать данные обратно и действовать в соответствии с полученной информацией. Как отмечают создатели проекта, Rapyuta наиболее полезна для выполнения таких сложных задач, как составление карты окружающего пространства, навигация и распознавание голосовых команд.

apploidnews.com



ИСТОРИЯ РАЗВИТИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ЗА РУБЕЖОМ

Идея создания искусственного подобия человеческого разума для решения сложных задач и моделирования мыслительной способности витала в воздухе с древнейших времен. Впервые ее выразил Р. Луллий (ок. 1235–1315), который еще в 14 веке пытался создать машину для решения различных задач на основе всеобщей классификации понятий.

В 18 веке Г. Лейбниц (1646–1716) и Р. Декарт (1596–1650) независимо друг от друга развили эту идею, предложив универсальные языки классификации всех наук. Эти идеи легли в основу теоретических разработок в области создания искусственного интеллекта.

Развитие искусственного интеллекта, как научного направления, стало возможным только после создания ЭВМ. Это произошло в 40-х годах 20 века. В это же время Н. Винер (1894–1964) создал свои основополагающие работы по новой науке – кибернетике.

Термин искусственный интеллект (artificial intelligence) предложен в 1956 году на семинаре с аналогичным названием в Станфордском университете (США). Семинар был посвящен разработке логических, а не вычислительных задач. Вскоре после признания искусственного интеллекта самостоятельной отраслью науки произошло разделение на два основных направления: нейрокибернетику и кибернетику «черного ящика». И только в настоящее время стали заметны тенденции к объединению этих частей вновь в единое целое.

Основную идею нейрокибернетики можно сформулировать следующим образом. Единственный объект, способный мыслить, – это человеческий мозг. Поэтому любое «мыслящее» устройство должно каким-то образом воспроизводить его структуру.

Таким образом, нейрокибернетика ориентирована на аппаратное моделирование структур, подобных структуре мозга. Физиологами давно установлено, что основой человеческого мозга является большое количество (до 1021) связанных между собой и взаимодействующих нервных клеток – нейронов. Поэтому усилия нейрокибернетики были сосредоточены на создании элементов, аналогичных нейронам, и их объединении в функционирующие системы. Эти системы принято называть нейронными сетями, или нейросетями.

Первые нейросети были созданы в конце 50-х годов американскими учеными Г. Розенблаттом и У. Мак-Каллоком. Это были попытки создать системы, моделирующие человеческий глаз и его взаимодействие с мозгом. Устройство, созданное ими, получило название перцептрона. Оно умело различать буквы алфавита, но было чувствительно к их написанию, например, буквы А, А и А для этого устройства были тремя разными знаками. Постепенно в 70–80-е годы количество работ по этому направлению искусственного интеллекта стало снижаться. Слишком неутешительны оказались первые результаты. Авторы объясняли неудачи малой памятью и низким быстродействием существующих в то время компьютеров.

Однако в середине 80-х годов в Японии в рамках проекта разработки компьютера 5 поколения, основанного

на знаниях, был создан компьютер 6 поколения, или нейрокомпьютер. К этому времени ограничения по памяти и быстродействию были практически сняты. Появились транспьютеры – параллельные компьютеры с большим количеством процессоров. От транспьютеров был один шаг до нейрокомпьютеров, моделирующих структуру мозга человека. Основная область применения нейрокомпьютеров – распознавание образов.

В настоящее время используются три подхода к созданию нейросетей:

- аппаратный – создание специальных компьютеров, плат расширения, наборов микросхем, реализующих все необходимые алгоритмы;

- программный – создание программ и инструментариев, рассчитанных на высокопроизводительные компьютеры. Сети создаются в памяти компьютера, всю работу выполняют его собственные процессоры;

- гибридный – комбинация первых двух. Часть вычислений выполняют специальные платы расширения (сопроцессоры), часть – программные средства.

В основу кибернетики «черного ящика» лег принцип, противоположный нейрокибернетике. Не имеет значения, как устроено «мыслящее» устройство. Главное, чтобы на заданные входные воздействия оно реагировало так же, как человеческий мозг.

Это направление искусственного интеллекта было ориентировано на поиски алгоритмов решения интеллектуальных задач на существующих моделях компьютеров. В 1956–1963 годах велись интенсивные поиски моделей и алгоритма человеческого мышления и разработка первых программ. Оказалось, что ни одна из существующих наук – философия, психология, лингвистика – не может предложить такого алгоритма. Тогда кибернетики предложили создать собственные модели. Были созданы и опробованы различные подходы.

В конце 50-х годов родилась модель лабиринтного поиска. Этот подход представляет задачу как некоторый граф, отражающий пространств состояний, и в этом графе проводится поиск оптимального пути от входных данных к результирующим. Была проделана большая работа по разработке этой модели, но в решении практических задач идея большого распространения не получила.

Начало 60-х годов – эпоха эвристического программирования. Эвристика – правило, теоретически не обоснованное, но позволяющее сократить количество переборov в пространстве поиска. Эвристическое программирование – разработка стратегии действий на основе известных, заранее заданных эвристик.

В 1963–1970 годах к решению задач стали подключать методы математической логики. На основе метода резолюций, позволившего автоматически доказывать теоремы при наличии набора исходных аксиом, в 1973 году создается язык Пролог.

Обычно теорема формулируется так: при таких-то условиях выполняется то-то. Здесь происходит эквивалентная замена на доказательство следующего: найдем противоречие в утверждении, что условия теоремы истинны,

а результат ложен. Под множеством логических формул из предыдущего параграфа понимается множество всех условий теоремы + отрицание ее вывода.

При этом программа рассматривается как набор логических формул, описывающих предметную область, совместно с теоремой, которая должна быть доказана. Логическое программирование избавляет программиста от необходимости определения точной последовательности шагов выполнения вычислений.

Существенный прорыв в практических приложениях искусственного интеллекта произошел в середине 70-х годов, когда на смену поискам универсального алгоритма мышления пришла идея моделировать конкретные знания специалистов-экспертов. В США появились первые коммерческие системы, основанные на знаниях, или экспертные системы. Пришел новый подход к решению задач искусственного интеллекта – представление знаний. Созданы MYCIN и DENDRAL – ставшие уже классическими экспертные системы для медицины и химии. Объявлено несколько глобальных программ развития интеллектуальных технологий – ESPRIT (Европейский Союз), DARPA (Министерство обороны США), японский проект машин 5 поколения.

Начиная с середины 80-х годов происходит коммерциализация искусственного интеллекта. Растут ежегодные капиталовложения, создаются промышленные экспертные системы. Растет интерес к самообучающимся системам.

История развития искусственного интеллекта в России

В 1954 году в МГУ под руководством профессора А.А. Ляпунова (1911–1973) начал свою работу семинар «Автоматы и мышление». В этом семинаре принимали участие крупнейшие физиологи, лингвисты, психологи, математики. Принято считать, что именно в это время родился искусственный интеллект в России. Как и за рубежом, выделились направления нейрокибернетики и кибернетики «черного ящика».

Среди наиболее значимых результатов, полученных российскими учеными, следует отметить алгоритм «Кора» М.Бонгарда, моделирующий деятельность человеческого мозга при распознавании образов (60-е годы).

В 1945–1964 годах создаются отдельные программы и исследуется поиск решения логических задач. В Ленинграде (ЛОМИ – Ленинградское отделение математического института им. В.А. Стеклова) создается программа, автоматически доказывающая теоремы (АЛПЕВ ЛОМИ). Она основана на оригинальном обратном выводе С.Ю. Маслова, аналогичном методу резолюций Робинсона.

В 1965–1980 годы получает развитие новая наука – ситуационное управление (соответствует представлению знаний в западной терминологии). Основоположник этой научной школы – профессор Д.А. Поспелов. Разработаны специальные модели представления ситуаций – представления знаний.

В 1980–1990 годы проводятся активные исследования в области представления знаний, разрабатываются языки представления знаний, экспертные системы (более 300). В Московском государственном университете создается язык РЕФАЛ.

В 1988 году создается АИИ – Ассоциация искусственного интеллекта. Ее членами являются более 300 исследователей. Президент Ассоциации – Д.А. Поспелов. Крупнейшие

центры – в Москве, Петербурге, Переславле-Залесском, Новосибирске.

В рамках Ассоциации проводится большое количество исследований, собираются конференции, издается журнал. Уровень теоретических исследований по искусственному интеллекту в России ничуть не ниже мирового. К сожалению, начиная с 1975 года на развитии этого направления сказалось прогрессирующее отставание в технологии. На данный момент отставание в области промышленных интеллектуальных систем составляет порядка 5–7 лет.

treeofknowledge.narod.ru

НАНОТЕХ pcb@pcb.by www.pcb.by
г. Минск, ул. Седых 12А, пом. 2Н

- Печатные платы**
 (одно-, двухсторонние, многослойные, на алюминиевой подложке)
- Трафареты для пасты**
 (лазерной резкой из нержавеющей стали)
- Паяльные пасты**
 (безотмывочные, канифольные, водосмываемые, и др.)

тел: +375 17 237 29 34
 тел: +375 17 237 29 35
 тел/факс: +375 17 237 29 36
 тел/факс: +375 17 281 35 36
 тел. моб: +375 29 101 35 36
 тел. моб: +375 29 876 35 36

ДПП Качество и компетентность в мире печатных плат

ОАО «Минский часовой завод»

ВАШ НАДЕЖНЫЙ ИЗГОТОВИТЕЛЬ ПЕЧАТНЫХ ПЛАТ

220095, г. Минск, пр. Независимости, 95, т./ф. +375 (17) 280-49-55, моб. +375 (29) 750-45-50, bogdashich@mail.ru

Типы плат

- ДПП, МПП (до 24 слоев) любого класса точности
- Гибкие печатные платы
- Платы для ВЧ/СВЧ
- Платы на алюминиевой подложке
- Платы для смарт-карт

Возможности

- Проектирование плат
- Технологическая поддержка
- Покрытия: HASL, иммерсионное золото, иммерсионное олово, ПОС, Ni-B
- Формирование контура любой формы
- Материалы: FR-4, Rogers, Duroid, алюминий, лавсан

Качество

- Сертификат соответствия ВУ/112 05.01.0030030

АКЦИЯ -10% в апреле

Срок изготовления от 2 дней до двух недель

ОТЧЕТ О ПРОВЕДЕНИИ МЕЖДУНАРОДНОЙ НАУЧНО-ТЕХНИЧЕСКОЙ КОНФЕРЕНЦИИ OSTIS-2013

(OPEN SEMANTIC TECHNOLOGY FOR INTELLIGENT SYSTEMS – ОТКРЫТЫЕ СЕМАНТИЧЕСКИЕ ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ)

1–23 февраля 2013 года в Белорусском государственном университете информатики и радиоэлектроники прошла III-я Международная научно-техническая конференция OSTIS-2013, которая была проведена на базе кафедры интеллектуальных информационных технологий БГУИР.

Основной целью ежегодных конференций OSTIS (Open Semantic Technology for Intelligent Systems) является создание условий для расширения сотрудничества различных научных школ, вузов и коммерческих организаций, направленного на разработку и применения комплексной массовой и постоянно совершенствуемой технологии компонентного проектирования интеллектуальных систем.

Конференцию OSTIS 2013-го года организаторы посвятили 80-летию Дмитрия Александровича Поспелова – первого Президента Российской ассоциации искусственного интеллекта.

Соорганизаторы конференции OSTIS-2013:

- Белорусский государственный университет информатики и радиоэлектроники;
- Российская ассоциация искусственного интеллекта;
- Государственное учреждение «Администрация Парка высоких технологий» (Республика Беларусь);
- Объединенный институт проблем информатики Национальной академии наук Беларуси;
- Тверской государственный технический университет;
- Научно-исследовательский институт «Прикладная семиотика» АН РТ;
- Институт информатизации образования Российской академии образования;
- Международная ИТ-компания «Itransition»;
- Компания «Melesta»;
- Международная сервисная ИТ-компания «Exigen® Services»;
- Компания «Qulix Systems»;
- Digital-агентство «ARTOX media»;
- Компания «Речевые Технологии»;
- Компания «ВирусБлокАда»;
- Компания «Сакрамент»;
- Компания «htc»;
- Компания «ИксБи Софтваре»;
- Частное унитарное предприятие «Лаборатория стартапов»;
- Компания «Адмирал Софт»;
- Компания «Antalika»;
- Компания «Эксадел»;
- Компания «СофтКлуб»;
- Компания «СофтТеко».

Соорганизаторы, оказавшие информационную и техническую поддержку конференции OSTIS-2013:

- Международный журнал «Программные продукты и системы»;
- Научный журнал «Информатика»;
- Научно-практический журнал для специалистов «Электроника инфо»;
- Научно-практический журнал «Речевые технологии»;
- Научный журнал «Онтология проектирования»;
- Открытый проект «MyGui»;
- Открытый проект «Ogre».

Работа конференции OSTIS-2013 проводилась по следующим направлениям:

1. Принципы построения и структура технологии компонентного проектирования интеллектуальных систем;
2. Базовые семантические модели интеллектуальных систем и их реализация;
3. Семантические модели, средства и методы компонентного проектирования баз знаний и пакетов программ, ориентированных на обработку знаний;
4. Семантические модели, средства и методы компонентного проектирования решателей задач;
5. Семантические модели, средства и методы компонентного проектирования пользовательских интерфейсов интеллектуальных систем;
6. Комплексные методы, онтология и менеджмент компонентного проектирования интеллектуальных систем;
7. Логико-семантические модели прикладных интеллектуальных систем.

По результатам рецензирования, представленных на конференцию материалов, были сформированы и опубликованы Сборник материалов конференции, в который вошли 92 статьи. Реально было заслушано 58 докладов. Среди них: 9 докладов профессоров, примерно по 17 докладов доцентов и докторов наук, 35 докладов кандидатов наук, а также 31 доклад молодых ученых.

Участники конференции могли ознакомиться с текстами докладов до начала конференции на официальном сайте конференции <http://conf.ostis.net>.

В конференции OSTIS-2013 приняли участие более 70 человек. Среди них 17 профессоров, 24 доктора и доцента наук, 18 кандидатов наук. А также студенты, магистранты и аспиранты кафедры ИИТ и с других кафедр БГУИР.

География участников конференции OSTIS-2013 весьма обширна и охватывает 18 городов России, Беларуси, Украины и Казахстана: Москва, Минск, Киев, Астана, Санкт-Петербург, Владивосток, Новосибирск, Иркутск, Волгоград, Казань, Самара, Ульяновск, Тверь, Пермь, Апатиты, Брест, Гродно, Кременчуг.

В рамках конференции OSTIS-2013 Программным комитетом конференции был проведен конкурс докладов молодых ученых. В конкурсе участвовал 21 доклад. Наивысшую оценку жюри получили следующие доклады:

- «Реализация хранилища унифицированных семантических сетей» Корончика Д.Н., кафедра ИИТ, БГУИР;
- «Технологии связывания данных в пространстве открытых данных на примере математической коллекции» Кирилловича А.В., Казанский федеральный университет;
- «Проблемы синонимов в тезаурусе интеллектуального помощника проектанта» Одиной С.А. и Княжиной Ю.Е., Самарский государственный аэрокосмический университет имени академика С.П. Королева (Национальный исследовательский университет).

В рамках конференции OSTIS-2013 был также проведен конкурс студенческих проектов интеллектуальных систем. В финале конкурса было отобрано 6 проектов. В жюри конкурса вошли члены Программного комитета конференции и представители различных фирм, являющиеся резидентами Парка высоких технологий. Высшую оценку жюри получили следующие проекты:

- Ахрамович А., Валенда С. «Адаптивный анализ данных социальных интерактивных ресурсов на основе OLAP-технологии», Белорусский государственный университет;
- Буров А., Тарасюк А. «Интеллектуальная справочная система по геометрии», Белорусский государственный университет информатики и радиоэлектроники.



Регистрация участников конференции OSTIS-2013



Торжественное открытие конференции. Зал заседаний полностью заполнен зрителями. Примечательно, что значительную часть зрителей традиционно составляют студенты

На открытии конференции OSTIS-2013 выступили:

- заведующий кафедрой интеллектуальных информационных технологий БГУИР – Голенков Владимир Васильевич;
- проректор по учебной работе и менеджменту качества – Живицкая Елена Николаевна;
- заместитель директора государственного учреждения «Администрация Парка высоких технологий» – Концевая Инесса Владимировна;

- генеральный директор Объединенного института проблем информатики НАН Беларуси – Тузиков Александр Васильевич;
- академик НАН Беларуси, доктор технических наук, профессор – Чернявский Александр Федорович;
- председатель совета РАИИ Кузнецов Олег Петрович;
- вице-президент Академии наук Республики Татарстан – Сулейманов Джавдет Шевкетович.





Все они отметили важность конференции, как инструмента популяризации идей искусственного интеллекта, обмена идеями, налаживания межнациональных научных и коммерческих связей, воспитания подрастающего поколения ученых и инженеров.

Заключительным этапом конференции стал круглый стол. Профессор Хорошевский В.Ф. заставил еще раз подумать присутствующих мэтров и молодежь над зна-

чением слова «онтология» и над уместностью его употребления в различных контекстах.



Обсуждение на круглом столе поддержали такие представители мэтров, как Кузнецов О.П., Смирнов С.В., Со-

снин П.И., Невзорова О.А., Лукашевич Н.В. Подытожил все выше сказанное профессор Голенков В.В.

«АВТОМАТИЗАЦИЯ. ЭЛЕКТРОНИКА-2013»

16-Я МЕЖДУНАРОДНАЯ СПЕЦИАЛИЗИРОВАННАЯ ВЫСТАВКА.

«ЭЛЕКТРОТЕХ. СВЕТ-2013»

13-Я МЕЖДУНАРОДНАЯ СПЕЦИАЛИЗИРОВАННАЯ ВЫСТАВКА

С 12 по 15 марта 2013 года в Минске при поддержке Министерства промышленности Республики Беларусь, Белорусской Ассоциации промышленных энергетиков прошли международные специализированные выставки «Автоматизация. Электроника-2013» и «Электротех. Свет-2013».

Выставка «Автоматизация. Электроника» – одна из крупнейших в Беларуси специализированных выставок отечественных и мировых производителей, поставщиков средств автоматизации и электронных компонентов, технологического оборудования и материалов для электронной и электротехнической промышленности. Сегодня ее органично дополняет вторая выставка – «Электротех. Свет».

В павильоне на ул. Даумана, 23 была представлена обширная экспозиция отечественных и мировых производителей, поставщиков средств автоматизации и электронных компонентов, технологического оборудования и материалов для производства электронной и электротехнической промышленности.

Более 100 компаний из Беларуси, России, Украины, Польши, Германии, Литвы, Швейцарии стали участниками выставки в 2013 году.

Новые разработки можно было увидеть на стендах ведущих производителей и поставщиков оборудования и материалов: «Энергопро», «Белроспром», Европейская электротехническая компания, «Логопром», «Локальные системы», «Элос», «Вектор технологий», «Системный анализ С», «Энергокомплект», «Европрибор», «Авком-трейд», «Белэнергокип», «Холтика Электро», «Атава», «БелПлата», «Вольна».

Среди зарубежных представителей – JUMO GmbH & Co. KG (Германия), ENDRESS+HAUSER INSTRUMENTS INTERNATIONAL AG (Швейцария), С.Н. Erbsloh Polska sp.Z o.o (Польша), Samson AG (Германия), «ТД ВЭЛАН»,

Курский электроаппаратный завод (Россия), НПО «Эр-кон» (Россия).

Успех форума подтверждается решениями экспонентов: более 60 % из них являются постоянными участниками выставок.

Выставка имеет высокий рейтинг среди руководителей и технических специалистов электротехнической промышленности и энергетики. Она ориентирована на широкий круг специалистов, которые принимают решение об использовании современных электронных компонентов, обеспечивают снабжение производства, используют их в новых разработках. Ежегодно выставку посещает более 6000 человек.

На весеннем форуме электронщиков и электротехников проводились презентации, конференции, семинары и обучающие программы, которые обогатили тематику выставок «Автоматизация. Электроника» и «Электротех. Свет» и, безусловно, позволили с научной точки зрения рассмотреть и проанализировать процессы, происходящие в отрасли.

Ключевым мероприятием деловой программы выставки стал Международный научно-практический семинар «Автоматизация технологических процессов и производств», организаторами которого выступили Министерство образования Республики Беларусь, Национальная академия наук Беларуси, Белорусский национальный технический университет.

Информационную поддержку выставке оказывали более 30 специализированных изданий и интернет-сайтов.

Выставки «Автоматизация. Электроника», «Электротех. Свет» – идеальная площадка для продвижения продукции и брендов, изучения рынка, встреч со специалистами и потенциальными заказчиками из разных регионов Беларуси и зарубежья.





ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ – ВНЕ СФЕРЫ ИНТЕРЕСОВ ГОСУДАРСТВА

Карпов Валерий Эдуардович, руководитель лаборатории робототехники кафедры «Вычислительные системы и сети» МИЭМ НИУ ВШЭ, кандидат технических наук, доцент, вице-президент Российской ассоциации искусственного интеллекта

– Валерий Эдуардович, наверное, у каждого мальчишки был игрушечный робот, а уж сколько фантастики на эту тему было перечитано! Тогда казалось – вот еще немного – и нашу жизнь заполнят те самые разумные машины. И вот мальчишки стали взрослыми, солидными дядями, а роботов на наших улицах все нет и нет. Что же происходит сегодня в области искусственного интеллекта и робототехники?

– Разговор о современном состоянии исследований в области искусственного интеллекта хотелось бы начать с небольшого экскурса в историю. Когда в 40-х, начале 50-х годов ученые стали задумываться о создании искусственного разума, или разумной машины, то казалось, что все предпосылки созрели – к тому времени уже были определенные достижения в нейробиологии, стали появляться вычислительные машины, развиваться кибернетика – казалось уже не за горами тот момент, когда такая машина появится на свет.

Но потом наступил переломный момент, переосмысление задач и, когда ученые перестали говорить о «разумной машине» или «искусственном разуме», произошло разделение научного сообщества на тех, кто впоследствии стал называться робототехниками и тех, кого проблемы «железа» не волновали, а куда больше интересовали проблемы, связанные с информатикой. Чуть позже, когда выяснилось, что непонятно не только, как строить «разумную машину», но и что такое «искусственный интеллект», произошло дробление на более или менее узкие направления уже в области самого искусственного интеллекта.

Поэтому сегодня, говоря «специалист по искусственному интеллекту», тут же потребуются уточнение, чем конкретно он занимается – компьютерной лингвистикой, генетическими алгоритмами, интеллектуальными системами управления и так далее. Такая сегрегация – показатель кризиса в исследуемой области, иначе говоря, – за последние 30 лет никаких качественно новых идей так и не появилось – в лучшем случае происходит перетасовка, переделка старых механизмов, методов, теорий. Возможно я могу показаться излишне категоричным, но в целом ситуация примерно так и выглядит.

– Так что же сегодня следует понимать под термином «искусственный интеллект»?

– Сейчас под искусственным интеллектом (ИИ) понимают область информатики, которая занимается весьма специфическими задачами. А специфика состоит в том,



что мы либо занимаемся решением задач, эффективный алгоритм решения которых, кроме метода перебора, нам неизвестен, и тогда искусственный интеллект пытается придумать какие-то новые методы, механизмы, что бы хоть как-то этот тяжелый перебор ограничить. Второе, чем занимается ИИ – это решение таких задач, условия которых мы пока не можем точно формализовать, например, задачи, связанные с разумным поведением или, точнее, поведением, которое кажется нам разумным; задачи, связанные с творческим процессом, где сложно сформулировать, о чем конкретно идет речь и т.д.

– А в чем основная проблема, почему невозможно сформулировать условия задачи?

– Вопрос можно поставить и так: «А какое поведение можно назвать разумным или интеллектуальным»? Второе, кстати, несколько проще для формализации. Были попытки создания единых определений, единых понятий, создавались и создаются справочники-тезаурусы, но, к сожалению, все эти дефиниции обладают определенным изъяном.

Строго говоря, каждое определение само по себе противоречиво, да и задача определения состоит не в том, чтобы четко описать понятие, а чтобы отделить одно понятие от другого. Например, мы вводим определение, согласно которому вот эта машина – интеллектуальная, а вот этот пылесос – нет. Вот это – робот, а вот то – просто автомат. До сих пор нет внятных определений, и до сих пор ведутся споры о том, что является предметом ИИ.

Скажем, на последней конференции по ИИ, которая состоялась прошлой осенью в Белгороде, был организован специальный круглый стол, чтобы попытаться понять, что такое ИИ. Представляете, 50 лет искусственному интеллекту, а мы все сидим и спорим, что же является предметом ИИ! В ближайшее время состоится семинар по проблемам ИИ, и опять будет подниматься тот же вопрос – что является предметом искусственного интеллекта?

Показательный тому пример – не так давно я попросил одного известного специалиста по ИИ выступить на семинаре с докладом, посвященном методам интеллектуального планирования. Мне и самому интересна эта тема, более того, хотел пригласить своих студентов и аспирантов, чтобы они из первых уст послушали о том, что сделано, какие новые теории появились. А в ответ услышал весьма обескураживающее заявление: «о старом добром STRIPS (Stanford Research Institute Problem Solver) 60-х годов, конечно, я расскажу, но ничего принципиально прорывного в этой области, увы, нет!»

– Получается, ИИ – едва ли не тупиковая ветвь научной мысли, если за такой солидный отрезок времени не происходит движения вперед?

– Нет, конечно! Есть вполне определенный круг задач, которые надо решать – надо строить интерфейсы «человек-машина», надо уметь распознавать, надо уметь планировать поведение тех или иных технических объектов, надо принимать решения. Задачи есть и будут, и исторически сложилось так, что специалисты, умеющие решать подобного рода задачи, именуют себя специалистами в области ИИ. Существует, правда, и обратная тенденция – когда некоторые направления в науке, изначально считавшиеся «искусственно-интеллектуальными», со временем отпочковываются и перестают быть предметом ИИ. Мы чудесным образом умеем работать со знаниями, представлять их, обрабатывать, накапливать и, в определенном смысле, получать их автоматически. Инженерия знаний, которая была одним из центральных направлений в ИИ, сейчас спокойно существует как отдельное направление – и это правильно! Когда мы научимся решать задачи моделирования искусственной жизни – эта область тоже выйдет из ИИ, и там останутся только те задачи, которые мы не умеем решать эффективно. Мне кажется, это вполне естественный процесс, никакого тупика я не вижу.

– С развитием робототехники подобная ситуация?

– Если вспомнить историю, то увидим, что первое разделение между специалистами, которые изначально хотели строить разумные машины, произошло тогда, когда одни сказали – «мы будем решать вполне определенные задачи – создавать манипуляторы, транспортные системы и прочее», а другие занялись проблемами информатики. Долгое время оба направления прекрасно развивались, практически не пересекаясь между собой. Но в последнее время, эти области стали активно сближаться, и в этом я нахожу определенную заслугу специалистов по ИИ – робототехники стали понимать, что без умения решать интеллектуальные задачи, никакого прорыва не будет. Также выяснилось, что многое из того, что ранее относили к абстрактной информатике – все эти методы и модели, прекрасно работающие «на бумаге» или на компьютере, не работают на реальном «железе», здесь надо знать определенную специфику. В том, что эти два направления стали активно сближаться, не последнюю роль сыграла Российская ассоциация искусственного интеллекта (РАИИ) – и сейчас на всех наших конференциях постоянно работают секции робототехники и, наоборот, на робототехнических конференциях присутствуют секции, посвященные интеллектуальным системам управления.

– А можно немного подробнее о работе Ассоциации искусственного интеллекта России?

– Это профессиональное сообщество, объединяющее видных специалистов данной отрасли, ведет свою историю от Советской ассоциации искусственного интеллекта. Отметим, что в ее работе всегда участвовали специалисты смежных областей – биологи, медики, лингвисты. Мы проводим конференции, различного рода семинары, издается журнал, словом, стараемся, если не определять направление развития искусственного интеллекта, то хотя бы влиять на это. Это очень важно, поскольку ИИ

из сферы интересов государства в последние годы выпал напрочь – даже по формальным признакам. Например, существует перечень критических или особо важных технологий, которые интересуют государство, и он с определенной частотой публикуется. Так вот, вначале из него выпал ИИ, а потом и робототехника.

– Это как-то связано с отсутствием отечественной элементной базы для создания роботов?

– Отсутствие своей элементной базы – это еще полбеды, по-хорошему, все равно, где покупать эти элементы. Но вот когда государство не задумывается о том, что без этого никуда, становится грустно.

Вдумайтесь только, СССР до своего распада занимал второе место в мире по количеству роботов после США. Япония, к слову, далеко не самая роботизированная страна, чтобы там не говорили. И первое место по номенклатуре выполняемых операций! И буквально за два-три года количество промышленных роботов в нашей стране сократилось до нуля! А ведь следует понимать, что промышленный потенциал любой страны определяет роботизированное производство и оно должно быть выгодным для государства. Сейчас этого нет по вполне понятным экономическим причинам – дорого, нет необходимой инфраструктуры и т.д. Наличие таких роботизированных линий предполагает поставку на них деталей вполне определенного качества. В итоге, в 90-е годы роботы были сданы на слом, а производственные площади были сданы в аренду под коммерцию.

И догнать развитые страны – чертовски тяжело, на мой взгляд, наша робототехническая школа сегодня практически исчерпала себя. Нет и элементной базы, и, самое печальное, ушли старые специалисты, а у молодежи нет достаточной мотивации учиться строить роботы. Опыт общения с западными робототехниками на всевозможных выставках только усиливает понимание того, что мы – страна даже не третьего, а уже тридцать третьего мира...

– Грустно, бесспорно! Но может, не все еще погибло – ведь проводятся какие-то олимпиады по робототехнике, наши ребята ездят за границу, соревнуются со сверстниками? Ведь есть что-то, чем не стыдно поделиться?

– Когда говоришь с кем-то о молодежной робототехнике, обычно у собеседника самая первая ассоциация – LEGO. То, что современная школьная и студенческая робототехника скатилась до LEGO – очень печально и показательно. Я всегда себя относил к ярким противникам леги-роботов, на мой взгляд, это была большая, продуманная диверсия. Не касаясь экономической стороны вопроса, скажу, что такие наборы напрочь отбивают всяческое желание творить самостоятельно. Как правило, такие леги-роботы – первый и последний шаг молодого человека в робототехнике. Настоящий классический инженер-робототехник, обучаясь, должен пощупать деталь, понять ее назначение.

Наши студенты проходят, так сказать, школу молодого бойца, изготавливают печатные платы, работают с паяльниками, осциллографами, занимаются настоящим предметным программированием – во всяком случае,

нам есть что показать и на мировом уровне. Бог с ней, со спортивной робототехникой – через это проходят все ребята – наши студенты конструируют квадрокоптеры, изучают коллективное поведение роботов – у нас есть небольшой коллектив роботов, которые реализуют «стайное» поведение, это и шагающие адаптивно, по-умному, роботы. Мы можем похвалиться и исследованиями в области глубокого погружения в контур управления – это когда вы сидите за компьютером, в виртуальных очках, обвешанные кучей исполнительных механизмов и начинаете чувствовать то, что воспринимают датчики робота – и реально управляете техническим объектом. Это то, что называется эффектом присутствия – когда вы и робот

становитесь единым целым. Добавим сюда исследования, связанные с распознаванием, более абстрактные, меньше связанные с «железом». Самое светлое пятно – это молодые ребята, которые занимаются всем этим, понимая, что много денег они на этом не заработают, и послаблений в учебе за работу в лаборатории не последует. Мы живем насыщенной разнообразной жизнью – в нашей лаборатории работают не только студенты МИЭМа, к нам приходят ребята из Бауманки, из Физтеха, МГУПИ и МИФИ.

– Спасибо!

Материал опубликован с согласия
В.Э. Карпова

НОВОСТИ

GOOGLE НАЧАЛ РАЗРАБОТКУ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Корпорация Google намерена создать систему искусственного интеллекта, которая станет основой нескольких служб поискового гиганта, в том числе и основной поисковой системы компании. Об этом в интервью Singularity Hub сообщил известный ученый-футуролог Рэймонд Курцвейл, который около месяца назад объявил о своем согласии на предложение компании Google присоединиться к команде разработчиков систем искусственного интеллекта.

Курцвейл заявил, что будущая система искусственного интеллекта превзойдет по возможностям систему искусственного интеллекта суперкомпьютера Watson компании IBM, которая считается самой совершенной системой на сегодняшний день.

«Проект, который я собираюсь реализовать в компании Google, сосредоточен на восприятии и понимании естественного языка. У такой системы может быть множество применений, но, в первую очередь, мы хотим дать компьютерам способность понять и оценить то, что они читают. Основной конечной целью проекта является применение системы искусственного интеллекта в ядрах основных служб Google, таких как поиск, ответы на вопросы и определение личности каждого пользователя системы» – заявил Курцвейл.

С одной стороны, проект Курцвейла кажется захватывающим, поскольку он проливает больше света на то, каким образом «компьютер будет следить за вами, вашими беседами, перепиской по электронной почте и за тем,

какие сайты вы посещаете и что вы там читаете». То, что здесь описано, будет являться некоей экосистемой, через которую у Google будет иметься доступ к информации о каждом аспекте вашей жизни.

В интервью Рэй Курцвейл немного описывает, как будет работать механизм их системы искусственного интеллекта.

«Система будет знать на семантически глубоком уровне все то, чем вы интересуетесь, а не только основные темы ваших интересов. Я предполагаю, что через несколько лет, система сможет дать пользователю ответы на большинство вопросов раньше, чем эти вопросы будут заданы, ведь система уже будет досконально знать все то, что интересует вас, и что вы хотите увидеть», – сказал Курцвейл.

Напомним, японский Национальный институт информации (NII) разрабатывает систему искусственного интеллекта, называющуюся *Todai Robot*. Целью программы является совершенствование искусственного интеллекта автоматизированной системы до уровня, который позволил бы успешно пройти вступительные испытания в ведущий ВУЗ страны; этого планируют достичь к 2021 году. Компания Fujitsu планирует использовать полученный в рамках участия в данной программе опыт для создания систем искусственного интеллекта, интегрируемых с информационными технологиями будущего.

rbc.ua

СУПЕРКОМПЬЮТЕР IBM WATSON: ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ «ГОЛУБОГО ГИГАНТА»

Компания IBM официально объявила о передаче Политехническому институту Ренсселера (Rensselaer Polytechnic Institute, RPI) модификации суперкомпьютера Watson с системой искусственного интеллекта.

Главная особенность суперкомпьютера Watson – это понимание вопросов, сформулированных на естественном языке, и быстрый поиск обоснованных ответов на

них. Компании IBM и RPI преследуют две цели: обучить Watson новым словам и заставить его работать с неструктурированной информацией (поток изображений, видеозаписей и сообщений электронной почты). Надеемся, что Watson даст ученым RPI ответы на множество волнующих человечество вопросов.

Gizmag.com

ПАРАДОКС ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Если почитать фантастические романы (и даже научно-популярные статьи), написанные лет так 40 назад, в частности, и в Советском Союзе, то одно из важных достижений будущего, которых касались писатели, – это искусственный интеллект.

Вот прошли десятки лет, а воз и ныне там. Искусственный интеллект, если его понимать, как самостоятельную мыслящую систему, все еще не создан. Случайность или закономерный ход событий? Предлагаем вашему вниманию мнение одного из участников конгресса Global Future 2045, старшего научного сотрудника НИИЯФ МГУ Александра Панова, высказанное им во время лекции на конгрессе.



– Мое мнение состоит в том, что надежды на искусственный интеллект очень сильно переоценены.

В основном надежды на возникновение искусственного интеллекта, более сильного, чем человеческий, связаны ... с законом Мура, ростом мощности аппаратной базы (т.е., имеется в виду, что в какой-то момент мощность аппаратной базы достигнет мощности мозга). У меня возникают два вопроса к этой кривой. Первый вопрос: как оценивалась мощность мозга, скорость, с которой работает мозг? Она оценивалась на количестве нейронов, количестве синапсов и частоте срабатывания синапсов. Это очень примитивная оценка. Вот возьмем амебу. У нее вообще нет нервной системы, а она обладает, тем не менее, очень сложным поведением. И все эти информационные аспекты, которые есть у амёбы, должны присутствовать и в мозге. Поэтому, скорее всего, скорость работы мозга очень сильно недооцененная величина, может быть, на десяток порядков (если возможности амёбы оценивать вышеупомянутым образом, через нейроны и синапсы, то они должны быть нулевыми, а это не так. Поэтому, вероятно, и для мозга все оценки очень занижены). И, второе, самое главное: закон Мура и рост мощности аппаратной базы дают только необходимые, но не достаточные условия возникновения мощного искусственного интеллекта.

Я просто прочитаю одну цитату: «За прошедшие 15 лет разум наших электронных вычислительных машин улучшился в миллионы раз. В течение нескольких десятилетий следует ожидать увеличения характеристик разума машин по крайней мере в несколько десятков тысяч раз. Разум таких машин по основным параметрам будет заведомо превосходить разум человека». Это было написано в 1975 году, и такие слова говорили непрерывно на протяжении развития компьютерных технологий. И говорят сейчас в той же самой форме и теми

же словами. С чем связана эта ошибка? Она связана с тем, что необходимое программное обеспечение гораздо более консервативно, на много порядков величин, чем развитие аппаратной базы.

Есть несколько примеров такого консерватизма, я на этом останавливаться не буду. Хочу только отметить, что те технологии, которые сейчас используются при программировании искусственного интеллекта – нейронные

сети, эвристическое программирование, экспертные системы, эволюционное программирование, – были изобретены более пятидесяти лет назад и с тех пор не было придумано ничего принципиально нового. В области программирования искусственного интеллекта наблюдается глубочайший застой. И, видимо, дело в том, что мы просто не понимаем, как на самом деле думает человек.

Непростое сознание

Какие проблемы мешают создать искусственный интеллект на основе компьютера? По мнению Александра Панова, компьютер обрабатывает информацию, человек обрабатывает смыслы. По умолчанию предполагается, что смыслы могут быть выражены информационным образом, но никто никогда не доказал, что это так. Простой пример: если смысл представляется квантовыми состояниями, то это не есть информация. Потому что важнейшим свойством информации является то, что она может быть скопирована, сдублирована. А квантовые состояния, согласно соответствующей теореме о клонировании состояний, скопировать невозможно.

Это лишь один пример, но на самом деле ситуация может быть гораздо хуже. Существует теорема Роджера Пенроуза, она состоит в следующем: существует очень много активностей человеческого мозга, которые не могут быть представлены активностью компьютера, но очень трудно доказать, что это действительно невозможно.

Однако есть исключения. Можно доказать строгую теорему, что любой конечный автомат, основанный на любых известных в настоящее время физических принципах, не может представить некоторые формы математической активности мозга (эта теорема – родственница первой теоремы Геделя о неполноте). Из теоремы следует: чтобы создать искусственный интеллект, превосходящий по мощности человеческий, нужно открыть неизвестные физические принципы. Предсказать же, когда они будут открыты, не представляется возможным.

festivalnauki.ru

ПОЧТИ РАЗУМНЫЙ СУПЕРКОМПЬЮТЕР

Владимир Калинин,

Юго-Западный государственный университет, г. Курск

Искусственный интеллект все еще фантастика. Однако год назад появилась система, которая справляется с обработкой конструкций человеческого языка лучше, чем человеческий мозг. Это еще не мышление, но уже кое-что.

Где-то далеко в Америке живет суперкомпьютер, которого зовут IBM Watson. Он совсем не отличается по своему «железу» от других суперкомпьютеров. Возникает закономерный вопрос, почему же его считают своего рода гением среди них?

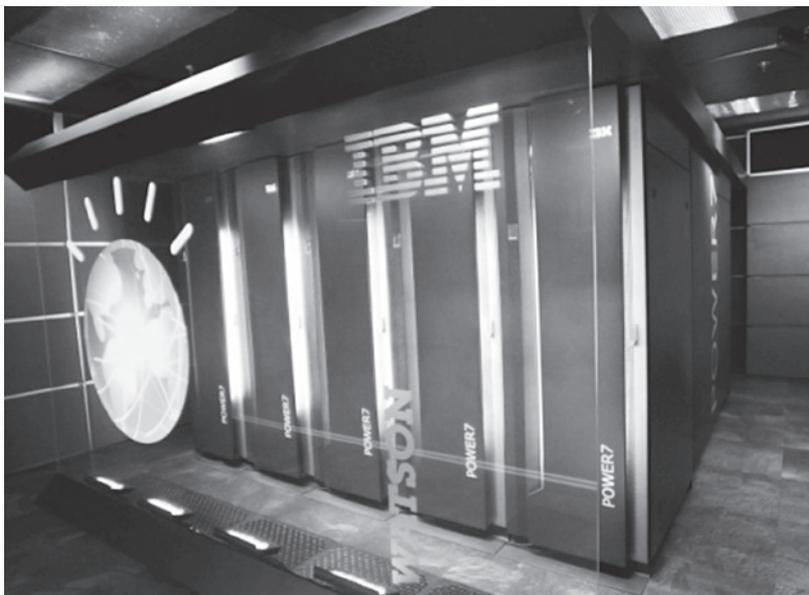
Watson – это удивительный самообучающийся механизм, который имеет уникальные возможности по обработке естественных языков, чем может похвастаться далеко не каждый его собрат. «Этот парень» умеет распознавать смысл предложений и различных вопросов, а также отлично конструирует ответы на них, используя усвоенные объемы данных.

Watson занимает целую комнату, 10 серверных шкафов с 90 серверами IBM Power 750 – размер, поражающий воображение! Первоначально было заявлено, что он имеет 2800 процессорных ядер и 15 терабайт памяти. Клиентские устройства размером со среднестатистический ноутбук устанавливают в помещениях, чтобы они связывались с Watson посредством Интернета. Для того, чтобы обработать вопрос, Watson сначала разбивает вопрос на более простые составляющие, затем ищет ответ на него и синтезирует умозаключение.

Watson успел прославиться еще в самом начале своего восхождения. Первым делом, после своего рождения, он участвовал в телевизионном шоу «Jeopardy!», русский аналог викторины «Своя игра», обыграв при этом двух самых выдающихся игроков – Кена Дженнингса (Ken Jennings) и Брэда Руттера (Brad Rutter). Один из участников, с большой радостью признал свое поражение и рядом с ответом финальный вопрос приписал: «От всей души приветствую наших новых компьютерных правителей».

Однажды с Watson произошла очень забавная история. Его создатели для того, чтобы он лучше понимал смысл вопросов, закалили в его память словарь уличного жаргона. Но этот запас слов оказался исключительно вреден для суперкомпьютера. После случая, когда на вопрос он ответил нецензурным ругательством, разработчики решили очистить его память от сленга и поставить фильтр на выдаваемые слова, чтобы не допустить ненормативной лексики в прямом эфире.

После своей незаурядной победы, IBM Watson пошел учиться в мединститут, ведь как раз для этого он был создан на свет. Проведя всего лишь 2 года в медицинском колледже Кливлендской клиники, Watson изучил около 600 тыс. медицинских документов, что составляет примерно 2 млн страниц. Джеймс Столлер, доктор медицинских наук, председатель Института образования в клинике Кливленда



отзывается о IBM Watson следующим образом: «Врачи завтрашнего дня будут использовать новые инструменты и технологии, чтобы дополнить свои собственные знания и опыт в области медицины. Технология никогда не заменит врача, но она способна сделать нас лучше». Перед тем, как проходить врачебную практику, будущий доктор Watson проанализировал и проработал около 25 тысяч историй болезни для очень тонкой настройки своих внутренних алгоритмов. В дополнение к этим знаниям, компания IBM заключила соглашение о сотрудничестве с одним из центров изучения рака Memorial Sloan Kettering, где собран поразительный объем медицинской информации. Watson способен не только ставить диагноз, но и определять наиболее оптимальный курс лечения для пациентов...

Точность назначения оптимального лечения после диагностирования рака легких в различных больницах США составляет около 50 процентов. То есть в половине случаев врачи рекомендуют далеко не самый идеальный курс лечения. Так вот, у IBM Watson точность назначения оптимального лечения приближается к 90 процентам. В своем расчете курса лечения и назначения препаратов, он учитывает малейшие нюансы из медицинской карты больного и его генетическую информацию. Ожидается, что новая «звезда» медицины сможет в простой и удобной форме показать врачам цепочку своих рассуждений и источники, на основе которых он делает свое умозаключение. В феврале 2013 года в компании IBM объявили, что сотрудничество с клиникой Memorial Sloan Kettering в области диагностики рака станет первым коммерческим применением компьютера Watson. В дальнейшем он «планирует» поработать и в сфере финансового анализа. Кто-то сорвет большой куш?

ПРИНЦИПИАЛЬНЫЕ ОСНОВЫ ОТКРЫТОГО ПРОЕКТА, НАПРАВЛЕННОГО НА СОЗДАНИЕ ТЕХНОЛОГИИ КОМПОНЕНТНОГО ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

УДК 004.8

В.В. Голенков, Н.А. Гулякина,
БГУИР, г. Минск

Аннотация

В работе рассматривается открытый проект, направленный на создание и развитие технологии компонентного проектирования интеллектуальных систем.

Введение

Анализ современных технологий проектирования интеллектуальных систем показывает, что они имеют ряд недостатков:

- технологии искусственного интеллекта не ориентированы на широкий круг разработчиков интеллектуальных систем и, следовательно, не получили массового распространения;

- велики сроки разработки интеллектуальных систем и велика трудоемкость их сопровождения;

- высока степень зависимости технологий искусственного интеллекта от платформ, на которых они реализованы, что приводит к существенным изменениям технологий при переходе на новые платформы;

- для эффективной реализации даже существующих моделей представления знаний и моделей решения трудно формализуемых задач современные компьютеры оказываются плохо приспособленными, что требует разработки принципиально новых компьютеров;

- современное состояние в области проектирования интеллектуальных компьютерных систем представляет собой «вавилонское столпотворение» самых различных моделей, методов, средств, платформ;

- отсутствуют подходы, позволяющие на некоторой универсальной основе интегрировать научные и практические результаты в области искусственного интеллекта, что порождает высокую степень дублирования результатов. В частности, высока трудоемкость интеграции различных моделей представления информации, моделей обработки информации, моделей решения задач и, следовательно, различных интеллектуальных компьютерных систем.

Искусственный интеллект является междисциплинарной научной дисциплиной. Этим обусловлен большой ее потенциал, так как на стыках научных направлений рождаются сильные результаты. Но этим же обусловлены и большие трудности, так как развитие искусственного интеллекта требует глубокого взаимопонимания и сотрудничества исследователей, имеющих разный стиль мышления, разный подход к объекту и предмету исследования, разный менталитет, разные целевые установки и традиции. Современный этап развития искусственного интеллекта остро нуждается в преодолении указанных трудностей.

Важнейшей задачей искусственного интеллекта в настоящее время является построение общей комплексной теории интеллектуальных систем, в рамках которой бы сочетались самые разные направления искусственного интеллекта – и теория представления знаний, и теория решения задач (в том числе различные исчисления, эвристики, стратегии), и теория программ (процедурных, декларативных, параллельных, последовательных), и архитектуры интеллектуальных систем, в том числе детализированные до уровня аппаратной поддержки, и теория интеллектуальных пользовательских интерфейсов, и компьютерная лингвистика [1, 5]. Сейчас эпицентром развития искусственного интеллекта является не столько разработка отдельных его направлений, сколько их глубокая семантическая интеграция, целью которой должна быть не только общая теория интеллектуальных систем, но и общая, доступная технология их комплексного проектирования.

Графодинамические модели

В качестве формальной основы проектируемых интеллектуальных систем предлагается использовать графодинамические модели обработки информации.

Графодинамическая модель обработки информации трактует процесс обработки информации как процесс преобразования графовой структуры, в ходе которого меняется не только состояние элементов этой графовой структуры, но и конфигурация этой структуры (появляются или удаляются ее вершины, а также связи между ними). Заметим, что для создания графодинамических моделей обработки информации недостаточно тех видов графовых структур, которые в настоящее время исследуются в теории графов [4, 6]. Приведем общее определение графовой структуры, на основе которого можно строить практически полезные графодинамические модели обработки информации.

Графовая структура G задается пятеркой $\langle V, C, K, M, I \rangle$, где

V – множество вершин (первичных элементов, терминальных элементов);

C – множество связующих элементов графовой структуры, каждый из которых задает некоторый фрагмент графовой структуры;

K – множество ключевых вершин графовой структуры, каждая из которых задает некоторый класс эквивалентных (однотипных) в определенном смысле элементов графовой структуры ($K \subset V$);

M – множество меток элементов (алфавит элементов) графовой структуры, каждая из которых задает неко-

торый базовый класс эквивалентных в определенном смысле элементов графовой структуры. К таким классам элементов, в частности, относятся:

- класс всех вершин графовой структуры;
- класс всех связующих элементов графовой структуры;
- класс всех ключевых вершин графовой структуры;
- класс всех меток графовой структуры;
- класс всех используемых в графовой структуре отношений инцидентности элементов графовой структуры.

I – множество используемых в графовой структуре отношений инцидентности элементов. Все эти отношения инцидентности являются бинарными ориентированными отношениями.

Параллельные асинхронные графодинамические модели

Без организации параллельной обработки информации невозможно рассчитывать на необходимую производительность подавляющего числа практически полезных интеллектуальных систем. Асинхронные модели обработки информации являются более гибкими, их легче интегрировать и наращивать новыми функциональными возможностями.

Графодинамическая модель параллельной асинхронной обработки информации, которую будем также называть графодинамической параллельной асинхронной машиной, трактуется нами как абстрактная многоагентная система, состоящая из:

- абстрактной графодинамической памяти, в которой хранятся обрабатываемые графовые структуры;
- коллектива агентов, работающих над общей для них графодинамической памятью и обменивающихся информацией только через эту память (в т.ч. и для координации своих действий).

Графодинамическая память носит реконфигурируемый, структурно перестраиваемый характер, поскольку процесс обработки графовых структур в конечном счете сводится к генерации и удалению различных элементов графовых структур, а также к генерации и удалению пар инцидентности между этими элементами. Другими словами, процесс обработки информации в графодинамической памяти сводится не только к изменению состояния элементов памяти, но и к изменению конфигурации связей между ними.

Агенты, работающие над общей графодинамической памятью, делятся на три вида:

- внутренние агенты, каждый из которых реагирует на определенного вида ситуации или события в графодинамической памяти и осуществляет изменение состояния графодинамической памяти, соответствующее своему функциональному назначению;
- рецепторные агенты, каждый из которых реагирует на определенные события во внешней среде и осуществляет первичное отражение этих событий в графодинамической памяти;
- эффекторные агенты, каждый из которых реагирует на определенного вида команды, формируемые внутренними агентами в графодинамической памяти, и осуществляет соответствующее изменение материального (физического) состояния интеллектуальной системы, которое определенным образом влияет на изменение ее внешней среды.

Агенты могут работать параллельно, если одновременно возникают условия инициирования агентов. Асинхронность деятельности внутренних агентов заключается в том, что наличие условия инициирования агента еще не означает начала его работы. Время реакции каждого внутреннего агента в известной мере субъективно и достаточно произвольно. Для обеспечения эффективного взаимодействия агентов, работающих над общей графодинамической памятью, наряду с предоставляемой им свободой, необходима разработка таких правил их поведения, которые гарантируют безопасность и производительность каждого из них. Для обеспечения безопасного выполнения своей задачи агент блокирует некоторые элементы графовой структуры, которая хранится в общей графодинамической памяти. Блокировка – это запрет, установленный заданным агентом и адресованный другим агентам, на выполнение тех или иных действий над заданным элементом хранимой графовой структуры. Сложность комплексного перехода на графодинамическую парадигму параллельной асинхронной обработки информации определяется исключительно психологическими обстоятельствами – это непривычно и, следовательно, боязно. Но накопленный человечеством опыт по созданию компьютерных систем и, в частности, интеллектуальных систем позволяет этот переход сделать достаточно быстро, так как многие проблемы, возникающие при реализации и применении графодинамических моделей, имеют достаточно близкие аналоги в традиционных компьютерных системах, но многие из них могут быть решены значительно проще и элегантнее. Кроме того, в результате перехода к графодинамическим моделям «обнажается» целый ряд проблем, которые ранее просто были не видны. Эпицентром такого перехода является формализация семантики и разработка семантически совместимых языковых средств представления различных видов знаний.

Унификация семантического представления знаний

В основе предлагаемого подхода к созданию технологии проектирования интеллектуальных систем лежит принцип унификации семантических моделей интеграции знаний и семантических моделей интеграции целых интеллектуальных систем.

На основе унифицированных семантических сетей необходимо обеспечить построение унифицированных семантических моделей интеграции знаний (понимания знаний) и использовать эти модели, во-первых, как основу процесса приобретения интеллектуальной системой новых знаний, как со стороны конечных пользователей, так и со стороны разработчиков; во-вторых, как основу интеграции программ и различных семантических моделей расширения задач; в-третьих, как основу интеграции абстрактных логико-семантических моделей интеллектуальных систем.

Унифицированный способ кодирования различных видов знаний необходим для обеспечения совместимости различных видов знаний и различных языков представления знаний, что, в свою очередь, необходимо для интеграции знаний [1, 2], а также для интеграции различных языков представления знаний. Без приведения

интегрируемых знаний к одинаковой (общей, унифицированной) форме интеграция невозможна.

В качестве такой унифицированной формы записи всевозможных знаний предлагается SC-код (Semantic Code) [3, 4, 7]. Это предполагает разработку соответствующего стандарта, выделяющего из всего многообразия абстрактных языков семантических сетей определенный базовый универсальный язык семантических сетей, который мы назвали SC-кодом (Semantic Computer code).

В основе минимизации алфавита элементов лежит следующее свойство семантических сетей. Метки, входящие в состав алфавита элементов семантической сети, и ключевые узлы этой семантической сети семантически эквивалентны в том смысле, что просто являются синтаксически различными способами выделения (задания) различных классов элементов семантической сети. При этом, заметим, что в отличие от алфавита символов линейного текста, все элементы алфавита (все метки) семантической сети, как и все ее ключевые узлы, имеют семантическую интерпретацию на описываемой предметной области. Таким образом, метки элементов семантической сети без какого-либо изменения семантики этой семантической сети можно «превращать» (преобразовывать) в ее ключевые узлы. При этом семантическая интерпретация каждого такого ключевого узла будет совпадать с семантикой соответствующей преобразованной метки.

Унифицированные семантические сети (sc-тексты) – это абстрактная семантическая модель знаний, являющаяся инвариантом различных способов представления и кодирования этих же знаний (в том числе и самих семантических сетей). Наличие такого инварианта необходимо для решения проблемы интеграции самых различных видов знаний. На основе унифицированных семантических сетей можно строить семантические модели различных компьютерных систем и решить проблему интеграции таких систем.

SC-код является абстрактным языком в том смысле, что способ изображения (материализации) его текстов не уточняется. Следовательно, можно разрабатывать различные графические уточнения SC-кода (например, SCg-код), различные варианты изображения sc-текстов в виде строк символов (например, SCs-код), различные варианты машинного представления sc-конструкций в адресной памяти традиционных компьютеров, а также в специальной структурно перестраиваемой ассоциативной памяти будущих компьютеров, ориентированных на обработку баз знаний. Все sc-элементы, кроме терминальных sc-узлов, являются обозначениями множеств, состоящих из sc-элементов (множеств sc-элементов).

SC-код представляет собой достаточно простой компьютерный код семантических сетей, который является не «инородным» представлением семантических сетей, а их представлением тоже в виде семантических сетей, но максимально простого вида – с минимальным алфавитом и с бинарными связками.

SC-код ориентирован на представление информации в компьютерной памяти и может рассматриваться как основа модели структурно перестраиваемой ассоциативной памяти будущих компьютеров, ориентированных на обработку семантических сетей. SC-код можно

рассматривать как универсальную основу машинного кодирования знаний в памяти будущих компьютеров, ориентированных на обработку семантических сетей. В такой памяти биты и байты «уступят место» sc-дугам, sc-ребрам и sc-узлам.

На базе SC-кода можно создавать целое семейство совместимых специализированных языков, ориентированных на представление самых разных видов знаний (логических формул и высказываний, программ, вопросов, поведенческих целей, различных видов моделей динамических систем и т.п.), таким образом, чтобы тексты всех этих специализированных языков полностью соответствовали SC-коду (т.е. были sc-текстами). Такие специализированные языки, общим носителем которых является SC-код, будем называть sc-языками. Каждый sc-язык определяется своим расширением множества ключевых узлов SC-кода.

SC-код представляет собой ядро универсального открытого языка семантических сетей, являющегося результатом интеграции всевозможных языков семантических сетей, построенных на основе SC-кода, и задаваемого:

- фиксированным алфавитом (алфавитом SC-кода);
- постоянно расширяемым (открытым) семейством ключевых узлов, в состав которого входят все ключевые узлы всех интегрируемых языков.

SC-код представляет собой единство языка и метаязыка. Так, например, в виде sc-конструкций можно описать синтаксис, семантику и онтологию SC-кода. С формальной точки зрения SC-код можно трактовать как метаязык базовой семантической спецификации sc-элементов с помощью специального набора ключевых узлов SC-кода. Единство языка и метаязыка в SC-коде проявляется в том числе и на самом низком уровне – на уровне sc-дуг принадлежности $\langle si, ei \rangle$, в которых сам sc-элемент ei , а не обозначаемый им объект, является элементом множества, обозначаемого sc-узлом si .

SC-код позволяет описать структуру любой информационной конструкции, не принадлежащей SC-коду, на любом уровне (на любом этапе синтаксического и семантического анализа). В частности, первичную синтаксическую структуру любой информационной конструкции можно представить в виде изоморфной sc-конструкции. Следовательно, SC-код может быть использован в качестве метаязыка для описания любого внешнего языка, т.е. языка, тексты которого не являются sc-конструкциями.

SC-код хорошо приспособлен к использованию в условиях так называемых не-факторов – нестационарности, неточности, противоречивости, неактуальности знаний, а также неполноты знаний (нечеткости, несформированности множеств, несформированности внешних информационных конструкций).

Информационные конструкции SC-кода (sc-конструкции) легко визуализируются.

Графовые языки программирования

Для описания способов решения задач и поведения агентов над общей графодинамической памятью предлагается использовать графовые языки программирования, которые ориентированы на обработку унифицированных

семантических сетей и программы которых сами являются унифицированными семантическими сетями.

Если все используемые в интеллектуальной системе графовые языки программирования привести к общему унифицированному стандарту – к SC-коду (это требует представления в виде sc-текстов не только самих программ, но и обрабатываемых ими данных), то можно достаточно эффективно решать проблему формализации семантической совместимости программ, написанных не только на одном, но и на разных языках программирования.

В традиционных языках программирования синтаксическая структура и семантика хранящихся в памяти обрабатываемых данных отдается на откуп программисту, в результате чего это делается по принципу «кто во что горазд».

Наряду с применением в интеллектуальных системах множества sc-языков самого различного назначения, востребованным является использование целого семейства совместимых sc-языков программирования, которые могут иметь разный уровень, могут быть последовательными, процедурными и декларативными.

Важнейшей особенностью всех этих языков является использование ассоциативного доступа к обрабатываемым фрагментам хранимого в графодинамической памяти sc-текста.

Операционная семантика каждого такого графового языка программирования (точнее, sc-языка программирования) задается коллективом агентов над общей графодинамической памятью, которые обеспечивают интерпретацию любой программы указанного языка программирования, хранящейся вместе с обрабатываемой информацией в указанной графодинамической памяти.

Программы представленные в виде семантической сети и описывающие обработку семантических сетей, а также соответствующие им языки программирования фактически открывают новую страницу в теории программирования, которую можно назвать семантической теорией программ и языков программирования.

Библиотека типовых семантически совместимых компонентов интеллектуальных систем и методика модульного проектирования интеллектуальных систем

В целях ускорения процесса проектирования интеллектуальных систем предлагается создать общую библиотеку многократно используемых семантически совместимых компонентов интеллектуальных систем, на основе которой разработать методику модульного (компонентного, сборочного) проектирования интеллектуальных систем. В указанной библиотеке можно выделить следующие разделы (частные библиотеки):

- библиотека многократно используемых компонентов баз знаний. Прежде всего, в эту библиотеку входят самые различные по содержанию, но семантически совместимые онтологии;

- библиотека компонентов семантических моделей информационного поиска. Сюда, прежде всего, входят различные информационно-поисковые агенты;

- библиотека компонентов семантических моделей интеграции знаний и машин обработки знаний;

- библиотека интерпретаторов программ, соответствующих различным языкам программирования;

- библиотека различных стратегий решения задач, различных моделей решения задач и агентов, входящих в состав таких моделей;

- библиотека компонентов пользовательских интерфейсов.

Все компоненты, включаемые в состав общей библиотеки компонентов интеллектуальных систем, оформляются как компоненты интеллектуальной собственности (intellectual property), поэтому будем их также называть ip-компонентами.

Особо подчеркнем то, что модульное проектирование интеллектуальных систем возможно только в том случае, если отбор компонентов, включаемых в состав рассмотренной библиотеки, будет осуществляться на основе тщательного анализа качества этих компонентов. Одним из важнейших критериев такого анализа является семантическая совместимость анализируемых компонентов со всеми компонентами, имеющимися в текущей версии библиотеки.

Для обеспечения семантической совместимости таких компонентов интеллектуальных систем, которые являются унифицированными семантическими моделями (sc-моделями знаний, sc-моделями машин обработки знаний, sc-агентов, sc-моделями интеллектуальных подсистем), необходимо (1) согласовать семантику (смысл) всех используемых ключевых узлов и (2) согласовать глобальные идентификаторы ключевых узлов, используемых в разных компонентах. После этого интеграция всех компонентов, входящих в состав библиотеки, и в любых комбинациях осуществляется автоматически, без вмешательства разработчика.

Платформенно-независимый характер проектирования интеллектуальных систем

Для максимальной платформенной независимости технологии необходимо обеспечить четкое разделение процесса проектирования формального описания логико-семантической модели разрабатываемой интеллектуальной системы от процесса реализации (интерпретации) этой модели на той или иной платформе. Если каждой интеллектуальной системе соответствует своя уникальная логико-семантическая модель, то каждый интерпретатор абстрактных логико-семантических моделей интеллектуальных систем должен обеспечивать интерпретацию целого класса таких моделей, а в идеале – интерпретацию любой такой модели. Следовательно, разработка указанных интерпретаторов может осуществляться абсолютно независимо от разработки логико-семантических моделей конкретных интеллектуальных систем.

Таким образом, SC-код, обеспечивающий унификацию семантического представления любых знаний, вместе с языком SCP [3, 4], обеспечивающим унификацию формального описания агентов, работающих над семантической памятью, являясь средством унификации логико-семантических моделей интеллектуальных систем, выполняют в рамках предлагаемой технологии роль, аналогичную той, которую выполняет язык VHDL в современных микроэлектронных технологиях. В лице SC-кода и языка SCP мы имеем стандарт полного фор-

мального описания логико-семантических моделей интеллектуальных систем, обеспечивающий независимость проектирования абстрактных логико-семантических моделей конкретных интеллектуальных систем от разработки различных вариантов реализации (различных вариантов их интерпретации на различных платформах). Такой стандарт является своего рода «водоразделом» между полным платформенно-независимым описанием интеллектуальной системы (абстрактной логико-семантической моделью) и платформенно зависимой реализацией (интерпретацией) этой абстрактной модели.

Полностью построенная абстрактная логико-семантическая модель проектируемой интеллектуальной системы:

- является открытой, поскольку ее можно легко пополнять новыми знаниями и навыками, интегрируя их в текущую версию модели;
- концентрирует внимание на семантические аспекты функционирования интеллектуальной системы и не содержит никаких лишних деталей, обусловленных тем или иным способом ее технической реализации (интерпретации);
- является абстрактным инвариантом целого множества самых различных способов ее технологической реализации (в том числе и с помощью принципиально новых компьютеров).

Разработка прототипа интеллектуальной системы завершается разработкой полной sc-модели этой системы, которая записывается в виде исходного текста с использованием таких языковых средств, как SCg-код, SCs-код, SCn-код. После этого разработчик выбирает один из универсальных (!) вариантов интерпретации (реализации) sc-моделей, загружает разработанные им исходные тексты в выбранный интерпретатор и получает прототип, пригодный для опытной эксплуатации и последующего совершенствования.

Очевидно, что для каждого варианта интерпретации sc-моделей интеллектуальных систем указанная методика будет иметь свои особенности. Следовательно, нижние уровни детализации проектируемых интеллектуальных систем, в отличие от верхнего (логико-семантического) являются платформенно-зависимыми. Можно говорить о различных модификациях технологии проектирования интеллектуальных систем, соответствующих разным платформам. Напомним при этом, что основная трудоемкость проектирования интеллектуальных систем, полностью определяющая уровень ее возможностей (уровень знаний и навыков) концентрируется именно на 1-ом этапе проектирования – на разработке ее абстрактной логико-семантической модели.

Таким образом, проектирование интеллектуальной системы можно организовать как два следующих самостоятельных процесса, выполняемых одновременно и независимо друг от друга:

- процесс разработки абстрактной унифицированной логико-семантической модели проектируемой интеллектуальной системы;
- процесс совершенствования выбранного интерпретатора абстрактных унифицированных логико-семантических моделей интеллектуальных систем.

Заметим, что сама идея обеспечения кросс-

платформенной разработки компьютерных систем путем внедрения формального языка, обеспечивающего описание абстрактных (логических) моделей этих систем не нова. Существует целый ряд кросс-платформенных технологий. Вопрос в том (1) о каком классе разрабатываемых компьютерных систем идет речь, (2) какими свойствами обладают используемые абстрактные модели компьютерных систем, (3) какими достоинствами обладает технология разработки самих этих абстрактных моделей.

Вопросы программной реализации и, в частности, web-ориентированной реализации унифицированных логико-семантических моделей интеллектуальных систем рассмотрены в работе [8].

Семантический ассоциативный параллельный компьютер

Для реализации унифицированных логико-семантических моделей интеллектуальных систем предлагается использовать не только современные компьютеры, но и семантические ассоциативные параллельные компьютеры, специально ориентированные на аппаратную реализацию таких моделей.

Очевидно, что для указанных компьютеров базовый графовый язык программирования (язык SCP) является их ассемблером, т.е. аппаратно интерпретируемым языком программирования.

В связи с проблемой создания компьютеров, ориентированных на обработку знаний, необходимо отметить следующее:

- в таких компьютерах принципиально важна поддержка именно параллельной обработки знаний;
- опыт использования параллельных компьютеров показывает, что эффективное их использование предполагает разработку качественных параллельных программ требует особой профессиональной подготовки и высокой квалификации. Мир параллельного программирования требует особой культуры, особого стиля мышления. Еще более серьезная профессиональная подготовка требуется для разработки параллельных программ, ориентированных на обработку знаний и использующих ассоциативный доступ к обрабатываемой информации;
- уровень развития микроэлектронных технологий сейчас позволяет достаточно быстро реализовывать самые смелые компьютерные архитектуры и модели обработки информации. Необходима только четкая постановка задачи;
- созданию параллельных компьютеров для обработки знаний должно предшествовать создание технологии проектирования интеллектуальных систем, в основе которой лежат те модели параллельной обработки знаний, которые будут аппаратно поддерживаться в указанных компьютерах;
- предлагаемая технология проектирования интеллектуальных систем как раз и предполагает последовательное выполнение следующих этапов:
 - разработка технологии проектирования абстрактных унифицированных логико-семантических моделей интеллектуальных систем;
 - разработка нескольких вариантов программной реализации абстрактных унифицированных логико-

семантических моделей интеллектуальных систем, выполненных на современных компьютерах;

– разработка, эксплуатация достаточно большого количества прикладных интеллектуальных систем и совершенствование технологии проектирования интеллектуальных систем на основе приобретенного опыта;

– разработка семантического ассоциативного компьютера, появление которого не отменит абсолютно ничего, сделанного ранее. Просто появится еще один, но уже аппаратный вариант реализации абстрактных унифицированных логико-семантических моделей интеллектуальных систем, применение которого для уже разработанных абстрактных унифицированных логико-семантических моделей самых различных прикладных систем для конечных пользователей этих интеллектуальных систем абсолютно ничего не изменит, кроме существенного повышения быстродействия.

Рассматривая абстрактную sc-машину обработки знаний на самом верхнем уровне, мы не уточняем (не детализируем) «внутреннее устройство» sc-агентов обработки знаний. Разработав язык SCP, мы получили возможность формально описывать (детализировать) поведение sc-агентов обработки знаний. Если трактовать язык SCP как ассемблер семантического ассоциативного компьютера, то проектирование этого компьютера можно рассматривать как формальный переход к sc-машинам более низкого уровня, обеспечивающим интерпретацию sc-машин более высокого уровня. Существенным здесь является то, что при этом мы не выходим за пределы класса абстрактных sc-машин. Просто вводится последовательность sc-языков программирования все более и более низкого уровня, каждый из которого обеспечивает формальное описание sc-агентов, входящих в состав sc-машины, интерпретирующей программы непосредственно предшествующего ему sc-языка программирования более высокого уровня. При этом число таких уровней, т.е. число таких специальных sc-языков программирования (которые можно назвать sc-языками микропрограммирования) должно быть столько, сколько необходимо для доведения формального описания sc-машин до такого уровня детализации, который позволяет перейти от соответствующего абстрактного языка микропрограммирования к формальному описанию цифровой аппаратуры на языке VHDL.

Архитектуру аппаратной реализации семантических моделей обработки знаний можно рассматривать как иерархию абстрактных машин, описывающих переход от агентов, имеющих доступ ко всей семантической памяти, к агентам, имеющим доступ только к своей семантической окрестности и, в конечном счете, взаимодействующим только со своими семантическими соседями. Аппаратная интерпретация абстрактных sc-машин предполагает создание реконфигурируемой памяти с распределенными в ней процессорными элементами. Такую интеграцию памяти и процессора будем называть процессоро-памятью. Реконфигурируемость (структурная перестраиваемость) памяти может быть обеспечена коммутационной средой для процессорных элементов. Можно рассматривать целый ряд подходов к реализации реконфигурируемой семантической ассоциативной процессоро-памяти.

В частности, процессорным элементам можно ставить в соответствие узлы обрабатываемых унифицированных семантических сетей, а коммутируемым каналам связи между процессорными элементами – коннекторы этой семантической сети. В этом случае текущее состояние конфигурации коммутируемых каналов связи будет полностью соответствовать текущему состоянию конфигурации обрабатываемой семантической сети. Следовательно, память «превращается» из пассивного хранилища байтов в коммутационную среду между процессорными элементами.

Доступность и открытость технологии

Доступность и открытость технологии проектирования интеллектуальных систем предполагает обеспечить максимально возможное расширение контингента разработчиков интеллектуальных систем, использующих предлагаемую технологию.

Если технология проектирования интеллектуальных систем ориентирована на широкое, массовое распространение и на интенсивное собственное развитие, опирающееся на накапливаемый опыт ее использования, она должна быть доступной и открытой. Это означает:

– свободный доступ ко всей документации и основным средствам автоматизации (компьютерной поддержки) проектирования интеллектуальных систем;

– открытость исходных текстов всех основных средств компьютерной поддержки проектирования интеллектуальных систем, всех основных многократно используемых (типовых) компонентов интеллектуальных систем;

– открытость исходных текстов всех «пилотных» проектов прикладных интеллектуальных систем, выполняющих роль «образцово-показательных» проектов;

– открытый характер организации (project-менеджмента) процесса развития технологии, имеющий форму открытого (open source) проекта, участником которого может быть любой желающий, в том числе, и любой пользователь этой технологии, указывающий на различные ошибки и высказывающий различные пожелания.

Эволюционная методика проектирования

Использование методики поэтапного эволюционного проектирования интеллектуальных систем предполагает:

– быстрое проектирование;

– скорейшее введение в эксплуатацию первых версий проектируемой системы с минимальными, но практически полезными возможностями;

– эволюционное поэтапное совершенствование проектируемой интеллектуальной системы путем ее расширения новыми знаниями и навыками непосредственно в ходе эксплуатации интеллектуальной системы и активным привлечением ее конечных пользователей.

С формальной точки зрения проектирование унифицированной логико-семантической модели (sc-модели) интеллектуальной системы, в конечном счете, сводится к проектированию sc-модели базы знаний этой интеллектуальной системы, поскольку scp-программы, описывающие поведение sc-агентов, можно рассматривать как часть базы знаний. Таким образом, проектируемая база знаний включает в себя:

– базу знаний предметной (основной) интеллектуальной системы;

– тексты всех scr-программ, описывающих поведение sc-агентов;

– текст документации, представленный в виде базы знаний интеллектуальной help-системы, обеспечивающей всестороннее информационное обслуживание пользователей проектируемой интеллектуальной системы.

Начальный этап проектирования базы знаний интеллектуальной системы – это уточнение иерархической системы предметных областей, которые должны быть описаны в проектируемой базе знаний. Каждой такой предметной области ставится в соответствие определенный раздел проектируемой базы знаний. Среди выделенных разделов проектируемой базы знаний имеются разделы, которые делятся (декомпозируются) на подразделы, а также атомарные (недекопируемые) разделы. Далее процесс проектирования всей базы знаний сводится к проектированию каждого ее атомарного раздела с последующей их интеграцией в единую базу знаний.

В целом, начальную стадию проектирования всей интеллектуальной системы на основе предлагаемой технологии можно условно разбить на следующие четыре этапа:

1. Разработка 1-й версии интеллектуальной системы, которая включает в себя:

- 1-ю версию ее базы знаний;
- типовое ядро интеллектуальной информационно-поисковой машины, которое входит в состав библиотеки многократно используемых компонентов интеллектуальных систем;
- типовое ядро интеллектуального решателя, которое входит в состав библиотеки многократно используемых компонентов интеллектуальных систем;
- типовое ядро пользовательского интерфейса, которое входит в состав библиотеки многократно используемых компонентов интеллектуальных систем.

Разработанная 1-я версия интеллектуальной системы уже обладает определенной целостностью, ее можно тестировать и запускать в предварительную опытную эксплуатацию.

2. Разработка 2-й версии интеллектуальной системы, которая включает в себя:

- 2-ю версию ее базы знаний;
- 1-ю версию ее информационно-поисковой машины;
- типовое ядро ее интеллектуального решателя;
- типовое ядро ее пользовательского интерфейса;

3. Разработка 3-й версии интеллектуальной системы, включающей в себя:

- 3-ю версию ее базы знаний;
- 2-ю версию ее информационно-поисковой машины;
- 1-ю версию ее интеллектуального решателя;
- типовое ядро ее пользовательского интерфейса;

4. Разработка 4-й версии интеллектуальной систем, включающей в себя:

- 4-ю версию ее базы знаний;
- 3-ю версию ее информационно-поисковой машины;
- 2-ю версию ее интеллектуального решателя;
- 1-ю версию ее пользовательского интерфейса.

Дальнейшее развитие проектируемой интеллектуальной системы может акцентировать внимание на самых разных направлениях, приоритетность которых определяется самим приложением.

Более подробно методика эволюционного коллективного проектирования унифицированных семантических моделей интеллектуальных систем, на основе содержательной структуризации знаний, описана в работе [9].

Реализация предлагаемой технологии в виде интеллектуальной метасистемы

Указанная интеллектуальная метасистема должна включать в себя:

– теорию (принципы построения) проектируемых интеллектуальных систем, которая входит в состав базы знаний метасистемы;

– библиотеку типовых многократно используемых компонентов (ip-компонентов) интеллектуальных систем, которая входит в состав базы знаний рассматриваемой метасистемы;

– средства автоматизации синтеза, анализа и имитационного моделирования проектируемых интеллектуальных систем и их компонентов (это подсистема интеллектуальной метасистемы, ориентированная на решение задач проектирования интеллектуальных систем);

– интеллектуальную help-систему, являющуюся подсистемой рассматриваемой интеллектуальной метасистемы ориентированной на информационное обслуживание и обучение разработчиков интеллектуальных систем;

– методику проектирования интеллектуальных систем, которая оформляется как часть базы знаний метасистемы;

– методику обучения проектированию интеллектуальной системы, которая также является частью базы знаний метасистемы;

– интеллектуальную подсистему управления проектированием самой метасистемы;

– интеллектуальную подсистему управления информационной безопасностью метасистемы;

– семейство различных вариантов реализации интерпретаторов унифицированных абстрактных логико-семантических моделей интеллектуальных систем.

Учитывая рассматриваемые выше принципы построения предлагаемой нами технологии, она названа Открытой Семантической Технологией проектирования Интеллектуальных Систем (Open Semantic Technology for Intelligent Systems – OSTIS). Соответственно этому, интеллектуальную метасистему, ориентированную на поддержку проектирования интеллектуальных систем, будем называть метасистемой OSTIS.

В интеллектуальной метасистеме OSTIS можно выделить целый ряд подсистем, ориентированных на поддержку проектирования различных компонентов интеллектуальных систем, таких, как:

– базы знаний и различные фрагменты баз знаний (онтологии, формальные теории, программы);

– информационно-поисковые машины, машины интеграции знаний, решатели задач;

– пользовательские интерфейсы (графические, естественно-языковые, мультимодальные).

В интеллектуальной метасистеме OSTIS можно также выделить семейство интеллектуальных подсистем, ориентированных на поддержку проектирования различных классов интеллектуальных систем, таких, как:

– интеллектуальные справочные системы (системы информационного обслуживания);

- интеллектуальные обучающие системы (имеющие подсистемы интеллектуального управления обучением);
- интеллектуальные help-системы для пользователей различных компьютерных систем;
- интеллектуальные системы автоматизированного проектирования;
- интеллектуальные системы управления проектами.

Заключение

Основные проблемы повышения эффективности интеллектуальных систем касаются не конечных пользователей, а их разработчиков. Современным требованиям не удовлетворяет не столько качество разрабатываемых интеллектуальных систем, сколько недостаточное их количество, высокая трудоемкость и наукоемкость их разработки и, соответственно, недопустимо большие сроки их проектирования.

Подчеркнем при этом, что технологии проектирования интеллектуальных систем, как и информационных технологий в целом, имеют во многом похожие проблемы и требуют существенного переосмысления.

Таким образом, создание описанной в данной статье технологии позволит решить ряд существенных проблем, рассмотренных в данной работе, что, в свою очередь, радикально изменит возможности разработчика, проектирующего интеллектуальные системы и расширит спектр решаемых подобными системами задач.

Литература:

1. Глоба, Л.С. Модели и методы интеграции информационных и вычислительных ресурсов / Л.С. Глоба, Р.Л. Новогрудская // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем». – Минск, 2012. – С. 447–452.
2. Гаврилова, Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский. – СПб: Питер, 2000. – 384 с.
3. Голенков, В.В. Принципы построения массовой семантической технологии компонентного проектирования интеллектуальных систем / В.В. Голенков, Н.А. Гулякина // В кн Междунар. научн.-техн. конф. «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2011). Материалы конф. [Минск, 10-12 февр. 2011 г.]. – Минск: БГУИР, 2011. – С. 21–59.

систем» (OSTIS-2011). Материалы конф. [Минск, 10-12 февр. 2011 г.]. – Минск: БГУИР, 2011. – С. 21–59.

4. Голенков, В.В., Гулякина Н.А. Графодинамические модели параллельной обработки знаний: принципы построения, реализации и проектирования / В.В. Голенков, Н.А. Гулякина // В кн Междунар. научн.-техн. конф. «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2012). Материалы конф. [Минск, 16-18 февр. 2012 г.]. – Минск: БГУИР, 2012. – С. 23–52.

5. Кузнецов, О.П. Интеллектуализация поддержки управляющих решений и создание интеллектуальных систем / О.П. Кузнецов // Теория и методы разработки программного обеспечения систем управления. Спецвыпуск журнала «Проблемы управления» №3.1, 2009.

6. Sowa, J. Conceptual Graphs / John F. Sowa, F. van Harmelen, V. Lifschitz, B. Porter // eds., Handbook of Knowledge Representation. – Elsevier, 2008. – P. 213–237.

7. Открытая семантическая технология проектирования интеллектуальных систем [Электронный ресурс]. – 2012. – Режим доступа: <http://www.ostis.net>. – Дата доступа: 18.11.2012.

8. Колб, Д.Г. Web-ориентированная реализация семантических моделей интеллектуальных систем / Д.Г. Колб // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» – Минск, 2012. – С. 111–122.

9. Давыденко, И.Т. Комплексная методика проектирования семантических моделей интеллектуальных справочных систем / И.Т. Давыденко // Материалы международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» – Минск, 2012. – С. 457–466.

Abstract

This article is devoted to the open project, aimed at creation of technology of component design of intellectual systems. Thus, the creation of the technology, which is described in this article, will solve a number of important issues discussed in this work, which, in turn, will radically change the capabilities of developer, designing intelligent systems and expand the range of such systems functions.

Поступила в редакцию 02.04.2013 г.

ПОДПИСКА-2013

Подписку на журнал «Электроника инфо» можно оформить в отделении связи по месту жительства, а также в редакции с любого месяца.

ПОДПИСКА В БЕЛАРУСИ: «Белпочта»
(подписной индекс – 00822).

ПОДПИСКА В РОССИИ: «Роспечать»
(подписной индекс – 00822),
«АРЗИ – Почта России»
(подписной индекс – 91654).

Читатели также могут подписаться по национальным каталогам: агентств «МК-Периодика», «Информнаука», «Интерпочта-2003» и «Урал-Пресс»; «Пресса» (Украина).

УВАЖАЕМЫЕ АВТОРЫ!

Присылайте Ваши статьи на электронный адрес редакции i.asmalouskaya@electronica.by или по адресу 220015, Минск, ул. Пономаренко, 35 А, к. 302 для публикации в журнале «Электроника инфо».

Тел.: +375 (17) 204 40 00. Материалы для рубрики «рецензируемые научные статьи» должны удовлетворять требованиям ВАК РБ (Электроника инфо, № 8, 2012, с. 104).

СОВРЕМЕННЫЕ РАЗРАБОТКИ В ОБЛАСТИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Какие возможности предоставляет искусственный интеллект в наши дни? Краткий ответ на этот вопрос сформулировать сложно, поскольку в этом научном направлении существует слишком много подобластей, в которых выполняется очень много исследований. Ниже приведен краткий обзор современных разработок с применением технологий искусственного интеллекта.

Автономное планирование и составление расписаний

Работающая на удалении в сотни миллионов километров от Земли программа Remote Agent агентства NASA стала первой бортовой автономной программой планирования, предназначенной для управления процессами составления расписания операций для космического аппарата. Программа Remote Agent вырабатывала планы на основе целей высокого уровня, задаваемых с Земли, а также контролировала работу космического аппарата в ходе выполнения планов: обнаруживала, диагностировала и устраняла неполадки по мере их возникновения.

Ведение игр

Программа Deep Blue компании IBM стала первой компьютерной программой, которой удалось победить чемпиона мира в шахматном матче, после того как она обыграла Гарри Каспарова со счетом 3,5:2,5 в показательном матче. Каспаров заявил, что ощущал напротив себя за шахматной доской присутствие «интеллекта нового типа». Журнал Newsweek описал этот матч под заголовком «Последний оборонительный рубеж мозга». Стоимость акций IBM выросла на 18 миллиардов долларов.

Автономное управление

Система компьютерного зрения Alvinn была обучена вождению автомобиля, придерживаясь определенной полосы движения. В университете CMU эта система была размещена в микроавтобусе, управляемом компьютером NavLab, и использовалась для проезда по Соединенным Штатам; на протяжении 4586,6 км система обеспечивала рулевое управление автомобилем в течение 98 % времени. Человек брал на себя управление лишь в течение остальных 2 %, главным образом на выездных пандусах. Компьютер NavLab был оборудован видеокамерами, которые передавали изображения дороги в систему Alvinn, а затем эта система вычисляла наилучшее направление движения, основываясь на опыте, полученном в предыдущих учебных пробегах.

Диагностика

Медицинские диагностические программы, основанные на вероятностном анализе, сумели достичь уровня опытного врача в нескольких областях медицины. Хекерман описал случай, когда ведущий специалист в области патологии лимфатических узлов не согласился с

диагнозом программы в особо сложном случае. Создатели программы предложили, чтобы этот врач запросил у компьютера пояснения по поводу данного диагноза. Машина указала основные факторы, повлиявшие на ее решение, и объяснила нюансы взаимодействия нескольких симптомов, наблюдавшихся в данном случае. В конечном итоге, эксперт согласился с решением программы.

Планирование снабжения

Во время кризиса в Персидском заливе в 1991 году в армии США была развернута система DART (Dynamic Analysis and Replanning) для обеспечения автоматизированного планирования поставок и составления графиков перевозок. Работа этой системы охватывала одновременно до 50 000 автомобилей, единиц груза и людей; в ней приходилось учитывать пункты отправления и назначения, маршруты, а также устранять конфликты между всеми параметрами. Методы планирования на основе искусственного интеллекта позволяли вырабатывать в течение считанных часов такие планы, для составления которых старыми методами потребовались бы недели. Представители агентства DARPA (Defense Advanced Research Project Agency – Управление перспективных исследовательских программ) заявили, что одно лишь это приложение стоило окупить тридцатилетние инвестиции в искусственный интеллект, сделанные этим агентством.

Робототехника

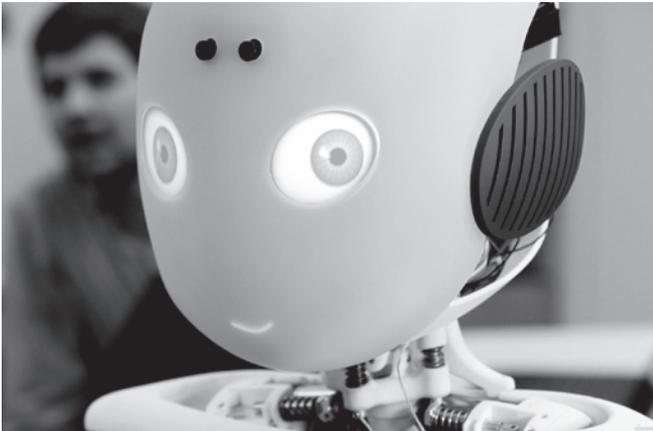
Многие хирурги теперь используют роботов-ассистентов в микрохирургии. Например, HipNav – это система, в которой используются методы компьютерного зрения для создания трехмерной модели анатомии внутренних органов пациента, а затем применяется робототехническое управление для руководства процессом вставки протеза, заменяющего тазобедренный сустав.

Понимание естественного языка и решение задач

Программа Proverb – это компьютерная программа, которая решает кроссворды намного лучше, чем большинство людей; в ней используются ограничения, определяющие состав возможных заполнителей слов, большая база с данными о встречавшихся ранее кроссвордах, а также множество различных источников информации, включая словари и оперативные базы данных, таких как списки кинофильмов и актеров, которые играли в этих фильмах. Например, эта программа способна определить, что одним из решений, подходящих для ключа «Nice Story», является слово «ETAGE», поскольку ее база данных содержит пару ключ-решение «Story in France/ETAGE», а сама программа распознает, что шаблоны «Nice X» и «X in France» часто имеют одно и то же решение. Программа не знает, что Nice (Ницца) – город во Франции, но способна разгадать эту головоломку.

aiportal.ru

РОБОТ-МАЛЬЧИК ROBOU



Ученые из Цюриха представили прототип робота-мальчика Robou. Антропоморфный робот должен будет помогать пожилым людям вести домашнее хозяйство.

Robou небольшого роста – 1,2 м. Он состоит из скелета и синтетических мышц, покрытых мягкой кожей. Пока неясно, какое у Robou будет лицо. Облик машины выберут Интернет-пользователи по итогам конкурса, проведенного в социальной сети Facebook.

По замыслу создателей робота, сотрудников Лаборатории искусственного интеллекта при университете Цюриха (Швей-

цария) и 15 других организаций, Robou должен стать первым в поколении роботов-помощников, которые смогут облегчить ведение быта пожилым людям и позволить им дольше сохранять самостоятельность.

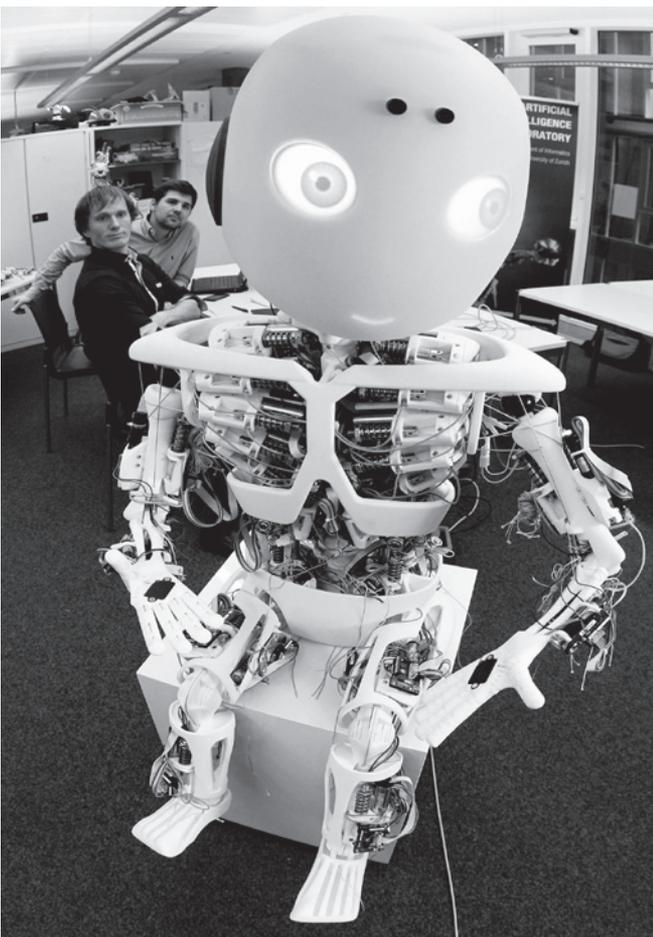
«Старение населения в Европе и Америке заставляет нас задуматься о пожилых людях, о том, как обеспечить им возможность жить самостоятельно и вести хозяйство как можно дольше. Мы думаем, что создание роботов-помощников в будущем может оказаться одним из решений этой проблемы», – заявляют руководители проекта.

Интересно, что создание робота-мальчика – первый такого рода проект, не имеющий финансирования. Robou создается полностью на пожертвованные деньги. Проект разработчики оценили в 500 тыс. евро, из них собрано уже почти 300 тыс.

Руководитель проекта, профессор Рольф Пфейфер, говорит, что робот будет обладать двигательной системой, функционирующей за счет искусственных сухожилий, полностью аналогичных человеческим. «Это означает, что двигаться робот будет с тем же изяществом и грацией, как и настоящий человек», – сказал он.

«Родители» Робоя – 40 ученых и инженеров, работающие при поддержке десятка компаний-партнеров, и возглавляемые проф. Рольфом Пфейфером – обещают вывести в свет свое творение к середине 2013 года.

pixanews.com



ИНТЕЛЛЕКТУАЛЬНАЯ ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНАЯ СИСТЕМА НА ОСНОВЕ LABVIEW ДЛЯ ЛАБОРАТОРНЫХ ПРАКТИКУМОВ ПО РАДИОЭЛЕКТРОНИКЕ

УДК 621.382.

С.В. Здоровцев, И.Г. Круглов, А.Г. Петрович,
ОАО «МНИПИ», г. Минск

Аннотация

В статье описывается эффективное решение задачи реализации лабораторных практикумов по радиоэлектронике с использованием интеллектуальной информационно-измерительной системы на основе универсальной среды графического программирования LabVIEW. В предлагаемом решении создан комплект аппаратно-программных средств, обеспечивающих оптимальный выбор режимов работы системы, сбор, обработку, хранение измерительной информации с последующим оформлением электронных отчетов.

Введение

В настоящее время быстрыми темпами осуществляется развитие технических и программных средств создания учебных стендов и экспериментальных установок, используемых в учебных процессах и лабораторных практикумах многих ведущих университетов. При этом обычно применяются программные средства LabVIEW и встраиваемые дорогостоящие измерительные модули фирмы National Instruments, что препятствует широкому использованию информационных технологий в учебном процессе. Высокая стоимость виртуальных приборов National Instruments обусловлена затратами на реализацию PCI-интерфейса для встраиваемых модулей.

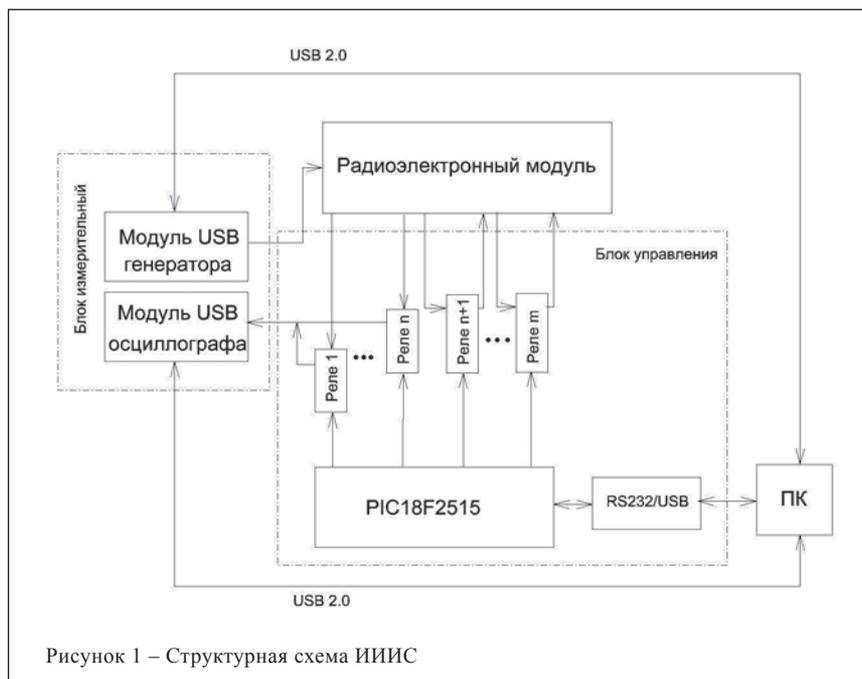
В настоящее время рядом фирм выпускаются виртуальные измерительные приборы с USB-интерфейсом [1], которые позволяют создать интеллектуальные средства измерений с расширенными функциональными возможностями.

Применение интеллектуальных измерительных устройств позволяет снизить требования к центральным управляющим компьютерам, а также сократить номенклатуру измерительных приборов до нескольких моделей, что, несомненно, имеет положительный эффект при проектировании и производстве. Следует также отметить специфику программных средств, используемых при создании архитектуры интеллектуальных измерительных устройств [2].

Основная часть

Предлагаемая ИИИС структурно состоит из трех функциональных групп. Первая группа включает радиоэлектронные модули первичной обработки информации. Вторая группа ИИИС представляет собой блок измерительный (БИ), включающий комплект интеллектуальных измерительных USB-приборов с соответствующим программным обеспечением. Третья группа представляет

собой блок управления (БУ) и информационный блок, включающий математическое, алгоритмическое и программное обеспечения системы на основе персонального компьютера (ПК). Первая и третья группы могут видоизменяться в зависимости от способа контроля параметров и функционального назначения ИИИС. Структурная схема рассматриваемой ИИИС представлена на рисунке 1.



Измерительный блок был построен на базе двух модулей: функционального USB-генератора сигналов и двухканального цифрового USB-осциллографа. Такой выбор обусловлен широкими возможностями интеллектуальных измерительных USB-приборов, использующих постоянно увеличивающиеся вычислительные возможности и гибкость ПК, основными достоинствами которых являются:

- удобство представления и обработки измерительной информации;
- настраиваемый интерфейс пользователя;
- возможность функционального расширения и наращивания;
- запись времени и комментариев вместе с данными;
- автоматизация процесса измерений;
- встроенные в измерительные процедуры возможности мультимедиа;
- взаимодействие с базами данных и информационными системами.

При этом автоматизированные средства разработки прикладных приложений, например, LabVIEW или LabWindows/CVI, делает простым процесс создания как

специализированных устройств, так и универсальных, комбинирующих возможности нескольких приборов.

В качестве основного критерия, определяющего технические требования, предъявляемые к USB-приборам, был выбран критерий соответствия параметров модуля генератора сигналов (форма сигнала, амплитуда, диапазон задаваемой частоты) и характеристик осциллографического модуля (количество входных каналов, диапазоны разверток амплитуды и времени) параметрам и характеристикам исследуемых радиоэлектронных модулей первичной обработки информации.

При разработке программной части компьютерной информационно-измерительной системы были использованы следующие виртуальные инструменты LabVIEW:

- Interpolate 1D.VI – программный интерполятор;
- Harmonic Distortion Analyzer.VI – измеритель нелинейных искажений;
- Measure_for_1chan(SubVI).VI – измеритель параметров сигнала;
- Basic Averaged DC-RMS.VI – измеритель постоянного напряжения;

– Square Function.VI – формирователь прямоугольного импульса.

Для функционирования ИИИС разработан протокол информационного взаимодействия узлов и подсистем, и организована синхронизация процессов измерений различных параметров радиоэлектронных блоков. В среде LabVIEW было реализовано управление интеллектуальными измерительными приборами (генератор сигналов, цифровой осциллограф), а также управление как отдельными функциональными узлами ИИИС, так и всей системы в целом. На рисунке 2 представлена блок-диаграмма программы USB-генератора сигналов.

Графический интерфейс пользователя реализован в виде графических образов панелей управления приборов. На рисунке 3 показан пример рабочего окна информационного дисплея в одном из режимов работы ИИИС при измерении параметров радиоэлектронного модуля в соответствии с методическими материалами [3]. В представленном рабочем окне отображается анализируемая электрическая схема радиоэлектронного модуля, виртуальные панели

измерительных приборов (генератор сигналов, цифровой осциллограф), параметры и характеристики исследуемого модуля в выбранном режиме ИИИС. В представленном рабочем окне отображен режим исследования амплитудно-частотной характеристики (АЧХ) исследуемого радиоэлектронного модуля.

Во время выполнения лабораторного практикума симулируется работа генератора сигналов, при этом пользователь может управлять частотой, амплитудой и формой сигнала, а с помощью двухканального USB-осциллографа измерять параметры и характеристики исследуемого модуля в контрольных точках при различных режимах работы схемы. Для подключения к исследуемой схеме тех или иных элементов используется коммутатор, который выполнен на малогабаритных реле, управляемых сигналами PIC-контроллера.

Блок управления связан с ПК через мост RS-232/USB, что позволяет упростить обмен данными путем использования механизма виртуальных COM-портов. Изменение точек подключения входов USB-осциллографа, элементов схемы производится путем подачи команд управления в PIC-контроллер, что обеспечивает возможность выполнения широкого круга лабораторных практикумов.

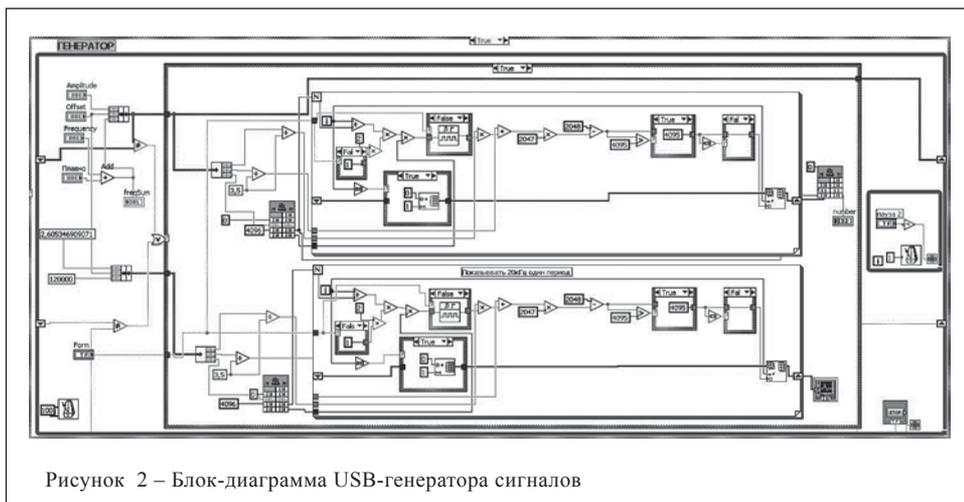


Рисунок 2 – Блок-диаграмма USB-генератора сигналов

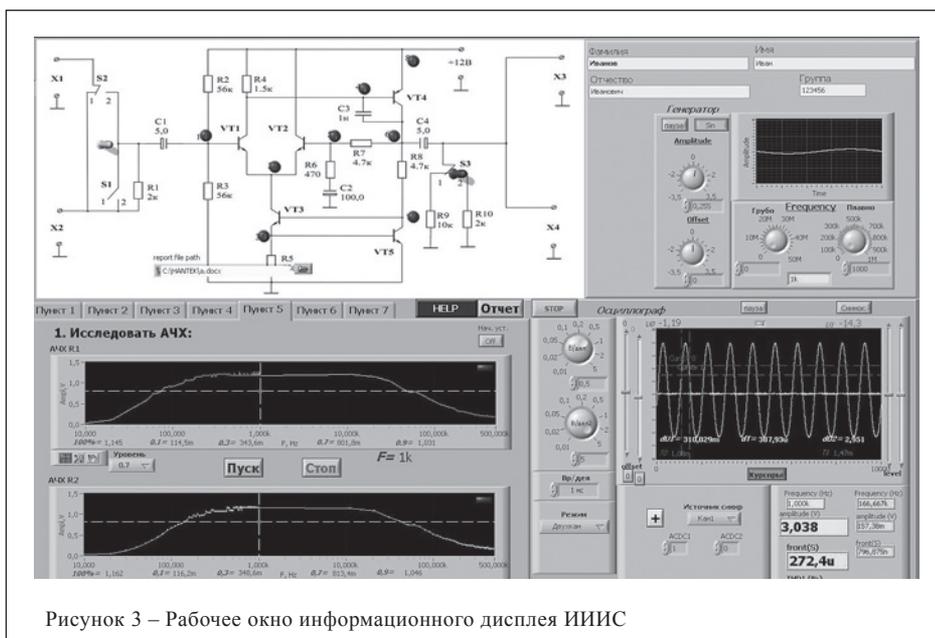


Рисунок 3 – Рабочее окно информационного дисплея ИИИС

Заключение

Разработанная интеллектуальная информационно-измерительная система представляет интерес при решении измерительных задач, требующих автоматизации процессов измерения, хранения, анализа измерительной информации, а также для реализации инновационных методов подготовки технических специалистов различного уровня на основе интеллектуальных аппаратно-программных средств в соответствии с современными образовательными программами.

Литература:

1. Афонский, А.А., Суханов, Е.В. LabVIEW в USB лаборатории. Контрольно-измерительные приборы и системы.
2. Раннев, Г.Г. Интеллектуальные средства измерений: учебник для студ. высш. учеб. заведений / Г.Г. Раннев. – М.: Издательский центр «Академия», 2010. – 272 с.

3. Крушев, В.Т. Лабораторный практикум по курсу Аналоговые электронные устройства для студентов спец. «Радиотехника», «Радиотехнические системы» и «Радиоинформатика» всех форм обучения / В.Т. Крушев, Э.Г. Попов. – Минск: БГУИР, 2004. – 58 с.

Abstract

The article describes an effective solution for the implementation of laboratory-based research in radioelectronics involving intelligent information-measuring system based on LabVIEW universal graphical programming environment. The proposed solution creates a set of hardware and software offering the best system operation mode selection, collection, processing and storage of measurement data with the subsequent creation of electronic reports.

Поступила в редакцию 19.03.2013 г.

НОВОСТИ**ST ВО ФРАНЦИИ ПРЕДЛАГАЕТ МЭМС НА МУЛЬТИПРОЕКТНЫХ ПЛАСТИНАХ**

STMicroelectronics стала партнером CMP (Circuits Multi Projects) во Франции для создания производственного процесса THELMA MEMS, который уже доступен для изготовления опытных образцов в университетах, исследовательских лабораториях и проектных компаниях с помощью технологии мультипроектных пластин.

Производство мультипроектных пластин компании CMP позволяет получать небольшие количества, как правило, от нескольких десятков до нескольких сотен экземпляров полупроводниковых устройств, изготовленных по тем же техпроцессам, что были бы использованы в крупносерийном производстве.

ST заявила, что она запускает этот техпроцесс для третьих лиц как опытное и foundry-производство, чтобы вдохновить на новые разработки в приложениях определения движения для потребительского, автомобильного, промышленного рынков и здравоохранения.

Поверхностный 0,8-мкм процесс микромашиных устройств THELMA (Thick Epitaxial Layer for Microgyroscopes and Accelerometers – толстый эпитакси-

альный слой для микрогироскопов и акселерометров) совмещает изменяемые толстые и тонкие слои для структур и соединений. Правила и инструменты проектирования для процесса THELMA доступны для университетов и компаний микроэлектроники по первому запросу.

Подложки FD-SOI (полностью обедненный КНИ)

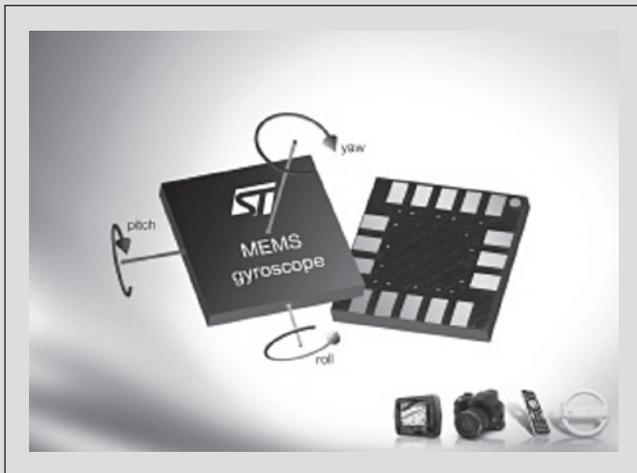
ST уже сотрудничает с CMP для того, чтобы предоставить университетам и проектным фирмам доступ к производственным полупроводниковым процессам ST – от 130-нм КМОП, введенном в 2003 году, до 28-нм FD-SOI, запущенном в опытное производство в конце 2012 года.

«Доступность мелкосерийного производства для нашего передового МЭМС-процесса вместе с КМОП-технологиями, включая FD-SOI, дополненного широкими производственными возможностями CMP, предлагает беспрецедентный доступ к современнейшему микроэлектронному производству для стартапов и исследовательских лабораторий, желающих проектировать интеллектуальные системы датчиков», – сказал Бенедетто Вигна (Benedetto Vigna), генеральный директор группы аналоговой электроники, МЭМС и датчиков в компании STMicroelectronics.

Перспективы интеллектуальных систем для «Интернета вещей» были представлены Бенедетто Вигна во вступительном докладе-приветствии на конференции DATE (Design, Automation & test in Europe) в Гренобле (Франция).

Напомним, что зеленоградский «Микрон» с середины прошлого года также присматривается к МЭМС-технологиям своего давнего технологического партнера ST, однако о каких-либо конкретных подвижках в этом направлении до сих пор официально не сообщалось.

ElectronicsWeekly.com



КОМПОНЕНТ УПРАВЛЕНИЯ ДОСТУПОМ СИСТЕМЫ УПРАВЛЕНИЯ ДОКУМЕНТАМИ

УДК 621.391

М.В. Стержанов, БГУИР, г. Минск

Аннотация

В статье рассматриваются вопросы построения компонента управления доступом в системе управления документами. Описывается общая функциональность системы, приводятся основные сущности модели базы данных системы, рассматривается процедура проверки прав доступа пользователя на элемент контента.

Введение

Одним из современных направлений развития системы конструкторско-проектных работ является интеграция информационных технологий в процесс документооборота. Актуальной является задача принципиально нового конструирования содержания и организации документации. Очевидно, что от механизмов доступа к документам и их хранения напрямую зависит производительность работы сотрудников организации (архитекторов, чертежников). Авторами разрабатывается система управления техническими документами (СУД) Stagirites, автоматизирующая процессы ввода, модификации, хранения и преобразования информации.

Любая система управления документами является многопользовательской системой. Она позволяет разделять редактирование контента между несколькими

пользователями с различным уровнем доступа. Одним из важнейших принципов разработки системы управления техническими документами (СУД) Stagirites является обеспечение защиты информации от несанкционированного доступа. Несанкционированный доступ к информации (НСД) – это доступ к информации, нарушающий правила разграничения доступа с использованием штатных средств, предоставляемых средствами вычислительной техники или автоматизированными системами. Защита от НСД – это предотвращение или существенное затруднение несанкционированного доступа [1].

Информационная безопасность системы обеспечивается за счет гибкого регулирования прав доступа к данным.

Общая функциональность системы

Интерфейс системы состоит из трех главных частей: каталога, панели инструментов и области просмотра и редактирования. Каталог предназначен для структуризации и краткого описания ресурсов, размещающихся в репозитории. Панель инструментов позволяет выполнять над содержимым системы операции преобразования и редактирования. Область просмотра и редакци-

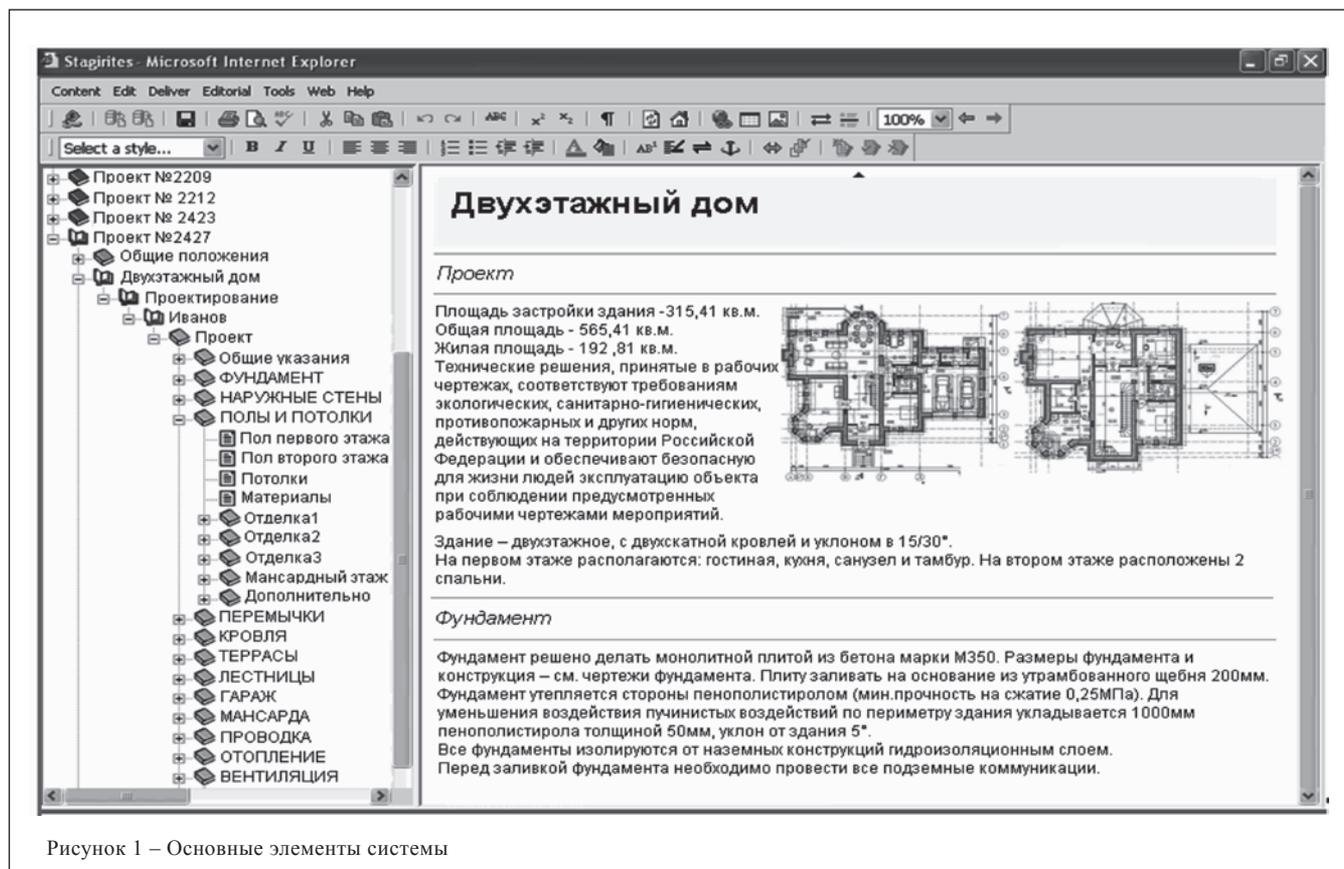
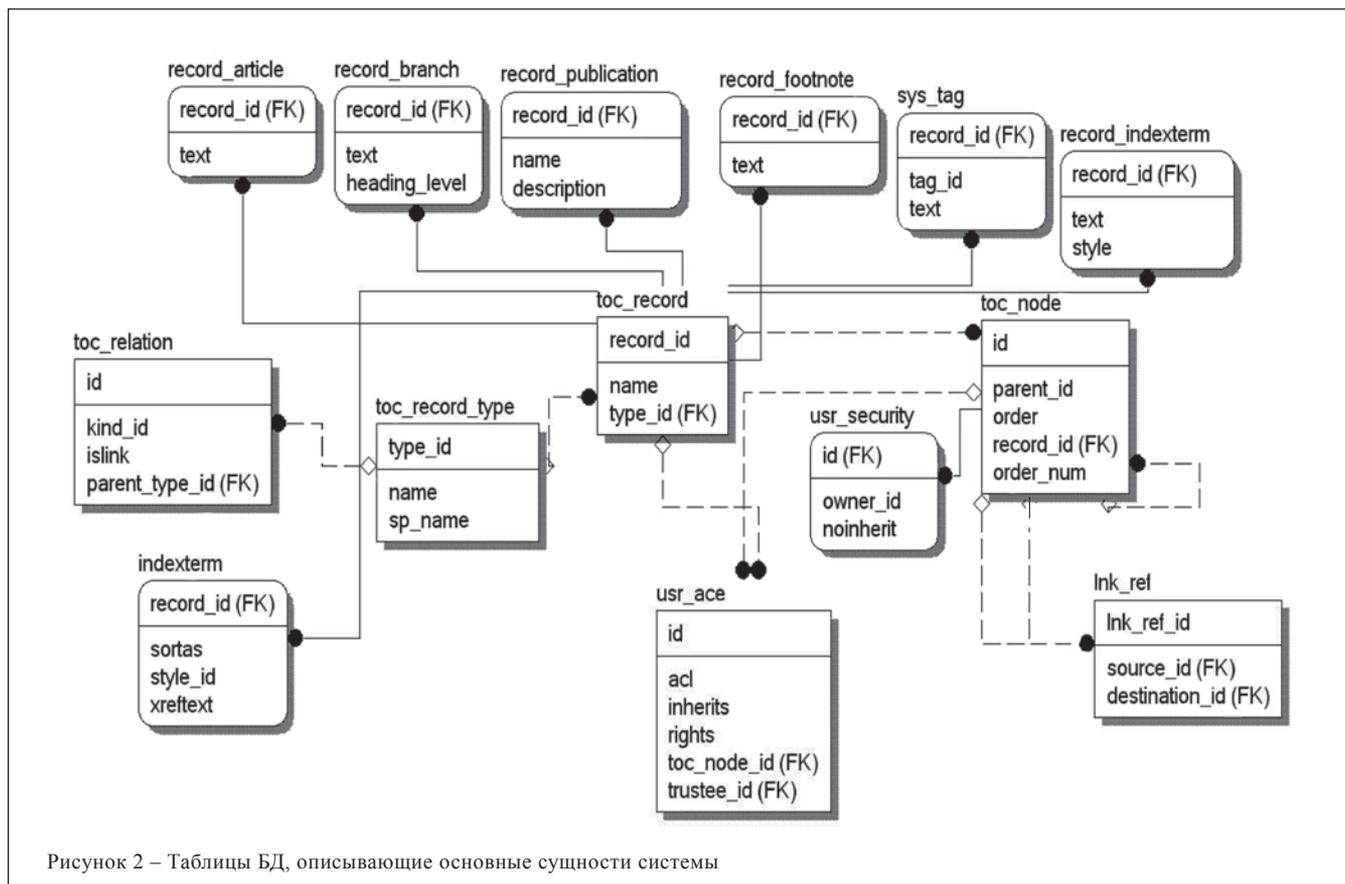


Рисунок 1 – Основные элементы системы



рования отображает текущие пять записей и позволяет вносить в текст активного документа изменения.

Система представляет контент в виде множества отдельных записей. Записи упорядочиваются в каталоге с помощью иерархической древовидной структуры, называемой деревом публикаций. Выбор такой структуры представления данных является не случайным. Одним из центральных принципов работы с информацией является принцип модульности. В соответствии с этим принципом контент структурируется в виде отдельных блоков (публикаций и заголовков). Текстовые записи бывают трех типов: публикация, заголовок, статья. Запись, соответствующая корневому узлу дерева, называется публикацией. Каждая публикация содержит набор логически сгруппированных документов и может представлять отдельный проект, дело, книгу и т.п. Некорневые записи-контейнеры называются заголовками. Заголовками являются структурные элементы проектной документации, описывающие отдельные разделы. Статьи являются листьями дерева публикаций и хранят в себе текст, таблицы, изображения. Каждая запись содержит следующую метainформацию: дату создания, имя создателя, дату последней модификации и имя пользователя, который осуществил последнюю модификацию записи. На рисунке 1 показано главное окно системы.

Дерево публикаций обеспечивает целостность представления информации. Дерево является гибкой структурой, содержание каждого проекта может легко дополняться или изменяться. Помимо очевидных опе-

раций над узлами (добавление, удаление, копирование, перемещение) реализована операция «повторного использования» контента, которая помогает ликвидировать дублирование информации и позволяет строить новые узлы контента по уже имеющимся. При этом информация репозитория разделяется между узлами дерева. Данный принцип аналогичен концепции символической ссылки в UNIX.

Основные сущности модели БД системы представлены на рисунке 2.

Для каждого объекта в системе существует запись в таблице `toc_record`, которая служит для хранения дескриптора объекта и имени записи, отображаемой в дереве публикаций. При помощи внешнего ключа на таблицу `toc_record_type` можно узнать тип записи (элемент дерева публикаций, системный тег) и определить таблицу, содержащую непосредственно данные записей этого типа. Данное отношение имеет вид «один к одному» и реализуется посредством миграции первичного ключа `id` таблицы `toc_record` в качестве внешнего первичного ключа в другие таблицы. Например, текст публикаций хранится в таблице `record_publication`, текст заголовков – в таблице `record_branch`, текст статей – в таблице `record_article` (контент хранится в формате XHTML).

Таблица `toc_node` служит для структуризации объектов системы и представления связей между записями таблицы `toc_record` в виде отношения «родитель-потомок». Иерархия узлов дерева публикаций хранится именно в этой таблице. Каждый узел имеет уникальный

идентификатор, связь с родителем (parent_id), а также порядковый номер, соответствующий позиции по отношению к смежным узлам.

Таблица toc_relation позволяет задать для каждого типа узла разрешенные типы дочерних объектов. Например, для публикации в качестве дочерних узлов могут выступать только заголовки и статьи. С помощью таблицы toc_filter можно разрешить отображение в дереве публикаций только записей определенного типа. Например, компонент, представляющий дерево публикаций, может быть использован для отображения не только структуры документов, но и конфигурационных параметров системы. Данные два представления должны быть доступны разным пользователям. Поэтому для каждого пользователя дерево будет строиться с применением необходимого фильтра.

Ограничение доступа к элементам контента

При разработке компонента ограничения доступа к элементам контента было решено руководствоваться следующими требованиями:

- быстроедействие. Проверка прав доступа должна осуществляться при выполнении каждой операции в системе, следовательно, эта операция должна требовать минимальное количество ресурсов;
- независимость. Необходимо минимизировать зависимости между компонентом прав доступа и другими объектами для того, чтобы компонент можно было отключить или при необходимости заменить;
- гибкость. Функциональность компонента должна быть расширяемой. Компонент должен поддерживать различные схемы прав для удовлетворения потребностей различных групп пользователей.

Права на элементы контента выдаются абстрактной сущности «Опекун», в качестве которой может выступать пользователь системы, группа, системная учетная запись. Общая информация об «Опекуне» представлена в таблице toc_record в виде имени и идентификатора объекта. Информация о конкретных типах «Опекунов» хранится в таблицах usr_account, usr_group, usr_system. Избирательное управление доступом реализовано с использованием списка контроля доступа. Таблица ACE (Access Control Entry) служит для указания прав «Опекуна» на элементы контента. Объект системы представляется идентификатором узла в дереве контента, «Опекун» представляется своим идентификатором (toc_record.record_id).

Помимо этих идентификаторов, каждая запись ACE содержит маску прав, тип, маску наследования. Маска прав представляет собой 32-битный массив, в котором:

- биты 0–15 служат для указания специальных прав, специфичных для каждого типа объекта;
- биты 16–23 слу-

жат для указания стандартных прав, единых для всех объектов;

- биты 24–27 являются зарезервированными;
- биты 28–31 служат для указания стандартных прав доступа: 28 – полный доступ, 29 – разрешение на выполнение, 30 – разрешение на запись, 31 – разрешение на чтение.

Тип записи ACE определяет отношение записи к списку разрешающих или запрещающих прав. Маска наследования является битовым массивом разрядности 32. На данный момент поддерживаются следующие значения маски наследования:

- 1. ACE наследуется для подконтейнеров;
- 2. ACE наследуется для подобъектов;
- 4. ACE наследуется только для непосредственных потомков;
- 8. ACE наследуется для подобъектов, однако, не применяется для данного объекта.

Каждый объект в системе имеет владельца и признак наследования прав от родителей. Эта информация задается в таблице usr_security. По умолчанию, если для объекта не задан владелец, то его владельцем является владелец родителя. Также по умолчанию объект наследует права доступа всех «Опекунов» своего родителя.

Опишем процедуру проверки прав доступа пользователя на элемент контента.

Шаг 1. Пользователь запрашивает требуемые права доступа для данного узла.

Шаг 2. Система выбирает список объектов ACE L, относящихся к данному узлу и применимых: к пользователю; к группам, в которые он входит; к системным аккаунтам, которым соответствует пользователь.

Шаг 3. На основании запрашиваемой маски доступа и списка L принимается решение о разрешении или запрете доступа:

- если в списке L имеется хотя бы один объект, запрещающий любое действие, указанное в запрашиваемой маске, то в доступе отказывается;
- если конъюнкция масок прав всех объектов списка L содержит все запрашиваемые права, то доступ разрешается.

Ограничение доступа к модулям системы

Помимо доступа к элементам контента требуется разграничить доступ пользователей к различным вспомогательным модулям, входящих в состав СУД (напри-



Рисунок 3 – Структура роли

мер: «Загрузка данных», «Преобразование данных», «Управление пользователями»). Все пользователи системы в ходе своей работы явно или неявно обращаются к различным данным. В качестве таких данных могут выступать элементы контента, конфигурации системы, а также файлы, используемые в процессе работы системы. Данные логически объединяются и хранятся в контейнере, называемом группой данных (ГД). Помимо обращения к данным, пользователь выполняет над данными различные операции. В качестве таких операций можно выделить: изменение содержания, удаление, создание новых объектов данных. Набор логически связанных операций называется группой операций (ГО). Для получения одновременного доступа к ГД и ГО введено понятие роли. Роль является совокупностью некоторых возможностей пользователя в системе. Обладание ролью дает право доступа ко всем объектам данных и операций, входящих в ГД и ГО, относящихся к этой роли.

При назначении роли определяется период действия, т.е. доступ к некоторому модулю системы устанавливается на определенный период времени (возможно, неограниченный).

Заключение

Описанный подход к разграничению прав в системе Stagirites позволяет управлять доступом к конкретным

элементам контента, что в совокупности с механизмом настройки ролей позволяет оперативно создавать специфические автоматизированные рабочие места без привлечения разработчиков системы. Механизм настройки ролей отображает полную картину разделения прав в системе, облегчает понимание и контроль сфер ответственности каждого пользователя, позволяет администратору гибко и прозрачно модифицировать права доступа.

Благодарности

Данная работа выполнялась при поддержке гранта Ф11-М210 Белорусского республиканского фонда фундаментальных исследований.

Литература:

1. Защита от несанкционированного доступа к информации. Термины и определения / Утверждено решением председателя Гостехкомиссии России от 30 марта 1992 г.

Abstract

Access control component for DMS is detailed. General functionality, DB design and rights check algorithm is discussed.

Поступила в редакцию 23.10.2012 г.



ПРОИЗВОДСТВО ПЕЧАТНЫХ ПЛАТ ОАО «МПОВТ»
основано в 1978 году.

Прогрессивное оборудование в сочетании с высоким уровнем технологии и накопленным производственным опытом позволяет изготавливать в серийном производстве по доступным ценам многослойные печатные платы с отличными техническими характеристиками.

Предприятие производит так же сверла и фрезы для обработки печатных плат и широкую гамму изделий из твердого сплава. Изделия и заготовки производятся с допусками, согласованными с заказчиком.

Если Вы нуждаетесь в высококачественных печатных платах, твердосплавном инструменте для их обработки, изделий из твердого сплава – производство печатных плат к Вашим услугам!

ОАО «МПОВТ»
Производство печатных плат

Республика Беларусь
220140, г. Минск,
Ул. Притыцкого, 62

Тел.: (+375 17) 253 65 04
(+375 17) 253 74 32
Факс: (+375 17) 253 60 75

e-mail: contact@mpovt.by
mpovt.by

Технические характеристики изготавливаемых печатных плат:

- Максимальные габариты плат: 460×510 мм;
- Количество слоев – 1÷22;
- Соотношение толщины платы к диаметру отверстия – 6:1;
- Минимальный диаметр металлизированного отверстия – 0,3 мм;
- Минимальная ширина проводников – 0,1 мм;
- Гальваническое покрытие КПК: Ni 6;
- Минимальное расстояние между проводниками – 0,12 мм;
- Покрытие под пайку: HALS;
- Защитная маска: жидкая сеткографическая, жидкая фотоформируемая;

- Обработка контура ПП: фрезерование (ЧПУ), вырубка, скрайбирование;
- Маркировка – сеткографическая.

Приемка плат осуществляется по ГОСТ 23752-79 и в соответствии с техническими требованиями заказчика. Возможно изготовление печатных плат с приемкой «5» На предприятии используются сертифицированные базовые материалы.

ГЕНЕРИРОВАНИЕ И ФУНКЦИОНАЛЬНО-СТРУКТУРНАЯ ОПТИМИЗАЦИЯ БАЗОВОГО КОМПЛЕКТА ТАБЛИЦ ДЛЯ МУЛЬТИПЛИКАТИВНОЙ МИМА-СХЕМЫ МОНТГОМЕРИ

УДК 681.3.06(082)

А.А. Коляда, А.Ф. Чернявский, Е.В. Шабинская, БГУ, г. Минск

Аннотация

Для базовых модулярных систем счисления (МСС) схемы Монтгомери для умножения по большим модулям разработаны методы и компьютерные алгоритмы генерирования комплекта таблиц. В частности, синтезированы процедуры формирования таблиц для суммирования вычетов по основаниям МСС аккумулятивно-табличным методом, а также таблиц для расчета интервально-индексных характеристик и расширения кодов. Представленный комплекс методологических и алгоритмических средств дает широкие возможности для разгрузки процесса умножения по большим модулям, осуществляемого в реальном времени, от расчетов, которые могут быть выполнены предварительно. Для генерируемых комплектов таблиц приведены оценки необходимой памяти.

Введение

В спектре многочисленных современных приложений модулярной вычислительной технологии (МВТ) разработки в области защиты информации занимают особое место. К разряду наиболее эффективных и успешных приложений данного класса, в первую очередь, следует отнести высокоскоростные реализации с применением арифметики МСС – модулярной арифметики (МА) процедур умножения и возведения в степень по большим модулям [1–5], составляющих, как известно, основу криптографических преобразований. Кодовый параллелизм МСС, обусловивший отсутствие межразрядных связей в арифметических операциях, дает модулярным вычислительным структурам гораздо большие возможности в сравнении с позиционными структурами для проведения быстрых, точных расчетов, особенно на диапазонах больших чисел.

Говоря о МВТ-приложениях, прежде всего, отметим, что подлежащие реализации в МСС целевые функции решаемых задач обычно сводятся к модели, которая состоит, главным образом, из модульных сегментов. При этом, используемый динамический диапазон должен включать конечные результаты счета на всех сегментах. Корректность такого режима вычислений достигается при помощи операций масштабирования. С обозначенной моделью идеально согласуется мультипликативная МА-схема Монтгомери. Применяемый в ней целочисленный масштаб требует предельно простой операции масштабирования частного вида – деления нацело, т.е. операции. Важнейшим фактором в данном случае является то обстоятельство, что для результата масштабирования – произведения Монтгомери всегда можно обеспечить принадлежность к диапазону операндов. Это позволяет осуществлять модульное возведение в степень без коррекции произведений. Отмеченные реализационные достоинства мультипликативной МА-схемы Монтгомери обусловили повышенный интерес к ней специалистов по криптографической защите информации.

Значительное упрощение алгоритмов умножения по большим модулям на основе модулярной схемы Монтгомери достигается при использовании минимально избыточной МА (МИМА) [5–7], которая отличается от неизбыточной МА существенно большей эффективностью немодульных процедур. В настоящей статье представлены методологические и алгоритмические средства табличной реализации МИМА-схемы типа Монтгомери для умножения по большим модулям, позволяющие минимизировать временные затраты на выполнение немодульной компоненты схемы.

1 Базовые обозначения и терминология

- 1.1. \mathbf{Z} – множество целых чисел (ЦЧ).
- 1.2. $\lfloor x \rfloor$ и $\lceil x \rceil$ – наибольшее и наименьшее ЦЧ, соответственно не большее и не меньше вещественной величины x .
- 1.3. $\mathbf{Z}_m^+ = \{0, 1, \dots, m-1\}$ и $\mathbf{Z}_m^- = \{-\lfloor m/2 \rfloor, -\lfloor m/2 \rfloor + 1, \dots, \lfloor m/2 \rfloor - 1\}$ – множества (кольца) наименьших неотрицательных и абсолютно наименьших вычетов по натуральному модулю $m > 1$ соответственно.
- 1.4. $|x|_m$ – элемент множества \mathbf{Z}_m^+ , сравнимый с x (в общем случае рациональной величины) по модулю m .
- 1.5. $\text{НОД}(m, n)$ – наибольший общий делитель натуральных чисел m и n .
- 1.6. $\varphi(m)$ – функция Эйлера, ставящая в соответствие модулю m число вычетов χ кольца \mathbf{Z}_m^+ взаимно простых с m (удовлетворяющих условию $\text{НОД}(\chi, m) = 1$).
- 1.7. $g = gm$ – первообразный корень по модулю m , определяемый как элемент множества Gm , такой, что $\text{НОД}(g, m) = 1$ и $\min\{\delta \mid |g^\delta|_m = 1 (\delta > 0)\} = \varphi(m)$.
- 1.8. $\text{ind}_g \chi$ – индекс вычета $\chi \in \mathbf{Z}_m^+ (\chi \neq 0)$ по модулю m и основанию g (g – первообразный корень), определяемый равенством $\chi = |g^{\text{ind}_g \chi}|_m$.
- 1.9. $Mn = \prod_{i=1}^n m_i$, $M_{i,n} = M_n / m_i$ ($i = \overline{1, n}$), где m_1, m_2, \dots, m_n – натуральные модули ($n \geq 1$).
- 1.10. $(|X|_{m_1}, |X|_{m_2}, \dots, |X|_{m_k})$ – код МСС с основаниями m_1, m_2, \dots, m_k ($k > 1$) или модулярный код (МК) ЦЧ X по базису $M = \{m_1, m_2, \dots, m_k\}$.
- 1.11. TInd_i и TAInd_i – таблицы индексов и антииндексов по модулю m_i , определяемые для $i = \overline{1, k}$ соответственно по правилам

$$\text{TInd}_i[\chi] = \begin{cases} -1 & \text{при } \chi = jm_i \quad (j = \overline{0, \lfloor m_{\text{max}}/m_i \rfloor}) \\ \text{ind}_g |\chi|_{m_i} & \text{при } \chi \in \mathbf{Z}_{m_{\text{max}}}^+ \text{ и } \chi \neq jm_i \end{cases}$$

$$\left\{ \begin{array}{l} \\ (j = \overline{0, \lfloor m_{\text{max}}/m_i \rfloor}) \end{array} \right\} \quad \text{и}$$

$$\text{TAInd}_i[j] = |g^{\lfloor m_{\text{max}}/m_i - j \rfloor}|_{m_i} \quad (j = \overline{0, 2m_i - 3}),$$

m_{max} – верхний ограничительный порог для оснований МСС.

1.12. GTMpl_Const_Mod(m, n, c, TM) – процедура, осуществляющая формирование таблицы

$$TM[\chi] = |c\chi|_m \text{ (} c \in \{1, 2, \dots, m-1\}, \chi = \overline{0, n-1} \text{),}$$

где m и n – некоторые модули [8].

1.13. $EC(X; M_1, M_2)$ и $EC(X; M_2, M_1)$ – операции расширения МК числа X по базисам $M_1 = \{m_1, m_2, \dots, m_l\}$ и $M_2 = \{m_{l+1}, m_{l+2}, \dots, m_k\}$ ($1 < l < k$) на основании базисов M_2 и M_1 соответственно [8].

1.14. p – рабочий модуль (большое ЦЧ) для мультипликативных операций.

2 Применение индексного метода модульного умножения для расчета констант мультипликативной МИМА-схемы Монтгомери

Пусть A и B – операнды, подлежащей выполнению операции умножения по некоторому большому модулю p , и пусть A, B, p заданы кодами МСС с базисом M :

$$A = (\alpha_1, \alpha_2, \dots, \alpha_k), \quad B = (\beta_1, \beta_2, \dots, \beta_k), \quad p = (\pi_1, \pi_2, \dots, \pi_k)$$

$$(\alpha_i = |A|_{m_i}, \beta_i = |B|_{m_i}, \pi_i = |p|_{m_i} \text{ (} i = \overline{1, k} \text{)})$$

По методу Монтгомери в качестве искомого произведения операндов A и B принимается ЦЧ $\hat{Y} = |\hat{\gamma}|_p = |ABM_i^{-1}|_p$. При этом, как показано в [5–7], результирующая мультипликативная схема Монтгомери, базирующаяся на МИМА, сводится к операционной последовательности:

$$\langle C = AB = (\gamma_1, \gamma_2, \dots, \gamma_k) \text{ (} \gamma_i = |\alpha_i \beta_i|_{m_i} \text{ (} i = \overline{1, k} \text{))};$$

$$D = |CF|_{M_i} = (\delta_1, \delta_2, \dots, \delta_l) \text{ (} F = |-p^{-1}|_{M_i} =$$

$$= (\varphi_1, \varphi_2, \dots, \varphi_l) \text{ (} \varphi_i = |-\pi_i^{-1}|_{m_i} \text{ (} i = \overline{1, l} \text{))}; (\hat{\delta}_{l+1}, \hat{\delta}_{l+2}, \dots, \hat{\delta}_k) =$$

$$= EC(\hat{D}; M_1, M_2);$$

$$\hat{\gamma} = \hat{C} / M_l = (\hat{\gamma}_{l+1}, \hat{\gamma}_{l+2}, \dots, \hat{\gamma}_k)$$

$$(\hat{\gamma}_j = |(\gamma_j + |\hat{\delta}_j \pi_j|_{m_j}) M_l^{-1}|_{m_j} \text{ (} j = \overline{l+1, k} \text{))};$$

$$(\hat{\gamma}_1, \hat{\gamma}_2, \dots, \hat{\gamma}_l) = EC(\hat{\gamma}; M_2, M_1; \tilde{\gamma} = |\hat{\gamma}|_p \rangle. \quad (1)$$

Фигурирующие в (1) процедуры расширения кодов – $EC(\hat{D}; M_1, M_2)$ и $EC(\hat{\gamma}; M_2, M_1)$ реализуются по расчетным соотношениям:

$$\hat{\delta}_j = \left| \sum_{i=1}^{l-1} M_{i,l-1} |M_{i,l-1}^{-1} \delta_i|_{m_i} |_{m_j} + \right. \\ \left. + |M_{l-1} \hat{I}_l(D)|_{m_j} |_{m_j} \text{ (} j = \overline{l+1, k} \text{)}; \quad (2)$$

$$\hat{I}_l(D) = \left| \sum_{i=1}^l R_{i,l}(\delta_i) \right|_{m_l}; \quad (3)$$

$$R_{i,l}(\delta_i) = \left| -m_i^{-1} |M_{i,l-1}^{-1} \delta_i|_{m_i} \right|_{m_l} \text{ (} i = \overline{1, l-1} \text{),} \\ R_{i,l}(\delta_l) = |M_{i,l-1}^{-1} \delta_l|_{m_l}; \quad (4)$$

$$\hat{\gamma}_j = \left| \sum_{i=l+1}^{k-1} \left| \frac{M_{k-1}}{M_i m_i} \left| \frac{M_l m_l \hat{\gamma}_i}{M_{k-1}} \right|_{m_i} \right|_{m_j} + \right. \\ \left. + \left| \frac{M_{k-1}}{M_l} I'_k(\hat{\gamma}) \right|_{m_j} \right|_{m_j} \text{ (} j = \overline{1, l} \text{)}; \quad (5)$$

$$I'_k(\hat{\gamma}) = \begin{cases} \hat{I}'_k(\hat{\gamma}), & \text{если } \hat{I}'_k(\hat{\gamma}) < m_0, \\ \hat{I}'_k(\hat{\gamma}) - m_k, & \text{если } \hat{I}'_k(\hat{\gamma}) \geq m_k - m_0 - k + l + 2; \end{cases} \quad (6)$$

$$(mk \geq 2m_0 + k - l - 2, m_0 \geq k - l - 2);$$

$$\hat{I}'_k(\hat{\gamma}) = |I'_k(\hat{\gamma})|_{m_k} = \left| \sum_{i=l+1}^k R'_{i,k}(\hat{\gamma}_i) \right|_{m_k}; \quad (7)$$

$$R'_{i,k}(\hat{\gamma}_i) = \left| -m_i^{-1} \left| \frac{M_l m_l \hat{\gamma}_i}{M_{k-1}} \right|_{m_i} \right|_{m_k} \text{ (} i = \overline{l+1, k-1} \text{),}$$

$$R'_{k,k}(\hat{\gamma}_k) = \left| \frac{M_l}{M_{k-1}} \hat{\gamma}_k \right|_{m_k}. \quad (8)$$

Входящие в состав интервально-модулярной формы (ИМФ) (2), (5) и, как следствие, в расчетные соотношения (3), (4), (6)–(8) для интервально-индексных характеристик и расширения МК коэффициенты являются произведениями оснований МСС и/или содержат мультипликативные инверсии ЦЧ по модулям. Это относится и к другим операциям схемы (1). В свете сказанного, свойства рассмотренного в [8] индексного метода модульного умножения для таблично-сумматорной технологии синтеза компьютерных МИМА-процедур Монтгомери на основе мультипликативной схемы (1) представляют особую важность. Особенности применения индексного метода модульного умножения ниже демонстрируются на примере алгоритма расчета коэффициентов ИМФ ЦЧ в форме, адаптированной к базовым немодульным процедурам мультипликативной МИМА-схемы (1). Синтезированный алгоритм состоит в нижеследующем.

Входные данные алгоритма: основания m_1, m_2, \dots, m_k МСС.

Выходные данные: коэффициенты ИМФ, отвечающие базисам $\{m_1, m_2, \dots, m_l\}$ и $\{m_{l+1}, m_{l+2}, \dots, m_k\}$ усеченных МСС, а также МК ($|M_{i,l-1}^{-1}|_{m_{i+1}}, |M_{i,l-1}^{-1}|_{m_{i+2}}, \dots, |M_{i,l-1}^{-1}|_{m_k}$) масштабирующего множителя M_i^{-1} для схемы (1).

Предварительно полученные данные: таблицы TIndi индексов и TAIndi антииндексов по модулям m_i ($i = \overline{1, k}$) (см. 1.11).

Тело алгоритма

Расчет коэффициентов ИМФ нижнего (внутреннего) уровня для первой усеченной МСС.

ПК.1. Переменной i цикла присвоить начальное значение $i=1$.

ПК.2. Вычислить сумму $s = \sum_{\substack{j=1 \\ j \neq i}}^{l-1} TIndi[m_j]$.

ПК.3. Получить константу

$$C_i = |M_{i,l-1}^{-1}|_{m_i} = TAIndi[m_i - 1 - |s|_{m_{i-1}}].$$

ПК.4. Если $i \neq l-1$, то i увеличить на 1 ($i \neq l+1$) и перейти к ПК.2.

ПК.5. Найти $s = \sum_{j=1}^{l-1} TIndl[m_j]$.

PK.6. Определить константу

$$C_l = |M_{l-1}^{-1}|_{m_l} = \text{TAInd}[m_l^{-1} - |s|_{m_{l-1}}].$$

Расчет коэффициентов ИМФ верхнего (внешнего) уровня для первой усеченной МСС.

PK.7. Положить $j=l+1$.

PK.8. Вычислить $s = \sum_{j=1}^{l-1} \text{TInd}[m_j]$.

PK.9. Найти индексы

$$c_{l,j} = \text{ind}(|M_{l-1}|_{m_j}) = |s|_{m_{j-1}}$$

и

$$c_{i,j} = \text{ind}(|M_{i,l-1}|_{m_j}) = |c_{l,j} - \text{TInd}[m_i]|_{m_{j-1}} \quad (i = \overline{1, l-1}).$$

PK.10. Определить масштабирующий множитель для схемы (1) по модулю m_j :

$$|M_l^{-1}|_{m_j} = \text{TAInd}[m_j - c_{l,j} + m_j - \text{TInd}[m_l] - 2].$$

PK.11. Если $j \neq k$, то j увеличить на 1 ($j=j+1$) и перейти к PK.8.

Расчет коэффициентов ИМФ нижнего уровня (коэффициентов нормировки цифр МК), отвечающих базису второй усеченной МСС.

PK.12. Переменной i цикла присвоить значение $i=l+1$.

PK.13. Получить сумму $s = \sum_{\substack{j=l+1 \\ j \neq i}}^{k-1} \text{TInd}[m_j]$.

PK.14. Определить константу

$$C'_i = |M_l m_i / M_{k-1}|_{m_i} = \text{TAInd}[m_i - 1 - |s|_{m_{i-1}}].$$

PK.15. Если $i \neq k-1$, то увеличить i на 1 ($i=i+1$) и перейти к PK.13.

PK.16. Вычислить сумму $s = \sum_{\substack{j=l+1 \\ j \neq i}}^{k-1} \text{TInd}[m_j]$.

и получить коэффициент

$$C'_k = |M_l / M_{k-1}|_{m_k} = \text{TAInd}[m_k - 1 - |s|_{m_{k-1}}]$$

Расчет коэффициентов ИМФ верхнего уровня, отвечающих базису второй усеченной МСС.

PK.17. Положить $j=1$.

PK.18. Определить $s = \sum_{i=l+1}^{k-1} \text{TInd}[m_i]$

и найти индексы:

$$c'_{k,j} = \text{ind}(|M_{k-1} / M_l|_{m_j}) = |s|_{m_{j-1}},$$

$$c'_{i,j} = \text{ind}(|M_{k-1} / (M_l m_i)|_{m_j}) = |c'_{k,j} - \text{TInd}[m_i]|_{m_{j-1}} \quad (i = \overline{l+1, k-1}).$$

PK.19. Если $j \neq l$, то увеличить j на 1 ($j=j+1$) и перейти к PK.18. По достижении равенства $j=l$ завершить работу алгоритма.

3 Генерирование базового комплекта таблиц для мультипликативной МИМА-схемы Монтгомери

Таблично-сумматорная технология синтеза и компьютерной реализации мультипликативных процедур по большому модулю p , базирующихся на МИМА-схеме Монтгомери (1) предполагает использование не только таблиц TInd_i , TAInd_i , TRes_UP_i , TRes_i ($i = \overline{1, k}$) умножения и свертки двоичных кодов (приведения к остаткам) ЦЧ по основаниям m_i МСС [8], но и таблиц, которые предназначены для расчета интервально-индексных характеристик и расширения кодов. Из расчетных соотношений (2), (3), (4), (5), (7), (8) этих операций видно, что ключевая роль в процедурах генерирования базового комплекта таблиц (КТ) должна быть отведена таблицам TRes_Norm_i ($i = \overline{1, l-1}$) и _TRes_Norm_i ($i = \overline{l+1, k-1}$) нормированных остатков по модулям m_i соответственно базисов $\{m_1, m_2, \dots, m_l\}$ и $\{m_{l+1}, m_{l+2}, \dots, m_k\}$. Указанные таблицы формируются по правилам:

$$\text{TRes_Norm}_i[\chi] = \left| |M_{i,l-1}^{-1}|_{m_i} \chi \right|_{m_i} \quad (\chi = \overline{0, m_i - 1}; i = \overline{1, l-1}); \quad (9)$$

$$\text{_TRes_Norm}_i[\chi] = \left| \frac{|M_l m_i|}{|M_{k-1}|_{m_i}} \chi \right|_{m_i} \quad (\chi = \overline{0, m_i - 1}; i = \overline{l+1, k-1}). \quad (10)$$

Операция нормировки применяется также и при получении согласно (1), МК ($\hat{Y}_{l+1}, \hat{Y}_{l+2}, \dots, \hat{Y}_k$) ЦЧ \hat{Y} . Для табулирования цифр данного МК будем использовать таблицы.

$$\text{_TMpli}[x] = \left| |M_l^{-1}|_{m_i} x \right|_{m_i} \quad (x = \overline{0, 2m_i - 2}; i = \overline{l+1, k}). \quad (11)$$

Генерирование таблиц (9)–(11) осуществляется процедурами $\text{GTmpl_Const_Mod}(m_p, m_p, |M_{i,l-1}^{-1}|_{m_i}, \text{TRes_Norm}_i)$ ($i = \overline{1, l-1}$), $\text{GTmpl_Const_Mod}(m_p, m_p, |M_l m_i / M_{k-1}|_{m_i}, \text{_TRes_Norm}_i)$ ($i = \overline{l+1, k-1}$), $\text{GTmpl_Const_Mod}(m_p, 2m_i - 1, |M_l^{-1}|_{m_i}, \text{_TMpli})$ ($i = \overline{l+1, k}$). Требуемые для указанных процедур системные константы: $|M_{i,l-1}^{-1}|_{m_i}$, $|M_l m_i / M_{k-1}|_{m_i}$ и $|M_l^{-1}|_{m_i}$ вычисляются в рамках алгоритма PK.1–PK.19.

Таблицы (9) и (10) нормированных цифр МК выполняют вспомогательную роль. Они служат основой

для генерирования таблиц интервальных индексов (ИИ) и расширения кодов. В таблицах ИИ (присвоим им имена $\text{ТП}i$ ($i = 1, l$), $\text{ТП}i$ ($i = l + 1, k$)) хранятся вычеты типа (4) и (8), используемые для вычисления компьютерных ИИ ЦЧ в первой и второй усеченных МСС. Таблицы расширения кодов $\text{TE}i_j$ ($i = 1, l$); $j = l + 1, k$) и $\text{TE}i_j$ ($i = l + 1, k$; $j = 1, l$) предназначены для хранения слагаемых расчетных соотношений (2) и (5). Ниже приводятся алгоритмы генерирования таблиц ИИ и расширения кодов.

Алгоритм генерирования таблиц ИИ

Входные данные алгоритма: основания m_1, m_2, \dots, m_k МСС.

Выходные данные: таблицы ИИ, которые в соответствии с (4) и (8) формируются по правилам:

$$\begin{aligned} \text{ТП}i[\chi] &= R_{i,l}(\chi) = |-m_i^{-1} | M_{i,l-1}^{-1} \chi |_{m_i} |_{m_l} = \\ &= |-m_i^{-1} |_{m_l} \text{TRes_Norm}i[\chi] |_{m_i} \\ &(\chi = \overline{0, m_i - 1}; i = \overline{1, l - 1}); \end{aligned} \tag{12}$$

$$\begin{aligned} \text{ТП}l[\chi] &= R_{l,l}(\chi) = |-M_{l-1}^{-1} |_{m_l} \chi |_{m_l} \\ &(\chi = \overline{0, m_l - 1}); \end{aligned} \tag{13}$$

$$\begin{aligned} \text{ТП}i[\chi] &= R'_{i,k}(\chi) = |-m_i^{-1} | (M_l m_i / M_{k-1}) \chi |_{m_i} |_{m_k} = \\ &= |-m_i^{-1} |_{m_k} \text{TRes_Norm}i[\chi] |_{m_i} \\ &(\chi = \overline{0, m_i - 1}; i = \overline{l + 1, k - 1}); \end{aligned} \tag{14}$$

$$\begin{aligned} \text{ТП}k[\chi] &= R'_{k,k}(\chi) = | M_l / M_{k-1} |_{m_k} \chi |_{m_k} \\ &(\chi = \overline{0, m_k - 1}); \end{aligned} \tag{15}$$

Предварительно полученные данные: таблицы нормированных остатков $\text{TRes_Norm}i$ ($i = \overline{1, l - 1}$), $\text{TRes_Norm}i$ ($i = \overline{l + 1, k - 1}$), определяемые по формулам (9), (10), таблицы индексов и антииндексов $\text{TInd}l$, $\text{TInd}k$, $\text{TAInd}l$, $\text{TAInd}k$, формируемые в рамках алгоритма ГИА.1–ГИА.11 и системные константы: $| M_{l-1}^{-1} |_{m_l}$, $| M_l / M_{k-1} |_{m_k}$, используемые в (13), (15) и вычисляемые процедурой РК.1–РК.19.

Тело алгоритма

Генерирование таблиц ИИ, отвечающих базису $\{m_1, m_2, \dots, m_j\}$

ГТИИ.1. Переменной i цикла присвоить начальное значение $i=1$.

ГТИИ.2. Найти индекс i_l по модулю m_l величины

$$\begin{aligned} |-m_i^{-1} |_{m_l} &= |(m_l - m_i)^{-1} |_{m_l} \text{ по формуле} \\ i_l &= m_l - 1 - \text{TInd}l[m_l - m_i] \text{ (см. (44)).} \end{aligned}$$

ГТИИ.3. В таблицу $\text{ТП}i$ записать нулевой элемент: $\text{ТП}i[0]=0$.

ГТИИ.4. Для $\chi = 1, m_i - 1$ выполнить операции $\text{ТП}i[\chi] = \text{TAInd}l[i_+ + \text{TInd}l[\text{TRes_Norm}i[\chi]]]$.

ГТИИ.5. Если $i \neq l - 1$, то инкрементировать i ($i=i+1$) и перейти к ГТИИ.2.

ГТИИ.6. Согласно (13) с помощью процедуры $\text{GTmpl_Const_Mod}(m_p, m_p, | M_{l-1}^{-1} |_{m_l}, \text{ТП}l)$, реализуемой по алгоритму ТК.1–ТК.5, сформировать искомое содержимое таблицы $\text{ТП}l$.

Генерирование таблиц ИИ, отвечающих базису $\{m_{l+2}, \dots, m_k\}$

ГТИИ.7. Переменной i цикла присвоить начальное значение $i=l+1$.

ГТИИ.8. Получить индекс $i_k = m_k - 1 - \text{TInd}k[m_k - m_i]$ вычета $|-m_i^{-1} |_{m_k}$ по модулю m_k .

ГТИИ.9. Положить $\text{ТП}i[0]=0$.

ГТИИ.10. Для всех $\chi = 1, m_i - 1$ выполнить $\text{ТП}i[\chi] = \text{TAInd}k[i_k + \text{TInd}k[\text{TRes_Norm}i[\chi]]]$.

ГТИИ.11. Если $i \neq k - 1$, то увеличить i на 1 ($i=i+1$) и перейти к шагу ГТИИ.8.

ГТИИ.12. В соответствии с (15) при помощи процедуры $\text{GTmpl_Const_Mod}(m_k, m_k, | M_l / M_{k-1} |_{m_k}, \text{ТП}k)$ сформировать содержимое таблицы $\text{ТП}k$ и завершить работу алгоритма.

Алгоритм генерирования таблиц расширения кодов

Входные данные алгоритма: основания m_1, m_2, \dots, m_k , а также вспомогательный модуль m_0 , удовлетворяющий условиям $m_0 \geq k - 1 - 2 = l - 2, m_k \geq 2m_0 + l - 2$ и вычисляемый по формуле $m_0 = \lfloor (m_k - (k - l) + 2) / 2 \rfloor = \lfloor (m_k - l + 2) / 2 \rfloor$.

Выходные данные: таблицы расширения кодов $\text{TE}i_j$ и $\text{TE}i_j$, которые в соответствии с (2), (5), (6), (9), (10), формируются по правилам:

$$\begin{aligned} \text{TE}i_j[\chi] &= | M_{i,l-1} | M_{i,l-1}^{-1} \chi |_{m_i} |_{m_j} = | C_{i,j} \text{TRes_Norm}i[\chi] |_{m_j} \\ (C_{i,j} &= | M_{i,l-1} |_{m_j}; \chi = \overline{0, m_i - 1}; i = \overline{1, l - 1}; j = \overline{l + 1, k}); \end{aligned} \tag{16}$$

$$\begin{aligned} \text{TE}l_j[\chi] &= | M_{l-1} \chi |_{m_l} + | C_{l,j} \chi |_{m_j} \\ (C_{l,j} &= | M_{l-1} |_{m_j}; \chi = \overline{0, m_l - 1}; j = \overline{l + 1, k}); \end{aligned} \tag{17}$$

$$\begin{aligned} \text{TE}i_j[\chi] &= \left| \frac{M_{k-1}}{M_l m_i} \left| \frac{M_l m_i}{M_{k-1}} \chi \right|_{m_i} \right|_{m_j} = \\ &= | C'_{i,j} \text{TRes_Norm}i[\chi] |_{m_j} \end{aligned}$$

$$(C'_{i,j} = \left| \frac{M_{k-1}}{M_l m_i} \right|_{m_j}; \chi = \overline{0, m_i - 1}; i = \overline{l + 1, k - 1}; j = \overline{1, l}); \tag{18}$$

$$\begin{aligned} \text{TE}k_j[\chi] &= \begin{cases} | C'_{k,j} \chi |_{m_j}, & \text{если } \chi < m_0, \\ | C'_{k,j} (\chi - m_k) |_{m_j}, & \text{если } \chi \geq m_0, \end{cases} \\ (C'_{k,j} &= | M_{k-1} / M_l |_{m_j}; \chi = \overline{0, m_k - 1}; j = \overline{1, l}); \end{aligned} \tag{19}$$

В рамках рассматриваемого алгоритма соотношения (16)–(19) реализуются с помощью индексного метода модульного умножения.

Предварительно полученные данные: таблицы индексов и антииндексов $\text{TInd}i$, $\text{TAInd}i$ ($i = 1, k$), рассчитываемые согласно п. 1.11, таблицы нормированных

остатков $TRes_Norm_i (i = \overline{1, l-1})$, $_TRes_Norm_i (i = \overline{l+1, k-1})$, определяемые по формулам (9), (10), а также системные константы $c_{i,j} = \text{ind } C_{i,j} (i = \overline{1, l-1}; j = \overline{l+1, k})$ и $c'_{i,j} = \text{ind } C'_{i,j} (i = \overline{l+1, k-1}; j = \overline{1, l})$ вычетов $C_{i,j}$ и $C'_{i,j}$ по модулям m_j из (16)–(19), вычисляемые в рамках алгоритма РК.1–РК.19 (см. РК.9, РК.18).

Тело алгоритма генерирования таблиц расширения кодов

Генерирование таблиц расширения кода ЦЧ по основаниям первой усеченной МСС на основании базиса второй усеченной МСС

ГТР.1. Переменной j цикла присвоить начальное значение $j=l+1$.

ГТР.2. Переменной i цикла присвоить значение $i=1$.

ГТР.3. Положить $TE_{i,j}[0]=0$ и для $\chi = 1, m_j - 1$ выполнить операции: $\tilde{\chi} = TRes_Norm_i[\chi]$,

$$TE_{i,j}[\chi] = \begin{cases} 0, & \text{если } \tilde{\chi} = m_j, \\ TA_{\text{ind}j}[c_{i,j} + T_{\text{ind}j}[\tilde{\chi}]], & \text{если } \tilde{\chi} \neq m_j. \end{cases}$$

ГТР.4. Если $i \neq l-1$, то увеличить i на 1 ($i=i+1$) и перейти к ГТР.3.

ГТР.5. В таблицу $TE_{i,j}$ поместить набор вычетов по модулю m_j согласно правилам: $TE_{i,j}[0]=0$, $TE_{i,j}[\chi]=0$, $TA_{\text{ind}j}[c_{i,j} + T_{\text{ind}j}[\chi]] (\chi = \overline{1, m_j - 1})$, причем в случае $m_j < m_i$ осуществить коррекцию: $TE_{i,j}[m_j]=0$.

ГТР.6. Усли $j \neq k$, то инкрементировать $j (j=j+1)$ и перейти к ГТР.2.

Генерирование таблиц расширения кода МИМСС с основаниями $m_{l+1}, m_{l+2}, \dots, m_k$ на модули m_1, m_2, \dots, m_l

ГТР.7. Переменной j присвоить начальное значение $j=1$.

ГТР.8. Положить $i=l+1$.

ГТР.9. В таблицу $_TE_{i,j}$ поместить нулевой элемент: $_TE_{i,j}[0]=0$ и для всех $\chi = 1, m_i - 1$ реализовать операции:

$$\tilde{\chi} = _TRes_Norm_i[\chi],$$

$$_TE_{i,j}[\chi] = \begin{cases} 0 & \text{при } \tilde{\chi} = m_j, \\ TA_{\text{ind}j}[c'_{i,j} + T_{\text{ind}j}[\tilde{\chi}]], & \text{при } \tilde{\chi} \neq m_j. \end{cases}$$

ГТР.10. Если $i \neq k-1$, то i увеличить на 1 ($i=i+1$) и перейти к ГТР.9.

ГТР.11. В соответствии с (19) сформировать искомое содержимое таблицы $TE_{k,j}$, выполняя последовательность действий:

а) положить $_TE_{k,j}[0]=0$, $_TE_{k,j}[\chi]=TA_{\text{ind}j}[c'_{k,j} + T_{\text{ind}j}[\chi]] (\chi = \overline{1, m_0 - 1})$;

б) переменной $\chi=m_0$;

в) получить величину $x = m_j - m_k + \chi$;

г) если $x < 0$, то увеличить x на $m_j (x=x+m_j)$;

д) в таблицу $_TE_{k,j}$ записать очередной элемент $_TE_{k,j}[\chi]=TA_{\text{ind}j}[c'_{k,j} + T_{\text{ind}j}[x]]$;

е) при $\chi \neq m_k - 1$ инкрементировать $\chi (\chi=\chi+1)$ и перейти к б).

ГТР.12. Если $j \neq l$, то увеличить j на 1 ($j=j+1$) и перейти к ГТР.8. По достижении равенства $j=l$ завершить работу алгоритма.

Представленный комплекс методологических и алгоритмических средств, составляющий базовое математическое обеспечение таблично-сумматорной технологии реализации мультипликативной МИМА-схемы Монтгомери (1), дает широкие возможности для разгрузки процесса умножения по большому модулю p , осуществляемого в реальном времени от расчетов, которые могут быть выполнены предварительно. Это обусловлено тем, что генерирование таблиц происходит именно на этапе предварительных вычислений. В рамках таблично-сумматорной реализационной технологии наибольшая производительность, естественно, достигается при использовании максимального базового КТ:

$$\begin{aligned} CT_MAX = & \langle T_{\text{ind}l} - T_{\text{ind}k}; TA_{\text{ind}l} - \\ & - TA_{\text{ind}k}; TRes_UP1 - TRes_UPk; \\ & TRes1 - TResk; T_{\text{ind}i}(i = \overline{1, l}); _T_{\text{ind}i}(i = \overline{l+1, k}); \\ & TE_{i,j}(i = \overline{1, l}; j = \overline{l+1, k}); \\ & _TE_{i,j}(i = \overline{l+1, k}; j = \overline{1, l}); \\ & _TM_{\text{pli}} i = \overline{l+1, k} \rangle. \end{aligned} \quad (20)$$

Отметим, что в комплект (20) не включены таблицы $TRes_Norm_i(i = \overline{1, l-1})$ и $_TRes_Norm_i(i = \overline{l+1, k-1})$ нормированных цифр МК. Эти таблицы выполняют вспомогательную роль. Они используются при формировании таблиц ИИ и расширения кодов (см. (12), (14), (16), (18)) и после реализации процедур ГТИИ.1–ГТИИ.12, ГТР.1–ГТР.12 могут быть удалены из оперативной памяти.

Суммарное количество таблиц максимального КТ (20) составляет

$$N_{T, \text{max}} = 2l_l + 5k + 1. \quad (21)$$

на i -е основание приходится

$$N_{T, \text{max}, i} = \begin{cases} l + 4 & \text{при } i = \overline{1, l-1}, \\ k + 4 & \text{при } i = l, \\ l + 5 & \text{при } i = \overline{l+1, k-1}, \\ k + 5 & \text{при } i = k \end{cases}$$

таблиц, которые содержат

$$\begin{aligned} N_{ЭТ, \text{max}, i} = & E_0 + E_1 + m_max + 3m_i - 2 + \\ & \left\{ \begin{aligned} & \sum_{j=l+1}^k m_j & \text{при } i = \overline{1, l-1}, \\ & \sum_{j=1}^k m_j & \text{при } i = l, \\ & (2m_i - 1) + \sum_{j=1}^l m_j & \text{при } i = \overline{l+1, k-1}, \\ & (2m_k - 1) + \sum_{j=1}^k m_j & \text{при } i = k \end{aligned} \right. \end{aligned} \quad (22)$$

элементов разрядностью $bi = \lceil \log_2 m_i \rceil$ бит. Напомним, что фигурирующие в (22) параметры $E_0=2^{b_0}$ и $E_1=2^{b_1}$ представляют собой мощности соответственно диапазонов

изменения (p₀)- и (p₁)-битовых частей (младших и старших), накапливаемых согласно аккумулятивно-табличному методу сумм вычетов из кольца Z_{m_j} ; $b_{-0} + b_{-1} = b_{-}$; $E_{-0} \leq m_j$. Как показано в разделе 4.3, разрядность b_{-} вычисляемых модульных сумм практически не превышает 24 бита. Исходя из этого, положим $b_{-0} = b_{-1} = 12$ бит.

Из (22) следует, что максимальный КТ (20) в общей сложности занимает память объемом

$$M_{T,max} = \frac{1}{8} \sum_{i=1}^k N_{ЭТ,max,i} b_i = \frac{1}{8} \left(\sum_{i=1}^k (2^{13} + m_{-max} + 3m_i - 2)b_i + \sum_{i=l+1}^k (2m_i - 1)b_i + \left(\sum_{j=l+1}^k m_j \right) \left(\sum_{i=1}^l b_i + b_k \right) + \sum_{j=1}^l m_j \sum_{i=l}^k b_i \right) \quad (23)$$

байтов. Из (23) с учетом (21) для $M_{T,max}$ получаем верхнюю оценку:

$$M_{T,max} < 2k 2^{13} + 8k 2^{16} + 4_{-l} 2^{16} + 2(l+1)_{-l} 2^{16} + 2l_{-(l+1)} 2^{16} = k 2^{14} + (_{-l} + 4k + _{-(l+1)} + l_{-(l+1)}) 2^{17} = k 2^{14} + (_{-(l+N_{T,max})}) 2^{17} \quad (24)$$

байт.

В таблице 1 приведены значения характеристик $N_{T,max}$ и $M_{T,max}$, рассчитанные по (21) и (24) для случаев, когда разрядность модуля p ограничена сверху порогами 1024 и 2462 бита. Мощности l и $_{-l}$ базисов $\{m_1, m_2, \dots, m_l\}$ и $\{m_{l+1}, m_{l+2}, \dots, m_k\}$ получены с помощью ПЭВМ согласно методике, разработанной в [8].

В максимальном КТ (20) наиболее значимая часть (по размеру занимаемой памяти) приходится на таблицы TEi_{-j} и $_{-}TEi_{-j}$, суммарное число которых определяется формулой $N_{T,p} = 2l_{-l}$. В соответствии с (21) в КТ (20) таблицы расширения кодов составляют долю:

$$\frac{N_{T,p}}{N_{T,max}} = \frac{2l_{-l}}{2l_{-l} + 5l + 6_{-l}} = \left(1 + \frac{2,5}{_{-l}} + \frac{3}{l} \right)^{-1} \quad (25)$$

Если $l = _{l} = 65$ (см. таблицу 1), то отношение (25) принимает значение $N_{T,p}/N_{T,max} \approx 0,922$, а в случае $l = _{l} = 155$ величина $N_{T,p}/N_{T,max} \approx 0,966$. С ростом l и $_{-l}$, что имеет место при увеличении разрядности модуля p , доля таблиц расширения кодов в максимальном КТ приближается к 1.

Из изложенного ясно, что отказ от хранения в оперативной памяти таблиц TEi_{-j} и $_{-}TEi_{-j}$ (полностью или частично) позволяет значительно сократить объем необходимой

табличной памяти для мультипликативной схемы (1). Для реализации обозначенного подхода удобно использовать минимальный КТ вида:

$$CT_MIN = \langle TIndl - TIndk; TAIndl - TAIndk; TRes_UP1 - TRes_UPk; TResl - TResk; TRes_Normi(i = 1, l - 1); TRes_Normi(i = l + 1, k - 1); TИi(i = 1, l); TИi(i = l + 1, k); TЭk_{-j}(j = 1, l); TМpli(i = l + 1, k) \rangle. \quad (26)$$

Общее количество таблиц, отвечающее КТ (26), включая таблицы системных констант, рассчитываемых в рамках алгоритма РК.1-ПК.19, составляет

$$N_{T,min} = 8k. \quad (27)$$

Суммарный объем памяти, требуемый для размещения КТ (26), а также таблиц констант определяется формулой:

$$M_{T,min} = 2 \left(5 \sum_{i=1}^{l-1} m_i + 4m_l + 7 \sum_{i=l+1}^{k-1} m_i + (l+6)m_k + k(m_{-max} + E_{-0} + E_{-1} + 2) + 2l_{-l} \right) \quad (28)$$

байтов. При $m_{-max} = 2^{16}$, $E_{-0} = E_{-1} = 2^{12}$ и основаниях m_1, m_2, \dots, m_k , выбираемых по методике, изложенной в [8], компьютерные расчеты по (27), (28) для модулей p разрядностью 1024 и 2462 бита соответственно дают $N_{T,min} = 8 \times 130 = 1040$ таблиц, $M_{T,min} = 153$ Мб и $N_{T,min} = 8 \times 310 = 2480$ таблиц, $M_{T,min} = 362$ Мб.

Из таблицы 1 следует, что при p разрядностью 1024 и 2462 бита максимальный КТ занимает больше памяти, чем минимальный соответственно в $(1155,78/153) \approx 7,55$ и $(6243,59/362) \approx 17,25$ раз. Вместе с тем, как показывают алгоритмы ГТИИ.1-ГТИИ.12 и ГТР.1-ГТР.12, продемонстрированная в них вычислительная технология с применением таблиц индексов и антииндексов и, в частности, типовой операции вида $TAind[c+TInd[\chi]]$ ($c \in Z_{m-1}$; $\chi \in \{1, 2, \dots, m_{-max}-1\}$; $m \in \{m_1, m_2, \dots, m_k\}$) позволяет весьма эффективно производить расчет слагаемых модульных сумм (5) и (2). Таким образом, использование минимального КТ (26) для реализации мультипликативной схемы Монтгомери (1), требуя небольших затрат табличной памяти, обеспечивает и высокую производительность.

Таблица 1 – Оценки затрат памяти для максимального и минимального КТ

Разновидность КТ	Параметры базовой мультипликативной схемы и МСС					Затраты табличной памяти	
	$\lceil \log_2 p \rceil$	$\log_2 m_{max}$	l	$_{-l}$	k	Число таблиц	Суммарный объем памяти (в Мб)
CT_max	1024	16	65	65	130	9165	1155,78
CT_max	2465	16	155	155	310	49755	6243,59
CT_min	1024	16	65	65	130	1040	153,0
CT_min	2462	16	155	155	310	2480	362,0

Заключение

Основные результаты представленной разработки по проблеме создания табличной реализационной базы мультипликативных МИМА-процедур типа Монтгомери состоят в нижеследующем.

1. На примере задачи расчета системных констант мультипликативной МИМА-схемы Монтгомери, а также таблиц ИИ и расширения кодов показана высокая эффективность табличного индексного метода модульного умножения. Он применим не только для вычисления произведений, но и для получения мультипликативных инверсий ЦЧ по модулям МСС, причем с использованием только аддитивных операций. Благодаря введенной избыточности в базовые таблицы индексов и антииндексов, предложенная конфигурация метода позволяет исключить преобразования вычетов при переходе от одних оснований МСС к другим.

2. Создан комплекс методологических и алгоритмических средств, составляющий базовое математическое обеспечение таблично-сумматорной технологии реализации мультипликативной МИМА-схемы Монтгомери. Применяемая технология дает гибкий инструментарий для разгрузки процесса умножения по большим модулям, осуществляемого в реальном времени, от трудоемких расчетов, которые могут быть выполнены предварительно. Это приводит к значительному повышению скорости реализации криптографических преобразований рассматриваемого класса.

3. В целях оптимизации режима распределения вычислений по этапам предварительных расчетов и расчетов, реализуемых в реальном времени, предложены алгоритмические средства, позволяющие устанавливать приемлемый баланс между объемом базового КТ и скоростью преобразований на основе операций умножения и возведения в степень по большим модулям. В рамках разработанного оптимизационного механизма получены оценки необходимой памяти для минимального и максимального КТ.

Литература:

1. Kawamura, S. Cox-Rower architecture for fast parallel Montgomery multiplication / Shinichi Kawamura, Masanobu Koike, Fumihiko Sano, Atsushi Shimbo // Eurocrypt 2000, LNCS. – Vol. 1807. – Berlin, 2000. – P. 523–538.
2. Nozaki, H. Implementation of RSA Algorithm Based on RNS Montgomery Multiplication / H. Nozaki, M. Motoyama, A. Shimbo, S. Kawamura // Proc. Cryptographic Hardware and Embedded Systems (CHES 2001). – Sept., 2001. – P. 364–376.
3. Bajard, J.-C. A Full RNS Implementation of RSA / J.-C. Bajard, L. Imbert // IEEE Trans. Comp. – 2004. – Vol. 53, № 6. – P. 769–774.
4. Lim, Z. An RNS-Enhanced microprocessor implementation of public key cryptography / Z. Lim, B.J. Phillips // Signals, Systems and Computers. – 2007. – ACSSC 2007. Conf. Rec. of the forte-first Asilomar Conf. – 4–7 nov., 2007. – P. 145–1434.
5. Коляда, А.А. Умножение по большим модулям с использованием минимально избыточной модулярной схемы Монтгомери / А.А. Коляда, А.Ф. Чернявский // Информатика. – 2010. – № 3. – С. 31–48.
6. Чернявский, А.Ф. Умножение по большим модулям методом Монтгомери с применением минимально из-

быточной модулярной арифметики / А.Ф. Чернявский, А.А. Коляда, Н.А. Коляда, Е.В. Шабинская: материалы Всерос. науч. конф. с элементами научной школы для молодежи «Параллельная компьютерная алгебра», Ставрополь, 11–15 окт. 2010 г. / Ставрополь. гос. у-т // Неирокомпьютеры: разработка, применение. – 2010. – № 9. – С. 38.

7. Каленик, А.Н. Компьютерно-арифметическая и реализационная база быстрых процедур умножения по большим модулям на основе модифицированной модулярной схемы Монтгомери / А.Н. Каленик, А.А. Коляда, Н.А. Коляда и др. // Электроника инфо. – 2012. – № 7. – С. 94–100.

8. Чернявский, А.Ф. Интервально-индексная технология расширения модулярного кода / А.Ф. Чернявский, А.А. Коляда, Н.А. Коляда, Е.В. Шабинская // Электроника инфо. – 2010. – № 6. – С. 7277.

Abstract

Methods and computer algorithms of generating of the complete set of tables for base modular number systems (MNS) of the Montgomery's scheme for multiplication on the big modules are developed. In particular procedures of formation of tables for summation of deductions on bases MNS by an accumulatively-tabular method, and also tables for calculation of intervally-index characteristics and expansion of codes are synthesized. The presented complex of methodological and algorithmic means gives ample opportunities for unloading of process of multiplication on the big modules, carried out in real time from calculations which can be executed preliminary. For generated complete sets of tables estimations of necessary memory are resulted.

Поступила в редакцию 12.11.2012 г.

Программаторы для любых микросхем

SEEPROM SEEPROM SEEPROM SEEPROM SEEPROM SEEPROM SEEPROM SEEPROM SEEPROM
EEPROM
EPROM
FLASH
PIC
MCU
PAL PAL

+375 (17) 266-32-09 www.chipstar.ru

ОДО «Вектор-Систем»

1. Ремонт, модернизация металлообрабатывающего оборудования с программным управлением
2. Модернизация электрорезонных станков
3. Внедрение электронных устройств ввода-вывода информации в УЧПУ
4. Замена устаревших приводов постоянного тока на экономичные частотные преобразователи с векторным управлением общепромышленными асинхронными электродвигателями
5. Установка электронных измерительных систем и устройств цифровой индикации
6. Комплексное решение задач автоматизации производства
7. Внедрение современных технологий и инструмента в металлообработку

г. Гомель, ул. Тролейбусная, д.5, к.6, тел./факс: +375 232 57-34-73
GSM: (8-10-375) 297-35-61-62, 298-86-29-16,
E-mail: stanok-cnc@yandex.ru www.vector-sys.com

ПОСТРОЕНИЕ ОТКАЗОУСТОЙЧИВЫХ ПРОГРАММИРУЕМЫХ ЦИФРОВЫХ УСТРОЙСТВ НА ОСНОВЕ ПРОГРАММНЫХ И АППАРАТУРНО-ПРОГРАММНЫХ СРЕДСТВ САМОВОССТАНОВЛЕНИЯ

УДК 681.518.54

Н.А. Коротаев, В.И. Попечиц, БГУ, г. Минск

Аннотация

Рассмотрены вопросы построения отказоустойчивых цифровых устройств на программируемых логических интегральных схемах (ПЛИС) на основе программных и аппаратурно-программных средств самовосстановления. Приведена упрощенная структурная модель отказоустойчивой СБИС на ПЛИС с аппаратурно-программными средствами самовосстановления, предназначенная для исследования отказоустойчивости и выбора наиболее эффективной структуры. Дана оценка значения выхода годных СБИС на ПЛИС с аппаратурно-программными средствами самовосстановления в зависимости от среднего числа дефектов, коэффициента группирования дефектов и числа резервных элементов.

Введение

Необходимость обеспечения отказоустойчивости цифровых устройств, построенных на новой элементной базе – программируемых логических интегральных схемах (ПЛИС) [1], обусловлена высокими требованиями к их быстродействию, точности, надежности и безотказности [2, 3]. Одним из перспективных путей обеспечения отказоустойчивости цифровых устройств на ПЛИС является использование при их построении встроенных средств самовосстановления. Обеспечение отказоустойчивости программируемых логических интегральных схем на основе встроенных аппаратурных средств самовосстановления рассмотрено в работе [4]. Принципов построения отказоустойчивых программируемых цифровых устройств на основе программных и аппаратурно-программных средств самовосстановления в известных работах авторами не найдено, поэтому в данной работе мы предлагаем использовать при построении отказоустойчивых программируемых цифровых устройств программные или аппаратурно-программные средства самовосстановления.

Построение отказоустойчивых программируемых цифровых устройств на основе программных и аппаратурно-программных средств самовосстановления

Программные средства самовосстановления подразделяются на средства повторения, дублирования и деградации.

Средства повторения предусматривают повторное выполнение программ, фрагментов программ или отдельных операций (команд или микрокоманд) в зависимости от количества ошибочной информации и дают правильный результат, если ошибка проявляется в виде сбоя, т.е. является случайной (такая ошибка исчезает в процессе восстановления). Если обнаружена ошибка в отдельной операции, то повторяется выполнение этой операции до восстановления правильного вычислительного процесса. Повторное выполнение фрагмента программы, т.е. возвращение к контрольной точке осуществляется в том случае,

когда последствия ошибки успели отразиться на таком количестве информации, что восстановление путем повторения отдельных операций невозможно или не имеет смысла. При этом контрольная точка используется в качестве исходной при условии, что все результаты в данной точке сохранены и не содержат ошибок. Повторение программы применяется тогда, когда в результате действий ошибок разрушено такое количество информации, что восстановление путем повторного выполнения фрагментов программы невозможно или нецелесообразно. Следует отметить, что данные программные средства повторения вызывают задержку вычислительного процесса, которая зависит от конкретной области применения отказоустойчивой СБИС ПЛИС.

Средства дублирования представляют собой программную избыточность, которая применяется на различных уровнях: системном, модульном, операции данных с целью восстановления работоспособности цифрового устройства при обнаружении ошибки, проявляющейся в виде отказа. Самовосстановление на системном уровне организуется за счет избыточности процесса в мультипрограммной или мультипроцессорной среде (т.к. различные комбинации процессов создают различную среду для системы). Избыточность на модульном уровне позволяет самовосстанавливать функционирование устройства путем выполнения автоматической реконфигурации, которая состоит в замене отказавшего модуля его исправной копией и возобновлении вычислительного процесса, начиная с определенной контрольной точки. Восстановление на уровне операций осуществляется с помощью использования средств защиты или избыточной информации, хранимой в программе, или сохранения копии. Применение избыточности на уровне данных позволяет восстанавливать данные, испорченные в результате действия ошибок, и обеспечивать защиту целостности данных.

Средства деградации представляют собой некоторую программную избыточность, с помощью которой организуется перестройка-реконфигурация цифрового устройства (структуры), в котором отсутствуют или уже использованы резервные элементы, с целью устранения действия отказов и обеспечения работоспособности устройства. При этом происходит снижение вычислительных способностей устройства, т.к. из-за ошибок теряется часть функций.

Таким образом, программные средства самовосстановления, представляя собой программную избыточность, увеличивают стоимость системы, т.к. требуют дополнительных аппаратурных средств (память, процессор и т.д.). Однако этот недостаток компенсируется тем, что рассматриваемые средства самовосстановления существенно улучшают показатели функционирования

и надежности СБИС ПЛИС, которые имеют тенденцию уменьшения стоимости аппаратуры и повышения ее быстродействия.

Аппаратурно-программные средства самовосстановления представляют собой сочетание аппаратурных и программных средств, которые применяются на различных иерархических уровнях системы и обеспечивают исправление широкого класса неисправностей и оперативное восстановление работоспособности цифрового устройства. Эти средства подразделяются на статические и динамические.

Статические аппаратурно-программные средства осуществляют самовосстановление в момент времени, когда устройство не применяется по своему назначению, т.е. при снятии основных задач. В этом случае процесс самовосстановления выполняется поочередно или синхронно всеми ресурсами системы кристалла (цифрового устройства). В устройствах с централизованным управлением восстановление выполняется под управлением единого программного (аппаратурного) управляющего средства, что позволяет достичь оптимального распределения ресурсов. При этом управляющие средства должны быть самодиагностируемыми и самовосстанавливаемыми, иначе наличие отказа в них приведет к полному отказу всего устройства (СБИС ПЛИС). В устройствах с децентрализованным управлением самовосстановление выполняется либо выделенным в некоторый момент времени управляющим программным (аппаратурным) средством, либо с помощью программных (аппаратурных) управляющих средств, выполняемых каждым микропроцессором самостоятельно (в этом случае необходимо согласование действий микропроцессоров).

Динамические аппаратурно-программные средства самовосстановления применяются во время решения основных задач, т.е., когда цифровое устройство применяется по своему назначению. При этом наличие или отсутствие того или другого вида аппаратурных и программных средств восстановления работоспособности цифрового устройства (СБИС ПЛИС): дублирования, программной коммутации и т.д. определяется их назначением, требованиями к надежности при эксплуатации, количеством и сложностью логических схем и экономическими показателями.

Самовосстанавливаемые цифровые устройства должны обладать следующими свойствами:

- обнаружение неисправности в момент времени ее первого проявления (данное свойство реализуется с помощью самопроверяемых схем встроенного контроля);
- локализация неисправности с заданной глубиной (свойство реализуется путем применения встроенных средств самотестирования);
- перестройка-реконфигурация структуры устройства (реализуется с помощью программных средств);
- восстановление нормального функционирования устройства (реализуется соответствующими встроенными аппаратурно-программными средствами).

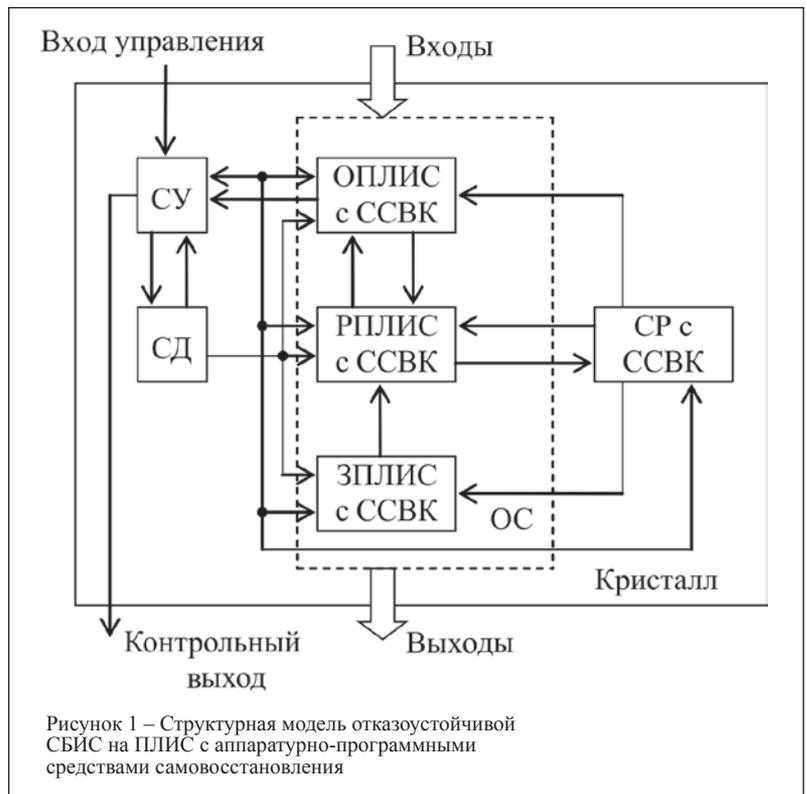


Рисунок 1 – Структурная модель отказоустойчивой СБИС на ПЛИС с аппаратурно-программными средствами самовосстановления

На основе данных свойств предлагается следующая упрощенная структурная модель отказоустойчивой СБИС на ПЛИС с аппаратурно-программными средствами самовосстановления (рисунок 1).

Модель содержит: средства управления (СУ) процессом самовосстановления; средства самодиагностирования (СД), включающие генератор тестовой последовательности и (или) контролирующие и локализирующие тесты, записанные в ПЗУ; объект самовосстановления (ОС), включающий работоспособные ПЛИС (РПЛИС), запасные ПЛИС (ЗПЛИС) и отказавшие ПЛИС (ОПЛИС) с самопроверяемыми схемами встроенного контроля (ССВК); средства реконфигурации (СР) с ССВК.

Работоспособные ПЛИС с ССВК, выполняющие все рабочие функции СБИС, составляют исходную структуру (вычислительное ядро) устройства. Возникшая неисправность либо маскируется избыточной информацией (в случае сбоя), либо обнаруживается и локализуется средствами самодиагностирования (в случае отказа).

Неисправная ПЛИС средствами реконфигурации исключается из исходной структуры и заменяется на запасную ПЛИС, а отказавшая ПЛИС включается в число ОПЛИС, подлежащих восстановлению. Средства самовосстановления возобновляют правильное функционирование СБИС. При этом в случае восстановления отказавшей ПЛИС, она пополняет число ЗПЛИС. Если резерв исчерпан, выполняется реконфигурация исходной структуры СБИС без замены ОПЛИС на ЗПЛИС. В этом случае наблюдается деградация системы, т.е. понижение ее вычислительных способностей. Однако деградация будет почти не наблюдаться, если учесть то, что в настоящее время число элементарных схем (транзисторов) в одном корпусе интегральной схемы выросло до сотен миллионов и это позволяет создавать достаточный запас для обеспечения высокой отказоустойчивости СБИС.

Данная структурная модель позволяет представить различные структуры программируемого цифрового устройства с аппаратурно-программными средствами самовосстановления, моделирование которых на компьютере дает возможность сравнения надежностных характеристик исследуемых отказоустойчивых структур [4], оценки параметров различных топологий, определения оптимальных значений вносимой избыточности и выбора эффективной структуры. Для оценки эффективности отказоустойчивой структуры важным является анализ достигаемого улучшения выхода годных СБИС ПЛИС в зависимости от определенных параметров. Расчеты значения выхода годных (Y) СБИС ПЛИС (типа ПЛМ – программируемая логическая матрица) в зависимости от среднего числа дефектов (λ) на кристалле, коэффициента группирования дефектов (α) и числа резервных элементов (K_p), выполненные на основе формул (1–3) [5], позволили получить следующие результаты (рисунки 2, 3, 4):

$$Y = \beta Y_6 Y_{нс} \frac{A}{A + K_p} \quad (1)$$

где β – уровень потока дефектов (обычно принимается $\beta=1$); Y_6 – выход восстановимой площади, в которой можно устранить до S дефектов, вычисляется по формуле (2):

$$Y_6 = \sum_{n=0}^S \frac{\Gamma(\alpha + n)(\lambda/\alpha)^n}{n! \Gamma(\alpha)(1 + \lambda/\alpha)^{\alpha+n}}; \quad (2)$$

$Y_{нс}$ – выход невозстановимой площади, в которой число устранимых дефектов $S=0$, вычисляется с помощью формулы (3):

$$Y_{нс} = (1 + \lambda/\alpha)^{-\alpha}; \quad (3)$$

A – исходная площадь, K_p – дополнительная площадь (число резервных элементов), Γ – гамма-функция.

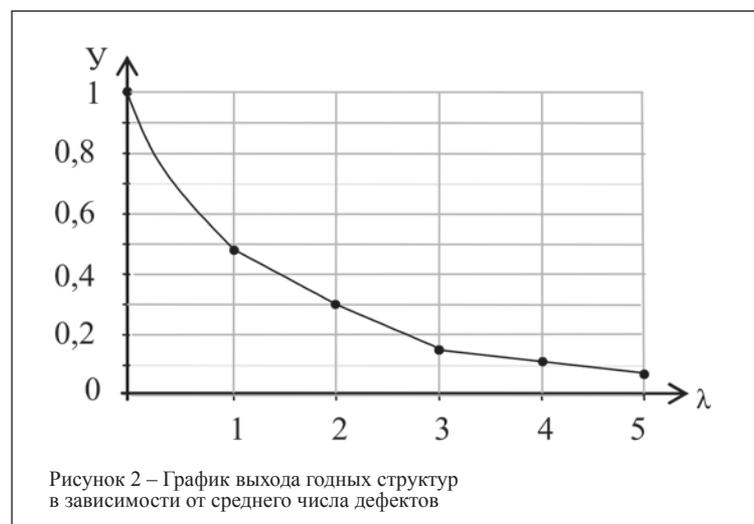


Рисунок 2 – График выхода годных структур в зависимости от среднего числа дефектов

График (рисунок 2) показывает, что увеличение среднего числа дефектов в структуре СБИС ПЛИС уменьшает выход годных структур.

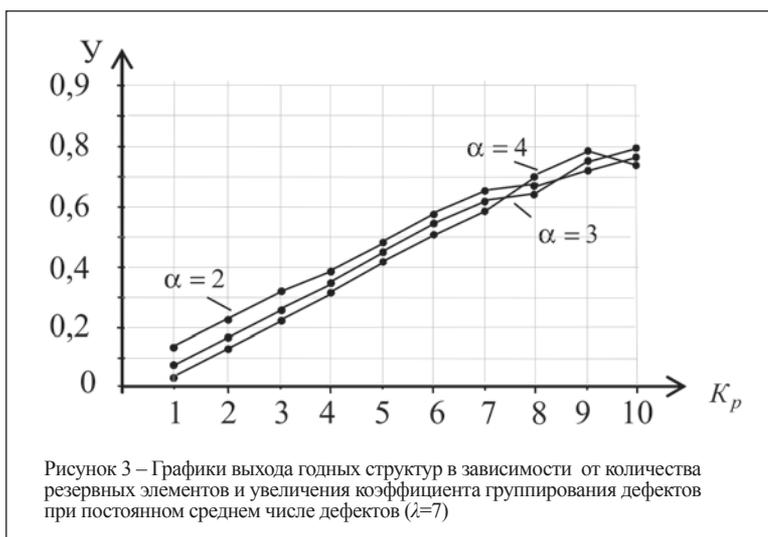


Рисунок 3 – Графики выхода годных структур в зависимости от количества резервных элементов и увеличения коэффициента группирования дефектов при постоянном среднем числе дефектов ($\lambda=7$)

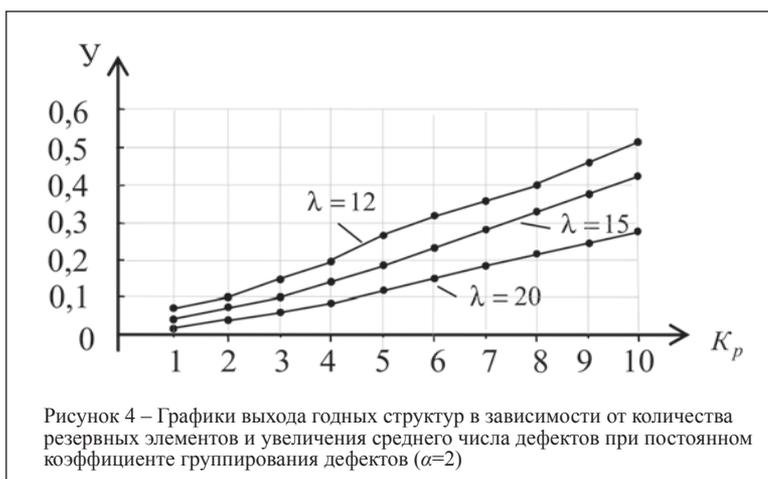


Рисунок 4 – Графики выхода годных структур в зависимости от количества резервных элементов и увеличения среднего числа дефектов при постоянном коэффициенте группирования дефектов ($\alpha=2$)

Графики (рисунок 3) показывают, что увеличение коэффициента группирования дефектов при постоянном среднем числе дефектов незначительно влияет на выход годных схем, а увеличение числа резервных элементов существенно увеличивает выход годных кристаллов.

Графические зависимости (рисунок 4) указывают на то, что значение выхода годных структур существенно увеличивается с уменьшением среднего числа дефектов при постоянном коэффициенте группирования дефектов.

Заключение

Предложены программные и аппаратурно-программные средства самовосстановления для обеспечения отказоустойчивости цифровых устройств на программируемых логических интегральных схемах. На основе анализа вычисленных значений надежностных характеристик аппаратурных, программных и аппаратурно-программных средств самовосстановления можно сделать вывод о том, что лучшими в обеспечении отказоустойчивости работы СБИС на ПЛИС являются аппаратурно-программные средства самовосстановления.

Приведена упрощенная структурная модель отказоустойчивой СБИС на ПЛИС с аппаратурно-программными средствами самовосстановления, моделирование которой на компьютере позволяет исследовать отказоустойчивость различных программируемых устройств и выбрать эффективную структуру устройства.

Вычислены значения выхода годных СБИС на ПЛИС (типа ПЛИМ) с аппаратурно-программными средствами самовосстановления в зависимости от среднего числа дефектов, коэффициента группирования дефектов и числа резервных элементов.

Аналогично могут быть вычислены и проанализированы надежностные характеристики для программируемых цифровых устройств на ПЛИС других типов (программируемых пользователем вентильных матриц, программируемых запоминающих устройств, программируемых матриц логики и сложных логических устройств).

Актуальность и важность рассматриваемой задачи в области построения отказоустойчивых программируемых цифровых устройств на основе аппаратурно-программных средств самовосстановления требует дальнейших исследований.

Литература:

1. Соловьев, В.В. Проектирование цифровых систем на основе программируемых логических интегральных схем / В.В. Соловьев. – М.: Горячая линия – Телеком, 2001. – 636 с.
2. Виноградов, В.И. Микро- и нанoeлектронные технологии компактных и масштабируемых компьютерных

систем высокого быстродействия / В.И. Виноградов // Электроника инфо. – 2007. – № 9. – С. 44–46.

3. Согомоян, Е.С. Самопроверяемые устройства и отказоустойчивые системы / Е.С. Согомоян, Е.В. Слабаков. – М.: Радио и связь, 1989. – 208 с.

4. Коротаев, Н.А. Обеспечение отказоустойчивости программируемых логических интегральных схем на основе самовосстановления / Н.А. Коротаев, В.И. Попечиц // Электроника инфо. – 2012. – № 1. – С. 99–102.

5. Ha, D.S. On the design of high – yield reconfigurable PLA's / D.S. Ha, Y.P. Kumar // IEEE Trans. Comput. – 1990. – Vol. ED – 39, № 4. – P. 470–479.

Abstract

Questions of creation of failure-safe digital devices on programmed logic integrated schemes (PLIS) on the basis of program and equipment-program softwares of self-restoration are considered. The simplified structural model of failure-safe SBIS on PLIS with equipment-program softwares of self-restoration intended for research of fault tolerance and a choice of the most effective structure is given. The assessment of value of an yield of suitable SBIS on PLIS with equipment-program softwares of self-restoration depending on average of defects, factor of grouping of defects and number of reserve elements is given.

Поступила в редакцию 20.11.2012 г.



БелПлата тел. +375 17 287 85 66
 факс +375 17 287 85 65
 тел.моб. +375 29 684 43 09
 220068, г. Минск, ул.Некрасова, 114,
 оф.238, 2 этаж, e-mail: info@belplata.by

Разработка и поставка печатных плат:
 любой класс точности, широкий спектр покрытий, изготовление образцов от 5 дней.

Поставка фотошаблонов

Поставка трафаретов:
 из нержавеющей стали и латуни.

Материалы для печатных плат:
 защитные маски, маркировочные краски, фоторезисты, паяльные пасты.

Поставка изделий из феррита:
 любые виды сердечников CI, EE, EEM, EP, EER, ETD, EC, EF, ED, EFD, EI, EPO, EPX, EPC и т.д.

Поставка электронных компонентов:
 STMicroelectronics, NXP Semiconductors, Vishay, Holtek Semiconductor.

www.belplata.by



ООО "ГорнТрейд"

поставка электронных компонентов

контрактное производство

тел.: +375 17 290 0082
 факс: +375 17 290 0084
 e-mail: info@horntrade.net

ПТСС ЗАО «Промтехсервиснаб»

г. Минск, ул. Богдановича, 120 Б, ком. 6, Тел.: 266-23-94; 266-23-92,
 e-mail: uslugi@zapservis.by, моб.: (029) 676-02-56,
www.zapservis.by моб.: (044) 773-72-04,
 факс: 266-23-94.

Контрактное производство РЭА:

- ✓ Автоматизированный монтаж SMD-компонентов любой сложности.
- ✓ Комбинированный монтаж с использованием компонентов DIP (выводных), а также SMD (планарных).
- ✓ Сборка корпусных деталей и узлов.
- ✓ Ультразвуковая отмывка печатных плат и узлов высококачественными импортными растворами.
- ✓ Нанесение влагозащитных покрытий.

**Высокотехническое импортное оборудование обеспечивает высокое качество производимой продукции.*

Продажа неликвидов ПКИ для РЭА:

- ✓ Резисторы, конденсаторы, транзисторы, микросхемы и т.д.

HUD: ПРОЕКЦИЯ БУДУЩЕГО

Светлана Сысоева
Dr.Gold@sysoeva.com

Термин HUD (Head-Up Display) описывает технологию представления актуальной информации в фокусе поля зрения пилота поверх окружающей обстановки в виде дополнительного изображения, виртуальные элементы которого буквально «висят в воздухе». HUD используются в авионике и автомобилях, классифицируются по типу используемых источников освещения и ранжированы от ЭЛТ до светодиодных, дисплейных и наиболее перспективных лазерных (голографических и проекционных) технологий.

В 2012 году объем рынка HUD приблизился к 1 млрд долл. США. Ожидается, что в следующие пять лет данная цифра утроится, главным образом, за счет новых применений в автоэлектронике и дальнейшего развития технологий. Перспективными можно считать такие применения, как навигация, системы безопасности, опции контекстной осведомленности, дополненная реальность, технологии контактно-аналоговых дисплеев, 3D-визуализация, обнаружение жестов, распознавание голоса, интерактивные touch-дисплеи и др.

Введение

История HUD ведет отсчет с начала 1940-х годов, когда была впервые озвучена концепция применения этой технологии в военной авиации. Разработка увидела свет уже после Второй мировой войны и нашла также широкое

применение в гражданской авиации. Полет истребителя осуществляется на высокой скорости – свыше 800 км/ч. Отвлечение для взгляда на дисплей приборной панели буквально на 1 с эквивалентно тому, что несколько сотен метров самолет летит вслепую. Конечно, в обычных условиях навигация ведется по приборам, но во время взлета/посадки или в воздушном бою требуется также и визуальный обзор. Чтобы минимизировать отвлечение пилота, была разработана концепция HUD, что переводится с английского как «надголовный» дисплей. В авиационных системах на нем отображается скорость полета, высота, угол тангажа, курс и другая важная информация (рисунок 1 а, б). Таким образом, системы HUD позволяют пилоту получать необходимую информацию без отвлечения на HDD (Headdown Display)¹ и другие кабинные дисплеи, число которых в коммерческих самолетах достигает нескольких десятков.

С конца 1980-х годов началось применение HUD в автоэлектронике, где концепция «надголовного» дисплея, или представления актуальной информации перед глазами водителя без ограничения обзора дороги и отвлечения на приборную панель представляет значительный интерес для повышения дорожной безопасности. В легковых автомобилях на HUD отображаются скорость, данные навигационной системы (рисунок 1 б, в), и т. д.

Виртуальное изображение, спроецированное на ветровое стекло, воспринимается как парящее в воздухе выше капота, и видно только водителю. Такая форма представления информации менее изнурительна для глаз, поскольку позволяет избежать постоянного переключения для считывания информации с ближнего на дальнее расстояние.

HUD представляет собой довольно сложную систему, в состав которой входят проектор, прозрачный экран, проецируемый на лобовое стекло летчика или водителя, видео-генератор (мини-компьютер), цифровой оптический объединитель (combiner) и управляющая панель. Данные для HUD, получаемые с бортового компьютера, после соответствующей обработки проецируются на экран. Объединитель служит для согласования виртуального изображения, выдаваемого на поверхность вогнутой или плоской формы для позиционирования информации в



Рисунок 1 (а–г) – Ключевые применения HUD/HMD:
а) HUD в военном истребителе F16 Lockheed Martin;
б) коммерческий авиационный HUD Elbit Systems;
в) стандартный вид современного HUD, применяющегося в автомобилях BMW;
г) новое применение технологии HUD для автомобильной навигации (источник: Pioneer);

¹ Head Down Display (HDD) – дисплей в кабине, помещенный ниже или в одном ряду с приборной панелью. Под это определение подпадают дисплеи, требующие небольшого наклона головы вниз или в сторону.

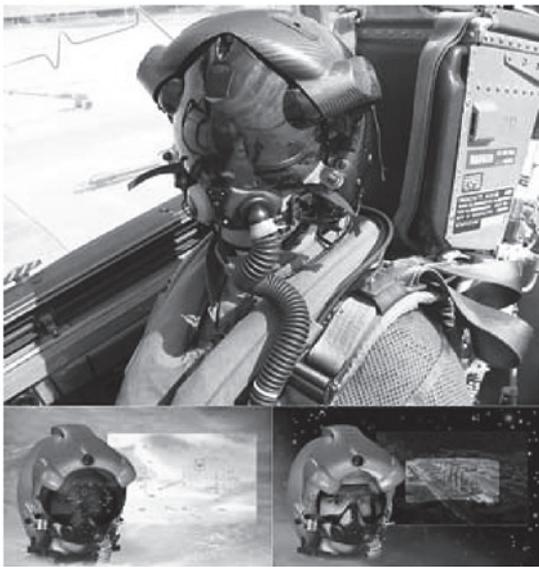


Рисунок 1 д – Ключевые применения HUD/HMD:
HMD в истребителе F35 (альтернатива HUD)

поле зрения пилота или водителя. Это может быть ветровое стекло или специальный экран, на котором отражается монохроматическое (или полноцветное) изображение от проектора таким образом, чтобы оно было видимо поверх воспринимаемой реальной ситуации. Изображение видно, как правило, только водителю или пилоту в пределах так называемого eyebox² – коридора восприятия глазами.

Отображение актуальной информации на уровне глаз также представляет интерес для повышения безопасности и навигации мотоциклистов. Но в данном случае практическая реализация отличается тем, что необходимо обеспечить восприятие виртуального изображения через очки или прозрачную поверхность шлема, например, мотоциклиста. Идея обеспечения доступности актуальной информации для летчика, мотоциклиста или солдата (рисунок 1 д) осуществляется посредством так называемых HMD (Helmet/Head Mounted Display) – дисплеев дополненной реальности, устанавливаемых на шлеме/голове.

Еще одна технология, известная под названием «виртуальный ретинальный дисплей» (Virtual retinal display, VRD, или Retinal Scan display, RSD), представляет собой технологию вывода визуальной информации непосредственно на сетчатку глаза. Мобильные проекционные дисплеи – еще один вариант использования тех же технологий, которые могут применяться в системах HUD. В зависимости от контекста, перечисленные применения можно считать альтернативой, дополнением или расширением технологий HUD.

Первым поколением HUD были дисплеи, основанные на растровой технологии ЭЛТ (CRT) формирования изображений на экране. Следующее поколение HUD основывалось на использовании ламп высокого давления, а затем и твердотельных источников освещения, также модулируемых для получения изображения с использованием LCD, т.е. проекторы реализовывались на основе ЖК-модулятора и светоди-

одных источников света. Лазерные сканирующие MEMS HUD являются наиболее «продвинутыми» продуктами, которые могут проецировать изображение с лучшим разрешением и большей четкостью, чем это возможно посредством других технологий. Благодаря развитию именно этих технологий, отличающихся компактностью и, в перспективе, более высоким качеством изображения, причем по более низкой цене, становится возможным более широкое распространение технологий на массовых рынках – от автомобильной до потребительской электроники. Эти рынки благодаря высоким объемам выпуска продукции в состоянии обеспечить дальнейшее развитие технологий, создающих виртуальные изображения, и отдельных направлений, включая, как уже говорилось, дополненную реальность, голосовой контроль, обнаружение жестов, 3D-визуализацию и т.д.

Примеры применения HUD в авиации

Как уже было сказано, концепция HUD впервые появилась в 1940-х годах, и со временем была расширена до показа различных баллистических переменных – скорости летательного аппарата, скорости цели, расстояния и др. В 1958 году британский самолет Blackburn Buccaneer впервые был оснащен HUD, и, таким образом, можно говорить, что с 1960-х годов эта технология массово применяется в военной авиации.

Вскоре в гражданской (коммерческой) авиации HUD стали важными и неотъемлемыми составными частями самолетов и вертолетов. Sextant Avionique (ныне торговая марка Thales) в 1975 году разработала первый HUD для самолета Dassault Mercure, затем Sundstrand и Douglas Aircraft Company установили HUD в самолетах MD80 (в конце 1970-х). В 1990-х годах уже большинство коммерческих авиалиний использовало HUD. Сегодня эти системы являются неотъемлемым атрибутом любого летательного аппарата как в военной, так и в гражданской авиации.

Типичные HUD летательных аппаратов отображают воздушную скорость, высоту, линию горизонта (авиагоризонт), разворот/крен (turn/bank), курс, индикаторы скольжения и скольжения на крыло (slip/skid). Эти инструменты являются минимальным требованием 14 CFR Part 91³. Пилот может смотреть на проецируемые данные, не теряя контроль за окружающим пространством, и визуально считывать только актуальную информацию. Полезность HUD особенно высока в условиях низкой видимости и при посадке.

HUD в системах авионики обычно получают данные непосредственно от датчиков (приемников воздушного давления, гироскопов, навигационных систем и т.п.), установленных на борту летательного аппарата и выполняющих собственные вычисления вместо использования вычисленных данных от бортовых компьютеров. В ряде летательных аппаратов данные HUD для взлета в условиях низкой видимости поступают от компьютера, управляющего автопилотом (Boeing 787). Компьютеры интегрированы с системами летательных аппаратов и допускают подключение посредством различных шин данных (ARINC 429, ARINC 629 и MIL-STD-1553).

Разработанная Thales Group (www.thalesgroup.com) система D-HUDS (Digital Head-Up Display System) создавалась

² DHUD eyebox, а также head motion box – 3D-область в пространстве, где HUD может быть виден по крайней мере одним глазом.

³ Code of Federal Regulations (CFR). Part 91 – General Operating and Flight Rules. <http://ecfr.gpoacce>



Рисунок 2 – Применение системы HUD от Thales на самолете Airbus A350-XWB

как всеобъемлющее решение для коммуникации, навигации и наблюдения. Thales использует для системы HUD LCD-технологии и производит также дисплейное стекло с активными матрицами. D-HUDS предоставляет информацию для всех фаз полета, обеспечивает высокое графическое качество, поддерживает клиентскую символику, обладает возможностями улучшенных систем видения нового поколения EVS (Enhanced Vision System), синтетического (синтезированного) зрения SVS (Synthetic Vision System), системы поверхностного (наземного) наведения SGS (Surface Guidance System). Оборудованный компьютером, D-HUDS воспроизводит яркие и качественные изображения. Один из примеров применения системы HUD от Thales – на самолете Airbus A350-XWB – показан на рисунке 2.

Компания Rockwell Collins (www.rockwellcollins.com) имеет в своем портфолио систему наведения Head-up Guidance System (HGS), которая содержит в себе переходную стадию от использования дисплеев HDD к сопровождаемой HUD фазе полета. HGS собирает и оценивает важнейшие данные полета: положение самолета в воздухе, скорость, высоту и навигационные входы для вычисления информации, необходимой для приземления с использованием запатентованной компьютерной архитектуры с высоким уровнем целостности, разрешающей посадки категории III, регулирует взлеты в условиях низкой видимости. HGS также отображает специальную символику для событий TCAS⁴, рекомендации по восстановлению при смещении ветров и избеганию хвостовых ударов. Модель HGS-3500 выводит точные указания на лобовое стекло. Для получения изображения используется освещаемый светодиодами LCD-дисплей, на который выводится информация от систем синтетического видения окружающего мира (рисунок 3 а). HGS-3500 – первая в промышленности HUD-система для средних и легких коммерческих самолетов (рисунок 3 б).



Рисунок 3 – HUD-система Rockwell Collins HGS-3500: а) внешний вид; б) вид в кабине



Рисунок 4 – Иллюстрация технологии Q-HUD от BAЕ Systems

BAE Systems (www.baesystems.com) разработала систему Q-HUD (рисунок 4) – недорогое решение HUD, которое значительно меньше, легче, дешевле, чем системы-предшественники. Q-HUD также отображают важную информацию, включая скорость, высоту, угол тангажа, курс, маркер траектории полета и т.д., поверх области обзора пилота. В технологии BAE Systems используются голографические волноводы, эффективно генерирующие символику на стекле без проецирования при этом изображения на поверхность, как большинство HUD. Благодаря методу генерации изображения, система Q-HUD не требует тяжелой и дорогой оптики и громоздких проекционных систем в области головы пилота. Системы Q-HUD используют низкопрофильный дисплейный блок, который может быть закреплен выше приборной панели так, что объединитель всегда является видимым. Это допускает свободу расположения HUD в кабине. На основе данной технологии BAE Systems производит продукты MonoHUD и IHUD, а также HMD.

ANVIS IHUD (Aviator Night Vision Imaging System Improved Head-Up Display) – система ночного видения с улучшенным HUD от BAE – обеспечивает визуальное наблюдение данных полета при поддержке обзора ситуации во внешней среде. Первичное назначение IHUD – повышение безопасности экипажа. Новая IHUD система меньше, легче, энергоэкономичнее и обладает лучшими системными характеристиками.

С 2012 года BAE Systems выполняет заказ армии США в объеме 21,7 млн долл. США на поставку в неопределенном количестве дисплеев HMD, также известных как удаленный блок формирования изображений на дисплее очков Remote Eyepiece Display Imager, или RED-I, совместимый с военными датчиками теплового зрения.

⁴ Traffic alert and Collision Avoidance System – бортовая сигнализационная система предупреждения воздушных столкновений, выдающая экипажу информацию о необходимых маневрах для предотвращения катастрофы.

Примеры применения HUD в автоэлектронике

Развитие технологий HUD

Уже более двух десятилетий HUD устанавливаются в автомобилях. Впервые они были применены General Motors (www.gm.com) в конце 1980-х годов. Первый цветной дисплей появился в 1998 году. Toyota Motor Corporation (www.toyota-global.com) в 1991 году выпустила HUD для внутреннего рынка, а в 2003 году компания BMW Group AG (www.bmwgroup.com) стала первым европейским автопроизводителем, выпустившим HUD. Но популярными в автомобильном применении HUD стали только в последнее время – с развитием технологий и повышением характеристик дисплеев в плане яркости, контрастности, цветности виртуального изображения.

В настоящее время для автомобильных HUD используются следующие технологии:

- зеркальное отражение изображения, сформированного с использованием обычного светодиодного источника или дисплея (Continental, BMW, Audi, DENSO);

- лазерная голография с использованием оптических волноводов (Delphi);

- лазерная сканирующая МЭМС-технология (Pioneer).

Перспективы HUD связывают с лазерными технологиями и расширением дисплейных применений, например, для навигации (BMW, Pioneer), ночного видения (Denso), защиты пешеходов (Volvo). Почти все производители имеют в своем портфолио HUD, обладающие средствами для создания дополненной реальности с виртуальными объектами, привязанными к физическим. Ранее HUD были автономными системами, но сейчас многие поставщики стали объединять их с другими системами и приборами, включая дисплеи систем навигации или развлечений, touch-дисплеи. Распознавание голоса, контроль жестов, 3D-визуализация – новые направления в технологиях автомобильных HUD, которые способствуют появлению новых коммерческих решений от различных поставщиков.

General Motors:

лазерная проекция на полноэкранных HUD

В настоящее время технология является стандартным признаком автомобиля GMC Acadia Denali модельного ряда 2013 года (рисунок 5 а) и доступна на других автомобилях – Buick LaCrosse, Chevrolet Camaro.

В Acadia HUD выполняет функции проецирования изображений из приборной панели, используя два зеркала для отражения четкого цифрового изображения от ветрового стекла (рисунок 5 б), посредством чего возникает виртуальное изображение, видимое водителю как висящее над капотом. Водители могут регулировать яркость и местоположение изображения или отключать его полностью. Видимая информация – скорость и обороты двигателя, различные предупреждения, активация сигналов поворота, температура и другая информация.

General Motors планирует и далее повышать характеристики устройства посредством интеграции лазерных указателей и систем улучшенного видения в сложных условиях. Компания сообщает, что в будущем малые УФ-лазеры смогут взаимодействовать с автомобильными датчиками и камерами и отображать их показания на ветровом стекле, автоматически высвечивать край дороги

или другие объекты и предоставлять визуальные данные для помощи в вождении (рисунок 5 в, г).

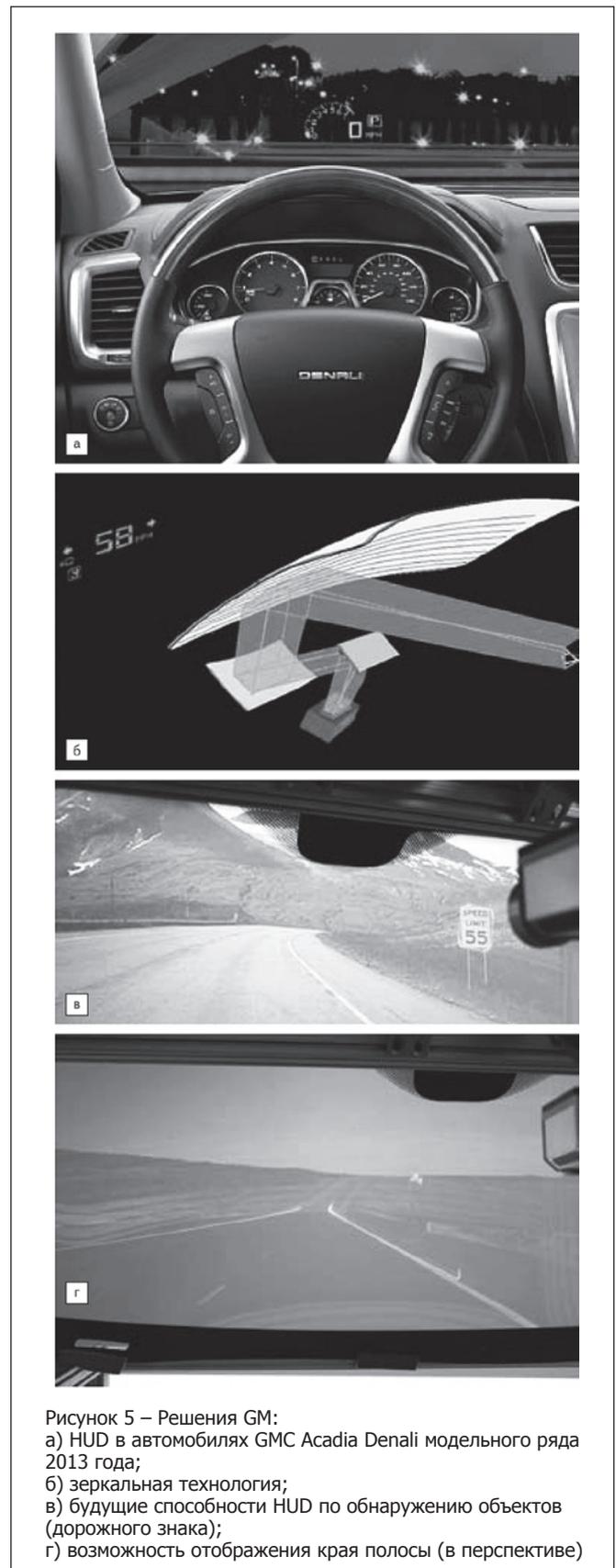


Рисунок 5 – Решения GM:

а) HUD в автомобилях GMC Acadia Denali модельного ряда 2013 года;

б) зеркальная технология;

в) будущие способности HUD по обнаружению объектов (дорожного знака);

г) возможность отображения края полосы (в перспективе)

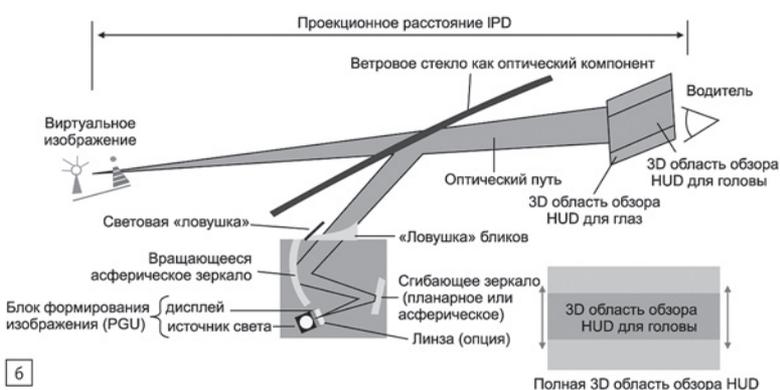


Рисунок 6 (а, б) – HUD второго поколения от Continental: а) в автомобилях BMW 3-й серии; б) оптическая система

Исследования GM проводит в сотрудничестве с университетом Карнеги-Меллон и университетом Южной Калифорнии, а также другими институтами. Цель – повысить видимость с помощью полноэкранный HUD, сформированного при покрытии фосфор-содержащим составом всего ветрового стекла, которое отражает видимый свет при излучении от компактного лазера. Для лучшей интеграции с полем зрения водителя технология HUD GM следующего поколения трансформирует лобовое стекло в прозрачный HUD большого размера, отображающий информацию непосредственно от датчиков – приборов ночного видения, GPS и камер. Система также включает новую концепцию GPS, позволяющую водителю видеть точки интереса на экране в реальном времени.

BMW и Audi: зеркальные системы HUD от Continental

В настоящее время Continental (www.conti-online.com) поставляет HUD-дисплеи второго поколения двум премиум-автопроизводителям – BMW и Audi (рисунок 6–8). Компания производит HUD с 2003 года, а в 2010 году приступила к серийному производству HUD второго поколения для Audi (www.audi.com). С начала 2012 года легкие, компактные, отличающиеся улучшенной функциональностью, дисплеи HUD второго поколения устанавливаются на популярные автомобили средней ценовой категории – BMW 3-й серии (рисунок 6 а).

Изображение появляется перед глазами водителя выше капота, на расстоянии примерно 2 м, и позволяет без отрыва взгляда от дороги получать информацию обо всех важных событиях (рисунок 6 б). HUD показывают релевантную информацию: скорость, данные от систем навигации и развлечений, предупреждения.

Система HUD проецирует виртуальное изображение на ветровое стекло, сгенерированное LCD-TFT дисплеем со светодиодной подсветкой в качестве источника света. Посредством зеркал лучи, освещающие дисплей, адресуются на ветровое стекло. Асферические зеркала свободной формы могут компенсировать дисторсию вследствие формы ветрового стекла и увеличивают изображение.

Водитель может конфигурировать HUD в кабине в соответствии со своим ростом (рисунок 6 в) посредством настроек и шагового двигателя, наклоняющего оптическое зеркало HUD для изменения положения плавающего изображения, отображаемого над капотом.

Ключевые технические характеристики технологии следующие:

- расстояние виртуального изображения от глаз 2–2,5 м;
- размер изображения 262×87 мм для проекционного расстояния в 2,5 м;
- видимая яркость адаптирована к окружающей яркости (день/ночь) > 10 000 кд/м²;
- световой источник светодиодный, RGG (Red-Green-Green), RGB HP-LED (Red-Green-

Blue High Performance, RGB с высокими характеристиками) или белый светодиод;

- максимальное разрешение TFT 480×240 пикс.;
- оптическая система с одним-двумя асферическими зеркалами;
- возможна дисторсия (искривление);
- объем инсталляции в кабине за комбинацией приборов – 2–4 л.

Качество изображения дисплеев HUD второго поколения является более высоким. В сравнении с предыдущим поколением с ограниченным цветовым спектром, новые белые светодиоды проецируют весь световой спектр от изображения, сгенерированного дисплеем TFT. Повышенная яркость белой светодиодной технологии в новом блоке формирования изображения означает более высокую контрастность новых HUD и позволяет повышать качество изображения в условиях ночного видения. Новый блок формирования изображения включает 15 светодиодов, что снижает потребляемую мощность на треть по сравнению с предыдущей версией, задействующей для генерации изображений 128 светодиодов. Эти параметры делают привлекательным использование HUD и в электрическом транспорте.

Компактный дизайн и низкая цена производства HUD соответствуют требованиям автомобилей среднего ценового диапазона. Инженеры Continental добились снижения размера, веса и цены, используя новую схему оптических

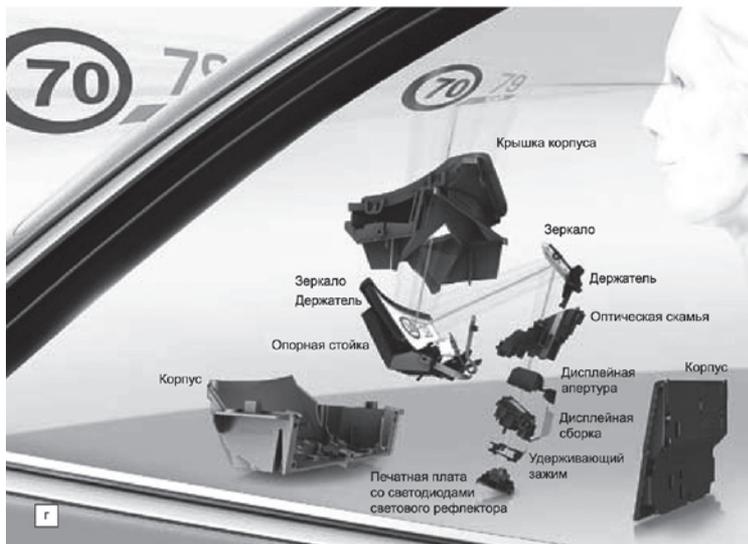


Рисунок 6 (в, г) – HUD второго поколения от Continental:
 в) возможности регулировки положения HUD в зависимости от положения сиденья водителя;
 г) ключевые составные части Continental HUD: светодиоды, зеркала и TFT-дисплей



Рисунок 7 а – Контактно-аналоговые HUD в автомобилях BMW: стандартные возможности

информация будет высвечиваться непосредственно в области, к которой она относится. В случае предупреждения о небезопасной дистанции, автомобиль впереди может быть высвечен, например, красным цветом. Водитель может видеть предупреждение о выходе из своей полосы или указатели навигационной системы непосредственно на дороге.

Контактно-аналоговые дисплеи с дополненной реальностью

BMW была первым европейским автопроизводителем, адаптировавшим технологию авиационных дисплеев HUD для автомоби-

лей, производимых в массовых объемах (рисунок 1 в, рисунок 6 а). С января 2004 года данная инновационная система помощи водителя стала составной частью BMW ConnectedDrive для BMW 5 серии. Дальнейшие разработки и оптимизация системы привели к тому, что в настоящее время полноцветные дисплеи HUD доступны, как опция, почти для всех серий автомобилей.

По графическому представлению, функциональности и гибкости новая технология HUD-дисплеев BMW привносит значительный вклад в активную безопасность посредством отображения релевантной информации с разрешением высокого качества в пределах поля зрения водителя, который может не отрывать взгляд от дороги (рисунок 7 а).

С дисплеем HUD время, требуемое для восприятия информации водителем, снижается более, чем вдвое. Яркость изображения регулируется в зависимости от окружающей обстановки. Воспроизведение данных осуществляется посредством интенсивного светового источника, который локализован в приборной панели и светит через полупрозрачный TFT-дисплей, изображение передается к ветровому стеклу посредством специально сформированных зеркал.

Continental также связывает будущие разработки HUD с дополненной реальностью. Согласно концепции Augmented Reality, отображаемые на виртуальном экране, поступающие от систем навигации и помощи водителю, данные сливаются с фактическим полем зрения так, что важная

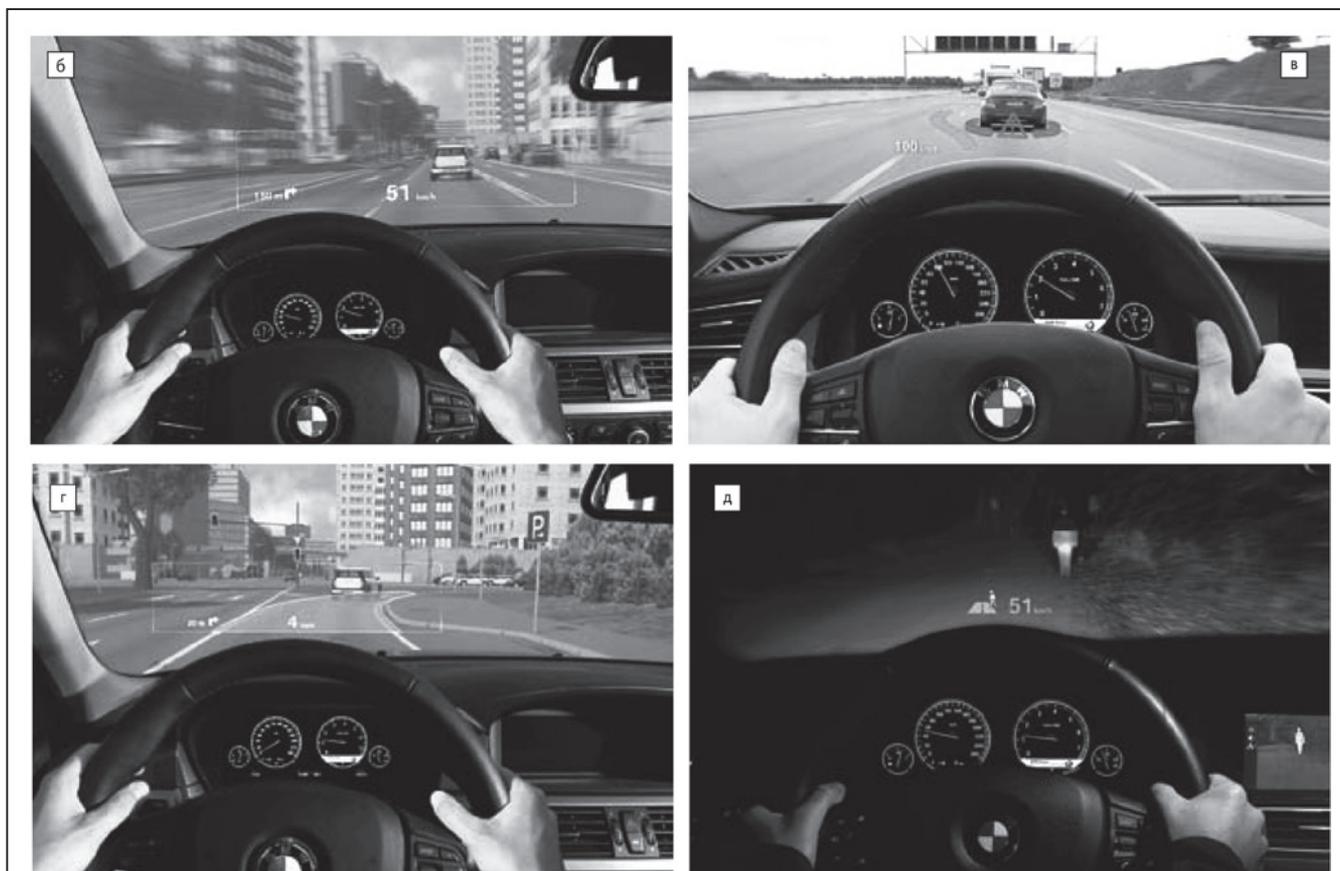


Рисунок 7 (б–д) – Контактно-аналоговые HUD в автомобилях BMW:
 б) применение технологии для привязки к объектам;
 в) вычисление маневра: виртуальный маркер, привязанный к реальному объекту;
 г) виртуальный маркер, наложенный поверх дороги;
 д) отображение информации от систем ночного видения

Вследствие выпуклой формы и определенных физических свойств использование ветрового стекла, как рефлектора, является сложным процессом. В ветровом стекле световой путь преломляется, что может давать двойные изображения. BMW применяет сверхтонкую фольгу, интегрированную в ветровое стекло, что гарантирует безукоризненное представление проецированных изображений.

В опубликованном документе BMW Innovation Days Connected 2011 содержится информация о разработке следующего поколения HUD с контактно-аналоговой функциональностью. В контактных аналоговых дисплеях дополнительная информация интегрирована в корректной перспективе и привязана точно к тем внешним объектам, к которым она относится (рисунок 7).

В смартфонах и мобильном мире контактный аналоговый метод является опорой для дополненной реальности и может интегрироваться в различные мобильные приложения. В автомобилях BMW данный метод реализуется посредством отметок, которые накладываются на реальные объекты во внешнем окружении, так что навигационная информация от систем помощи водителю может отображаться в точках адекватно водителю восприятию дорожной сцены. Навигационные инструкции могут смешиваться с дорогой, а автомобили или объекты, имеющие отношение к безопасности, высвечиваются или маркируются в контексте (рисунок 7 а–г).

HUD-дисплеи могут применяться также для отображения информации от систем ночного видения (рисунок 7 д).

Audi также предлагает различные дисплейные и другие технологии, позволяющие водителю сохранять контроль над ситуацией (рисунок 8 а). Следующее поколение данных устройств, продемонстрированное на CES 2012 (рисунок 7 б), является значительным шагом вперед. Когда автомобиль Audi приближается к перекрестку с активированной навигационной системой, водитель видит прозрачные стрелки маршрута за пределами автомобиля, позиционированные точно на реальном перекрестке. При приближении перекрестка стрелки увеличиваются. На расстоянии порядка 10 м они выглядят, как реальные физические стрелки на дороге. В холмистой местности навигационная стрелка может показывать направление дороги за верхом холма. При вождении с активным круиз-контролем отображается расстояние до автомобиля, едущего впереди. Если активна система ночного видения, и она регистрирует пешехода, идущего по улице, то дисплей показывает направление движения пешехода и расстояние до него. В контактно-аналоговом HUD проекционный 3D-цилиндр для глаз локализован несколько выше на ветровом стекле, чем в современном варианте, и имеет больший размер.

HUD будущего от Audi будут способны показывать информацию водителю или пассажиру на переднем сиденье,

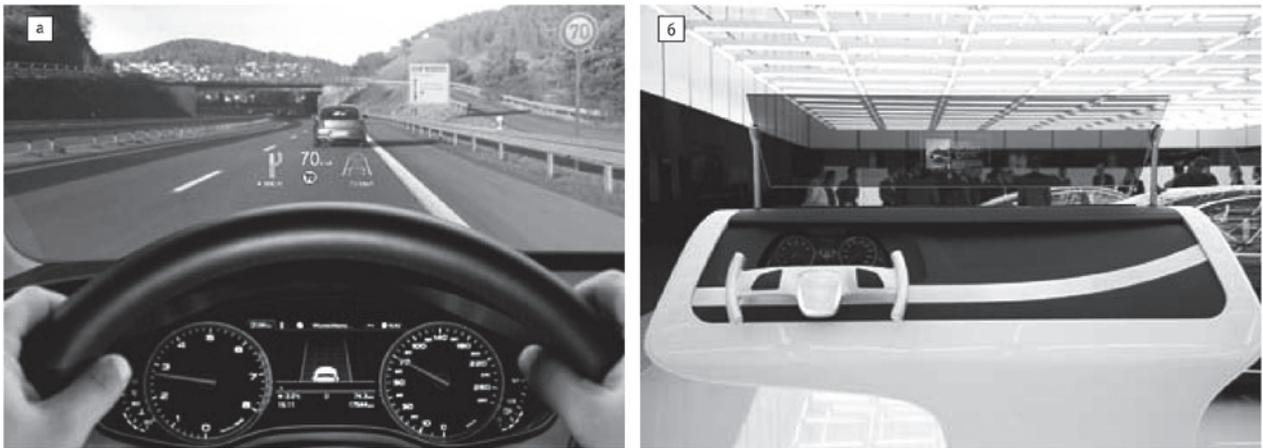


Рисунок 8 – Решения Audi:
 а) серийно выпускаемые HUD 2-го поколения от Continental в автомобилях А6 и А7;
 б) демонстрация технологии HUD Audi на CES 2012

а также всем пассажирам автомобиля. В данном сценарии водитель и пассажир будут иметь индивидуальный HUD, показывающий попутные цифровые руководства, новости или изображения видео-звонков. Дисплей, видимый для всех пассажиров, – это третья, центральная проекция на ветровом стекле.

Информация для водителя отображается посредством контактно-аналогового дисплея в форме символов, не-

подвижных изображений, простой анимации. В это время пассажир на переднем сиденье сможет пользоваться видео-функциями. Изображения, которые он видит, формируются с использованием технологии TI DLP (Digital Light Processing) с высокой яркостью и контрастностью.

Audi планирует реализовать в HUD идею распознавания жестов. Если пассажир, находящийся на переднем сиденье, видит в навигаторе интересующее его место, он может

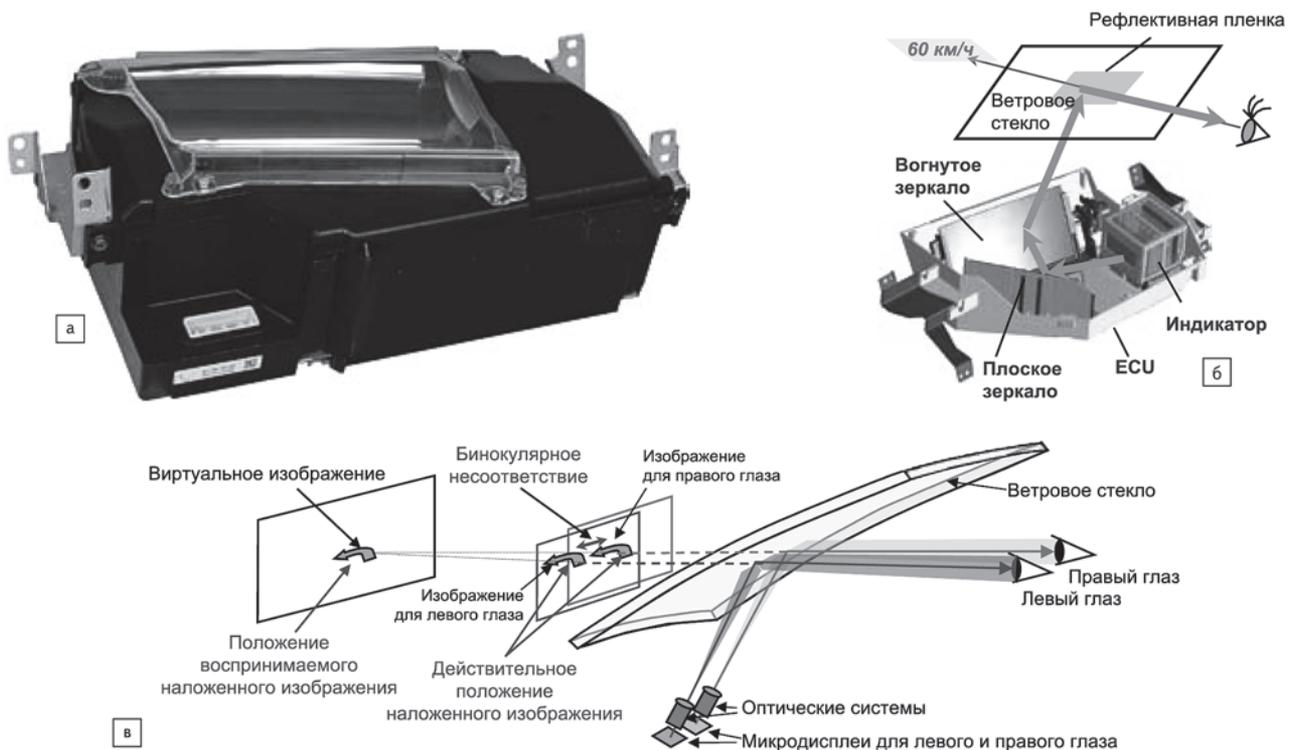


Рисунок 9 – Технологии DENSO:
 а) внешний вид блоков HUD для OEM;
 б) текущая структура HUD;
 в) дисплейная оптическая система на ветровом стекле следующего поколения, использующая два проекционных дисплея – для левого и правого глаза



Рисунок 10 – Иллюстрация лазерной голографической технологии HUD Delphi

простым движением (жестом) перебросить информацию на центральный экран для водителя, который также может аналогичным жестом осуществить переброс изображения на свой HUD для добавления места назначения в навигационную систему, контролирующую маршрут. Миниатюрная камера регистрирует движения пассажира и водителя и направляет системе соответствующие сигналы.

Audi работает над технологией multi-touch для HUD, сходной с технологией, применяемой в современных смартфонах и планшетах, которая, в частности, ускоряет использование списков, карт, другой информации. Еще один проект компании сосредоточен на улучшении голосового контроля.

DENSO:

системы ночного видения и 3D HUD

DENSO Corporation (www.globaldenso.com) поставляет широкий набор вариантов интерфейса человек-машина (HMI), включая комбинации приборов, системы развлечений, удаленные Touch-контроллеры и дисплеи HUD. Разработку автомобильных систем навигации компания начала еще в 1980-х годах, а первый HUD был представлен в 1991 году.

В 2003 году DENSO представила первый в мире HUD, который по выбору показывает информацию, например, скорость автомобиля, или изображения систем ночного видения. Система ночного видения излучает свет, близкий к инфракрасному, вперед по направлению движения автомобиля и фиксирует отраженное от препятствия излучение. Это создает изображение на экране HUD, что помогает водителям в ночное время.

Текущие HUD DENSO используют компактный проекционный блок, обеспечивающий высокую яркость при минимизированной потребляемой мощности (рисунок 9 а, б).

Следующее поколение приборов DENSO – дисплей на ветровом стекле (рисунок 9 в), представляющей собой комбинацию HUD, сенсорной технологии и интерфейса человек-машина. Для такого устройства компания разработала новую оптическую систему (рисунок 9 в), в которой задействованы два проекционных дисплея – для левого и правого глаза, что допускает контроль положения 3D-изображений. Например, водитель может видеть стрелку, указывающую направление для навигации, наложенную на вид дороги впереди.

Особенности HUD DENSO:

- обеспечение водителей информацией в удобном представлении, повышающем степень безопасности вождения. Благодаря HUD расстояние между глазами водителя и изображением больше, чем в случае с обычной приборной панелью, что облегчает фокусировку;

- вогнутое зеркало использует точки прогрессивной фокусировки (разработка DENSO) для обеспечения высококачественного изображения без дисторсий;

- индикатор излучает свет однородно с высокой яркостью, что позволяет водителям видеть виртуальное изображение даже в дневное время.

Данная технология реализована на основе опыта DENSO в симуляции световой проекции и технологии дизайна и производства оптических компонентов. Компания работает над тем, чтобы сделать HUD полноцветным и увеличить размер изображения на ветровом стекле.

Delphi:

лазерные голографические HUD

Компания Delphi Corporation (www.delphi.com) в сотрудничестве с французской научно-исследовательской организацией Vehicle of the Future разработала HUD на основе лазерной технологии и предлагает ее автопроизводителям для использования в составе информационных систем водителя (рисунок 10). Преимущества системы – малый корпус, гибкость интеграции, адаптация к различным размерам автомобилей, низкая цена, допускающая применение HUD в недорогих автомобилях.

Показ информации в 20-градусной области обзора минимизирует отвлечение водителя от дороги и повышает безопасность.

Виртуальное изображение отличается цветовой насыщенностью (монохромные и полноцветные опции, поддерживается до четырех цветов), высокой контрастностью с автоматической регулировкой для дневного и ночного времени. Вес и потребление мощности снижены на 10 %, а занимаемое место на 30 % меньше, чем у текущих систем. Дизайн масштабируемый и допускает апгрейд.

Системная интеграция допускает отображение ключевых данных от других автомобильных систем: предупреждения о столкновении, комбинации приборов, навигационной системы, развлечений.

Volvo:

технологии HUD для защиты пешеходов

Компания Volvo (www.volvo.com) адаптировала применение технологии HUD для защиты пешеходов. Иллюстрация работы функции распознавания пешеходов с полным автоматическим торможением показана на рисунок 11 а, б. В чрезвычайной ситуации водитель получает аудио-предупреждение вместе с мерцающим светом на ветровом стекле. При отсутствии реакции водителя торможение осуществляется автоматически.

Компания Volvo Cars занималась разработкой системы обнаружения пешеходов с автоматическим торможением более пяти лет. Тестовые автомобили исследовались при всевозможных вариантах в различных условиях дорожного движения и климата. Данные технологии реализованы в моделях Volvo V60 и V40 (рисунок 11 б).

В автомобиле V40 реализовано предупреждение о небезопасной дистанции Distance Alert. Посредством кнопок на рулевом колесе возможен выбор одной из пяти настроек, которая



Рисунок 11 – HUD от Volvo в системах защиты пешеходов и другие дисплейные технологии:
 а) принцип работы функции распознавания пешеходов с полным автоматическим торможением;
 б) обнаружение пешеходов посредством камеры переднего обзора, вывод информации на HUD (мерцающий свет) и Volvo Sensus (предупреждение об автоматическом торможении);
 в) touch-экран Concept You, связанный с HUD

и отображается на дисплее спидометра. Свет на HUD информирует водителя, когда расстояние значительно сокращается. В этой же модели установлен Volvo Sensus – новый графический приборный кластер, позволяющий персонализировать приборные схемы и выводимую на приборный дисплей информацию, например, о пешеходах (рисунок 11 б) и дорожных знаках.

В автомобиле Concept You экран в центральной консоли остается в спящем режиме, пока водитель не посмотрит на него (рисунок 11 в). Скрытая ИК-камера регистрирует движение глаз, и информация отображается на экране. Другие датчики регистрируют движение рук и переключают системы развлечения. Водитель может перебрасывать данные от центрального touch-экрана на HUD и обмениваться ими с пассажирами.

Рассмотренные примеры показывают, что HUD только в редких случаях может рассматриваться отдельно от остальных систем. Это подтверждает и следующий пример, которым является применение технологии HUD для навигации от Pioneer.

Pioneer CYBER NAVI – первая коммерческая лазерная система навигации, включающая HUD с дополненной реальностью

Pioneer Corporation (www.pioneer.jp) выпустила первую в мире автомобильную GPS систему навигации, которая поставляется с блоком AR-HUD – первым в мире лазерным HUD, проецирующим информацию дополненной реальности (Augmented Reality, AR) на ветровое стекло (рисунок 12 а). С Cyber Navi езда на автомобиле

напоминает фантастическое приключение, в чем можно убедиться, посмотрев захватывающие видеоролики, выложенные в Интернете [1, 2].

В июле 2012 года на японский рынок были выпущены две модели Cyber Navi: AVIC-VH99HUD (high-end класса в двойном 1-DIN блоке, 3,770 долл. США) и AVIC-ZH99HUD (в блоке 2-DIN, 4,021 долл. США). Обе модели (рисунок 12 б) основаны на лазерных HUD и технологии PicoP от MicroVision (www.microvision.com) и включают LCD-экран в центральной консоли и блок AR-HUD. Дисплей представляет собой лист прозрачного пластика, который крепится в поле зрения водителя напротив лобового стекла. Для осуществления обмена данными между блоком HUD и навигационной системой используется канал Bluetooth.

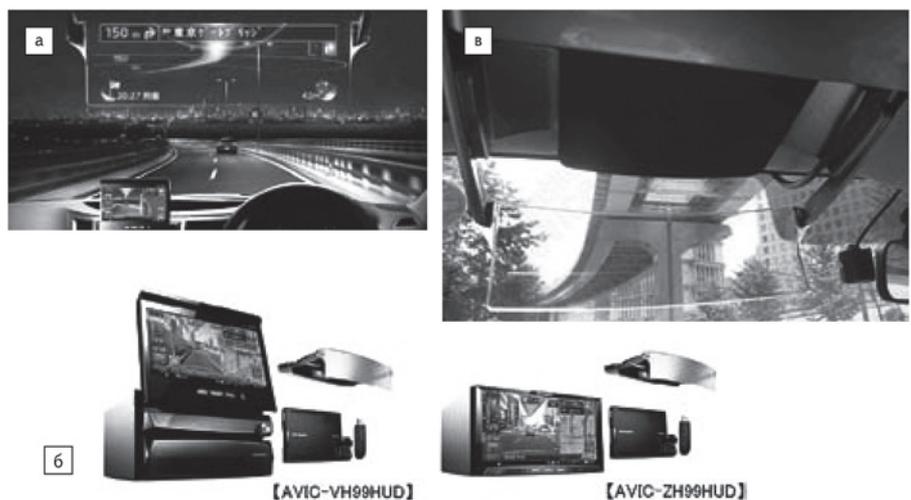


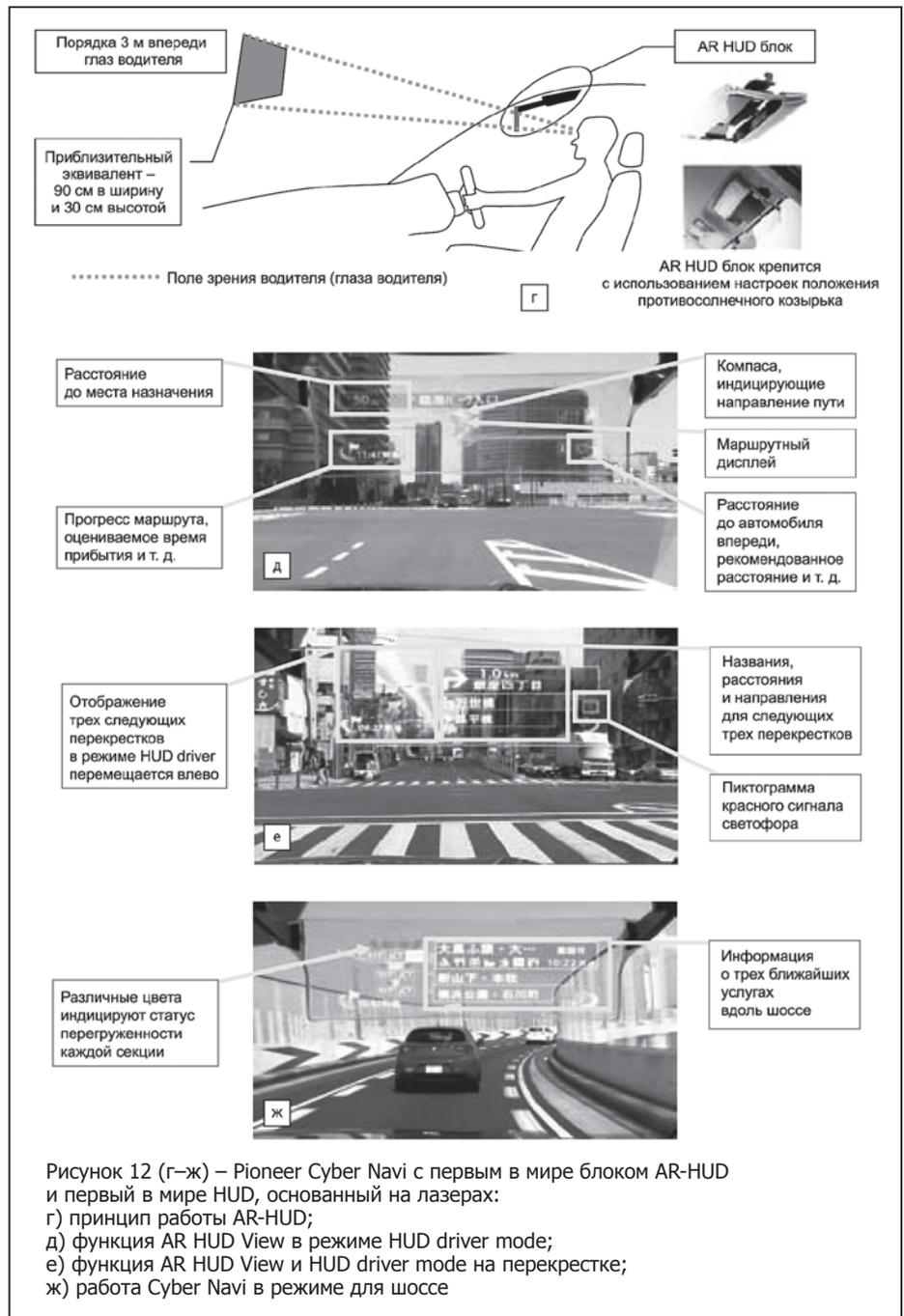
Рисунок 12 (а-в) – Pioneer Cyber Navi с первым в мире блоком AR-HUD и первый в мире HUD, основанный на лазерах:
 а) навигационная система Cyber Navi;
 б) AVIC-VH99HUD и AVIC-ZH99HUD;
 в) вид на HUD из кабины

Проекторный модуль AR-HUD устанавливается в положение противосолнечного козырька сбоку сиденья водителя, что позволит обеспечить установку системы более чем на 70 % существующих автомобилей (рисунок 12 в). В целях безопасности Pioneer использует резиновый проекционный экран, чтобы исключить риск травмы головы в случае аварии. Изображение на HUD формируется посредством сканирующих МЭМС-зеркал, проецирующих лазерные лучи трех базовых цветов пространства RGB. В результате создается виртуальный дисплей диагональю около 37" (90×30 см), который проецирует информацию на виртуальном расстоянии порядка 3 м от глаз водителя (рисунок 12 г). На виртуальном экране с разрешением 720×260 пикселей отлично читаются пиктограммы и информация. Напротив ветрового стекла поверх реальной сцены выводятся скорость, направление движения и другие важные данные. Блок AR-HUD позволяет отображать информацию с высоким уровнем контрастности, используя для этого лазерный RGB источник.

Новая технология PicoP Gen2 с использованием твердотельных зеленых лазеров позволила значительно повысить удобство восприятия водителем информации, отображаемой AR-HUD. Для взаимной релевантности функций и ситуаций Cyber Navi предлагает четыре типа режимов вождения (с сопровождающим удаленным контролем для переключения между ними): режим водителя, шоссейный, навигации с использованием карт и так называемый режим скутера.

Режим HUD driver mode (HUD для водителя) отображает информацию о расстоянии до автомобиля впереди и показывает маршрут к месту назначения (рисунок 12 д). В данном режиме возможен вывод информации о ситуациях на перекрестках (отображение списка пересечений HUD) (рисунок 12 е). Когда автомобиль останавливается на светофоре, дисплей автоматически переходит к показу соответствующей информации, например, руководства по направлению и названия следующих трех перекрестков. Блок также обнаруживает красные сигналы светофора и показывает соответствующую иконку. Когда включается зеленый свет или стоящее впереди ТС начинает движение, дисплей автоматически снова переходит в режим водителя HUD driver, активный при движении (рисунок 12 д).

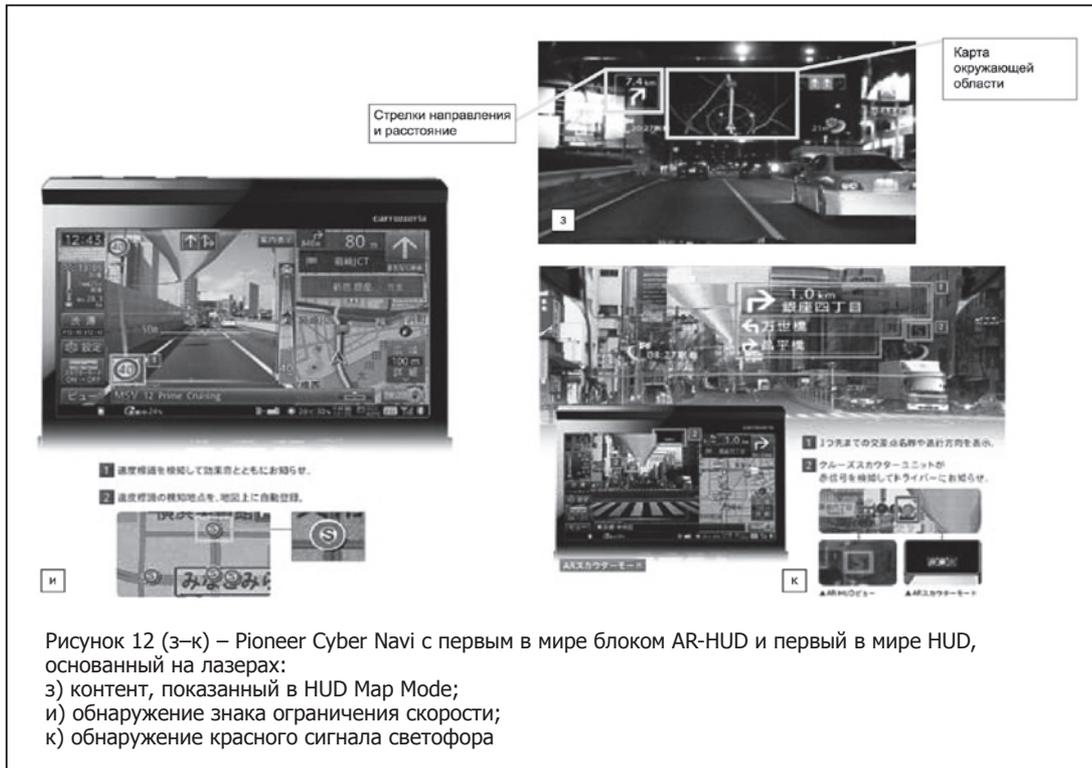
Режим HUD highway mode проиллюстрирован на рисунке 12 ж. Устройство автоматически переключается в него, когда автомобиль входит на автомагистраль/шоссе.



В данном режиме показывается расстояние, скорость, информация о зонах сервисов и услугах парковки. При этом отображается также дорожный статус с использованием различных цветов для индикации интенсивности движения и пробок на ближайших участках.

В режиме HUD Map Mode (рисунок 12 з) отображается карта местности вокруг автомобиля, указываются с использованием различных цветов платные, национальные и обычные дороги. Карта имеет шесть вариантов zoom – в масштабе от 200 м до 10 км. В данном режиме также выводится информация, относящаяся к полосе движения, стрелка направления, пиктограмма, индицирующая красный сигнал светофора.

Так называемый режим AR Scouter Mode дополнен новыми функциями, включая обнаружение знаков ограничения



congestion information охватывают в общей сложности 700 000 км дорог в Японии.

Как уже говорилось, с июля 2012 года японские автомобилисты уже могут использовать Cyber Navi – первую в мире коммерческую навигационную систему с дополненной реальностью, обеспечиваемой лазерным блоком AR-HUD. Ожидаются новые шаги в отношении коммерциализации со стороны разработчиков технологий, систем или новых применений HUD.

Заключение

Спектр ключевых применений и техно-

логии HUD быстро расширяется. Распространению в массовых сегментах и развитию технологий способствуют возможности HUD и связанных с ними систем идентифицировать и высвечивать в реальном времени наиболее важную информацию в форме, хорошо подходящей для быстрого и удобного восприятия водителем/пилотом. Продолжают развиваться наиболее перспективные технологии и отдельные направления, включая опции дополненной реальности, контекстную осведомленность, голосовой контроль, обнаружение жестов, 3D-визуализацию, виртуальные touch-дисплеи. В перспективе ожидается дальнейшее развитие систем лазерного проецирования, а также смежных и конкурирующих с ними HUD, HMD, мобильных проекционных дисплеев. Современный пользователь постепенно получает все те технологии, которые когда-то были разработаны для военной и гражданской авиации – спутниковые навигационные системы, блоки инерциальных измерений, интегрированные в сотовые телефоны и автомобили. А теперь к ним добавились HUD – технологии, которые когда-то были созданы для военных летчиков.

Режим AR Scouter характеризуется четкими указующими стрелками с бордюрами, которые отображаются, например, в том случае, если запланированному маршруту соответствует поворот за угол на перекрестке. Функция обнаружения красного сигнала светофора (рисунок 12 к) использует также пиктограмму для уведомления водителя о том, когда включается зеленый свет.

Функция обозревателя парковки Parking Watcher отображает степень загруженности автостоянок. Эта информация периодически актуализируется.

Функция Road Creator автоматически генерирует новые дороги на картах при фактическом прохождении их. Дороги создаются на картах с использованием точной технологии GPS-позиционирования Cyber Navi и затем включаются в поиск и навигацию.

Высококачественные поиски маршрутов посредством системы информирования о пробках Smart Loop traffic

логий HUD быстро расширяется. Распространению в массовых сегментах и развитию технологий способствуют возможности HUD и связанных с ними систем идентифицировать и высвечивать в реальном времени наиболее важную информацию в форме, хорошо подходящей для быстрого и удобного восприятия водителем/пилотом. Продолжают развиваться наиболее перспективные технологии и отдельные направления, включая опции дополненной реальности, контекстную осведомленность, голосовой контроль, обнаружение жестов, 3D-визуализацию, виртуальные touch-дисплеи. В перспективе ожидается дальнейшее развитие систем лазерного проецирования, а также смежных и конкурирующих с ними HUD, HMD, мобильных проекционных дисплеев. Современный пользователь постепенно получает все те технологии, которые когда-то были разработаны для военной и гражданской авиации – спутниковые навигационные системы, блоки инерциальных измерений, интегрированные в сотовые телефоны и автомобили. А теперь к ним добавились HUD – технологии, которые когда-то были созданы для военных летчиков.

Литература:

1. http://pioneer.jp/carrozzeria/cybernavi/avic_vh99hud_avic_zh99hud/?link=product_main_CYBERNAVI<http://pioneer.jp/press/2012/0508-1.html>
2. <http://youtu.be/RfZ4IHenItE>
3. Сысова, С. МОЭМС – доступные технологии генерации и сканирования оптической информации / С. Сысова. – Компоненты и технологии. – 2010. – № 8–9.
4. www.boeing.com
5. www.dassault.fr
6. www.elbitsystems.com
7. www.lockheedmartin.com
8. www.mdc.com

ВКЛЮЧЕНИЕ СВЕТОДИОДНОГО ОСВЕЩЕНИЯ НИЗКОЙ МОЩНОСТИ: МЕТОД УПРАВЛЕНИЯ ПИТАНИЕМ

**Ezana Haile, Principal Applications Engineer,
Analog & Interface Microchip Technology Inc.**

Интенсивность излучения светодиодов, более яркого по сравнению с традиционными электрическими источниками света, является существенным фактором при определении качества видимого света. Включение всех светодиодов, входящих в светильники при максимальном световом излучении невозможно из-за ограничения мощности источника питания. Чтобы получить оптимальную яркость, необходимо эффективно управлять мощностью рассеяния на светодиодах, так как источник питания является маломощным.

Для этого требуется использовать систему управления питанием, которая обеспечивает включение группы светодиодов в определенные интервалы времени. Управляемые группы светодиодов останутся визуально необнаруженными, поскольку будет достигнута требуемая интенсивность излучения. Должна быть установлена необходимая мощность источника питания и интенсивность светового потока, чтобы определить число светодиодов для данного временного интервала. Требуемый уровень яркости и характеристики прямого тока позволяют выбрать необходимое число светодиодов.

Число светодиодов в группе равно отношению максимального тока источника питания к суммарному току светодиодов.

Частота переключения группы светодиодов в режиме вкл/выкл должна быть выбрана такой, чтобы время свечения было достаточно длительным для максимальной

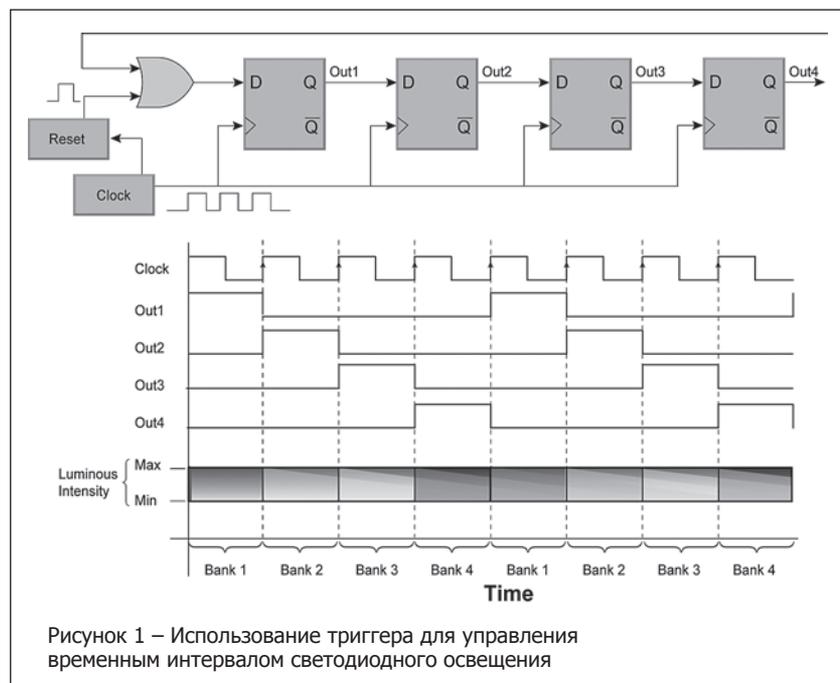
интенсивности светового потока группы светодиодов, и ограничено в режиме «выкл», чтобы не было колебаний среднего уровня яркости. Это ограничивает число дополнительных групп светодиодов, которыми можно управлять, изменяя временной интервал. Данный метод может быть реализован устройством при низкой стоимости тактовых генераторов и триггеров, которое будет управлять группами светодиодов и логическими элементами. Рисунок 1 показывает блок-схему конфигурации D-триггера для управления четырьмя группами светодиодов.

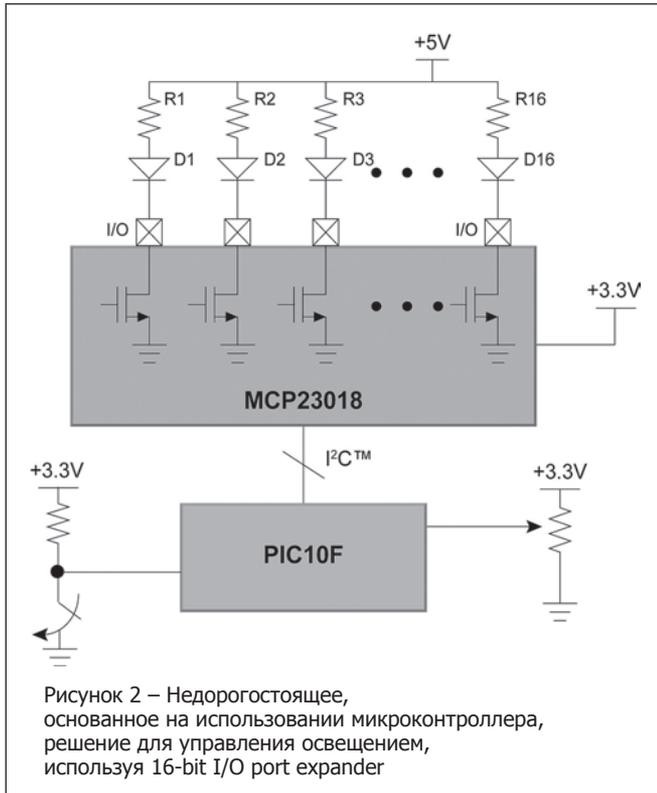
Чтобы запустить триггер, находящийся в исходном состоянии, требуется на его вход подать первый пусковой импульс от тактового генератора. Длительность этого импульса не может быть больше, чем один такт. Группа светодиодов с нарастающей интенсивностью увеличивает яркость, и человеческий глаз не успевает заметить разницу в интенсивности излучения, и оно воспринимается как излучение четырех групп одновременно, как будто все они включены постоянно.

Реализация этого алгоритма не обеспечивает конструкторскую гибкость. Эта схема может быть использована для жидкокристаллической подсветки, но при потускнении или изменении режима генерации излучения необходим микроконтроллер (MCU) – базовая схема предоставляет возможность большой гибкости при минимальном влиянии на общую стоимость устройства.

Она более проста и позволяет создать устройство управления с меньшим количеством компонентов, поскольку MCU управляет каждой группой светодиодов и позволяет вводить данные пользователя для управления серым цветом и выбором светового рисунка.

Дешевый 8-разрядный микроконтроллер семейства PIC10F или PIC12F с расширителем порта I/O типа MCP23018 позволяет реализовать эту идею. Расширители ввода-вывода могут также быть полезными для управления светодиодами, когда схема освещения удаленно расположена относительно MCU. Расширитель ввода-вывода I/O увеличивает возможность портов I/O микроконтроллера. В этом приложении MCU управляет портами расширителя ввода-вывода через протокол I²C™, чтобы включать или выключать светодиоды, а контакты ввода-вывода MCU могут использоваться, чтобы вводить данные пользователем через кнопочный выключатель, или использовать встроен-





ный аналого-цифровой преобразователь для регулировки затемнения светодиодов. Микроконтроллеры могут работать при напряжении 3,3 В или ниже. Преимущество такого решения состоит в том, что оно позволяет работать светодиодам при напряжении выше 5 В, в то время, как микроконтроллер и Port expander I/O работают при более низком напряжении. MCP23018 – 16-разрядный Port expander I/O с открытым выходом и интерфейсом I²C.

Когда уровень сигнала на входе порта I/O низкий, тогда напряжение порта расширителя ввода-вывода равно 0 В и электрический ток прямого смещения включает светодиоды. Резистор, включенный последовательно с группой светодиодов, ограничивает протекающий через него прямой ток, необходимый для получения заданной яркости и одновременно является нагрузочным резистором для открытого выхода порта I/O. Когда уровень сигнала на входе порта I/O высокий, вывод отключен (высокий импеданс), напряжение на нагрузочном резисторе порта расширителя ввода-вывода возрастает до 5 В, и ток не протекает через светодиод. Конфигурация открытого вывода имеет то преимущество, что когда порт сконфигурирован, как высокий

импеданс, увеличивается время разряда паразитной емкости светодиода. Это позволяет следующей группе светодиодов включиться на более длительное время, по сравнению с подобными приложениями с двухтактным выводом.

Порты MCP23018's 16 I/O могут управлять 16 светодиодами. Суммарный ток, который может проходить в порт I/O, когда светодиод полностью включен, может быть ограничен выходным током расширителя ввода-вывода. Низкоуровневое напряжение порта I/O не должно превышать 0,6 В при максимальном токе 8,5 мА. Если ток больше 8,5 мА, то низкоуровневое напряжение по экспоненте увеличится. Текущий абсолютный максимум составляет 25 мА. Если, например, ток источника ограничен 50 мА, а 2 мА планируются для микроконтроллера, расширителя ввода-вывода и резисторов ввода данных пользователем, то оставшая часть доступного тока может быть выделена для светодиодного освещения. В группе могут управляться 4 светодиода, если ток через светодиод равен приблизительно 10 мА. В этом примере значение токоограничивающего резистора будет приблизительно 440 Ом. Временная диаграмма, показанная в рисунке 1, может быть зациклена, используя короткую систему команд MCU. Например, в соответствии с Pseudo Code 1, основная подпрограмма может быть бесконечным циклом с условием продолжения. В интервалах таймера MCU (Timer0) процедура обработки прерывания отправляет команду I²C в расширитель ввода-вывода, чтобы включить только одну группу светодиодов. Процедура обработки прерывания отслеживает состояние группы светодиодов, которая в настоящее время включена, и устанавливает соответствующую конфигурацию порта расширителя ввода-вывода, как высокую или низкую.

```

void main (Void){
    initialize(); //initialize the PICmicro peripherals including Timer0, and I/O expander peripherals
    bank_counter = 0; //Clear the bank counter variable
    while(1){} //infinite while loop
}

Void Interrupt int_service(void){
    TurnOffAll_LEDs(); //subroutine to turn off all previously lit LEDs
    I2C_start(); //I2C protocol – start signal subroutine
    I2C_send(Comand_byte); //I2C protocol – send byte subroutine

    If(bank_counter == bank1){ // bank1 is a constant defined as 0
        I2C_send(Address_pionter_bank1n2); //Send peripheral register address pointer
        I2C_send(*0000 1111*); //I2C protocol – send bank status (bank 1 on and bank 2 off)
        bank_counter = bank_counter + 1;
    }
    If(bank_counter == bank2){ // bank1 is a constant defined as 1
        I2C_send(Address_pionter_bank1n2); //Send peripheral register address pointer
        I2C_send(*1111 0000*); //I2C protocol – send bank status (bank 2 on and bank 1 off)
        bank_counter = bank_counter + 1;
    }
    If(bank_counter == bank3){ // bank1 is a constant defined as 2
        I2C_send(Address_pionter_bank3n4); //Send peripheral register address pointer
        I2C_send(*0000 1111*); //I2C protocol – send bank status (bank 3 on and bank 4 off)
        bank_counter = bank_counter + 1;
    }
    If(bank_counter == bank4){ // bank1 is a constant defined as 3
        I2C_send(Address_pionter_bank3n4); //Send peripheral register address pointer
        I2C_send(*1111 0000*); //I2C protocol – send bank status (bank 4 on and bank 3 off)
        bank_counter = 0;
    }
    I2C_stop(); //I2C protocol – stop signal subroutine
}
    
```

Pseudo Code 1: Microcontroller interrupt service routine code flow

```

*
*
*
If(bank_counter == bank1){ // bank1 is a constant defined as 0
    I2C_send(Address_pionter_bank1n2); //Send peripheral register address pointer
    Bank1n2_pattern = ~Bank1n2_pattern; //complement the variable using '~'
    I2C_send((Bank1n2_pattern | '0000 1111')); //I2C protocol – send bank status
    // (use OR '|' to set bank 1 on and bank 2 off)
    bank_counter = bank_counter + 1;
}
*
*
*
Pseudo Code 2: Pattern generation

```

```

*
*
*
Void Interrupt int_service(void){
    TurnOffAll_LEDs(); //subroutine to turn off all previously lit LEDs
    If (Delay_ON == ON){ // ON is a constant defined as 1
        Update_Timer0_Counter(Timer0delay_interval); //subroutine to update Timer0
        // Timer0delay_interval is a constant to set the minimum delay
        Delay_Counter = Delay_Counter - 1; //count down the number of interrupts for delay
        If(Delay_Counter == 0){
            Delay_ON = OFF; // Clear the delay flag for the next interrupt
            Delay_Counter = Get_Delay_Counter(); //subroutine to detect user input
            //and set the delay counter variable
        }else{
            Delay_ON = ON; // leave delay flag ON
            Return; //exit the interrupt service routine
        }
    }

    I2C_start(); //I2C protocol start signal subroutine
    I2C_send(Comand_byte); //I2C protocol – send byte subroutine
    If(bank_counter == bank1){ // bank1 is a constant defined as 0
        I2C_send(Address_pionter_bank1n2); //Send peripheral register address pointer
        Bank1n2_pattern = ~Bank1n2_pattern; //complement the variable using '~'
        I2C_send((Bank1n2_pattern | '0000 1111')); //I2C protocol – send bank status
        // (use OR '|' to set bank 1 on and bank 2 off)
        bank_counter = bank_counter + 1;
        Delay_ON = ON; // turn on the delay flag
    }
}
*
*
*

```

Pseudo Code 3: delay code implementation to view the lighting pattern

```

Void get_PWM_ratio(void) {
    Double PotScale; //local variable to store ADC output scale
    Double PWM_Percentage; //local variable to store PWM change percentage

    PotScale = (ADRESH)/16 + 1; //Scale ADC output high byte
    PWM_Percentage = 1/16 * PotScale; //Scale the output from from 0 to 1, equivalent to 100%
    PWM_High = Frequency_counter * PWM_Percentage; //set PWM high Timer0 value
    PWM_Low = Frequency_counter * (1 - PWM_Percentage); //set PWM low Timer0 value
    //PWM_High and PWM_Low are global variables,
    //and Frequency_counter is a constant Timer0 value to set frequency.
}

```

Pseudo Code 4: Subroutine to calculate PWM ratio

Первоначально, или после сбросов MCU, на первом прерывании Timer0, группа 1 включена. На втором прерывании группа 1 выключена и включается группа 2. Затем на третьем прерывании группа 2 выключается и включается группа 3. Наконец, на четвертом прерывании выключается группа 3 и включается группа 4. Счетчик групп очищен. На следующем прерывании цикл повторяется. Другие переменные, такие как Command_byte, Address_pointer_bank1n2 и Address_pointer_bank3n4 являются константами, определенными в заголовочном файле. С такой реализацией схема обеспечивает требуемую силу света, и кажется, что все светодиоды включены одновременно.

Режим динамического управления LED

Интересные образцы освещения могут быть легко сгенерированы с основанным на микроконтроллере приложением. Выводы расширителя порта I/O загружены 2 байтами переменных RAM, маркированных как Bank1n2_pattern и Bank3n4_pattern. Каждый байт соответствует каждой группе светодиодов. Предопределенная таблица режимов содержит различные временные интервалы (1 и 0 с) для каждой группы. Постоянный контроль кнопочного переключателя вкл\выкл осуществляется добавлением оператора THEN к коду MCU – в бесконечном цикле при условии продолжения основной подпрограммы. Когда кнопка на мгновение нажата, код из таблицы поиска, загруженный в виде двух байтов RAM, маркированных как Bank1n2_pattern и Bank3n4_pattern. Когда происходит прерывание Timer0, новый код отправляется в расширитель ввода-вывода, и светодиоды светят согласно его значению. Основные циклы подпрограммы осуществляются через таблицу поиска, поскольку кнопочный переключатель продолжает на мгновение нажиматься пользователем. Чтобы получить переменное свечение светодиодов отправьте дополнение предыдущего кода. Например, если код bank1 равняется «0101», тогда дополнение – «1010», как показано в Pseudo Code 2, является отрывком от Pseudo Code 1.

Однако задержка должна быть добавлена в подпрограмме прерывания перед группой вкл\выкл, чтобы визуально осмотреть переменное свечение. Это создается с флагом Delay_ON так, чтобы на следующем прерывании счетчик считал в обратном порядке число прерываний для задержки. Значение задержки может также быть выбрано пользователями, используя потенциометр, где движок соединен с аналого-цифровым преобразователем MCU на микросхеме (ADC). Цифровые данные ADC масштабируются от минимума до максимальной задержки, обнаруживая лучшие четыре бита, которые обеспечивают 16 уровней. Лучшее отношение может быть установлено, обнаруживая лучшие пять битов или 32 уровня. Максимальная задержка является самой медленной, светодиоды могут мигнуть с минимальной задержкой и кажется, что все они полностью включены. Временная диаграмма на рисунке 3 показывает позицию задержки t_{DELAY} .

Чтобы управлять продолжительностью временного интервала каждой группы, тусклое управление использует широтно-импульсную модуляцию (PWM). У продолжительности прерывания Timer0 есть два значения: один на высокое время и другой на низкое время, пропорциональное отношению процента PWM. Уровень разрешения может быть скорректирован, выбирая лучшие 4 или 5 битов из ADC. Позиция счетчика Timer0 скорректирована метрически с масштабируемыми данными ADC, где 100 % PWM соответствует полному свечению, и 0 % минимальному или самому низкому тусклому свечению. Pseudo Code 4 показывает PWM, масштабирование для 16 уровней ADC, и рисунок 3 показывает временную диаграмму для t_{PWM_LOW} и t_{PWM_HIGH} .

Процедура обработки прерывания должна обновить позиции счетчика Timer0 для следующего прерывания. Это также должно показать, является ли продолжительность для PWM высокой или низкой. Поэтому, несколько инструкций должны быть добавлены, чтобы обнаружить значение кода для тусклого управления. Timer0 отвечает за позиции PWM_High и PWM_Low, как показано в Pseudo Code 4. Pseudo Code 5 показывает код для того, чтобы скорректировать PWM; оператор THEN используется, чтобы обнаружить состояние PWM.

Эта методология может быть реализована на среднем 8-разрядном микроконтроллере с дополнительной памятью,

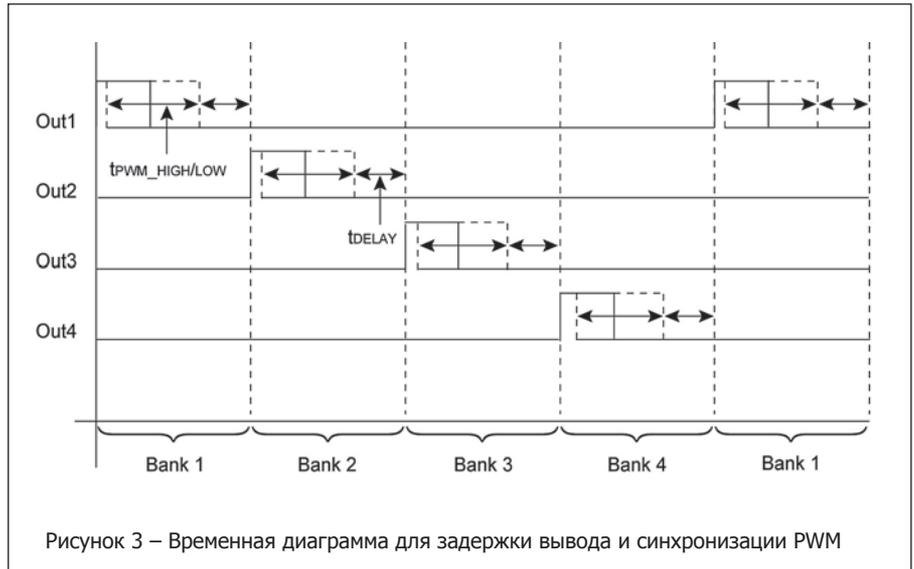


Рисунок 3 – Временная диаграмма для задержки вывода и синхронизации PWM

```

*
*
*
If(bank_counter == bank1){ // bank1 is a constant defined as 0
    If (PWM_High_Low_flag == OFF){ //check PWM status flag
        get_PWM_ratio(); //Detect user input for PWM ratio
        Timer0_counter = 65535 - PWM_High; //set Timer0 counter variable
        Update_Timer0_Counter(Timer0_counter); //subroutine to update Timer0

        I2C_send(Address_pionter_bank1n2); //Send peripheral register address pointer
        Bank1n2_pattern = ~Bank1n2_pattern; //complement the variable using '~'
        I2C_send((Bank1n2_pattern | '0000 1111')); //I2C protocol – send bank status
        //(use OR '|' to set bank 1 on and bank 2 off)

        PWM_High_Low_flag = ON; //Set flag
        Delay_ON = OFF;

    }else{
        Timer0_counter = 65535 - PWM_Low; //set Timer0 counter variable
        Update_Timer0_Counter(Timer0_counter); //subroutine to update Timer0

        I2C_send(Address_pionter_bank1n2); //Send peripheral register address pointer
        I2C_send('1111 1111'); //I2C protocol – turnoff all LEDs
        bank_counter = bank_counter + 1;
        PWM_High_Low_flag = OFF; //Clear flag
        Delay_ON = ON;

    }
}
*
*
*

```

Pseudo Code 5: Code flow for PWM ratio setting

таким, как из семейства микрочипов PIC16F. Обновление позволило бы основной подпрограмме обрабатывать сложные образцы освещения, такие как бегущие огни.

Разработчики всегда ищут новые способы сократить издержки, не ставя под угрозу Производительность устройств. Есть много методов, чтобы эффективно управлять группами светодиодов, используемых в жидкокристаллических подсветках, или включать оригинальную иллюминацию.



OSTIS-2014

IV МЕЖДУНАРОДНАЯ НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ «ОТКРЫТЫЕ СЕМАНТИЧЕСКИЕ ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ»

**Open Semantic Technologies for Intelligent Systems
20–22 февраля 2014 г. Минск. Республика Беларусь**

ИНФОРМАЦИОННОЕ СООБЩЕНИЕ

Приглашаем принять участие в IV Международной научно-технической конференции «Открытые семантические технологии проектирования интеллектуальных систем» (OSTIS-2014), которая будет посвящена 90-летию Виктора Владимировича Мартынова. Конференция OSTIS-2014 открывает цикл научных мероприятий приуроченных 50-летию основания Белорусского государственного университета информатики и радиоэлектроники.

Конференция пройдет в период с 20 по 22 февраля 2014 года в Белорусском государственном университете информатики и радиоэлектроники, г. Минск, Республика Беларусь.

Рабочие языки конференции: русский, белорусский, английский.

ОСНОВНЫЕ ОРГАНИЗАТОРЫ КОНФЕРЕНЦИИ

- Российская ассоциация искусственного интеллекта (РАИИ);
- Белорусский государственный университет информатики и радиоэлектроники (БГУИР);
- Государственное учреждение «Администрация Парка высоких технологий» (Республика Беларусь);
- Объединенный институт проблем информатики Национальной академии наук Беларуси (ОИПИ НАН Беларуси);
- Тверской государственный технический университет;
- Научно-исследовательский институт «Прикладная семиотика» АН РТ.

ПРОГРАММНЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

Боргест Н.М., к.т.н., доц., РФ
 Гаврилова Т.А., д.т.н., проф., РФ
 Глоба Л.С., д.т.н., проф., Украина
 Голенков В.В., д.т.н., проф., РБ
 Головкин В.А., д.т.н., проф., РБ
 Гордей А.Н., д.фил.н., проф., РБ
 Грибова В.В., д.т.н., РФ
 Гулякина Н.А., к.ф.-м.н., доц., РБ
 Еремеев А.П., д.т.н., проф., РФ
 Ефименко И.В., к. фил. н., РФ
 Заболевая-Зотова А.В., д.т.н., РФ
 Загоруйко Ю.А., к.т.н., доц., РФ

Клещев А.С., д.т.н., проф., РФ
 Кобринский Б.А., д.мед.н., РФ
 Козлов О.А., д.п.н., проф., РФ
 Комарцова Л.Г., д.т.н., РФ
 Кузнецов О.П., д.т.н., проф., РФ
 Курейчик В.М., д.т.н., проф., РФ
 Лобанов Б.М., д.т.н., проф., РБ
 Найденова К.А., к.т.н., РФ
 Невзорова О.А., к.т.н., доцент, РФ
 Осипов Г.С., д.ф.-м.н., проф., РФ
 Палюх Б.В., д.т.н., проф., РФ
 Петровский А.А., д.т.н., проф., РБ
 Плесневич Г.С., к.ф.-м.н., РФ
 Родченко В.Г., к.т.н., доц., РБ
 Рычкова Л.В., к.фил.н., доц., РБ
 Смирнов С.В., д.т.н., проф., РФ
 Соловьев С.Ю., д.ф.-м.н., проф., РФ
 Соснин П.И., д.т.н., проф., РФ
 Сулейманов Д.Ш., академик АН Татарстана, РФ
 Тарасов В.Б., к.т.н., доц., РФ
 Тельнов Ю.Ф., д.э.н., проф., РФ
 Тузиков А.В., д.ф.-м.н., проф., РБ
 Харламов А.А., д.т.н., РФ
 Хорошевский В.Ф., д.т.н., проф., РФ
 Чернявский А.Ф., академик НАН Беларуси
 Шарипбаев А.А., д.т.н., проф., Казахстан
 Щербак С.С., к.т.н., доц., Украина

СТАТЬИ НА КОНФЕРЕНЦИЮ БУДУТ ПРИНИМАТЬСЯ ПО СЛЕДУЮЩИМ НАПРАВЛЕНИЯМ:

1. Принципы построения и структура технологии компонентного проектирования интеллектуальных систем;
2. Базовые семантические модели интеллектуальных систем и их реализация;
3. Семантические модели, средства и методы компонентного проектирования баз знаний и пакетов программ, ориентированных на обработку знаний;
4. Семантические модели, средства и методы компонентного проектирования решателей задач;
5. Семантические модели, средства и методы компонентного проектирования пользовательских интерфейсов интеллектуальных систем;

- 6. Комплексные методы, онтология и менеджмент компонентного проектирования интеллектуальных систем;
- 7. Логико-семантические модели прикладных интеллектуальных систем.

**ЦЕЛЬ И ФОРМАТ
ПРОВЕДЕНИЯ КОНФЕРЕНЦИИ**

Целью конференции является обсуждение проблем создания открытой комплексной технологии компонентного проектирования интеллектуальных систем. Этим определяется и формат ее проведения, предполагающий (1) ориентацию на круглые столы, посвященные обсуждению различных вопросов создания указанной технологии, (2) обеспечение возможности всем авторам не только выступить с докладами, но и продемонстрировать свои результаты на выставочных стендах.

Важнейшей задачей конференции является привлечение к ее работе не только ученых и аспирантов, но и студенческой молодежи, интересующейся проблемами искусственного интеллекта, а также коммерческих организаций, готовых сотрудничать с научными коллективами, работающими над созданием современных технологий проектирования интеллектуальных систем.

ПОРЯДОК ПРЕДСТАВЛЕНИЯ СТАТЕЙ

Статьи (только по перечисленным выше направлениям) представляются в готовом для публикации виде. Текст статьи должен быть логически законченным и содержать новые научные и практические результаты.

Крайний срок получения статей Оргкомитетом – 1 декабря 2013.

Минимальный объем статьи, включая иллюстрации, библиографический список и заключительную аннотацию (англоязычную аннотацию, в случае русскоязычной или белорусской статьи, или русскоязычную аннотацию в случае англоязычной статьи) – 4 полностью заполненные страницы. Статьи, занимающие объем более 6 страниц, подлежат согласованию с Программным комитетом.

Переписка с авторами будет вестись только по электронной почте. Адрес электронной почты Оргкомитета: ostisconf@gmail.com. Шаблон оформления статей размещен на сайте конференции.

**ФОРМИРОВАНИЕ
ПРОГРАММЫ КОНФЕРЕНЦИИ**

Программа конференции формируется Программным комитетом по результатам рецензирования представленных статей.

**ПУБЛИКАЦИЯ
МАТЕРИАЛОВ КОНФЕРЕНЦИИ**

Оргкомитет конференции предполагает публикацию отобранных Программным комитетом статей в Сборнике материалов конференции и на официальном сайте конференции <http://conf.ostis.net>. Неимущественные права принадлежат авторам статей, поэтому публикация и распространение материалов статей на иных информационных ресурсах допускается только с согласия авторов статей.

УСЛОВИЯ УЧАСТИЯ В КОНФЕРЕНЦИИ

Участие в конференции не предполагает организационного взноса.

СВЯЗЬ

С ОРГАНИЗАТОРАМИ КОНФЕРЕНЦИИ

Сайт: <http://conf.ostis.net>

Email: ostisconf@gmail.com

www.polden.by
ПОЛДЕН
ПАЯЛЬНО-РЕМОНТНОЕ ОБОРУДОВАНИЕ

ПЛУС

(017)222-50-23
(017)222-51-02
(029)648-71-31

БЕЛОРУССКИЙ ПРОИЗВОДИТЕЛЬ ЭЛЕКТРОНИКИ

ЭЛЕКТРОКОНТИНЕНТ

Республика Беларусь, 220026, г. Минск, пер. Бехтерева, 8, офис 35
тел. (+375 17) 205 06 94, 296 31 61, Velcom (+375 29) 115 35 75
e-mail: info@elcontinent.com
www.elcontinent.by

- КОНТРАКТНОЕ ПРОИЗВОДСТВО ЭЛЕКТРОНИКИ
- АВТОМАТИЧЕСКИЙ И РУЧНОЙ МОНТАЖ ПЕЧАТНЫХ ПЛАТ
 - ПОСТАВКА ПЕЧАТНЫХ ПЛАТ
 - ПОСТАВКА ЭЛЕКТРОННЫХ КОМПОНЕНТОВ
- ПОСТАВКА ТРАФАРЕТОВ ИЗ НЕРЖАВЕЮЩЕЙ СТАЛИ
 - РАЗРАБОТКА ЭЛЕКТРОНИКИ НА ЗАКАЗ
 - СВЕТОДИОДНАЯ ПРОДУКЦИЯ :
СВЕТОДИОДНЫЕ ЭКРАНЫ, СВЕТОДИОДНЫЕ ТАБЛО,
СВЕТОДИОДНЫЕ ВЫВЕСКИ,
ТАБЛО "БЕГУЩАЯ СТРОКА", ИНФОРМАЦИОННЫЕ ТАБЛО,
СВЕТОДИОДНАЯ ПРОДУКЦИЯ ДЛЯ ОСВЕЩЕНИЯ

- Современное оборудование
- Передовые технологии
- Большой положительный опыт – позволяет обеспечить высокое качество нашей продукции
- Оптимальные сроки
- Приемлемые цены

НАИМЕНОВАНИЕ ТОВАРА	ЦЕНА	НАЗВАНИЕ ОРГАНИЗАЦИИ	АДРЕС, ТЕЛЕФОН
ЭЛЕКТРОТЕХНИЧЕСКАЯ ПРОДУКЦИЯ			
Индукционные лампы Saturn 40, 80, 120, 150, 290, 300W. В комплекте электронный балласт	80-380 у.е.	ООО «ФЭК»	г. Минск. Тел./ф.: 200-34-23, тел.: 200-04-96. E-mail: lighting@fek.by
Индукционные лампы Smart Dragon 40, 80, 120, 150, 200, 300W.	80-380 у.е.		
Дроссели, ЭПРА, ИЗУ, пусковые конденсаторы, патроны и ламподержатели для люминесцентных ламп	Договор	ООО «АльфаЛидер групп»	г. Минск. Тел./ф.: 391-02-22, тел.: 391-03-33. www.alider.by
Мощные светодиоды (EMITTER, STAR), сборки и модули мощных светодиодов, линзы ARLIGHT	Договор	ООО «СветЛед решения»	г. Минск. Тел./ф.: 214-73-27, 214-73-55. E-mail: info@belaist.by www.belaist.by
Управление светом: RGB-контроллеры, усилители, диммеры и декодеры			
Источники тока AC/DC для мощных светодиодов (350/700/100-1400 mA) мощностью от 1W до 100W ARLIGHT			
Источники тока DC/DC для мощных светодиодов (вход 12-24V) ARLIGHT			
Источники напряжения AC/DC (5-12-24-48V/ от 5 до 300W) в металлическом кожухе, пластиковом, герметичном корпусе ARLIGHT, HAITAIK			
Светодиодные ленты, линейки открытые и герметичные, ленты бокового свечения, светодиоды выводные ARLIGHT			
Светодиодные лампы E27, E14, GU 5.3, GU 10 и др.			
Светодиодные светильники, прожектора, алюминиевый профиль для светодиодных изделий			
КВАРЦЕВЫЕ РЕЗОНАТОРЫ, ГЕНЕРАТОРЫ, ФИЛЬТРЫ, ПЬЕЗОКЕРАМИЧЕСКИЕ ИЗДЕЛИЯ			
Любые кварцевые резонаторы, генераторы, фильтры (отечественные и импортные)	от 0,10 у.е.	УП «Алнар»	г. Минск. Тел./ф.: 209-69-97, тел. (029) 644-44-09. E-mail: alnar@alnar.net www.alnar.net
Кварцевые резонаторы Jauch под установку в отверстия и SMD-монтаж	от 0,10 у.е.		
Кварцевые генераторы Jauch под установку в отверстия и SMD-монтаж	от 0,50 у.е.		
Термокомпенсированные кварцевые генераторы	от 2,20 у.е.		
Резонаторы и фильтры на ПАВ			
Пьезокерамические резонаторы, фильтры, звонки, сирены	от 0,04 у.е.		
СПЕЦПРЕДЛОЖЕНИЕ			
Большой выбор электронных компонентов со склада и под заказ	Договор	ЧТУП «Чип электроникс»	г. Минск. Тел./ф.: 269-92-36. E-mail: chipelectronics@mail.ru www.chipelectronics.by
Широчайший выбор электронных компонентов (микросхемы, диоды, тиристоры, конденсаторы, резисторы, разъемы в ассортименте и др.)	Договор	ООО «АльфаЛидер групп»	г. Минск. Тел./ф.: 391-02-22, тел.: 391-03-33. www.alider.by
Мультиметры, осциллографы, вольтметры, клещи, частотомеры, генераторы отечественные и АКИП, APPA, GW, LeCroy, Tektronix, Agilent	1-й поставщик	ООО «Приборостроительная компания»	г. Минск. Тел./ф.: 284-11-18, тел.: 284-11-16. E-mail: 4805@tut.by

А л в ф а Ч И П ЛТД

Новые возможности ваших идей

- Средства автоматизации
- ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ
- Датчики
- Сенсоры
- Устройства индикации
- Светодиоды

Прямые поставки от мировых производителей

Разработка и Техническая поддержка новых проектов



E-MAIL: ANALOG@ALFA-CHIP.COM
WWW.ALFACOMPONENT.COM
WWW.ALFA-CHIP.COM

БЕЛАРУСЬ, МИНСК, УЛ. Я.КОЛАСА, 3, ОФИС 8
ТЕЛ.: +375(17) 209-80-45
ФАКС: +375(17) 209-80-47



ЭЛЕКТРОКОНТИНЕНТ
КОНТРАКТНОЕ ПРОИЗВОДСТВО
ЭЛЕКТРОНИКИ

**БЕЛОРУССКИЙ
ПРОИЗВОДИТЕЛЬ
ЭЛЕКТРОНИКИ**

Контрактное производство электроники

- Автоматический и ручной монтаж печатных плат
- Поставка печатных плат
- Поставка электронных компонентов
- Поставка трафаретов из нержавеющей стали
- Разработка электроники на заказ
- Светодиодная продукция:
светодиодные экраны
светодиодные табло
светодиодные вывески
табло «бегущие строки»
информационные табло
светодиодная продукция для освещения

СОВРЕМЕННОЕ ОБОРУДОВАНИЕ
ПЕРЕДОВЫЕ ТЕХНОЛОГИИ
БОЛЬШОЙ ПОЛОЖИТЕЛЬНЫЙ ОПЫТ
ОПТИМАЛЬНЫЕ СРОКИ
ПРИЕМЛЕМЫЕ ЦЕНЫ

Республика Беларусь, 220026
г. Минск, пер. Бехтерева, 8, офис 35
тел.: (+375 17) 205 06 94, 296 31 61
VELCOM (+375 29) 115 35 75
e-mail: info@elcontinent.com
WWW. ELCONTINENT.BY



ВСЁ НЕОБХОДИМОЕ ДЛЯ АВТОМАТИЗАЦИИ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ И ВСТРАИВАЕМЫХ СИСТЕМ

Группа компаний **ЭЛТИКОН**

- Промышленные компьютеры, серверы, центры обработки и хранения данных;
- Встраиваемые и бортовые вычислительные системы, в т.ч. для жестких условий эксплуатации;
- ПЛК и микроконтроллеры, распределенные системы управления и сбора данных;
- Средства операторского интерфейса: мониторы, панели оператора, консоли управления, клавиатуры, трекболы, указательные устройства, информационные табло и мониторы для уличных применений;
- Устройства локального и удаленного ввода-вывода сигналов, АЦП, ЦАП, решения для управления движением, нормализаторы сигналов;
- Сетевое и коммуникационное оборудование для различных сетей, шлюзы данных, коммутаторы Ethernet, медиа-конвертеры, сетевые контроллеры, модемы, удлинители сетей, преобразователи интерфейсов, протоколов и т.п.;
- Датчики для различных применений;
- Источники вторичного электропитания для промышленных, медицинских, бортовых и специальных применений, инверторы электропитания, программируемые источники питания;
- Решения на основе полупроводниковых источников света для уличного освещения и архитектурной подсветки;
- Специализированные датчики, контроллеры и устройства для «умного дома»
- Корпуса, конструкции, субблоки в стандарте евромеханика, шкафы, стойки, компьютерные корпуса;
- Крепежные элементы, клеммы, монтажный инструмент, провода и кабели, кабельные вводы, соединители;
- Программное обеспечение всех уровней АСУТП, SCADA-система Genesis, OPC-серверы и средства их разработки

ADVANTECH



PROSOFT®
АВТОРИЗОВАННЫЙ ДИЛЕР

Akiwa Technology, Inc.
SUBSIDIARY OF CHANGHONG ELECTRONICS (TAIWAN)



OCTAGON
SYSTEMS

Getac

BELDEN
SENDING ALL THE RIGHT SIGNALS

TDK-Lambda



iBASE

PEPPERL+FUCHS
ELCON



Transcend



innODISK®
Beyond your imagination

Schroff®

GENCOP

ifm electronic

WAGO®

PLANAR



Fastwel

DATAFORTH®



VIPA®
art of automation

InduKey®
Industrial Input Devices

TRI-M ENGINEERING

- ✓ Более 50 вендоров в программе поставок
- ✓ Широкий диапазон продукции "из одних рук"
- ✓ Сервисный центр и послегарантийное обслуживание продукции

- ✓ Компетентный анализ технических решений с гарантией совместимости и работоспособности конфигурации
- ✓ Наличие сертификатов и ГТД

- ✓ Развитая система логистики, нестандартные схемы поставок, склады в Минске, Москве и Гамбурге
- ✓ Производство промышленных компьютеров, шкафов автоматизации, сборка телекоммуникационных шкафов

220125 Минск, пр-т Независимости, 183 • Тел. (017) 289-6333 • Факс (017) 289-6169 • E-mail: info@elticon.ru • Web: www.elticon.ru

УНП 191092652