

Раздел I. Групповое управление роботами

УДК 004.771

В.П. Андреев, К.Б. Кирсанов

ТЕХНОЛОГИЯ МНОГООПЕРАТОРНОГО УПРАВЛЕНИЯ МОБИЛЬНЫМИ РОБОТАМИ ЧЕРЕЗ ИНТЕРНЕТ*

Приведены результаты исследований, направленных на создание программно-аппаратного комплекса, позволяющего осуществлять полнофункциональное географически распределённое супервизорное управление группой робототехнических систем (включая мобильные роботы) через каналы Интернет. Термин «многооператорное» означает, что робототехнические устройства, входящие в группировку, могут управляться одновременно несколькими операторами, находящимися на географически удалённых территориях. Под полнофункциональным управлением понимается возможность с пульта оператора через Интернет управлять всеми разрешёнными функциями роботизированных систем. Для обеспечения данного функционала разработанный программно-аппаратный комплекс удовлетворяет следующим требованиям: а) децентрализованное управление сетевой структурой комплекса и её автоматическая online реконфигурируемость; б) возможность включения в структуру управления робототехнических систем различных моделей и производителей без переработки их низкоуровневого программного кода; в) возможность дистанционного (через Интернет) динамического перепрограммирования всех вычислительных устройств, включая бортовые вычислительные устройства мобильных роботов и микропроцессоры; г) защита от несанкционированного доступа; д) возможность оператору дистанционно (через Интернет) в реальном масштабе времени (с минимальными задержками) зрительно наблюдать за выполнением команд; е) устойчивость к нарушениям связи. Использована иерархическая структура организации управления группой робототехнических систем и мехатронных устройств, основанная на сетевых решениях, через объединение их информационно-измерительных и управляющих систем в локальную вычислительную сеть с мобильными узлами. Описана техническая реализация пилотного проекта программно-аппаратного комплекса на примере создания действующей Интернет-лаборатории, посредством которой реализовано полнофункциональное управление учебными роботами АМУР (МЛ «Сенсорика») и Robotino (Festo), расположенными в роботариях МГТУ «СТАНКИН» и Центра технологической поддержки образования РГГУ.

Территориально-распределённое управление; супервизорное управление; групповое управление; информационно-измерительная и управляющая система; сетевые технологии; локальная вычислительная сеть; робототехнический комплекс; мобильный робот; мехатронное устройство.

V.P. Andreev, K.B. Kirsanov

TECHNOLOGY OF MULTIOPERATOR MOBILE ROBOT CONTROL VIA THE INTERNET

We present the results of studies on the creation of hardware and software system ensuring a full-featured geographically distributed supervisory control over a group of robotic systems (including mobile robots) via the Internet. The term "multioperator" means that the robotic devices involved in the group can be controlled simultaneously by several operators that are located at

* Работа выполнена при частичной поддержке РФФИ: гранты № 13-07-01032, 13-07-00988.

spatially remote places. The term "full-featured control" means that the operator panel can be used to control all permitted operations of robotic systems via the Internet. This functionality is provided the following features of the hardware and software system developed by us: (a) the network structure of the system is controlled in a decentralized way and can be automatically reconfigured online; (b) the control structure can involve robotic systems of different models and manufacturers without rewriting their low-level program codes; (c) all computing devices (including onboard computers of mobile robots and microprocessors) can be remotely (via the Internet) and dynamically reprogrammed; (d) protection from unauthorized access; (e) the operator can visually monitor the execution of commands remotely (via the Internet) and in real time (with a minimum delay); and (f) resistance to communication breaks. We used a network-based hierarchical structure of control of the group of robotic systems and mechatronic devices by combining their information measuring and control systems into a local area network with mobile nodes. We describe the technical implementation of the pilot project of the hardware and software system using the case of an operating Internet-laboratory that provides full-featured control of AMUR (IL "Sensorika") and Robotino (Festo) educational robots located in the "robotariums" of the Moscow State Technological University "STANKIN" and the Center for Technological Support of Education of the International Institute of New Educational Technologies (CTSE IINET) at the Russian State University for the Humanities (RSUH). A similar communication channel also exists between the Keldysh Institute of Applied Mathematics, Russian Academy of Sciences (Moscow), CTSE IINET RSUH, and the Far East Federal University (Vladivostok).

Spatially-distributed control; supervisory control; group control; Information measuring and control system; network technologies; local area network; robotic system; mobile robot; mechatronic device.

Введение. Для обозначения программного обеспечения (ПО), предназначенного для взаимодействия прикладного ПО с сетью и обеспечивающего унификацию взаимодействия между программами в условиях неоднородности вычислительных платформ, используется термин «Связующее программное обеспечение» (Middleware) [1]. Такого рода ПО выступает в качестве подсистемы в более крупном программном обеспечении. Если в современных технологиях Баз Данных (БД) и Интернет связующее ПО подробно изучается [2], то применительно к робототехнике эти исследования не достигли аналогичного уровня. По всей видимости, это связано с тем, что до недавнего времени ПО для робототехнических систем было достаточно простым (с точки зрения программной инженерии) и не выходило за рамки организации простейших взаимодействий между объектами управления.

Реализация территориально-распределённого управления группой робототехнических устройств возможна с помощью таких широко известных программных каркасов (framework), как the Robotics Operating System (ROS) [3] and Microsoft Robotics Developer Studio (MRDS) [4], а также их предшественников – Palyer Project [5], LAAS GenoM и URBI [6]. Эти системы можно условно отнести к «большим», т.е. включающим в себя множество компонентов для решения разнообразных задач: SLAM, калибровка сенсоров, обработка изображений и т.п. Однако существуют и «малые» робототехнические фреймворки, например, Concurrency and Coordination Runtime [7] (включён в MRDS), осуществляющие лишь передачу сообщений между компонентами.

Перечисленные комплекты ПО эффективно решают поставленную задачу лишь в рамках одной территории – лаборатории, цеха, завода; но они не решают задачу географически-распределённого управления. Вследствие этого в них отсутствует механизм организации полнофункционального управления через Интернет.

Одной из особенностей такого способа управления является организация визуальной обратной связи по каналам Интернет (с учётом задержек в сети и с негарантированным временем доставки пакетов) для наблюдения за исполнением мехатронным устройством команд/программ управления, что особенно критично в системах дистанционного и супервизорного управления.

В случае мобильной робототехники, когда коммуникационный канал включает в себя радиоканал, возникает проблема обеспечения устойчивости системы к нарушениям связи и высокой производительности при передаче как длинных сообщений, например, многопоточное видео, так и коротких (команды). Перечисленные системы рассчитаны на работу в рамках локальной вычислительной сети (ЛВС) с устойчивой связью и практически не содержат средств, предназначенных для корректного восстановления после сбоев в канале передачи данных.

Известна практическая реализация дистанционного управления мехатронным устройством через Интернет и радиоканал с организацией обратной связи [8, 9]. Проект был выполнен в 2009–2011 гг. в ЦНИИ РТК и РКК «Энергия» (Россия) совместно с Институтом робототехники и мехатроники Немецкого аэрокосмического центра (DRL-RMC). Исследовалась возможность использования Интернет для управления с Земли двухзвенным роботом ROBOTIC, установленным на внешней поверхности российского сегмента Международной космической станции (МКС). Сеансы связи с бортом МКС проводились из ЦНИИ РТК (Санкт-Петербург, Россия) через DRL-RMC с использованием Интернет для связи ЦНИИ РТК с DRL-RMC. Особенность проекта заключалась в использовании человеко-машинного интерфейса с силовой обратной связью. В данном проекте рассматривалось лишь влияние задержек в каналах связи на процесс управления.

Анализ недостатков существующего ПО показал, что для создания технологии географически-распределённого управления группой мобильных РТК необходимо разработать качественно новую программную архитектуру. Ниже рассматривается задача управления через Интернет объединёнными в группу робототехническими комплексами (РТК), состоящими из мобильных роботов (МР) разных моделей и производителей, причём операторы МР могут находиться на географически удалённых территориях.

1. Географически-распределённое управление. Цифровое представление данных в информационно-измерительной и управляющей системе (ИИУС) мобильного РТК позволяет выполнить объединение его электронных и мехатронных модулей в локальную вычислительную сеть (ЛВС) с мобильными узлами [10]. Такой способ объединения даёт возможность распространить свойства сетевой организации на структуру ИИУС как отдельного робота, так и нескольких РТК, используя для этого сетевые технологии. Наиболее существенными в нашем случае следует считать такие свойства, как реконфигурируемость и масштабируемость. Эти свойства означают возможность легко наращивать или изменять любые ресурсы РТК: датчики, вычислители, микропроцессоры, формирователи радиоканала, а также иные IP-системы, в том числе объединять ИИУС нескольких РТК в единую географически-распределённую систему [11].

Использование сетевых технологий также позволяет организовать географически-распределённое «многооператорное» управление группировкой мобильных РТК. Термин «многооператорное», в данном случае, означает, что каждый из робототехнических устройств, входящих в группировку, может управляться несколькими операторами, находящимися на разных территориях. Такое управление допускает, при определённых условиях, режим «один-ко-многим», т.е. один оператор, если ему предоставлено соответствующее право, может вмешиваться в работу электронных и мехатронных устройств любых РТК, входящих в группировку. Пример такой структуры приведён на рис. 1.

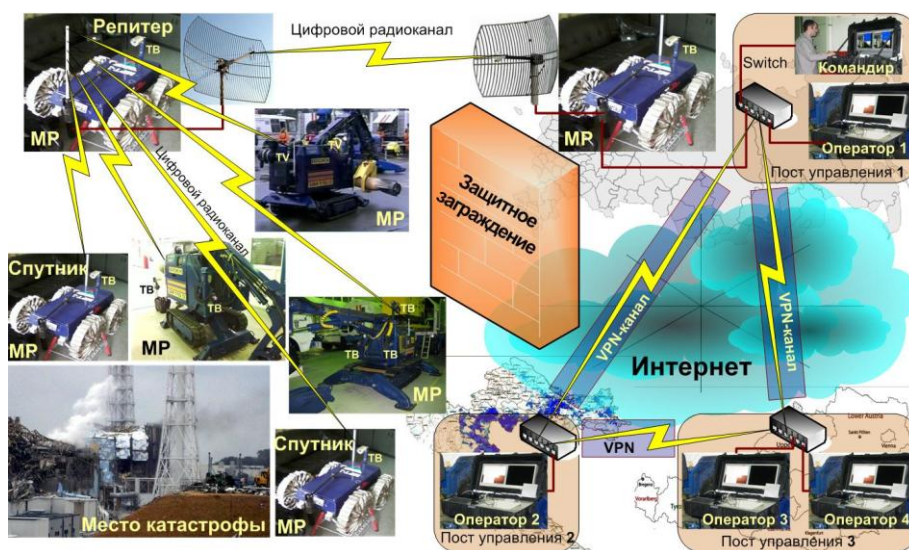


Рис. 1. Географически-распределённое «многооператорное» управление группировкой мобильных роботов

Коммуникационная система, предназначенная для организации управления через Интернет и радиоканал группой мобильных мехатронных устройств, должна обладать следующими свойствами:

- ◆ обеспечивать как «непосредственную» (с бортовых ТВ-камер), так и «стороннюю» (вид со стороны из зоны действия МР) визуальную обратную связь для дистанционного наблюдения за исполнением команд/программ управления;
- ◆ быть устойчивой к нарушениям связи;
- ◆ обеспечивать высокую производительность при передаче как длинных, так и коротких сообщений.

Рассмотрим технические и программные решения, позволяющие обеспечить указанные свойства. Сначала сформулируем основные требования, которым должен удовлетворять многокомпонентный РТК [12]:

- ◆ электронные и мехатронные компоненты ИИУС РТК должны быть элементами локальной вычислительной сети с мобильными узлами;
- ◆ связь между мобильными и стационарными модулями комплекса должна осуществляться по цифровым радиоканалам (Wi-Fi, Wi-Max и т.п.);
- ◆ увеличение дальности и повышение качества радиосвязи следует достигать путём создания оптимальной, дистанционно и оперативно управляемой архитектуры распределённой ИИУС;
- ◆ необходимо использовать многокамерную систему технического зрения: несколько ТВ-камер, в том числе с функцией PTZ (Pan-Tilt-Zoom – панорамирование-наклон-масштабирование), на борту мобильного модуля РТК и телекамеры на выносных устройствах – «Спутниках» (стационарных или устанавливаемых на мобильных платформах);
- ◆ предобработка видеосигналов и сигналов от других сенсоров должна выполняться автономно на борту мобильного модуля РТК (фильтрация шумов, коррекция искажений, сжатие изображений и т.п.);
- ◆ обязателен дистанционный (с пульта управления) оперативный контроль состояния энергосистемы и качества радиоканала;

- ◆ пульт управления оператора должен обеспечивать:
 - непрерывное (со стандартной кадровой частотой) отображение видеопотоков со всех телекамер в виде растровых изображений;
 - мнемоническое отображение на экранах мониторов показаний иных сенсоров и состояния контролируемых параметров (в том числе звуковую сигнализацию достижения критических значений);
 - возможность удалённого управления функциями телекамер (например, PTZ) и параметрами узлов ЛВС (например, выбор алгоритма сжатия изображений или его параметров);
 - возможность дистанционно контролировать исполнение на бортовой ЭВМ управляющего кода и динамически вносить в него исправление без перезагрузки системы;
- ◆ каналы связи должны быть защищены от несанкционированного доступа (например, за счёт использования VPN-каналов).

В основу разработанного механизма сетевого взаимодействия, посредством которого был организован полнофункциональный доступ через Интернет к мехатронным устройствам различного типа, была положена концепция «драйверов».

2. Концепция «драйверов». Минимальным объектом управления в предлагаемой структуре является любое электронное или мехатронное устройство, будь то бортовой вычислитель, микроконтроллер, сенсор, манипулятор, или иной исполнительный механизм. Поэтому одной из ключевых задач являлась разработка унифицированной системы правил создания ПО для конкретных моделей робототехнических устройств, позволяющих включать мехатронные компоненты в ЛВС *без переработки их низкоуровневого программного кода*. Иными словами, требовалось разработать конечный набор программных инструкций управления и сетевых протоколов для создания интерфейса программирования приложений (API – application programming interface). Такой интерфейс должен обеспечивать прозрачное интегрирование мехатронных устройств и их ПО в информационное пространство ИИУС комплекса.

Для реализации API предлагается использовать концепцию «драйверов» – для каждого мехатронного устройства должен быть создан унифицированный сетевой управляющий протокол на базе существующих низкоуровневых программных интерфейсов взаимодействия. Тогда производитель мехатронных устройств может сравнительно просто интегрировать своё устройство в сеть, создав лишь дополнительный драйвер, реализующий детерминированные протоколом программные инструкции управления и протоколы сетевого взаимодействия. При появлении в сети нового мехатронного устройства разработанное ПО подсистемы управления в процессе взаимодействия с драйвером автоматически включает это устройство в общую сеть. Одновременно выполняется публикация доступных мехатронных устройств и их характеристик в общем каталоге технического описания.

Драйвер каждого мехатронного устройства представляет собой отдельный вычислительный процесс, функционирующий на бортовой ЭВМ, и совместную работу этих процессов необходимо организовать и синхронизировать. Для решения данных задач используются следующие подходы:

- ◆ каждая команда или данные от сенсоров снабжаются *меткой времени* при прохождении каждого узла ЛВС;
- ◆ каждый драйвер содержит собственный экземпляр «*Службы имён*», в которой хранится информация о соседних (с точки зрения сетевой топологии) драйверах;
- ◆ в каждом драйвере динамически формируется *граф взаимодействий*;

- ◆ используется *текстовый формат* (ASCII) обмена данными;
- ◆ способ обмена данными позволяет организовать передачу не только отдельных команд, но и целых программ.

Добавление *метки времени* в сообщения позволяет избежать нарушения последовательности событий при сбоях, точно синхронизировать работу драйверов и учесть временные задержки, а также разность во времени между несколькими географически-распределёнными узлами сети. За счёт использования алгоритма векторных часов удаётся выполнить синхронизацию, независимую от показаний часов реального времени бортовых ЭВМ.

«*Служба имён*» – это специализированное ПО, которое каждые 20 секунд сканирует сеть и ищет аналогичные службы, а обнаружив, – обменивается с ними всей имеющейся информацией по алгоритму, близкому к распределённым хеш-таблицам – Distributed Hash Table (DHT). Благодаря этому появляется качество децентрализованности. Так, например, робототехнический комплекс, настроенный в рамках конкретной ЛВС, без изменений может быть отключён от неё, и при этом не будет нарушено его нормальное функционирование.

Граф взаимодействий представляет собой структуру данных, хранящую информацию о соседних (с точки зрения сетевой топологии) драйверах и их взаимодействиях. Граф взаимодействий автоматически восстанавливается после нарушений связи даже в случае незначительного изменения топологии сети. Тем самым обеспечивается восстановление информационной топологии сети после её разрушения в результате сбоев в каналах связи. Появляется качество автоконфигурирования, что позволяет динамически изменять топологию географически-распределённой ИИУС группировки МР, но в рамках одной сети.

Текстовый формат обмена данными позволяет упростить отладку разрабатываемого ПО и организовать взаимодействие с системой без дополнительных библиотек, что крайне важно в задачах интеграции.

Для обеспечения интеграции МР в географически-распределённую группировку необходимо было реализовать не только взаимодействие между электромеханической частью робота и бортовой ЭВМ, но и интегрировать его ИИУС непосредственно в общую иерархию управления.

3. Программно-аппаратная архитектура взаимодействия. При реализации программно-аппаратного комплекса было обеспечено выполнение следующих свойств:

- ◆ универсальность – возможность включения в систему робототехнических и иных мехатронных систем *различных моделей и производителей*;
- ◆ сетевая надёжность – *гарантированная доставка пакетов* между управляющими интерфейсами на всех уровнях сетевого взаимодействия;
- ◆ целостность – наличие *механизма управления предоставлением прав доступа* и разграничением полномочий между участниками;
- ◆ гибкость и масштабируемость – *децентрализация управления* сетевой структурой, её *реконфигурируемость* в режиме on-line;
- ◆ безопасность – *защита от несанкционированного доступа* к управляющим интерфейсам.

Организация обмена данными между элементами распределённой ИИУС реализована на основе хорошо зарекомендовавших себя сетевых технологий и решений с использованием синхронного и асинхронного взаимодействия на основе стека протоколов TCP/IP. Она состоит из следующих компонент:

- ◆ протоколы синхронного и асинхронного обмена сообщениями;
- ◆ единый каталог сервисов;
- ◆ программный модуль оператора.

Согласно классификации известной модели OSI, взаимодействие в нашем случае происходит на прикладном уровне, который мы условно разделяем на два подуровня: *транспортный* и *логический*.

К *транспортному* подуровню предъявляются следующие требования:

- ◆ высокая производительность при передаче как длинных сообщений (мегабайты), так и коротких (байты);
- ◆ устойчивость к обрывам связи;
- ◆ возможность организации взаимодействия один-ко-многим, что необходимо для задач группового управления.

К *логическому* подуровню предъявляются следующие требования:

- ◆ кроссплатформенность и многоязычность – возможность вести разработку в различных операционных системах (Windows, Linux x86\64\arm, Android, МЦСТ “Эльбрус”) и на различных языках (C, Java, Python, Fortran);
- ◆ человекочитаемость – возможность воспринимать получаемую информацию без использования специфических программных средств (данные передаются в текстовом виде, кроме случаев, когда это критично для производительности, например, в случае передачи видеопотока);
- ◆ возможность изолированной работы – работоспособность взаимодействующих мехатронных устройств, входящих в ЛВС того или иного уровня иерархии, не должна нарушаться при отключении их от внешней для них сети (более высокого уровня);
- ◆ децентрализованность и способность к самоорганизации – отсутствие необходимости в существовании какого-либо “центрального” сервера, регламентирующего работу всех компонент системы, кроме случаев ограничения доступа и регистрации участников.

Для формального описания взаимодействия в многоагентной сети была применена модель акторов [13]. В качестве механизма передачи сообщений используется библиотека ZeroMQ [14], которая выполняет роль высокоуровневой «обёртки» над стандартными сетевыми сокетами операционной системы. Этот механизм включает такие функции, как автоматическое восстановление связи после обрыва без потери контекста, возможность туннелирования через протоколы нижнего уровня: UDP, TCP, IPC, или через PGM (Pragmatic General Multicast). Кроме того, в отличие от более сложных протоколов (например, Concurrency and Coordination Runtime), ZeroMQ достаточно прост, что даёт возможность реализовать его непосредственно на микроконтроллере Arduino с модулем Ethernet Shield, но с некоторыми ограничениями, связанными с его малой производительностью. Так, при использовании данного микроконтроллера максимальная частота обработки команд в тестовой реализации составила 20 Гц.

На рис. 2 приведена общая схема сетевого взаимодействия: два мехатронных устройства RC1 и RC2 (RobotComponent) асинхронно проходят процедуру регистрации в Сервисе Имен NS (NameServer). Затем Система Управления CS (Control System), запросив список зарегистрированных устройств, осуществляет синхронное и асинхронное взаимодействие с мехатронными устройствами. Именем в данном случае может быть IP-адрес и его порт, произвольный текст (например, “двигатель”), или целый путь в иерархии (например, “Университет → лаборатория1 → Robotino → Двигатели → 3 → ШИМ”).

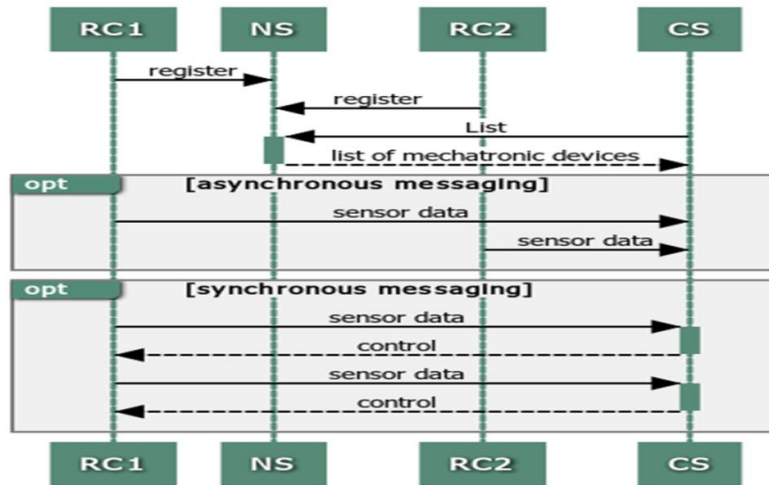


Рис. 2. Механизм сетевого взаимодействия

Схема передачи сообщений приведена на рис. 3. Команды управления и данные передаются в формате JSON, который достаточно прост и может быть реализован на маломощных процессорах, в том числе на AVR (Arduino). JSON – это текстовый (ASCII) формат представления структурированных данных, позволяющий сериализовать, т.е. представлять в виде последовательности байт, списки (вектор, массив) и ассоциативные массивы, т.е. наборы пар “ключ-значение”, что даёт возможность вручную формировать управляющие команды, не прибегая к специализированным библиотекам, как это требуется в других подобных системах. Кроме того, данное решение существенно облегчает использование нескольких языков программирования или программно-аппаратных платформ. Это достигается тем, что синтаксические анализаторы (parsers) JSON и ZeroMQ уже реализованы для абсолютного большинства платформ и языков.

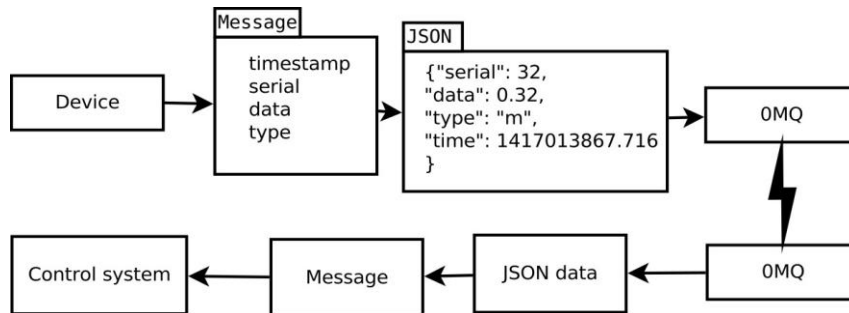


Рис. 3. Схема передачи сообщений: устройство (Device) формирует сообщение (Message), которое кодируется в формат JSON и передается в OMQ (ZeroMQ)

Появляется возможность снабжать передаваемые данные дополнительной информацией, не модифицируя принимающее их программное обеспечение, или произвольным образом менять порядок полей в рамках одного сообщения. Благодаря этому становится возможным организовать передачу не только отдельных команд, но и целых программ, которые сразу по получению исполняются интерпретатором, функционирующим на бортовой ЭВМ [15]. Таким образом организуется *динамическое перепрограммирование* мехатронного устройства, т.е., не оста-

навливая или «перезагружая» бортовые вычислительные устройства МР, можно обновлять их ПО или корректировать их работу. Это существенно упрощает одновременную поддержку различных версий одного драйвера в рамках единой сети мехатронных устройств в задачах географически-распределённого управления.

Синтаксический анализатор (парсер) JSON является стандартной библиотекой во многих языках программирования. Однако для AVR пришлось создать собственный парсер, реализующий лишь минимально необходимое подмножество JSON – без поддержки структур произвольной вложенности, но обеспечивающий удовлетворительную производительность – 40 Гц.

Избыточность информации, возникающая за счёт её текстового представления, не оказывает существенного влияния на пропускную способность, поскольку команды, как правило, компактны даже в текстовом представлении и сравнимы по размеру с заголовком IP-пакета – 20 байт.

Однако предлагаемый текстовый протокол малоэффективен для передачи многопоточного видео, поэтому для такого рода информации используются хорошо зарекомендовавшие себя протоколы, такие как RTSP (Real Time Streaming Protocol) или M-JPEG поверх HTTP.

В результате, для интеграции мехатронного устройства в систему управления, необходимо соблюсти всего лишь два основных требования к программно-аппаратной платформе подключаемого устройства:

- ◆ наличие в системе управления интерфейса, позволяющего подключить устройство к сети;
- ◆ возможность установить на устройство программу, реализующую минимально необходимое подмножество используемого в системе протокола, т.е. ZeroMQ.

В соответствии с описанным подходом авторами был создан «драйвер» для микроконтроллера Arduino. Дополнительно используется ПО iPython Notebook [16], позволяющее в интерактивном режиме изменять выполняющиеся на роботе программы.

С целью визуализации обнаруженных в сети компонент для экранного интерфейса оператора было создано управляющее программное обеспечение, представляющее все эти компоненты в виде дерева, как показано на рис. 4.

Name	Adress	Port
AMUR	VPN	
lab1	192.168....	
RSUH	192.168....	
External camera	192.168....	
AMUR-07	192.168....	
US	192.168....	120123
Engine	192.168....	120124
Camera	192.168....	8080
Battery	192.168....	120125
Robotino	192.168....	
Kuka	192.168....	
local		

Рис. 4. Одно из окон экранного интерфейса: представление в виде дерева компонент, обнаруженных у робота AMUR (US – УЗ-датчики, Engine – электродвигатели, Camera – ТВ-камера, Battery – аккумуляторная батарея)

Созданный программный продукт полностью удовлетворяет перечисленным выше требованиям. Дополнительным преимуществом является использование свободно распространяемого ПО – библиотеки ZeroMQ по лицензии GNU LGPL V3 [17, 18].

Заключение. Предложенная структура географически-распределённого управления через каналы Интернет группой мобильных роботизированных устройств была реализована в виде программно-аппаратного комплекса и показала свою эффективность в части создания действующих VPN-каналов Интернет для *многооператорного* управления учебными роботами типа «АМУР» (Автономный Учебный Мобильный Робот, МЛ «Сенсорика») и «Robotino» (немецкий концерн Festo). Такой постоянно действующий канал в настоящее время функционирует между МГТУ «СТАНКИН» (г. Москва) и Центром технологической поддержки образования Международного института новых образовательных технологий (ЦТПО МИНОТ) РГГУ (г. Москва). Аналогичный канал действует также между Институтом прикладной математики им. Келдыша РАН (г. Москва), ЦТПО МИНОТ РГГУ и ДВФУ (г. Владивосток).

Использование данной географически-распределённой сетевой структуры ИИУС обеспечивает простоту подключения к управлению мобильными роботами группировки специалистов разного профиля, находящихся далеко от места проведения работ, в том числе из фирм - изготовителей.

Данная информационно-коммуникационная инфраструктура может использоваться в системе дистанционного образования [12] для организации доступа учащихся к реальным мехатронным системам, расположенным на удалённых территориях, например, для подключения филиалов к лабораторной базе основного учебного заведения. Это позволит сосредоточить технические ресурсы в одном месте и организовать обучение групп студентов дистанционно (из филиалов), не снижая качество обучения. Кроме того, данная структура позволяет объединить территориально разнесённые научно-исследовательские коллективы вокруг какого-либо одного, обладающего соответствующей материально-технической базой, или же объединить мехатронные устройства разных коллективов в единую систему с доступом каждого-к-каждому.

Бионические сборочные системы также являются хорошим примером для использования описанной технологии [19, 20].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Emmerich W., Aoyama M., Sventek J.* The impact of research on the development of middle-ware technology // ACM Transactions on Software Engineering and Methodology. – N. Y.: ACM, 2008. – Vol. 17, No. 4. – P. 19-48.
2. *Liangzhao Z., Benatallah B., Anne HH Ngu, Dumas M., Kalagnanam J., and Chang H.* Qos-aware middleware for web services composition // Software Engineering, IEEE Transactions on 30. – 2004. – No. 5. – P. 311-327.
3. ROS. URL: <http://www.ros.org> (дата обращения: 17.09.2015).
4. Microsoft Robotics Developer Studio 4. URL: <http://www.microsoft.com/en-us/download/details.aspx?id=29081> (дата обращения: 17.09.2015).
5. *Gerkey B., Vaughan R., and Howard A.* The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems // Proceedings of the International Conference on Advanced Robotics. – 2003. – P. 317-323.
6. *Baillie Jean-Christophe.* URBI: Towards a universal robotic low-level programming language // In Intelligent Robots and Systems, 2005 (IROS 2005). 2005 IEEE/RSJ International Conference on, IEEE. – 2005. – P. 820-825.
7. *Xiaohong Q., Geoffrey C. Fox, Huapeng Yuan, Seung-Hee Bae, George Chrysanthakopoulos, and Henrik Frystyk Nielsen.* High performance multi-paradigm messaging runtime integrating grids and multicore systems // In e-Science and Grid Computing, IEEE International Conference on, IEEE. – 2007. – P. 407-414.
8. *Niemeyer, G., Slotine J.-J.E.* Telem Manipulation with time delays // Int. Journal of Robotics Research. – September 2004. – No. 23 (9). – P. 873-890.
9. *Artigas J., Jee-Hwan R., Preusche C. and Hirzinger G.* Network representation and passivity of delayed teleoperation systems // Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on 25-30 Sept. – 2011. – P. 177-183.

10. Pryanichnikov V., Andreev V. The Application of Network Technologies to Constructing Group Controlled Systems with Machine Vision for Mobile Robots // Annals of DAAAM for 2012& Proceedings of the 23th International DAAAM Symposium, 24-27th October 2012 Zadar, Croatia, ISSN 2304-1382. – 2012. – Vol. 23, No. 1. – P.1167-1174.
11. Andreev V.P., Kirsanov K.B., Pletenev P.F., Poduraev Yu.V., Pryanichnikov V.E., Prysev E.A. Technology Supervisory Control for Mechatronic Devices via the Internet // 25th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAM 2014. Procedia Engineering, 2015. – Vol. 100. – P. 33-40.
12. Andreev V., Pryanichnikov V., Poduraev Y., Kuvshinov S. Education on the basis of virtual learning robotics laboratory and group-controlled robots // 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAM 2013. Procedia Engineering. – 2014. – Vol. 69. – P. 3 5-40.
13. Carl H., Bishop P., Steiger R. A universal modular actor formalism for artificial intelligence // In Proceedings of the 3rd international joint conference on Artificial intelligence. Morgan Kaufmann Publishers Inc., 1973. – P. 235-245.
14. Hintjens Pieter. ZeroMQ // Messaging for Many Applications, O'Reilly Media, Inc., 2013.
15. Kirsanov K. The Architecture of Robotics Control Software for Heterogeneous Mobile Robots Network // 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAM 2013. Procedia Engineering. – 2014. – Vol. 69. – P. 216-221.
16. Fernando Pérez, Brian E. Granger. IPython: A System for Interactive Scientific Computing // Computing in Science and Engineering. – May/June 2007. – Vol. 9, No. 3. – P. 21-29.
17. ØMQ Licensing. URL: <http://zeromq.org/area:licensing> (дата обращения: 17.09.2015).
18. GNU lesser general public license. Version 3, 29 June 2007. URL: <http://www.gnu.org/licenses/lgpl.html> (дата обращения: 17.09.2015).
19. Каталинч Б., Пряничников В.Е., Цезарик П., Кеттлер Р. Сценарии и программы бионических сборочных систем, моделируемых с использованием роботов «АМУР» // Информационно-измерительные и управляющие системы. – 2013. – Т. 11, № 4. – С. 34-43.
20. Katalinic B., Kukushkin I., & Haskovic D. Bionic Assembly System Cloud: Functions, Information Flow and Behavior // In 9th International Conference of DAAAM Baltic, Industrial Engineering, 2014. – P. 103-108.

REFERENCES

1. Emmerich W., Aoyama M., Sventek J. The impact of research on the development of middleware technology, *ACM Transactions on Software Engineering and Methodology*. N. Y.: ACM, 2008, Vol. 17, No. 4, pp. 19-48.
2. Liangzhao Z., Benatallah B., Anne HH Ngu, Dumas M., Kalagnanam J., and Chang H. Qos-aware middleware for web services composition, *Software Engineering, IEEE Transactions on* 30, 2004, No. 5, pp. 311-327.
3. ROS. Available at: <http://www.ros.org> (дата обращения: 17.09.2015).
4. Microsoft Robotics Developer Studio 4. URL: <http://www.microsoft.com/en-us/download/details.aspx?id=29081> (accessed 17 September 2015).
5. Gerkey B., Vaughan R., and Howard A. The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems, *Proceedings of the International Conference on Advanced Robotics*, 2003, pp. 317-323.
6. Baillie, Jean-Christophe. URBI: Towards a universal robotic low-level programming language, *In Intelligent Robots and Systems, 2005 (IROS 2005). 2005 IEEE/RSJ International Conference on, IEEE*, 2005, pp. 820-825.
7. Xiaohong Q., Geoffrey C. Fox, Huapeng Yuan, Seung-Hee Bae, George Chrysanthakopoulos, and Henrik Frystyk Nielsen. High performance multi-paradigm messaging runtime integrating grids and multicore systems, *In e-Science and Grid Computing, IEEE International Conference on, IEEE*, 2007, pp. 407-414.
8. Niemeyer, G., Slotine J.-J.E. Telemanipulation with time delays, *Int. Journal of Robotics Research*, September 2004, No. 23 (9), pp. 873-890.
9. Artigas J., Jee-Hwan R., Preusche C. and Hirzinger G. Network representation and passivity of delayed teleoperation systems, *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on 25-30 Sept., 2011*, pp. 177-183.

10. Pryanichnikov V., Andreev V. The Application of Network Technologies to Constructing Group Controlled Systems with Machine Vision for Mobile Robots, *Annals of DAAAM for 2012 & Proceedings of the 23th International DAAAM Symposium, 24-27th October 2012 Zadar, Croatia, ISSN 2304-1382*, 2012, Vol. 23, No. 1, pp.1167-1174.
11. Andreev V.P., Kirsanov K.B., Pletenev P.F., Poduraev Yu.V., Pryanichnikov V.E., Prysev E.A. Technology Supervisory Control for Mechatronic Devices via the Internet, *25th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAM 2014. Procedia Engineering*, 2015, Vol. 100, pp. 33-40.
12. Andreev V., Pryanichnikov V., Poduraev Y., Kuvshinov S. Education on the basis of virtual learning robotics laboratory and group-controlled robots, *24th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAM 2013. Procedia Engineering*, 2014, Vol. 69, pp. 3 5-40.
13. Carl H., Bishop P., Steiger R. A universal modular actor formalism for artificial intelligence, *In Proceedings of the 3rd international joint conference on Artificial intelligence. Morgan Kaufmann Publishers Inc.*, 1973, pp. 235-245.
14. Hintjens Pieter. ZeroMQ, *Messaging for Many Applications*, O'Reilly Media, Inc., 2013.
15. Kirsanov K. The Architecture of Robotics Control Software for Heterogeneous Mobile Robots Network, *24th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAM 2013. Procedia Engineering*, 2014, Vol. 69, pp. 216-221.
16. Fernando Pérez, Brian E. Granger. IPython: A System for Interactive Scientific Computing, *Computing in Science and Engineering*, May/June 2007, Vol. 9, No. 3, pp. 21-29.
17. ØMQ Licensing. URL: <http://zeromq.org/area:licensing> (дата обращения: 17.09.2015).
18. GNU lesser general public license. Version 3, 29 June 2007. Available at:<http://www.gnu.org/licenses/lgpl.html> (accessed 17 September 2015).
19. Katalinich B., Pryanichnikov V.E., Tsezarik P., Kettler R. Stsenarii i programmy bionicheskikh sborochnykh sistem, modeliruemykh s ispol'zovaniem robotov «AMUR» [Scripts and programs bionic Assembly systems that are modeled with the use of robots "AMUR"], *Informatsionno-izmeritel'nye i upravlyayushchie sistemy [Information-measuring and Control Systems]*, 2013, Vol. 11, No. 4, pp. 34-43.
20. Katalinich B., Kukushkin I., & Haskovic D. Bionic Assembly System Cloud: Functions, Information Flow and Behavior, *In 9th International Conference of DAAAM Baltic, Industrial Engineering*, 2014, pp. 103-108.

Статью рекомендовал к опубликованию д.ф.-м.н., профессор А.К. Платонов.

Андреев Виктор Павлович – Московский государственный технологический университет «СТАНКИН»; e-mail: andreevvipa@yandex.ru; 127540, Москва, ул. Дубнинская, 12, корп. 1, кв. 199; тел.: +79652107951; кафедра сенсорных и управляющих систем; д.т.н.; профессор.

Кирсанов Кирилл Борисович – Институт прикладной математики им. М.В. Келдыша РАН; e-mail: kkirsanov@gmail.com; 107370, Москва, Открытое шоссе, 6, корп. 6, кв. 7; тел.: +79037454677; м.н.с.

Andreev Victor Pavlovich – Moscow State Technological University "STANKIN"; e-mail: andreevvipa@yandex.ru; 12-1-199, Dubninskaya street, Moscow, 127540, Russia; phone: +79652107951; the department of sensory and control systems; dr. of eng. sc.; professor.

Kirsanov Kirill Borisovich – Keldysh Institute of Applied Mathematics, Russian Academy of Sciences; e-mail: kkirsanov@gmail.com; 6-6-7, Otkritoe shosse, Moscow, 107370, Russia; phone: +79037454677; junior researcher.