

курсам. Например, задача 4 должна ждать задачу 11. Данный пример показывает, как рабочие процессы могут мешать друг другу.

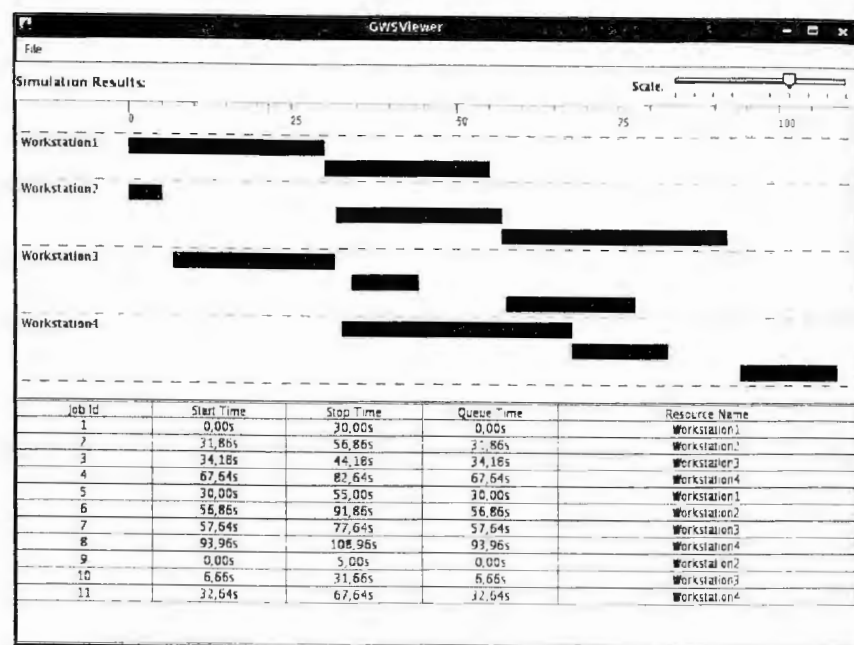


Рисунок 3 - Результаты моделирования рабочего потока

#### Список использованных источников

1. Takefusa, A. Overview of a performance evaluation system for global computing scheduling algorithms/ A. Takefusa, S.Matsuoka, K. Aida, H. Nakada, U. Nagashima - In HPDC'99: Proceedings of the The Eighth IEEE International Symposium on High Performance Distributed Computing, May, 1999.
2. The GridBus Project/ The Cloud Computing and Distributed Systems (CLOUDS) Laboratory, University of Melbourne. - <http://www.gridbus.org>, 2011 - Режим доступа: <http://www.cloudbus.org/dissertations.html> - Загл. с экрана.
3. SimGrid/ Inria gforge. - <http://gforge.inria.fr>, 2012. - Режим доступа: <http://gforge.inria.fr/projects/simgrid/> - Загл. с экрана.
4. Lamahemedi, H. Data Replication Strategies in Grid Environments // Lamahemedi, B. Szymanski, Z. Shentu, and E. Declman - Proc. Of the Fifth International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP'02), 2002.

5. Prokkola, J. OPNET - Network Simulator/ J. Prokkola - <http://www.telecomlab>, 2007. - Режим доступа: [oulu.fi/kurssit/521365A\\_tietoliikenteen\\_tekniikan\\_simuloinnit\\_jatyo\\_kalut/Opnet\\_esittely\\_07.pdf](http://www.telecomlab). - Загл. с экрана.

6. Varga, A. The omnet++ discrete event simulation system./ A. Varga. - <http://www.omnetpp.org>, 2001. - Режим доступа: <http://www.omnetpp.org/download/docs/papers/esm2001-meth48.pdf>. - Загл. с экрана.

7. Plattner, R. A Comprehensive Grid and Network Simulation Tool for Workflow based Applications Master Thesis in Computer Science/ R. Plattner, distributed & parallel systems group, institute of computer science, university of Innsbruck. - <http://www.nsnam.org>, 2012. - Режим доступа: <http://www.nsnam.org/docs/models/ns-3-model-library.pdf>. - Загл. с экрана.

УДК 621.7.06, 621.9.06

### МОДЕЛИРОВАНИЕ И РЕАЛИЗАЦИЯ ВЗАИМОДЕЙСТВИЯ СИСТЕМЫ ЧПУ С АВТОНОМНЫМ ПЛК НА ОСНОВЕ ПРОТОКОЛА TCP/IP

Г. М. Мартинов, Н. В. Козак, С. В. Богданов

Московский государственный технологический университет «СТАНКИН», г. Москва

#### Введение

Современное промышленное предприятие представляет собой набор разнообразного автоматизированного оборудования, соединенного в разномасштабные сети. Для обеспечения согласованной и эффективной работы всего парка оборудования на предприятиях в качестве управляющей платформы используют устройства электроавтоматики. В зависимости от назначения, устройства электроавтоматики подразделяются на: командные (кнопки, путевые выключатели, датчики и др.); логические (реле, программируемые логические контроллеры и др.); исполнительные (контакты, электрические магниты и муфты, исполнительные двигатели); источники питания и преобразователи напряжений и защитные (предохранители, автоматические выключатели, тепловые реле). На сегодняшний момент наибольший интерес представляют именно логические элементы электроавтоматики. Это объясняется возможностями программируемых логических контроллеров (ПЛК) решать обширный и постоянно пополняющийся список задач. Одна из главных задач ПЛК заключается в передаче собранных сведений в систему управления вышестоящего уровня [1, 2].

Целью данной работы является построение коммуникационной задачи взаимодействия ЧПУ с ПЛК с использованием возможностей протокола TCP/IP<sup>1</sup>.

<sup>1</sup> Работа выполнена по договору № 14.124.13.6495-МК об условиях использования гранта Президента РФ, для государственной поддержки молодых российских ученых МК-6495.2013.9.

Формирование требований к протоколу взаимодействия интерфейса ПЛК - ЧПУ

Взаимодействие между устройствами, входящими в состав автоматизированной системы (компьютерами, контроллерами, датчиками, исполнительными устройствами), происходит в общем случае через локальную сеть. К протоколам передачи данных по этой сети предъявляются жесткие требования:

- обеспечение надежности доставки сообщений;
- широкая поддержка разнообразных программных и аппаратных составляющих;
- передача данных объемом около 1Мб-та в одном сообщении;
- возможность циклической передачи данных;
- универсальность адресации устройств [3, 4].

Наиболее полно перечисленные требования удовлетворяет стек протоколов ТСР/Р. Данный стек имеет открытую архитектуру, стандартизированные высокоуровневые протоколы и получил широкое применение в области сетевых технологий.

### Организация взаимодействия

При разработке управляющей программы ПЛК для взаимодействия с ядром системы числового программного управления (ЧПУ) вышестоящего уровня по протоколу ТСР/Р была построена общая структурная схема модулей POU (Program Organization Unit), используемых на стороне ПЛК (рисунок 1).

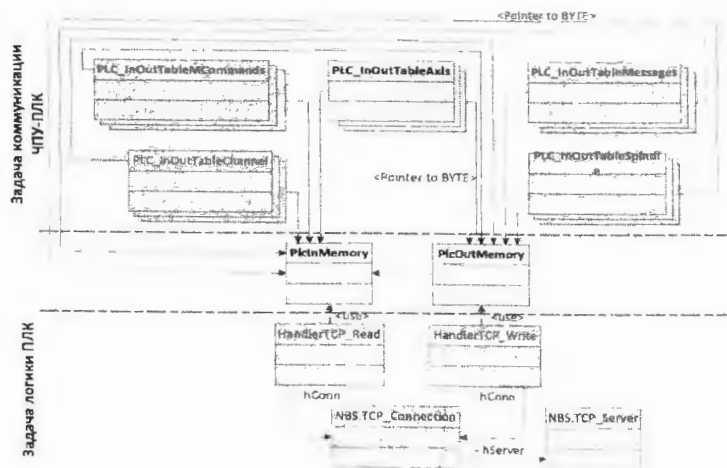


Рисунок 2 - Структура модулей POU управляющей программы ПЛК

Модули POU уровня коммуникации обеспечивают установку соединения и обмен данными для интерфейса взаимодействия ПЛК - ЧПУ. Такое взаимодействие организовывается по принципу мастер-слейв (клиент - сервер), где в качестве мастера выступает компьютер верхнего уровня. ПЛК в свою очередь постоянно находится в режиме ожидания запроса от систем ЧПУ. Только при получении адресованного непосредственно контроллеру запроса, осуществляется однократная пересылка ответа.

В качестве примера, на рисунке 2 изображена UML диаграмма, показывающая последовательность установки соединения.

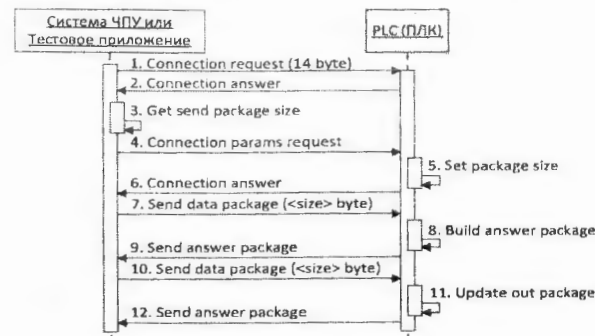


Рисунок 3 - Последовательность установки соединения между ПЛК и ЧПУ

Объекты PclnMemory и PlcOutMemory представляют собой массив данных и обеспечивают связь с классами уровня коммуникации. Эти объекты используются для хранения информации из последнего входного/выходного пакета данных от ядра ЧПУ. Эта информация содержит в себе данные об общем количестве и состоянии подключенных устройств и оборудовании с которым работает система управления.

Модули POU PLC\_InOutTableAxis, PLC\_InOutTableChannel, PLC\_InOutTableMessages, PLC\_InOutTableSpindel и PLC\_InOutTableMCommands можно отнести к вышестоящему уровню работы с данными. В процессе работы создается различное количество объектов таблиц классов в зависимости от количества подключенного оборудования, которое является входной информацией для управляющей программы ПЛК. В описанных модулях POU происходит составление актуальной информации о текущем состоянии всего оборудования, подключенного к ПЛК. Детализация взаимосвязей модулей POU, представлена на рисунке 3

В управляющей программе ПЛК осуществляется периодический вызов классов PLC\_AxeHandler, PLC\_ChannelHandler, PLC\_CommandHandler, PLC\_MessageHandler, PLC\_SpindleHandler для обновления информации о текущем состоянии соответствующих устройств. Это информация может поступать как от разнообразных датчиков, установленных непосредственно на производ-

стве и связанных с ПЛК, так и от пользователя во время режима отладки вышестоящих устройств ЧПУ.

В свою очередь объекты класса InternalServerAxiOMA с использование подклассов обеспечивают расчёт и предоставление информации о первом байте, хранящем информацию о соответствующем устройстве, в ответном пакете.

На рисунке 4 схематично представлена последовательность в циклическом обмене данными между ЧПУ к ПЛК.

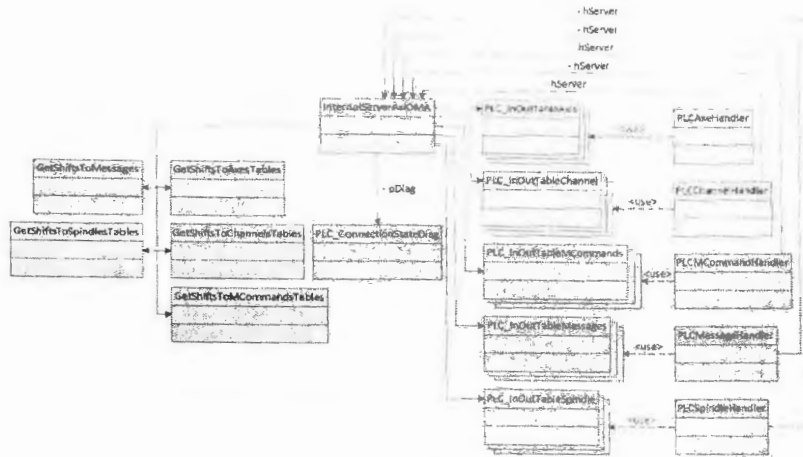


Рисунок 3 - Структура модулей POU для обработки входного пакета данных от системы ЧПУ

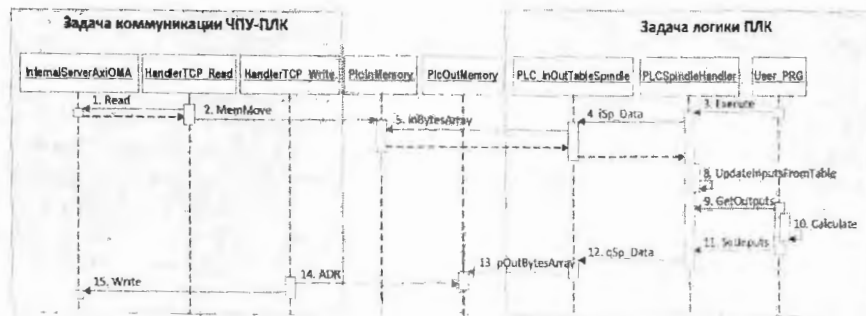


Рисунок 4 - Задачи (потоки управления) в управляющей программе ПЛК

Получение запроса от ЧПУ и формирование ответа на него в логике управляющей программы ПЛК происходит асинхронно. Это обеспечивается двумя независимыми задачами (потоками) выполняемыми в управляющей про-

грамме ПЛК. На стороне задачи логики ПЛК, программа с заданной циклическостью читает данные из буфера входных данных (PlcInMemory) и обновляет состояние выходов функциональных блоков (например, PlcSpindleHandler). В обратном направлении, данные входов функциональных блоков обрабатываются (запаковываются) и кладутся в буфер выходных данных ПЛК (PlcOutMemory). Задача коммуникации ЧПУ-ПЛК обеспечивает обновление данных во входном буфере, в соответствии с данными полученными от ЧПУ, и отправляет данные выходного буфера в ЧПУ.

Разработанная логика управляющей программы ПЛК для установки соединения с устройствами вышестоящих уровней является полностью обособленной. Представленное решение может быть выделено в качестве специализированной библиотеки (рисунок 5). Такая библиотека будет содержать в себе минимально необходимый набор модулей POU для установки соединения и обмена данными с оборудованием по протоколу TCP/IP. Использовать описанный механизм возможно при построении разнообразных управляющих программ для различных моделей ПЛК, поддерживающих соединение по протоколу TCP/IP [5].

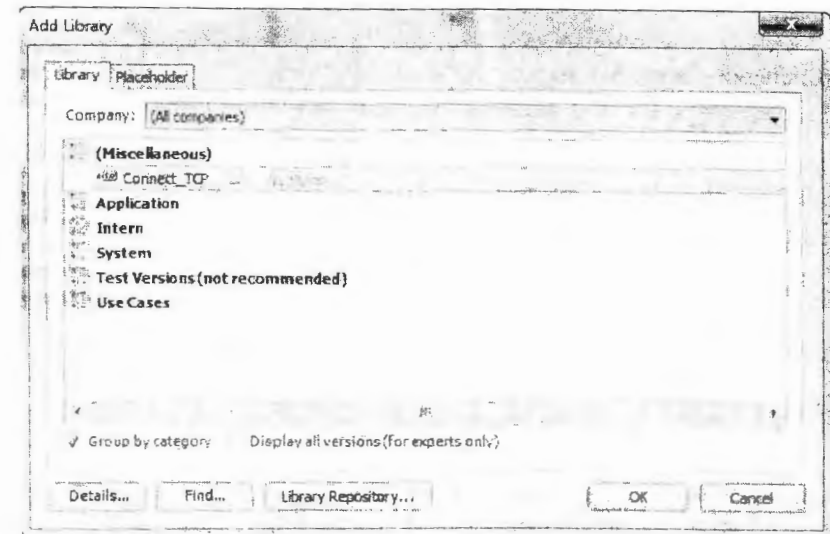


Рисунок 5 - Библиотека компонентов для циклического обмена данными ПЛК с системой ЧПУ на основе протокола TCP/IP

### Заключение

Практическими результатами проделанной работы стало решение коммуникационной задачи взаимодействия ЧПУ с ПЛК, основанное на построении специализированной управляющей программы для ПЛК, обеспечивающей цик-

личный прием и передачу данных в ЧПУ, а также создание специализированного протокола взаимодействия управляющей программы коммуникации для реализации взаимодействия с системами управления вышестоящего уровня.

Научная составляющая работы содержится в разработке структуры РОУ модулей, используемых для создания управляющих программ ПЛК, которые обеспечивают циклический обмен данными о технологическом процессе, а также в разработке алгоритмов взаимодействия, обеспечивающих установку соединения и циклический обмен данным на основе протокола TCP/IP.

#### Список использованных источников

1 Мартинов, Г. М. Кроссплатформенный программно-реализованный логический контроллер управления электроавтоматикой станков с ЧПУ / Г. М. Мартинов, Р. А. Нежметдинов // Автоматизация и современные технологии. - 2013. - № 1. - С. 15-23.

2 Григорьев, С. Н. Научно-технические проблемы построения современных технологических систем с числовым программным управлением / С. Н. Григорьев // Мехатроника, автоматизация, управление. - 2011. - № 4. - С. 19-26.

3 Козак, Н. В. Концепция построения средств диагностики и управления устройствами электроавтоматики на базе OPC технологии / Н. В. Козак, Р. А. Абдуллаев // Системы управления и информационные технологии. - 2010. - № 3. - С. 28-32.

4 Мартинова, Л. И. Реализация открытости управления электроавтоматикой станков в системах ЧПУ класса PCNC / Л. И. Мартинова, Н. В. Козак, Р. А. Нежметдинов, Р. Л. Пушков // Приборы и системы. Управление, контроль, диагностика. - 2011. - № 02. - С. 11-16.

5 Козак, Н. В. Графические системы и интерфейс оператора: учебное пособие / Н. В. Козак, Р. А. Нежметдинов. - М.: МГТУ «Станкин», 2010. - 81с.

УДК 608.2

### УСОВЕРШЕНСТВОВАНИЕ МЕТОДА БРИДЖМЕНА ДЛЯ ПОЛУЧЕНИЯ МОНОКРИСТАЛЛОВ ПРОСТЫХ И НЕБОЛЬШИХ РАЗМЕРОВ

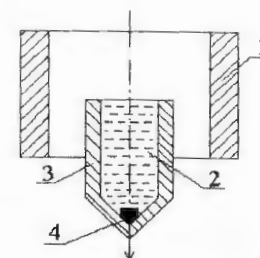
Нгуен Хоанг Линь

Оренбургский государственный университет, г. Оренбург

В настоящее время за основу всех способов получения монокристаллов применяют методы Бриджмена и Чохральского, заключающиеся в создании условий для целенаправленной кристаллизации вокруг одного центра.

По методу Бриджмена (рисунок 1) тигель 3 с коническим концом, в котором находится расплав 2, медленно вытаскивается из печи 1, обеспечивающей градиент температуры вдоль своего рабочего пространства. При перемещении вершина конического конца тигля из более горячей области в холодную остывает

раньше, что приводит к образованию единственного центра кристаллизации расплава, но в то же время его остальная часть находится в жидком состоянии. По мере продвижения изложницы к выходу из печи расплав кристаллизуется вокруг единственного образованного центра 4 и в результате получаем монокристаллический стержень. Для реализации этого метода требуется применение сложных технических устройств, обеспечивающих медленное продвижение тигля, и дорогостоящей печи с градиентом температуры, но он позволяет получить стержень монокристалла большой длины при относительно низкой производительности.



1 – печь с градиентом температуры; 2 – расплав; 3 – тигель; 4 – монокристалл.

Рисунок 1 – Метод Бриджмена для выращивания монокристалла

Анализ механизма и теплового баланса при кристаллизации стержня в тигле с коническим концом позволяет усовершенствовать процесс получения монокристалла, но с меньшими размерами.

Рассмотрим тепловой баланс стержня с коническим концом (рисунок 2). Количество теплоты, необходимое для нагрева и расплавления стержня при температуре  $T$ , равно:

$$Q = Q_1 + Q_2, \quad (1)$$

$$Q_1 = cm(T - T_0), \quad (2)$$

$$Q_2 = \lambda m, \quad (3)$$

где  $Q_1$  – количество теплоты, необходимое для разогрева с температуры  $T_0$  до  $T$ ;

$Q_2$  – количество теплоты, необходимое для расплавления.

$c$  – удельная теплоемкость материала стержня;

$m$  – масса стержня;

$\lambda$  – удельная теплота плавления.