

DP могут выступать как ПК, так и ПЛК. Ведомыми устройствами могут быть ПЛК и другое оборудование, взаимодействие с которыми осуществляется по протоколу Modbus RTU.

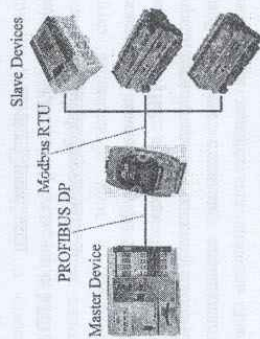


Рисунок 1. Схема интеграции устройств электроавтоматики

В работе были использованы две модели шлюзов: PROFIBUS/Modbus GW-7552 и Modbus TCP server-RTU slave/CANopen master GW-7433D.

Конфигурирование шлюзов осуществляется с помощью прикладного ПО, предоставляемого компанией-разработчиком. В качестве примера на схеме (Рисунок 2) рассмотрена последовательность этапов, выполняемых пользователем при настройке шлюза GW-7552.

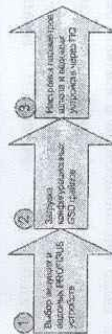


Рисунок 2. Конфигурирование шлюза GW-7552

Настройка шлюзов других производителей производится по аналогии с приведенной схемой.

#### Выводы

В процессе работы были определены основные особенности и области применения шлюзов промышленных протоколов передачи данных в различных информационных сетях. Был выявлен ряд существенных

преимуществ, которые получают разработчики при использовании возможностей таких шлюзов в структуре системы управления.

При подключении и настройке параметров шлюза процесс его конфигурирования был структурирован и представлен в виде последовательности этапов для каждой модели. Подобное обобщение может быть применимо и к другим производителям шлюзов в силу схожести их принципа функционирования и прикладного ПО.

#### Список литературы

1. Сосонкин В.Л., Мартинов Г.М. Программирование систем числового программного управления: учеб. Пособие. – М.: Логос; Университетская книга; 2008. – 344 с. + 1 компакт-диск.
2. Официальный сайт компании ICP DAS <http://www.icpdas.com/>

#### АРХИТЕКТУРА КРОССПЛАТФОРМЕННОГО ДРАЙВЕРА ПРИВОДОВ ДЛЯ СИСТЕМ ЧПУ

А.И. Бондаренко  
Россия, г. Москва, МГТУ «СТАНКИН»  
[bondarenko.nes@gmail.com](mailto:bondarenko.nes@gmail.com)

А.Б. Любимов, и.с.,

МГТУ «СТАНКИН», кафедра «Компьютерные системы управления»

Работа выполнена по Госконтракту П717 от 20 мая 2010г. и 16.740.11.0228 от 22 сентября 2010г. на проведение НИР в рамках ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013

Создание кроссплатформенного драйвера следящих приводов для систем ЧПУ является важным аспектом увеличения открытости и гибкости системы в целом. Однако, реализация инвариантности системы управления по отношению к используемой архитектуре невозможна без наличия продуманной архитектуры модуля-координатора, выполняющего роль программного драйвера цифровых приводов [1].

Место рассматриваемого модуля в структуре системы – между интерпретатором системы ЧПУ и исполнительными устройствами технологического комплекса (приводы следящие, главного движения) [2].

Решая вопрос о выборе платформ, на которых следует обеспечить работоспособность драйвера, было решено остановиться на трех вариантах: Win32 (самая распространенная в настоящее время операционная система для PC), RTX (расширение реального времени для операционных систем семейства Windows (NT/2000/XP/Vista), Linux RT (наиболее перспективный вариант ввиду высокой надежности и наличия свободной лицензии). В качестве альтернативных вариантов рассматриваются такие операционные системы реального времени, как QNX, VxWorks, Windows CE (рисунок 1).



Рисунок 1. Перспективные платформы для реализации модуля

Фактически, результатом выполнения кода драйвера может являться создание файлов типа драйвера (.sys, .tss и .ko соответственно), для дальнейшей загрузки в систему ЧПУ, при этом модуль .tss является условно-подгружаемым (исходя из специфики функционирования программного кода расширения реального времени), но предусмотрено для единообразного функционирования системы. Приложение (система управления) взаимодействует с драйверами посредством команд IoControl (рисунок 2).

С целью соблюдения принципов повторного использования программного кода, используется модуль, содержащий общий код на языке C/C++. Такой код содержит в себе универсальные алгоритмы, написанные без использования API конкретной платформы. Он может использоваться всеми тремя драйверами (реализуя, например, различные виды обработки циклических данных – функции регуляторов, вычисления и т.д.)

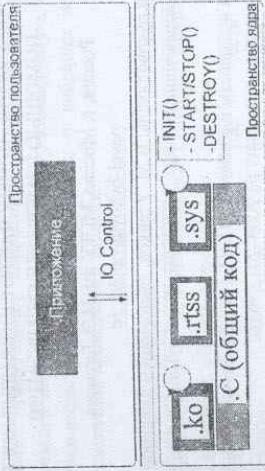


Рисунок 2. Схема взаимодействия модуля

Функции из такого модуля, находясь на уровне ядра, могут обращаться к любым объектам уровня ядра. К примеру, имея адрес структуры, описывающей конкретный привод, эта функция может считывать из некой, заранее определенной ячейки этой структуры, адрес (смещение), по которому следует класть циклические данные.

#### Список литературы

1. Григорьев С.Н., Андреев А.Г., Мартинов Г.М. Перспективы развития кроссплатформенных компьютерных систем числового программного управления высокотехнологичного оборудования // Автоматизация в промышленности, 2011. №5, с. 3-8.
2. Мартинов Г.М., Бондаренко А.И. Использование SERCOS-интерфейса для управления двигателями в компьютерных системах управления // VI Региональная конференция научно-практическая конференция студентов и аспирантов: сборник трудов. – Старый Оскол: СТИ НИТУ МИСИС, 2010. - С.63-66