

Разработка подсистемы конфигурирования

программно реализованного контроллера для систем ЧПУ*

Бондаренко Арсений Иванович, Любимов Александр Борисович

МГТУ «СТАНКИН», кафедра «Компьютерные системы управления»

bondarenko.ncs@gmail.com

Формально, с точки зрения теории управления, нет разницы, каким объектом управлять: будет ли это станочный комплекс, оборудование пивоваренного завода или же лифт - принципы одни и те же. Система управления (СУ) – понятие широкое, а с нынешними темпами развития автоматизации скоро оно перейдёт в категорию необъятных. Однако, исторически сложились различные виды современных систем управления технологическим оборудованием, рассмотрим наиболее распространённые из них.

Программируемый логический контроллер (ПЛК) – (Programmable logic controller, PLC) представляет собой цифровое специализированное вычислительное устройство, используемое для автоматизации технологических процессов. В качестве основного режима длительной работы ПЛК, зачастую в неблагоприятных условиях окружающей среды, выступает его автономное использование, без серьёзного обслуживания и практически без вмешательства человека. Применительно к системам ЧПУ реализует ряд вспомогательных операций, таких как сигнал подачи смазывающе-охлаждающей жидкости (СОЖ), снятия стружки, смены инструмента, управление вращением шпинделя. Для программирования используются как графические языки (LD, SFC, FBD), так и текстовые (IL, ST).

Программируемый контроллер автоматизации (ПКА) - (Programmable Automation Controller, PAC) это компактный контроллер, сочетающий в себе характерные черты и возможности как системы управления на базе персонального компьютера, так и программируемого логического контроллера. В отличие от последних, контроллеры автоматизации, как правило, обладают открытой модульной архитектурой, более развитыми инструментальными средствами программирования, усовершенствованной системой обработки сигналов с портов входов/выходов.

Системы числового программного управления представляют собой наиболее универсальный тип систем управления, в частности, может включать в себя оба типа вышеперечисленных систем, поэтому ее рассматривают как обобщённый представитель промышленных систем управления. В дальнейшем будем рассматривать решения в рамках систем ЧПУ. Область их применения (преимущественно машиностроительный сектор) обуславливает широкую потребность в подобных системах, а разнообразие решаемых задач зачастую диктует необходимость использования разнородного оборудования для их решения.

Наиболее современные системы ЧПУ класса PCNC-4 строятся на базе однокомпьютерной архитектуры [1]. Сегодня вычислительные мощности персонального компьютера позволяют решать все задачи управления (геометрическую, терминальную, диагностическую) исключительно программным путём, без какой-либо дополнительной аппаратной поддержки. Операционная система реального времени обеспечивает жёсткие такты исполнения управляющих сигналов, а внешний интерфейс может быть построен на базе любого стандартного промышленного протокола. Весьма перспективной является идея реализации логической задачи в рамках того же персонального компьютера за счет использования программного реализованного контроллера электроавтоматики. Подобный подход позволяет обойти использование аппаратных вычислительных модулей (ПЛК или ПАК), перенести логику на уровень ядра системы ЧПУ, а в качестве исполнительных устройств использовать простые и дешёвые модули удалённого ввода/вывода.

Таким путём достигаются следующие преимущества:

- отсутствие необходимости приобретения дорогостоящих вычислительных устройств;

*Работа выполнена по Госконтракту №П858 на проведение НИР в рамках ФЦП "Научные и научно-педагогические кадры инновационной России" на 2009-2013 гг.

- программная реализация более гибкая, такую систему проще масштабировать или модифицировать;
- независимость от среды программирования контроллеров конкретных производителей или сторонних разработчиков;
- жёсткий цикл обмена данными при использовании стандартных интерфейсов связи с системой ЧПУ (например, 250 мкс при скорости 100 Мбит/сек. по интерфейсу SERCOS III [3]).

Помимо достоинств, применение программно реализованного контроллера электроавтоматики имеет ряд недостатков:

- сложность разработки математического обеспечения, выполняющего логическую функцию в режиме реального времени;
- необходимость создания собственного редактора и интерпретатора управляющих программ;
- необходимость решения задачи конфигурирования системы ЧПУ для использования различных модулей ввода/вывода (каждый из которых может состоять из ряда различных исполнительных устройств, например, слотов аналоговых входов или дискретных выходов).

В настоящем докладе будет рассмотрен предлагаемый способ реализации подсистемы конфигурирования программно реализованного контроллера электроавтоматики на примере интеграции модуля удалённого ввода/вывода (бас каплера, от англ. bus coupler) в систему ЧПУ «АксиОМА Контрол».

Архитектура системы ЧПУ в вертикальном сечении имеет несколько уровней, в этой иерархии можно грубо выделить три основных блока: терминальная часть (человеко-машинный интерфейс), уровень ядра системы ЧПУ (координация всех модулей системы), уровень драйверов ядра операционной системы (решение задач взаимодействия с оборудованием в режиме реального времени). Для того чтобы сконфигурировать систему на работу с внешними разнородными устройствами, надо очень чётко знать, какие именно структуры данных на каком из уровней будут представлять конкретные параметры оборудования, как обеспечить передачу (а также сохранение, загрузку и возможность изменения) параметров [2].

Структура модулей удалённого ввода/вывода предполагает использование слотов расширения. Такими слотами могут быть слоты аналоговых входов, выходов, цифровых входов, выходов, слоты для взаимодействия по стандартному последовательному интерфейсу (SSI) (рисунок 1).

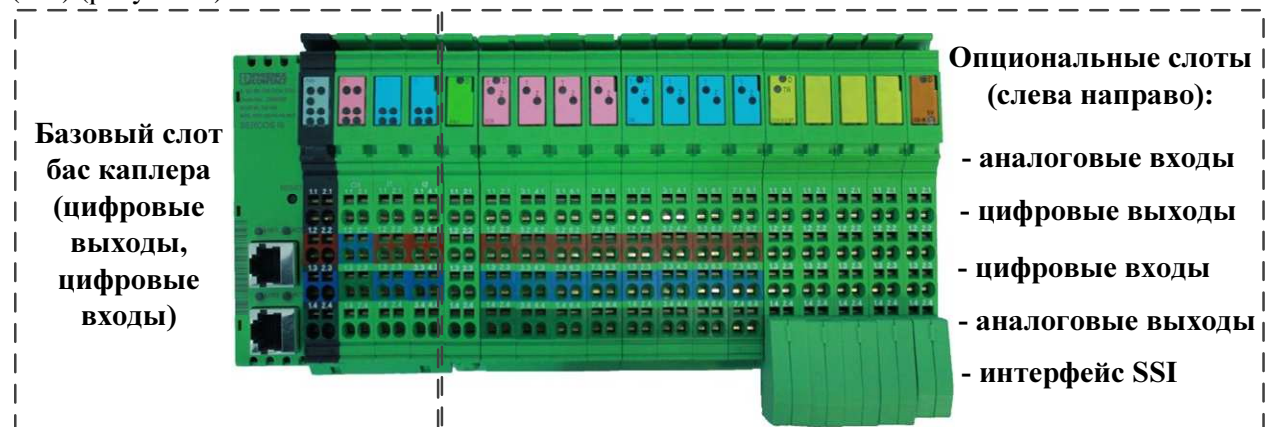


Рисунок 1 – Модуль удалённого ввода/вывода на примере бас каплера производства фирмы Phoenix Contact

Для обеспечения взаимодействия системы ЧПУ с требуемым количеством подобных модулей была разработана подсистема конфигурирования, которая решает следующие задачи:

- предоставляет возможность оператору добавить нужное количество устройств в систему и удобным образом параметризовать их;
- обеспечивает сохранение/загрузку параметров конфигурации в файл;

- реализует перенос структур параметров вниз по иерархии от терминальной части к драйверам уровня операционной системы.

Первым этапом проектирования подсистемы было определение структур данных на всех трёх уровнях, с учётом специфики каждого из них. Для терминальной части (разработка которой ведётся на языке C# платформы .Net) это набор из трёх классов, которые несут в себе максимум полезной для оператора информации (вплоть до словесного описания, составленного самим оператором или пиктограммы, для облегчения работы с многочисленными устройствами). Кроме того эти классы реализуют протокол взаимодействия программных модулей в системе ЧПУ (канал связи X-Data), что позволяет осуществлять пересылку пакетов данных по каналу TCP/IP между ядром и терминальной частью.

Уровень ядра и уровень драйверов операционной системы разделяют набор структур, определяющих минимальное количество параметров, необходимых для корректной работы всей системы в режиме жёсткого реального времени (например, в рамках процедуры обработки прерываний). В ядре системы ЧПУ эти структуры являются полями классов, реализующих взаимодействие по каналу X-Data (приём пакетов с заполненной конфигурацией из терминальной части, передача пакетов с конфигурацией, загруженной из файла); механизм сохранения/загрузки данных в файл конфигурации; передачу команд управления состоянием программного реализованного контроллера на уровень драйвера. На уровне драйвера структуры содержат информацию самого низкого уровня (размеры выделяемой памяти под взаимодействие с каждым конкретным устройством; адреса смещений относительно базового адреса мастер-устройства, куда будет происходить запись выходов и чтение входов), эти структуры имеют наиболее компактный вид, для упрощения их обхода в процедуре обработки прерываний.

На рис.2 представлен вид окна редактирования конфигурации программно реализованного ПЛК, оно состоит из трёх областей. Редактор иерархии объектов позволяет наглядно представить текущую конфигурацию подключённого оборудования, а также легко (по контекстному меню) добавить удалить устройства. При выделении любого объекта оператору показываются его основные параметры в окне информации, а расширенные возможности настройки доступны в окне свойств текущей конфигурации.

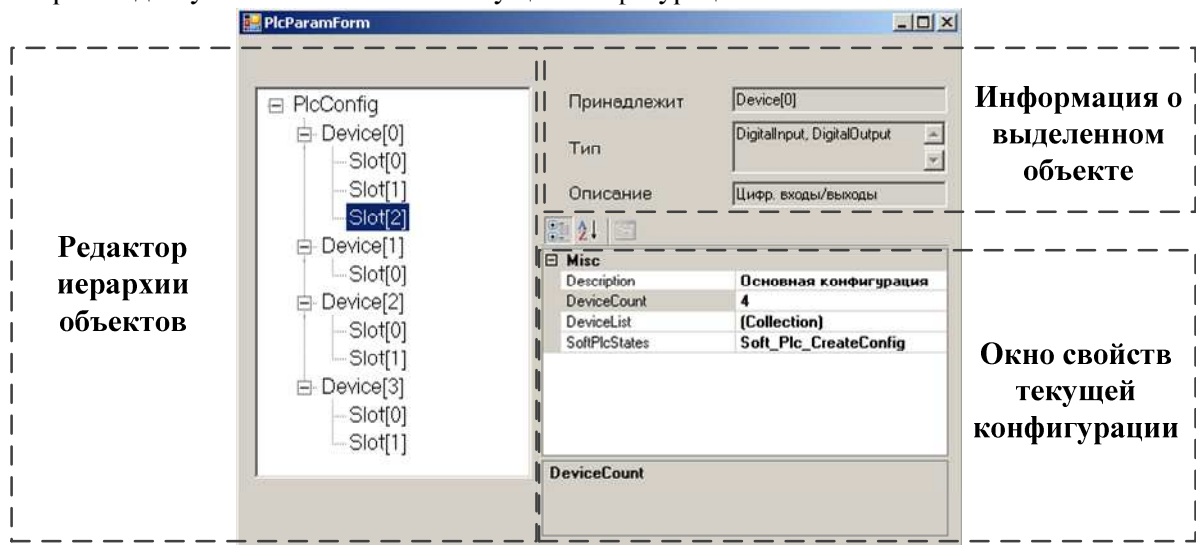


Рисунок 2 – Окно редактирования конфигурации программно реализованного контроллера

В настоящее время происходит тестирование реализованной функциональности и интеграция программного кода с другими модулями программно реализованного контроллера в рамках системы ЧПУ «АксиОМА Контрол» (редактор функциональных блоков, математическое ядро ПЛК, модуль коммуникации).

Список литературы:

1. Сосонкин В.Л., Мартинов Г.М. Системы числового программного управления: Учеб. пособие. – М.: Логос, 2005. – 296с.
2. Григорьев С.Н., Мартинов Г.М. Концепция построения базовой системы числового программного управления мехатронными объектами // Информационные технологии в проектировании и производстве, 2011. №2, с.21-27.
3. Бондаренко А.И., Любимов А.Б. Построение распределённых гетерогенных вычислительных систем управления на базе промышленных сетей // XI Международная конференция "Системы проектирования, технологической подготовки производства и управления этапами жизненного цикла промышленного продукта (CAD/CAM/PDM - 2011)" – С.76