

Построение портативных терминальных решений для контроля и управления технологическим оборудованием

*П.А. Нукишечкин,
к.т.н., доц., pnikishechkin@gmail.com,
Н.Ю. Червоннова,
преп., 4ervonnova@list.ru,
А.Н. Нукич,
асп., n_nikich@me.com,
МГТУ «СТАНКИН», г. Москва*

В статье представлены теоретические аспекты разработки механизма для создания гибких и изменяемых интерфейсов оператора для реализации управления разнородным технологическим оборудованием; практические аспекты разработки среды проектирования и программного сопровождения интерфейса пользователя, а также средств мониторинга за работой технологического оборудования и пример его использования.

The article presents the theoretical aspects of the development of a mechanism for creating flexible and changeable operator interfaces for implementing the management of heterogeneous technological equipment; practical aspects of the development of the design environment and software support of the user interface, as well as means of monitoring the work of technological equipment and an example of its use.

Современные тенденции развития промышленности предусматривают повышение уровня автоматизации и информатизации всех процессов и создание многоуровневой, глобальной технологической и организационной системы, которая подразумевает интеграцию в единое информационное пространство всех технологических операций и сопровождающих их процессов. В сферу промышленности все больше интегрируются портативные, мобильные устройства для контроля и управления технологическими процессами, а для хранения и доступа к большим объемам данных применяют технологии облачного хранения.

Управление сложным станочным оборудованием обычно осуществляется с помощью систем числового программного управления (ЧПУ), представляющих собой комплекс аппаратных компонентов, встроенных в них математических и логических алгоритмов управления, а также средств человеко-машинного взаимодействия. Для подобных систем проектирование интерфейса оператора «с нуля» под каждый объект управления невозможно в связи со сложностью программной реализации. Поэтому для решения задач человеко-машинного взаимодействия при управлении станочным оборудованием и робототехническими комплексами используются штатные терминальные решения, набор которых определен заранее производителем системы и не предоставляет возможностей к его изменению. В то же время, развитие современных технологий производства и промышленного оборудования приводит также и к увеличению объемов данных, которые необходимо контролировать при управлении технологическими процессами. Зачастую подобных штатных терминальных средств бывает недостаточно для полноценного контроля и управления сложными процессами, особенно когда оборудование имеет сложную электроавтоматику, что требует визуализации и контроля большого числа параметров. также это представляет проблему, когда станочное оборудование имеет большие габариты, что требует портативности терминального решения. В этой связи одной из важнейших задач является реализация интерфейса пользователя систем управления, который позволяет анализировать весь необходимый спектр данных, передаваемый с объектов управления и оперативно реагировать на различные ситуации. Таким образом, актуальной является задача создания механизмов для построения дополнительных портативных терминалов с удобным и гибким пользовательским интерфейсом, предназначенным для решения задач контроля и управления технологическим оборудованием. При этом графические объекты интерфейса оператора должны иметь связь с главной, базовой системой управления и получать все необходимые данные с объекта управления. В особенности это является актуальным для управления оборудованием повышенной сложности, которым являются станки с ЧПУ [1,2].

Разработанная на кафедре «Компьютерные системы управления» базовая вычислительно-управляющая платформа позволяет решать широкий спектр задач промышленной автоматизации. Основным продуктом, построенным на базе данной платформы, является система ЧПУ «АксиОМА Контрол», позволяющая управлять сложным станочным оборудованием различных видов. Реализация системы ЧПУ выполнена по модульному принципу, что позволяет исключать, или, наоборот, добавлять определенную функциональность системы для решения задач различной сложности. В состав разработанной системы ЧПУ входит также встроенный программно-реализованный контроллер, позволяющий решать задачи управления электроавтоматикой станков. Система ЧПУ «АксиОМА Контрол» имеет двухкомпьютерную архитектуру построения и состоит из ядра системы, функционирующего в режиме жесткого реального времени и осуществляющее все вычислительные и управляющие функции. Терминальная часть предоставляет функционал для человеко-машинного взаимодействия и функционирует в машинном времени. В качестве терминала обычно используются специализированные решения в виде панели оператора и станочной панели. В статье предлагается создание механизма для разработки дополнительных, в том числе мобильных терминальных решений, взаимодействующих с ядром системы ЧПУ, позволяющих визуализировать информацию о работе основных узлов технологического оборудования и параметрах работы системы ЧПУ. Структурная модель, визуализирующая принцип интеграции дополнительных терминальных решений представлена на рисунке 1.

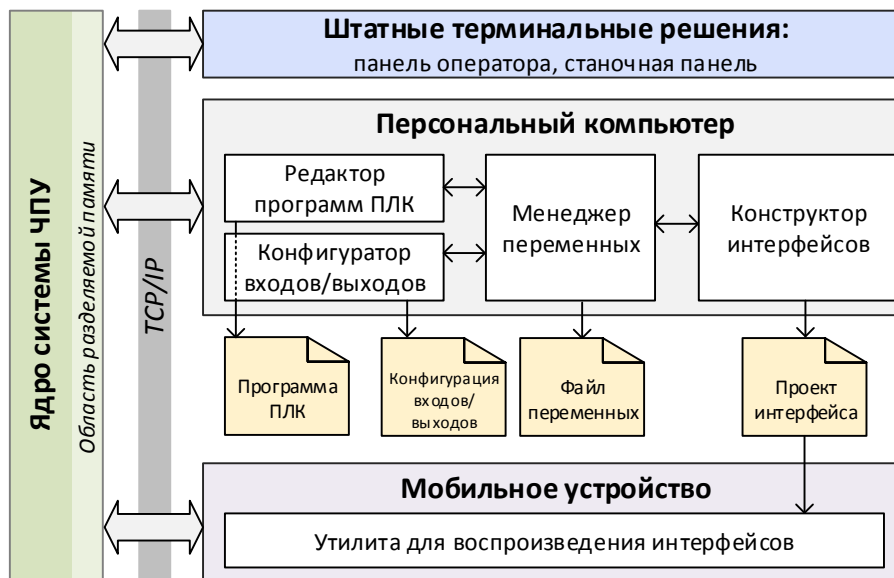


рис. 1 Структурная модель взаимодействия системы ЧПУ с дополнительными терминальными решениями

Взаимодействие между терминальной частью и ядром системы ЧПУ реализуется через коммуникационный клиент. Он обеспечивает передачу данных между выносным/удаленным терминалом и ядром системы управления, используя протокол TCP/IP: через Ethernet соединение или беспроводное соединение (Wi-Fi). Одним из способов передачи основной информации между ядром и терминальной частью системы является механизм разделяемой памяти, что представляет собой выделенную область памяти в ядре системы, в которой заранее отведены ячейки, где хранится информация о состоянии входов контроллера, выходов, а также значений пользовательских переменных, в число которых входит базовая информация о работе системы ЧПУ. Данная область памяти доступна терминальным клиентам для записи собственных выходных сигналов и чтения необходимой входной информации. Описанный способ обмена данными является наиболее быстрым между процессами работы ядра и терминальных решений, и позволяет обеспечить кроссплатформенность, т.е. независимость от операционной системы, на которой функционирует ядро или интерфейс (Windows/OS на базе Linux). Предполагается, что основной информацией, необходимой для передачи в дополнительные терминальные клиенты, является информация о работе электроавтоматики, то есть состояния всех входов/выходов, управляемых встроенным контроллером, а также базовая информация о работе системы ЧПУ: статус работы, положения осей, скорость и величина подача при обработке [1,3].

Создание алгоритмов управления электроавтоматикой станка реализуется в специализированном редакторе ПЛК, позволяющий создавать программы на языке FBD. С помощью конфигуратора устройств ввода/вывода производится настройка и параметризация аппаратной части контроллера, - устройств ввода/вывода, сервоприводов. После разработки программы и конфигурации аппаратных устройств систематизируется набор основных переменных, которые необходимо визуально контролировать или управлять их значениями, для чего используется специальный менеджер ПЛК переменных (рисунок 2).

ID	Только для чтения	Имя	Описание переменной	Тип значения	Адрес байта	Индекс бита	Значение
51	<input checked="" type="checkbox"/>	Wm_CoolSpndrl	Авария СОЖ через шпиндель	Битовый	104403	6	
52	<input checked="" type="checkbox"/>	Wm_LowCoolSpnd	Низкий уровень СОЖ через шпиндель	Битовый	104403	7	
53	<input checked="" type="checkbox"/>	WmHighCoolSpnd	Высокий уровень СОЖ через шпиндель	Битовый	104404	0	
54	<input type="checkbox"/>	PrgRun		Битовый	104400	3	
55	<input type="checkbox"/>	Spndl_run		Битовый	104400	0	
56	<input checked="" type="checkbox"/>	CoolantSpndrl	СОЖ через шпиндель	Битовый	104408	1	
57	<input type="checkbox"/>	CoolantSpndrl		Битовый	104408	1	
58	<input type="checkbox"/>	M3key		Битовый	272	0	
59	<input type="checkbox"/>	CoolFilter		Битовый	204809	0	
60	<input checked="" type="checkbox"/>	CoolantSpndrl	СОЖ через шпиндель включен (с встроенным фи...	Битовый	104408	1	
61	<input type="checkbox"/>	ON_Shneki	Включить шнеки	Битовый	104408	2	
62	<input checked="" type="checkbox"/>	MDI_mode		Битовый	104401	6	
63	<input checked="" type="checkbox"/>	AUTO_mode		Битовый	104401	2	
64	<input checked="" type="checkbox"/>	Not_JOG_mode	НЕ Jog режим в ЧПУ	Битовый	104401	5	
65	<input type="checkbox"/>	M110_finished	успешное выполнение M110	Битовый	104410	6	
66	<input checked="" type="checkbox"/>	WmShnekRight	Авария шнек направо	Битовый	104404	1	
67	<input type="checkbox"/>	WmShnekLeft	Авария шнек налево	Битовый	104404	2	
68	<input type="checkbox"/>	WmShnekRight		Битовый	104404	1	
69	<input type="checkbox"/>	WmShnekLeft		Битовый	104404	1	
70	<input type="checkbox"/>	WmShnekLeft		Битовый	104404	2	
71	<input checked="" type="checkbox"/>	BtnEmStop	Кнопка аварийного останова нажата	Битовый	104406	3	

рис.2. Экран создания и настройки ПЛК переменных

Каждая переменная имеет свой набор характеристик, среди которых её описание для оператора, тип хранимого значения (битовый/целочисленный/с плавающей точкой), и адрес в области разделяемой памяти, задаваемый путем

указания адреса байта и бита, к которому привязана данная переменная. Переменные также принадлежат к одной из трех категорий: пользовательские, аппаратные (соответствующие входам/выходам) и ЧПУ-переменные (содержат информацию о работе системы). Используя подобный набор переменных имеется возможность создания дополнительных терминальных решений, визуализирующих их значения и позволяющий их изменять, для контроля за процессом и его управления [4].

Создание визуальной части дополнительных терминальных решений осуществляется с помощью специального разработанного конструктора интерфейсов. Конструктор позволяет визуально располагать необходимые визуальные компоненты и производить их параметризацию, т.е. привязку к ПЛК-переменным, описанным выше. Набор визуальных компонентов включает в себя элементы для контроля параметров управляемого процесса (метки, индикаторы), а также компоненты для управления процессом (кнопки, регуляторы, поля ввода значений). На рисунке 3 представлен пример создания терминального решения с помощью конструктора, для контроля и управления технологическим оборудованием.

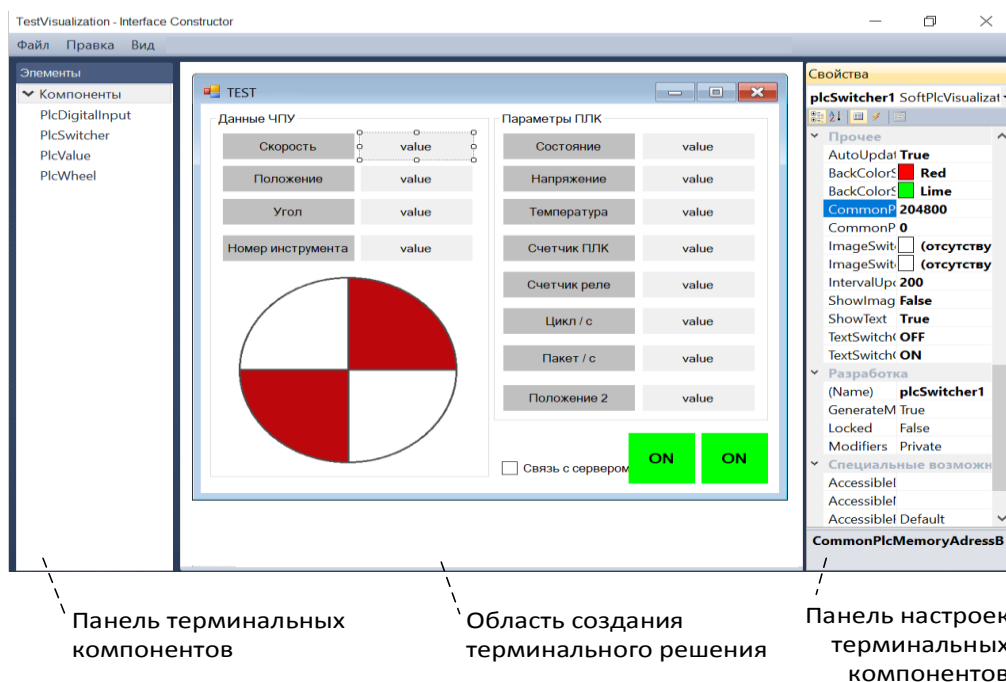


рис. 3. Среда разработки терминальных решений

Конструктор терминальных интерфейсов содержит панель компонентов, в которой находятся основные объекты, которые может создавать оператор для управления или мониторинга процесса; область создания терминального решения – главное окно, в котором создается интерфейс оператора; панель настроек терминальных компонентов – область, в которой производится настройка визуального отображения каждого из компонентов и указываются настройки их привязки к областям памяти ядра системы. В представленном примере используются объекты PlcValue, предназначенные для отображения статуса как битовых переменных (true/false), так и переменных с числовыми значениями. Помимо этого, в представленном примере используется компонент PlcSwitcher для возможности подачи битовых сигналов на выходы контроллера, а также компонент PlcWheel для визуального отображения положения и скорости объекта вращения (например, круговой оси) [2,5].

Привязка визуальных компонентов к значениям переменных ПЛК может производиться как вручную путем задания адреса разделяемой памяти, так и автоматически, используя привязку к определенной переменной из соответствующей таблицы в ПЛК редакторе. После создания терминальных компонентов и их параметризации производится сохранение проекта интерфейса в специальный файл.

Воспроизведение разработанного проекта интерфейса осуществляется с помощью специально разработанной утилиты, позволяющей отображать разработанный интерфейс оператора с привязкой к значениям из областей памяти ядра системы. Таким образом реализуется возможность производства дополнительного контроля за состоянием работы отдельных узлов технологического оборудования, а также получать информацию о работе системы управления.

Представленный в статье механизм по созданию дополнительных терминальных решений предоставляет возможность оперативно создавать человеко-машинный интерфейс для управления разнородным технологическим оборудованием. Особенностью представленного механизма является решение для связи терминальных компонентов с ядром системы и непосредственно объектом управления путем использования механизма разделяемой памяти, что позволяет получать информацию от объектов управления и осуществлять его управление без вмешательства в работу ядра системы. Разработанный набор терминальных компонентов позволяет покрыть большинство задач логического управления, в частности, осуществления контроля за работой объекта, а также подаче на него сигналов управления [4,7].

Помимо этого, разработанный механизм позволяет повысить уровень гибкости системы управления, предоставляя ее наладчикам возможности по созданию дополнительных терминальных решений для контроля и управления процессом. Важным фактором здесь является отсутствие необходимости участия в данном процессе производителя системы управления, что говорит о высоком уровне открытости разработанных механизмов. Также это позволяет упростить процесс запуска и ввода в эксплуатацию технологического оборудования, повысить возможность контроля за работой оборудования и уровень информатизации на предприятии, поскольку разработанный механизм предоставляет также возможности передачи полученных данных на более высокие уровни предприятия.

Литература

1. Мартинов Г.М., Никишечкин П.А., Григорьев А.С., Червоннова Н.Ю. “Организация взаимодействия основных компонентов в системе ЧПУ АксиОМА Контрол для интеграции в нее новых технологий и решений” // Автоматизация в промышленности. 2015. №5. с.10-15.
2. Нежметдинов Р.А., Никишечкин П.А., Ковалев И.А., Червоннова Н.Ю. Подход к построению специализированных терминалов для контроля и управления технологическим оборудованием // Автоматизация в промышленности. – 2018. – №6. – С. 58–62.
3. Jorge E. Correa, Nicholas Toombs, Placid M. Ferreira “A modular-architecture controller for CNC systems based on open-source electronics” // Journal of Manufacturing Systems, vol. 44, Part 2, July 2017, pp. 317-323, 2017.
4. Kovalev I.A., Nikishechkin P.A., Grigoriev A.S. “Approach to Programmable Controller Building by its Main Modules Synthesizing Based on Requirements Specification for Industrial Automation” // 2017 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), 16-19 May, 2017.p.1-4.
5. Никишечкин П.А., Ковалев И.А., Григорьев А.С., Никич А.Н. “Построение кроссплатформенной системы для сбора и обработки диагностической информации о работе технологического оборудования на промышленных предприятиях” // Автоматизированные технологии и производства. 2016. №4(14).- С. 51-56.
6. Georgi M. Martinov, Sergey V. Sokolov, Lilija I. Martinova, Anton S. Grigoryev, Petr A. Nikishechkin “Approach to the Diagnosis and Configuration of Servo Drives in Heterogeneous Machine Control Systems” // 8th International Conference, ICSI 2017 Fukuoka, Japan, July 27 – August 1, 2017 Proceedings, Part II, pp.586-594.
7. Никишечкин П.А., Ковалев И.А., Григорьев А.С., Никич А.Н. “Кроссплатформенная система сбора и обработки диагностической информации о работе технологического оборудования” // Вестник МГТУ Станкин. – 2017. – № 1 (40). – С. 94-98.