

УДК 621.7.06; 621.9.06

Г.М. Мартинов, д-р техн. наук, проф., Н.В. Козак, канд. техн. наук,
Р.А. Нежметдинов, канд. техн. наук, А.Б. Любимов
(Московский государственный технический университет «Станкин»)

neramil@ncsystems.ru

СПЕЦИФИКА ПОСТРОЕНИЯ ПАНЕЛЕЙ УПРАВЛЕНИЯ СИСТЕМ ЧПУ ПО ТИПУ УНИВЕРСАЛЬНЫХ ПРОГРАММНО-АППАРАТНЫХ КОМПОНЕНТОВ

Предложено архитектурное решение для построения терминальной части современных систем ЧПУ. Раскрыты механизмы программной эмуляции для аппаратной реализации панелей управления. Приведены практические примеры использования микроконтроллеров и контроллеров электроавтоматики при построении терминальной части систем ЧПУ. Работа выполнена по Госконтракту № П693 на проведение НИР в рамках ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг.

Ключевые слова: система ЧПУ, микроконтроллер, программируемый логический контроллер, панель оператора, станочная панель, электроавтоматика станка.

Architectural decision for modern CNC systems terminal part construction is proposed. Program emulation machinery for control panels are disclosed for hardware realization. Practical examples are resulted for using microcontrollers and electroautomatics controllers at construction CNC terminal part. Work is performed in accordance with state contract № П693 on NIR (scientific-research work) fulfill in the framework of FSP (Federal target program) «Scientific and scientific-pedagogical personnel of innovative Russia» for 2009–2013.

Key words: CNC system, microcontroller, programmed logical controller, operator panel, machine-tool panel, machine-tool electroautomatic.

Введение. Необходимость в разработке собственных устройств и приборов для систем управления оправдывает себя только при недостаточности конструкторских решений на рынке. Панель оператора и станочная панель традиционно разрабатываются согласно специфичным требованиям конкретного станкостроителя.

Вне зависимости от происхождения программно-аппаратных средств (покупные изделия или оригинальное производство) возникает вопрос об их интеграции в системы управления. При этом определяющими факторами становятся: открытость архитектуры системы управления [1, 2] и используемых технологий, наличие стандартизированных интерфейсов взаимодействия с драйверами встраиваемых устройств.

Практически все системы ЧПУ выпускаются с панелями управления, но каждый производитель использует своё решение. Проблема заключается в отсутствии единых подходов и стандартов реализации.

Подключение панелей управления в двухкомпьютерную архитектуру системы ЧПУ. Терминальная часть системы ЧПУ имеет модульную архитектуру (рис. 1). В её состав входят:

панель оператора, содержащая жидкокристаллический монитор с широким углом обзора (не

менее 150° по горизонтали и вертикали), и ряд машинных и функциональных клавиш;

станочная панель, управляющая функционалом системы ЧПУ;

компьютер промышленного исполнения, реализующий терминальную задачу управления системы ЧПУ;

клавиатура промышленного исполнения для ввода текстовой информации.

Применение микроконтроллеров на основе процессора ARM при реализации панелей оператора предоставляет ряд преимуществ:

компактность устройств;

программирование на привычных для разработчиков языках ANSI C и C++, что сокращает время разработки;

поддержка стандартных каналов связи с периферийными устройствами, таких как USB, RS-485;

возможность быстрого перепрограммирования непосредственно на объекте управления через USB-порт;

возможность подключения по каналу связи RS485 для взаимодействия в режиме реального времени;

возможность отладки управляющих программ контроллеров непосредственно на панели оператора; работа в промышленном диапазоне температур.

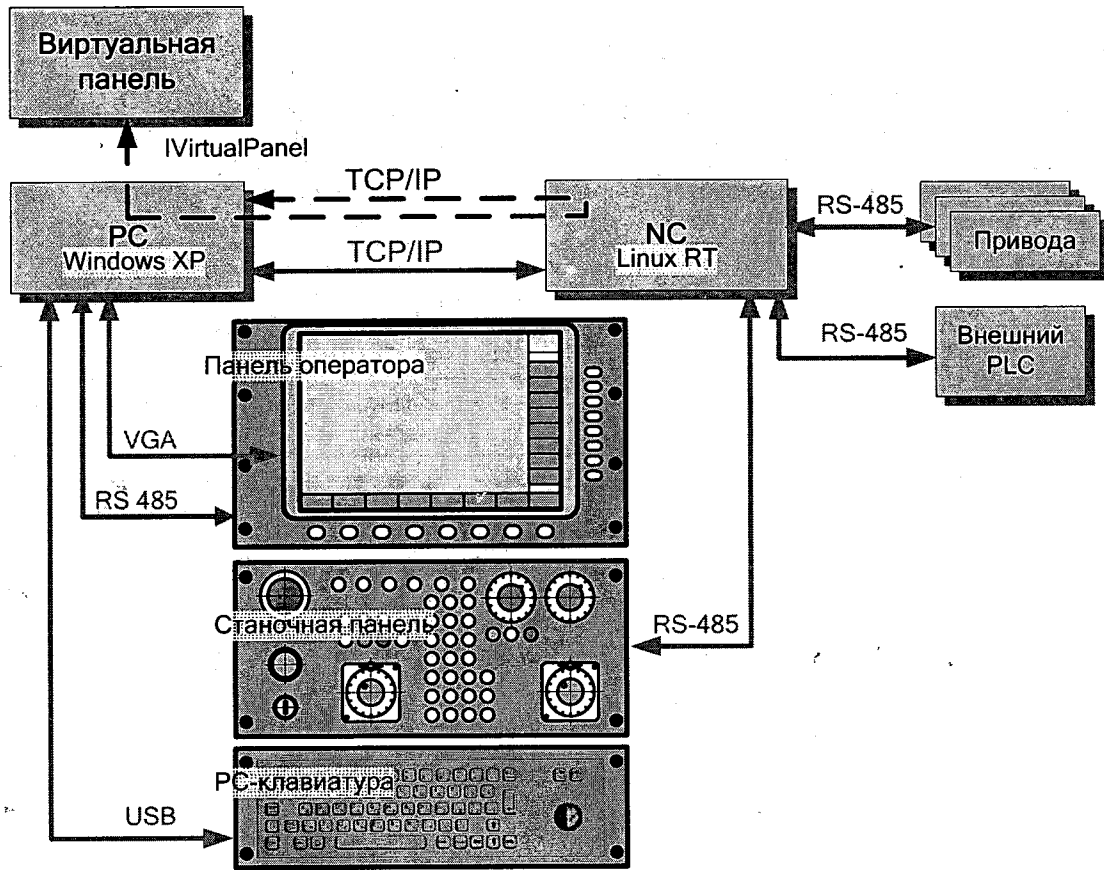


Рис. 1. Подключение панелей управления системы ЧПУ

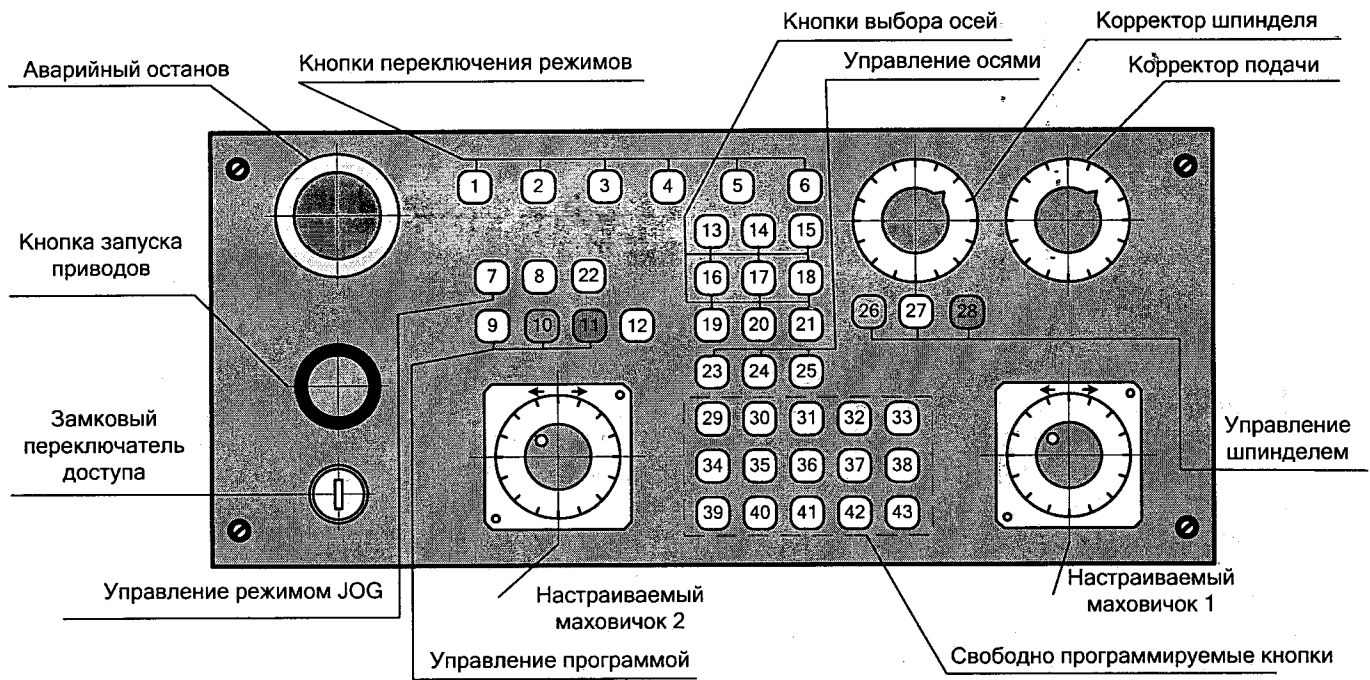


Рис. 2. Обобщённый вид станочной панели с двумя маховичками

Т а б л и ц а 1

Систематизация элементов управления станочной панели

Тип	Механизм работы	Элементы управления
Механический замок	Активирует систему при повороте уникального ключа	<i>Замковый переключатель</i> , блокирует кнопку запуска приводов и используется для санкционированного доступа к станку
Кнопки с фиксацией	Кнопка с фиксацией нажатия	<i>Аварийный останов</i> , останавливает привод и шпиндели с последующим отключением привода станка. При этом система ЧПУ и контроллер PLC не должны обесточиваться
Кнопки без фиксации	Кнопки без фиксации нажатия	<i>Кнопка запуск приводов</i> , включает силовое питание на приводы
Радиокнопки	Группа кнопок, лишь одна из которых активна.	Кнопки переключения режимов: <i>реферирования оси, ручной, возврат на контур, полуавтоматический, ручного ввода, автоматический</i> Управления программой: <i>запуск программы, останов программы</i> Кнопки выбора осей Управление шпинделем: <i>пуск шпинделя, останов шпинделя</i>
Импульсные кнопки	Кнопка активна лишь в момент её удерживания в нажатом состоянии	Сброс канала, перемещение оси в положительном и отрицательном направлениях
Кнопки с «памятью»	При однократном нажатии кнопка активируется, при последующем нажатии деактивируется.	Управление ручным режимом, включение ускоренного перемещения, реверс шпинделя
Потенциометры	Обеспечивают ступенчатую регулировку параметров	<i>Корректор шпинделя, корректор подачи</i> регулируют частоты вращения шпинделя и подачу соответственно
Маховички (штурвалы)	Управление приводами в ручном режиме	Штурвал 1, штурвал 2.

Т а б л и ц а 2

Пакет данных запроса о состоянии клавиатуры

Байт	0	1	2	3	4	5	6
Значение	ENQ	Адрес устройства		К	А	Контрольная сумма	

Таким образом, применение готовых встраиваемых типовых решений на базе ARM является достаточно гибким и расширяемым по функциям программного обеспечения.

Станочная панель (рис. 2) включает в себя элементы управления, которые систематизированы в табл. 1.

Выбран интерфейс связи RS-485 в силу его доступности и широкого выбора на рынке оборудования для его реализации, а также из-за его высокой устойчивости к помехам и дешевизны решений.

На основе интерфейса RS-485 был реализован протокол взаимодействия панелей управления с

системой ЧПУ, в котором в качестве ведущего устройства выступает ядро реального времени, а в качестве ведомых — терминальная панель оператора и станочная панель.

Формирование протокола обмена данными с панелями оператора. К протоколу предъявляют следующие ключевые требования:

- высокая скорость обмена данными при работе со станочной панелью;
- возможность расширения;
- обеспечение совместимости с предыдущими версиями;
- простота реализации.

Таблица 3

Пакет данных чтения состояния панели оператора

Значение	0	1	2	3	4	...	9	10	...	15	16	17	18	19	20	21	22	23	24	25	26	27
STX																						
Адрес устройства																						
Байт конфигурации пакета																						
Биты скан-кода 1																						
Биты скан-кода 2																						
Положение переключателя 1																						
Положение переключателя 2																						
Абсолютное положение штурвала 1																						
Абсолютное положение штурвала 2																						
Контрольная сумма																						

Опрос аппаратных элементов управления панелей терминальной части производится с установленным периодом в 20 мс. Такая частота опроса реализуется при установленной скорости соединения RS-485 в 921600 бит/с. Команда запроса предусматривает 6 байт и имеет обобщенный формат, представленный в табл. 2.

Тип запроса определяется в трёх и четырёх байтах пакета:

КА — чтение всего состояния клавиатуры, при котором запрашиваются все данные о нажатых между опросами клавишах панели управления, положения переключателей и абсолютные значения положения штурвалов;

КС — чтение текущего состояния клавиатуры, реализуется посредством запроса информация только об изменениях, произошедших между опросами;

КВ — чтение текущих данных только о нажатии клавиш панели.

Информация о нажатых или зажатых клавишах передается в виде скан-кода в обязательной части пакета данных протокола обмена данными (табл. 3). Необязательная часть пакета передается только в случае запроса КА и КС.

Запрос чтения текущего состояния клавиатуры (КС) включает в необязательную часть запроса только данные об изменившихся значениях. Например, если за время интервала опроса оператор вращает лишь один штурвал, то в необязательную часть пакета добавляются байты данных об абсолютном положении этого штурвала.

Пакет данных чтения состояния панели оператора может иметь различную длину. Минимальный размер пакета данных чтения состояния панели составляет 18, а максимальный — 28 байт. Набор частей пакета определяется в байте конфигурации (третий байт). Байт конфигурации устанавливается в соответствующее значение по мере необходимости включения или не включения информации об изменениях положения переключателей и штурвалов панели.

Способ универсального подключения панели управления к системе ЧПУ. Разработка и отладка механизмов работы ядра системы ЧПУ должна производиться на той же исполнительной платформе, под которой она будет функционировать (в данном случае в среде Linux). Однако имеется возможность использовать все преимущества интегрированных сред разработки платформы MS Windows [3], таких как, например, MS VisualStudio, которая обладает более широким инструментарием разработки кода приложений и диагностики его исполнения. Для этого была применена библиотека Cygwin, которая реализует интерфейс прикладного программирования POSIX на основе системных вызовов Win32. Библиотека Cygwin предоставляет набор свободных программных инструментов, которые позволяют превратить операционную систему MS Windows в некоторое подобие UNIX-системы.

Обеспечение совместимости достигается посредством классов оболочек (*wrappers*), инкапсулирующих платформо-зависимые API-вызовы (рис. 3). Классы и модули, реализующие логику работы ядра системы управления, используют предоставляемые уровнем оболочек функции, что обеспечивает им платформо-независимую реализацию.

Код функций классов оболочек имеет две реализации, разграниченные соответственно глобальными флагами:

_Win32: с применением функций среды Windows или функций Cygwin, для которых в Win32 API нет аналогов;

_Linux: с применением вызовов функции платформы Linux.

Например, функции для управления процессами под Linux используют механизм сигналов на основе функции *sigaction*, который удобнее использовать по сравнению с функцией *signal* из Win32 API. Функция обращения к функции *sigaction* имеет двойную реализацию, где в части **_Win32** производится обращение функции импорта из библиотеки Cygwin:

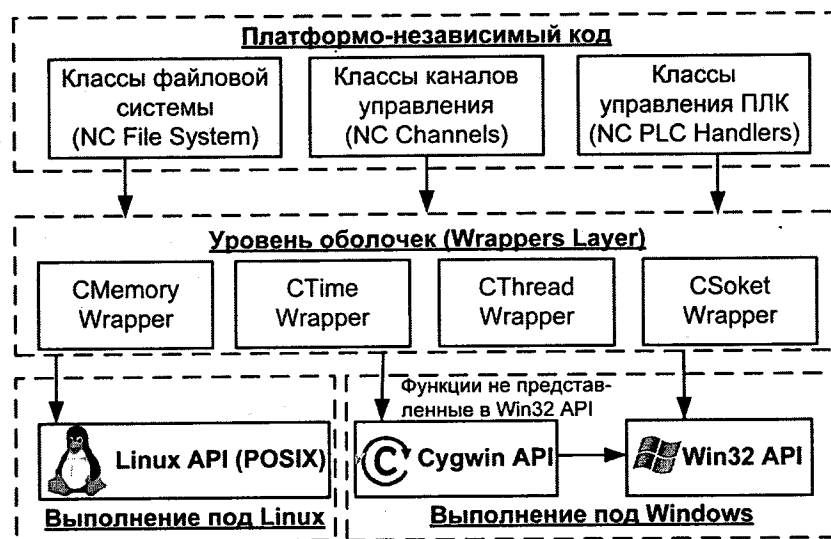


Рис. 3. Структура реализации отладочной версии ядра системы ЧПУ

```
int CFuncWrapper::Func_sigaction(int
sig, struct sigaction * act,
struct sigaction * oact)
{
#ifdef __linux__
return sigaction(sig, act, oact);
#elif defined _WIN32
return m_s_this->m_p_sigaction(sig,
act, oact);
#endif
}
```

Инструментарий библиотеки Cygwin позволяет в среде Windows:

проверять правильность Linux-синтаксиса кода; осуществлять сборку программной библиотеки с помощью компилятора gcc;

выполнять программный код, а также с помощью отладчика gdb производить отладку из набора Cygwin.

Естественно, что ограничения на выполнение функций ядра в среде Windows будет отсутствие реального времени и ограниченные возможности по использованию аппаратных устройств (таких как, например, платы ввода – вывода для взаимодействия с приводами и контроллерами электроавтоматики).

Создан эмулятор станочной панели, реализующий её функции. Эмулятор станочной панели применяют в обучающих версиях систем ЧПУ, избегая необходимости использования аппаратных панелей управления.

Механизм передачи данных (рис. 4) от аппаратных панелей станка или от их программных эмуляторов на терминальную часть системы управления идентичен. Идея заключается в том, что в ядре системы область памяти, предназначенная для хранения данных о состоянии внешних устройств,

распределяется один раз (поз. 1 на рис. 4). Поток опроса устройств (поз. 1.1) циклически запрашивает данные у станочной панели, контроллеров электроавтоматики, у прочих требуемых устройств и записывает их в специальную область. Далее поток обновления данных (поз. 2.1) сканирует область данных и обновляет поля объектов данных внешних устройств (машинных клавиш – CMKey, функциональных клавиш – CFKey, клавиш станочной панели – CNCKey, переключателей – CNCSelector, штурвалов – CNCWheel), выставляя в измененных объектах флаги о том, что данные об объекте изменены и требуют отправки в терминальную часть. В свою очередь, поток событий (поз. 3.1) обрабатывает объекты с выставленным флагом и посылает протокол TCP/IP серверу (поз. 3) информацию об изменениях в объектах данных. На основе этой информации сервер формирует пакеты об обновлениях в состоянии устройств, которые отправляются в терминальную часть системы управления.

Преимущества предложенного механизма следующие:

возможность реализации асинхронного взаимодействия с драйверами устройств посредством применения нескольких потоков;

независимость от источника данных: из реальных панелей или от их программных аналогов (механизм работы с ними в ядре идентичен);

данные о состоянии панелей управления предоставляются одновременно как в ядро реального времени, так и в терминальную часть.

Открытость архитектуры рассматриваемого программного обеспечения предоставляет станкостроителям возможность использования машинных клавиш терминальной панели для реализа-

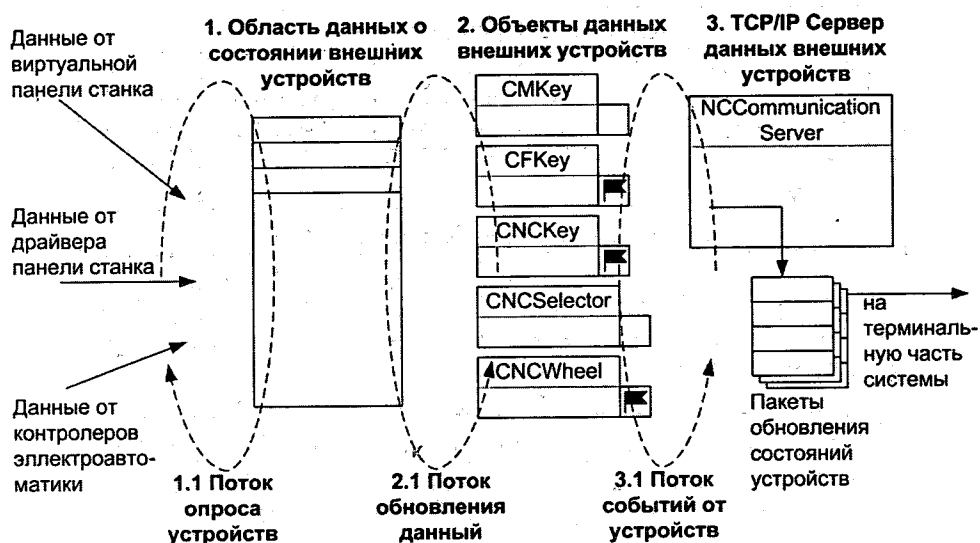


Рис. 4. Механизм передачи данных от устройств станка

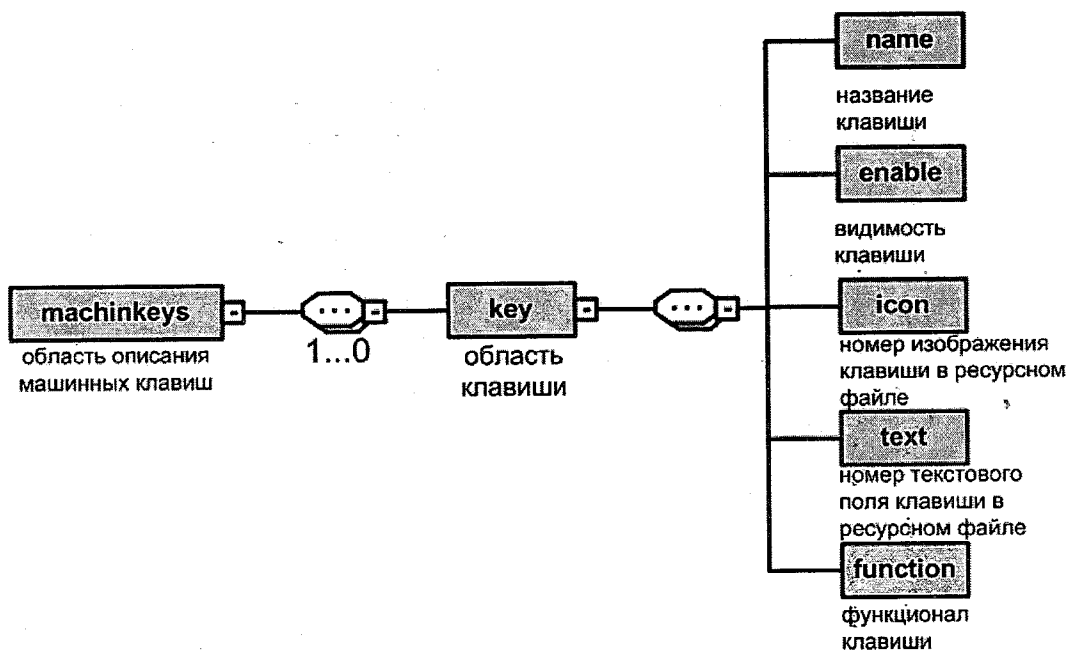


Рис. 5. XSD-схема, описывающая функциональные клавиши

ции собственных функций управления электроавтоматикой станка.

Механизм программирования и работы М-клавиш терминальной панели. Машинные клавиши (М-клавиши) панели оператора системы ЧПУ привязывают к функциям электравтоматики — это может быть вызов функции принудительного включения охлаждающей жидкости, отвод стружки и т. д. Логика их работы под конкретный станок реализуется станкостроителем [3]. Разработчики систем ЧПУ предоставляют лишь механизм программирования М-клавиш.

Параметры, необходимые для отображения строк и картинок на функциональных М-клавишах, хранятся в XML-файлах настройки. Эти файлы формируются на основе XSD-схем (рис. 5). Программное обеспечение системы ЧПУ, отвечающее за настройку параметров отображения на М-клавишах, осуществляет визуализацию параметров на экране оператора в соответствии с XML-файлом. В XML-документе указываются регистры программируемого логического контроллера (ПЛК), из которых считываются данные, необходимые для формирования визуально-

