

Свойства композиционных материалов зависят от состава компонентов, их сочетания, количественного соотношения и прочности связи между ними. Армирующие упрочняющие материалы могут быть в виде волокон, жгутов, нитей, лент, многослойных тканей.

Содержание упрочнителя в ориентированных материалах составляет 60—80 %, внеориентированных (с дискретными волокнами и штевидными кристаллами) — 20—30 %. Чем выше прочность и модуль упругости волокон, тем выше прочность и жесткость композиционного материала. Свойства матрицы определяют прочность композиции при сдвиге и сжатии и сопротивление усталостному разрушению.

Преимуществом композиционных материалов являются высокие прочность и жесткость $\sigma_b = 65 \div 170 \text{ кгс/мм}^2$, $E = 12000 \div 18000 \text{ кгс/мм}^2$, хорошее сопротивление хрупкому разрушению, жаропрочность и термическая стабильность. Плотность композиционных материалов составляет от 1,35 до 4,8 г/см³.

Композиционные материалы являются перспективными конструкционными материалами для различных отраслей машиностроения.

Список использованных источников

1. Материаловедение и технология металлов: учебник для ВУЗов по машиностроительным специальностям / Г. П. Фетисов, М. Г. Карпман, В. М. Матюнин и др. — М.: Высшая школа, 2000. — 637 с.: ил.
2. Богодухов, С. И. Курс материаловедения в вопросах и ответах: учеб. пособие для ВУЗов, обуч. по направлению подгот. бакалавров «Технология, оборуд. и автомат. машиностр. пр-в» и спец. «Технология машиностроения», «Металлорежущие станки и инструмент» и др. / С. И. Богодухов, В. Ф. Гребенюк, А. В. Сипохин. — М.: Машиностроение, 2003. — 255 с.: ил.
3. Лахтин, Ю. М. Материаловедение: учебник для машиностроительных вузов / Ю. М. Лахтин, В. П. Леонтьева. — 2-е изд., перераб. и доп. — М.: Машиностроение, 1980. — 493 с., ил.

УДК 621.7.06, 621.9.06

АВТОМАТИЗАЦИЯ ПРОЦЕССА ТЕСТИРОВАНИЯ МНОГОФУНКЦИОНАЛЬНЫХ СИСТЕМ ЧПУ

Л. И. Мартинова, И. А. Ковалев, А. В. Комаров

Московский государственный технологический университет «СТАНКИИ»,
г. Москва

Современные станки с ЧПУ оснащены большим количеством механических, пневматических, гидравлических и электронных устройств и элементов, от корректного и надежного функционирования которых в значительной степени зависит точность исполнения заданной программы и, соответственно, качество изготавливаемого изделия. Так, автоматизированная система, управляющая крупногабаритными станками, имеет сложную структуру, распределенную на

территории завода или предприятия. Это особенно ощутимо в многофункциональных станках с ЧПУ, работающих в составе высокотехнологичных комплексов, таких как автоматизированные линии, конвейеры и пр., где значительно больше различных элементов станка, таких как датчики, контроллеры, модули ввода-вывода. Алгоритмы управления, в таких системах, сложные. При этом структура системы управления должна соответствовать структуре самого объекта автоматизации [1].

Специфика построения систем ЧПУ, обслуживающих крупногабаритные комплексы, состоит в том, что функции сбора, обработки данных, вычисления, диагностики и управления оказываются распределенными, в том числе, и между несколькими терминалами управления (рисунок 1).

В каждом терминале установлен свой контроллер управления, работающий со своей группой устройств и обслуживающий свою часть объекта управления (станка). Тенденция приближения терминалов к объектам управления является общей для всех систем автоматизации, чем вызвано появление много-терминальных систем. Задача отладки системы до уровня конечного продукта является важной частью на стадии разработки и изготовления. Достичь этого возможно путем качественного тестирования систем управления на различных этапах. Для этого необходимо иметь специальные испытательные стенды, конфигурируемые под конкретные задачи многофункциональных систем ЧПУ.

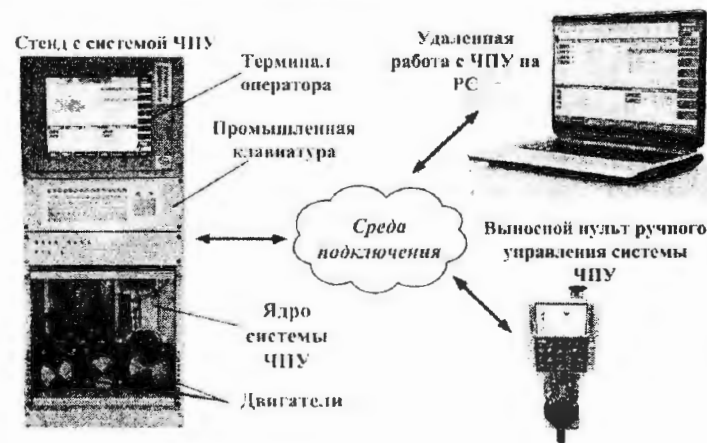


Рисунок 1 — Обобщенная структура распределенной компьютерной системы

На кафедре компьютерных систем управления МГТУ «СТАНКИИ» разрабатываются испытательные стенды для тестирования систем ЧПУ различного назначения, такие как стенды с интерфейсами EtherCAT, CAN, SERCOS и другие [2]. При создании испытательных стендов важным этапом является планирование архитектуры двухкомпьютерной системы так, чтобы она позволяла оп-

Оптимизировать проведение испытаний и ускорить процесс обработки данных (рисунок 2).

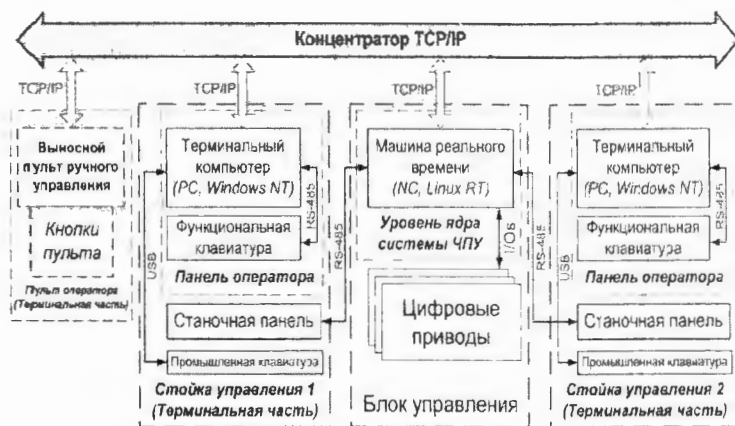


Рисунок 2 – Архитектура испытательного стенда

В представленной архитектуре используется двухкомпьютерная модель, предполагающая размещение PC-подсистемы на одном компьютере, а NC-подсистемы на другом. В PC-подсистеме наиболее целесообразна ОС семейства WindowsNT с платформой .NET, а в NC-подсистеме – операционная система LinuxRT [3]. Обе операционные системы совместимы в том смысле, что поддерживают коммуникационные протоколы ТСР/Р. Это позволяет построить коммуникационную среду, объединяющую подсистемы. Включение в эту среду прикладного уровня с функциями доступа к интерфейсам модулей создает виртуальную шину, оказывающую низкоуровневые услуги доступа [4].

Архитектурный вариант (рисунок 3) дает общее представление о принципах открытой двухкомпьютерной архитектуры применительно к ЧПУ. В системную платформу интегрированы такие элементы, как:

- операционная система;
- коммуникационная среда;
- кроссплатформенная реализация ядра;
- различные средства конфигурации.



Рисунок 3 - Открытая архитектурная двухкомпьютерная система типа PCNC

Интерфейс API допускает использование переносимых модулей, в том числе и от разных разработчиков. Коммуникационная среда является единственным средством информационного обмена между архитектурными объектами как в пределах одной вычислительной среды, так и за ее пределами в распределенной системе. Стандартные протоколы коммуникационной среды обеспечивают единообразные форматы данных и фиксированные наборы соотношений [5].

Многотерминальная система управления требует специальной отладки по набору параметров. Отладка и тестирование проводятся в лабораторных условиях в целях выявления проблем в работе терминалов, станочных панелей, контроллеров, их взаимодействий между собой и в механизмах передачи данных. Результаты тестирования должны быть представлены в виде протоколов, которые передаются разработчикам для внесения соответствующих изменений.

Многофункциональная система ЧПУ обладает широким набором свойств, предназначенных для управления большим спектром механообрабатывающего оборудования [6]. Отладка столь сложной архитектуры требует массу времени и опыта специалиста по тестированию. В связи с чем, необходимо было выявить основные параметры по тестированию систем с ЧПУ. Исследования показали, что такими параметрами являются:

- работоспособность станочных панелей и панелей оператора;
- работоспособность всех элементов, в том числе маховиков и переключателей, на панелях управления;
- достоверность передачи данных между ядром и терминалами в режиме реального времени;
- корректность межмодульного взаимодействия кроссплатформенного программного обеспечения;
- синхронная и корректная обработка данных терминалами;
- корректная работа приводов, контроллеров и модулей ввода-вывода;
- правильное подключение и настройка всех элементов, включая настройку машинных параметров;

– отсутствие сбоев при длительной работе всей системы в разных режимах.

Наиболее полному функциональному тестированию подвергаются много-терминальные системы управления крупногабаритным оборудованием (рисунок 4).

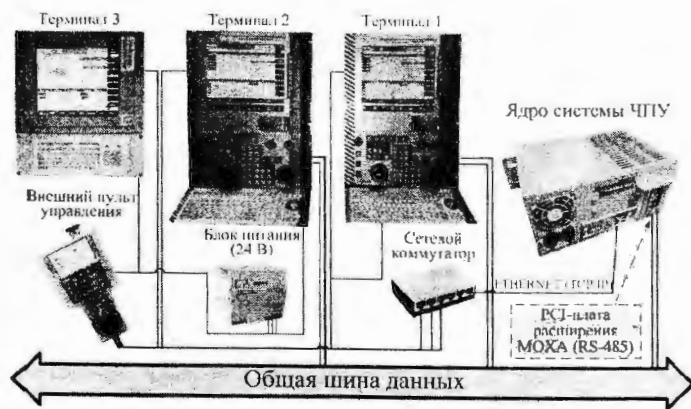


Рисунок 4 - Архитектура многотерминальной системы ЧПУ

Каждый терминал состоит из панели оператора, станочной панели и промышленной клавиатуры. Они получают информацию из ядра системы ЧПУ через сетевой коммутатор в Ethernet-сети. Информация передается в реальном времени, в соответствии со стандартом POSIX.1b. Станочные панели подключены к PCI-плате в ядре по специальным кабелям MOXA. Источник питания в 24В снабжает оборудование необходимым для работы их элементов, в том числе и кнопки аварийного торможения, электричеством. Промышленная клавиатура подключается к панели оператора по USB. Внешний пульт управления служит для дистанционной работы оператора со станком.

Архитектура тестируемых систем может различаться в зависимости от задач автоматизации, но испытательный стенд должен позволять проводить оценку системы по определенному набору показателей. ГОСТ 4.405-85 устанавливает номенклатуру основных показателей качества программируемых устройств ЧПУ. В таблице 1 представлены общие характеристики тестируемой системы.

Все практические аспекты построения конфигурации испытательных стендов, начиная от электрических схем соединения и заканчивая многоуровневыми программными алгоритмами, проверяются в лабораторных условиях в прямом взаимодействии с разработчиками программного обеспечения. Помимо приведенных выше параметров тестирования проверке подвергаются и ряд дру-

гих показателей. В частности, для оценки качества работы электрооборудования выполняют:

- проверку правильности подключения кабелей, разъемов, электрических схем;
- проверку качества экранирования и разводки кабелей;
- проверку аварийных блоков, во избежание нештатных ситуаций.

Таблица 1 – Общие характеристики тестируемой системы

Общие характеристики тестируемой системы	
Технические данные	Параметры
Управление станками	Токарными, фрезерными, шлифовальными, лазерными, расточными, специальными
Реализуемые оси	Линейные, круговые, бесконечно-круговые, переключение шпиндель/ось
Количество каналов	Многоканальная система (до 8 каналов)
Количество шпинделей	До 4
Количество управляемых осей	До 32 (до 16 осей на канал)
Типы интерполяции	Линейная, круговая, сплайновая
Программирование	G-коды, язык высокого уровня
Количество терминалов	До 8-ми терминалов, включая пульты ручного управления
Количество режимов работы канала	9
Технологические циклы	Расточка, токарная обработка, фрезерование
Интерфейсы	SECOS, EtherCAT, CANbus, UCSNet, Step Dir, Memobus, Modbus, PROFInet

Работа программной и/или аппаратной части оценивается по следующим параметрам:

- проверкой алгоритмов работы с набором аппаратных средств, реализованных в ядре ЧПУ;
- проверкой работы по внутреннему протоколу с ЧПУ;
- тестированием работы системы в различных конфигурациях;
- длительным тестированием на качество программной и аппаратной работы системы.

Для тестирования параллельного подключения двух мультипортовых плат PCI и PCI-Express написана специальная утилита. С её помощью тестировался драйвер для плат MOXA "mxchar", написанный под ОС Linux, где каждый порт имеет один или более файлов устройств в каталоге "/dev" ассоциированных с ним [7]. Для тестирования было необходимо обратиться к адресу порта в файловой системе и работать с ним

Листинг 1. Демонстрация кода для тестирования последовательных портов

```
void open_port(void) {
    int fd1, fd2; // файловый дескриптор для 1 и 2 порта
    ...
    // Откроем 1 порт
    fd1 = open("/dev/mxcharA0", O_RDWR | O_NOCTTY | O_NDELAY);
    ...
    // Откроем 2 порт
    fd2 = open("/dev/mxcharA1", O_RDWR | O_NOCTTY | O_NDELAY);
    ...
    // Переменная для записи данных в порт
    int byteswrite; // Переменная для чтение данных из порта
    ...
    byteswrite = write(...) // Запись в порт
    ...
    bytesread = read(...) // Чтение из порта
    ...
}
```

Формализация объекта тестирования является важной частью процесса отладки. При проведении тестирования вся полученная информация тщательным образом перепроверяется и документируется в протокол тестирования. Руководитель группы тестирования проверяет протоколы и адресует их разработчикам для исправления ошибок (рисунок 5). Решение багов перепроверяется группой тестирования и, если ошибка не повторилась, измененный исходный код закладывается в репозиторий.



Рисунок 5 – Концептуальная схема разработки программного обеспечения

Для хранения информации используются разного рода системы отслеживания ошибок - «багтрекеры» (рисунок 6), основной задачей, которых является сбор информации о зарегистрированных проблемах в программном обеспечении, а также слежение за процессом устранения ошибок проекта. Основная задача при обработке отчета об ошибке разработчиками - подтвердить наличие или отсутствие проблемы и сообщить, когда она будет исправлена.

Главным компонентом этой системы является база данных, которая содержит сведения о найденных ошибках в работе системы. Эти сведения включают в себя следующую информацию:

- идентификатор или номер ошибки;
- кем и когда была обнаружена ошибка;
- текущая версия продукта;
- приоритет ошибки;
- описание последовательности действий, приводящих к ошибке;
- воспроизводится ли ошибка на эмуляторе или только на «железе»;
- кто ответственен за устранение ошибки;
- комментарии по решению ошибки и устранению последствий;
- текущее состояние (статус) ошибки;
- версия продукта с исправленной ошибкой.

Кроме того, развитые системы предоставляют возможность прикреплять файлы, помогающие описать проблему (например, дампы памяти или скриншоты).

Используемый подход к созданию испытательных стендов перспективен для дальнейшего роста, в частности, применение нового оборудования, расширения конфигурации, усложнения электрической (принципиальной) схемы. Улучшение и последующая адаптация системы к промышленным условиям позволит достичь новых технологий в развитии системы ЧПУ.

Issue #68 **RESOLVED**

Неправильно реализован возврат из "Смены экрана" (Е. Трофимову)



Ncs Test created an issue 2/13/05-08

Из окна инициализации выйти в режим "Авто" или нажать "F2"

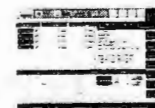
Нажать кнопку "Смена экрана" или "F7"

Нажать кнопку "Возврат" или "F9"

4. В результате вернуться в окно инициализации, вместо перехода в главное окно режима "Авто"

1. Нажать кнопку "Авто" или "F7" -> возвращаемся обратно в окно "Смена экрана"

Комментарий А В



Смена_экрана.jpg

Рисунок 6 - Скриншот разрешенной ошибки

В ходе тестирования многофункциональных систем ЧПУ были получены следующие выводы:

- испытательные стенды – надежный инструментарий отладки создаваемых систем управления;
- испытание разрабатываемых систем на стендах позволяет выявлять ошибки и проблемы в аппаратной и программной частях на стадии создания;
- стенды позволяют выполнять длительное тестирование в автономном режиме;
- архитектура стендов расширяема и может пополняться для испытания новых функциональностей и новых компонентов систем управления;
- обработка результатов тестирования и использование системы отслеживания ошибок (BugTrackingSystem) позволяет разработчикам оперативно вносить изменения в проект.

Список использованных источников

1. Денисенко, В. В. Компьютерное управление технологическим процессом, экспериментом, оборудованием / В. В. Денисенко. – М.: Горячая линия–Телеком, 2009. – 608 с.
2. Мартинов, Г. М. Подход к построению мультипротокольной системы ЧПУ [Текст] / Г. М. Мартинов [и др.] // Автоматизация в промышленности. – 2012. – № 5. – С. 8–11.
3. Мартинов, Г. М. Принцип построения распределенной системы ЧПУ с открытой модульной архитектурой / Г. М. Мартинов, Н. В. Козак, Р. А. Нежметдинов, Р. Л. Пушков // Вестник МГТУ «СТАНКИН». – 2010. – № 4 (12). – С. 116–122.
4. Мартинова, Л. И. Организация межмодульного взаимодействия в распределенных системах ЧПУ. Модели и алгоритмы реализации / Л. И. Мартинова, Г. М. Мартинов // Мехатроника, автоматизация, управление. – 2010. – № 11 (116). – С. 50–55.
5. Сосонкин, В. П. Системы числового программного управления / В. П. Сосонкин, Г. М. Мартинов // М.: Логос, 2005. – 296 с.
6. Григорьев, С. Н. Концепция построения базовой системы числового программного управления мехатронными объектами / С. Н. Григорьев, Г. М. Мартинов // Информационные технологии в проектировании и производстве, 2011. – № 2. – С. 21–27.
7. Michael R. Sweet Serial Programming Guide for POSIX Operating Systems / R. Michael // 5th Edition, 6th Revision. Copyright 1994–2005.

УДК 004

ПРИМЕНЕНИЕ СТАНДАРТНЫХ ВОЗМОЖНОСТЕЙ ПРОГРАММ ПРИ СОЗДАНИИ И ИСПОЛЬЗОВАНИИ БАЗ ДАННЫХ

А. Д. Припадчев, В. В. Шевченко, Р. З. Давлетьяров
Оренбургский государственный университет, г.Оренбург

В настоящее время для упрощения и оптимизации ПО, предназначенного для работы с данными, математическими зависимостями и графикой, используют все больше компьютерных программ при проектировании и отработке ЛА, что ведет к накоплению различного рода данных, неупорядоченных по принадлежности к исполняющей программе в электронном виде.

Создание единой информационной базы позволяет сократить время на поиск исходных данных для расчета ЛА. Но, как правило, у баз данных нет связи с исполняющими (расчетными, графическими, моделирующими испытания и др.) программами. Большинство современных стандартных текстовых, математических, графических и др. программ поддерживают ссылки перехода на другие программы, например пакеты Microsoft office, Matkad, APM Win Machin, Компас и др.

Для упрощения работы с формулами, графиками, текстовым содержанием документов, эмпирическими величинами и дальнейшего переноса их в соответствующие программы необходимо во время создания встроить в базу данных ссылки перехода на последующие программы. Таким образом, база данных и последующие программы будут связаны ссылками перехода. Ссылки перехода можно назначать не только на числовые значения, но и на формулы или группы формул. Так же для обратной связи вместе с данными из базы данных будет копироваться ссылка перехода для возврата в базу данных и ссылка на следующую программу (если требуется использовать данные полученные в предыдущей программе).

Данный принцип работы показан на рисунке 1.



Рисунок 1- Схема взаимодействия баз данных и различных программ