ления, который соответствует работе на компьютере [1-3]. Однако есть и более глубокая причина: внедрение новой технологии объектно-ориентированного программирования, без которой создание программного обеспечения систем ЧПУ в объеме многих мегабайт попро-

ний базовой платформы;

- доступ к информации о состояниях любого программного модуля системы, а также к диагностической информации аппаратуры, приводов и объекта;

- включение системы в сетевую коммуникационную среду.

Конфигурирование распространяется на выбор пользователем собственного диалога с системой; на дизайн многооконного экрана как на основе стандартной "галереи стилей", так и на основе собственных экранных управляющих элементов (control elements); на настройку системы на любую версию языка управляющих программ; на включение новых алгоритмов интерполяции (например, сплайновой интерполяции в реальном времени) и использование любой комбинации алгоритмов интерполяции в многокоординатном пространстве; на включение системы в сетевую коммуникационную среду [5].

Далее рассмотрены некоторые принципы построения систем ЧПУ типа PCNC с открытой архитектурой. Статья отражает позицию и практический опыт авторов при разработке подобных систем.

2. Модульная архитектура систем ЧПУ

Структура системы ЧПУ обусловлена традиционными задачами управления [6]:

- геометрической, подразумевающей в конечном счете управление следящими приводами станка с целью получения детали с заданной геометрией;
- логической, организующей управление электроавтоматикой станка;
- технологической, гарантирующей поддержание необходимых параметров технологического процесса или оптимизацию процесса;
- диспетчеризации, обеспечивающей управление на прикладном уровне четырьмя предыдущими задачами в реальном времени;
- терминальной, включающей диалог с оператором, отображение состояния системы; разработку, верификацию и хранение управляющих программ.

При всей традиционности этих задач последнее время привнесло в их состав и алгоритмику множество новых идей, реализованных в виде конкретных функций [7-10].

В целом, структура системы ЧПУ (Рис. 1.) представляет собой совокупность базовых модулей (обведены сплошными линиями) и дополнительных модулей (обведены пунктирными линиями). При этом каждый из модулей закреплен за определенной задачей управления. К дополнительным модулям отнесены покупные программные системы. Модуль автономен и является вложенным объектом, то есть обладает алгоритмической структурой и структурой данных, а также и интерфейсной оболочкой, ориентированной на интерактивную работу в клиент-серверной среде. При этом каждый модуль является, как правило, сложной объектно-ориентированной структурой.

Эта структура представлена двумя подсистемами - NC-подсистемой и PC-подсистемой. Первая является ведущей и формирует среду функционирования в реальном времени ЧПУ-ориентированных модулей: интерпретатора (ISO-процессора [8]), интерполятора, встроенного программируемого контроллера, базы данных реального времени и (возможно) специального NC-приложения пользователя. PC-подсистема образует среду Windows-образного интерфейса пользователя и включает также инструментальную систему подготовки и тестирования управляющих программ, а также (возможно) другие приложения.

Взаимодействие модулей осуществляется посредством объектно-ориентированной магистрали, которая не только поддерживает программно-аппаратные коммуникационные протоколы, но и выполняет прикладные серверные функции благодаря специальной объектной оболочке. Это значит, что объектно-ориентированная магистраль служит единым механизмом предоставления модулям информационных услуг. Такая возможность отражена и в самих интерфейсах модулей: они могут предоставлять данные, или запрашивать данные, или делать и то, и другое.

Все модули, подключенные к общей объектно-ориентированной магистрали, могут запрашивать данные синхронным, асинхронным способами или по событию. Выбор механизма реализации запроса зависит от конкретной задачи. При синхронном запросе клиент (то есть модуль, осуществляющий запрос) останавливается в точке запроса и ждет ответа от модуля, обслуживающего запрос, до истечения таймаута. При асинхронном запросе клиент продолжает свою работу, а обработка ответа, независимо от времени его получения, выполняется специальной функцией (callback). Запрос по событию (синхронный, асинхронный) означает, что ответ будет получен только после изменения данных.

Структура, представленная на Рис.1, обозначает набор модулей, позволяет специфицировать их интерфейсы, выявляет типы запро-

сов, помогает составить техническое задание на объектно-ориентированную магистраль.

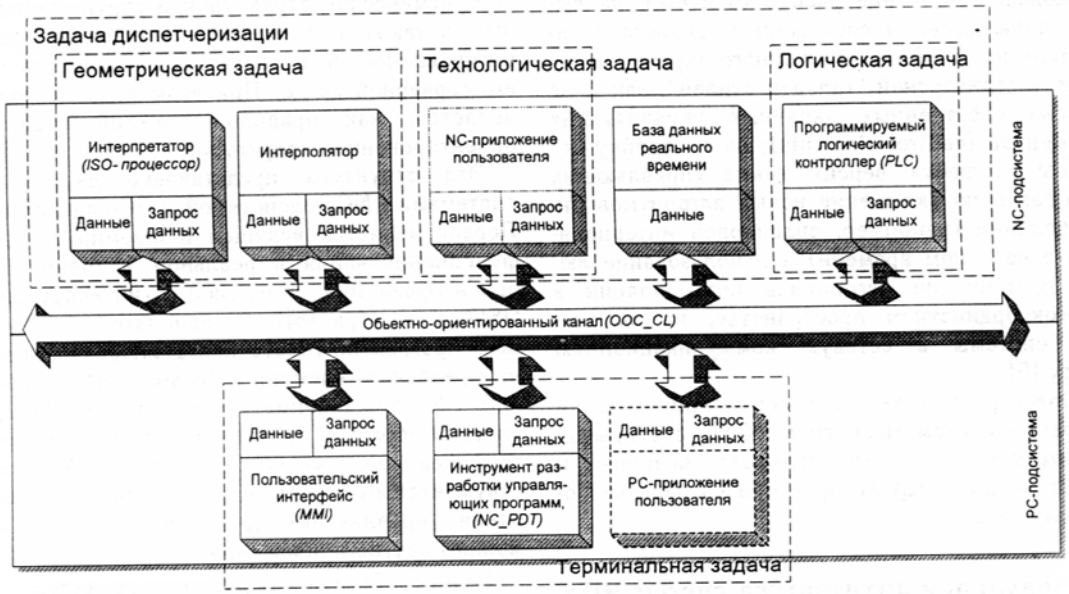


Рис.1. Модульная архитектура системы ЧПУ типа PCNC и задачи управления

3. Открытая архитектура систем ЧПУ

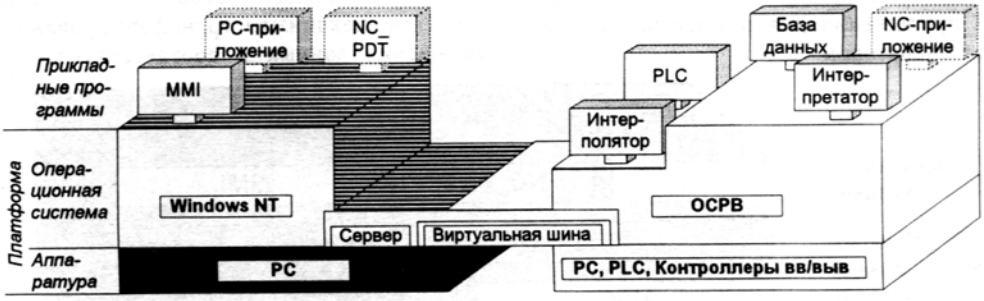
Сегодня особо гибкие и сложные системы ЧПУ типа PCNC с открытой архитектурой, ориентированные на многокоординатную, многостаночную, высокоскоростную, высокоточную обработку, - выполняются согласно двухкомпьютерной архитектурной модели (Рис.2а). В перспективе же - все преимущества за однокомпьютерным архитектурным вариантом (Рис.2б).

Двухкомпьютерная модель предполагает размещение PC-подсистемы на обычном PC-компьютере, а NC-подсистемы - на другом NC-компьютере. Вторым компьютером может быть традиционный, оснащенный дополнительными специальными устройствами, или компьютер на базе RISC-процессора.

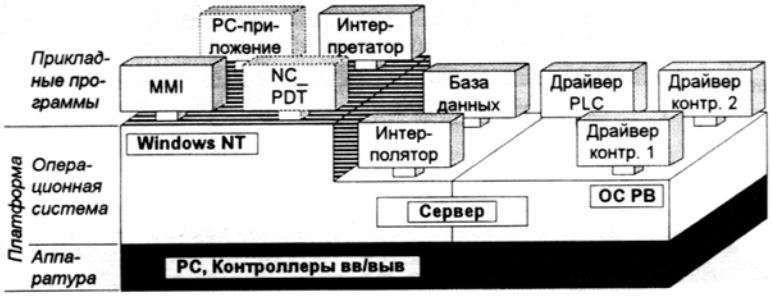
В PC-подсистеме наиболее целесообразна операционная система Windows NT, а в NC-подсистеме - операционная система UNIX. Обе операционные системы совместимы в том смысле, что поддерживают коммуникационные протоколы TCP/IP, и это позволяет организовать коммуникационную среду, объединяющую две подсистемы. Включение в эту среду некоторого прикладного уровня с функциями доступа к интерфейсам модулей (а общее число таких функций может достигать нескольких сот) создает виртуальную шину, оказывающую низкоуровневые услуги доступа. Наконец, объектная надстройка над этой шиной формирует глобальный сервер - единую для обеих подсистем объектно-ориентированную магистраль.

Однокомпьютерная модель предполагает использование обычного PC-компьютера, оснащенного специальными устройствами в виде плат-контроллеров. В их числе могут быть контроллер приводов подачи, программируемый контроллер PLC (Programmable Logic Controller), специфические устройства для управления технологическим процессом и так далее. Переход от двухкомпьютерной к однокомпьютерной модели осуществляется формальным переносом математического обеспечения PC-подсистемы в NC-подсистему на уровне задач.

Архитектурные варианты, показанные на Рис.2, дают общее представление о принципах открытой архитектуры применительно к ЧПУ: четкое разграничение между системным, прикладным и коммуникационным компонентами; возможность независимого развития любого из этих компонентов как на основе оригинальных разработок, так и путем встраивания покупных программных систем; клиент-серверная организация взаимодействия подсистем, стандартизация интерфейсов и транзакций.



а)



б)

Рис.2. Открытая архитектура систем типа PCNC

4. Виртуальная модель PC-подсистемы ЧПУ

В "вертикальном сечении" PC-подсистема имеет многоуровневую структуру (Рис.3) и в полной мере соответствует модели виртуальной машины [11].

Нижний уровень составляет компьютерная аппаратура, выше размещается операционная система (Windows NT, Windows 95) вместе с драйверами виртуальных устройств (VxD), обеспечивающими управление внешними устройствами, например, контроллером панели оператора. Доступ к операционной системе и службам осуществляется посредством API-слоя (Application Interface, прикладной интерфейс), который поддержан Win32-функциями и NC-функциями, обеспечивающими "вход" в Windows NT-подсистему и NC-подсистему. Функции реализованы в виде DLL (Dynamic Link Library, библиотека с динамическим связыванием).

Поверх API-слоя расположен объектно-ориентированный сервер, служащий фундаментом для всех приложений в PCNC системе. В числе классов объектов - стандартные из библиотеки MFC (Microsoft Foundation Classes), а также и специально разработанные классы OOC_CL объектно-ориентированной магистрали OOC (Object Oriented Channel). Сервер содержит общие для всех приложений алгоритмы, такие как обработчики ошибок (error handlers), средства форматирования и конвертирования данных, управляющие элементы экрана (control elements) и другие.

На прикладном уровне размещаются разнообразные приложения: интерфейс пользователя MMI (Man Machine Interface), инструмент разработки и верификации управляющих программ NC_PDT (NC Program Data Tool) и другие.

5. Виртуальная модель объектно-ориентированной магистрали OOC

В составе классов OOC_CL виртуальной модели магистрали OOC, представленной на Рис.4, выделены три основных уровня:

- базовый уровень, осуществляющий обмен данными с NC-подсистемой, их форматирование и контроль;
- уровень отображения информации на экран;

- уровень приложения, реализующий скелет MMI приложения для осуществления диалога с оператором.

eXchange). Классы отображения реализуют основные механизмы трансформирования и отображения данных в соответствующую среду

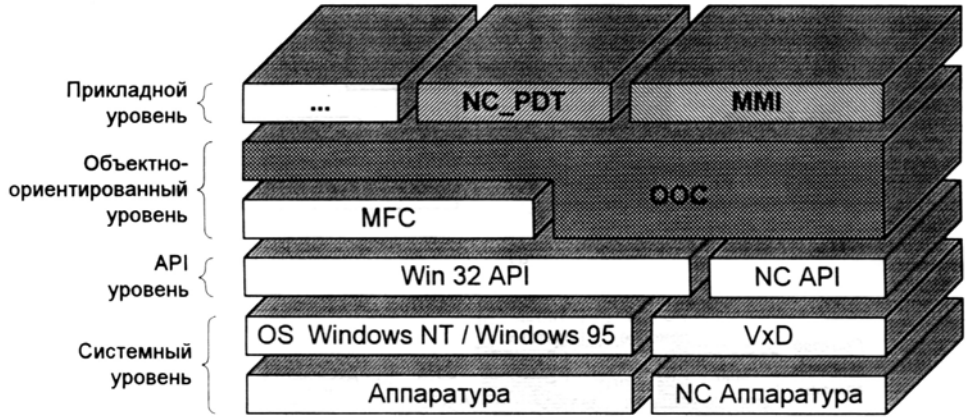


Рис.3. Виртуальная модель PC подсистемы

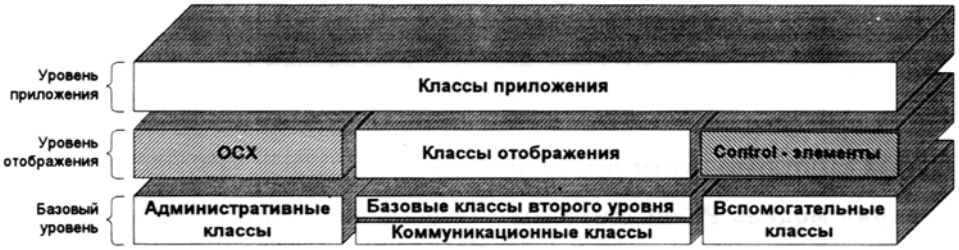


Рис.4. Виртуальная модель OOC

Базовый уровень включает административные, коммуникационные и вспомогательные классы объектов. Административные классы производят открытие и закрытие магистрали, устанавливают тайм-аут для синхроного запроса, производят диагностику и тестирование как отдельных модулей, так и самой магистрали. Коммуникационные классы поддерживают транзакции (сессии) различных типов запросов. На базе коммуникационных классов строятся классы рабочих процессов ЧПУ, эти классы запускают и контролируют процессы в NC-подсистеме. Вспомогательные классы переводят данные из одной размерности в другую, осуществляют масштабирование и форматирование данных.

На уровне отображения находятся классы отображения, галерея управляющих (control) элементов и галерея OOC_OCX элементов (OCX - это разновидность управляющих элементов, построенных по типу OLE Control

(экран, файл, принтер и так далее). Галерея управляющих элементов - это библиотека, включающая в себя управляющие элементы, специфические для экрана системы ЧПУ.

6. Настройка на различные версии языка программирования

Одна из ключевых проблем системы ЧПУ типа PCNC с открытой архитектурой состоит в необходимости настройки на любую версию языка управляющих программ, контуры которого определены в стандарте DIN 66025. Подобную настройку осуществляет специализированный модуль, названный ISO-процессором [8].

ISO-процессор является одним из важнейших модулей PCNC системы и обычно представляется двумя способами. В NC-подсистеме ISO-процессор реализован в виде интерпретатора; его задача заключается в интерпретации кадра управляющей программы, в эквидистант-

ных расчетах, в расчетах коэффициентов сплайнов и в формировании данных для интерполятора. В PC-подсистеме ISO-процессор реализован как эмулятор в инструменте разработки управляющих программ. Здесь его задача состоит в контроле и интерпретации кадра, а также формировании данных в формате интерполятора и в отображении этих данных на экране в виде модели рабочей траектории.

7. Инструмент настройки и верификации управляющих программ NC_PDT

Инструментальная система NC_PDT представляет собой среду программирования, маскирующую для пользователя низкоуровневые функции языка ISO-7bit (DIN 66025) управляющих программ.

Задачи инструментальной системы состоят в следующем:

Рис. 5. Обеспечение гибкости ISO-процессора путем выбора системы команд из базового набора

На Рис.5 представлен метод настройки ISO-процессора путем выбора системы команд из базового набора. Прообразом данного метода служит аппарат настраивания программных сред под конкретное аппаратное обеспечение. В ISO-процессоре выделены две зоны:

- зона управления при помощи пользовательского интерфейса;
- зона, невидимая для пользователя, - это внутренняя структура ISO-процессора, доступная только системным программистам.

Управляющие программы разбиты на пакеты по версиям языков, на которых они написаны. Выбор пакета управляющих программ определяет соответствующий файл загрузки системы команд ISO-процессора. Самотестируемая программа формирует G-вектор по умолчанию, анализируя каждую G-функцию текущей системы команд ISO-процессора по ее признаку. На системном уровне в интерполяторе обеспечивается поддержка всего набора доступных системных команд ISO-процессора. Это отражается в соответствующей конфигурации групповых интерпретаторов и их структуре.

- создавать и редактировать файлы управляющих программ, объем которых превышает десятки мегабайт;
- осуществлять полный синтаксический и частичный семантический контроль кадров управляющей программы;
- поддерживать блоковые операции с подпрограммами, в том числе операции конвертирования, масштабирования и перенумерации кадров;
- эмулировать отработку управляющей программы до заданной точки и вычислять активные G-функции (G-вектора);
- использовать графическое меню для составления управляющих программ; предоставлять полный сервис настройки пользовательского набора стандартных циклов (Рис.6).

Окно "Параметры" служит для ввода и редактирования кадра управляющей программы с такой панели оператора, которая не имеет полного набора буквенных клавишей стандартной клавиатуры (наиболее частый практический случай). Специальный язык описания (script

language) позволяет описывать версию языка управляющей прогаммы ISO-7bit (DIN 66025) в конфигурационном файле. В нем же описывают содержимое окна "Параметры" для каждой G-функции. Настройку и локализацию интерфейса NC_PDT на конкретного пользователя осуществляют с помощью файла инициализации.

Литература

1. Ю.М.Соломенцев, В.Л.Сосонкин. Управление гибкими производственными системами М. Машиностроение, 1988, с.352.
2. В.Л.Сосонкин. Взгляд на предстоящую эволюцию устройств ЧПУ. Станки и инструмент, 1992, №9, с.27-32.

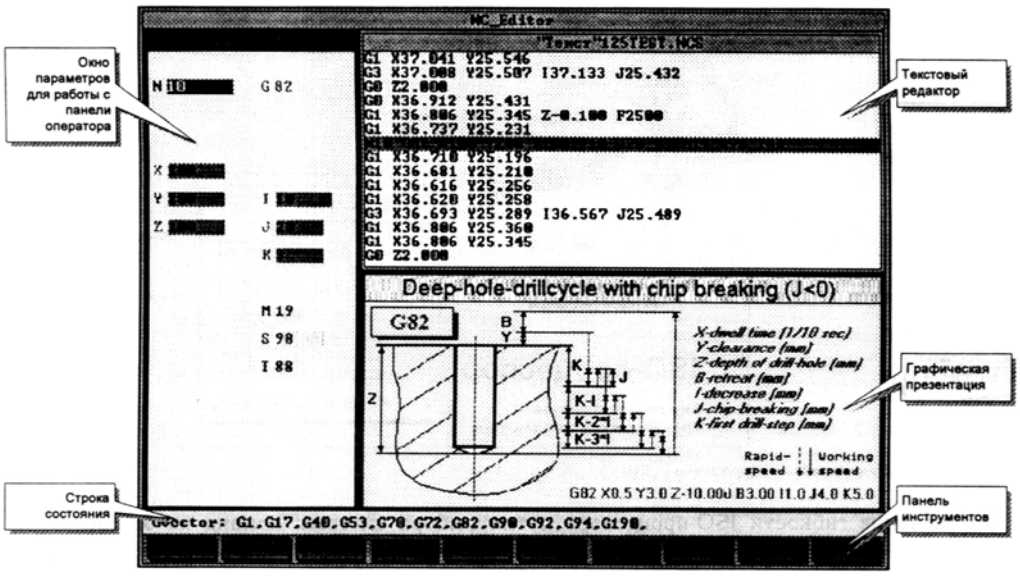


Рис.6. Инструмент подготовки и тестирования управляющих программ (NC_PDT)

Выводы

Основными признаками систем ЧПУ нового поколения являются принадлежность к классу персональных систем управления PCNC и использование принципов открытой архитектуры. Открытая архитектура в данном контексте предполагает глобальную гибкость (конфигурируемость) системы, использование клиент-серверного подхода в организации транзакций, привлечение объектно-ориентированного подхода к определению макро-структуры, а также и на уровне технологии программирования. Все это предопределяет принципиально иную (в сравнении с известными решениями) организацию системы ЧПУ, в которой даже модули с традиционными наименованиями имеют новое функциональное и алгоритмическое наполнение, а также и новую программную реализацию. Особо важную роль приобретает PC-подсистема, которая определяет пользовательские характеристики и уровень сервиса для оператора.

3. В.Л.Сосонкин. Концепция персональных систем управления в реальном времени. Приборы и системы управления, 1995, №7, с.15-17.
4. В.Л.Сосонкин, Г.М.Мартинов. Принципы построения систем ЧПУ с открытой архитектурой. Приборы и системы управления, 1996, №8, с.18-21.
5. В.Л.Сосонкин. Сетевая коммуникационная среда персональной системы управления. СТИН, 1996, №5, с.12-17.
6. В.Л.Сосонкин. Задачи числового программного управления и их архитектурная реализация. Станки и инструмент, 1988, №10, с.39-40.
7. В.Л.Сосонкин, В.Л.Потаскуев. Концепция программируемого контроллера нового поколения. Приборы и системы управления, 1992, №6, с.7-10.
8. В.Л.Сосонкин, Г.Мартинов. Концепция геометрического ISO-процессора для систем ЧПУ. СТИН, 1994, №7, с.17-20.

9. В.Л.Сосонкин. Разработка диспетчеров для систем управления с персональным компьютером. Приборы и системы управления, 1995, №2, с.14-18.
10. В.Л.Сосонкин, В.И.Клепиков. Принцип подчиненного управления в логических системах управления. Приборы и системы управления, 1995, №12, с.16-18.
11. В.Л.Сосонкин, Ю.Тилеш. Представление о процессорном устройстве числового программного управления оборудованием как виртуальном вычислителе. Машиноведение, 1981, №6, с.50-57.

Соломенцев Юрий Михайлович. Родился в 1939 году. Окончил Челябинский политехнический институт в 1968 г. Чл.-корр. РАН с 1987 г.. Профессор, доктор технических наук. Автор более 200 научных работ. Специалист в области конструкторско-технологической информатики, автоматизированных систем машиностроения. Ректор МГТУ "СТАНКИН", зав. кафедрой автоматизации проектирования и управления.

Сосонкин Владимир Лазаревич. Родился в 1939 г. Окончил МГТУ СТАНКИН в 1959 году. Доктор технических наук, зав. кафедрой компьютерных систем управления МГТУ СТАНКИН. 250 научных работ, 13 монографий, 28 авторских свидетельств. Область научных интересов - компьютерные системы управления.

Мартинов Георги Мартинов. Родился в 1966 году. Окончил МГТУ СТАНКИН в 1992 году. Кандидат технических наук, докторант кафедры компьютерных систем управления. Автор 9 научных работ. Область научных интересов - компьютерные системы управления.