

Федеральное агентство по образованию  
Государственное образовательное учреждение  
высшего профессионального образования  
Владимирский государственный университет

А.Д. ПОЗДНЯКОВ

ПРИБОРНО-МОДУЛЬНЫЕ СИСТЕМЫ  
КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА  
РАДИОАППАРАТУРЫ

Учебное пособие

Владимир 2005

УДК 621.396: 681.518.3

ББК 32.811.3

П47

Рецензенты:

Доктор технических наук, профессор  
зав.кафедрой радиотехники и радиосистем  
Владимирского государственного университета  
*О.Р. Никитин*

Доктор физико-математических наук, профессор  
зав.кафедрой теоретической физики  
Владимирского государственного педагогического университета  
*В.Г. Рау*

Печатается по решению редакционно-издательского совета  
Владимирского государственного университета

**Поздняков, А. Д.**

П47 Приборно-модульные системы контроля, испытаний и мониторинга радиоаппаратуры: учеб. пособие / А. Д. Поздняков; Владим. гос. ун-т. – Владимир: Ред.-издат. комплекс ВлГУ, 2005. – 118 с. – ISBN 5-89368-000-0.

В пособии приводятся общие теоретические сведения по построению приборно-модульных систем контроля, испытаний и мониторинга радиоаппаратуры. В результате изучения дисциплины студент должен понимать работу автоматизированных контрольно-измерительных и управляющих систем различного назначения, оптимизировать характеристики систем по заданным техническим требованиям, проводить экспериментальные исследования и программировать измерительные приборы, имеющие выход в канал общего пользования (КОП).

Предназначено для подготовки студентов в области проектирования приборно-модульных испытательных систем различного назначения: внешних и встроенных, универсальных и специализированных, технологических и эксплуатационных. Рекомендуются для студентов всех форм обучения направления «Радиотехника».

Табл. 11. Ил. 32. Библиогр.: 11 назв.

УДК 621.396: 681.518.3

ББК 32.811.3

ISBN 5-89368-000-0

© Владимирский государственный  
университет

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	3
Глава 1. ПРИНЦИПЫ ПОСТРОЕНИЯ СИСТЕМ КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА .....	6
1.1. Задачи автоматизации контроля, испытаний и мониторинга радиоаппаратуры .....	6
1.2. Архитектура систем контроля, испытаний и мониторинга .....	12
1.3. Программные средства систем контроля, испытаний и мониторинга.....	21
Глава 2. ПРИБОРНО-МОДУЛЬНЫЕ СИСТЕМЫ .....	26
2.1. Основные шины и линии интерфейса IEEE-488 .....	26
2.2. Интерфейсные команды и интерфейсные функции .....	29
2.3. Коды, форматы и обмен данными по шине .....	33
2.4. Реализация интерфейсных функций.....	37
2.5. Быстродействие АКИС с шиной IEEE-488.....	43
2.6. Принципы реализации интерфейса .....	49
2.7. Архитектура систем контроля, испытаний и мониторинга .....	52
2.8. Аппаратные средства расширения шины КОП.....	56
Глава 3. ПРОГРАММНЫЕ СРЕДСТВА СИСТЕМ КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА .....	61
3.1. Базовые и системные программные средства .....	61
3.2. Стандартные команды, форматы и протоколы .....	64
3.3. Компьютерные платы контроллера шины IEEE-488 и их программное обеспечение .....	71
3.4. Стандартные команды программируемых приборов .....	81
3.5. Методы и средства разработки программного обеспечения.....	106
ЗАКЛЮЧЕНИЕ .....	115
ПРИНЯТЫЕ СОКРАЩЕНИЯ.....	116
БИБЛИОГРАФИЧЕСКИЙ СПИСОК .....	117

## ВВЕДЕНИЕ

Современное массовое и серийное производство радиоаппаратуры характеризуется непрерывным обновлением выпускаемой продукции. Например, в секторе бытовой электроники и систем связи стабильный цикл процесса производства за последние годы сократился с 18 до 6 месяцев. В условиях жесткой конкуренции производитель вынужден модернизировать и осваивать выпуск новых видов изделий, непрерывно улучшая их качество. При таких темпах обновления для снижения затрат приходится искать пути сокращения расходов на контроль и испытания, поскольку доля этих расходов может достигать 50% от общих затрат.

Технологический процесс изготовления радиотехнических систем (РТС) различного назначения предусматривает типовые испытания, в частности приемо-сдаточные и периодические, а также проведение технологической тренировки (электрической, климатической и вибрационной) с проверкой функционирования и основных параметров испытуемого изделия. При многократных испытаниях РТС на этапах регулировки, сдачи продукции представителю ОТК и приемки заказчик затрачивает огромные ресурсы, что приводит к значительному увеличению себестоимости. Определяющим фактором гибкости производства становится быстрая разработка автоматизированных систем испытаний, позволяющих: повысить производительность, сократить стоимость и время разработки тестирующего оборудования, уменьшить затраты на создание и эксплуатацию систем контроля. Похожие проблемы возникают не только в сфере производства, но также при создании исследовательских СКИМ на этапе разработки новых РТС, при проведении комплексных испытаний и организации длительного мониторинга параметров сигналов и характеристик сложных технических объектов на этапе эксплуатации.

Таким образом, повышение эффективности *систем контроля, испытаний и мониторинга* (СКИМ) радиоэлектронной аппаратуры является важной народнохозяйственной задачей, решение которой обеспечивает

увеличение производительности и повышение качества выпускаемой продукции. Снижение стоимости испытаний, а также уменьшение времени разработки специализированных СКИМ – это ключевые задачи, от решения которых зависят производственные затраты, время внедрения продукции на рынок сбыта и в конечном итоге прибыль. Производителю сегодня нужны такие СКИМ, которые могут гибко изменять свои возможности в соответствии с совершенствованием выпускаемой продукции, т.е. они должны быть адаптивными, перепрограммируемыми, наращиваемыми и при этом недорогими.

Современные принципы построения автоматизированных СКИМ предполагают использование персональных компьютеров (ПК), модульных архитектур и стандартных интерфейсов [1-4, 6]. Компьютерные технологии позволяют с наименьшими затратами обеспечить высокое качество, точность и достоверность результатов испытаний. СКИМ сегодня – это гибкая совокупность аппаратно-программных средств, основой которых является ПК, обеспечивающий измерения, управление экспериментом, цифровую обработку данных, передачу и хранение информации.

Распространение компьютерных систем определяется большой вычислительной мощностью ПК, наличием развитого периферийного оборудования, удобством интегрирования систем в локальные сети, относительно невысокой стоимостью основного оборудования, разнообразием программ обработки данных и документирования. Клавиатура и экран ПК, простота интегрирования с другими внешними устройствами и системами предоставляют значительно более широкие возможности пользователю по сравнению с теми, которые могут дать автономные приборы. Вычислительная мощь ПК позволяет подвергать собранные с его помощью данные практически любой, даже очень сложной, обработке. Такой подход позволяет ограничиться минимальными затратами. Он соответствует современным задачам производственного технологического контроля и испытаний, непрерывного эксплуатационного мониторинга, потребностям научных лабораторий в проведении длительных экспериментальных исследований, нуждам образовательного процесса по техническим специальностям.

# Глава 1. ПРИНЦИПЫ ПОСТРОЕНИЯ СИСТЕМ КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА

## 1.1. ЗАДАЧИ АВТОМАТИЗАЦИИ КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА РАДИОАППАРАТУРЫ

### Жизненные циклы радиоэлектронной аппаратуры (РЭА)

Все радиоэлектронные средства проходят ряд типовых стадий жизненного цикла (ЖЦ). Проведем качественный анализ задач систем контроля и испытаний на основных стадиях ЖЦ РЭА.

1. *Проектирование РЭА.* Осуществляются исследования и обработка идей, формирование уровня качества, разработка проектной документации, изготовление образцов РЭА, заводские и государственные испытания опытного образца, разработка рабочей документации. Этапы формирования уровня качества и испытания опытного образца базируются на достижениях научно-технического прогресса не только в области построения РЭА, но и в значительной мере в области испытаний. Новое качество во многих случаях может быть достигнуто за счет компьютеризации, так как допускает использование более точных алгоритмов и методик косвенных измерений с компенсацией наводок, шумов и погрешностей приборов.

2. *Изготовление РЭА.* Включает этапы технологической подготовки производства, изготовления и испытания качества РЭА. Последний этап предусматривает проведение приемо-сдаточных, климатических, вибрационных и других испытаний, а также многосуточную тренировку РЭА с последующей проверкой основных технических характеристик. В ряде случаев необходима проверка функционирования РЭА в условиях, приближенных к реальным, с использованием специальных имитаторов канала распространения радиоволн. Компьютеризация технологических испытаний, тренировки и проверки функционирования может значительно повысить качество РЭА и существенно снизить затраты производства.

3. *Эксплуатация РЭА.* Это основная стадия целевого использования изделия в соответствии с назначением. Можно выделить этапы технического обслуживания (ТО), ремонта и восстановления РЭА после отказа. Качество РЭА на этапе эксплуатации реализуется через совокупность показателей, которые могут подразделяться на единичные и комплексные.

Единичными показателями качества на этапе эксплуатации служат: наработка на отказ, ресурс РЭА и др. Комплексные показатели (КП) определяют совместно несколько простых свойств. Примером КП может быть коэффициент технического использования:

$$K_{\text{ТИ}} = T_0 / (T_0 + \tau_B + \tau_{\text{ТО}}),$$

где  $T_0$  – средняя наработка на отказ;  $\tau_B$  – среднее время восстановления;  $\tau_{\text{ТО}}$  – средняя продолжительность технического обслуживания (ТО).

Величина  $K_{\text{ТИ}}$  зависит от безотказности, восстанавливаемости и трудоемкости технического обслуживания, в том числе от времени испытаний. Чем меньше времени затрачивается на ТО, тем выше  $K_{\text{ТИ}}$ . Таким образом, на стадии эксплуатации важно сокращать продолжительность ТО, чему будет способствовать компьютеризация оценки качества. На стадии эксплуатации РЭА расходуется заложенный технический ресурс. Для компенсации воздействия деградиционных процессов ведутся работы по ТО и профилактическому ремонту РЭА, а в случае возникновения отказа – по восстановлению работоспособного состояния. В этом периоде осуществляется периодическое управление параметрами РЭА (поддержание в заданных пределах).

### **Параметры радиосигналов и радиотехнических систем (РТС)**

В общем случае радиосигнал можно представить в виде гармонического колебания:

$$s(t) = A(t) \cos[2\pi f_n(t)t + \varphi(t)],$$

где  $A$ ,  $f_n$  и  $\varphi$  обозначают соответственно амплитуду, несущую частоту и фазу радиосигнала.

Аналоговая или цифровая кодированная информация передается путем изменения значений амплитуды, частоты или фазы радиосигнала, а в некоторых случаях – комбинированным изменением этих параметров. Изменение параметров несущего сигнала в соответствии с изменением значений модулирующего называется модуляцией или манипуляцией в случае передачи дискретных сообщений. При амплитудной модуляции (АМ) и манипуляции (АМП) для передачи информации непрерывно или дискретно изменяется амплитуда несущей.

При частотной модуляции (ЧМ) и манипуляции (ЧМП) для передачи информации используется управление частотой. Для ЧМП-сигнала амплитуда и начальная фаза сигнала постоянные, а частота принимает дискрет-

ные значения. При фазовой модуляции (ФМ) и манипуляции (ФМП) информация передается путем управления фазой. Для ФМП-сигналов последовательность передаваемых чисел кодируется  $M$ -значениями фазы. Существует большое количество ФМП-сигналов: двухпозиционные, четырехпозиционные, а также сигналы с большим количеством позиций (8, 16, 32). Например, для ФМП-сигнала с четырьмя позициями используется фазовая манипуляция со сдвигом на  $45^\circ$ .

Вид манипуляции, в котором одновременно используются возможности АМ и ФМ, получил название квадратурной амплитудной манипуляции (КАМ). В общем случае КАМ представляет собой сочетание многоуровневой АМП и многоуровневой ФМП.

Совокупность характеристик, определяющих качество РТС, может быть разделена на группы: технических, эксплуатационных, экономических, пользовательских и др. Наиболее важными являются:

- рабочая (минимальная и максимальная) дальность действия;
- пропускная способность и число каналов;
- типы оконечных устройств (пользователей);
- безотказность и эффективность;
- масса, габариты и потребляемая мощность.

Технические характеристики, в свою очередь, могут быть разделены на подгруппы параметров сигналов, исследуемых в режиме пассивного измерения, и параметров устройств, исследуемых в режиме активного воздействия на объект испытаний. Состав и количество технических характеристик зависит от функционального назначения РТС. К техническим параметрам относят следующие основные группы: частотные, модуляционные, энергетические, согласования, статистические, паразитные, шумовые и др.

Вместе с тем современная СКИМ должна измерять не только параметры объекта, но также ряд вспомогательных параметров, характеризующих условия испытаний, среди которых наиболее важными являются параметры радиоканала и энергопитания. При этом важной функцией СКИМ также является управление внешними воздействиями: климатическими факторами, параметрами помех, вибраций и др.

Типовые методики испытаний РТС, например радиостанций, предполагают использование ограниченного числа автономных приборов: ваттметра, генераторов низкой и высокой частоты, частотомера, универсального и селективного вольтметров, измерителей модуляции и нелинейных искажений – всего от 6 до 10 приборов.



## Радиоканал и его характеристики

Особенностью любого радиоканала передачи информации является наличие помех естественного и искусственного происхождения. В зависимости от природы возникновения различают помехи сосредоточенные по времени, сосредоточенные по частоте и флуктуационные. Источниками естественных помех являются атмосферные явления. Заметное влияние на распространение радиосигнала может оказать поглощение радиоволн зданиями и естественными препятствиями, доплеровский сдвиг частоты при радиосвязи с подвижными объектами, рефракция, изменение поляризации, помехи от теплового излучения тропосферы. Кроме внешних причин возникновения помех существуют и внутренние, к которым относятся собственные шумы приемника, шумы квантования, наводки различного происхождения.

В самом общем случае с учетом воздействия помех сигнал можно представить в виде

$$s_n(t) = F[s(t), n(t)],$$

где  $s(t)$  – исходный сигнал,  $n(t)$  – случайный процесс, характеризующий помеху.

Для анализа и практической реализации приемных трактов используются упрощенные математические модели, описывающие принимаемую смесь сигнала и помехи. Если такую смесь представить в виде  $s_n(t) = s(t) + n(t)$ , то помеха называется аддитивной. Наиболее простой моделью канала связи является модель с аддитивным гауссовским шумом. В этом случае значения  $s$  и  $n$  предполагаются статистически независимыми, а  $n$  является гауссовской случайной величиной с нулевым средним значением и дисперсией  $\sigma^2$ . Если  $s_n(t) = s(t)n(t)$ , то помеха является мультипликативной. Если в канале присутствуют аддитивные и мультипликативные помехи, тогда

$$s_n(t) = n_m(t)s(t) + n_a(t),$$

где  $n_a(t)$  и  $n_m(t)$  – соответственно аддитивная и мультипликативная составляющие помехи.

Изменения во времени свойств среды распространения и многолучевость прихода радиоволн в точку приема приводят к появлению так называемых замираний, при которых возникают случайные изменения уровня и фазы сигналов на входе приемных устройств. Процесс замираний сигналов

характеризуется глубиной и скоростью [7]. Под скоростью замираний понимают средний промежуток времени между двумя последовательными минимумами или максимумами амплитуды сигнала. По скорости замираний разделяют на быстрые, когда амплитуда сигнала заметно меняется в течение длительности элементарной посылки, и медленные, когда амплитуды двух соседних посылок практически постоянны. Например, в большинстве коротковолновых каналов, присутствуют медленные замирания со средним периодом от 0,1 до 2 с. В многолучевых каналах наименьший период замираний порядка 0,002 с.

Для канала распространения радиоволн с замираниями сигнала обычно используется модель со случайными амплитудой и фазой. При наличии аддитивного шума сигнал можно представить в виде

$$s_n(t) = as(t)\cos(2\pi f_n t + p) + n(t),$$

где  $s(t)$  – низкочастотный модулирующий сигнал,  $a$  и  $p$  – случайные величины с соответствующими плотностями вероятности.

### **Задачи автоматизации испытаний РТС**

Задачи автоматизации испытаний РТС должны решаться на основе комплексного подхода. Методики испытаний и аппаратно-программные средства, включая модули сопряжения с внешними измерительными комплексами, необходимо создавать на стадии разработки РТС, внутренняя структура которых должна отвечать требованиям автоматизированного контроля. Особую важность приобретает снижение затрат и времени разработки системы испытаний, что возможно за счет использования унифицированных модулей, которые можно совмещать друг с другом, сокращая время конфигурирования СКИМ. Все существующие СКИМ можно разделить на два больших класса:

- встроенные (интегрированные) – для оперативного контроля функционирования радиосистем;
- внешние (автономные) – для технологического, периодического и непрерывного контроля состояния радиосистем и комплексов.

Современная автономная СКИМ предназначена для решения широкого круга пользовательских задач, поэтому она должна быть:

- универсальной, позволяющей проводить испытания на разных стадиях жизненного цикла РТС, в том числе приемо-сдаточные и климатические испытания, а также тренировку и проверку функционирования;

- переконфигурируемой, позволяющей использовать различные аппаратно-программные средства в оптимальном сочетании;
- гибкой, позволяющей вносить изменения границ, допусков, диапазонов, кодов управления объектом и т.д.;
- достаточной, обеспечивающей контроль и измерение основной структуры параметров РТС, а также параметров внешних условий, включая контроль параметров питания;
- модульной, позволяющей выбирать требуемые модули алгоритмов измерения и приборов из библиотек аппаратно-программных средств;
- развиваемой, позволяющей вводить в систему испытаний новые алгоритмы и аппаратно-программные модули виртуальных и автономных приборов;
- простой, позволяющей модернизировать программу испытаний в диалоговом режиме работы с ПК, используя удобный интерфейс пользователя;
- открытой, позволяющей передавать информацию внешним пользователям по стандартным шинам ПК (RS-485, RS-232 и др.);
- доступной, комплектуемой наиболее распространенными аппаратно-программными средствами;
- обрабатывающей, позволяющей не только накапливать базу данных, но также выполнять статистическую обработку данных по всем параметрам;
- управляющей, позволяющей передавать команды управления в объект испытаний, периферийные и вспомогательные модули системы, в том числе в термошкафы, холодильные камеры, вибростенды, регуляторы влажности и др.

У специалиста в конкретной прикладной области могут возникнуть трудности во взаимодействии с автоматизированной системой, поскольку он не обладает достаточной квалификацией в сфере вычислительной техники и программирования. Важно предоставить такому пользователю максимальные удобства для работы, когда управление системой и представление результатов экспериментов осуществляется в привычных для него формах.

## 1.2. АРХИТЕКТУРА СИСТЕМ КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА

### Международная стандартизация

Компьютерные технологии автоматизации испытаний основаны на использовании унифицированных программно-аппаратных средств и стандартных интерфейсов. В настоящее время в эксплуатации находятся тысячи измерительных приборов и систем, управляемых ПК, аппаратное и программное обеспечение для которых поставляют сотни фирм [3].

Основой современного этапа развития СКИМ является алгоритмизация измерений, совершенствование элементной базы, использование модульных структур, в основе которых лежат международные стандарты. Программируемые приборы, выпущенные в шестидесятые годы, использовали большое количество патентованных интерфейсов и коммутационных протоколов. На базе внутреннего стандарта фирмы Hewlett-Packard (HP-IB) в 1975 году Институт инженеров по электротехнике и радиоэлектронике (Institute of Electrical and Electronics Engineers) одобрил интерфейс IEEE-488, который допускал полную свободу изготовителей приборов по использованию различных форматов программирования и передачи данных. В 1980-х годах с целью повышения совместимости приборов началась работа по созданию дополнительных стандартов, основанных на IEEE-488.

В 1987 году стандарт IEEE-488 был переименован в IEEE-488.1 и утвержден новый стандарт IEEE-488.2 "Коды, форматы, протоколы и общие команды" для использования совместно с IEEE-488.1. Новый стандарт определил структуру взаимодействия и форматы представления информации, хотя основные команды по управлению приборами не были стандартизованы. Даже в изделиях одной компании употреблялись различные варианты команд. Таким образом, одинаковые по типу приборы могли подходить для стандарта IEEE-488.2, но в то же время иметь различные команды управления.

В 1985 году фирма Hewlett-Packard приступила к разработке стандартного языка программирования. Он был назван языком для систем измерения и тестирования (TMSL – Test and Measurement System Language). Язык основывался на IEEE-488.2 и определял универсальный набор команд управления для всех типов приборов. К 1989 году TMSL включал в себя примерно 850 стандартизованных команд.

Несомненная эффективность использования TMSL способствовала образованию консорциума по стандартным командам для программируемых приборов (SCPI – Standard Commands for Programmable Instruments). В апреле 1990 года консорциум без каких-либо существенных изменений предложил HP TMSL как базис для SCPI и добавил к нему модифицированную версию общего формата для аналоговой информации компании Tektronix. Основными достижениями SCPI являются: снижение времени разработки программных средств и обеспечение взаимозаменяемости приборов. С появлением стандартных команд управления облегчается задача по созданию программного обеспечения, отпадает необходимость в учете синтаксических отличий команд различных приборов.

Взаимозаменяемость приборов разных поколений называется вертикальной совместимостью, а взаимозаменяемость между приборами разных видов и классов – горизонтальной совместимостью. Например, мультиметр, использующий SCPI и выпущенный в 2004 году, будет пользоваться все теми же командами, что и мультиметр 1992 года. SCPI дает гарантию совместимости изделий, даже если прибор 2004 года будет иметь какие-то дополнительные возможности и команды. Управляющие команды в 2004 году будут надстраиваться над командами 1992 года.

Горизонтальная совместимость требует, чтобы любая команда, например "Измерить частоту", была одинаковой как для частотомера, так и для анализатора спектра или любого другого прибора, измеряющего частоту. Так как все команды одинаковы, то программистам необходимо только раз их освоить и использовать далее для всех типов приборов. При замене прибора на новый он может интегрироваться со всей системой без изменения управляющих кодов. Так как постоянно появляются новые приборы и новые задачи, SCPI способен адаптироваться к этим нововведениям. Имеются специальные правила по добавлению в стандарт новых команд и возможностей. По мере появления новых разработок они предлагаются на рассмотрение консорциума по SCPI.

Основными членами консорциума являются: Hewlett-Packard, Tektronix, Philips, Bruel & Kjaer, Fluke, Keithley, Racal-Dana, Rohde & Schwarz, National Instruments, Wavetek. Различная аппаратура с SCPI доступна уже сегодня. Например, среди изделий Hewlett-Packard SCPI используют модули VXIbus, приборы HP-IB и модульные системы измерения MMS (Modular Measurement System). Все современное производимое оборудование является SCPI-совместимым.

## Варианты архитектуры СКИМ

По архитектуре СКИМ можно разделить на следующие классы:

1. *Приборно-модульные системы (ПМС)* строятся на основе серийных приборов с использованием интерфейса IEEE-488 (ГОСТ 26.003-80) [1]. Вариант компоновки ПМС, содержащей 9 автономных приборов, приведен на рис. 1.1.

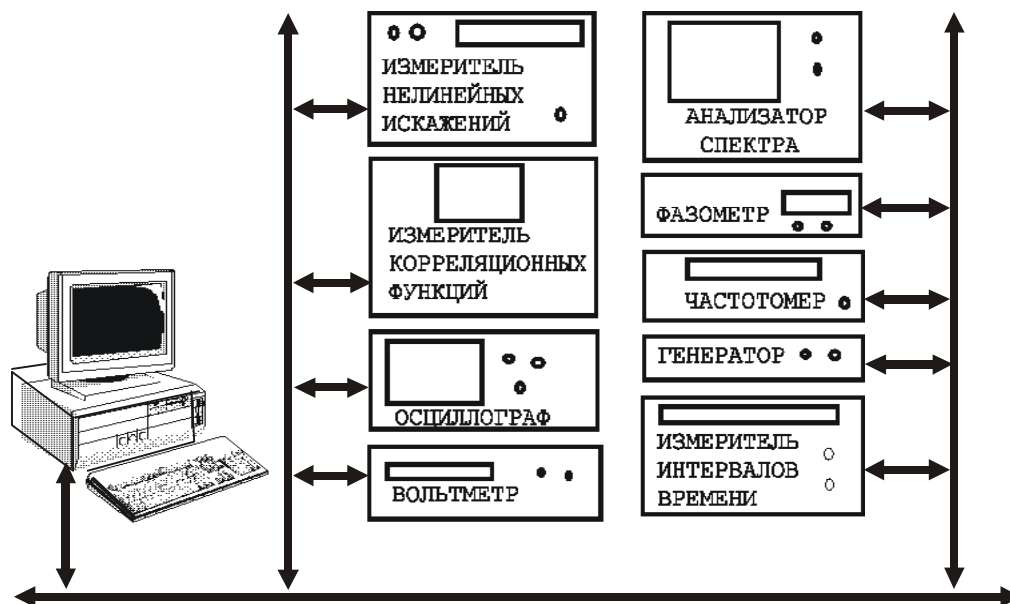


Рис. 1.1. Приборно-модульная система испытаний

Системы гибкие, управляемые ПК, перепрограммируемые, относительно быстро создаваемые. Их преимуществом является наличие метрологического обеспечения для каждого модуля-прибора, который может использоваться автономно. Интерфейс IEEE-488 [8] ориентирован на локальное подключение к ПК программируемых приборов и периферийных устройств через кабельные сегменты. Интерфейс нормируется ГОСТ 26.003-80 и именуется в России как канал общего пользования (КОП). Без дополнительных аппаратных средств к одному ПК можно подсоединить до 15 устройств с общей длиной магистрали до 20 м. Применение шинных расширителей позволяет увеличить число приборов-модулей при однобайтовой адресации до 31, а при двухбайтовой – до 961. Специальные удлинители позволяют увеличить магистраль до 300 м, а при использовании модемов – до километров.

Вместе с тем системы с интерфейсом IEEE-488 обладают аппаратной и функциональной избыточностью, они громоздкие, энергоемкие, имеют эксплуатационные возможности, ограниченные характеристиками модулей.

2. *Крейтовые модульные системы (КМС)* строятся на основе серийных модулей, вставляемых в общий корпус – крейт. Широко используемой в крейтовых системах является шина VXI, которая сегодня опередила по всем основным характеристикам устаревшую шину CAMAC. Шина VXI является основой для нового поколения автоматического тестового оборудования одноплатных приборов. В 1987 году ведущие производители средств контроля и измерения (Tektronix, Colorado Data Systems, Hewlett-Packard, Racal-Dana Instruments и Wavetek) основали консорциум VXI. Открытость архитектуры, малые габариты, высокая производительность оборудования, взаимозаменяемость и совместимость модулей разных производителей характеризуют VXI. Сегодня получают также развитие крейтовые системы с шинами PXI и SCXI. Они более дешевые при использовании в технологических целях. Вариант компоновки крейтовой системы приведен на рис. 1.2.

Крейтовые системы практически лишены избыточности, они имеют малые габариты, высокую производительность и надежность, однако на сегодняшний день стоимость таких систем неоправданно велика.

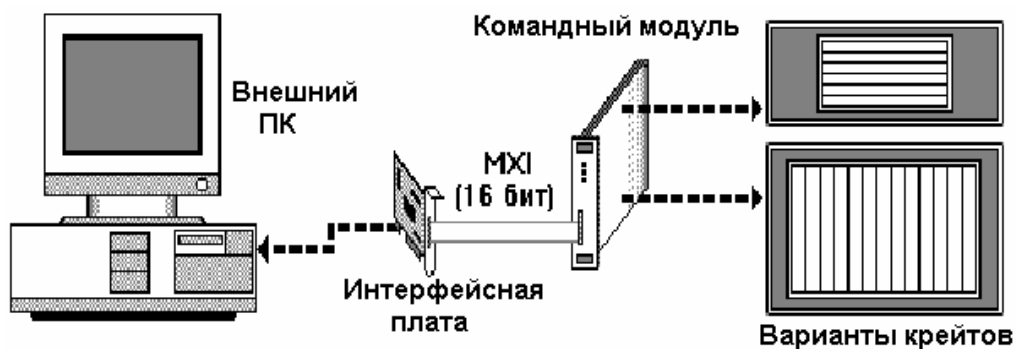
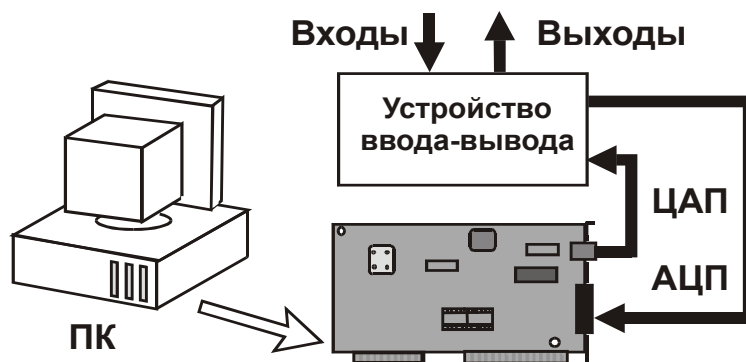


Рис. 1.2. Крейтовая модульная система с внешним ПК

3. *Системы компьютерных приборов (СКП)* [2] строятся на базе ПК. Благодаря современным аппаратно-программным средствам и возможностям графического представления данных компьютер легко преобразуется в виртуальный прибор (ВП) и испытательный комплекс. Для этого достаточно оснастить его быстродействующими эффективно функционирующими в реальном масштабе времени преобразователями и формирова-телями, как показано на рис. 1.3.

ВП представляет собой комбинацию компьютера, универсальных аппаратных средств ввода/вывода сигналов и специализированного про-

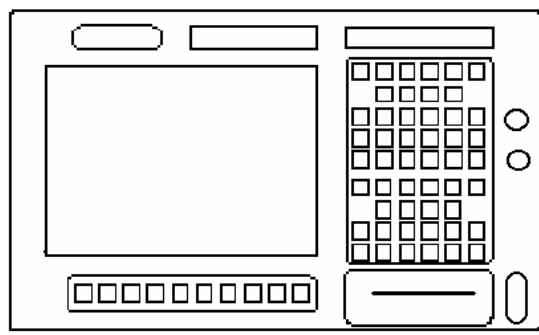
граммного обеспечения (ПО). Использование серийных ПК закладывает основу относительно низкой стоимости виртуальных систем, которые имеют к тому же малые габариты. Аппаратная и программная база СКП в настоящее время очень интенсивно развивается. полоса рабочих частот и быстродействие виртуальных приборов уже приблизились к параметрам автономных приборов.



*Рис. 1.3. Компьютерные приборы на базе плат расширения*

Главной проблемой и формальным ограничением широкого применения ВП может стать законодательная база в виде действующих стандартов, в которых не предусмотрено применение перепрограммируемых виртуальных приборов.

4. Комплексы комбинированных приборов (ККП) сегодня достаточно широко распространены. Их также называют сервисными мониторами. Они представляют собой объединенные конструктивно программируемые приборы, предназначенные для автоматизированного измерения параметров заданного класса РЭА, например средств связи. Внешний вид ККП представлен на рис. 1.4.



*Рис. 1.4. Комплекс комбинированных приборов*



ККП управляется встроенным контроллером и при необходимости может быть объединен с внешним ПК. Портативные системы приборов, такие как AGILENT 8920A/B, предназначены для комплексного тестирования и измерений как аналоговых, так и цифровых средств связи. ККП позволяют производить измерения основных параметров (напряжения, частоты, мощности) и производных характеристик (стабильности, глубины модуляции и др.). Достаточные для многих сервисных задач характеристики ККП достигаются благодаря применению современных процессорных средств и развитого программного обеспечения.

Основными недостатками ККП являются ограниченные метрологические возможности встроенных приборов. По сравнению с автономными и крейтовыми системами модули ККП более простые и дешевые. Реконфигурация и наращивание возможностей ККП затруднены.

5. *Индивидуальные специализированные системы (ИСС)* предназначены для решения конкретных задач. Они применяются в качестве систем встроенного контроля мобильных технических объектов, обладающих ограниченным энергетическим ресурсом и малым объемом. ИСС имеют жесткую программу контроля ограниченного круга параметров в заданном диапазоне значений и содержат специализированные, как правило простые, преобразователи и схемы сравнения.

Структурная схема одного из многочисленных вариантов ИСС приведена на рис. 1.5. С целью сокращения затрат на разработку в структуру ИСС могут включаться универсальные модули и приборы. Программа контроля размещается в памяти управляюще-вычислительного модуля. Специализированные преобразователи позволяют сократить число унифицированных измерительных приборов. Индикация может быть допусковая, цифровая, аварийная. Для контроля параметров с индикацией "В допуске" – "Вне допуски", "Годен" – "Не годен" используется эталонная схема и устройства сравнения.

ИСС оптимизированы по всем основным характеристикам, поэтому они конструктивно простые, надежные, но имеют ограниченные возможности. Специализированные СКМ не позволяют гибко изменять и наращивать возможности, хотя нередко управляются с помощью серийного ПК.

Разработка и изготовление ИСС обходится дорого, требует значительных затрат времени. Оправдано это лишь в условиях серийного или

массового производства изделий, а также для обслуживания заданной серии сложной электронной аппаратуры или для построения мобильных систем со специальными эксплуатационными возможностями по климатическим, ударным и вибрационным нагрузкам, по радиационной и электромагнитной защищенности.

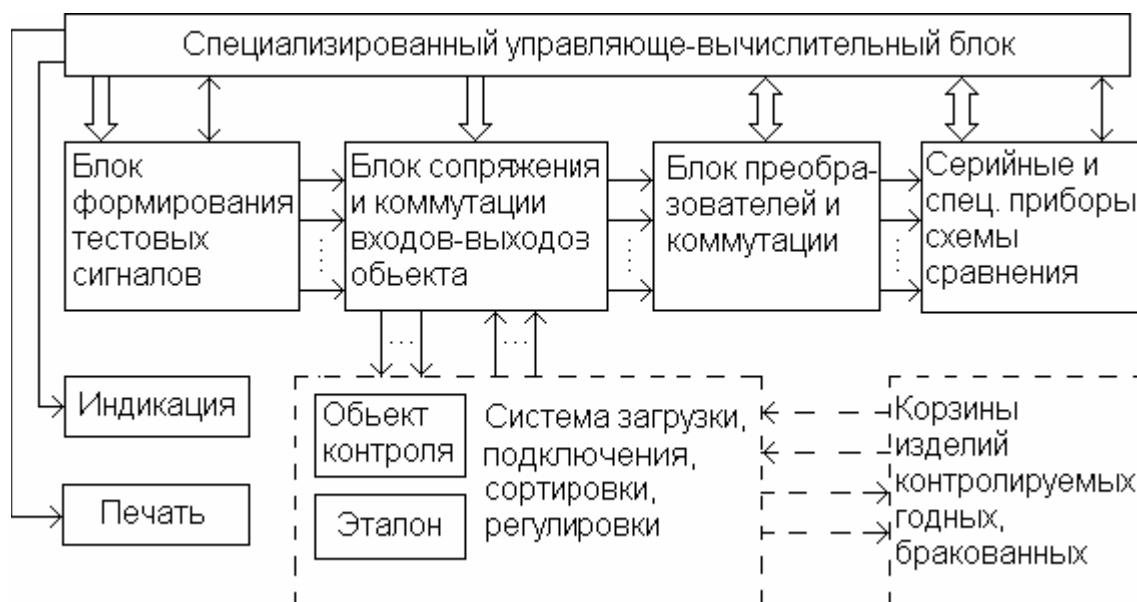


Рис. 1.5. Вариант структурной схемы специализированной СКИМ

6. *Комбинированные системы (КС)* представляют собой оптимальное сочетание рассмотренных выше вариантов построения СКИМ, позволяя решать задачи испытаний с минимальными затратами и с учетом наиболее существенных для пользователя факторов, таких как:

- сроки разработки и внедрения;
- стоимость разработки и покупных изделий;
- наличие комплектующих, которые не нужно покупать или которые можно изготовить с использованием имеющейся базы;
- требования к эксплуатационным и техническим характеристикам системы, включая ремонтпригодность и надежность;
- перспективы развития системы;
- наличие персонала для эксплуатации системы.

### **Оптимизация СКИМ**

Создание СКИМ с заданными метрологическими параметрами, а также целесообразность применения тех или иных методов и аппаратуры не-

разрывно связаны с необходимостью выбора критерия эффективности, с помощью которого можно было бы сравнить различные варианты. Задачу оптимизации можно рассматривать как управление качеством, которое определяется как степень соответствия СКИМ своему назначению. В вероятностных терминах СКИМ обеспечивает возможность достижения погрешности каждого испытания  $\Delta_i$  с заданной вероятностью  $P_i$ :

$$P(r \leq \Delta) = 1 - \varepsilon \text{ или } P(r > \Delta) \leq \varepsilon,$$

где  $\Delta$  и  $\varepsilon$  – заданные малые величины;  $r$  – фактическое расстояние между целью (истинным значением параметра) и конечным состоянием (результатом измерения). Значения  $\Delta$ ,  $\varepsilon, \dots$ , определяющие качество СКИМ, считаются фиксированными константами, играющими роль показателей качества. Условия эксплуатации и свойства используемой элементной базы предполагаются заданными. При этом система должна удовлетворять ряду дополнительных требований по габаритам, стоимости и т. п. Задача оптимизации связана с нахождением предельных возможностей СКИМ. Критерием качества может быть погрешность  $r(y_n, y_p)$ :

$$u_{\text{opt}}^* = \arg \min_{u \in U} r(y_n, y_p) = \arg \min_{u \in U} r(A_n, A_p),$$

где  $y_n$  и  $y_p$  – идеальный и реальный результаты измерения;  $A_n$  и  $A_p$  – идеальный и реальный операторы функционирования системы.

Соответственно показателям  $\varepsilon$ ,  $\Delta$ ,  $T$ ,  $C$  можно записать следующие критерии оптимизации:  $P$  – критерий вероятности выполнения задачи;  $\Delta$  – критерий точности;  $T$  – критерий долговечности;  $C$  – критерий стоимости. Математически эти критерии могут быть представлены в виде систем уравнений:

$$\left. \begin{aligned} u_P^* &= \arg \min_{u \in U} P(u) \\ \Delta, T, C &= \text{const} \end{aligned} \right\}; \quad \left. \begin{aligned} u_\Delta^* &= \arg \min_{u \in U} \Delta(u) \\ \varepsilon, T, C &= \text{const} \end{aligned} \right\};$$

$$\left. \begin{aligned} u_T^* &= \arg \max_{u \in U} T(u) \\ \varepsilon, \Delta, C &= \text{const} \end{aligned} \right\}; \quad \left. \begin{aligned} u_C^* &= \arg \min_{u \in U} C(u) \\ \varepsilon, \Delta, T &= \text{const} \end{aligned} \right\}.$$

Можно использовать обобщенные критерии вида:

$$\prod_{i=1}^m \varphi_i(u) / \prod_{i=m+1}^k \varphi_i(u) = \max,$$

где в числителе стоят функционалы, которые следует максимизировать, а в знаменателе – те, которые следует минимизировать. Поскольку функционалы  $P$ ,  $\Delta$ ,  $T$ ,  $S$  являются функциями большого числа переменных, можно реализовать процедуру «декомпозиции» системы, в результате чего сложная задача отыскания экстремума одной функции с очень большим числом переменных сводится к совокупности более простых, поддающихся решению задач отыскания экстремумов многих функций, каждая из которых зависит от малого числа переменных. Этап проектирования СКИМ включает в себя решение задачи комплексирования, которая состоит в том, чтобы определить показатели для отдельных модулей с таким расчетом, чтобы вся система удовлетворяла установленным требованиям.

Оптимизация возможна по стоимости, быстродействию, надежности, габаритам и т.д. При выборе оптимального варианта конфигурации необходимо уже на этапе выбора принципов построения предвидеть развитие проектируемой системы. Проблемы автоматизации испытаний РТС могут быть решены на базе комплексного подхода, в основе которого лежит объединение модулей, реализующих алгоритмические измерения [7] и цифровую обработку данных. Наиболее оптимальной с точки зрения стоимости является комбинация приборно-модульной и виртуальной систем. При этом можно заменить более дорогие автономные приборы их виртуальными аналогами, реализуемыми с помощью недорогих плат аналого-цифрового и цифроаналогового преобразования (АЦП и ЦАП). Комбинированные системы обладают более гибкими возможностями.

Проведенные исследования позволяют определить области наиболее рационального использования СКИМ: при измерении и регистрации параметров и функционалов РТС; при прогонах и испытаниях (климатических, механических), когда необходимо периодически производить измерения и регистрацию результатов; при приемо-сдаточных испытаниях и проверке соответствия паспортных данных и действительных значений параметров и функционалов РТС.

При выборе оптимального варианта конфигурации прежде всего необходимо учитывать назначение разрабатываемой системы. Часто в процессе эксплуатации возникает необходимость модернизировать и наращивать возможности существующих СКИМ, поэтому уже на этапе выбора принципов построения необходимо предвидеть развитие проектируемой системы. Комбинированные системы обладают значительно более гибки-

ми возможностями, поскольку базовая система для удовлетворения новым техническим требованиям может быть легко расширена за счет подключения новых аппаратно-программных модулей.

### **1.3. ПРОГРАММНЫЕ СРЕДСТВА СИСТЕМ КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА**

#### **Задачи и структура программного обеспечения**

Сегодня разработаны программные средства для различных классов приложений от распределенного сбора данных до высокоскоростного анализа информации. Основные требования: высокая скорость сбора данных, наличие опций графики, развитые алгоритмы обработки, совместимость с другими программами, большое число каналов.

Функции программного обеспечения:

1. Реализация удобного пользовательского интерфейса.
2. Формирование алгоритмов измерения и управления.
3. Сбор и высокоскоростное преобразование данных.
4. Представление и сохранение данных.

*Интерфейс оператора* – это графическая оболочка – интерактивная среда программирования и работы, позволяющая с помощью клавиатуры и "мышки" создавать рабочее программное обеспечение, удобное для дальнейшего использования.

*Алгоритмы измерения и управления* – это правила, по которым система автоматически начинает и останавливает процесс или изменяет его параметры без человеческого вмешательства. Методы измерения и управления значительно различаются в зависимости от задач, решаемых пользователями.

*Сбор и преобразование данных* определяется аппаратными возможностями устройств ввода-вывода и аналитическими возможностями компьютера. Манипулирование всеми типами данных может быть выполнено в режиме реального времени или анализа записанных данных.

*Представление и сохранение данных* предполагает использование графических возможностей ПК и его прикладных программ, а также запись с целью сохранения и дальнейшего анализа на диск результатов теста или эксперимента, а также параметров управления.

Анализ материалов фирм-производителей ПО позволяет выявить следующие функциональные модули, характерные для программного обеспечения СКИМ:

- модули графических элементов интерактивного управления (панелей, ручек управления, полос прокрутки и т.д.);
- модули отображения цифровой и графической информации, а также сигнализации (формирующие, например, аварийный сигнал при выходе значения заданного параметра за допустимые пределы);
- модули специальных программ (драйверов) для обеспечения взаимодействия с внешними техническими средствами (с контроллерами стандартных интерфейсов RS-232, IEEE-488, MXI и др.);
- модули режимов и алгоритмов измерения специальных параметров (частоты опроса, диапазонов измерений, числа каналов и др.);
- модули подготовки и преобразования измерительных данных (масштабирование, взвешивание, форматирование и др.);
- модули цифровой обработки информации, обеспечивающие статистический, спектральный и корреляционный анализ, а также другие методы цифровой обработки сигналов, в том числе фильтрации;
- модули систематизации и хранения результатов измерений;
- модули обработки отказов и неисправностей, позволяющие выявить скрытые ошибки технического и программного обеспечения;
- модули построения сетей с возможностью маршрутизации и обмена результатами измерений, графической информацией и т.д.

Базовое программное обеспечение плат, обычно включающее написанные на Ассемблере драйверы, позволяет пользователю создать на языках высокого уровня собственные программы испытаний, а также дополнить пакет недостающими функциями обработки данных и построить с его помощью СКИМ. Для сокращения затрат на создание программного обеспечения, гибкой настройки системы могут успешно применяться инструментальные средства автоматизации проектирования СКИМ. Такие средства, как правило, ориентированы на конечного пользователя.

В качестве примеров систем автоматизации проектирования виртуальных приборов могут служить пакеты LabVIEW (фирма National Instruments) и HP VEE (фирма Hewlett Packard).

## **Возможности специализированных пакетов**

Рассмотрим некоторые возможности специализированных пакетов программных средств, представив их в виде перечня основных функций. Можно выделить несколько уровней работы пользователя в среде программирования:

### *Уровень 1. Пошаговый*

На этом уровне можно работать как с программируемым калькулятором, обеспечивая основную математику: арифметику, статистику, тригонометрические и другие функции. При этом могут быть реализованы расширенные возможности:

- параллельный анализ данных;
- управление с обратной связью;
- алгебраические, корреляционные, тригонометрические и статистические функции;
- одновременная индикация данных измерений и расчетных результатов.

Данные могут обрабатываться в процессе сбора или после сохранения в файле. Часто используемые функции, уравнения или подпрограммы могут быть сохранены в виде макрокоманд.

### *Уровень 2. Библиотечный*

На этом уровне оператор работает с готовыми модулями управления и обработки, которые можно редактировать. Сюда входят модули цифровой фильтрации, преобразования Фурье и др. Используя алгоритм БПФ, можно преобразовать данные области времени в данные частотной области. Некоторые возможности работы:

- вычисление преобразований Фурье, Гильберта и др. ;
- цифровая фильтрация;
- анализ энергетического спектра;
- вычисления с комплексными числами;
- представление данных в линейном и логарифмическом масштабах;
- объединение на графике одних данных с другими;
- вывод табличных данных и графиков на принтер.

### *Уровень 3. Программируемый*

На этом уровне в интерактивном режиме можно компоновать модули, подключать подпрограммы обработки, формировать систему управле-

ния. Дополнительный анализ данных может быть реализован с использованием других программных пакетов. Некоторые возможности:

- интерактивная и полностью программируемая среда;
- управление исследованиями и внешними объектами;
- автоматизация испытаний с пошаговым анализом;
- синхронизация действий по условиям и сигналам;
- текущий контроль состояния;
- поддержка обмена данными с внешними устройствами;
- редактор для отладки и тестирования;
- поддержка аналоговых и цифровых устройств вывода;
- автоматическая, редактируемая графика.

На программируемом уровне имеются расширенные возможности составления новых команд. Можно предписывать новое слово функциональной клавише. Любая процедура может быть сохранена на диске так, чтобы каждый раз при загрузке системы она включалась в персональный словарь. Программы могут быть сохранены в стандартных текстовых файлах ASCII.

### **Примерные функции модулей программных средств**

#### **ФУНКЦИИ МОДУЛЯ ОБРАБОТКИ ДАННЫХ:**

*Арифметические действия* и специальные функции: +, -, ×, /, \*\*, min, max, neg, abs, inv, sqrt, ln, exp, conj, sin, cos, tan, sec, csc, cot, sinh, cosh, tanh, sech, csch, coth, asin, acos, atan, asec, acsc, acot, asinh, acosh, atanh, asech, acsh, acoth. Все арифметические операторы работают непосредственно (без циклов) со всеми элементами из массива. Разрешаются смешанные выражения массивов и скаляров или массивов, отличающихся размерностью.

*Статистические функции:* mean (средний); variance; mode (режим); median (медиана); moments (моменты); среднее квадратичное отклонение; типовые законы распределения (гаусовское, Хи-квадрат, Стьюдента, равномерное и др.); генерация случайных чисел; сортировка; дисперсионный анализ.

*Функции обработки массива:* сравнение, транспозиция, ввод данных массива, управление форматами вывода данных массива, умножение матриц.

*Графика:* выбор разрешения, наложение графиков, окна, полярные координаты, автомасштабирование, логарифмический масштаб по любой оси, представление эмпирической кривой и др.



*Ввод-вывод:* загрузка стандартного текста или произвольных определяемых пользователем файлов, сохранение и загрузка рабочей области окна на диске, прямой ввод-вывод в ASCII.

#### ФУНКЦИИ МОДУЛЯ АНАЛИЗА:

*Математика и оценка:* полиномиальное умножение и деление, интегрирование, дифференцирование, смещение, извлечение корня.

*Улучшенная графика:* аксонометрические графики и графики с теневыми линиями.

*Векторы и матрицы:* инверсия матриц, детерминанты, разложение на множители, уменьшение порядка матрицы.

*Манипулирование данными:* фильтрация и сглаживание данных, дифференцирование и интегрирование, пиковое детектирование.

#### ФУНКЦИИ МОДУЛЯ СБОРА ДАННЫХ:

*Аналоговый ввод-вывод:* разовая или многократная выборка, генерация сигналов, программируемое усиление, управление буфером и скоростью передачи данных, включая прямой доступ в память.

*Цифровой ввод-вывод:* управление разрядностью, буферизацией и скоростью обмена, запись и чтение с адресацией вплоть до индивидуальных битов.

*Оперативная синхронизация:* запуск, синхронизация.

#### ФУНКЦИИ МОДУЛЯ УПРАВЛЕНИЯ:

*Типовые и индивидуальные протоколы:* зависимые и независимые от устройств команды, параллельный и последовательный опрос, синхронная и асинхронная работа, прямой доступ в память.

*Буферизация:* буферизация массива данных и команд.

*Оперативная синхронизация:* запуск, синхронизация.

## Глава 2. ПРИБОРНО-МОДУЛЬНЫЕ СИСТЕМЫ

### 2.1. ОСНОВНЫЕ ШИНЫ И ЛИНИИ ИНТЕРФЕЙСА IEEE-488

Особенностью интерфейса является отсутствие ограничений на конструктивную реализацию. Стандарт определяет только магистраль, которая не содержит активных элементов. Все активные элементы размещаются на платах сопряжения [1].

Стандарт предусматривает длину магистрали до 20 м с числом присоединяемых устройств до 15. В зависимости от элементной базы и конфигурации системы скорость передачи данных может достигать 1 Мбайт/с. При использовании параллельных расширителей магистрали длина магистрали может быть увеличена до 300 м. Имеются аппаратные средства увеличения числа подключаемых устройств (более 15).

В соответствии со стандартом адресные сигналы передаются с использованием мультиплексирования по тем же линиям, что и данные. При однобайтовой передаче адреса по пяти линиям максимальное количество адресов 31 (одно состояние запрещено). При двухбайтовой передаче – 961 адрес.

Максимальная скорость передачи информации 1 Мбайт/с достигается при длине кабелей до 15 м с эквивалентной нагрузкой, включенной через 1 м, при емкостной нагрузке каждого модуля менее 50 пф. Кроме того, нужны возбуждители с тремя состояниями.

В соответствии со стандартом все устройства, подключаемые к шине IEEE-488, подразделяются на четыре группы, как это показано на рис. 2.1.

КОП содержит в своей структуре три шины:

- шину данных;
- шину синхронизации (согласования передачи);
- шину общего управления.

**ШД – шина данных** – содержит 8 линий, обозначаемых ЛД0...ЛД7 (линии данных). По этим линиям осуществляется обмен информацией бит-параллельным, байт-последовательным способом (словами). Шина используется для приема/передачи данных измерений, а также данных адресных, программных, управляющих, в том числе данных состояний.

Тип информации, передаваемой по ШД, определяется предысторией и состоянием линии УП (ШУ). Период времени, в течение которого информация ШД действительна, зависит от сигнала линии СД (ШС).

Обмен информацией может быть между "говорящим" и "слушающими" приборами напрямую, а также с управляющим контроллером. Особенность шины такова, что одновременно может работать только один "говорящий" и несколько "слушающих" приборов. Контроллер организует взаимодействие модулей, его команды указывают, какой модуль "говорит", а какие "слушают". Если нужна обработка данных, то применяют ПК.

**ШС – шина согласования передачи** – содержит три линии управления передачей информации по ШД:

1. СД – сопровождение данных. Перевод в низкое состояние СД "говорящим" прибором свидетельствует о достоверности байта на ШД. Обязательным условием перевода СД в низкое состояние является высокое состояние линии ГП (приемники обработали всю информацию).

2. ГП – готов к приему. Линия обмена сигналами между приемником и передатчиком. Линия ГП управляется приемниками (адресованными на прием) или всеми устройствами, когда линия УП (ШУ) имеет низкое состояние. Устройства, не адресованные на прием, должны иметь высокое состояние ГП.

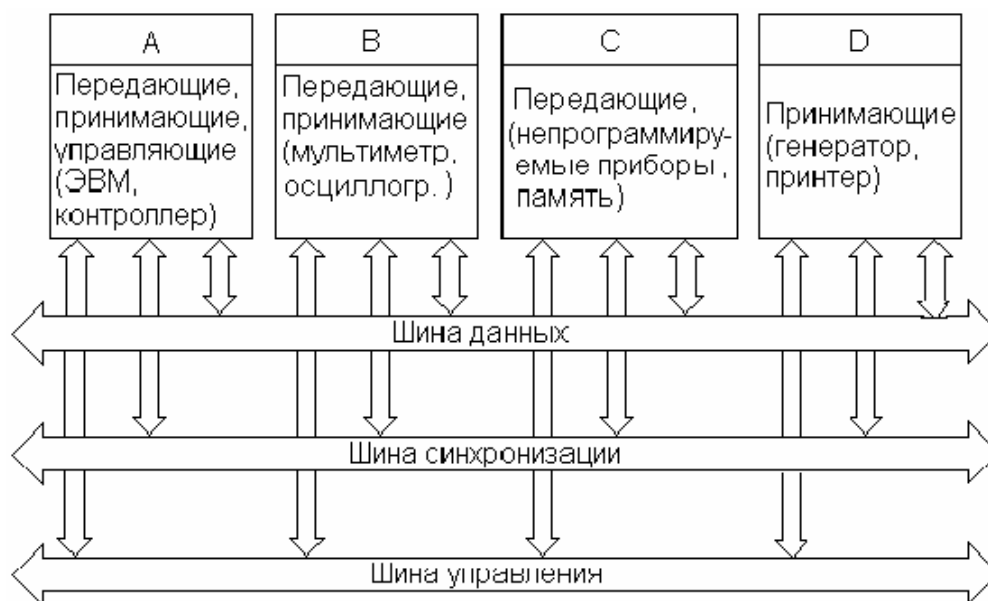


Рис. 2.1. Схема интерфейса КОП

3. ДП – данные приняты. Высокое состояние ДП указывает на конец приема информации "приемниками". Линия ДП управляет всеми устройствами, когда УП (ШУ) в низком состоянии, или теми устройствами, которые адресованы на прием, если состояние УП высокое (аналогично ГП).

**ШУ – шина управления** – содержит пять линий, по которым передаются сигналы управления между контроллером и другими устройствами:

1. УП – управление. Линией управляет контроллер. Низкий уровень УП переводит все приборы в режим ожидания (все приборы, кроме контроллера, "слушающие"). Во время низкого уровня контроллер может передавать различные сообщения устройствам. Осуществляется организация измерительных приборов для какого-либо измерения. Контроллер системы последовательно передает адреса приемников и устанавливает их в требуемый режим работы. Далее для названных приемников назначается "говорящий" прибор. При этом называется его адрес на передачу, и это устройство с переходом УП в высокое состояние становится "говорящим". Устройство остается "говорящим" до команд "не передавать" или "очистить интерфейс".

2. ОИ – очистка интерфейса. Этой линией также управляет контроллер. По низкому уровню он производит очистку интерфейсов и перевод их в исходное состояние.

3. КП – конец передачи. Эта линия используется и контроллером и другими устройствами. Линия работает в двух режимах:

- а) в режиме "передача" от "говорящего" к "слушающим" приборам;
- б) в режиме "идентификации" при параллельном опросе устройств контроллером. При параллельном опросе каждой линии ШД соответствует определенное состояние опрашиваемых устройств (так как шина данных 8-разрядная, то опрашивается до 8 устройств). Здесь не нужно адресовать устройства на передачу.

4. ЗО – запрос на обслуживание. Низкий уровень устанавливается приборами, требующими внеочередного обслуживания (нарушение синхронизации, ошибочные команды, неисправности интерфейса).

5. ДУ – разрешение дистанционного управления. Низкий уровень, подаваемый с контроллера, переключает управление приборов на дистанционное. Обычно перевод в режим ДУ осуществляется одновременно с адресацией прибора.

Все сообщения в системе с интерфейсом КОП можно разделить на местные и дистанционные. Дистанционными называются сообщения, передаваемые по КОП. Местными (внутренними) называются сообщения между устройствами и интерфейсом КОП. Они обозначаются тремя строчными буквами. Схема передачи сообщений представлена на рис. 2.2.

Все дистанционные сообщения, передаваемые по КОП, делятся на информационные (сообщения устройств) и интерфейсные (команды). В каждый момент должно передаваться только одно сообщение (устройства или интерфейсное).

Интерфейсные команды передаются при низком уровне УП. Они позволяют организовать протокол общения и создать заданную конфигурацию системы. Интерфейсные команды управляют только функциями интерфейса. Сообщения устройств передаются при высоком УП и они не влияют на состояние интерфейсных.

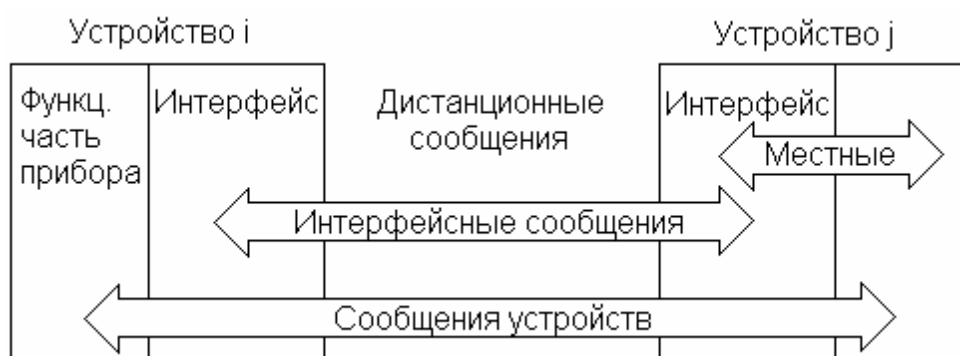


Рис. 2.2. Схема передачи сообщений

Сообщения устройств подразделяют на программные данные, основные данные и данные о состоянии.

Программные (управляющие) данные – это сообщения, используемые для подготовки устройства к исполнению основной задачи (код значения функции, код типа функции).

Основные данные – это сообщения, относящиеся непосредственно к выполнению основной задачи устройства (представление напряжения с вольтметра, частоты с выхода частотомера).

Данные о состоянии – это сообщения, характеризующие состояние устройства.

## 2.2. ИНТЕРФЕЙСНЫЕ КОМАНДЫ И ИНТЕРФЕЙСНЫЕ ФУНКЦИИ

Команды, передаваемые по шине КОП, могут быть однолинейными, передаваемыми по выделенным линиям шины управления, и многолинейными, передаваемыми по шине данных.

Стандарт IEEE-488 определяет пять групп многолинейных интерфейсных команд, передаваемых по шине данных, перечень которых представлен в табл. 2.1.

**Группа универсальных команд (ГУК)** – это команды, вызывающие действие во всех устройствах: ОПО, ЗПО, ДПР, СБУ, ЗПМ, ИДТ (идентификация).

Таблица 2.1

Обозначения и наименования многолинейных команд	ЛД7	ЛД6	ЛД5	ЛД4	ЛД3	ЛД2	ЛД1	ЛД0
1. ГАК – группа адресных команд	x	0	0	0	x	x	x	x
2. ГУК – группа универсальных команд	x	0	0	1	x	x	x	x
3. ГАП – группа адресов приемников	x	0	1	x	x	x	x	x
4. ГАИ – группа адресов источников	x	1	0	x	x	x	x	x
5. ГВК – группа вторичных команд	x	1	1	x	x	x	x	x
6. ПНМ – переход на местное управление	x	0	0	0	0	0	0	1
7. СБА – сброс адресный	x	0	0	0	0	1	0	0
8. КПР – конфигурация параллельного опроса	x	0	0	0	0	1	0	1
9. ЗАП – запуск устройства	x	0	0	0	1	0		
10. ВУП – взять управление	x	0	0	0	1	0	0	0
11. СБУ – сброс универсальный	x	0	0	1	0	1	0	1
12. ДПР – деконфигурация параллельного опроса	x	0	0	1	0	1	0	1
13. ОПО – отпирание последовательного опроса	x	0	0	1	1	0	0	0
14. ЗПО – запираание последовательного опроса	x	0	0	1	1	0	0	1
15. ЗПМ – запираание местного управления	x	0	0	1	0	0	0	1
16. ЗПР – запираание параллельного опроса	x	1	1	1	н	н	н	н
17. ОПР – отпирание параллельного опроса	x	1	1	0	с	п	п	п
18. НПМ – не принимать	x	0	1	1	1	1	1	1
19. НПД – не передавать	x	1	0	1	1	1	1	1

*Примечание.* x – линию использовать не обязательно; с – бит "считывания" (параллельный опрос возможен, если этот бит совпадает с битом устройства); п – биты, приписывающие линию данных на параллельный опрос; н – биты сообщения, на которое приемник не должен реагировать.

**Группа адресных команд (ГАК)** – это команды, вызывающие действие только в устройствах, которые адресованы: ПНМ, СБА, КПр, ЗАП, ВУП.

**Группа адресов приемников (ГАП)** – это команды, используемые для включения на прием определенных устройств. Каждому приемнику должен быть присвоен "мой адрес приемника" (МАП), младшие пять бит которого могут совпадать с аналогичными битами "мой адрес источника" (МАИ).

**Группа адресов источников (ГАИ)** – это команды для включения определенного устройства на передачу и выключения другого источника.

**Группа вторичных команд и адресов (ГВК)** – это команды, используемые в качестве второго адресного байта при адресации устройств на прием или на передачу. Сюда относятся также команды ЗПр и ОПР.

Классификация команд КОП и условия их передачи представлены на рис. 2.3.

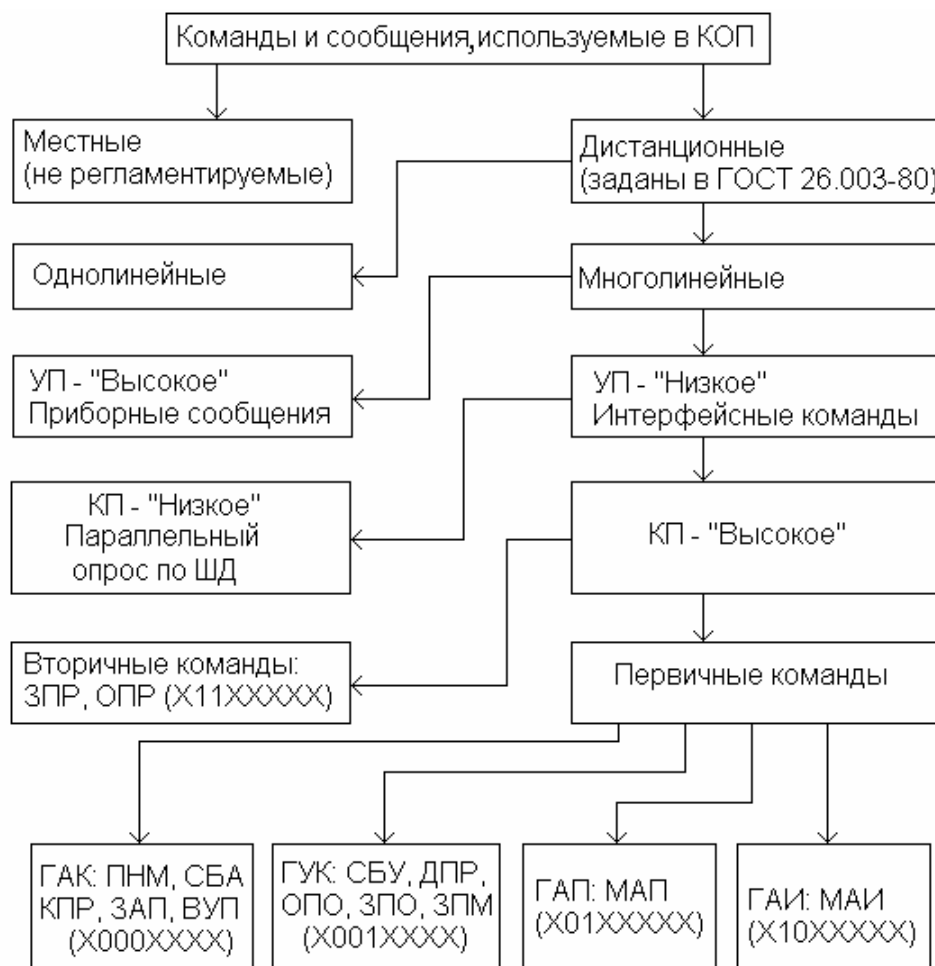


Рис. 2.3. Классификация команд КОП

## Функции устройств КОП

Каждому устройству в системе присущи три вида функций: приборные; кодирования информации и интерфейсные.

Приборные функции определяют область назначения устройства и его применения (диапазон измерений, режим работы, вид параметров измерения). Эти функции стандартом не регламентируются.

Кодирование дистанционной информации предполагает преобразование дистанционных сообщений в значения сигналов на интерфейсных линиях в соответствии с регламентированными кодами и форматами.

Интерфейсные функции представляют собой совокупность типовых операций, выполняемых при обмене данными. Интерфейсные функции могут быть реализованы аппаратно или программно. Всего их 10:

- 1) синхронизация передачи источника (СИ);
- 2) синхронизация приема (СП);
- 3) источник (И) или источник с расширением (ИР);
- 4) приемник (П) или приемник с расширением (ПР);
- 5) контроллер (К);
- 6) запрос на обслуживание (З);
- 7) дистанционное/местное управление (ДМ);
- 8) параллельный опрос (ОП);
- 9) очистить устройство (СБ);
- 10) запуск устройства (ЗП).

В конкретных приборах используются детализированные обозначения интерфейсных функций, информирующие пользователя о степени выполнения этой функции данным прибором (полностью, частично, не выполняется). Рассмотрим интерфейсные функции более подробно.

СИ – синхронизация передачи источника – обеспечивает правильную передачу устройством многолинейных сообщений асинхронно. Функция СИ контролирует начало и окончание передачи многолинейных сообщений.

СП – синхронизация приема – обеспечивает правильное получение устройством дистанционных многолинейных сообщений. Функция СП может задерживать начало или окончание передачи кодированных сообщений до тех пор, пока не будет готовности продолжения процесса приема данных. Функция СП использует (как и СИ) сообщения СД, ГП и ДП.

И (ИР) – источник (источник с расширением) – позволяет "говорящему" прибору передавать через интерфейс формируемые им данные дру-



гим приборам. Это основные данные или данные состояния. Функция И (ИР) реализуется тогда, когда осуществлена адресация "источника" на передачу. Функция И использует однобайтовый адрес, ИР – двухбайтовый.

П (ПР) – приемник (приемник с расширением) – выполняется, когда прибор адресован на прием. Функция П использует однобайтовый адрес, ПР – двухбайтовый.

З – запрос на обслуживание – позволяет прибору запрашивать у контроллера операции по обслуживанию.

ДМ – дистанционное/местное управление позволяет делать выбор между входной информацией интерфейса и органов управления с передней панели.

ОП – параллельный опрос позволяет прибору выдавать ответную информацию в контроллер без предварительной команды "Передача информации". Устройство не адресуется при этом на передачу. Линии ЛД0...ЛД7 используются при ОП для передачи битов о состоянии. Каждая линия используется одним устройством, то есть при ОП обслуживается до восьми устройств.

СБ – очистка устройства – приводит прибор в исходное состояние либо индивидуально, либо в составе группы устройств (например, все адресованные устройства могут быть группой).

ЗП – запуск прибора – позволяет начать выполнение основной работы (измерений) либо одному прибору, либо группе устройств.

К – контроллер – является функцией, позволяющей данному устройству передавать другим приборам, подключенным к интерфейсу, адреса и универсальные команды. Если в системе имеется несколько устройств, имеющих функцию К, то все они, кроме одного, должны находиться в состоянии "Холостой ход контроллера". В каждый момент действующим является один контроллер. Имеется алгоритм перехода из состояния "Холостой ход контроллера" в состояние "Действующий контроллер" системы. Функция К выполняется только при подаче сигнала УП.

### **2.3. КОДЫ, ФОРМАТЫ И ОБМЕН ДАННЫМИ ПО ШИНЕ**

Если нет специальных указаний, то передаваемый код соответствует ГОСТ 27463-87. Наименьший бит помещается на линию ЛД с наименьшим номером, то есть линиям ЛД0...ЛД7 соответствуют восьмеричный, уплот-

ненный шестнадцатеричный, шестнадцатеричный, уплотненный двоично-десятичный, двоично-десятичный и другие коды. Наиболее часто применяются 7-битный код ASCII (ЛД0 – ЛД6).

Форматы сообщений состоят из трех полей: ЗД – заголовок (буквенный); ТД – тело (цифровое); ОД – окончание (ограничитель) данных. Не обязательно, чтобы каждое сообщение имело три поля. Обычно заголовок определяет характер значения тела данных, представляемый в цифровой форме. Поле ОД применяется для создания пары ЗД – ТД.

Обычно используются один или несколько из стандартных четырех ограничителей, передаваемых одним байтом: ВК – возврат каретки; КБ – конец блока; ПС – перевод строки; КТ – конец текста.

### Примеры программных и измерительных данных

1. Программные данные для вольтметра на измерение и выдачу постоянного напряжения (F0) в диапазоне 10 В (R4) при внутреннем запуске (T1), выходном режиме (M3). Выполнение программы при получении символа P:

F0	R4	T1	M3	P
ЗД ТД	ЗД ТД	ЗД ТД	ЗД	ТД ЗД

2. Измерительные данные вольтметра:

OL DC	+12002E-03	ПС
ЗД	ТД	ОД

Здесь OL – перегрузка; DC – постоянный ток; ТД в масштабированном представлении 12.002 В на пределе 10 В (12002E-03 = 12.002).

3. Программные данные для источника питания на установку напряжения постоянного тока + 5.25 В с максимальным током 120 мА:

U	5250E-03	I	120E-03	ПС
ЗД	ТД	ЗД	ТД	ОД

Обозначения режимов, пределов приводятся в техническом описании приборов. Там же могут быть примеры программирования режимов работы.

Интерфейс КОП предполагает асинхронную передачу информации, когда источник передает информацию в темпе, определяемом скоростью приема абонентами (по приему последнего абонента). Алгоритмы работы приемника и источника, а также процесс синхронизации при передаче двух байтов данных приведены соответственно на рис. 2.4, 2.5.

Пояснения к рис. 2.4:

- 1 – готовность всех приемников;
- 2 – данные истинны и могут быть приняты;
- 3 – ДП – высокое, когда все приемники приняли;
- 4 – с этого момента данные недействительны.

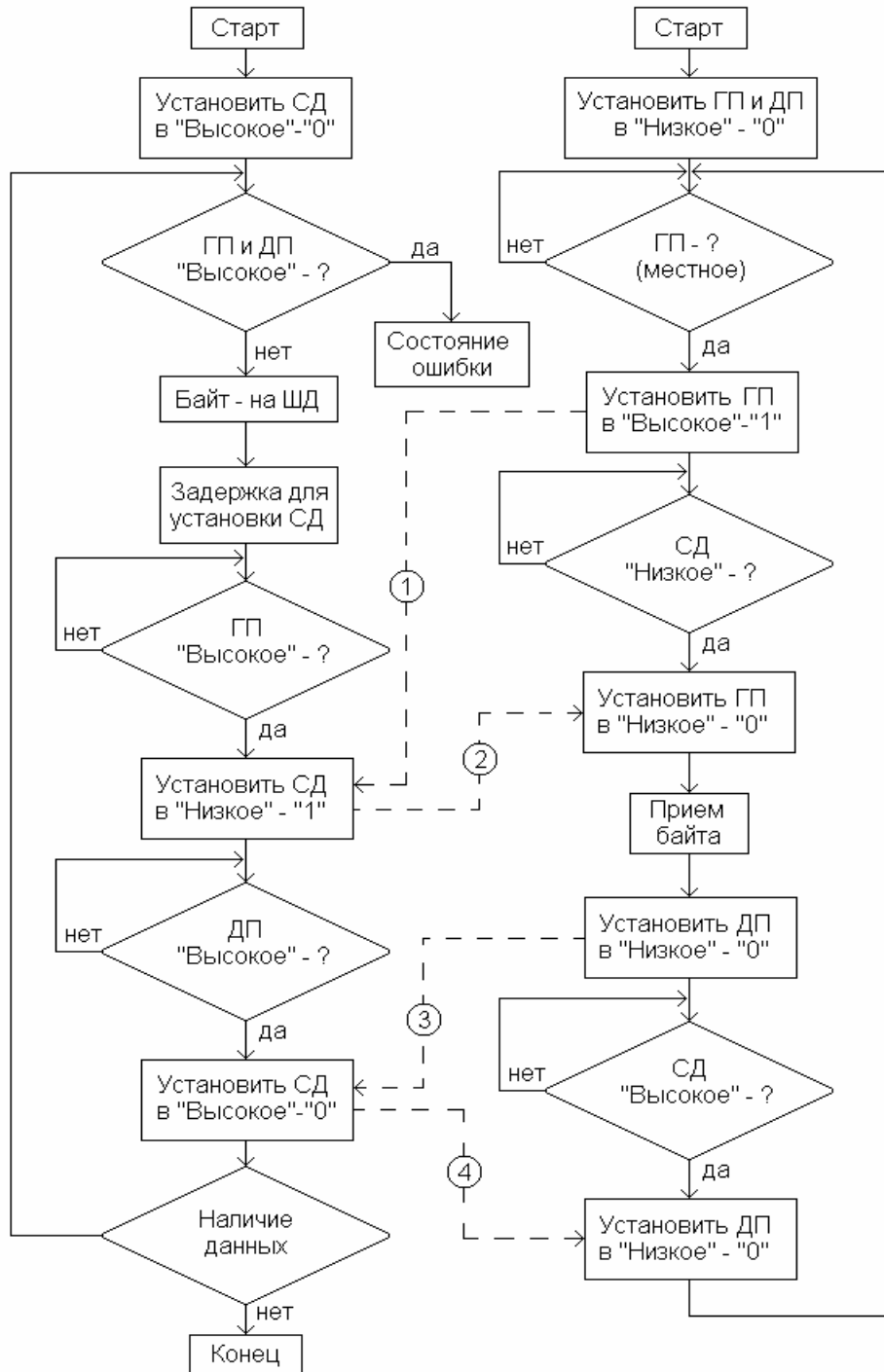


Рис. 2.4. Алгоритмы асинхронного обмена источника и приемника

Пояснения к диаграммам, представленным на рис. 2.5:

- 1 – источник установил СД в высокое состояние (данные недействительны);
- 2 – приемники установили ДП и ГП в низкое состояние;
- 3 – источник перед посылкой байта данных проверяет состояние ГП и ДП (при одном из них высоком – состояние ошибки);
- 4 – источник задерживает подтверждение истинности данных до готовности всех приемников;
- 5 – все приемники готовы к приему первого байта (ГП высокое);
- 6 – после приема сигнала готовности источник устанавливает СД в низкое состояние (истинность данных);
- 7 – после перехода СД в низкое состояние приемники переводят ГП в низкое (не готовы к приему следующего байта);
- 8 – первый приемник принял данные первого байта;

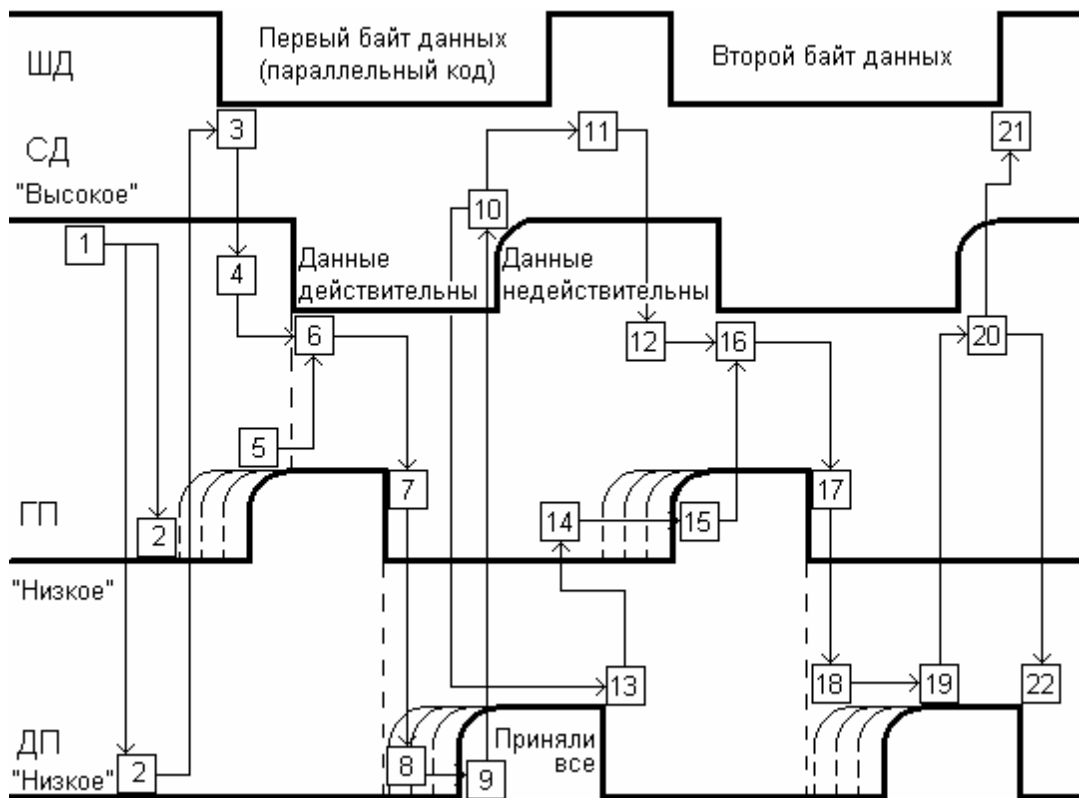


Рис. 2.5. Временная последовательность процесса синхронизации для одного источника и четырех приемников

- 9 – последний приемник принял данные и установил ДП в высокое;
- 10 – источник, приняв информацию о том, что ДП в высоком, устанавливает СД в высокое (данные на ШД недействительны);

- 11 – источник установил на ШД новые данные;
- 12 – источник задерживает подтверждение истинности данных до готовности всех приемников;
- 13 – первый приемник по высокому СД установил ДП в низкое для следующего цикла работы;
- 14 – первый приемник готов к приему;
- 15 – последний приемник готов и установил ГП в высокое;
- 16 – источник по высокому ГП установил СД в низкое (истинность данных);
- 17 – первый приемник установил ГП в низкое и принимает данные с ШД;
- 18, 19, 20 соответствуют 8, 9, 10;
- 21 – источник снимает байт данных с ШД (конец передачи) после установления СД в высокое состояние;
- 22 – приемник по высокому СД устанавливает ДП в низкое для подготовки к следующему циклу.

## **2.4. РЕАЛИЗАЦИЯ ИНТЕРФЕЙСНЫХ ФУНКЦИЙ**

Физическими носителями информационных и интерфейсных сообщений служат кодированные сигналы, причем в КОП логическому нулю соответствует ВЫСОКИЙ УРОВЕНЬ напряжения, логической единице – НИЗКИЙ.

Интерфейсные функции, обеспечивающие прием, передачу и обработку данных, удобно описывать направленными графами состояний. Вершины графа отмечают последовательность состояний функции, а дуги – возможные переходы. Совокупность условий, при которых возникают эти переходы, отмечаются в виде логических выражений, состоящих из местных и дистанционных сообщений совместно с операторами И, ИЛИ, НЕ. Местные сообщения – это сообщения между функциями устройства и интерфейса, обозначаемые тремя строчными буквами, как показано в табл. 2.2.

### **Интерфейсные функции СП и СИ**

Некоторые возможные состояния интерфейсных функций СП и СИ представлены в табл. 2.3. Рассмотрим (рис. 2.6 и 2.7) направленные графы состояний этих интерфейсных функций.

Исходное состояние СП при включении "Холостой ход". Переход к следующему состоянию "Не готов" возможен лишь при наличии внешнего сигнала УП. Переход в состояние "Готов" возможен лишь при наличии двух внутренних (местных) сообщений:

- а) прибор переведен в асинхронный режим работы;
- б) прибор готов к приему очередного сообщения.

Таблица 2.2

Обозначение	Наименование местного сообщения
вкл	Питание включено
инб	Имеется новый байт
гтп	Готов для принятия следующего сообщения
прм	Принимать
тпд	Только передать
тпм	Только принимать
вус	Взять управление синхронно

Таблица 2.3

Обозначение	Наименование интерфейсной функции
СПХС	Состояние приемника "Холостой ход"
СПНГ	<< << "Не готов"
СПГТ	<< << "Готов"
СПРМ	<< << "Прием данных"
СПОЦ	<< << "Ожидание нового цикла"
СИХС	<< << "Холостой ход"
СИГР	<< << "Генерация"
СИЗД	<< << "Задержка"
СИПД	<< << "Передача"
СИОЦ	<< << "Ожидание нового цикла"

Переход в состояние "Прием" происходит при поступлении в приборный интерфейс внешнего сообщения сопровождения данных (СД), говорящего о достоверности последних. По истечении времени считывания данных функция переходит в режим ожидания, а после прекращения сигнала достоверности – в состояние "Не готов".

Приведем диаграммы состояний линий и интерфейсных функций СИ и СП при асинхронной передаче информации (рис. 2.8).

Пояснения к рис. 2.8:

- 1 – данные к передаче не готовы, приборы к приему не готовы;
- 2 – источник формирует очередной байт;

3 – формируется местный сигнал готовности источника «Имеется новый байт». Функция синхронизации передачи источника (СИ) переходит из состояния "Генерация" (источник формирует очередной байт) в состояние "Задержка" (СИЗД), когда сообщение "Имеется новый байт" истинно. Это состояние сохраняется до тех пор, пока все приемники не будут готовы;

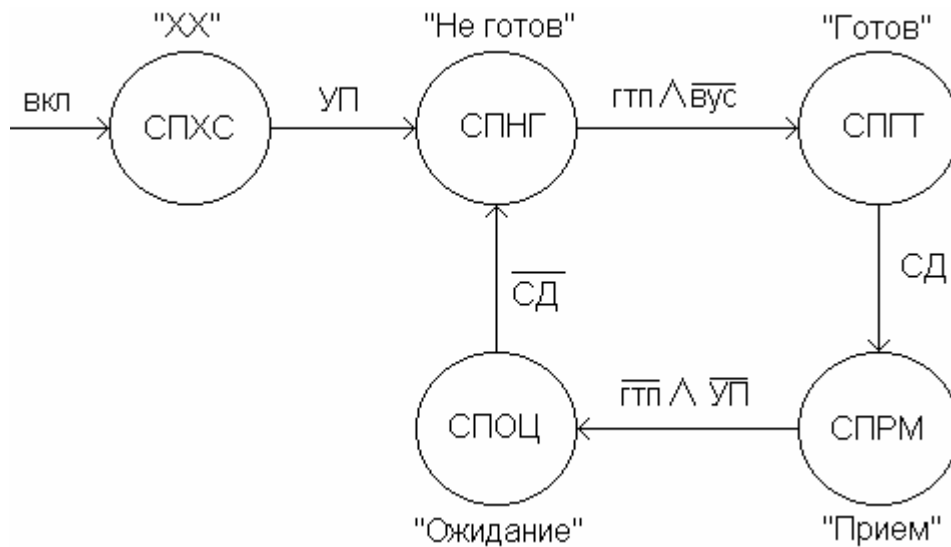


Рис. 2.6. Направленный граф состояний интерфейсной функции СП

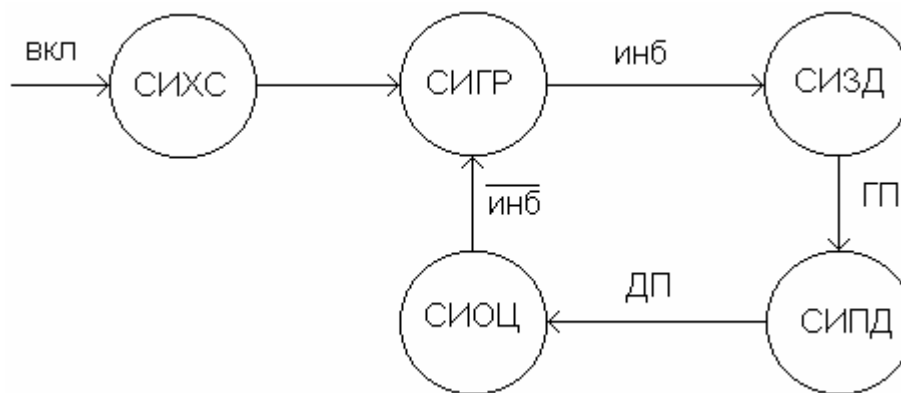


Рис. 2.7. Направленный граф состояний функции СИ

4 – приемник по местному сообщению гтп (готов к приему) переходит в состояние СПГТ;

5 – функции СП всех приемников перешли в состояние СПГТ. По мере готовности всех приемников ГП переходит в высокое состояние (логическая схема И-НЕ всех сигналов гтп);

6 – при появлении сигнала готовности на линии ГП функция СИ переходит в состояние "Передача данных" (СИПД) и формируется сигнал достоверности данных на линии СД (низкий уровень);

7 – при появлении сигнала достоверности функция СП переходит в состояние "Прием данных" и формируется сигнал низкого уровня на ГП, так как приемники не готовы для приема следующего байта;

8 – каждый приемник формирует местное сообщение «Не готов к приему», и функция СП переходит в состояние "Ожидание нового цикла" (СПОЦ);

9 – в состоянии ожидания по принятию данных всеми приемниками формируется сигнал "Данные приняты" (уровень ДП высокий);

10 – при появлении высокого ДП функция СИ переходит в состояние "Ожидание" (СИОЦ). Одновременно формируется сигнал недостоверности данных (высокий уровень СД);

11 – снимается байт с линий ЛД и готовится очередной;

12 – по сигналу "Данные недостоверны" функция СП переходит в состояние "Не готов" и формируется сигнал "Данные не приняты";

13 – функция СИ возвращается в исходное состояние СИГР (генерация источником нового байта сообщения).

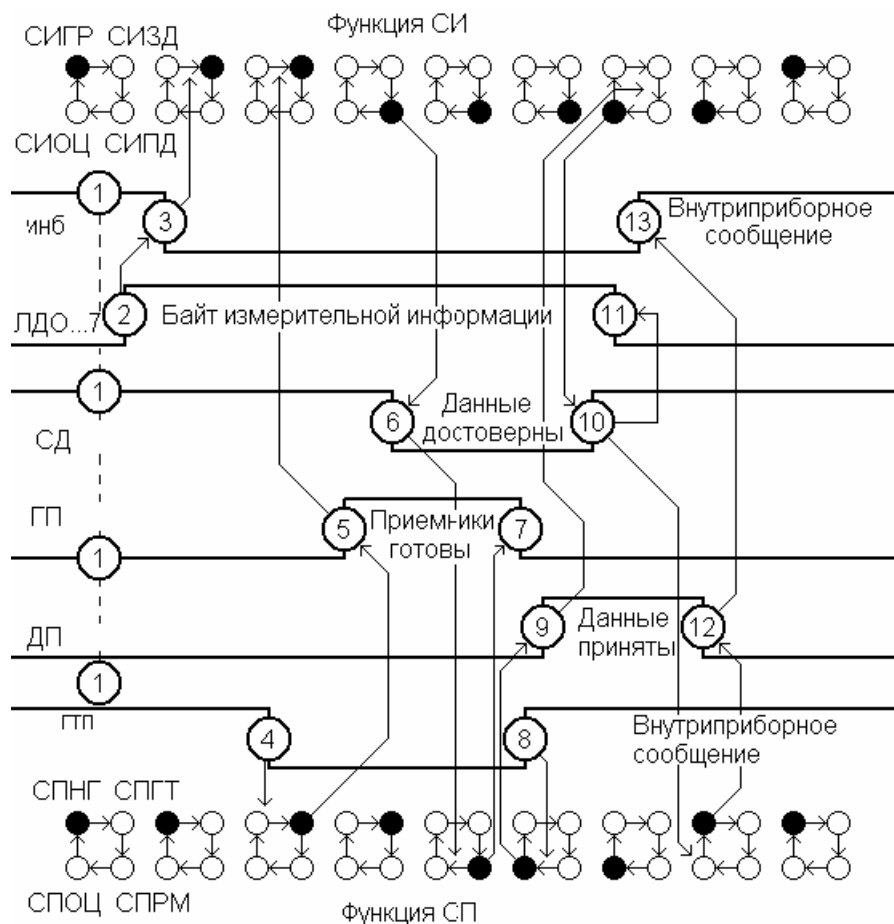


Рис. 2.8. Диаграммы состояний линий и интерфейсных функций



## Интерфейсная функция «Запрос на обслуживание»

Интерфейсная функция 3 используется прибором в тех случаях, когда он нуждается во внеочередном обслуживании. В табл. 2.4 приведена расшифровка состояний и условий перехода функции 3. Граф состояний приведен на рис. 2.9.

Сигнал ЗО формируется, например, если: прибор не работоспособен; прибор не готов; входной буфер переполнен; возникли ошибки программирования, синтаксиса или синхронизации.

Таблица 2.4

Наименование сообщения или состояния	Обозначение
Состояние «Последовательный опрос активен»	СПСА
Питание включено	вкл
Запрос на обслуживание	зно
Состояние «Отрицательная реакция на опрос»	СОРЗ
<< «Запрос на обслуживание»	СЗОб
<< «Положительная реакция на опрос»	СПРЗ

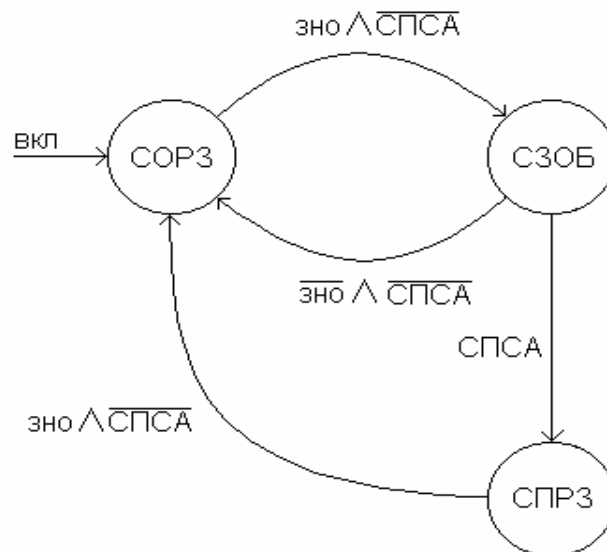


Рис. 2.9. Интерфейсная функция «Запрос на обслуживание»

Временная последовательность процесса идентификации при запросе на обслуживание показана на рис. 2.10.

Пояснения к рис. 2.10:

- 1 – прибор устанавливает линию ЗО в “1” (низкое);
- 2 – контроллер включает цикл идентификации ЗО;

- 3 – установка УП в низкое состояние для передачи команд;
- 4 – передача команды отпирания последовательного опроса ОПО из структуры ГУК, т.е. без адресации;
- 5 – установка адреса на передачу первого прибора (первый по списку);
- 6 – снятие сигнала УП;
- 7 – первый прибор посылает байт своего состояния;
- 8 – контроллер проверяет байт состояния и интерпретирует данные (ЛД6=0 – прибор не запрашивает обслуживание, ЛД6=1 – прибор запрашивает обслуживание, ЛД5...ЛД0 – данные состояния);
- 9 – если опрос окончен, то переход к позиции 14, если нужно опросить другие приборы, то переход к позиции 10;
- 10 – установка адреса на передачу другого прибора;
- 11 – установка УП в высокое состояние;
- 12 – второй прибор посылает байт своего состояния;
- 13 – проверка данных состояния и возврат к 9;
- 14 – все приборы опрошены;
- 15 – посылка универсальной команды (ГУК) «Запирание последовательного опроса» (ЗПО);
- 16 – установка УП в высокое состояние и продолжение работы системы.

Предполагается, что после проведения опроса всех потенциальных запросчиков, выявлен истинный запросчик, проанализирован его байт состояния и выполнены действия, снимающие запрос обслуживания.

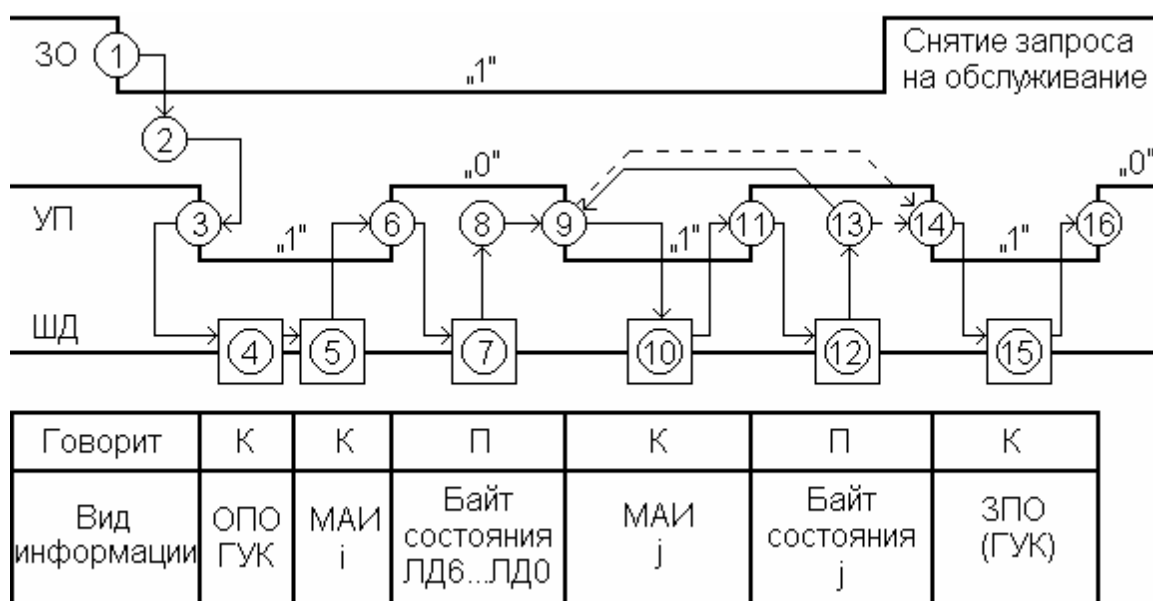


Рис. 2.10. Временная последовательность идентификации

## 2.5. БЫСТРОДЕЙСТВИЕ АКИС С ШИНОЙ IEEE-488

Работу интерфейса КОП в каждом измерительном цикле можно представить поэтапно:

**Этап конфигурирования:** *Формирование системным контроллером (компьютером) измерительной системы (конфигурирование)*

1. Инициализация подключенных к КОП приборов, проверка работоспособности системы. Сброс в исходное состояние по команде ОИ.

2. По команде УП по шине данных последовательно передаются адреса приемников и индивидуальная программная информация. Передаются команды о пределах измерений, параметрах генерируемых сигналов и т.д. В конце конфигурирования измерительной системы передается адрес источника – прибора, который будет передавать.

**Этап измерения:** *Выполнение программы измерений*

1. Контроллер снимает сигнал с линии УП и реализуется программа измерений. При необходимости может передаваться команда запуска.

2. Прерывается режим общего управления и вступает в действие механизм управления побайтной передачей измерительных данных от источника к приемникам. Обычно приборы передают в компьютер результаты измерений. Вместе с последним байтом прибор передает сигнал конца передачи (КП).

3. Контроллер восстанавливает сигнал УП и продолжает общее управление системой. Последовательно опрашиваются результаты измерений всех приборов, задействованных в данном цикле работы.

**Этап обработки:** *Обработка измерительной информации*

1. Вместе с опросом результатов измерений приборов системный компьютер осуществляет предварительную обработку данных: декодирование, масштабирование, нормирование и т.д.

2. После получения данных всех приборов ПК проводит комплексную обработку массива информации. Могут быть найдены их взаимосвязь, сумма, отношение и т.д.

Число измерительных циклов соответствует числу измеряемых параметров. После выполнения всех запланированных циклов контроллер обрабатывает информацию в целом и формирует заключение о состоянии объекта контроля. В зависимости от условий испытаний окончательное решение по дальнейшим шагам может приниматься или контролироваться оператором.

Полученные в результате обработки данные могут быть сохранены, распечатаны, переданы другим пользователям, применены в системах управления технологическим процессом и т.д.

Быстродействие измерительной системы зависит от ее архитектуры, алгоритмов настройки и измерения, возможностей приборов и контроллера, языка общения и программ обработки данных. Рассматривая быстродействие измерительных систем, необходимо разделить их на две основные группы: системы без автоподстройки и системы с автоподстройкой.

#### А. СИСТЕМЫ БЕЗ АВТОПОДСТРОЙКИ

Совокупное время измерения параметра  $i$ , не требующего дополнительных шагов для выхода в рабочую точку, можно представить в виде суммы:

$$T_{\square i} = T_{\text{конф } i} + T_{\text{изм } i} + T_{\text{пер } i} + T_{\text{обр } i} + T_{\text{польз } i},$$

где  $T_{\text{конф } i}$  – время конфигурирования измерительной системы из подключенных к шине приборов для измерения  $i$ -го параметра;

$T_{\text{изм } i}$  – время измерения  $i$ -го параметра в соответствии с выбранным алгоритмом;

$T_{\text{пер } i}$  – время передачи информации контроллеру;

$T_{\text{обр } i}$  – время обработки информации;

$T_{\text{польз } i}$  – время взаимодействия с пользователем информации.

Полное время измерения системы параметров определяется совокупным временем измерения каждого параметра (рис. 2.11), а также дополнительными затратами на подключение объекта ( $t_{\text{подкл}}$ ) и общую обработку информации ( $t_{\text{общ}}$ ):

$$T_{\text{пойлн}} = \sum_{i=1}^N T_{\Sigma i} + t_{\text{подкл}} + t_{\text{общ}}.$$

##### *1. Конфигурирование измерительной системы*

При конфигурировании измерительной системы приборы инициализируются, устанавливаются в требуемые режимы работы и программируются на выполнение заданных функций. В зависимости от вида прибора это время может быть от сотен микросекунд до десятков миллисекунд.

Следует обратить внимание, что использование внутренней автоматки приборов обычно увеличивает время выхода в рабочий режим, так как требуется дополнительные шаги на автоподстройку модулей системы. Например, режим автоматического выбора предела (АВП) вольтметра

предполагает сравнение сигнала с нижним и верхним пороговыми уровнями измерения на нескольких шкалах, начиная с наибольшего предела измерения до выхода в рабочий диапазон.

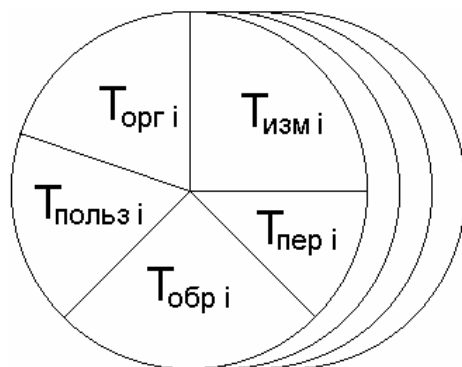


Рис. 2.11. Диаграмма затрат времени на измерение группы параметров

## 2. Время измерения

Если прибор находится в режиме многократных измерений, тогда на быстродействующие преобразователи влияние оказывает задержка начала измерения по отношению к готовности системы, как показано на рис. 2.12.

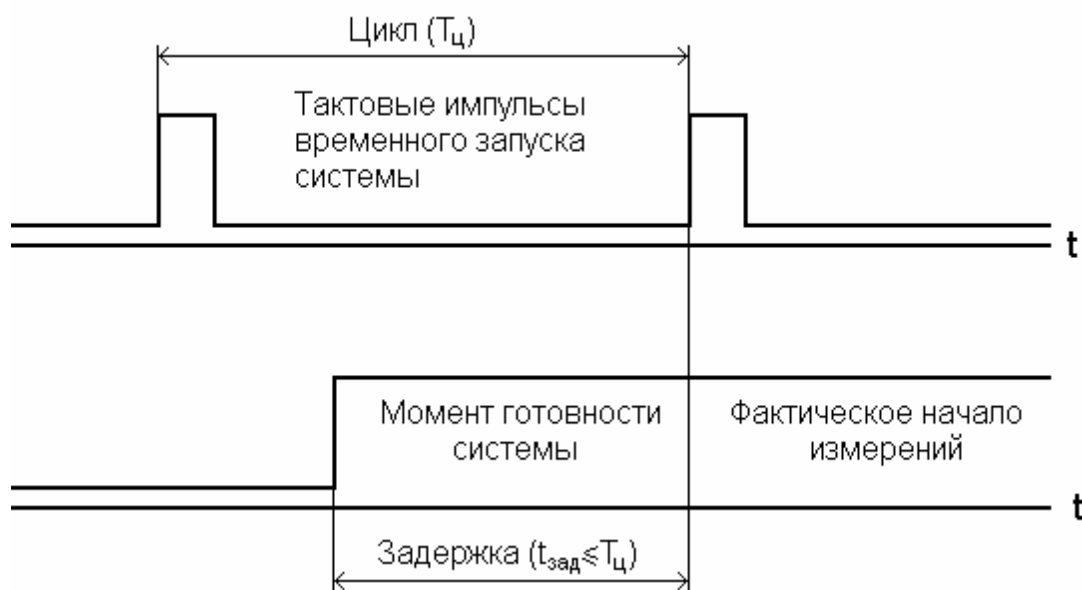


Рис. 2.12. Задержка начала измерения

В установившемся режиме прибору на измерение требуется время, которое определяется принципом работы первичного и аналого-цифрового преобразователей. Наибольшей инерцией обладают тепловые и механические преобразователи. Например, калориметрический преобразователь

может затрачивать на измерение до единиц минут. Все "интегрирующие" приборы работают медленно.

### *3. Время передачи данных*

Передача данных осуществляется побайтно. Максимальная скорость передачи в соответствии со спецификацией IEEE-488.1 достигает 1 Мбайт/с [1]. Обычно скорость передачи данных не ограничивает быстродействие системы. Исключением являются лишь случаи больших информационных потоков, например, при работе в реальном масштабе времени. Фирма National Instruments предложила в 1993 году дополнения к стандарту IEEE-488.1 в виде патентованного протокола скоростной передачи информации hs488 (high-speed).

Стандарт IEEE-488.1 с трехпроводной системой асинхронной передачи данных требует, чтобы приемник, установив сигнал ГП, ожидал подтверждения источником истинности каждого байта на линии СД и только затем обрабатывал байт информации, выставляя на линии сигнал ДП, разрешающий установку следующего байта. При этом байт нельзя передать за время меньшее, чем требуется на распространение сигналов по цепочке: приемник (ГП) – источник (СД) – приемник (ДП) – источник (новый байт на шине данных).

Протокол hs488 увеличивает производительность системы, удаляя задержки распространения сигналов. Сначала все приемники устанавливаются в состояние "Готов к приему". Далее источник формирует импульсный сигнал на линии ГП, который открывает протокол скоростной передачи. Это существенное отличие hs488, так как типовая процедура предполагает, что линией ГП управляют только приемники. Затем источник сообщений помещает байт на шину данных, ждет в течение предварительно запрограммированного времени и устанавливает сигнал СД. Далее источник снимает сигнал истинности данных и устанавливает следующий байт данных. Приемники устанавливают сигнал ДП и не снимают его в течение времени, когда передается блок данных.

Байты передаются в синхронном режиме без ожидания подтверждающих сигналов. Если какой-то приемник не успевает принять данные в установленном темпе запрограммированных задержек, тогда он останавливает процесс передачи, и источник переходит в обычный режим асинхронного обмена. Установка программируемых интервалов задержек осуществляется с учетом конфигурации системы. На среднюю скорость пере-

дачи информации значительное влияние оказывает архитектура контроллера и размер передаваемого сообщения. Для двух модулей, соединенных кабелем длиной 2 м, максимальная скорость передачи данных достигает 8 Мбайт/с. Для полностью загруженной системы с 15 устройствами и 15 м кабеля скорость передачи может достигать 1,5 Мбайт/с.

#### *4. Время обработки данных*

Эта составляющая определяется быстродействием процессора и особенностями обработки данных. Наиболее трудоемкими являются задачи решения систем уравнений, прямого и обратного преобразования Фурье, графических преобразований. Слабые возможности компьютера при решении сложных математических и графических задач могут потребовать затрат времени до десятков секунд. Однако для большинства обычных измерений на обработку результатов требуется от единиц до сотен миллисекунд.

#### *5. Время взаимодействия с пользователем*

Если процедура измерения, обработки данных или принятия решения рассчитана на непосредственное участие человека, то высокого быстродействия получить уже нельзя. Вместе с тем не все можно предусмотреть программно, и в наиболее ответственных случаях последнее слово остается за человеком, который, в отличие от ПК, обладает интуицией.

### **Б. СИСТЕМЫ С АВТОПОДСТРОЙКОЙ**

Если в процессе работы необходима автоподстройка системы, например поиск экстремума или точки пересечения, то время измерения в значительной степени зависит от алгоритма выхода в рабочий режим.

Примерами задач измерения функционалов положения и протяженности “обобщенного импульса” являются исследования диаграмм направленности антенн, амплитудно-частотных характеристик линейных четырехполюсников, спектральных функций и некоторых других характеристик. Функционал положения определяет расстояние вдоль оси аргумента от начала координат до “центра” функции. Функционал протяженности характеризует протяженность или ширину функции. Известны следующие методы измерения “обобщенного импульса”:

- метод выбранных точек или условных уровней, основанный на достаточно произвольном способе выбора значений функции;
- метод моментов, определяющий “центр тяжести” и “дисперсию”;
- метод нормированных функционалов.

Наиболее широкое распространение получил метод выбранных точек или условных уровней, благодаря возможности оценки функционалов по привычным графикам. Примерами функционалов положения и протяженности являются: центральная частота и полоса пропускания на уровне 3 дБ амплитудно-частотной характеристики четырехполосника; направление максимума главного лепестка и ширина диаграммы направленности антенны на уровне половинной мощности.

В специализированных автоматизированных приборах изменение аргумента осуществляется непрерывно и на экране панорамного индикатора вычерчивается исследуемая характеристика. Дискретное изменение аргумента в цифровых приборах вынуждает выбирать шаг квантования в соответствии с требуемой разрешающей способностью или точностью измерений. Большое число точек отсчета резко замедляет оцифровывание характеристики.

Для повышения быстродействия систем с автоподстройкой методы измерения функционалов положения и протяженности должны базироваться на оптимальных алгоритмах поиска точек отсчета. Поскольку при каждом измерении информация передается в контроллер и там анализируется для коррекции следующего шага, время полного измерения для  $k$  шагов автоподстройки системы будет иметь вид:

$$T_{\Sigma i} = T_{\text{конф } i} + k(T_{\text{изм } i} + T_{\text{пер } i} + T_{\text{ан } i}) + T_{\text{обр } i} + T_{\text{польз } i},$$

где  $T_{\text{ан } i}$  – время анализа и обработки каждого измерения.

Для выхода в рабочий режим измерения может использоваться метод последовательного приближения. Наиболее простой является процедура пошагового интерполирования и взвешивания.

#### **Общие рекомендации по повышению быстродействия:**

- 1) по возможности исключать автоматику медленных приборов (выбор полярности, автоматический выбор предела);
- 2) минимизировать задержку пуска;
- 3) оптимизировать алгоритмы взаимодействия модулей и конфигурации системы для каждого измерения;
- 4) использовать высокоскоростной протокол hs488 и двоичный формат передачи данных;
- 5) совмещать обработку данных с другими задачами (например, установкой приборов в номинальный режим);



- б) программировать системы в машинных кодах с применением ПЗУ для минимизации времени загрузки;
- 7) применять приборы с более высоким быстродействием;
- 8) по возможности исключать участие оператора из цикла работы;
- 9) оптимизировать алгоритм автоподстройки системы в точку измерения.

## 2.6. ПРИНЦИПЫ РЕАЛИЗАЦИИ ИНТЕРФЕЙСА

Типичная схема драйвера шины IEEE-488 требует использования 8-разрядных двунаправленных буферов с встроенными резистивными нагрузками и применения схем с открытым коллектором по выходу или схем на три состояния.

*Требования к возбуждителям:*

1. В сигнальных линиях ЗО, ГП, ДП должны использоваться возбуждители с открытым коллектором.

2. В сигнальных линиях ЛД, СД, ОИ, УП, ДУ, КП используются возбуждители с открытым коллектором или возбуждители на три состояния. Если в составе интерфейса содержится функция "Параллельный опрос", то на ШД должны использоваться только возбуждители с открытым коллектором. Возбуждители на три состояния применяются там, где нужна более высокая скорость работы.

3. Низкое состояние  $U_H < 0,5$  В при  $I < 48$  мА (втекающий ток). Высокое состояние определяется характеристикой нагрузки. Для возбуждителей с тремя состояниями  $U_B > 2,4$  В при  $I < -5,2$  мА. Знак минус у тока показывает, что он вытекающий (между сигнальной линией и логической землей).

*Требования к приемникам:*

1. Низкое состояние  $U_H < 0,8$  В, высокое состояние  $U_B > 2,0$  В.

2. Для повышения помехоустойчивости на всех сигнальных линиях рекомендуется использовать цепи типа "триггер Шмитта".

*Требования к нагрузке:*

1. Каждая сигнальная линия должна быть нагружена резистивной нагрузкой для стабилизации высокого состояния и для увеличения помехоустойчивости.

2. Каждая сигнальная линия должна иметь средства ограничения отрицательного напряжения (диодный ограничитель).

3. Внутренняя емкость в каждой сигнальной линии  $C < 100$  пФ. Типичная схема подключения к линии КОП приведена на рис. 2.13.

*Требования к разъему:*

1. Применяется разъем типа РПМ7-24 с 24 ленточными контактами ( $U < 150$  В;  $I < 1$  А;  $R_{\text{конт}} < 0,02$  Ом;  $R_{\text{изол}} > 1$  ГОм;  $f < 3$  МГц; 500 соединений).

2. Распайка в соответствии с табл. 2.5.

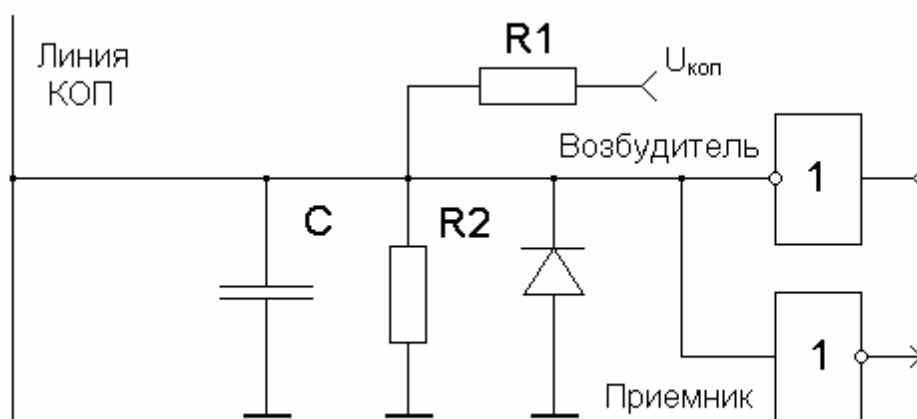


Рис. 2.13. Типичная схема подключения к линии:  $C < 100$  пФ;  $R1 = 3$  к;  
 $R2 = 6,2$  к;  $U_{\text{ист}} = (5 \pm 0,25)$  В

Таблица 2.5

% п/п	Линия	N% п/п	Линия	% п/п	Линия
1	ЛД0	9	КП	17	ОИ
2	ЛД1	10	ДУ	18	СП ОИ
3	ЛД2	11	СД	19	ЗО
4	ЛД3	12	СП СД	20	СП ЗО
5	ЛД4	13	ГП	21	УП
6	ЛД5	14	СП ГП	22	СП УП
7	ЛД6	15	ДП	23	Экран
8	ЛД7	16	СП ДП	24	Логическая земля

*Примечание.* СП СД (скрученная пара СД) обозначает, что провод 12 скручивается с 11. Прибор содержит розетку, а кабель на обоих концах – розетку и вилку (параллельное соединение), что позволяет одновременно подключать несколько устройств.

*Требования к кабелю:*

1. Каждая сигнальная линия СД, ГП, ДП, ОИ, УП, ЗО должна быть скручена с одним из проводов "логической земли".

2. Кабель должен иметь экран с плотностью оплетки не менее 85 %.
3. Максимальная емкость (на  $f = 1$  кГц) между любой сигнальной линией и любой другой, в том числе экранирующей, должна быть менее 150 пФ/м.

### Специализированные интегральные схемы

Резкое увеличение производства приборов и устройств, имеющих интерфейс КОП (IEC-625, IEEE-488), стимулировало разработку целого ряда БИС и схем поддержки интерфейса. Современные контроллеры КОП на БИС примерно в 10 раз дешевле аналогичных контроллеров, реализованных на ИС малой и средней степени интеграции и требующих применения от 40 до 60 корпусов ИС. В табл. 2.6 приведены характеристики некоторых зарубежных БИС для интерфейса IEEE-488.

Схема 8291 функционально идентична MC68488, но имеет более высокое быстродействие. Схемы 8291 и 8292 оптимизированы для МП типов 8080, 8085А и 8086. Отечественные аналоги: КР580ВК91 и КР580ВК92.

Схема MC68488 для реализации всех возможностей интерфейсных функций источника (И) и приемника (П) требует использования только двух драйверов шины. Драйвер MC3448 четырехразрядный, а MC3447 восьмиразрядный. Таким образом, сочетание только трех схем реализует весь интерфейс. Впоследствии была разработана БИС на основе программируемого микропроцессора МП6801, реализующая функции контроллера (К), что позволяет выполнить все функции И/П/К на четырех БИС.

Схема TMS9914 является первым прибором, реализующим все функции И/П/К. В сочетании с драйверами шин требуется только три схемы для реализации интерфейса.

Таблица 2.6

Фирма	Тип БИС	Рассеиваемая мощность, мВт	Скорость передачи данных, кБайт/с	Функции	Схемы поддержки
1) INTEL, NEC	8291	500	448	И/П/К	8293
2) INTEL, NEC	8292	625	448	И/П/К	8293
3) FAIRCHILD	96LS488	1250	1000	И/П	-
4) MOTOROLA, FAIRCHILD	MC68488	600	125	И/П (К)	6801,3447, 3448
5) TEX. INST	TMS9914	750	200	И/П/К	SN7561
6) PHIL / SIGN	HEF4738	1	200	И/П	Много

Применение технологии Шоттки в ИМС 96LS488 позволило встроить в схему драйверы шин и обеспечило максимальное быстродействие, равное в соответствии с требованиями стандарта 1 Мбайт/с.

### Универсальный интерфейс прибора

Функциональная схема универсального интерфейса измерительного прибора, реализующего все интерфейсные функции, кроме функции контроллера, приведена на рис. 2.14. Каждый узел реализует свою интерфейсную функцию, поэтому всего узлов девять.

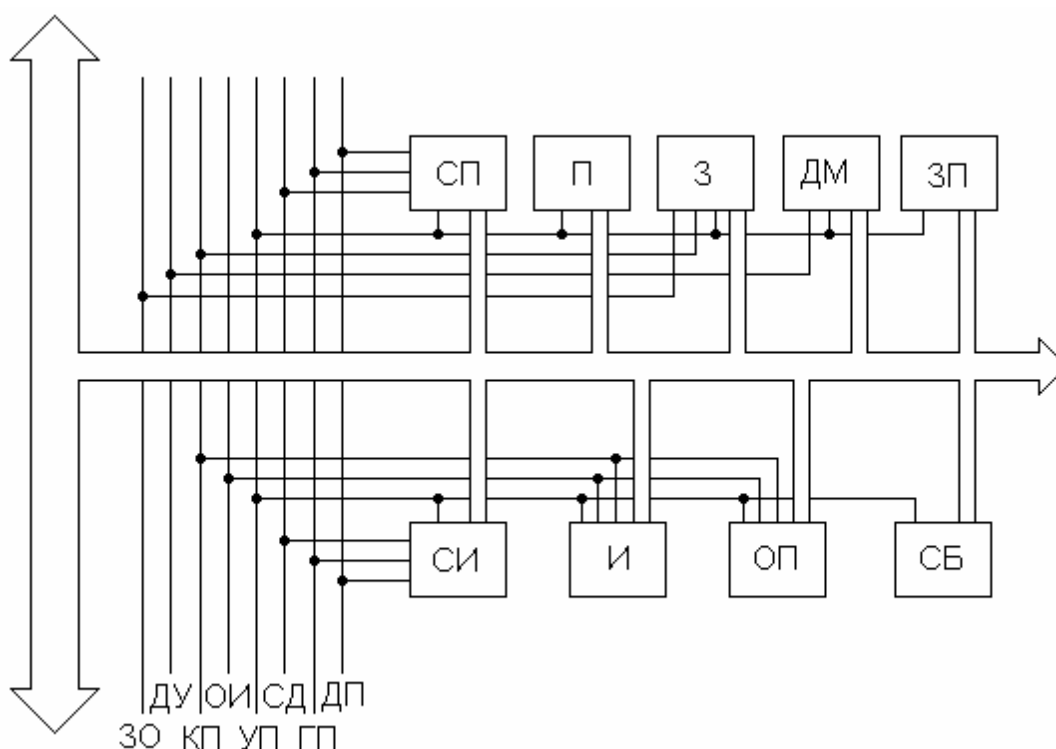


Рис. 2.14. Функциональная схема универсального интерфейса

## 2.7. АРХИТЕКТУРА СИСТЕМ КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА

Сложность и быстродействие контрольно-измерительных систем зависит от числа и особенностей измерительных каналов, возможностей аппаратной части (приборов-модулей, контроллера, ЭВМ) и программного обеспечения. Значительное влияние оказывают условия эксплуатации и характеристики объекта контроля.

По особенностям построения и виду архитектуры можно выделить следующие наиболее важные классы АКИС:

- одноканальные и многоканальные;
- однопараметрические и многофункциональные;
- однопроцессорные и многопроцессорные;
- компактные и разнесенные в пространстве.

Рассмотрим некоторые варианты архитектуры. На рис. 2.15 приведена схема простейшей одноканальной однопараметрической системы, содержащей генератор и измеритель сигнала.

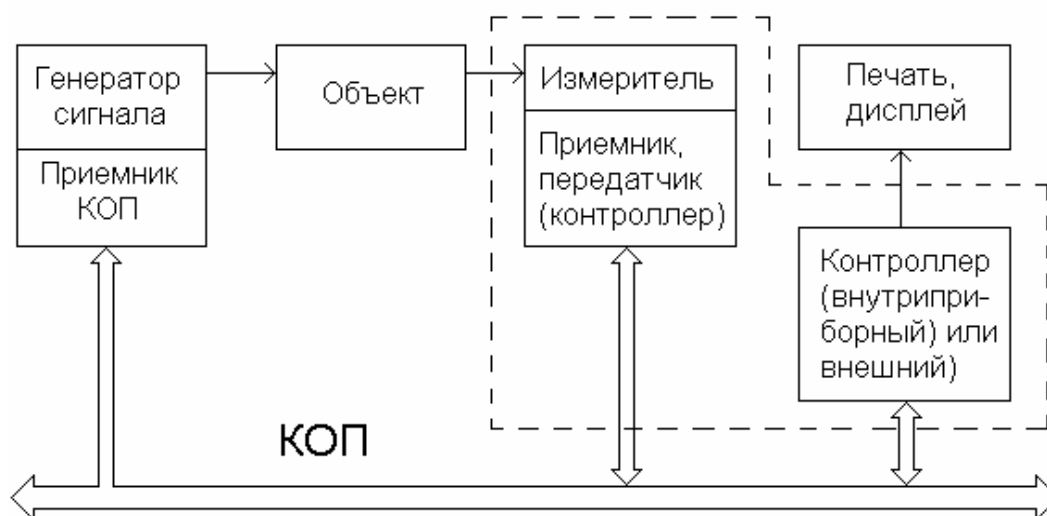


Рис. 2.15. Одноканальная система

Здесь нет коммутаторов и схемы сопряжения с объектом. Функции контроллера может выполнить один прибор. Алгоритм работы включает следующие этапы:

- а) контроллер устанавливает генератор;
- б) измеритель измеряет и передает информацию в контроллер;
- в) контроллер осуществляет обработку и представление данных.

Схема многоканальной однопараметрической АКИС приведена на рис. 2.16.

Здесь измеряется только один параметр, но число каналов контроля велико, поэтому приходится использовать управляемый сигнальный коммутатор, обеспечивающий требуемую маршрутизацию входов и выходов. Число приборов в системе небольшое.

На рис. 2.17 приведена схема многоканальной однопараметрической системы, состоящей из двух разнесенных в пространстве частей. Для ис-

следования прохождения сигнала в прямом и обратном направлении в каждой части имеется как генератор сигнала, так и прибор для измерения. Одновременно работает один генератор и один удаленный от него измеритель. Подобная схема построения АКИС необходима, например, при исследовании кабельных линий связи. В системе используется два контроллера, один из которых главный, а второй обеспечивает процедуру обмена информацией с удаленной частью.

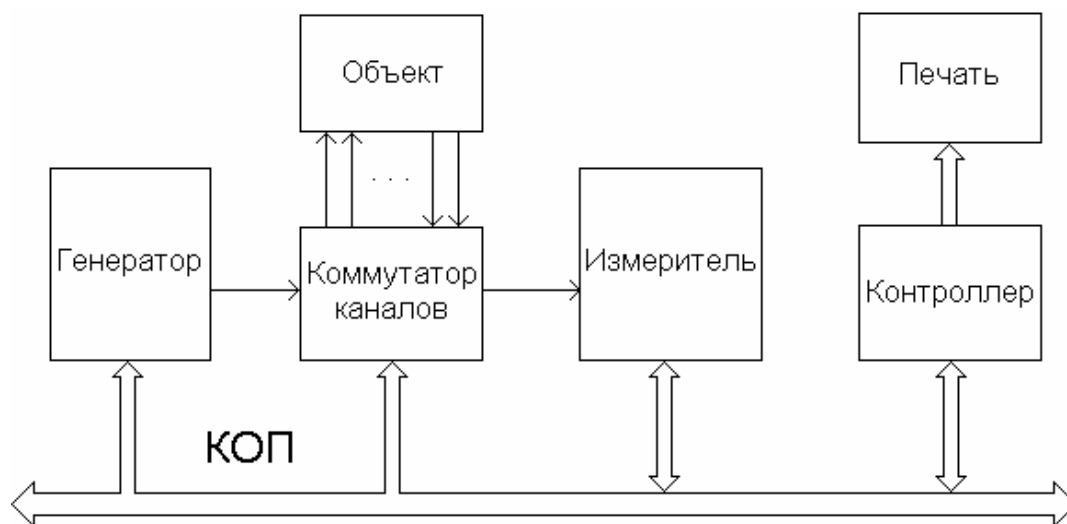


Рис. 2.16. Многоканальная система с коммутацией каналов

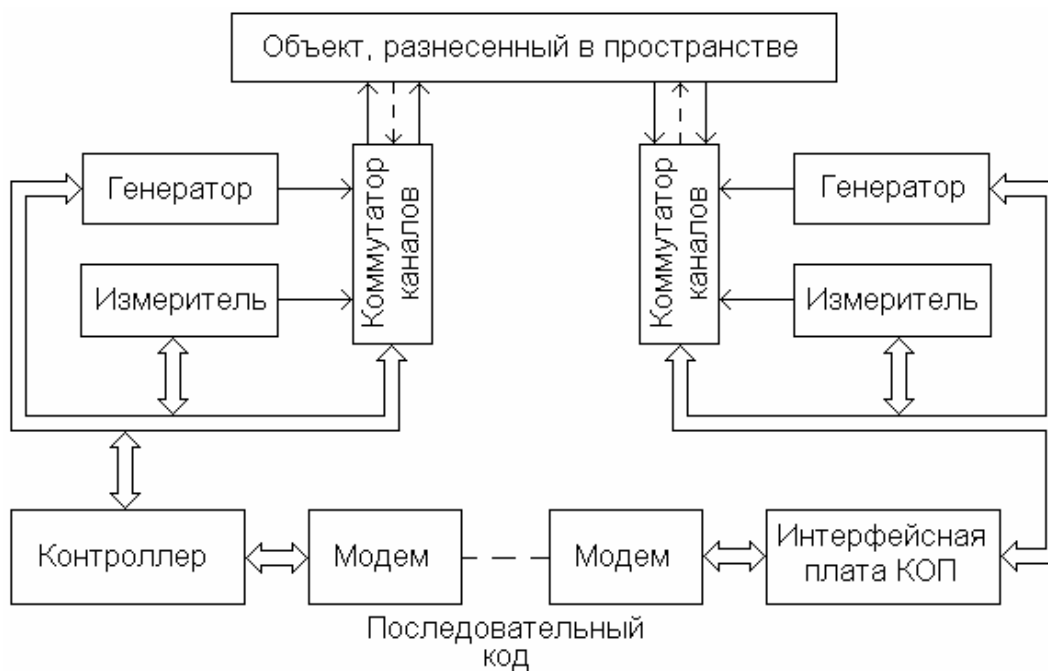


Рис. 2.17. Многоканальная система с разнесением в пространстве

На рис. 2.18 представлена схема многофункциональной АКИС, предназначенной для измерения параметров и исследования характеристик электромагнитной совместимости сложной радиоэлектронной системы, например системы связи. Блок сопряжения и коммутации обеспечивает подключение приборов и измерительных антенн. Измерения в области низких частот выходного сигнала приемника осуществляются мультиметром В7-34, который также используется для контроля питающих напряжений.

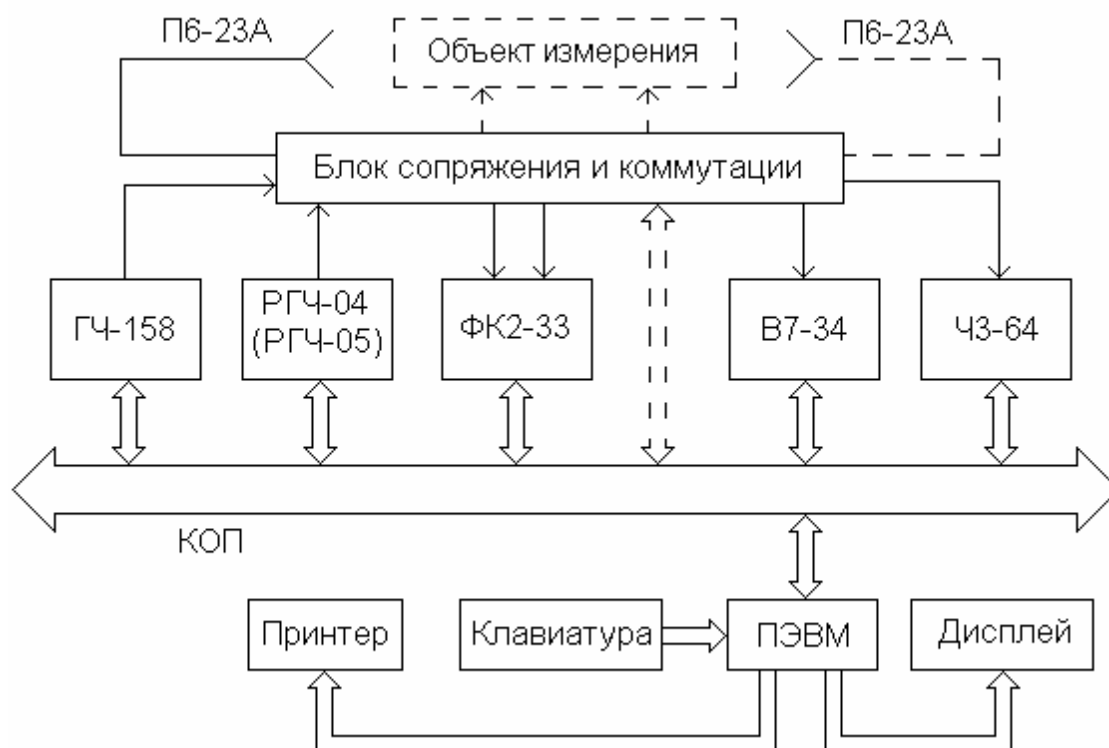


Рис. 2.18. Многоканальная и многофункциональная система

Частотно-временные измерения в диапазоне промежуточных частот передатчика и приемника осуществляются с использованием ЭСЧ ЧЗ-64 и генератора Г4-158. Измерения в диапазоне СВЧ предполагают применение прибора ФК2-33, генераторов РГ4-04, 05 и измерительных антенн П6-23А.

Объект контроля может также иметь интерфейс КОП и управляться сигналами ПК. Контроллер может общаться с периферийными модулями по шине КОП или через специальные порты ввода и вывода информации.

В систему могут вводиться приборы, не имеющие собственный интерфейсный модуль КОП. Но в этом случае требуются дополнительные модули сопряжения, обеспечивающие преобразование сигналов. В представленной на рис. 2.18 схеме таким прибором является генератор Г4-158.

Для его нормальной работы в системе применен специальный преобразователь, обеспечивающий “прозрачность” управления по шине КОП.

Блок сопряжения и коммутации обеспечивает согласование всех входов и выходов объекта измерения с соответствующими входами и выходами приборов, а также всю необходимую коммутацию.

## **2.8. АППАРАТНЫЕ СРЕДСТВА РАСШИРЕНИЯ ШИНЫ КОП**

Использование средств расширения шины КОП позволяет:

- превысить установленное стандартом КОП число подключаемых модулей (по стандарту до 15);
- обеспечить развязку высокочувствительных измерительных приборов и модулей, работающих с высокими напряжениями и создающих значительные помехи на линиях шины КОП;
- увеличить установленную стандартом длину магистрали КОП (по стандарту до 20 м).

### **Расширители числа модулей системы**

Простейший расширитель имеет два порта: входной и выходной. Он буферизует и повторяет сигналы главной и буферизируемой ветви шины КОП. Со стороны главной ветви он по нагрузочной характеристике эквивалентен одному модулю, позволяя к главной ветви подключить до 14 модулей. К выходу расширителя можно подключить еще 14 модулей, доведя общее число управляемых устройств до 28. При этом можно использовать расширитель для соединения удаленных устройств, передающих информацию по шине длиной до 40 м. Каждый новый сегмент расширителя, включаемый вместо одного из 14 модулей, позволит еще на 20 м удлинить магистраль и подключить дополнительную ветвь из 14 приборов. Имея два расширителя, можно подключить 41 модуль. При этом, однако, надо иметь в виду, что при однобайтовой адресации по стандарту КОП можно подключить только 31 модуль. Большее число модулей требует использования двухбайтовой адресации, что для обычных серийных приборов нереализуемо.

Идеально расширитель при включении его в систему должен быть прозрачным к прикладному программному обеспечению. Поскольку расширитель не нужно адресовать, все модули в системе связываются друг с другом так, как будто расширителя нет. Для работы нужно просто вклю-



чить его в систему. Применение расширителей не требует новой конфигурации аппаратных средств и программного обеспечения. Можно устанавливать контроллер с обеих сторон расширителя без изменения адресов модулей в системе.

Идеально при включении расширителя в систему, не должно происходить уменьшение быстродействия передачи между устройствами. Однако из-за задержки распространения сигнала сквозь расширитель передача данных между устройствами на различных сторонах расширителя медленнее, чем передача между устройствами на одной стороне расширителя. Эти задержки при распространении свойственны всем расширителям, однако величина задержки изменяется в зависимости от качества расширителя. Необходимо выбирать расширитель, который имеет минимальную задержку распространения.

### Расширители – изоляторы

Другая важная особенность расширителя – развязка напряжений между двумя портами шины КОП. Расширитель физически разделяет ветви системы, так что можно размещать чувствительные устройства с одной стороны, а шумящие устройства – с другой стороны расширителя. В то время как физически устройства в системе расширителя развязаны, логически они остаются частью единой системы. Можно развязывать устройства с разными рабочими мощностями, шумящие и чувствительные приборы, понижать возможное проникновение шумовых сигналов и помех.

На рис. 2.19 показана конфигурация систем КОП, при которой чувствительные модули изолированы от высоковольтных и создающих помехи устройств.



Рис. 2.19. Система развязки приборов КОП

Например, расширитель GPIB-120A Expander фирмы National Instruments электрически буферизует и позволяет увеличить число устройств,

подключаемых к системе, до 28. Он может использоваться в системах, которые требуют электрической развязки между модулями. Он специально создан для обеспечения прозрачности протокола КОП и имеет очень короткие задержки при распространении и передаче данных. Скорость обмена данными может превышать 1 Мбайт/с.

### **Расширители – удлинители шины КОП**

Во многих задачах мониторинга, испытаний и измерений требуется расположение отдельных модулей системы на удаленном расстоянии, достигающем многих сотен метров и даже километров. Решение данных задач возможно путем использования специализированных удлинителей шины КОП.

Каждый такой удлинитель состоит из двух специальных модулей, соединяемых между собой посредством коаксиальных, волоконно-оптических или многожильных кабелей. Каждый модуль имеет один порт шины КОП и один специальный порт удаленной связи. Удаленный модуль удлинителя преобразует сообщения согласно протоколу шины КОП.

Единственной проблемой взаимодействия остается реализация процедуры параллельного опроса, которая затруднена значительными задержками распространения сигналов. Для того чтобы реакция устройства успевала достичь контроллера, необходимо изменить время параллельного опроса, и тогда не потребуется изменять прикладное программное обеспечение.

Вводя удлинитель в систему, можно увеличивать расстояние между двумя устройствами до нескольких километров. Поскольку можно присоединить 14 устройств к порту каждого удлинителя, то это позволяет увеличить число устройств до 28, при этом оставаясь в пределах требований стандарта КОП.

Каждый удлинитель передает удаленной системе КОП кодированные сигналы и декодирует сигналы, полученные от удаленной системы. Идеальные удлинители подобно расширителям должны быть прозрачны к прикладному программному обеспечению. То есть включение или удаление удлинителя из системы не должно влиять на работу. При применении удлинителей нет необходимости изменять аппаратные средства и программное обеспечение.

Тем не менее каждый удлинитель шины КОП ухудшает быстродействие системы из-за больших расстояний и задержки распространения сигналов.

Типы кабелей, используемых в удлинителях: многожильный, коаксиальный, витая пара и волокно-оптический. Каждый тип кабеля имеет особенности, которые делают его наиболее подходящим к применению в конкретной системе. Многожильный кабель наиболее дорогой, но он на коротких расстояниях имеет самую высокую скорость передачи. Коаксиальная линия или витая пара наименее дорогие, но их восприимчивость к шуму и длинные задержки при распространении сигналов делают их неподходящими для применения на очень больших расстояниях, кроме тех случаев, когда быстродействие не является определяющим фактором.

Волоконно-оптические кабели могут удлинять систему на большее расстояние с наименьшим снижением быстродействия. Они обладают превосходной помехоустойчивостью и наиболее эффективны в промышленных применениях, когда рядом высоковольтные провода и работают агрегаты. Волоконно-оптический кабель непроводящий, и это делает его подходящим для расположения на открытом воздухе, где возможно воздействие грозы. Наконец, волоконно-оптические кабели увеличивают защиту информации. Поскольку кабели передают свет вместо электрического тока, то при передаче не происходит излучения электромагнитного поля. Тракт передачи сигналов устойчив к несанкционированному прослушиванию.

Удлинитель GPIB-130 фирмы National Instrument использует многожильный кабель, который позволяет увеличить расстояние до 300 м при подключении до 28 устройств. При этом скорость передачи данных достигает 900 кбайт/с. Удлинитель GPIB-110 предназначен для удлинения шины на большие расстояния. При использовании волоконно-оптического или коаксиального кабеля шина может быть удлинена до 2 км. Для гарантии безошибочной передачи данных используется схема их проверки и восстановления.

Наиболее экономичное решение задачи удлинения шины КОП до 300 м – использование GPIB-110 с коаксиальным кабелем. При этом решаются также задачи защиты от шумов и развязки. Если необходима высокоскоростная передача данных, то нужно выбрать GPIB-130 с многожильным кабелем.

На рис. 2.20 представлена общая схема включения нескольких расширителей, обеспечивающих решение сразу нескольких задач: увеличение числа модулей (более 15) в ближней зоне расположения ПК (ветвь расши-

рения); обеспечение развязки прецизионного оборудования, расположенного также в ближней зоне (ветвь развязки); обеспечение стандартного взаимодействия с удаленными от ПК модулями (ветвь удаления).

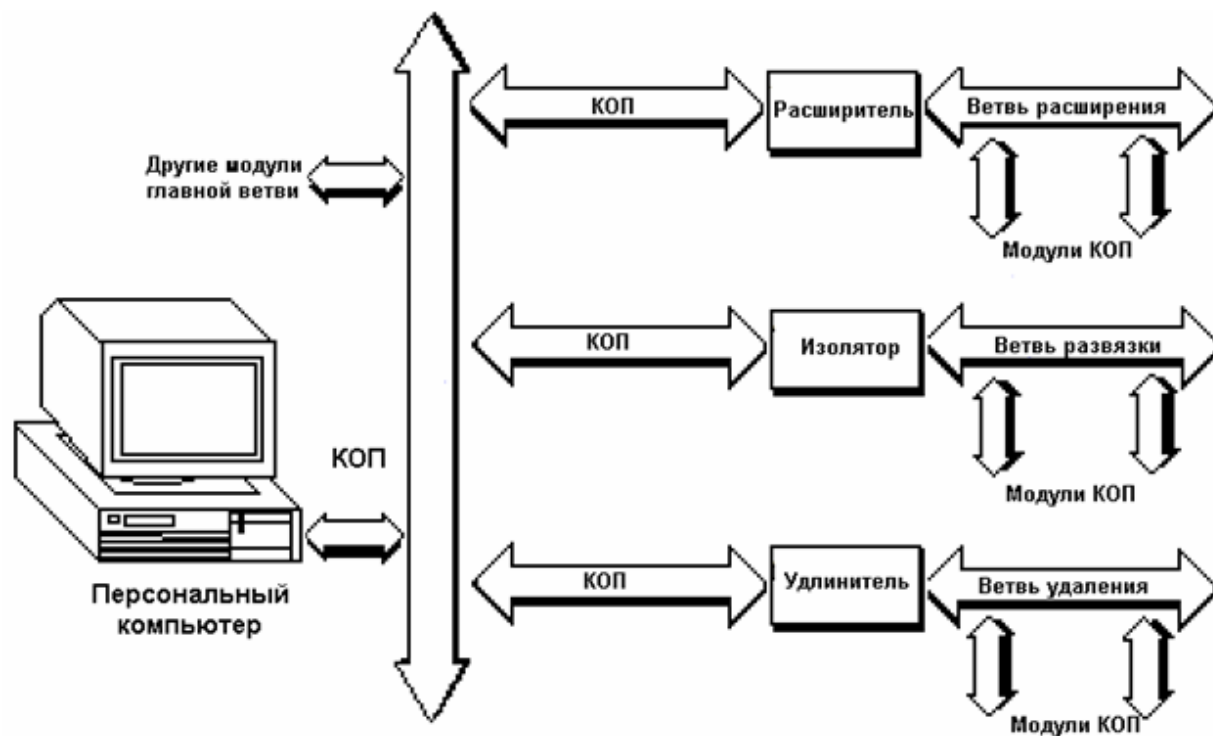


Рис. 2.20. Комбинированное использование расширителей

## **Глава 3. ПРОГРАММНЫЕ СРЕДСТВА СИСТЕМ КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА**

### **3.1. БАЗОВЫЕ И СИСТЕМНЫЕ ПРОГРАММНЫЕ СРЕДСТВА**

Программные средства, необходимые для компьютеризации экспериментальных исследований с помощью наиболее распространенных приборно-модульных систем с шиной КОП (ГОСТ 26.003-80 [1]), можно разделить на низкоуровневые и высокоуровневые. Низкоуровневые, или базовые, программные средства обеспечивают необходимые интерфейсные функции приборов в соответствии с требованиями стандарта. Они реализуются в интерфейсном модуле персонального компьютера (ПК) и обеспечивают асинхронный режим передачи данных по шине.

Высокоуровневые или системные программные средства [5,9,11] предназначены для решения прикладных задач по организации эксперимента, сбору, обработке данных и представления результатов инструментальных исследований. Они реализуются в ПК и определяются составом прикладных программ, а также возможностями используемого компьютера.

#### **Базовые программные средства**

Подпрограммы обслуживания КОП пишутся на Ассемблере или на языке программирования системы. Построенные на основе использования логических автоматов или микропроцессорных схем интерфейсные модули обычно реализуют следующие основные подпрограммы работы:

1. Инициализация системы КОП используется перед началом работы. Она проверяет работоспособность интерфейсного модуля и подключенных приборов.

2. Передача однолинейной интерфейсной команды, например, очистки интерфейса (ОИ), установки или снятия сигнала дистанционного управления (ДУ). Используется при конфигурировании измерительной системы в самом начале и в конце работы.

3. Передача универсальных интерфейсных команд, воспринимаемых всеми приборами, подключенными к шине. Информация по шине данных КОП передается в безадресном режиме.

4. Передача адресных команд по шине КОП необходима для адресного запуска или сброса устройства, передачи управления другому контроллеру, перевода прибора в режим управления с передней панели.

5. Передача информации по шине КОП в устройство с требуемым номером, т.е. в адресном режиме, необходима для программирования приборов. Перед вызовом подпрограммы прибору должен быть присвоен адрес приемника, а программной переменной – передаваемые данные.

6. Вывод информации по шине КОП из устройства необходим для считывания результатов измерения. Перед вызовом этой подпрограммы прибору должен быть присвоен адрес источника.

7. Вывод информации по шине КОП в безадресном режиме необходима для организации процедуры параллельного опроса.

8. Приостановка выполнения пользовательской программы до тех пор, пока в системе КОП не возникнет сигнал "запрос обслуживания" (ЗО). Она расширяет возможности организации измерений, делая их более гибкими.

9. Обеспечение реакции на сигнал ЗО позволяет в конце каждой подпрограммы обслуживания КОП осуществлять проверку наличия сигнала ЗО.

10. Опрос состояния устройства с требуемым номером, т.е. в адресном режиме, необходим для выявления прибора, запросившего внеочередное обслуживание по соответствующей линии КОП. Передаваемый с прибора байт содержит информацию о причине запроса. Перед вызовом этой подпрограммы прибору должен быть присвоен адрес источника.

11. Отмена реакции на сигнал ЗО необходима для выполнения пользовательской программы без опроса состояния линии ЗО.

Перечисленные выше подпрограммы могут дополняться более частными функциями управления шиной КОП.

### **Системные программные средства**

Почти каждая программа включает в себя следующие элементы:

1. Управление измерительными приборами, их режимами, конфигурацией и измерениями. Для этого необходимы приборные драйверы на языке программирования и знание кодов программирования.

2. Организация алгоритмов измерения. Большинство программ испытаний строятся по определенной логической схеме. В некоторых случаях в программе нужно остановиться. Нередко при испытаниях и контроле

осуществляется привязка к оси времени, например, при исследовании неустойчивости.

3. Обработка и представление информации. Анализ информации означает снятие данных с прибора и их анализ. Например, оценка среднеквадратичного отклонения показаний прибора или сравнение показаний прибора со значениями, являющимися границами допуска. Часто необходимо сохранить данные для последующего сравнения. Представление информации может быть в виде табличной распечатки результатов, а также в виде графиков.

4. Интерфейс пользователя может быть простым или сложным, использующим окна и меню. Большое количество времени тратится на создание удобного интерфейса пользователя.

Структура программного обеспечения измерительной системы и усредненный вес в процентах показан на рис. 3.1. Разработка собственной программы требует больших временных и материальных затрат. Ключом к уменьшению времени разработки является использование готовых унифицированных модулей программ.

Большинство программ экспериментальных исследований имеют одинаковые компоненты. Компьютер должен управлять приборами, получать от них информацию, анализировать ее, строить диаграммы, поддерживать связь с оператором и хранить результаты. Эти автономные сегменты можно многократно использовать путем создания библиотек подпрограмм и функций для пользователя.

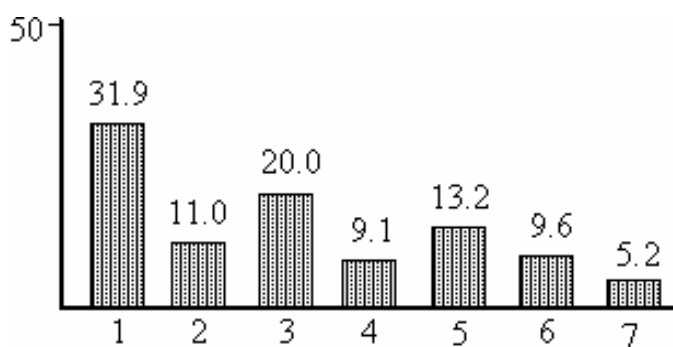


Рис. 3.1. Структура прикладного программного обеспечения: 1 – интерфейс пользователя; 2 – графические модули; 3 – алгоритмы измерений; 4 – приборные драйверы; 5 – анализ данных; 6 – управление данными; 7 – другие

При выборе языка программирования необходимо учитывать синтаксис и мобильность программных частей. Целесообразно выбрать язык, предполагающий высокую степень мобильности как между различными платформами контроллера, так и между модификациями данной платформы.

## 3.2. СТАНДАРТНЫЕ КОМАНДЫ, ФОРМАТЫ И ПРОТОКОЛЫ

В настоящее время список приборов с GPIB – портами включает в себя тысячи наименований. Однако уникальность команд, особенности программирования и представления данных для каждого прибора требуют создания специальных библиотек. В 1987 году был принят стандарт IEEE-488.2, устранивший несоответствия и проблемы совместимости, которые возникали у разработчиков систем при использовании приборов разных фирм. Спецификация IEEE-488.2 [9] регламентирует коды, форматы, протоколы обмена и общие команды, что необходимо пользователям при эксплуатации систем. Унифицированы не только форматы команд, посылаемых в приборы, но также форматы и способы кодирования результатов измерения в самих приборах. Новый стандарт уделяет также внимание возможности объединения в одну систему приборов, подчиняющихся спецификациям IEEE-488.1 и IEEE-488.2.

Стандарт также определяет перечень ошибок и процедуры обработки в случаях, когда прибор принимает многократные или незаконченные команды, а также прерывание в командах. Уточнен ряд требований и протоколов для контроллера, а также процедуры работы с линиями КП, ДУ, ЗО и ГП; упорядочено прохождение многократных сообщений. Стандарт определил 15 обязательных и четыре рекомендательных последовательностей управления.

Преимуществом использования всех нововведений становится более простое программирование и гибкость программы тестирования, хотя при этом предъявляются более жесткие требования к приборам-модулям и другим аппаратным средствам СКИМ.

### **Коды и форматы данных**

Стандарт IEEE-488.2 определяет форматы целых чисел, чисел с фиксированной точкой и чисел с плавающей точкой. В соответствии с требованиями стандарта IEEE-488.2 текстовые сообщения должны использовать 7-битный код ASCII, а двойные числа с плавающей точкой передаваться в формате стандарта IEEE-754.

Стандарт требует, чтобы новые устройства IEEE-488.2 были полностью совместимы с ранее выпущенными модулями IEEE-488.1 в широком диапазоне команд и форматов данных. С целью унификации форматов передаваемых данных введены новые понятия: “точный разговор“ («точная речь») и ”простительное слушание“ («снисходительное слушание»).



"Точный разговор" накладывает ограничения на формируемые приборами сообщения. В частности, показание прибора может быть передано только как число с плавающей точкой. Так, показание "1V" может быть передано в виде "+1.000E" или "+1.0E+0", но не как "1" или "1.0". "Точный разговор" упрощает программное обеспечение, так как для всех приборов необходимо только выделить плавающую точку.

При работе новых и старых приборов учитывается, что устройства, которые не соответствуют IEEE-488.2, могут формировать сообщения в других форматах. "Простительное слушание" в противоположность "точному разговору" предполагает корректную оценку показаний прибора, считывая их в произвольном формате: как целое, как величину с фиксированной точкой или как величину с плавающей точкой.

Существенно, что интерпретация формата данных возложена на прибор. Например, значение "1V" может быть выражено в виде "1" как целое, "1.00" – как величина с фиксированной точкой, "+1.00E+0" – как величина с плавающей точкой. Это требование освобождает от необходимости иметь точный формат данных. Тогда установка частоты генератора 24 Гц может быть выполнена любым из перечисленных ниже форматов:

Frequency 24.000; +24; 24; 2.4E+1; +2.43E+1.

Если генератор может формировать сигнал с дискретностью только 1 Гц, то он должен уметь одинаково хорошо принять все команды, округлив последний формат до 1 Гц, т.е. +2.4E+1. Округление всегда производится в соответствии с возможностями прибора.

### **Использование статусного регистра**

Многие операции, совершаемые по шине, являются общими для всех приборов, поэтому IEEE-488.2 уточняет программные команды, используемых для получения информации о состоянии устройства. Стандарт IEEE-488.2 предусматривает механизм адаптации процедуры взаимодействия с конкретным прибором в соответствии с его реальными возможностями, которые зафиксированы в статусном регистре. Так как IEEE-488.2 стандартизует формат отчета о состоянии, системный контроллер узнает, как именно объяснить информацию, пришедшую от любого прибора. Модель отчета о состоянии с более детальной расшифровкой построена на основе байта статуса IEEE-488.1. Формат информации, извлекаемой из статусного регистра, определен и предоставляет необходимые сведения об устройствах в системе, возможных ошибках, запросах и т.п.

## Общие команды

Стандарт задает формат команд, посылаемых приборам, а также формат и кодировку их ответов. Команды, как правило, являются аббревиатурами соответствующих слов английского языка. Установлено максимальное для команды число символов 12, но рекомендуемая длина – четыре символа. Команды-запросы, посылаемые приборам, предполагают обязательный ответ. Синтаксис запроса идентичен основной команде, в которую добавлен знак вопроса “?”.

Разработчики стандарта IEEE-488.2 обеспечили единство управления приборами с помощью набора общих команд. Некоторые из этих команд обязательны для всех приборов, некоторые необязательны, а другие обязательны только тогда, когда прибор имеет определенные особенности, например, способен отвечать на параллельный опрос шины. Все общие команды должны начинаться со звездочки. Стандарт запрещает аппаратно-зависимым командам использовать звездочку как заглавную букву. Список общих команд IEEE-488.2 приведен в табл. 3.1.

Таблица 3.1

Группа команд	Мнемоника	Описание команды (рус./англ.)	Обязательность
Автоконфигурации Auto configure	*AAD	Назначение адреса / Assign adress	Нет
	*DLF	Снятие функции приема / Disable listener function	Нет
Системные System data	*IDN?	Запрос идентификации / Identification query	Да
	*OPT?	Запрос возможности идентификации / Option-identification query	Нет
	*PUD	Защита информации пользователя / Protected user data	Нет
	*PUD?	Запрос защиты информации пользователя / Protected user data query	Нет
	*RDT	Передача описания ресурсов / Resource description transfer	Нет
	*RDT?	Запрос передачи описания ресурсов / Resource description transfer query	Нет
Внутренние Internal operation	*CAL?	Запрос калибровки / Calibration query	Нет
	*LRN?	Запрос обучения установки прибора / Learn device-setup query	Нет
	*RST	Сброс / Reset	Да
	*TST?	Запрос самопроверки / Self-test query	Да

Окончание табл. 3.1

Группа команд	Мнемоника	Описание команды (рус./англ.)	Обязательность
Синхронизации Synchronization	*OPC	Выполнение операции / Operation complete	Да
	*OPC?	Запрос выполнения операции / Operation-complete query	Да
	*WAI	Ожидание выполнения / Wait to complete	Да
Макрокоманды Macro commands	*DMC	Установка макро / Define macro	Нет
	*EMC	Разрешение макро / Enable macro	Нет
	*EMC?	Запрос разрешения макро / Enable-macro query	Нет
	*GMC?	Запрос получения содержания макро / Get-macro-contents query	Нет
	*LMC?	Запрос обучения макро / Learn-macro query	Нет
	*PMC	Очищение макро / Purge macros	Нет
Параллельного опроса Parallel poll	*IST?	Запрос индивидуального статуса / Individual-status query	Да (с ИФ ОП)
	*PRE	Разрешение параллельного опроса / Parallel-poll enable register	>>
	*PRE?	Запрос разрешения параллельного опроса / Parallel-poll enable-register query	>>
Статуса и события Status and event	*CLS	Очистка статуса / Clear status	Да
	*ESE	Разрешение установки статуса / Event-status enable	Да
	*ESE?	Запрос установки статуса / Event-status enable query	Да
	*ESR?	Запрос регистра установки статуса / Event-status register query	Да
	*PSC	Включение стирания / Power-on status clear	Нет
	PSC?	Запрос включения стирания / Power-on status-clear query	Нет
	*SRE	Разрешение функции запроса на обслуживание (ЗО) / Service request enable	Да
	*SRE?	Запрос разрешения функции ЗО / Service-request enable query	Да
	*STB?	Запрос чтения байта статуса / Read status-byte query	Да
Запуска Device trigger	*DDT	Задание аппаратного запуска / Define device trigger	Нет
	*DDT?	Запрос аппаратного запуска / Define device-trigger query	Нет
	*TRG	Запуск / Trigger	Да (с ЗП)
Контроллера Controller	*PCB	Возврат управления / Pass control back	Да с сист. контр.
Состояния Stored settings	*RCL	Вспомнить состояние устройства / Recall instrument state	Нет
	*SAV	Сохранить состояние устройства / Save instrument state	Нет

Общие команды не управляют измерениями, они стандартизируют возможности управления при калибровочном цикле, при работе самоконтролирующей программы и установке приборов в нужное состояние. Команды синхронизации позволяют контролировать последовательность операций в устройстве. Они позволяют активизировать несколько устройств одновременно.

Команды самоформирования позволяют контроллеру определять устройства на шине и назначать им адрес. Как известно, на задней панели оборудования IEEE-488 имеются переключатели для набора адреса. Используя команды самоформирования, система может формировать сама себя и автоматически подстраиваться к вновь подключенному оборудованию.

Команды статуса и результата позволяют создать тестирующие программы управления, используя запрос обслуживания модулей IEEE-488. Команды SRQ (service request) позволяют устройствам дифференцированно запрашивать обслуживание и формировать последовательные протоколы измерения. В стандарте IEEE-488.1 использование запроса обслуживания различно для каждого устройства. Стандарт IEEE-488.2 базируется на расширенных статус-моделях и группах команд статуса и результата, что позволяет стандартизировать эти сигналы. Для расшифровки информации стандартизованы и состояния устройств.

С течением времени все более увеличивается число устройств, которые совместно с приборами IEEE-488.2 позволяют тестовым программам стать универсальными. Когда все оборудование понимает общие команды, тогда становится возможным создать стандартные шаблоны управления большинством тестовых систем.

### **Унифицированные протоколы IEEE-488.2**

Унифицированные протоколы – это подпрограммы высокого уровня, которые объединяют ряд последовательностей управления для выполнения общих операций испытательной системы. Стандарт IEEE 488.2 определил два обязательных и шесть рекомендательных протоколов. Эти протоколы объединяют несколько команд и обеспечивают более быстрое выполнение наиболее общих операций, используемых в любой испытательной системе. Протоколы IEEE-488.2 уменьшают время разработки ПО. Они объединяют несколько команд для выполнения типовых операций, требуемых в любой тестирующей системе.

Стандарт IEEE-488.2 определяет два обязательных и шесть рекомендательных протоколов, как показано в табл. 3.2.

Таблица 3.2

№ п/п	Мнемоника	Наименование протокола	Требование
1	RESET	Перезапуск системы (сброс)	Обязательный
2	ALLSPOLL	Последовательный опрос устройств	>>
3	PASSCTL	Передача управления	Рекомендуемый
4	REQUESTCTL	Контроль запросов	>>
5	FINDRQS	Найти устройства с функцией 3	>>
6	FINDLSTN	Найти приемники	>>
7	SETADD	Установка адресов	Рекомендателен, но требует FINDLSTN
8	TESTSYS	Самотестирование системы	Рекомендуемый

Протокол RESET инициализирует GPIB и очищает все устройства так, чтобы они находились в исходном состоянии.

Протокол ALLSPOLL проводит последовательный опрос и анализ байта состояния каждого устройства.

Протоколы PASSCTL и REQUESTCTL передают управление шиной между активным и пассивным контроллером.

Имеется два конфигурационных протокола – FINDLSTN и FINDRQS. Протокол FINDLSTN используется для определения структуры адресов приемников на шине. Контроллер обеспечивает выполнение протокола FINDLSTN путем вывода адреса каждого приемника с дальнейшим анализом состояния ГП (свой-чужой адрес). Результатом протокола FINDLSTN является перечень всех адресов приема. Протокол FINDLSTN используется в начале работы для проверки правильности системной конфигурации и для уточнения действительного списка GPIB-устройств, которые могут быть использованы в системе испытаний. Кроме того, может быть получена информация о работоспособности модулей системы, т.е. для диагностики системы тестирования.

Когда конфигурация приемников уточнена, можно, используя протокол FINDRQS, оценить состав приборов, которые запрашивают обслуживание. При этом можно задать очередность списка приборов таким образом, чтобы более критичные приборы обслуживались первыми.

Протокол SETADD позволяет установить (назначить) адреса устройств, подключенных к системе, т.е. сконфигурировать систему программно.

Протокол TESTSYS позволяет выполнить самоконтроль модуля и сообщить контроллеру о своей готовности для выполнения операции или о возникших проблемах.

### **Требования к контроллерам IEEE-488.2**

Контроллер, удовлетворяющий требованиям стандарта, должен:

- формировать сигнал ОИ (IFC);
- устанавливать приборы в режим ДУ (REN);
- обнаруживать установку сигнала КП (EOI) при передаче источником сообщения последнего байта, а также управлять режимом параллельного опроса по выбранным линиям;
- посылать или принимать данные и общие команды IEEE-488.2 в установленных кодах и форматах с использованием стандартных протоколов;
- считывать состояние запроса на обслуживание (SRQ);
- анализировать байт-состояния;
- обнаруживать аварийные ситуации;
- обеспечивать требуемые задержки и скорости передачи команд и другой информации.

В дополнение к этим требованиям стандарта можно сформулировать преимущества, которые обеспечивают контроллеры IEEE-488.2:

1. Обеспечивая непрерывный мониторинг состояния шины IEEE-488, можно осуществлять контроль и диагностику модулей. Например, контроллер может определять, какие устройства активны, а какие слушающие.

2. Управляя индивидуально временами задержек, можно увеличить производительность прикладных программ за счет более быстрых обращений к высокоскоростным приборам.

3. Более гибкая обработка информации SRQ позволяет программно определять причины запроса и увеличить производительность системы.

### **Управляющие последовательности IEEE-488.2**

Управляющие последовательности (протоколы) делают приборы и контроллеры более совместимыми друг с другом. Поскольку исходный стандарт IEEE-488 не стандартизировал управляющие последовательности, индивидуальные разработчики создавали их сами. Это часто создавало проблемы совместимости. В табл. 3.3 показаны 15 обязательных и

4 рекомендательных управляющих последовательностей, определенных в стандарте IEEE 488.2.

Таблица 3.3

№ п/п	Последовательность обязательная	Описание
1	SEND COMMAND	Передача команд
2	SEND SETUP	Передача установочных данных
3	SEND DATA BYTES	Передача данных
4	SEND	Передача сообщения
5	RECEIVE SETUP	Прием установочных данных
6	RESEIVE RESPONSE MESSAGE	Прием данных
7	RECEIVE	Прием сообщения
8	SEND IFC	Подача команды ОИ
9	DEVICE CLEAR	Сброс устройства (СБ)
10	ENABLE LOCAL CONTROL	Установка в режим местного управления (МУ)
11	ENABLE REMOTE	Установка устройств в ДУ
12	SET RWLS	Установка устройств в ДУ с блокировкой МУ
13	SET LLO	Отключение МУ
14	READ STATUS BYTE	Чтение байта состояния IEEE-488.1
15	TRIGGER	Передача команды запуска (GET)
<b>Рекомендательная</b>		
1	PASS CONTROL	Передача управления
2	PERFORM PARALLEL POLL	Провести параллельный опрос
3	PARALLEL POLL CONFIGURE	Конфигурация устройств для параллельного опроса
4	PARALLEL POLL UNCONFIGURE	Деконфигурация параллельного опроса

### 3.3. КОМПЬЮТЕРНЫЕ ПЛАТЫ КОНТРОЛЛЕРА ШИНЫ IEEE-488 И ИХ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Интерфейсные платы обычно включаются в слот расширения ПК. Для большинства модулей допускается включение нескольких плат, как показано на рис. 3.2. Программный драйвер обеспечивает инициализацию и обмен информацией в соответствии со спецификацией IEEE-488. Поль-

зователь может связываться по шине, используя команды высокого уровня, например: REMOTE, LOCAL, ENTER, OUTPUT. Аппаратные и программные средства обрабатывают весь набор интерфейсных функций для приема, передачи и управления шиной GPIB.

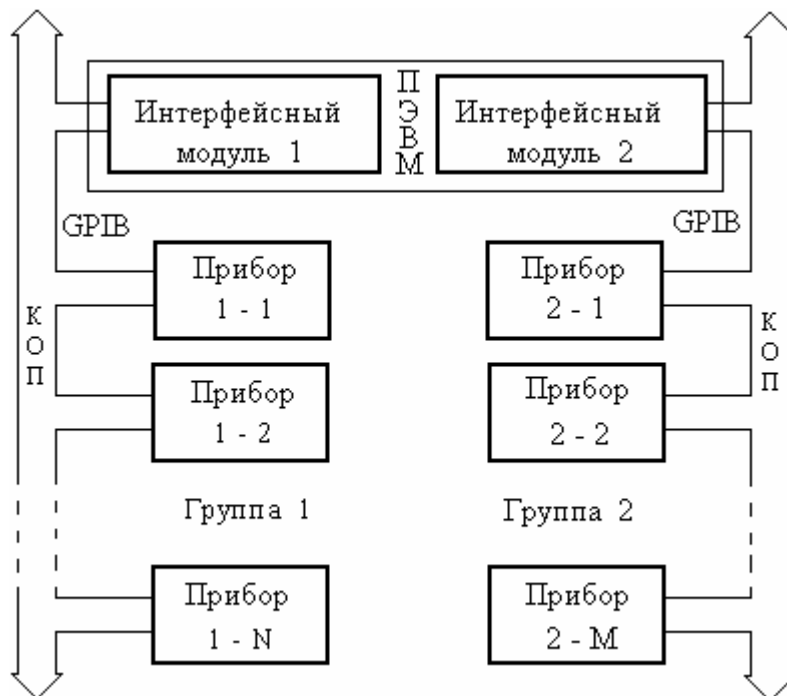


Рис. 3.2. Включение в ПК двух интерфейсных плат

Гибкость и уход от конфликтов с другими устройствами обеспечивается переключателем DIP. Это также дает возможность применения двух интерфейсных плат в одной ПК, как изображено на рис. 3.2. Пользователь может передавать данные между двумя группами устройств. Два контроллера обеспечивают обмен среди 30 различных устройств.

В настоящее время разными производителями выпускается большое число интерфейсных плат, возможности которых отличаются применяемыми чипами (БИС). В основном все они построены с использованием микросхем фирм NEC (NECmPD7210), Texas Instruments (TI9914) и National Instruments (NAT4882).

Программное обеспечение для простейших плат может размещаться на диске в виде резидентного драйвера, как это сделано для модуля МВС-488 фирмы Metra Byte. Другим вариантом является размещение программного обеспечения в ПЗУ непосредственно на плате сопряжения, как это сделано в модуле IE-488 этой же фирмы. Основные характеристики на-



званных интерфейсных модулей, построенных на NECmPD7210, представлены ниже:

- режимы прерывания и прямого доступа в память;
- скорость передачи информации до 450 Кбайт/с;
- поддержка до 15 устройств одновременно;
- совместимость со всеми высокоуровневыми языками;
- передача строки, слова и значения целого числа;
- функции контроллера C1....C5 для модуля IE-488.

Все команды определяются строкой ASCII, вызываемой оператором CALL в формате CALL IE-488 (COMMAND\$\$, VAR[\$], FLAG, BASADR), где: COMMAND\$\$ является командой универсальной, адресной или вторичной; VAR [\$] – входные или выходные данные, которые могут иметь вид целого числа или строки; FLAG характеризует состояние оператора CALL и при возникновении ошибки будет иметь шестнадцатеричный номер; BASADR – адрес платы интерфейса. Приведем перечень команд высокого уровня, облегчающих пользователю составление рабочей программы СКИМ:

#### 1. ABORT (АВАРИЙНОЕ ПРЕКРАЩЕНИЕ РАБОТЫ)

GRIB очищается и устанавливается в состояние по последней команде CONFIG. Эта команда подобна команде CLEAR. ПК должен быть активным контроллером. Формат команды: COMMAND\$ = "ABORT".

#### 2. CLEAR (ОЧИСТКА)

Очистка, или сброс, выбранных устройств или всех устройств. В безадресном режиме GPIB очищается и устанавливается в состояние по последней команде CONFIG. ПК должен быть активным контроллером. Формат команды при очистке устройств 10, 11 и 14: COMMAND\$ = "CLEAR 10, 11, 14".

#### 3. CONFIG (КОНФИГУРАЦИЯ)

Конфигурирует GPIB по запросу пользователя. GPIB будет оставаться в этом состоянии до реконфигурации. Формат команды для источника 6, приемников 12, 14 и ПК: COMMAND\$ = "CONFIG TALK=6, LISTEN=12, 14, MLA".

#### 4. ENTER (ВВОД)

Ввод данных из выбранного источника сообщений. ПК должен быть приемником сообщений. Формат команды при вводе из адреса 12 элементов массива от 0 до 18: COMMAND\$ = "ENTER 12, [\$,0,18)".

#### 5. EOI (КОНЕЦ ПЕРЕДАЧИ, ИДЕНТИФИКАЦИЯ)

Посылает EOI на выбранное устройство. Формат команды EOI при передаче информации приемнику: COMMAND\$ = "EOI 12 [\$]".

#### 6. LOCAL (МЕСТНОЕ УПРАВЛЕНИЕ)

Установка выбранных устройств в состояние местного управления. В безадресном режиме осуществляется установка всех устройств на шине. ПК должен быть активным контроллером. Формат команды установки устройств 10, 11: COMMAND\$ = "LOCAL 10,11".

#### 7. LOCKOUT (БЛОКИРОВКА)

Блокировка адресуемых устройств или (в безадресном режиме) всех устройств на шине GPIB. ПК должен быть активным контроллером. Формат команды для устройств 6 и 9: COMMAND\$ = "LOCKOUT 6,9".

#### 8. OUTPUT (ВЫВОД)

Выводит данные в выбранный приемник. ПК должен быть предварительно запрограммирован как источник сообщений. Формат команды вывода строки на устройства 10 и 12: COMMAND\$ = "OUTPUT 10,12 [E]".

#### 9. PASCTL (ПЕРЕМЕЩЕНИЕ КОНТРОЛЛЕРА)

Активный контроль GPIB перемещается по установленному адресу и ПК становится приемником или источником сообщений. По умолчанию IBM PC устанавливается в режим приема. Формат команды передачи управления устройству 6: COMMAND\$ = "PASCTL 6".

#### 10. PPCONF (КОНФИГУРАЦИЯ ПАРАЛЛЕЛЬНОГО ОПРОСА)

Установка на желательную параллельную конфигурацию. IBM PC должен быть активным контроллером. Формат команды параллельного опроса для устройства 14: COMMAND\$ = "PPCONG 14".

#### 11. PPUNCF (ДЕКОНФИГУРАЦИЯ ОПРОСА)

Сбрасывает конфигурацию параллельного опроса. ПК должен быть активным контроллером. Формат команды снятия параллельного опроса с устройства 14: COMMAND\$ = "PPUNCF 14".

#### 12. PARPOL (ПАРАЛЛЕЛЬНЫЙ ОПРОС)

Читает состояние шины данных GPIB, которое установилось для параллельного опроса. ПК должен быть активным контроллером. Формат команды параллельного опроса: COMMAND\$ = "PARPOL".

#### 13. REMOTE (ДИСТАНЦИОННЫЙ)

Устанавливает выбранные устройства в режим ДУ. ПК должен быть активным контроллером. Формат команды установки устройств 10,12 и 14 в дистанционный режим работы: COMMAND\$ = "REMOTE 10, 12, 14".

#### 14. REQUEST (ЗАПРОС)

Обслуживание запроса контроллера. Используется только тогда, когда ПК-неактивный контроллер. Формат команды запроса: COMMAND\$ = "REQUEST".

#### 15. STATUS (СОСТОЯНИЕ)

Последовательный опрос байта состояния выбранного устройства. IBM PC должен быть активным контроллером. Формат команды состояния устройства 12: COMMAND\$ = " STATUS 12".

#### 16. SYSCON (СИСТЕМНАЯ КОНФИГУРАЦИЯ)

Пользователь должен выполнить эту команду перед началом работы GPIB. Формат команды: COMMAND\$ = " SYSCON MAD=3, SIC=(0/1/2/3), NOB=(1/2), BAO = &Hdddd, ( BA1 = &Ndddd )", где указывается:

MAD = адрес IBM от 00 до 30 ( десятичные числа );

NOB = число IE488 плат ( 1 или 2 );

BA0 = базовый адрес для платы 1;

BA1 = базовый адрес для платы 2;

SIC = контроллеры шины (0 – нет, 1 – первый, 2 – второй, 3 – две отдельные GPIB-ветви ).

#### 17. RXCTL (ПОЛУЧЕНИЕ УПРАВЛЕНИЯ ШИНОЙ)

Команда RXCTL совместно с командой PASCTL позволяет управлять перемещением контроллера в прямом и обратном направлении. Формат команды: COMMAND\$ = " RXCTL".

#### 18. TIMEOUT (ОЖИДАНИЕ СОБЫТИЯ)

Устанавливает режим ожидания данных. Формат команды ожидания: COMMFND\$ = " TIMEOUT".

#### 19. TRIGGER (ЗАПУСК)

Посылает сигнал запуска в выбранные устройства. IBM PC должен быть активным контроллером. Формат команды запуска устройств 11 и 12: COMMAND\$ = " TRIGGER 11,12".

### **Изделия фирмы National Instruments**

Современные платы фирмы National Instruments полностью реализуют требования стандарта IEEE-488.2 благодаря использованию микроконтроллера NAT4882. Компания одна из первых разработала плату контроллера шины GPIB, которая полностью реализует требования стандарта IEEE-488.2. Новый однокристалльный прибор NAT4882 в сочетании с усовершенствованной программой позволил реализовать в контроллерной

плате AT-GPIB дополнительные возможности и режимы, которые нельзя было получить при использовании интегральных схем предыдущего поколения. Плата AT-GPIB, предназначенная для персонального компьютера PC/AT корпорации IBM, представляет собой 16-разрядный интерфейсный контроллер IEEE-488. Прибор NAT4882 реализует расширенный набор команд, позволяющих уменьшить программу-драйвер и размер кода, а также реализовать максимальную скорость передачи данных 1 Мбайт/с. Новый КМОП-прибор размещен в 68-контактном пластмассовом корпусе.

В 1987 году компания объявила о превышении стандартной скорости передачи (1Мбайт/с) по шине GPIB за счет использования Turbo488. Плата контроллера полностью совместима с программами-драйверами и прикладными программами, написанными для двух самых популярных однокристалльных контроллеров GPIB, приборов NECmPD7210 и TI9914 фирмы Texas Instruments. Когда плата работает с новой программой-драйвером NI-488.2 компании National Instruments, она автоматически реализует режим с расширенными возможностями.

В 1990 глду National Instruments выпустила модуль NAT4882, удовлетворяющий требованиям IEEE-488.2 и реализующий все функции передатчика, приемника и контроллера. Обеспечена программная совместимость с NECmPD7210 и TMS 9914A. Модуль NAT4882 имеет дополнительные регистры, которые осуществляют обязательные и рекомендуемые компоненты IEEE-488.2. Расширенный набор команд микроконтроллера NAT4882 предоставляет такие новые возможности, как увеличенный выбор условий прерывания, встроенные таймеры, а также выбор условий установления связи с задержкой. Возможности работы с наборами регистров приборов 7210 и 9914 остались неизменными, однако для прибора NAT4882 определены вспомогательные команды, рассчитанные на использование микроконтроллеров прежних выпусков. Эти команды обеспечивают страничный режим использования дополнительных регистров, доступ к скрытым регистрам или непосредственное управление.

Новый прибор позволяет в полной мере использовать функциональные возможности IEEE-488.2 в оборудовании с шиной GPIB. Кроме того, в отличие от прибора 7210 контроллер NAT4882 не выдает команд или данных на шину GPIB, когда принимающего абонента на шине нет, и обеспечивает управление и контроль для каждой линии шины.

В контроллере NAT4882 исключены проблемы, свойственные прибору 7210, и ограничения, налагаемые прибором 9914. Если прибор 7210 работает в режиме передающего абонента и в его выходной регистр записывается байт данных, а в это время на линию управления шины GPIB выдал свой сигнал другой контроллер, тогда этот байт данных будет теряться. С контроллером NAT4882 такого произойти не может.

Контроллер NAT4882 способствует повышению скоростей передачи данных благодаря трем факторам:

Во-первых, этот прибор может работать с тактовыми сигналами частотой 14 и 20 МГц против максимум 8 МГц для прибора 7210 и 5 МГц для прибора 9914.

Во-вторых, предусмотрен цикл автоматического переноса, который обеспечивает передачу последнего байта в режиме прямого доступа к памяти (ПДП). Если, например, по шине GPIB необходимо прочитать 10 байтов, то программа-драйвер обычно настраивается на чтение 9 байтов и ожидает сигнал EOI (End of Identify – конец или идентификация). Если такого сигнала нет, прибор завершает прямой доступ к памяти после обмена девятью байтами и задерживает установление связи до тех пор, пока не будет передан последний байт данных. Последний байт передается в режиме программного ввода-вывода. Цикл автоматического переноса NAT4882 позволяет передать все 10 байтов в режиме прямого доступа к памяти и автоматически обработать условия установления связи после завершения ПДП.

В-третьих, микроконтроллер может использовать минимально допустимую задержку T1 согласно спецификации IEEE-488 (350 нс), что позволяет выполнять операции записи с максимально возможной скоростью передачи данных по шине. Задержка T1 – это интервал времени между моментом, когда передающий абонент помещает данные на линии данных, и моментом, когда этот абонент может выдать сигнал на линию установления связи DAV (данные действительны).

Программа NI-488.2, в которой используются высокоуровневые функции, реализует протоколы шины GPIB и обеспечивает обмены в режиме ПДП с буферизацией прозрачным образом. Кроме того, эта программа-драйвер предоставляет пользователям возможность осуществлять настройку системы при помощи низкоуровневых функций любого распространенного языка, что обеспечивает максимальную гибкость. Прикладные

программы, написанные с ориентацией на старый драйвер NI-488, можно использовать без всяких модификаций.

В 1993 году появились специализированные чипы NAT7210, NAT9914 и TNT4882, предназначенные для IEEE-488.2. В дополнение к стандартным функциям TNT4882 позволяет осуществить протокол HS488 для высокоскоростной передачи данных до 8 Мбайт/с при использовании стандартных GPIB-кабелей. Максимальная скорость передачи зависит от компьютера и конфигурации системы. В 100-пиновом корпусе реализован высокоскоростной GPIB-интерфейс с функциями контроллера, передатчика и приемника. Для рабочих приложений TNT4882 сокращает программное обеспечение при стандартных (IEEE-488.1) и скоростных протоколах обмена данными.

Протокол HS488 повышает производительность GPIB-системы путем удаления задержек “3-проводной” IEEE-488.1 GPIB-системы. Протокол HS488 установлен на аппаратном уровне внутри чипа TNT4882 и не нуждается в изменении программного обеспечения. Реализация HS488 на уровне аппаратных средств позволила сохранить все прикладные программы, давая пользователям возможности гибкого применения новых и старых пакетов.

Шина PCI обеспечила единый стандарт для расширения компьютеров типа PC, Macintosh и Workstation. Плата PCMCIA-GPIB – функциональный аналог модуля AT-GPIB/TNT. Спецификация Plug & Play облегчает конфигурирование системы путем автоматического выбора каждого адреса, прерывания и DMA каналов аппаратных средств без какого-либо участия пользователя. Платы не имеют никаких DIP-переключателей, все установки полностью программно конфигурируемы операционной системой или пользователем.

В табл. 3.4 приведены данные о контроллерах GPIB фирмы National Instruments для наиболее распространенных компьютерных шин и портов.

Кроме этого National Instruments производит контроллеры GPIB для платформ таких компаний, как Macintosh, Hewlett-Packard, Sun и некоторых других, а также полный набор дополнительного оборудования (расширители, кабели и т. д.) для создания функционально законченной приборно-модульной системы.

Программное обеспечение NI-488.2 состоит из драйвера, который устанавливается как часть операционной системы. Оно включает в себя

утилиты, помогающие при разработке и отладке прикладных программ, содержит высокоуровневые программы и функции, которые автоматически осуществляют полное управление шиной, так что пользователю не обязательно знать детали программирования интерфейсной платы GPIB или протокола IEEE-488. Для максимальной гибкости и производительности прилагаются программы и функции низкого уровня. Программное обеспечение легко адаптируется к любой платформе, операционной системе и языкам программирования: Visual BASIC, C++, Visual C и др.

Таблица 3.4

Шина	Плата GPIB	Скорость передачи IEEE-488.1	Скорость передачи HS488
ISA	AT-GPIB/TNT	1,5 Мбайт/с	1,6 Мбайт/с (ISA) 3,4 Мбайт/с (EISA)
PCMCIA	PCMCIA-GPIB	1,5 Мбайт/с	2,3 Мбайт/с
PCI	PCI-GPIB	1,5 Мбайт/с	7,7 Мбайт/с
EISA	EISA-GPIB	1,5 Мбайт/с	7,5 Мбайт/с
PC Serial Port	GPIB-232CT-A	Зависит от параметров порта	-
PC Parallel Port	GPIB-1284CT	300 кбайт/сек (EPP)	-
PC Ethernet	GPIB-ENET	50 кбайт/с	-

Утилита конфигурации используется для редактирования средств аппаратного обеспечения, таких как базовый адрес ввода/вывода или канал DMA, и установки инструментов и плат, таких как первичные/вторичные адреса GPIB, вида конечного устройства (чтение/запись) и ограничения перерывов передачи данных. Обычно конфигурации инструментов выполняются программно, используя конфигурационные функции NI-488.2. Утилита интерактивного контроля является сильным приложением программного обеспечения и аварийным рабочим инструментом, который контролирует инструменты с помощью клавиатуры компьютера. Утилита проверки программного обеспечения NI-488.2 помогает различать приложения путем проверки, показа и записи всех вызовов программного обеспечения NI-488.2, сделанных приложениями. Широкий выбор интерфейсных библиотек языков помогает выполняющимся приложениям NI-488.2, используя популярные языки программирования. Утилита диагностики GPIB проверяет установку аппаратного и программного обеспечения NI-488.2.

Программное обеспечение NI-488.2 также предоставляет расширенные возможности настройки драйвера, что повышает гибкость разрабатываемых приборных систем. Все новые возможности и опции детально описаны в документации, поставляемой с пакетом NI-488.2. Существуют модификации NI-488.2 для работы под Windows 95, Windows NT, OS/2, Unix. Программное обеспечение NI-488.2 совместимо с такими программными продуктами фирмы National Instruments, как LabVIEW™, LabWindows™/CVI, Component Works™, Measure™. Имеются драйверы приборов различных производителей (более 500 приборов).

Программное обеспечение NI-488.2 полностью соответствует всем спецификациям IEEE-488.2:

- Командные байты не передаются, если нет устройств, подключенных к шине или включенных в сеть питания. Вместо этого выдается сообщение об ошибке ENOL (Error – No Listener – нет слушателя). При вызовах на уровне устройств NI-488.2 сообщает об ошибке EBUS (Error GPIB – ошибка шины).
- Программное обеспечение NI-488.2 выдает все команды GPIB согласно последовательностям, оговоренным в IEEE-488.2.

### **Шаблоны, утилиты и функции NI-488**

Шаблоны NI-488.2 обеспечивают выполнение типовых операций, последовательностей и протоколов, определенных в стандарте IEEE-488.2. В частности, они обеспечивают обмен данными с модулями, опрос и контроль состояния.

Утилиты используются для настройки СКИМ. Например, утилиту интерактивного контроля интерфейсной шины (IBIC) можно использовать как приложение при установке нового прибора и проверке его функций (чтение, запись, опрос и др.). В результате можно отследить сбои и определить местонахождение нефункционирующих приборов GPIB, а также оптимально установить все задержки, повысив скорость работы приложений.

Утилита проверки программного обеспечения (GPIB Spy) проверяет, записывает и отображает все обращения к палате и ПО. Это полезный инструмент для настройки и для подтверждения корректной работы всех приборов. Информация отображается в окне, она содержит наименование вызова и его параметры, информацию состояния и время. В дополнение к главному окну GPIB Spy можно получить детальную информацию по каж-



дому записанному вызову. Детальная информация отображает содержимое буферов, процесс выполняемых приложений и время начала и окончания вызова.

Высокоуровневые приборные функции автоматически регулируют коммуникационный протокол GPIB, необходимый для управления устройствами на шине. Низкоуровневые функции плат обеспечивают гибкость для регулирования необычных GPIB-ситуаций.

Программное обеспечение NI-488.2 имеет пять новых функций: IBCONFIG, IBSRQ, IBRDKEY, IBWRTKEY и IBLINES. Функция IBCONFIG используется для того, чтобы динамически конфигурировать все программные параметры драйвера, при этом отпадает необходимость использования утилиты конфигурации IBCONF. Функция IBSRQ используется для регистрации программы обработки прерываний, которая вызывается при утверждении SRQ. Функции IBRDKEY и IBWRTKEY используются для взаимодействия с платами интерфейса, которые оборудованы программируемыми аппаратными ключами защиты. Раньше аппаратные защитные ключи требовали программирование на уровне регистров. Функция IBLINES возвращает статус всех линий управления и данных GPIB. Это значительно упрощает мониторинг линий шины IEEE-488.

Структура функций:

1. Функции ввода/вывода GPIB обеспечивают чтение и запись.
2. Функции управления шиной обеспечивают системные действия.
3. Функции конфигурации устанавливают и возвращают в исходное состояние информацию о конфигурации.
4. Функции контроля устройств GPIB обеспечивают управление шиной и получение инструкций для устройств.
5. Низкоуровневые функции ввода/вывода передают более детальную информацию в нестандартных ситуациях.
6. Функции передающего/принимающего устройства используются в случаях, когда интерфейсный модуль не является контроллером.

### **3.4. СТАНДАРТНЫЕ КОМАНДЫ ПРОГРАММИРУЕМЫХ ПРИБОРОВ**

В 1990 году группа изготовителей приборов объявила спецификацию SCPI, которая определила принципы программирования и унифицированный набор команд измерений [10]. До SCPI каждый изготовитель исполь-

зовал собственные наборы команд для программирования приборов. Это затрудняло стандартизацию и вынуждало разработчиков испытательных систем изучать индивидуальные особенности программирования и специфические параметры всех приборов, используемых в прикладной программе. Определяя набор команд программирования, стандарт SCPI уменьшает время разработки, унифицирует программы тестов и способствует взаимозаменяемости приборов. Первая версия стандарта была выпущена в 1990 году.

Важным достоинством, ради которого был разработан стандартный язык программируемых приборов (SCPI), является возможность сделать максимально простым программирование прибора и осуществить высокую степень взаимозаменяемости.

На рис. 3.3 представлена схема условного разделения зон совместимости устройств в рамках стандартов IEEE-488.1, IEEE-488.2 и SCPI. Зона D представляет индивидуальные функции устройства, зона C – общие системные функции, B – функции передачи сообщений, A – представляет интерфейсные функции. Зоны D, C и B предполагают приборно-зависимые сообщения.

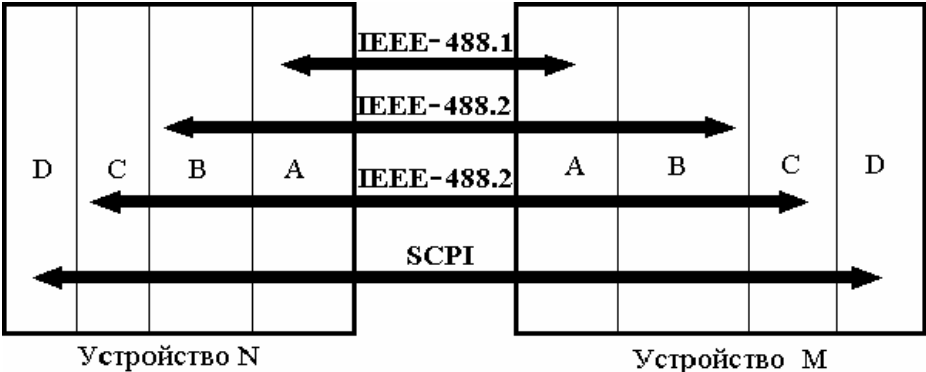


Рис. 3.3. Расширение зон унификации IEEE-488.1, IEEE-488.2 и SCPI

Язык SCPI многоуровневый. Высшим уровнем программного кода считаются SCPI-операторы MEASure. Этот уровень обеспечивает горизонтальную совместимость с использованием сигнально ориентированных команд. Они отражают суть задачи, которая ставится перед прибором. Таким образом, оператор MEASure для измерения частоты (FREQuency) может быть подан на осциллограф, счетчик или другой многофункциональный прибор, и требуемая функция будет реализована.

Более низкий уровень обеспечивает вертикальную совместимость приборов разных поколений, то есть их взаимозаменяемость. Самый низкий уровень не поддерживает никакой взаимозаменяемости. Он предполагает уникальные в своем роде возможности, которые нельзя реализовать другими приборами.

Стандарт имеет собственный набор общих команд, дополняющий IEEE-488.2. Он использует ключевые слова-команды и определяет способы их сокращения. Правила IEEE-488.2 по протоколам обмена сообщениями, а также форматы команд и способы кодирования параметров сохраняются. Можно использовать ключевые слова-команды в длинной или короткой форме. SCPI обеспечивает полный набор задач программирования, покрывающий все главные функции прибора. Этот стандартный набор команд гарантирует более высокую степень взаимозаменяемости приборов и минимизирует дополнительные усилия при включении новых приборов и расширении функций испытательных систем.

Набор команд иерархический, добавление команд для более специфических нужд или уникальных функций обеспечивается регламентированной процедурой развития SCPI. Каждый элемент функциональной модели измерительной системы имеет свой иерархический набор команд. Сигналы необходимо формировать, маршрутизировать и измерять, поэтому можно выделить модули источников, коммутаторов и приемников.

### **История и предпосылки создания SCPI**

В начале развития автоматизированных измерительных систем каждый разработчик инструментальных средств применял свои собственные команды программирования приборов с дистанционным управлением и свои варианты кодировки информации. В 1987 году стандарт IEEE-488.2 унифицировал для приборно-модульных систем коды, форматы, протоколы и общие команды. Новый стандарт уточнил взаимодействие модулей и форматы представления информации, однако команды по управлению приборами не были стандартизованы.

Для управления идентичными функциями приборов производители использовали различные команды. Например, для управления режимом работы универсального вольтметра при измерении постоянного тока могли посылаться команды: **VOLTDC**, **FUNCTION VOLTDC**, **DCVOLT** или **F1**. В то же время идентичные команды (мнемоники) могли иметь иные значения у приборов различных производителей и даже у различных приборов

одного производителя. Значение понятий “инициализация”, “запуск”, “загрузка”, “начало/остановка” также трактовалось неодинаково.

Ситуация усложнялась еще тем, что некоторые приборы содержат одинаковые функции, имеющие разный смысл. Например, осциллограф имеет функцию запуска, значение которой отличается от аналогичной функции вольтметра. Инструментальная терминология, как правило, развивалась в рамках некоторой профессиональной области. Поэтому параметры настроек приборов имеют индивидуальные названия или представлены неопределенной командой “Режим”.

Когда разные приборы (вольтметры, частотомеры, генераторы и т.д.) со своими индивидуальными мнемониками функций используются в одной и той же контрольно-измерительной системе, программист должен иметь глубокие знания не только обо всех индивидуальных командах и их значениях, но также о структуре и работе каждого прибора в отдельности. Он должен знать все тонкости управления каждым прибором, т.е. быть еще и метрологом.

Очевидно, что программирование в этом случае становится трудоемким и дорогим делом. И когда наконец после длительной коррекции и утомительного удаления из программы всех ошибок система измерений правильно заработает, пользователь не захочет с целью усовершенствования вносить в нее никакие изменения. Это также касается замены текущего прибора на более новый или лучший вариант.

При подобной модернизации команды управления должны быть заменены, и это может вызвать последствия для других частей программы, где изменения не планировались.

Решать вышеупомянутые проблемы можно только тогда, когда все приборы понимают и говорят на общем языке.

В 1985 году фирма Hewlett-Packard приступила к разработке стандартного системного языка программирования для испытаний и измерений – TMSL (Test and Measurement System Language). Язык определил универсальный набор команд управления приборов.

Фирма Hewlett-Packard начала с анализа списка мнемоник, используемых большинством приборов. Мнемоника – это короткое название или сокращение слова, применяемое как обозначение. Из всего списка мнемоник были отобраны наиболее общие сообщения для каждой группы приборов, которые предлагалось использовать в качестве набора взаимосвязанных команд программирования.

Каждый набор оптимизировался для управления отдельных видов приборов (вольтметрами, генераторами, частотомерами и т.д.). Однако виды приборов не являются абсолютно независимыми и имеют частичное перекрытие возможностей. Напряжение переменного тока может быть измерено вольтметром, анализатором спектра, осциллографом, измерителем нелинейных искажений и т.д. Это и стало главной причиной поиска, выработки общего подхода и последующего создания стандартного языка программирования для всех измерительных приборов.

Очевидная эффективность использования TMSL способствовала образованию консорциума по стандартным командам для программируемых приборов (SCPI – Standard Commands for Programmable Instruments). В апреле 1990 года консорциум предложил TMSL как базис для SCPI-стандарта и добавил к нему модифицированную версию общего формата для аналоговой информации компании Tektronix.

До SCPI каждый изготовитель использовал собственные наборы команд для программируемых приборов. Это затрудняло стандартизацию и вынуждало разработчиков испытательных систем изучать индивидуальные особенности программирования и специфические параметры различных приборов, используемых в прикладной программе.

Основная цель SCPI – упростить программирование контрольно-измерительной системы. Когда основные понятия и структура команд SCPI пользователю известны, ему будет довольно просто написать или изменить управляющую программу для индивидуальной системы. При этом время написания программы для новых или других приборов будет уменьшено не только благодаря тому, что схожие функции приборов управляются теми же командами, но также и из-за понятности и простоты команд SCPI.

Спецификация SCPI разработана таким образом, чтобы было полное соответствие стандарту GPIB. Однако, SCPI можно использовать и в других физических интерфейсах без требования совместимости со стандартом GPIB. Примерами таких интерфейсов являются RS-232 и VXI.

### **Совместимость программных и аппаратных средств**

Основной целью стандарта SCPI является обеспечение унифицированного управления идентичными функциями. Это позволяет сделать программирование прибора максимально простым и осуществить высокую степень взаимозаменяемости. Общепринятым считается, что два прибора

будут совместимыми только тогда, когда они будут подчиняться одинаковым командам и иметь одинаковые возможности. Отправной точкой является то, что приборы SCPI должны использовать одинаковые команды для управления схожими функциями приборов независимо от изготовителя, от типа или класса прибора. Дополнительно унифицируются требования к форматам данных, используемых в сообщениях, передаваемых прибором контроллеру в качестве ответа на команду запроса.

Тем не менее для решения частных измерительных задач приборы могут иметь уникальные возможности, которые управляются индивидуальными (несовместимыми) командами. Таким образом, для прибора индивидуальные особенности применения являются подмножеством общих возможностей, и такие приборы будут совместимыми. Число приборов в системе, использующих специализированные возможности, должно быть минимальным.

SCPI обеспечивает два типа совместимости программных средств:

*Вертикальная совместимость* означает, что все приборы одного типа должны иметь идентичное командное управление. Например, все осциллографы будут использовать те же самые команды управления для установки развертки, чувствительности и вида запуска. Вертикальная совместимость приборов разных поколений обеспечивает их взаимозаменяемость.

*Горизонтальная совместимость* означает, что приборы различных видов, выполняющие идентичные функции, должны использовать те же самые сигнально ориентированные команды независимо от применяемых методов измерений. Например, частотомер и осциллограф должны измерять интервал времени той же самой командой.

Команды SCPI используются для программирования приборов. Однако SCPI – не язык программирования типа Бейсика, Паскаля или Фортрана. Команды на этих языках используются для того, чтобы быть понятными компьютеру, а команды SCPI предназначены для управления приборами. Они направляются в порт ввода-вывода и понимаются только приборами. Компьютер лишь пересылает командное сообщение прибору на языке программирования. Содержание этого сообщения компьютеру неважно.

### **Сигнальный и интерфейсный статус прибора SCPI**

Команды SCPI – это стандартные сообщения для дистанционного управления программируемыми измерительными приборами, посылаемые

контроллером (обычно ПК) прибору. Контроллер механически и электрически связан с прибором через "интерфейс", называемый интерфейсной шиной. Совокупность приборов, контроллера и интерфейса называется контрольно-измерительной системой. Интерфейсная шина, соединяющая приборы-модули с ПК обычно представляет собой многожильный кабель (шина GPIB).

Стандарт SCPI содержит список команд для программных сообщений и ответов приборов. Эти команды точно определяют синтаксис (правописание) и семантику (значение) каждого SCPI-сообщения. Функции приборов, которые обеспечиваются стандартом, могут управляться только командами SCPI. Это, однако, не означает, что изготовители приборов ограничены функциями, определенными стандартом. Такое ограничение было бы недопустимо из-за увеличения степени интеграции и изменения свойств устройств, параметры которых нужно измерить. Новые команды непрерывно добавляются к стандарту.

В соответствии со стандартом IEEE-488 все устройства, осуществляющие взаимодействие по шине GPIB, делятся на четыре основные группы А, В, С и D, отличающиеся своим интерфейсным статусом (рис. 3.4):

controller – устройство управления обменом по шине;

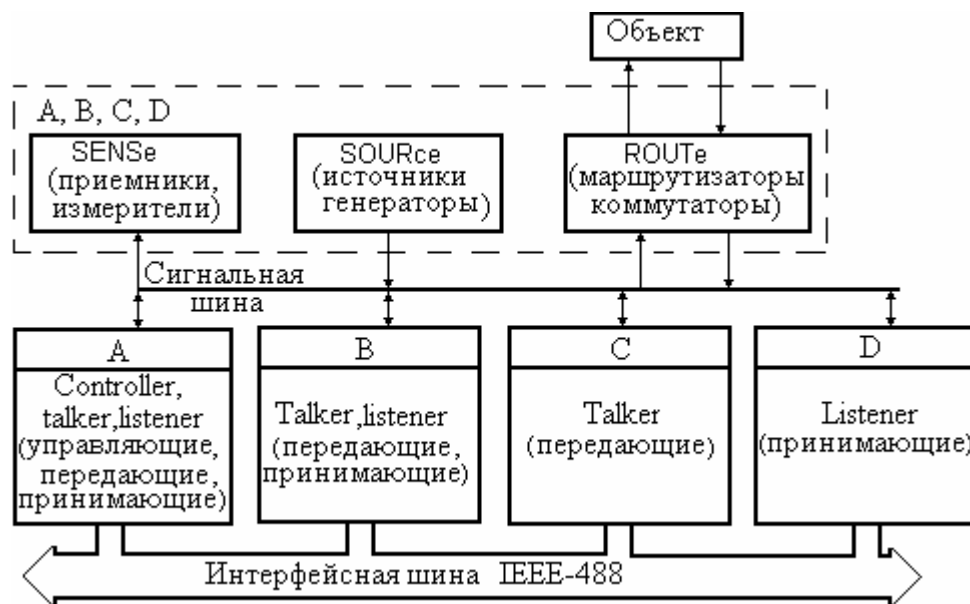


Рис. 3.4. Функциональная схема сигнального и интерфейсного статуса приборов-модулей, подключаемых к GPIB

talker – "говорящее устройство", являющееся источником дистанционных сообщений, передаваемых по шине;

listener – "слушающее устройство", являющееся приемником дистанционных сообщений, передаваемых по шине.

Статус устройств измерительной системы, исходя из сигнальных функций, хотя и зависит от интерфейсного статуса, но зависимость эта не является жесткой. Например, многие генераторы испытательных сигналов являются "слушающими" устройствами, но среди них есть и "говорящие". Мультиметры, используемые как универсальные измерители, обычно являются "слушающими" и "говорящими", но могут выполнять некоторые функции контроллера.

Стандартные команды **SCPI** ориентированы на управление сигнальным статусом приборов. Виды модулей в соответствии с тремя основными категориями сигнального статуса представлены в табл. 3.5.

Таблица 3.5

Функция	Тип прибора	Примеры
Прием сигнала (SENSe)	Измеритель, приемник	Вольтметр, осциллограф, частотомер и другие приборы
Генерация сигнала (SOURce)	Генератор, источник	Генератор импульсов, источник питания
Маршрутизация сигнала (ROUTe)	Коммутатор, переключатель	Мультиплексор, коммутатор, переключатель

Стандарт **SCPI** дополняет набор общих команд IEEE-488.2. Он использует ключевые слова-команды и определяет способы их сокращения. Правила IEEE-488.2 по протоколам обмена сообщений, а также форматы команд и способы кодирования сохраняются. **SCPI** обеспечивает полный набор задач программирования, покрывающий все главные функции прибора. Набор команд иерархический, поэтому добавление команд для более специфических нужд или уникальных функциональных возможностей легко обеспечивается регламентированной процедурой развития **SCPI**.

### **Функциональная модель модуля SCPI**

Чтобы правильно выбрать команды **SCPI** для управления функциональными возможностями, используют обобщенную модель программируемого прибора или системы, которая показана на рис. 3.5. Каждый элемент функциональной схемы имеет свой иерархический набор команд. Сигналы необходимо формировать, маршрутизировать и измерять, поэтому можно выделить модули источников, коммутаторов и приемников.



Очевидно, что большинство приборов не обладают всеми функциональными возможностями. Например, прибору, принимающему сигнал, не нужен компонент генерации и не обязателен компонент маршрутизации (ROUTE) сигнала. Другой прибор, такой как мультиплексор, содержал бы только подсистему ROUTE.

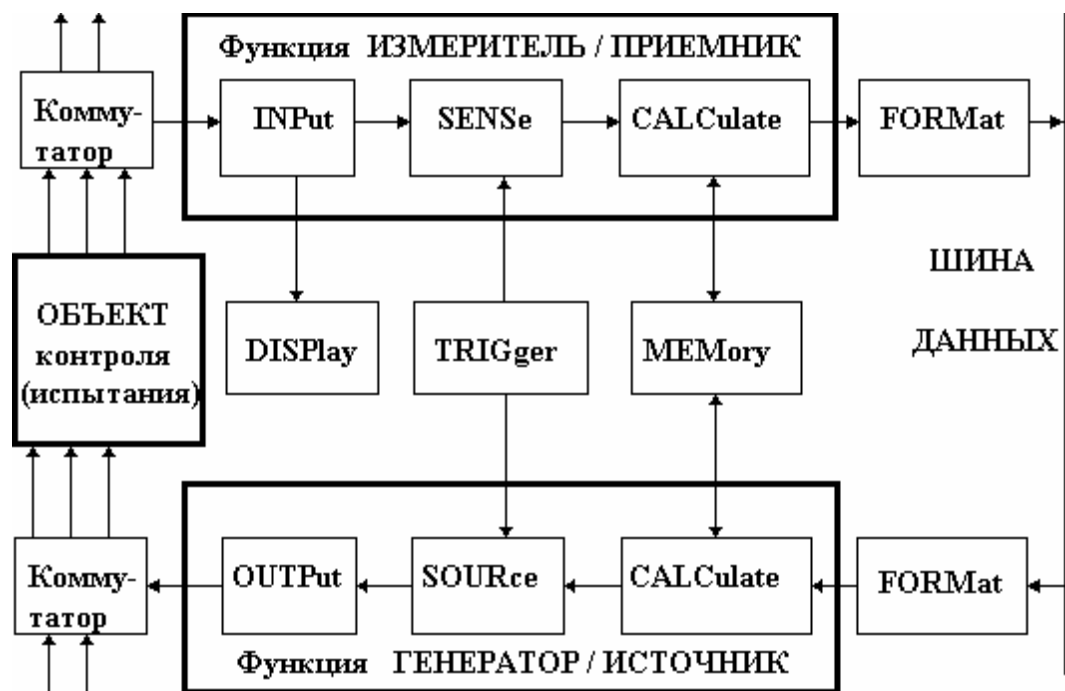


Рис. 3.5. Функциональная модель прибора или системы измерения

Основная задача измерителя – сбор данных. Его модель содержит три модуля: INPut (ввод), SENSe (приемник, измеритель...) и CALCulate (вычислитель). Функция генерации сигналов содержит модули OUTPut (вывод), SOURce (источник) и CALCulate (вычислитель). У многофункционального прибора могут быть общие модули. Модуль памяти (MEMory) сохраняет данные внутри прибора. Модуль формата (FORMat) преобразует приборные данные в форму, которую можно передавать по стандартной шине. Модуль запуска (TRIGger) синхронизирует все действия с внутренними и внешними событиями.

Компонент ИЗМЕРИТЕЛЬ разделен на три основные части: INPut, SENSe и CALCulate. Функциональный узел INPut является входным модулем, создающим необходимые условия (например, усиление или согласование) для входного сигнала прежде, чем он будет преобразован в данные блоком SENSe. Функции INPut включают также фильтрацию, преобразование и ослабление. Модуль SENSe преобразует сигналы во внутренние

данные, которыми можно управлять. Функции SENSE управляют такими параметрами, как диапазон, разрешающая способность, время счета и режим работы. Модуль CALCulate преобразует полученные данные в формат, необходимый для прикладной программы. Функции CALCulate включают процедуры вычисления, усреднения и преобразования данных.

Компонент ГЕНЕРАТОР предназначен для формирования измерительного сигнала. Он обеспечивает преобразование программных данных в выходные физические сигналы – аналоговые или цифровые. Генератор разделен на три функциональных модуля: OUTPUT, SOURCE и CALCulate. Компонент OUTPUT создает нормированные условия для выходного сигнала. Функции OUTPUT включают фильтрацию, преобразование и ослабление. Компонент SOURCE генерирует сигналы в соответствии с требованиями программных внутренних данных. Функции модуля SOURCE задают такие параметры сигнала, как вид и глубина модуляции, мощность или напряжение, частота, число каналов выхода и др. Компонент CALCulate преобразовывает программные данные во внутренние сигналы управления источником.

### **Основные особенности SCPI**

Основные команды SCPI общепонятны, что достигается выбором простых мнемоник и их иерархической организацией. В высокоуровневых языках команды состоят из ключевых слов, разделенных специальными знаками. Любой, кто уже имел навыки работы с современными файловыми структурами, легко адаптируется к иерархической структуре SCPI.

#### *1. Многоуровневость SCPI*

Многоуровневую структуру SCPI относят к схеме типа "дерево". Связанные по назначению команды группируются вместе в вершине дерева подобно тому, как листья одного уровня соединены с одной ветвью. Тонкие высокие ветви, образовавшиеся у корня дерева и присоединенные к телу дерева, считаются высшими в иерархии. Путь от корня к листьям описывают команды SCPI.

#### *2. Ключевые слова*

Каждое ключевое слово длиннее, чем четыре знака дается в укороченном варианте. Обычно в документации по SCPI командам придается следующая форма записи: ключевое слово дается в его длинной форме, с маленькими буквами после четырех больших букв. Обычно команды пишутся в их полной форме от корня к листьям.

Например: `SENSe:FREQuency:STARt 10 MHz`

Эта команда используется для установки начальной частоты измерителя на величину 10 МГц. Можно использовать ключевые слова-команды в длинной или короткой форме. Допустимые эквиваленты команды:

`sense:frequency:start 10 MHz`

`sens:freq:star 10 MHz`

В примере команда показана с конкретным параметром 10 МГц. Однако команды SCPI могут работать с диапазоном значений. В SCPI числовые значения описываются знаком `<numeric_value>`. Маркер `<numeric_value>` допускает применение также обозначений: `MAXimum` и `MINimum`. Ниже приведена команда установки начальной частоты к самому нижнему разрешенному пределу:

`SENSe:FREQuency MINimum`

При этом необходимо помнить, что текущее минимальное значение не всегда является абсолютным минимумом. Оно зависит от режима работы прибора.

Аналогично, используя `MAXimum`, можно установить начальную частоту на максимально допустимое в данном режиме работы прибора значение.

### *3. Запросная форма записи*

В SCPI команды имеют возможность записи в эквивалентной запросной форме так, чтобы пользователь имел возможность запросить или вернуть данные в исходную точку. Форма записи запроса такая же, как и командная, только добавляется знак вопроса (?). Запросная форма при выполнении измерений говорит прибору о необходимости передать результат в компьютер. Запросная форма команды установки начальной частоты имеет вид:

`SENSe:FREQuency:STARt?`

Если начальная частота ранее была 10 МГц, то запрос должен вернуть ее в исходное состояние. Задачей запросной формы является возврат ранее введенных данных, а в случае использования команд с `<numeric_value>` – возврат в конкретную точку. Однако запрос не возвращает единицы измерения.

Другим полезным свойством команд в запросной форме является облегченный механизм поиска доступного диапазона значений без изменения режима работы:

SENSe:FREQuency:STARt? MINimum

SENSe:FREQuency:STARt? MAXimum

Маркер `<numeric_value>` позволяет употреблять в командах и другие обозначения, используемые в приборах, такие как положительная бесконечность (INFINITY) и отрицательная бесконечность (NINFINITY).

#### *4. Автоматическая настройка прибора*

Диапазон и режимы работы зависят от функциональных возможностей выбранного прибора. Например, вольтметр может иметь шкалы измерения напряжения 1 В, 10 В и 100 В. Для обеспечения измерительной совместимости значения, задаваемые `<numeric_value>` и находящиеся внутри диапазона прибора, но не соответствующие фактическому значению в приборе, округляются. Округление производится в зависимости от прибора, команды и того, как близко стоит ближайшее фактическое значение. Если взять, к примеру, диапазон вольтметра, то запрос на значение 2 В подразумевает, что пользователю необходимо измерять сигналы до 2 В и выбранный прибор должен избрать шкалу 10 В. В случае выбора частотного значения или любого другого, наиболее вероятным действием будет округление до ближайшего разрешенного значения.

Выбор значения, которое выходит за рамки возможностей прибора, заставляет его генерировать ошибку "Выход из диапазона". Однако SCPI предусматривает округления, когда значение параметра лишь немного выходит за рамки. Например, в генераторах с диапазоном частот от 10 Гц до 20 кГц и четырьмя разрядами шкалы на запрос 50 кГц прибор выдаст ошибку "Выход из диапазона". В то же время на запрос 20,005 кГц может установить значение 20 кГц, так как дискрет по оси частот в точке установки больше 0,005 кГц.

#### *5. Числовые и символьные переменные*

Кроме числовых значений и обозначений типа MAXimum и MINimum, описываемых знаком `<numeric_value>`, в SCPI могут использоваться различные символьные переменные `<Boolean>` во многих, преимущественно управляющих командах. Команда, содержащая в своем описании `<Boolean>`, подразумевает состояния ON (включено) и OFF (выключено). Запрос на команду с параметром `<Boolean>` возвращает состояние 1 или 0.

Некоторые команды имеют подрежимы, которые описываются в символьном виде. Например, в строке `SOURce:FREQuency:MODE` символ `MODE` (вид колебания) предполагает режимы качания частоты (`SWEep`) или непрерывных колебаний (`CW-continious wave` или `FIXed`).

#### *6. Табличная форма записи (в виде столбца)*

В командной таблице наивысший узел графа иерархически занимает наиболее левую позицию, в то время как более низкий по значимости узел пишется на одну позицию ниже и правее корневого узла. Например, в строке ниже `SENSe` имеется ключевое слово корневого уровня `FREQuency` как его подуровень. В свою очередь, `FREQuency` имеет свои собственные три подуровня:

```
SENSe
:FREQuency
[:CW|:FIXed] <numeric_value>
:STARt <numeric_value>
:STOP <numeric_value>
```

#### *7. Передача нескольких команд в одном сообщении*

Обычно программы пишутся только с одной командой в каждом сообщении, однако количество команд в одном сообщении не ограничивается стандартом и может быть достаточно большим:

```
SENSe
:FREQuency
:STARt <numeric_value>
:STOP <numeric_value>
```

Здесь `STARt` и `STOP` – вершины дерева, имеющие одну родительскую вершину `FREQuency`. Поэтому эти команды можно записать:

```
SENSe:FREQuency:STARt 10 Mhz <PMT>
SENSe:FREQuency:STOP 50 Mhz <PMT>
```

Здесь символ `<PMT>` обозначает конец сообщения. Альтернативно эти команды могут быть включены в одно сообщение, поскольку для каждого набора посылаемых сообщений контроллер должен адресоваться к этому прибору. Преимущество передачи нескольких команд в одном сообщении в том, что указанная выше адресация прибора сохраняется. Приведенный выше пример с одним сообщением будет иметь вид:

```
SENSe:FREQuency:STARt 10 MHz;
:SENSe:FREQuency:STOP 50 MHz <PMT>
```

Команды в одном сообщении отделяются точкой с запятой. В начале каждого сообщения прибор ожидает указания команды от корня к ветвям. После точки с запятой добавляется двоеточие к началу следующей команды. Это означает, что команда также идет от корня. В случае, когда двоеточие опущено, команда идет от родительской вершины дерева последней команды. Этот механизм обеспечивает существенное сокращение времени написания программы и наш пример преобразуется в следующий вид:

```
SENSe:FREQuency:STARt 10 MHz;  
STOP 50 MHz <PMT>
```

#### *8. Допустимые варианты записи*

Внутри квадратных скобок приведенного выше примера есть два ключевых слова, разделенных вертикальной чертой. Эта черта свидетельствует о наличии выбора (ИЛИ) между ключевыми словами CW и FIXed. Таким образом, одну и ту же команду можно записать в нескольких вариантах:

```
SENSe:FREQuency [:CW|:FIXed]  
SENSe:FREQuency:CW 10 MHz  
SENSe:FREQuency:FIXed 10 MHz  
SENSe:FREQuency 10 MHz
```

#### *9. Режимы работы по умолчанию*

Один из вариантов подуровня FREQuency заключен в квадратные скобки. Такое изображение обозначает режим, принимающий значение по умолчанию. Язык SCPI часто использует режим "по умолчанию" с целью сделать команду как можно более простой. Когда прибор преимущественно источник или коммутирующее устройство, то по умолчанию подразумевается переход к корневым узлам SOURce или ROUTe. Таким образом, для генератора команда о смене фиксированной частоты выглядит следующим образом:

```
FREQuency 30 MHz
```

Режимы по умолчанию играют более существенную роль, чем упрощение неоднократно повторяющихся команд. Механизм режима по умолчанию – это ключ к расширению языка SCPI. Нижняя таблица показывает, как выглядит набор SCPI-команд для задания состояния и частоты фильтрации входного устройства:

```
NPut  
:FILTer
```

```
[:STATe] <Boolean>  
:FREQuency <numeric_value>
```

Пусть первоначально была задана фильтрация по низким частотам, но необходимо управлять также верхними частотами. Это достигается путем ввода текущей низкочастотной фильтрации в режим "по умолчанию" (Low PASs) и добавления нового графа (High PASs) для задания высокочастотных характеристик, как показано ниже:

```
INPut  
:FILTer [:LPASs]  
[:STATe] <Boolean>  
:FREQuency <numeric_value>  
:HPASs  
[:STATe] <Boolean>  
:FREQuency <numeric_value>
```

#### *10. Вертикальная совместимость приборов*

Программа, предназначенная для работы с приборами, имеющими устаревшую версию языка, будет работать и с новыми приборами, поддерживающими расширенный командный язык. SCPI поддерживает механизм обеспечения ограниченной обратной совместимости, когда необходимо использовать устаревший прибор с ранней версией языка. Однако если при написании прикладной программы основывались на новой версии языка и некоторые новые команды находятся в режиме "по умолчанию", то очевидно, что устаревший прибор не сможет выполнить новый граф. Для обеспечения обратной совместимости лучше исключить режимы "по умолчанию" из всех активных команд, посылаемых на старый прибор.

#### *11. Горизонтальная совместимость приборов*

Высшим уровнем программного кода считаются команды MEASure. Это самый высокий уровень горизонтальной совместимости приборов, базирующийся на использовании сигнально ориентированных команд, которые отражают суть задачи, стоящей перед прибором. Таким образом, команда MEASure для измерения частоты (FREQuency) может быть подана на осциллограф, счетчик или другой многофункциональный прибор, и требуемая функция будет реализована.

#### *12. Гибкость стандарта*

Успех внедрения SCPI основан на возможности пользователей совершенствовать стандарт и включать в него различные полезные нововве-

дения. Те стандарты, в которых возможности модернизации ограничены, не выдерживают жесткой конкуренции на мировом рынке сбыта. Функции приборов очень быстро развиваются и, конечно же, необходимо, чтобы стандарт не тормозил их развития.

Язык SCPI непрерывно пополняется разработками разных компаний. Предложения по расширению SCPI ставятся на рассмотрение рабочей группы консорциума. Этот орган решает, надо ли утвердить это предложение или нет, чтобы не нарушалась связь между новыми командами и общей стратегией SCPI. Если нововведения принимаются, то они становятся частью ежегодно распространяемого обновленного стандарта.

Пересмотренная версия стандарта содержит наименование года. Ежегодные версии сопровождаются нулем через точку. Например, стандарт в своем первом издании есть SCPI 1990.0. Если принимаются новые команды, то десятичная частица увеличивается. Первое изменение относится к SCPI 1990.1.

SCPI позволяет изготовителям приборов по-своему определять те функции, которые отсутствуют в стандарте. SCPI требует неукоснительного соблюдения только тех команд, которые утверждены. Используемые команды, не оговариваемые стандартом, называются "неутвержденными".

Прибор, в чей набор функций включены такие, которые описываются "неутвержденными" командами, не может быть полностью совместимым с новыми версиями стандарта. Добавление новых команд к стандарту запрещает прибору иметь полную принадлежность к новым версиям. Это проверяется командным запросом:

SYSTem:VERSion?

При дальнейшей эволюции конкретного прибора производителям аппаратуры необходимо пересматривать, какие команды являются стандартными, а какие "неутвержденными". При точном подборе прибора под какую-то конкретную задачу наиболее предусмотрительно использовать последнюю версию стандарта. Этот пересмотр дает возможность определить, нужны ли для его измерительной аппаратуры какие-либо "неутвержденные" команды. В результате сохраняется баланс между нововведениями и стандартизацией.

Таким образом, SCPI – это язык программирования, который по мере развития расширяется. Несмотря на большой набор уже имеющихся команд, SCPI предусматривает большие возможности по дополнению и изменению.



## Управление приборами (по функциональной модели)

Перечень команд с подробным описанием особенностей управления различными функциями прибора является основной частью стандарта SCPI. Для того чтобы программировать функции прибора, необходимо знать и понимать модель измерительного прибора. Команды управления функциональными возможностями, которые относятся к отдельному модулю, всегда начинаются с имени подсистемы. Например, аттенюатор входного усилителя обладает функциональными возможностями INPut. Команда управления будет иметь вид:

INPut:ATTenuation <значение>.

Обзор главных подсистем – команд, которые выполняют заданные действия с сигналом, представлен ниже. Прибор может поддерживать комбинацию команд.

### 1. ROUte (Маршрутизация – коммутация)

Подсистема обеспечивает функциональные возможности переключателя, направляя входящие сигналы на вход измерителя в модуль INPut. Может также использоваться в источниках, направляя сгенерированный сигнал к нужному выводу. В приборах, где главным является переключение сигналов, таких как коммутаторы и мультиплексоры, команда управляет маршрутом от порта к порту. Типичные функциональные возможности подсистемы ROUTe: отключение, подключение и просмотрисканирование.

### 2. INPut (Вход – ввод)

Управляет характеристиками входного порта датчиков. Важная часть функциональных возможностей, обеспечиваемых этой подсистемой, состоит в задании условий сигнала для его преобразования в блоке SENSE. Например, установка ослабления, входного сопротивления, фильтрации (ФНЧ и ФВЧ) и др. В подсистеме INPut могут программироваться также процедуры АЦП и ЦАП. Имеются команды для конфигурации механизма защиты входного порта.

### 3. SENSE (Считывание – измерение)

Подсистема выполняет сбор данных и преобразует входной сигнал во внутренние данные. Это одна из наиболее многосторонних подсистем SCPI. Существует много видов преобразований и большое число характеристик, которые могут быть измерены. Поэтому подсистема разделена на несколько подуровней команд, каждый из которых имеет дело с различным аспектом сбора и преобразования данных.

FUNCTION (Функция) и DATA (Данные) – наиболее важные подсистемы внутри SENSE. Они позволяют выбирать функции модуля и получать данные. Например, команды SENSE:FUNCTION "VOLTage" устанавливают модуль в режим определения напряжения сигнала. Команды SENSE:DATA? запрашивают текущий результат.

Внутри SENSE существуют подуровни, управляющие характеристиками диапазона. Например, чтобы установить диапазон измерения напряжения 1 В, используется команда SENSE:VOLTage:RANGe 1. Таким же образом могут быть установлены точность, ослабление, усиление, смещение и т.д. Модули, измеряющие мощность, частоту, температуру или что-либо другое, используют идентичные структуры для управления их параметрами и настройки команд.

Подсистема SENSE не ограничена измерениями сигналов. Она содержит несколько команд, которые управляют сбором массивов данных, как того требуют дискретизатор, осциллограф, измеритель АЧХ, анализатор спектра и т.д. Приборы, которые считывают массивы данных не как функции времени, но как функции других переменных могут также использовать команды из подсистемы SENSE.

Команда SENSE:SWEep:POINts 1000 устанавливает число (1000) отсчетных точек. Подсистема SENSE включает много команд для специальных методов преобразования и сбора данных. Группа команд улучшения метрологических и функциональных возможностей приборов связана с методами увеличения точности измерений путем усреднения или фильтрации.

#### 4. CALCulate (Вычисление)

Если сигнал преобразован в цифровой внутренний формат подсистемой SENSE, последующая обработка выполняется путем использования команд подсистемы CALCulate. Они могут использоваться для дополнительных цифровых преобразований, замены переменных, исправления внешних эффектов и т.д. Эти возможности можно использовать и до генерации сигнала подсистемой SOURCE.

Так как современные приборы обеспечивают функциональные возможности дополнительной обработки, то подсистема CALCulate содержит большое число команд. Имеются функции статистических вычислений, таких как AVERage (усреднение) и SMOothing (сглаживание). Поддерживается фильтрация по времени и частоте. Команда LIMit позволяет контролировать значение предела.

Дополнительная группа функций связана с математическими операциями, в том числе с вычислением дифференциалов и интегралов. Подсистема MATH позволяет пользователю (программисту) задавать выражения для математических действий над считанными или запомненными данными. Подсистема FORMat, включенная в подсистему CALCulate, выполняет простые вычисления с комплексными ( $X+jY$ ) или векторными переменными. Команды позволяют вычислять фазу, логарифмы и т.д.

Функциональные возможности, связанные с преобразованиями, обеспечиваются подсистемой TRANSform, входящей в подсистему CALCulate. Например, временной сигнал можно преобразовать в частотный спектр. Построение гистограмм по полученным данным включено в список команд.

Подсистема CALCulate содержит ряд команд, которые не только управляют вычислениями, но и позволяют устанавливать конфигурацию (последовательность) вычислений. Можно, например, сначала выполнить усреднение и затем обработать усредненные данные операциями MATH. Для этого используют команду установки пути CALCulate:PATH.

Вычисления обычно выполняются автоматически при поступлении новых данных, однако последовательность команд CALCulate:IMMEDIATE повторяет обработку уже имеющихся данных, без нового сбора. Это может потребоваться, если параметры настройки или ввода подсистемы изменились.

## 5. TRIGger (Запуск)

Команды подсистемы TRIGger используются для инициации всех видов действий приборов синхронно с внутренними или внешними событиями. Они базируются на гибкой модели синхронизации.

Модель описывается диаграммой изменения состояния, на которой отражаются реальные режимы запуска, например “жду команды” или “жду запуска”. Приход импульса на входном гнезде может быть событием, которое запускает сбор данных. Этот же импульс может вызвать переход от режима “жду команды” к режиму “жду запуска”, который реально инициируется по уровню входного сигнала.

Режимы, которые должны быть последовательно пройдены, до выполнения запуска устройства могут быть сгруппированы в виде шагов. Подсистема TRIGger поддерживает широкий диапазон событий для синхронизации действий устройств и управления прохождением через уровни

(изменения состояния). Например, могут быть выбраны VXI, (GPIB) BUS, EXTERNAL (внешний), линии ECLTrg, TTLTrg.

#### 6. DISPLAY (Представление)

Подсистема управляет функциональными возможностями, связанными с выбором и представлением различной информации: о данных измерений, о взаимодействии с пользователем, о состоянии приборов и контроллера. Команды от этой подсистемы поддерживают управление устройствами от простых индикаторных панелей до терминалов.

Подсистема DISPLAY обеспечивает управление контрастностью, яркостью, цветовой палитрой и т.д. Могут применяться сложные форматы представления, например, подсистема DISPLAY позволяет осуществить конфигурацию многих окон на экране, в которых отображаются графики, тексты. Когда окно используется контроллером, оно действует как терминал визуальной индикации.

Графики могут быть отображены в обычных XY или в полярных координатах. Имеются команды установки осей, сетки, меток и т.д.

#### 7. FORMAT (Форматирование)

Все SCPI-совместимые приборы должны поддерживать формат IEEE-488.2 для числовых данных. Иногда желательно использовать другие форматы, например IEEE-754 для чисел с плавающей точкой. Подсистема FORMAT поддерживает множество команд, которые обеспечивают выбор формата данных, посылаемых через внешний интерфейс.

#### 8. COPY (Копирование)

Подсистема управляет современными принтерами и графопостроителями. Она управляет выбором и представлением данных, подаваемых на графопостроитель или принтер, и не связана с конфигурацией интерфейса (принтера, графопостроителя), так как эти функциональные возможности поддерживаются командами SYSTEM:COMMUNICATE..

#### 9. SOURCE (Источник – генератор)

Подсистема содержит функциональные возможности цифро-аналогового преобразователя (ЦАП) и генерирует аналоговые сигналы, основанные на заложенных значениях цифровых данных. Таким образом, функциональные возможности подсистемы SOURCE противоположны возможностям подсистемы SENSE (Измеритель), которая включает аналого-цифровые функции.

Подсистема SOURce содержит команды для генерации различных сигналов. Значение термина "сигнал" не ограничено постоянным или изменяющимся напряжением, током или мощностью. В пределах контекста подсистемы SOURce количественные значения сопротивления и температуры также рассматриваются как "сигналы" наряду с другими возможными.

Современные источники могут управлять большим набором характеристик сигналов, поэтому подсистема SOURce стала почти такой же многосторонней, как подсистема SENSE. Обширные средства управления обеспечивают изменения характеристик сигналов во времени. Например, амплитуда переменного напряжения может изменяться как функция времени путем использования команды SWEEP (Развертка – качание) или может быть задана списком точек (амплитуды). Таким же образом может изменяться и частота сигнала.

Подсистема SOURce имеет средства управления модуляцией. Они доступны не только для амплитудной, частотной или фазовой модуляции, но и для всех цифровых методов модуляции и импульсной модуляции.

#### 10. OUTPUT (Выход – вывод)

Подобно тому как SOURce противоположно SENSE, функциональные возможности подсистемы OUTPUT противоположны функциональным возможностям INPUT (Вход). Основная задача подсистемы OUTPUT состоит в том, чтобы обеспечить необходимые характеристики выходного сигнала. В этой подсистеме содержатся типичные функциональные возможности, такие как ослабление, связь, фильтрация и т.д. Но также имеются команды для конфигурации механизма защиты выходного порта.

#### 11. TRACE (График – форма)

Подсистема включает команды для обработки и управления графиком как объектом, сохраненным в инструментальной памяти. Обычно график содержит ряд значений измеренных данных или результатов вычислений, а также определяемых пользователем значений данных для генерации сигналов.

Данные, сохраненные в графике, вызываются по имени, например, необходимая форма волны может быть названа My\_Wave. Данные могут быть запрошены TRACE? MY\_TRACE или DATA? MY\_TRACE.

Графики могут также использоваться в подсистеме CALCulate (Вычисления) в качестве входных данных для последующей обработки. Например, определяемый пользователем график может использоваться для

ссылки, служить верхним или нижним пределом в процессе вычислений LIMit (Предел).

## 12. MEMory (Память)

Подсистема управляет внутренней памятью прибора, в которой могут быть размещены следующие объекты: макрокоманды, таблицы, файлы и т.д. Подсистема MEMory содержит команды для копирования, очистки, удаления объектов, обмена и запросов о количестве данных, каталогах свободного пространства и т.д.

Подсистема MassMEMory (MMEMory) дает возможность получения памяти большой емкости для приборов, которые могут быть как внутренними, так и внешними. Подсистема обеспечивает те же возможности, что и обычная подсистема MEMory, а также возможности, которые обеспечиваются только в устройствах памяти большой емкости. Например, MMEMory поддерживает использование имен файлов, структур каталогов, идентификации запоминающих устройств по именам и т.д.

## 13. INSTrument (Прибор)

Автономный прибор может содержать множество частных приборов. Подсистема INSTrument обеспечивает механизм для идентификации и выбора отдельного логического прибора.

## 14. STATus (Состояние – статус)

В соответствии с общими командами стандарта IEEE-488.2 подсистема STATus поддерживает сообщения о состоянии, которые необходимы современной контрольно-измерительной аппаратуре. Выполнение подсистемы STATus является обязательным для всех SCPI-совместимых приборов.

Подсистема STATus базируется на стандарте IEEE-488.2, обеспечивая простой механизм, который учитывает скорректированные данные о состоянии, включая разнообразие условий и событий, таких как внутри-приборные ошибки. Восстановленная информация может быть детальной.

Чтобы уменьшить потери времени на контроль всех деталей, подсистема STATus позволяет пользователю объединять отдельные события вместе. Итог (бит) от каждой группы отражается в стандартном байте состояния. На следующем шаге пользователь может выбрать биты байта состояния, которые заставят сервисный механизм генерировать запрос на прерывание. Таким образом, пользователю дается гибкий механизм для выбора событий, которые, по его мнению, являются важными и нуждаются в немедленном запросе.

Имеются три важные группы состояний, которые должны поддерживаться SCPI-совместимым прибором. Первая группа – структура состояния, определенная стандартом IEEE-488.2. Внутри этой группы различают несколько различных типов сообщений об ошибках, а также некоторые общие события, такие как включение питания и окончание операции. Следующая обязательная группа называется сомнительным состоянием, которое дает индикацию качества измеряемого или генерируемого сигнала. Например, бит напряжения в этой группе может указывать, что уровень напряжения сигнала имеет сомнительное качество для точного измерения. Третья группа называется состоянием операции, которое дает информацию о состоянии нормальной работы устройства. Например, когда прибор ожидает запуска, когда он измеряет и т.д.

#### 15. SYSTem (Система)

Подсистема обеспечивает функциональные возможности, которые связаны с общими служебными действиями и установкой глобальных конфигураций, таких как время, дата и защита. К специальным относится запрос SYSTem:ERRor?, используемый для запроса сообщения об ошибках. Частью этой подсистемы является детализированная спецификация всех номеров ошибок и соответствующих текстов ошибок.

Важная группа команд доступна для конфигурации интерфейса с периферийными устройствами и удаленным контроллером. Поддерживаются интерфейсы CENTronics, GPIB и SERIAL RS 232. Для управления устройствами VXI и их функциональными возможностями определена отдельная подсистема.

Другая группа функций, о которой стоит упомянуть, связана с обеспечением пользователя информацией. При этом доступен запрос, который выводит список всех выполненных команд. Кроме того, можно восстановить точный синтаксис каждой выполненной команды.

#### **Управление измерениями (ориентация на задачу или сигнал)**

Для выполнения частной задачи измерения с использованием функциональной модели приборов должно быть выполнено множество последовательных шагов установки, таких как выбор функций измерения, диапазона и чувствительности, быстродействия, фильтрации, связи, входного импеданса, средств ручного управления и последующей обработки и т.д. Этот процесс трудоемкий, так как программист должен хорошо знать мо-

дели, инструментальные возможности и SCPI. Необходимо предусмотреть ошибки и сбои, которые могут произойти, например, когда не подается входной сигнал или когда уровень запуска в эксперименте не достигнут.

В дополнение к этому традиционному способу программирования по функциональной модели, который поддерживается SCPI, разработан и предусмотрен новый подход в управлении приборами: это измерения, ориентированные на задачу или сигнал. Они выполняются командами из группы команды измерений. Это наиболее высокий уровень программирования. Программист задает лишь характеристику сигнала, которую нужно измерить, и прибор автоматически выполнит необходимые измерения и сообщит результат. Когда измерение задано соответствующей командой, инструментальные параметры не нуждаются в программировании.

В зависимости от особенностей сигнала, внутренний инструментальный алгоритм определяет наиболее оптимальные параметры настройки прибора. При этом убирается традиционный и громоздкий процесс установки прибора. Например, когда сообщение MEASure:VOLTag:AC? посылается вольтметру, выполняется измерение переменного напряжения и результат автоматически посылается контроллеру. Сам прибор непосредственно выберет оптимальные установки для этой цели и выполнит требуемые измерения настолько хорошо, насколько возможно. Вопросительный знак в конце сообщения указывает запрос для возврата результатов контроллеру.

Пример программирования на измерение переменного напряжения для ожидаемого значения 20 В с точностью 0.001В:

```
MEASure:VOLTag:AC? 20.0.001
```

С командами измерения SCPI можно, не обладая знаниями о внутренних функциях приборов, быстро осуществлять нужные измерения, хотя в этом случае нет оптимального использования всех инструментальных возможностей. Если же возникает потребность уточнить детали, то в прикладную программу могут быть внесены необходимые добавления. Очевидно, что команды, осуществляющие использование частных возможностей прибора, требуют больших знаний.

Команда MEASure? (Измерение с возвратом результата) может быть разбита на команды CONFigure (Установка) и READ? (Считывание с возвратом). Действие двух команд CONFigure и READ? то же, что и одной



команды MEASure?, однако появляется возможность их разноса во времени. Команда CONFigure обеспечивает оптимальные параметры настройки прибора, которые могут быть длительными. Команда READ? запускает измерение с возвратом результатов контроллеру, когда прибор уже установлен в нужные режимы, поэтому время на ее выполнение требуется значительно меньшее, чем для команды MEASure?.

Разрыв CONFigure и READ? допускает вставку дополнительных команд, позволяя более точно настроить функции прибора. Это можно использовать для установок, которые являются важными для прикладной программы, но которые не гарантируются командой измерения (рис. 3.6). Например, входной блок прибора можно установить в нужный режим следующей последовательностью:

- CONFigure:VOLTage:AC 20,0.001 (0.001 является требуемой точностью; 20 В – ожидаемое значение сигнала);
- INPut:IMPedance 1E6 (входной импеданс устанавливается равным 1МОм);
- READ:VOLTage:AC? (запускается измерение с возвратом результата).

Запрос READ? инициирует сбор данных с их возвратом, однако, когда необходимо получить разные характеристики одних и тех же данных, использование запроса READ? запускало бы новый сбор данных для каждой новой характеристики. В этом случае запрос READ? разделяется на команду INITiate (Инициация), которая начинает сбор данных, и запрос FETCh? (Выборка), который сообщает прибору о необходимости возврата результатов измерений компьютеру. Таким образом, используя запрос FETCh?, можно последовательно получать различные характеристики от тех же самых данных.

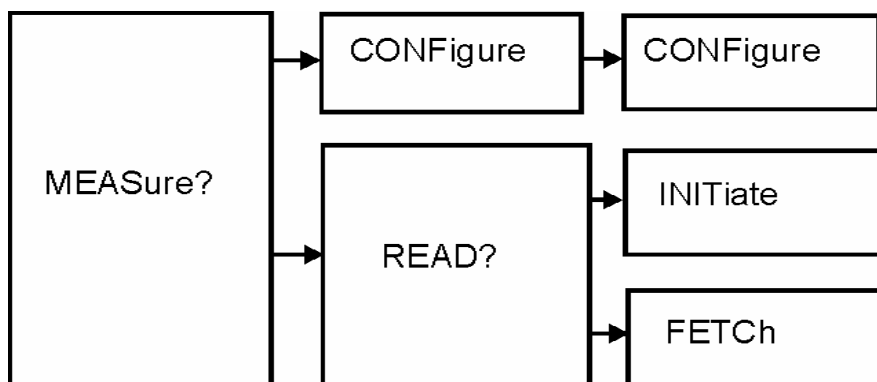


Рис. 3.6. Свойства команд измерения

Другой вариант управления:

```
CONFigure:VOLTage:AC 20,0.001
```

```
INPut:IMPedance 1E6
```

```
INITiate (начать измерения)
```

```
FETCh:VOLTage:AC? (передать данные о напряжении сигнала)
```

```
FETCh:VOLTage:PERiod? (передать данные о периоде сигнала)
```

Эти примеры показывают, что при программировании необходимы дополнительные знания о приборе, что он может измерять и период сигнала. Если это не так, то прибор выдаст сообщение об ошибке.

### Консорциум SCPI

Обычные стандарты выпускаются международными институтами, такими как IEC, ISO или ANSI/IEEE. Но из-за значительных финансовых инвестиций и взаимосвязанных коммерческих интересов для SCPI потребовался более динамичный и практичный подход.

Компании, которые утвердили первую версию стандарта в 1990 году, основали консорциум SCPI. Это компании: Hewlett-Packard, Tektronix, Philips, National Instruments, Racal Dana, Fluke, Keithley, Wavetek, Broel&Kjaer, а также Rohde&Schwarz, присоединившаяся к консорциуму сразу после его создания.

Консорциум SCPI имеет некоммерческую основу. Он обслуживает и расширяет стандарт, добавляя новые команды и ежегодно издавая новые его версии. Консорциум является международным и имеет штаб квартиру в Соединенных Штатах Америки.

Одна из задач консорциума SCPI – обеспечение открытости. Это означает, что любой производитель может представить на рынке SCPI приборы, если они соответствуют требованиям спецификации. Входящие в консорциум компании не имеют исключительных прав. Все изготовители контрольно-измерительных приборов могут участвовать в управлении и принятии решений консорциума.

### **3.5. МЕТОДЫ И СРЕДСТВА РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Все программные средства управления измерительными приборами можно разделить на пакеты программируемого типа и пакеты, управление которыми осуществляется посредством меню. Пакеты программируемого

типа требуют введения конкретных команд, хотя это могут быть высокоуровневые команды на широко используемых компьютерных языках: Паскаль, Си и др. Пакеты с управлением при помощи меню предусматривают выбор пользователем команд и функций из меню или ряда пиктограмм, воспроизводимых на экране. Такие пакеты обычно не имеют листинга кода в качестве конечного продукта. Они отличаются простотой использования и не требуют опыта программирования, однако их функциональные возможности обычно ограниченные. По гибкости они существенно уступают пакетам программируемого типа.

Имея в своем распоряжении пакет программируемого типа, предусматривающий создание кода программы, оператор может легко приспособить тест-программу для различных применений. Если код компилируется, то программа не будет ограничиваться какой-то одной рабочей средой выполнения. Кроме того, компилируемая программа характеризуется гораздо меньшими непроизводительными расходами и поэтому работает быстрее.

Главное достоинство унифицированных пакетов управляющих программ состоит в том, что они освобождают инженеров от необходимости самим решать задачи низкоуровневого управления контрольно-измерительными приборами. Стандартный пакет программных средств предусматривает программирование в удобной среде пользовательского интерфейса, содержит библиотеки контрольно-измерительных приборов, устройств и функций, а также обеспечивает выполнение математической обработки результатов измерений и представление данных в виде графиков, гистограмм и таблиц. В современных пакетах реализованы методы графического программирования. Компьютер используется не только для математического моделирования или обработки информации, но также для приема, обработки и анализа сигналов от реальных физических объектов и управления этими объектами.

За последнее время на рынок представлено много новых или усовершенствованных пакетов программ. Кроме повышения быстродействия, такие пакеты обычно отличаются тем, что содержат дополнительные средства поддержки и допускают работу в различных операционных системах.

Программные средства позволяют:

- получать данные от внешних устройств и сменных модулей ПК;
- управлять интерфейсом и внешним оборудованием (GPIB, VXI);

- отображать данные в виде графиков, текстов и объектов;
- использовать активные средства управления для расширения функциональных возможностей;
- связываться с другими программами Windows;
- сохранять данные в различных форматах (двоичном, ASCII);
- анализировать и обрабатывать данные;
- создавать рабочие программы и пользовательские интерфейсы;
- обращаться к любому модулю через Интернет.

Программы сбора данных и их анализа, позволяют разбивать полученный сигнал на отдельные сегменты, анализировать и строить графики.

Программные модули обработки данных обеспечивают:

- алгебраические преобразования;
- интегрирование и дифференцирование функций;
- измерение импульсных характеристик;
- статистический и корреляционный анализ;
- прямое и обратное преобразование Фурье;
- цифровую фильтрацию;
- интерполирование данных;
- формирование файлов данных.

Типовая структура пакета:

1. *Библиотека общих подпрограмм* содержит модули связи с приборами, подключенными к универсальной шине интерфейса, модули записи и чтения данных с диска, а также вычисления параметров сигналов. Подпрограммы написаны на Ассемблере, что упрощает прием и пересылку данных. Имеется справочное руководство для пользователя. Оно содержит примеры программ работы и особенно полезно, когда в составе оборудования входят приборы различных серий и типов.

2. *Библиотека обработки и вывода сигналов* обеспечивает сохранение и вывод информации на экран монитора. Настройка программы облегчается обширной помощью с использованием функциональной клавиши "HELP". Библиотека функций предоставляет оператору готовые программные модули.

3. *Библиотека сбора информации* содержит подпрограммы получения данных с приборов и представления их в формате, который необходим для использования этих данных.

4. *Библиотека графики* выводит информацию независимо от типа прибора, используя при этом драйвер из системы программного обеспечения графических средств. Этот драйвер расширяет диапазон применяемых принтеров и плоттеров, обеспечивает взаимодействие с высокоразрешающим дисплеем. Библиотека дает возможность пользователю управлять параметрами представления данных:

- логарифмическая или линейная зависимость;
- автоматическое или ручное масштабирование;
- установка меток;
- установка эталонных кривых.

На рис. 3.7 представлен спектр программ для испытаний, контроля и измерений. На нижнем уровне расположено готовое к применению прикладное ПО, которое можно использовать без каких-либо доработок. На более высокой ступени располагается ПО, требующее модернизации. Генераторы тестов позволяют выбрать и сконфигурировать в пределах имеющейся базы испытательные процедуры. Библиотеки прикладных программ обеспечивают выбор готового ПО, наиболее подходящего для решения конкретной задачи. Если прикладная программа разработана по техническим условиям заказчика, она является наиболее оптимальной, ее гибкость выше, но одновременно увеличиваются затраты на программирование.

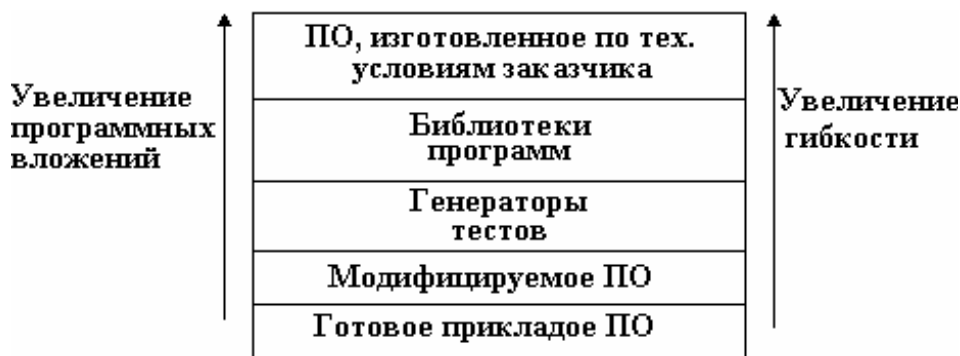


Рис. 3.7. Спектр ПО

Разработка программных средств автоматизированных контрольно-измерительных систем – трудоемкий и дорогой процесс. Чтобы сократить время на разработку программного обеспечения, необходимо выбрать соответствующие средства и методологию. Лучший способ – это создавать модульные программы, компоненты которых можно будет использовать потом для новых систем контроля.

Все возрастающее количество готовых прикладных программ, библиотек подпрограмм и систем управления базой данных затрудняет выбор. Если нет готового к работе оборудования, которое сразу может использоваться для решения поставленной цели или которое может быть модифицировано, тогда необходимо определить, какие части необходимого оборудования имеются на рынке и как эти части затем объединить.

Готовые прикладные программы обычно очень хорошо применяются для решения только тех задач, для которых они были созданы, но они не всегда полностью отвечают поставленным задачам. Необходимо решить, можно ли обеспечить модификацию, расширение системы или ее перенастройку. Модифицируемое ПО предполагает наличие возможностей расширения функций без серьезной доработки.

Пакеты унифицированных генераторов тестовых процедур позволяют создавать рабочую программу из готовых модулей, соединяя их в среде, удобной пользователю. Библиотеки универсальных программ дают пользователю еще большие возможности по применению готовых программ обработки и представлению информации, но они не могут решить все уникальные задачи заказчика. Только ПО, созданное по индивидуальным требованиям, позволяет решить проблемы в оптимальном варианте.

### **Системные контроллеры**

В качестве системных контроллеров шины КОП могут использоваться серийные и промышленные ПК, а также специализированные контроллеры. По мере развития компьютерной промышленности возникает проблема выбора оптимального контроллера СКИМ. Его характеристики, архитектура, возможности расширения, гибкость операционной системы, программное обеспечение – все это играет важную роль. Чтобы добиться успеха в автоматизации измерений, необходимо точно оценить функции аппаратуры, операционной системы и программного обеспечения. Автоматизированная система предъявляет требования к конструкции и конфигурации измерительных приборов, к устройствам коммутации и управления. Так как потребности со временем могут меняться, то используемый контроллер должен быть как можно более гибким. Эффективность контроллера напрямую связана с его пропускной способностью по входу и выходу. Имеется в виду не просто считывание данных с прибора, но и передача их через интерфейс в направлении к адресату. Когда пропускная способность определена, то необходимо рассмотреть скорость обработки и анализа

данных. Скорость процессора обычно вторична по отношению к характеристикам входа/выхода.

При выборе контроллера обращают внимание на следующие параметры и характеристики:

- компоновка и габариты;
- стоимость и эксплуатационные требования;
- быстродействие и операционная система;
- программное обеспечение и интерфейсы;
- гибкость и возможность расширения.

Уменьшение физического размера и переоборудование контроллеров в малые формы является современным направлением в компьютерной промышленности. Самыми важными задачами тестирования и измерения являются ввод информации в контроллер и пересылка команд к приборам и устройствам. Скорость системы будет зависеть от качества управления через интерфейс, который встраивается в контроллер или применяется как добавочное устройство.

На задней панели компьютеров имеются разъемы, к которым можно подключить дополнительные узлы. Контроллеры обычно имеют от двух до восьми слотов для дополнительной памяти, расширения входа/выхода или для потребностей сопряжения. Объединительная плата является внутренней частью контроллера. Ее интерфейс и построение определяют архитектуру компьютера и основные его возможности.

Интерфейс IEEE-488, известный как HP-IB или GPIB, может устанавливаться на всех видах оборудования. Интерфейс RS-232 также подходит ко многим конфигурациям. Он обычно используется для связи с удаленными модулями, принтерами и графопостроителями. Другие интерфейсы не являются определяющими при выборе контроллера, но они могут стать такими, если возникнет необходимость в большей гибкости.

Наиболее важной составляющей использования контроллера является программное обеспечение и операционная система. Выбор операционной системы создает условия программирования. Многие инженеры сначала выбирают программное обеспечение, а затем операционную систему.

Системные контроллеры, выпускаемые различными фирмами, обладают многими общими характеристиками, обеспечивающими специфические контрольно-измерительные функции.

Системные контроллеры обладают наиболее характерными особенностями, необходимыми для работы с измерительными приборами и системами. Приведем их основные характеристики:

- совместимость со стандартами;
- слабое излучение помех;
- возможность подключения дополнительных плат расширения;
- цветная графика;
- встроенный монохромный монитор с программируемыми клавишами и возможностью отключения программным путем;
- стандартный стоечный корпус;
- программируемая клавиатура;
- стационарный и сменный жесткие диски;
- контроллеры портов ввода/вывода;
- TTL – интерфейс входа/выхода (I/O);
- аналоговый I/O интерфейс.

Сменный жесткий диск имеет наряду с типичными признаками жесткого диска (малое время выборки) важное преимущество: с помощью ручки возможна его легкая замена. Это упрощает защиту данных и установку различного программного обеспечения.

Имеющийся встроенный монитор при измерениях, для которых допустимы предельно малые электромагнитные излучения помех, может быть выключен программным путем. Параллельно с встроенным монитором можно подключать внешний монитор.

В распоряжении пользователя имеется пять мест установки совместимых интерфейсных или специальных плат.

Клавиатура имеет, кроме всех функций стандартной клавиатуры, роликовую клавишу, при вращении которой генерируются цифровые коды четырех клавиш управления курсором. Особая защита клавиатуры предусмотрена в том случае, если требуется малый уровень излучения помех.

Программируемая встроенная клавиатура может быть установлена на переднюю панель. С ее помощью можно запрограммировать любую последовательность знаков для дальнейшего ввода нажатием одной клавиши. Клавиатура предусмотрена для обслуживания часто используемых программ, особенно при использовании СКИМ в стационарном режиме. Возможна замена функций внешней клавиатуры. С помощью поставляемых шаблонов клавиатуру можно маркировать индивидуально в соответствии с



требованиями пользователя. Встроенная клавиатура может программироваться с помощью меню путем вызова специальной конфигурирующей программы. Перепрограммирование клавиатуры возможно в ходе выполнения программ.

Интерфейс входа/выхода TTL необходим для коммутации и опроса внешних цифровых управляющих линий и поставляется в различных исполнениях.

### **Программное обеспечение**

Современные языки программирования являются удобной, управляемой с помощью меню средой со всеми необходимыми инструментами для быстрой разработки компактных и быстро работающих программ. Имеются редакторы проверки синтаксиса, режимы немедленного выполнения программы и интегрированный оконный отладчик. С помощью обучающей программы и экспертной системы можно за короткое время освоить эту программную среду. Языки обеспечивают модульное программирование (с процедурами, функциями, локальными переходами, структурами данных). Модули, написанные на QBasic можно вставлять в языки Си, Паскаль, Ассемблер и Фортран фирмы Microsoft. Расширения для задач измерительной техники позволяют достаточно просто программировать интерфейсы (ИЭС-шинный драйвер, цифровой и аналоговый интерфейсы ввода/вывода). Тестирование программ осуществляет отладчик, который использует оконный режим.

Современный язык высокого уровня содержит все элементы для структурного программирования, обеспечивает гибкость форматов данных и высокую скорость вычислений. Поставляемая функциональная библиотека языка Си имеет более чем 400 функций, среди которых: управление памятью, обработка данных, управление массивами, графика, ввод и вывод массивов и знаков, арифметика с целыми числами и числами с плавающей запятой, алгоритмы поиска и сортировки, обращение к MS-DOS и программный контроль. Для быстрой разработки программ поставляется интерактивная среда программирования, ориентированная на работу с окнами. Например, помощью программной среды LabTest неопытный пользователь (не имеющий навыков программирования сложных приборов) может оценивать измеряемый результат, представлять его графически, а также моделировать измерительный прибор. LabTest рисует на экране элементы регулировки и индикаторы приборов, подключенных к шине, после чего на

экране изображается передняя панель прибора. С помощью мыши или клавиатуры можно управлять моделируемыми приборами. LabTest имеет обширные библиотеки процедур и функций, предназначенные для профессиональных программистов.

Вместе с контроллером могут поставляться различные прикладные программы, в том числе:

1. Модули управления и обмена данными с приборами, связи со вторым ПК. Кроме пересылки данных, возможен запуск программ на внешнем ПК.

2. Модули цифровой обработки сигналов. Изменяющимся сигналом можно манипулировать как в области частот, так и в области времен.

3. Модули работы с графикой. Они представлены двумерной и трехмерной графикой, диаграммой Смитта и полярной диаграммой, а также удобными стандартными графическими программами с автоматическим выбором масштаба.

4. Модули поддержки дополнительных устройств, в том числе средств аналогового и цифрового ввода/вывода.

5. Библиотеки модулей КОП.

Все программы легко можно приспособить для разнообразного применения в измерительной технике.

## ЗАКЛЮЧЕНИЕ

Проектирование, изготовление и эксплуатация современного радиоэлектронного оборудования невозможны без наблюдения за изменением его состояния в ходе технологического процесса изготовления или непосредственной работы. Для организации такого наблюдения применяют компьютерные СКИМ, среди которых сегодня наиболее популярны приборно-модульные системы (ПМС), подробно рассмотренные в данной книге. Переход от «ручных» средств контроля к автоматизированным системам обусловлен рядом причин, в частности повышением требований к производительности и достоверности, а также развитием средств вычислительной техники, позволяющих эффективно решать задачи, связанные с автоматизацией процесса контроля и испытаний.

Внешняя приборно-модульная система автоматического контроля представляет собой автономный информационно-измерительный комплекс, который пространственно отделен от контролируемой аппаратуры и включает в себя приборы, измерительные преобразователи, коммутаторы и согласующие устройства. Уменьшение стоимости и времени разработки СКИМ предполагает использование унифицированных программных и аппаратных модулей, которые можно совмещать друг с другом. Как правило, используются серийные приборы, имеющие в соответствии с ГОСТ 26.003-80 интерфейс типа «Канал общего пользования» (КОП), но могут применяться и другие интерфейсы. Унификация позволяет создавать комплексы аппаратно-программных средств гибкие, перепрограммируемые и универсальные. Особое значение имеет программное обеспечение, позволяющее легко перепрограммировать СКИМ, поэтому современный радиоинженер должен понимать принципы взаимодействия в системе компьютер – приборы, а также уметь изменять режимы и программу испытаний.

## ПРИНЯТЫЕ СОКРАЩЕНИЯ

АКИС	Автоматизированная контрольно-измерительная система
ПО	Программное обеспечение
РТС	Радиотехническая система
РЭА СКИМ	Радиоэлектронная аппаратура
КОП	Система контроля, измерения и мониторинга
ANSI	Канал общего пользования American National Standards Institute – американский национальный институт стандартизации.
ASCII	American Standard Code for Information Interchange – американский стандартный код обмена информацией.
ATE	Automated Test Equipment – автоматическое испытательное оборудование (термин, применяемый к компьютерным системам).
DAQ	Data Acquisition – сбор данных
DIO	Digital Input/Output – цифровой ввод/вывод
GPIO	General Purpose Interface Bus – стандартная шина для соединения приборов и персонального компьютера.
HS488	High Speed 488 – высокоскоростной протокол для шины IEEE-488.
IEEE	Institute of Electrical and Electronic Engineers – институт электро- и радиоинженеров.
IEEE-488	Сокращенное обозначение стандартов IEEE-488-1978, 488.1 и 488.2-1987.
ISO	International Standards Organization – Международная Организация Стандартов.
LabVIEW	Laboratory Virtual Instrument Engineering Workbench – Лабораторный Пакет (Среда) для Проектирования Виртуальных Приборов,
SCPI	Standards Commands for Programmable Instruments – стандартные команды для программируемых приборов (расширение стандарта IEEE-488.2).

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. **ГОСТ 26.003-80.** Система интерфейса для измерительных устройств с байт-последовательным, бит-параллельным обменом информацией. – М.: Изд-во стандартов. – 1985. – 110 с.
2. **Гель, П.** Как превратить персональный компьютер в измерительный комплекс: пер. с фр. / П. Гель. – 2-е изд., испр. – М.: ДМК, 1999. – 144 с. – ISBN 5-89818-026-5.
3. **Мячев, А. А.** Интерфейсы средств вычислительной техники: справочник / А. А. Мячев. – М.: Радио и связь, 1993. – 352 с. – ISBN 5-256-00990-7.
4. **Поздняков, А. Д.** Автоматизация экспериментальных исследований, испытаний и мониторинга радиосистем / А. Д. Поздняков, В. А. Поздняков. – М.: Радиотехника, 2004. – 208 с. – ISBN 5-93108-066-X.
5. **Поздняков, А. Д.** Автоматизация экспериментальных радиофизических исследований: практикум / А. Д. Поздняков; Владим. гос. ун-т. – Владимир: Ред.-издат. комплекс, 2004. – 128 с. – ISBN 5-89368-474-5.
6. **Поздняков, А. Д.** Автоматизация радиоизмерений: учеб. пособие / А. Д. Поздняков. – Владимир: ВлГТУ, 1995. – 184 с. – ISBN 5-230-04783-6.
7. **Степанов, А. В.** Методы компьютерной обработки сигналов и систем радиосвязи / А.В. Степанов, С.А. Матвеев. – М.: СОЛОН – Пресс, 2003. – 208 с. – ISBN 5-98003-031-X.
8. ANSI/IEEE Standard 488.1-1987. *IEEE Standard Digital Interface for Programmable Instrumentation*. Institute of Electrical and Electronics Engineers, Inc. 345 East 47th St., New York, NY 10017, USA.
9. ANSI/IEEE Standard 488.2-1987. *IEEE Standard Codes, Formats, Protocols, and Common Commands*. Institute of Electrical and Electronics Engineers, Inc. 345 East 47th St., New York, NY 10017, USA.
10. Chojnacki R. SCPI – helping your productivity // Computer Aided Test Symposium. Hewlett-Packard. 1991. – P. 2-1 – 2-8.
11. Giesenhagen M. Evaluation new approaches to test software development // Computer Aided Test Symposium. Hewlett-Packard. 1991. – P. 5-1 – 5-8.

Учебное издание

ПОЗДНЯКОВ Александр Дмитриевич

ПРИБОРНО-МОДУЛЬНЫЕ СИСТЕМЫ  
КОНТРОЛЯ, ИСПЫТАНИЙ И МОНИТОРИНГА  
РАДИОАППАРАТУРЫ

Учебное пособие

Редактор И.А. Арефьева

Корректор В.В. Гурова

Компьютерная верстка Е.Г. Радченко

ЛР № 020275. Подписано в печать 29.04.05.

Формат 60×84/16. Бумага для множит. техники. Гарнитура Таймс.  
Печать на ризографе. Усл. печ. л. ?,??. Уч.-изд. л. ?,??. Тираж 100 экз.

Заказ

Редакционно-издательский комплекс  
Владимирского государственного университета.  
600000, Владимир, ул. Горького, 87.