

# Организация прикладного программного обеспечения контроллеров оборудования в новой Системе Управления комплексом У-70

С.И.Балакин, В.П.Воеводин, Е.В.Клименков  
*ГНЦ РФ Институт физики высоких энергий, Протвино, Россия*

Функции контроля и управления в новой Системе Управления комплексом У-70 распределены между компьютерами верхнего уровня — рабочими станциями — и контроллерами оборудования — нижний уровень. Так же распределено и прикладное программное обеспечение (ПО). Связь между верхним и нижним уровнями осуществляется посредством сети ETHERNET и магистрали МП-1553.

Контроллер оборудования (КО) представляет собой каркас в стандарте MULTIBUS-1, модуль процессора (I-186), терминал МП-1553, модуль приемника таймерных сообщений, драйвер магистрали КАМАК/СУММА с мультиплексором внешних прерываний. В настоящее время такую конфигурацию имеют практически все КО, работающие в новой Системе Управления. Контроллер работает в режиме реального времени и обеспечивает прием, обработку и запись в аппаратуру управляющих данных, необходимые измерения и контроль состояния подключенной аппаратуры. Кроме того, независимо от функционирования программ на верхнем уровне КО автоматически изменяет режим работы аппаратуры в соответствии с изменением режима работы ускорителя (PPM — Puls to Puls Modulation). Информацию о том, какой именно режим работы комплекса необходимо реализовать в следующем суперцикле, контроллер получает через модуль приемника от генератора таймерных сообщений.

Взаимодействие программ верхнего и нижнего уровней основано на принципе обмена данными средствами специализированной распределенной базы данных ССУДА. Информация хранится в виде трехмерных таблиц, которые физически распределены по рабочим станциям и контроллерам оборудования. Поскольку процессорный модуль не имеет энергонезависимой памяти для хранения динамически меняющихся данных, копии таблиц, предназначенных для приложений КО, хранятся на рабочих станциях и автоматически загружаются в контроллер при включении питания последнего. В процессе работы прикладные программы рабочей станции и контроллера взаимодействуют, осуществляя независимо друг от друга чтение/запись данных в одни и те же таблицы, которые физически хранятся в памяти КО.

В программном обеспечении контроллеров оборудования можно выделить системную часть, состав которой один и тот же для всех контроллеров. Это монитор VPV (хранится в EPROM) и пакет программ для работы с базой данных (СУБД). Основными функциями монитора являются: обслуживание линий связи на уровне приема/передачи сообщений, загрузка системных и прикладных программ с компьютеров верхнего уровня, работа с устройствами, расположенными на плате процессора, обработка внутренних прерываний, прием и передача прикладным программам таймерных сообщений.

## Распределение функций программного обеспечения

При реализации принципа распределенных вычислений, положенного в основу решения задач контроля и управления технологическими подсистемами У-70, были детерминированы основные функции прикладных программ на рабочей станции и в КО, которые управляют одной подсистемой.

Программа на станции обеспечивает:

- реализацию алгоритмов преобразования физических данных, с которыми работает оператор, в технологические, т.е. в данные, практически готовые для записи в аппаратуру;
- заполнение этой информацией общих таблиц базы данных в контроллере оборудования;
- периодическое считывание (асинхронно с технологическим процессом) измеренных данных из КО, их преобразование в физические величины и запись в базу данных на станции;
- периодическое считывание из контроллера диагностической информации о состоянии прикладной программы и аппаратуры управления с целью своевременного оповещения операторов в случаях аварийных ситуаций.

Прикладная программа в контроллере должна:

- периодически проверять наличие новых управляющих данных от рабочей станции и в случае их появления выполнить преобразования, отражающие особенности программирования конкретных модулей аппаратуры управления;
- синхронно с технологическим процессом осуществлять запись управляющей информации в аппаратуру и считывание необходимых данных, учитывая специфику работы различных модулей управляющей электроники;
- заполнять общие таблицы базы данных измеренными данными и диагностической информацией;
- выполнять необходимые действия в случае различных событий, происходящих на ускорителе, в соответствии с информацией, получаемой от приемника таймерных сообщений;
- обрабатывать внешние прерывания от аппаратуры управления, подключенной к контроллеру.

### **Организация интерфейса прикладных программ КО и рабочей станции**

В результате опыта разработки и эксплуатации ПО новой Системы Управления для бустерного синхротрона (включает 10 контроллеров оборудования) была выработана единая схема взаимодействия программ КО и рабочих станций и определен протокол обмена данными.

Для организации интерфейса используются два типа таблиц базы данных ССУДА: диалоговая и таблица данных контроля и управления. Диалоговая таблица служит для реализации динамического взаимодействия между ПО контроллера и рабочей станции. Все поля таких таблиц имеют фиксированное назначение, а записанные туда данные — фиксированный смысл для всех технологических подсистем. Для одного приложения, работающего в контроллере, используется одна диалоговая таблица.

Таблица данных содержит:

- 1) управляющие технологические данные для записи в аппаратуру — заполняет ПО рабочей станции;
- 2) измеренные данные — записывает ПО контроллера оборудования.

Количество таких таблиц, их размеры и форматы данных утверждаются на стадии проектирования прикладных программ для каждой технологической подсистемы.

Со стороны рабочей станции технологическая подсистема, управляемая КО, представляет собой совокупность устройств, для которых нужно сформировать управляющие данные и получить измеренные. Каждое такое устройство описывается строкой диалоговой таблицы и своей таблицей данных. Строка содержит три основных поля: команда рабочей станции (например, “Новые данные”), код состояния (фазы) работы приложения в КО, поле

флагов типа “запрос-подтверждение”. Соответствие строк диалоговой таблицы и таблиц данных определяется в начале разработки ПО для рабочей станции и КО. В зависимости от того, участвует ли устройство в реализации режима РРМ или нет, определяется число плоскостей таблиц, содержащих управляющие данные. В настоящее время определено 8 режимов работы комплекса или отдельных его частей. Данные одной плоскости описывают один режим работы устройства. Таким образом, число плоскостей может быть 1 или 8.

### **Общая структура прикладной программы контроллера оборудования**

Прикладная программа КО реализует бесконечный цикл следующего вида:

- 1) проверка наличия и выполнение команд от программы рабочей станции;
- 2) обработка глобальных событий на комплексе, информацию о которых получает приемник таймерных сообщений;
- 3) обработка локальных событий, порождаемых подключенной аппаратурой управления;
- 4) анализ информации об ошибках, полученной на первых трех шагах, и соответствующая диагностика для программы на рабочей станции.

Специфика выполнения команд от рабочей станции заключается в том, что новые данные поступают в контроллер асинхронно с работой ускорителя, а реальная запись кодов в различные модули может производиться только в определенные интервалы времени. Поэтому, получив новые данные, прикладная программа считывает их из соответствующей таблицы базы данных, производит их обработку в соответствии с логикой работы устройств и помещает готовые для записи коды в массив, с которым работает процедура записи в аппаратуру. Дождавшись события (сигнала) о том, что можно писать в модуль, программа осуществляет запись, а затем контрольное считывание данных. По результатам выполнения операций выставляется код возврата в соответствующем элементе диалоговой таблицы.

Генератор таймерных сообщений собирает (в виде электрических сигналов) информацию о событиях, происходящих на ускорителе, анализирует ее и выдает соответствующие сообщения во все КО, включенные в Систему Управления. Прикладные программы всех контроллеров трактуют смысл таких сообщений одинаково. Для своей работы различные приложения используют различные события. Поэтому, по аналогии с таблицей векторов прерываний монитора, в прикладной программе описывается таблица векторов глобальных событий. В ней определено, какие именно события необходимы программе и какие функции должны вызываться, если произойдет то или иное событие.

Электрические сигналы о локальных событиях заведены на контроллер прерываний процессорной платы и действительно вызывают прерывание работы прикладной программы. В силу этого, процедуры обработки, которые находятся в теле программы, должны обрабатываться по возможности быстро. Разработан ряд правил, которые необходимо соблюдать при программировании таких процедур.

### **Необходимость и возможность унификации прикладных программ в КО**

Необходимость унификации ПО в контроллерах оборудования стала очевидной уже на первых этапах создания новой Системы Управления У-70. В первую очередь потому, что писать прикладные программы начали люди, хорошо представляющие себе работу управляющей электроники, но не имеющие опыта разработки программ для распределенных вычислительных систем. Вторым определяющим моментом стала сложность поддержки уже работающего ПО в связи с постоянным развитием программного обеспечения верхнего уровня. Возможность унификации приложений в КО появилась после разработки и

отладки в сеансе работы ускорителя программного обеспечения на рабочих станциях и в контроллерах для нескольких технологических подсистем бустерного синхротрона. На этом этапе были полностью определены и отлажены перечисленные выше принципы управления и распределенных вычислений и функции приложений в контроллерах оборудования. На основании этого были разработаны:

- алгоритмы и протокол взаимодействия прикладных программ на станции и в КО;
- единый список глобальных событий на комплексе, принцип их формирования, пересылки в контроллеры, приема, дешифрации и передачи прикладным программам КО;
- надстройка над СУБД ССУДА, позволившая оптимизировать процедуры чтения/записи данных в базу с точки зрения их быстродействия.

Все это дало возможность реализовать унифицированную для всех контроллеров структуру прикладной программы. Общие данные и процедуры, реализующие общие для всех контроллеров функции, объединены и оформлены в виде ядра приложения. Вокруг/параллельно с ним работают драйверы модулей управляющей электроники.

Основными функциями ядра являются:

- обеспечение взаимодействия с программой на рабочей станции;
- работа с базой данных;
- управление глобальными и локальными событиями.

Драйверы реализуют алгоритмы программирования модулей электроники и обработку локальных прерываний по командам ядра. Полная настройка программы производится после ее старта одной из процедур ядра на основе таблицы конфигурации, в которой описаны все драйверы (их точки входа, адреса локальных данных и т.д.) и соответствующие им таблицы данных, необходимые для функционирования приложения.

Таким образом, программирование нового приложения сводится к написанию драйверов модулей электроники и созданию таблицы конфигурации. Это значительно ускоряет процесс разработки и уменьшает время отладки приложений в контроллерах оборудования.