

---

УДК 004.45

## **Управляющее программное обеспечение устройства сопряжения служебного канала управления спутника**

**Звонарева Г. А.<sup>1\*</sup>, Ибрагимов Ф. Р.<sup>2\*\*</sup>**

<sup>1</sup>*Московский авиационный институт (национальный исследовательский университет), МАИ, Волоколамское шоссе, 4, Москва, А-80, ГСП-3, 125993; Россия*

<sup>2</sup>*Научно-исследовательский институт «Аргон», Варшавское ш., 125, 117587; Россия*

*\* e-mail: zvonarev@uimail.ru*

*\*\* e-mail: appprx@gmail.com*

### **Аннотация**

Рассматриваются аспекты разработки управляющего программного обеспечения устройства сопряжения, входящего в состав бортовой аппаратуры спутника. При разработке управляющей программы используется операционная система реального времени, основанная на стиле «классной доски». Отладка и тестирование программного обеспечения устройства сопряжения проводится с помощью имитатора внешних связей.

### **Ключевые слова**

устройство сопряжения, спутник, управляющая программа, операционная система реального времени, стиль «классной доски», телеметрия, имитатор внешних связей

### **Введение**

В настоящее время в области телекоммуникаций искусственные спутники Земли играют ключевую роль. И на данный момент ведется разработка новой серии коммерческих телекоммуникационных спутников. Одним из основных модулей этих спутников является устройство сопряжения. Данное устройство входит в состав бортовой аппаратуры служебно-

го канала управления спутника и играет ключевую роль в управлении спутником с земной управляющей станции.

Основными функциями устройства сопряжения является прием сообщений с земной станции и отправка телеметрических пакетов на земную станцию. Также устройство сопряжения должно обрабатывать управляющие сигналы, которые могут прийти от бортовой аппаратуры. Устройство сопряжения состоит из аппаратной части и управляющего программного обеспечения. Большинство функций устройства сопряжения возлагаются на его управляющее программное обеспечение. Хотя в состав предыдущих версий спутника также входило устройство сопряжения, их аппаратные части, а также выполняющие ими функции существенно отличаются. В частности, отличаются типы сообщений, методы приема и обработки сообщений, сбора данных для передачи их на Землю, взаимодействие с бортовой аппаратурой, а также добавляются новые задачи. Таким образом, возникает необходимость разработки новой управляющей программы для нового устройства сопряжения. При этом увеличивается объем и сложность программного обеспечения, а также и предъявляемые к нему требования. В связи с этим, при разработке управляющего программного обеспечения необходимо рассмотреть использование новых подходов, в том числе с использованием операционных систем реального времени.

При разработке программного обеспечения нового устройства сопряжения было принято решение использовать операционную систему реального времени, использующую модульный принцип и принцип «классной доски». В предыдущих версиях устройства сопряжения задачи выполнялись последовательно в соответствии с заданной циклограммой. И при увеличении сложности выполняемых задач и их количества возникает необходимость в новом подходе организации вычислительного процесса, обеспечивающего возможность распределения приоритетов и возможность вытеснения различных модулей программного обеспечения. Использование выбранной операционной системы позволяет разделить всю управляющую программу устройства на отдельные модули, каждый из которых может быть разработан и использован относительно независимо от всех остальных. Принцип «классной доски» позволяет уменьшить количество коммуникационных интерфейсов между модулями. Операционная система дает возможность гибко распределять приоритеты и способы выполнения различных модулей программного обеспечения, что позволяет высвободить часть процессорного времени микропроцессора и укладываться в установленные временные ограничения. Все это позволяет в короткие сроки нарастить функционал устройства либо моди-

фицировать уже имеющиеся функции устройства для использования в новых сериях спутников.

### **Разработка управляющей программы**

В соответствии с основными функциями устройства сопряжения управляющее программное обеспечение содержит следующие основные модули:

- 1) модуль отправки пакетов;
- 2) модуль приема и обработки сообщений;
- 3) модуль обработки управляющих импульсов.
- 4) модуль межпроцессорного обмена;

В составе устройства имеются два вычислительных модуля, взаимодействие между которыми осуществляется через модуль межпроцессорного обмена.

Структурная схема управляющего программного обеспечения устройства сопряжения представлена на рисунке 1.

Модуль отправки пакетов непрерывно передает на Землю телеметрические пакеты. Это реализуется наличием двух буферов модуля: пока отправляется содержимое одного буфера на Землю, заполняется второй буфер. Во время заполнения телеметрического пакета особое место занимает заполнение поля пакета, отведенного под телеметрический кадр. Телеметрический кадр представляет собой данные, полученные от бортовой аппаратуры телесигнализации спутника, которые необходимо отправить на Землю. Для получения телеметрического кадра управляющее программное обеспечение отправляет запрос в бортовую аппаратуру спутника и ожидает ответа. Если ответ был получен, и получены корректные данные, что определяется соответствующим регистром состояния блока сопряжения с бортовой аппаратурой, то полученные данные копируются в соответствующее поле телеметрического пакета. Если были получены некорректные данные, то управляющая программа отправляет повторный запрос в бортовую аппаратуру. Если и после второй попытки были получены некорректные данные, то фиксируется ошибка получения телеметрического кадра.

Кроме основного телеметрического кадра в телеметрическом пакете также присутствует дополнительный телеметрический кадр, который содержит информацию о работе

управляющей программы, а также самого устройства сопряжения. Далее при приеме телеметрического пакета на земной станции операторы могут анализировать корректность работы управляющего программного обеспечения и устройства в целом.

При приеме сообщений с земной станции управляющее программное обеспечение в первую очередь определяет тип принятого сообщения. Если принятое сообщение является командой высокого уровня, то данная команда выполняется непосредственно самой управляющей программой. Если же была принята разовая команда либо массив данных, то управляющее программное обеспечение проверяет корректность полученных данных и при их корректности передает данные в блок управления бортовой аппаратуры. Результат передачи данных в блок управления проверяется наличием корректной квитанции от бортовой аппаратуры.

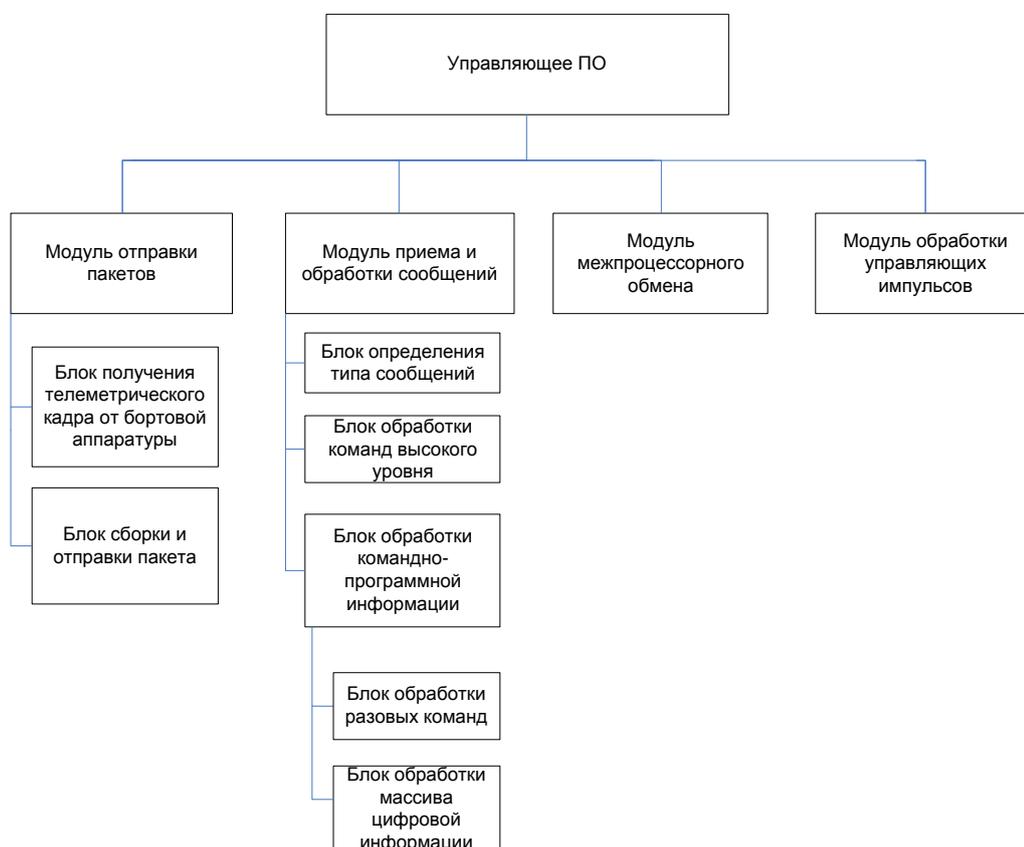


Рисунок 1 – Структурная схема управляющего программного обеспечения устройства сопряжения

Устройству сопряжения от бортовой аппаратуры могут прийти высокоприоритетные управляющие команды, обработкой которых занимается модуль обработки управляющих

импульсов. Данные команды предназначены для управления питанием устройства и режимами передатчиков спутника. Все управляющие команды дублируются по каналу межпроцессорного обмена на случай, если один из комплектов устройства сопряжения не смог принять команду по каналу выдачи управляющих импульсов.

В составе устройства сопряжения находится два вычислительных модуля, взаимодействие между которыми реализовывает модуль межпроцессорного обмена управляющего программного обеспечения. Каждый из комплектов устройства может быть либо в активном, либо в пассивном режиме работы. В каждый момент времени активным должен быть только один комплект, который и занимается отправкой телеметрических пакетов и обработкой полученных с Земли сообщений. В штатном режиме работы активным всегда должен быть основной комплект. Однако при выходе из строя какого-либо модуля основного комплекта есть возможность сменить активность комплектов. Таким образом, модуль межпроцессорного обмена управляющего программного обеспечения служит для определения активности каждого из комплектов устройства сопряжения, для смены активности, а также для дублирования управляющих импульсов соседнему комплекту.

Каждый из модулей управляющего программного обеспечения взаимодействует с другими модулями, т.е. необходимо реализовать совокупность интерфейсов взаимодействия между соответствующими модулями. Модули программы могут модифицироваться при изменении или расширении их функций. Например, в модуль приема и обработки сообщений можно добавить функцию аутентификации для проверки корректности отправителя принятых сообщений. Кроме того, могут добавляться новые модули, выполняющие новые задачи. Поэтому разработку программного обеспечения необходимо реализовать таким образом, чтобы была возможность гибкой модификации модулей и добавления новых модулей при допустимой сложности организации взаимодействия между модулями. В связи с этим, при разработке управляющего программного обеспечения было принято решение использовать операционную систему реального времени, использующую модульный принцип и принцип «классной доски». При использовании данной операционной системы программное обеспечение представляет собой совокупность программных компонентов, имеющих четко определенный интерфейс и допускающих их относительно независимую разработку и использование. Индивидуальные детали и сложность, соответствующая каждому модулю, инкапсулированы внутри модуля.

При организации вычислительного процесса в операционной системе используется так называемый стиль «классной доски», когда взаимодействие отдельных модулей программного обеспечения происходит через некоторую глобальную базу данных, называемую «классной доской». Информация, помещенная на «классную доску», является общедоступной для всех модулей, механизмов управления, заново добавляемых модулей. Взаимодействие модулей происходит только через «классную доску» косвенным образом. Косвенное взаимодействие уменьшает количество коммуникационных интерфейсов, которые должны поддерживаться среди взаимодействующих модулей. При функционировании системы модули выполняются в мультипрограммном режиме с приоритетным обслуживанием и с возможностью вытеснения [1]. Распределение приоритетов и способов выполнения различных модулей программного обеспечения позволяет гибко распределять процессорное время между различными модулями, что позволяет укладываться в установленные временные ограничения. Также это позволяет высвободить часть процессорного времени микропроцессора, что позволит при необходимости нарастить функционал устройства.

Системы, основанные на стиле «классной доски», уже используются для решения практических задач, в частности в области искусственного интеллекта, визуализации данных и даже в сетевых приложениях [2, 3]. Однако, как правило, данный подход используется для сбора информации от разных источников для последующей ее обработки в некотором главном модуле. При разработке же управляющего программного обеспечения устройства сопряжения «классная доска» играет ключевую роль при взаимодействии отдельных модулей управляющей программы, т.к. каждый из модулей взаимодействует с несколькими модулями.

Таким образом, при использовании выбранной операционной системы, использующей модульный принцип и принцип «классной доски», программное обеспечение, реализующее вычислительный процесс, разбивается на взаимодействующие между собой программные компоненты, которые имеют строго определенную структуру и интерфейс и обладают следующими свойствами:

- 1) каждый компонент представляет собой независимый программный модуль. Компонент выполняется в рамках потока управления и является единицей мультипрограммного управления, т.е. компоненты могут выполняться параллельно;

2) программные компоненты допускают повторное использование, повторный запуск, настройку и реконфигурацию;

3) для создания программных компонентов может быть использована любая среда программирования на языке высокого уровня С;

4) взаимодействие программных компонентов выполняется через общую область, называемую глобальной базой данных;

5) каждый программный компонент имеет входные и выходные параметры, которые хранятся в глобальной базе данных. Каждый программный компонент может изменять состояние системы, модифицируя переменные в базе данных;

6) набор переменных, с которыми работает программный компонент, включая входные и выходные параметры, образуют локальную базу данных этого компонента. Локальная база данных представляет собой поднабор глобальной базы данных. Программный компонент может использовать только переменные из локальной базы данных;

7) программные компоненты делятся на два класса: вычислительный компонент и интерфейсный компонент. Вычислительный компонент использует при своей работе только данные из базы данных. Интерфейсный компонент предназначен для взаимодействия с внешними устройствами, в его функции входит считывание/запись данных из/в устройства и преобразование данных из формата физического устройства к формату переменных в базе данных. Интерфейсный компонент по своим функциям аналогичен драйверу устройства. В состав драйвера может входить программа обработки прерываний от конкретного устройства;

8) для координации программных компонентов используется управляющий модуль, который выполняет диспетчеризацию и синхронизацию программных компонентов, обеспечивает запуск компонентов в соответствии с заранее установленной циклограммой и/или по событиям, обеспечивает все функции работы с базой данных и некоторые системные функции;

9) программные компоненты при своем функционировании используют только функции, предоставляемые управляющей программой.

Для написания и отладки управляющего программного обеспечения использовалась интегрированная среда разработки, поддерживающая архитектуру микропроцессора устройства сопряжения. Данная среда позволяет компилировать программы для данной целевой архитектуры, а также при отладке программ позволяет эмулировать эту архитектуру [4].

Для автономной отладки управляющего программного обеспечения, необходимо полученный бинарный файл программы загрузить в память устройства сопряжения. Для этого устройство подключается к персональному компьютеру, и с помощью специальной программы управляющая программа записывается в память устройства. Для проверки корректности работы управляющего программного обеспечения на персональном компьютере используется программа, считывающая данные с устройства.

Для тестирования и отладки работы управляющего программного обеспечения устройства сопряжения в условиях, близких к реальным, был использован имитатор внешних связей. Данное устройство имитирует работу реальной отсутствующей аппаратуры спутника и земной станции управления. На рисунке 2 представлена схема взаимодействия устройств.

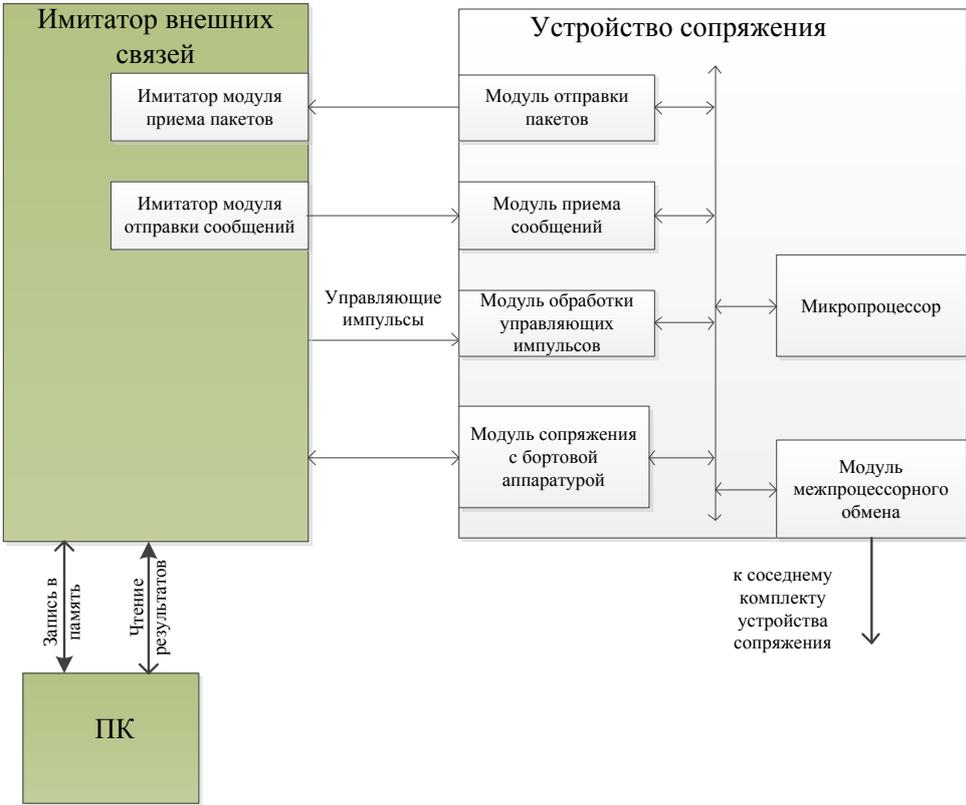


Рисунок 2 – Схема взаимодействия устройств при комплексном тестировании и отладке

Для тестирования были разработаны программа для имитатора внешних связей, а также программа для персонального компьютера, позволяющая запускать заданную последовательность тестов и выводить результаты тестирования. Тестировались все каналы передачи данных как в штатном режиме работы устройства, так и с имитацией нештатных ситуаций, таких как отказы одного из каналов передачи, отсутствие ответа от бортовой аппаратуры и т.д. Все проведенные тесты прошли успешно.

## **Выводы**

Таким образом, было разработано управляющее программное обеспечение устройства сопряжения служебного канала управления телекоммуникационного спутника с использованием операционной системы реального времени. Т.к. новые устройства сопряжения выполняют новые задачи, вследствие чего усложняется их программное обеспечение, то методы организации вычислительного процесса, основанные на выполнении различных функций устройства по заданной циклограмме, которые использовались в предыдущих сериях устройства сопряжения, уже не позволяют решить поставленную задачу. При разработке нового устройства сопряжения было принято решение использовать операционную систему, использующую модульный принцип и стиль «классной доски». Модули управляющей программы представляют собой отдельные компоненты операционной системы, что в случае изменения или расширения функций устройства сопряжения позволит легко изменить какой-либо отдельный модуль управляющей программы либо добавить новый модуль без изменения существующих. Особое место занимает распределение приоритетов и способов выполнения различных модулей программного обеспечения, что позволяет высвободить часть процессорного времени микропроцессора. Это позволит при необходимости нарастить функционал устройства. Кроме того, выделение в операционной системе функций, зависящих от аппаратной части, в отдельный модуль, позволит адаптировать разработанную управляющую программу для другого микропроцессора в случае такой необходимости. Принцип «классной доски» позволил упростить взаимодействие модулей за счет использование глобальной базы данных для обмена необходимыми данными.

Была проведена как автономная отладка программного обеспечения, так и комплексная в составе имитатора внешних связей. Автономная отладка управляющего программного обеспечения показала корректность работы отдельных его модулей. Тестирование же уст-

ройства сопряжения с помощью имитатора внешних связей подтвердило корректность работы управляющей программы в условиях, близких к реальным.

## **Библиографический список**

1. И.Б.Бурдонов, А.С.Косачев, Пономаренко В.Н. Операционные системы реального времени. М.: Институт системного программирования РАН, 2006, 98 стр.
2. Камерон Хьюз, Трейси Хьюз. Параллельное и распределенное программирование на C++: пер. с англ. М.: Издательский дом «Вильямс», 2004, 672 стр.
3. D. Rudenko and A. Borisov. An overview of blackboard architecture application for real tasks. Scientific proceedings of Riga technical university, Information Technology and Management Science, Issue 5, Vol.31 (2007), RTU, Riga, P. 50-56.
4. Green Hills Software, Inc. MULTI: Debugging. Green Hills Software, Inc. November, 2004, 969 стр.