

ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ

УДК 62529

DOI: 10.22213/2410-9304-2021-3-47-54

Разработка технологического модуля для контрольно-проверочной аппаратуры бортовой аппаратуры хранения временных данных

Д. Д. Баженова, ООО «РосА», Ижевск, Россия

М. А. Аль Аккад, кандидат технических наук, ИжГТУ имени М. Т. Калашникова, Ижевск, Россия

А. А. Ивакин, ООО «ИРЗ», Ижевск, Россия

Высокоскоростная радиолиния ВРЛ предназначена для передачи целевой информации от аппаратуры космических аппаратов на Землю. Блок бортовой аппаратуры хранения временных данных БАХВД входит в состав бортовой аппаратуры высокоскоростной радиолинии БАВРЛ. Перед запуском аппаратов в космос проводятся приемо-сдаточные испытания входного контроля. Для этого необходимо было разработать контрольно-проверочную аппаратуру КПА и программное обеспечение ПО для нее. КПА БАХВД предназначена для проверки БАХВД в целом и блоков из ее состава при автономных испытаниях. В рамках данной работы рассматривается одна из составляющих системы ПО КПА – подсистема технологического программного обеспечения, которая позволяет проверить работу ПЛИС в составе БА ХВД. Приведены основные алгоритмы подсистемы и функции, выполняемые подсистемой. В данной работе рассматривается взаимодействие оператора контрольно-проверочной аппаратуры с программируемыми логическими интегральными схемами ПЛИС, входящими в состав блока бортовой аппаратуры хранения временных данных БАХВД. Для обеспечения этого взаимодействия необходимо программное обеспечение для отладки микросхем. В ходе разработки устройства БАХВД была выбрана ПЛИС с архитектурой RISC-V, отладка через GRMON получилась невозможной, и пришлось выбрать OpenOCD. В результате выполнения работы был разработан модуль технологического программного обеспечения для проведения испытаний и обеспечения работоспособности ПЛИС в составе бортовой аппаратуры хранения временных данных. Разработана подсистема взаимодействия с устройством для обеспечения возможности отправлять команды и получать ответные сообщения, разработана подсистема служебных функций для преобразования ответных сообщений в читаемый для оператора вид, разработана подсистема взаимодействия модуля с основным каркасом ПО КПА, разработаны виджеты для обеспечения возможности ручного ввода команд от пользователя удобным ему способом.

Ключевые слова: бортовая аппаратура хранения временных данных, контрольно-проверочная аппаратура, технологическое программное обеспечение, ПЛИС, высокоскоростная радиолиния.

Введение

Бортовые и наземные средства приема, обработки, оперативного хранения, передачи по каналам связи входят в состав систем передачи целевой информации с космических аппаратов. Системы передачи информации постоянно развиваются и совершенствуются. Увеличение разрешающей способности аппаратуры наблюдения влечет за собой совершенствование космических систем дистанционного зондирования Земли и приводит к развитию производительности аппаратуры и эффективности функционирования бортовой и наземной аппаратуры радиолиний, оперативности регистрации и хранения. Требования к облику бортовой аппаратуры системы передачи целевой информации и высокоскоростной радиолинии (ВРЛ) определяются тенденциями развития техноло-

гии, что требует модернизации бортовой и наземной аппаратуры ВРЛ [1].

Контрольно-проверочная аппаратура (КПА) – аппаратно-программный комплекс, который необходим для проверки работоспособности и оценки правильности функционирования устройств [2]. Программное обеспечение контрольно-проверочной аппаратуры (ПО КПА) обеспечивает формирование последовательностей и объемов проверок, надлежащий порядок действий при проведении контроля и испытаний изделий, управляет ходом выполнения проверок [3].

Целью данной работы является разработка модуля технологического программного обеспечения (ТПО) для проведения испытаний и обеспечения работоспособности ПЛИС в со-

стве бортовой аппаратуры хранения временных данных (БА ХВД) [4].

Состав и назначение бортовой аппаратуры хранения временных данных

ВРЛ в составе систем передачи целевой информации предназначена для передачи целевой информации от аппаратуры космических аппаратов на Землю [5]. ВРЛ включает в себя бортовую аппаратуру [6]. Блок БА ХВД входит в состав бортовой аппаратуры ВРЛ [7]. БА ХВД предназначена для временного хранения целевой информации, переданной источником, и последующей передачей в приемник [8]. В качестве источника информации выступает бортовая аппаратура защиты информации. В качестве приемника выступает передающее устройство. Бортовая аппаратура хранения временных данных состоит из двух полуккомплектов: основного и резервного.

Каждый полуккомплект имеет собственные линии питания, интерфейсы приема целевой информации от основного и резервного источника, интерфейсы передачи целевой информации в основной и резервный приемник, интерфейсы управления, интерфейсы внешней телеметрии.

Бортовая аппаратура хранения временных данных состоит из следующих компонентов (рис. 1):

- ПЛИС – программируемая логическая интегральная схема, основной функциональный блок, реализующий вычислительную систему для управления остальными компонентами системы;

- оперативная память – энергозависимая память для промежуточного хранения данных и исполняемых программ;

- энергонезависимая память типа NOR – внешняя память, предназначенная для хранения прошивки ПЛИС, программного обеспечения и начальных данных;

- энергонезависимая память типа NAND – массив внешней памяти, предназначенный для хранения полезных данных;

- интерфейс приема – внешний интерфейс для приема полезной информации;

- интерфейс передачи – внешний интерфейс для передачи полезной информации;

- интерфейс управления – внешний интерфейс, предназначенный для управления и контроля БА ХВД;

- источник вторичного электропитания – блок, обеспечивающий преобразование входного напряжения до уровня, необходимого для обеспечения питания остальных блоков.

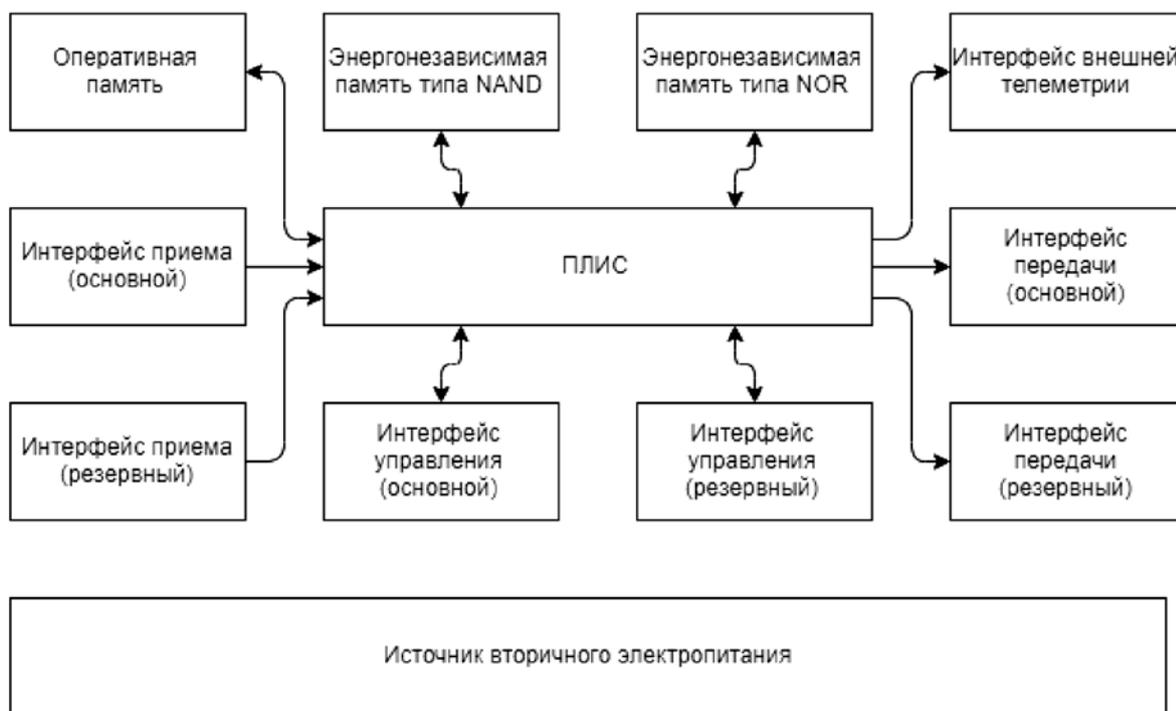


Рис. 1. Схема устройства БА ХВД

Fig. 1. Diagram of the onboard equipment for storing temporary data

Используемые подходы

В рамках данной работы рассматривается взаимодействие оператора КПА с ПЛИС, входящей в состав БА ХВД. Для обеспечения этого взаимодействия необходимо программное обеспечение для отладки микросхем. В ходе разработки системы были изучены два отладчика интегральных микросхем: GRMON [9] и OpenOCD [10]. GRMON – общий отладочный монитор для процессора LEON и для проектов типа «система на кристалле», обеспечивающий ненавязчивую среду отладки на реальном целевом оборудовании. GRMON включает следующие функции и инструменты:

- чтение (запись) всех системных регистров и памяти;
- встроенный дизассемблер;
- загрузку и выполнение приложений на процессоре LEON;
- дистанционный отладчик GNU (GDB);
- поддерживает USB, JTAG, RS232, PCI, Ethernet, SpaceWire интерфейсы;
- основное преимущество – наличие опыта работы с данным отладчиком при разработке подобных систем.

OpenOCD

Название OpenOCD произошло от сокращения Open On-Chip Debugger (открытый отладчик для чипов). OpenOCD предоставляет инструментарий отладки внутрисхемного программирования, внутрисхемного тестирования для встраиваемых систем (микроконтроллеров, FPGA и т. п.) [11]. OpenOCD предоставляет доступ к адаптеру для отладки – маленькому аппаратному модулю, который помогает получить требуемые сигналы для отладки целевого устройства (обычно с одной стороны адаптер подключается к компьютеру через USB, а с другой стороны имеется интерфейс JTAG, через который подключено отлаживаемое устройство) [12]. Такой адаптер нужен, так как у хоста отладки (компьютер, на котором запущен OpenOCD) нет поддержки специальных сигналов и коннектора, необходимых для подключения к целевому устройству [13]. В ходе разработки устройства БАХВД была выбрана ПЛИС с архитектурой RISC-V, что сделало не-

возможной взаимодействие с ней через GRMON и был выбран отладчик OpenOCD [14].

Материалы и методы

Для разработки пользовательского приложения использовалась среда разработки Qt для семейства Linux. Выбор среды разработки был обусловлен тем, что программное обеспечение, написанное с помощью Qt, запускается в большинстве операционных систем без изменения исходного кода, путем простой компиляции программы [15]. Включает в себя все основные классы, которые могут потребоваться при разработке прикладного программного обеспечения, начиная от элементов графического интерфейса и заканчивая классами для работы с сетью, базами данных и XML. Является полностью объектно ориентированным, расширяемым и поддерживающим технику компонентного программирования. Язык программирования – C++ [16]. Для работы с базой данных использовалась встраиваемая кросс-платформенная БД, которая поддерживает достаточно полный набор команд SQL – MySQL. Данная среда находится в общем доступе, не имеет ограничений на использование и обладает высокой степенью надежности.

Разработка системы

В ходе работы была создана подсистема ТПО, входящая в состав ПО КПА. Структурная схема ПО КПА представлена на рис. 2.

Подсистема ТПО состоит из следующих модулей:

- 1) интерфейс – обеспечивает удобство работы оператора с системой;
- 2) модуль взаимодействия с ядром системы для отслеживания хода работы с устройством. Для этого модуль должен использовать функции журналирования из состава каркаса ПО КПА;
- 3) модуль работы с устройством, для возможности отправлять и получать сообщения от устройства через отладчик;
- 4) модуль служебных функций обеспечивает ввод корректных данных и форматирования выводимых данных.

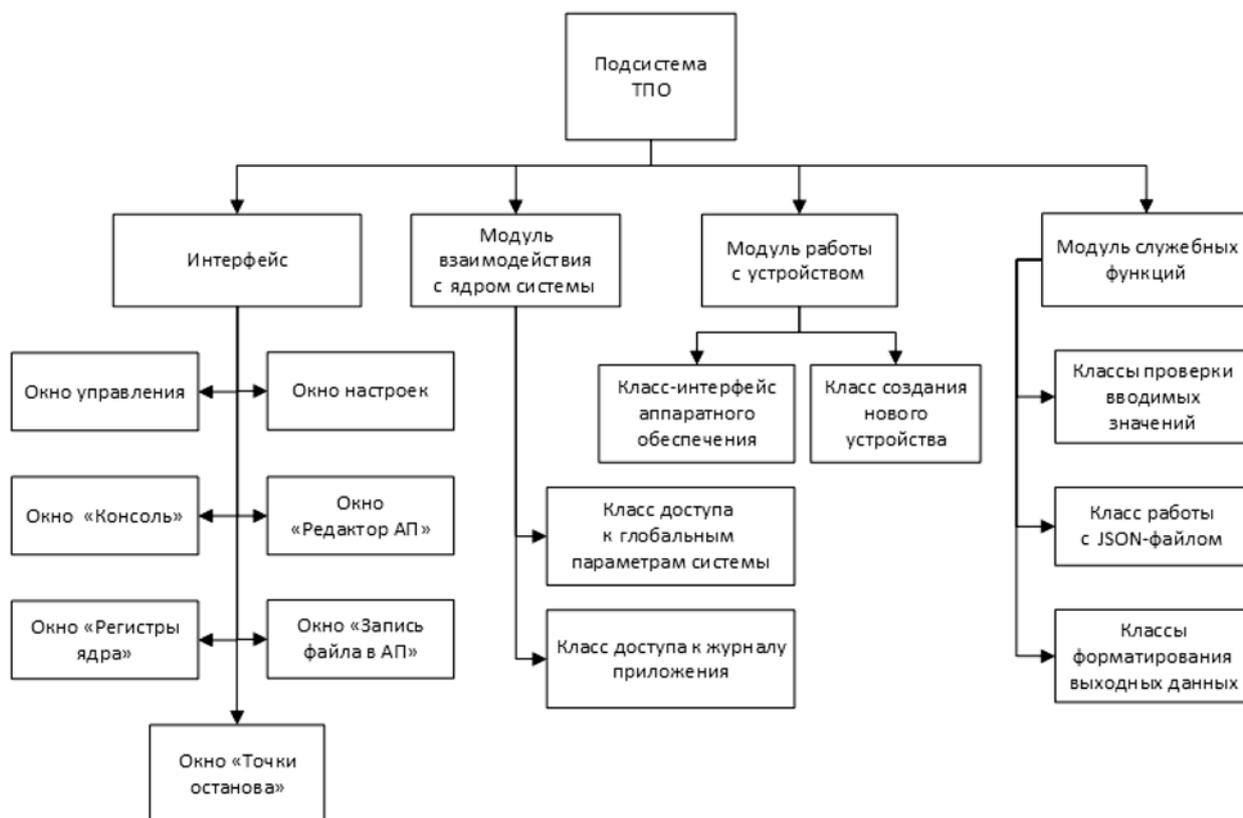


Рис. 2. Структурная схема ПО КПА

Fig. 2. Block diagram of the testing equipment software

Разработанная подсистема выполняет следующие функции:

1) ручной ввод команд для отладчика. В окне ввода содержатся: кнопки организации работы с журналом, позволяющие открыть ранее созданные журналы сессий работы с отладчиком, сохранить журнал в файл, а также очистить окно сообщений отладчика; область отображения процесса выполнения команд. Алгоритм обработки команд представлен на рис. 3;

2) просмотр значений выбранной группы регистров, а также обеспечивает возможность записи значений в регистры. Окно «Регистры ядра» содержит: область системы счисления с переключателями HEX (шестнадцатеричная) и DEC (десятичная), позволяющими выбрать систему счисления для отображения значений регистров; выпадающий список из двух групп регистров – регистры состояния и регистры общего назначения, переключаться между которыми можно по выбору из выпадающего списка соответствующего окна; поле отображения значений регистров; кнопку «Обновить таблицу регистров», по нажатии на которую происходит обновление значений регистров. Алгоритм работы окна представлен на рис. 4;

3) ввод параметров запуска, выбора конфигурационных файлов и настройки интерфейса;

4) просмотр, установка и удаление точек останова для отладки ПО. В полях окна «Редактор точек останова» записывается адрес, по которому необходимо установить точку. После нажатия кнопки «Установить», соответствующая точка записывается в список. Точкам, записанным в список, присваивается тип: программная точка останова имеет тип «soft»; аппаратная точка останова имеет тип «hard»; аппаратная точка наблюдения имеет тип «watch rw»;

5) запуск и остановку сессии работы с устройством;

6) чтение данных указанной длины по указанному пользователем адресу и позволяет записывать значения в выбранные пользователем ячейки памяти;

7) загрузка одного или несколько файлов в адресное пространство (АП) в поддерживаемых форматах, при этом проходит проверка их перед загрузкой. Окно «Загрузка в адресное пространство» содержит: кнопки управления списком загружаемых файлов в АП; таблицу файлов загрузки; кнопку «Восстановить список»; кнопку «Загрузить в память»; строку прогресса загрузки образа в память.

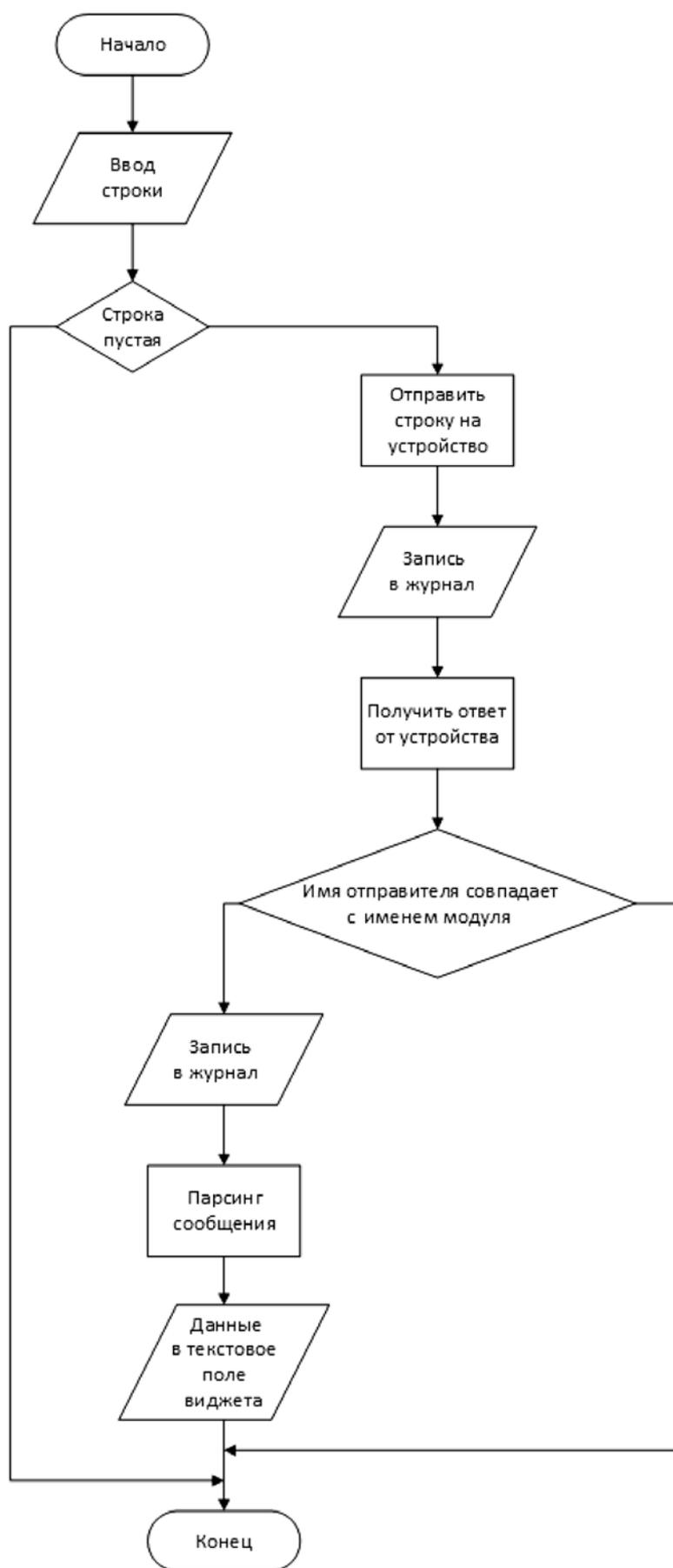


Рис. 3. Алгоритм обработки команд
Fig. 3. Algorithm for processing the commands

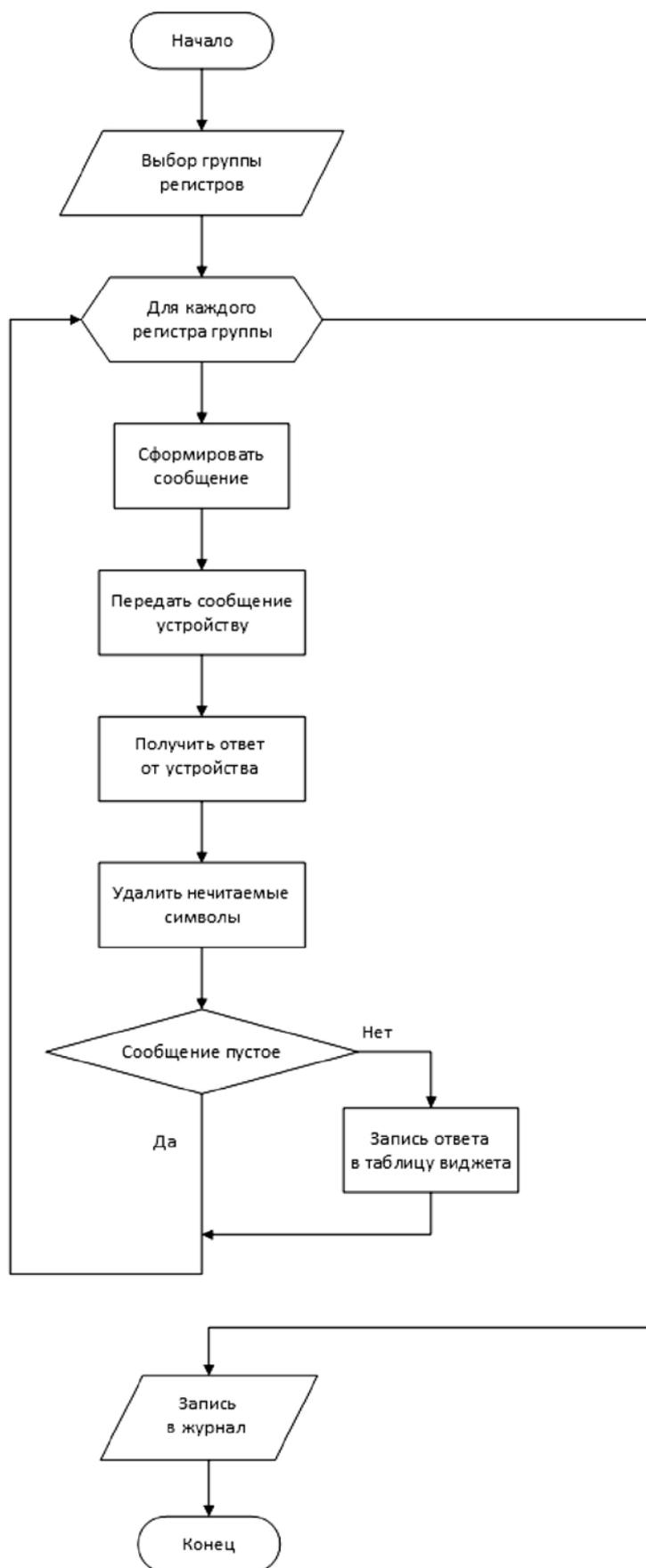


Рис. 4. Алгоритм чтения значений регистров
Fig. 4. Algorithm for reading the register values

Выводы

В ходе данной работы был создан модуль технологического программного обеспечения для работы с ПЛИС, были решены следующие задачи: переход с ранее использовавшегося отладочного программного обеспечения на новое, изменена архитектура подсистемы технологического программного обеспечения, созданы новые классы для работы с устройством, созданный модуль стал частью системы программного обеспечения контрольно-проверочной аппаратуры. Разработанная система внедрена на производстве и успешно справляется с поставленными задачами в ходе работы. В перспективе необходимо разработать класс функций для скриптов на языке JavaScript для реализации автоматизированных проверок бортовой аппаратуры хранения временных данных и минимизации участия человека в процессе проверок.

Библиографические ссылки

1. Шевчук Р. Б. Комплексы приема информации с российских спутников ДЗЗ // Геоматика. 2012. № 2.
2. Котюк А. Ф. Датчики в современных измерениях. М. : Радио и связь, 2006.
3. Белоблоцкая А. А., Поздова А. К., Тутов М. И. Разработка предложений по решению ключевых проблем создания высокоскоростных бортовых радиолиний // Молодой ученый. 2016. № 14 (118).
4. Zhang T. (2012). Instrumentation, Measurement, Circuits and Systems. Springer.
5. Космический комплекс «Ресурс-П». (2021). URL: <https://www.roscosmos.ru/24984>.
6. Graf R. F. (1996). Measuring Circuits, O'Reilly.
7. Janusz K. (2012). *Advances in Intelligent and Soft Computing*. Springer.
8. Williams R. (2015). *Algorithms for Circuits and Circuits for Algorithms: Connecting the Tractable and Intractable*. Computer Science Department, Stanford University, Stanford, CA, USA.
9. GRMON User's Manual. (2021). URL: <https://www.cse.wustl.edu/~roger/465M/grmon-1.0.5.pdf>.
10. OpenOCD User's Guide. (2021). URL: <http://openocd.org/doc-release/pdf/openocd.pdf>.
11. Brown S. D. (2011). *Fundamentals of Digital Logic with VHDL Design*. 2nd ed., McGraw-Hill.
12. Intel Quartus Prime Standard Edition User Guide. (2019).

13. Logisim, a graphical tool for designing and simulating large circuits. URL: <http://www.cburch.com/logisim> (2021).

14. Xilinx, ISE Design Suite 14: Release Notes, Installation, and Licensing. (2020).

15. Ogrodzki, J. (2018). *Circuit Simulation Methods and Algorithms*, CRC Press.

16. Шлее М. Qt 4.8. Профессиональное программирование на C++. СПб : Изд-во БХВ-Петербург, 2012.

References

1. Shevchuk, R.B. [Complexes for receiving information from Russian remote sensing satellites]. *Geomatics*. 2012. No. 2 (in Russ.).
2. Kotyuk A.F. *Datchiki v sovremennykh izmereniyakh* [Sensors in modern measurements]. Moscow: Radio and communication, 2006 (in Russ.).
3. Beloblotskaya, A.A., Pozdova, A.K., Titov, M.I. (2016). [Development of proposals for solving key problems of creating high-speed airborne radio lines]. *Molodoi uchenyi*. 2016. No. 14 (in Russ.).
4. Zhang, T. (2012). Instrumentation, Measurement, Circuits and Systems. Springer.
5. Space complex "Resource-P". Available at: <https://www.roscosmos.ru/24984/> (2021).
6. Graf R. F., (1996). Measuring Circuits, O'Reilly.
7. Janusz K. (2012). *Advances in Intelligent and Soft Computing*. Springer.
8. Williams, R. (2015). *Algorithms for Circuits and Circuits for Algorithms: Connecting the Tractable and Intractable*. Computer Science Department, Stanford University, Stanford, CA, USA.
9. GRMON User's Manual. (2021). Available at: <https://www.cse.wustl.edu/~roger/465M/grmon-1.0.5.pdf>.
10. OpenOCD User's Guide. (2021). Available at: <http://openocd.org/doc-release/pdf/openocd.pdf>.
11. Brown, S. D. (2011). *Fundamentals of Digital Logic with VHDL Design*. 2nd ed., McGraw-Hill.
12. Intel Quartus Prime Standard Edition User Guide. (2019).
13. Logisim, a graphical tool for designing and simulating large circuits. Available at: <http://www.cburch.com/logisim/> (2021).
14. Xilinx, ISE Design Suite 14: Release Notes, Installation, and Licensing. (2020).
15. Ogrodzki, J. (2018). *Circuit Simulation Methods and Algorithms*, CRC Press.
16. Schlee, M. Qt 4.8. (2012). Professional programming in C ++. St. Petersburg: Izd-vo BKhV-Peterburg.

Development of a Technological Module for Control and Verification of on-Board Equipment for Storing Temporary Data

D. D. Bazhenova, “RosA” LLC, Izhevsk, Russia

M. Aiman Al Akkad, PhD in Engineering, Kalashnikov ISTU, Izhevsk, Russia

A. A. Ivakin, “IRZ” LLC, Izhevsk, Russia

High-speed radio link HSRL is designed to transmit target information from spacecraft equipment to the ground. A block of onboard equipment for storing temporary data OESTD is a part of the onboard equipment of a high-speed radio link OEHSRL. Before the spacecraft is launched into space, acceptance tests of the input control are carried out. To do this, it was necessary to develop testing equipment TE and software for it. TE of OESTD is designed to check the OESTD in general and each block in particular during autonomous tests. This paper considers the subsystem of technological software – a component of the TE software system, which allows checking the operation of the FPGA as part of the OESTD. The subsystem main algorithms and functions performed by the subsystem are given. The interaction of the operator of control and testing equipment with programmable logic integrated circuits FPGA, which are part of the on-board equipment block for storing temporary data of the OESTD, is considered. Debugging software is required to enable this interaction. An FPGA with the RISC-V architecture was chosen, debugging via GRMON turned out to be impossible and OpenOCD was chosen. As a result, a technological software module was developed for testing and ensuring the operability of the FPGA as part of the onboard equipment for storing temporary data. The following components were developed: a subsystem for interaction with the device to ensure the ability to send commands and receive response messages, service functions to convert response messages into a readable form for the operator, a subsystem for interaction of the module with the main frame of the TE software, and widgets to provide the ability to manually enter commands from the user conveniently.

Keywords: onboard equipment for storing temporary data, control and testing equipment, technological software, FPGA, high-speed radio link.

Получено: 11.08.2021