Debugging method for spacecraft’s equipment based on ARM processors

E V Tabakov1,2,3, A I Zinina2 and D S Kireev1,2

1Design Bureau «Mars», 16 1st Shemilovsky lane, building 1, Moscow, 127473, Russia
2National Research University Moscow Aviation Institute, 4 Volokolamskoe avenue, Moscow, 125080, Russia
3E-mail: e-tab@rambler.ru

Abstract. This paper concerns the problem of onboard spacecraft control system equipment debugging in the part of providing correct operation of onboard software. The disadvantages of existing software tools for debugging are considered. The new method of interaction with ARM processor-based microcontrollers is proposed. This method is based on using OpenOCD and QT tools and allows to avoid considered disadvantages. The new software is designed based on this method. The main modes of its operation are described.

1. Introduction
Designing of spacecraft control system (CS) is extremely difficult and actual task [1]. During this designing significant attention is paid to the CS’s reliability. Spacecraft CS is based on the equipment which operates by software execution [2]. Important feature of the CS equipment is its redundancy — any equipment unit usually has duplicating analogs [4]. So, for example, onboard computer (OBC) consists of several similar programmable logic controllers (PLCs). In every PLC of CS equipment computational units based on ARM architecture processors are used. It is obviously that the CS functioning depends on normativeness of its equipment operation. Therefore, plural tests of this equipment are implemented at the testbeds.

During such tests there is the debugging task, that is providing of normative functioning for every equipment unit. One of the most important parts of the debugging is control of embedded software normativity operation. This control is implemented via special software tools, so-called debuggers, that allow read out and record data at the microcontroller memory devices and processor registers. It should be note, that so far at enterprise debuggers were used that have a number of disadvantages. These disadvantages reduce the convenience of equipment debugging and increase time. The main disadvantages are inability to implement the debugging in real time and deal with multiple microcontrollers simultaneously by using the same software instance.

Therefore, the development of new method for interaction with microcontrollers that are based on AMD processors is actual, that will allow to eliminate described disadvantages.

2. Onboard spacecraft CS
The spacecraft CS consists of the following components:

- OBC
• Power automation units
• Information-measuring elements (IME)
• Executive elements

The central part of the CS is OBC that execute onboard software modules. During the OBC operation it interacts with different CS parts and other spacecraft equipment [6]. OBC consists of four similar PLCs.

Information-measuring elements are sensors that collect data about spacecraft spatial orientation and its angular velocity and transmit them to OBC. Depending on the mission design, commands from Earth and data from IME the OBC transmit a control signals to executive elements. CS executive elements are flywheel engines.

3. Features of spacecraft CS element testing
In order to provide the reliability of spacecraft CS operation it is thoroughly tested on the ground. These tests are implemented in accordance with special technology, that comprises tests of CS logic part as well as hardware. During the tests of logic part there is no necessity to check electric interactions between equipment units. Such group of the test is implemented at the hybrid system testbed.

At the electrical functional model testbed complex CS testing is implemented focusing at interaction of equipment with other spacecraft systems.

One of the most important points during CS ground testing is the checkout of onboard software operation and correction of its execution. This is implemented by reading out of microcontroller random access memory and changing of this memory. Modification of program behavior is carried out by the changing the content of those memory cells in which processor commands are recorded. The reading out the memory cells allows to follow program behavior as well as to get values of parameters that characterize OBC operation. For example, in this way we can find out the value of parameter that reflect OBC configuration.

The debuggers allow the direct access to microcontroller memory. Their using during testing of spacecraft CS equipment gives enormous opportunities in the part of direct access to hardware [8]. The debuggers that were used at the enterprise so far during the testing of the CS equipment had the restricted functional capabilities. For instance, their functionality doesn’t allow simultaneously deal with several microcontrollers of the same computational devices. Meanwhile, such operation is very important. The PLCs of the computational device continuously perform plurilateral data communications. Because of that simultaneous operation with all microcontrollers is opened new opportunities.

Also, the previously used debuggers had additional disadvantage — they are interrupted the software execution while reading out the data. In addition, some debuggers could operate only on obsolete operating system DOS.

Thus, the task is appeared that concerning development of new debugger that is based on the new method of interaction with ARM architecture processors, that won’t have disadvantages described above.

4. Proposed method
Proposed method consists of following. The debugger client part is implemented in QT framework. During operation, client part that installed at programmer’s work computer send by TCP/IP protocol via enterprise local network commands corresponding OpenOCD standard in the part of interaction with ARM processors. The OpenOCD tool has been chosen due to its broad means for operation with microcontrollers [7]. Computer, that installed at the testbeds get these commands. This computer is connected with CS equipment. After the getting commands it provides interaction with connected microcontrollers via software OpenOCD.

OpenOCD tools are used to make it possible to read out memory cells content and change it. Also, in accordance with proposing method, due to OpenOCD there is window of opportunity in the part of
processors calculation management. For example, it is possible to run and stop processor. As well, we can see the command that will be done by processor and change it.

The interaction with databases is also implemented to provide the opportunity in the part of view whole values of parameters, that are recorded in microprocessor memory.

5. Software operation description

After the run of the program user sees window that is shown in figure 1. It should be note, that there are five modes of program operation:

- «MemoryContent»;
- «MemoryContent(Table)>>;
- «CellList»;
- «Processors»;
- «Registers».

The program is run in «CellList» mode, that allows to read out and change the content of memory cells and specific variables that are recorded in these cells. In this figure is shown the window view for the case when there are no connected microcontrollers.

![Figure 1. Start window.](image)

The view of the table in the «CellList» mode for the case when there is connected microcontroller is shown in figure 2. In the left column is written addresses of memory cells. Other columns are contained the values that are recorded in these cells for every microcontroller. There can be added additional microcontrollers by the appending of columns. Also, there can be added additional addresses by the appending of rows.

In the figure 3 is shown the view of information in «Registers» mode. This mode allows to implement the checking of the processor registers.
The view of the window in «MemoryContent» mode is shown in figure 4. This mode provides extended opportunities for operation with memory. The tools of this mode allow to choose the address for start point of reading, size of reading word and numeral system in which the results of reading are displayed.

Also, to meet the demands of users has been added the «MemoryContent(Table)» mode. It provides the same information as «MemoryContent» mode. However, in this mode the information is displayed in table view (figure 5).

The «Processors» mode allows to manage calculations in processor (figure 6). By this mode is possible to stop and run microprocessor. Also, it is possible to track current command that is performed by processor and to set the break points. At break point the processor is terminated its operation and by using this we can to see the process of software execution «step by step». As it is shown in figure, the debugger display information about commands, recorded in memory cells in disassembler.
6. Conclusion

The features of onboard spacecraft CS equipment design is considered in this work. The concept debugging of onboard equipment in the part of providing normative operation of onboard software is revealed. The disadvantages of the software tools that previously was used for debugging are analyzed. The new method of debugging implementation is proposed. This method consists in using opportunities of OpenOCD and QT to create the debugger that will meet the given requirements.

The new debugger is designed based on this method. Its main operation modes are considered. As the result, the instrument for ARM processors-based microcontrollers debugging is obtained. This instrument satisfies the given requirements.

References

[1] Shen Q, Yue C, Goh C H and Wang D 2019 Active fault-tolerant control system design for spacecraft attitude maneuvers with actuator saturation and faults IEEE Transactions on Industrial Electronics 66(5) 3763-72
[2] Leveson N 2004 Role of software in spacecraft accidents Journal of spacecraft and ROBCkets 41(4) 564-75
[3] Brovkin A, Burdygov B and Gordiijko C 2010 Onboard spacecraft’s control systems ed Syrov A 3 (Moscow: MOKB «MARS») p 29
[4] Nozhenkova L, Isaeva O and Vogorovskiy R 2017 Scenario approach to testing spacecraft’s onboard equipment command and software management DEStech Transactions on Engineering and technology research (Beijing, DEStech) pp 53-7
[5] Andeeev V P, Volovich N V, Glebov I V et al. 2011 Designing and Testing Onboard Control Systems (Moscow: MAI-PRINT) pp 151-6
[6] Eickhoff J 2011 Onboard computers, onboard software and satellite operations: an introduction (Berlin: Springer Science+Business Media) p 6
[7] Titzer B and Falsberg J 2005 Nonintrusive precision instrumentation of microcontroller software Proceedings of the 2005 ACM SIGPLAN/SIGBED conference on Languages, compilers, and tools for embedded systems 59-68
[8] Puri M and Jitendra S 2013 OpenOCD: The Yoke to ARM Programming on Ubiquitously Available Windows Platform Dept of Electronics and Communication Engg vol 1(Tumkur: SSIT)