MEXBIOS DEVELOPMENT STUDIO SOFTWARE PACKAGE FOR DEVELOPING CONTROL PROGRAMS AND MODELING ELECTRIC DRIVE SYSTEMS

Fayzullayev Dedakhuja Zokirjon oglu
Namangan Institute of Engineering and Technology
Assistant of the department "Automation and control of technological processes"
E-mail: dedaxujafayzullayev@gmail.com

Annotation
The concept of the MexBIOS system implies the presence of a "starter" project of embedded software for the microcontroller (which can also be the existing projects of the system user). The "starter" project, in addition to its own functions, launches a specialized MexBIOS software kernel, which in turn launches the library elements in accordance with the rules defined by the user in the process of graphical programming on a personal computer. In fact, the user must specify the composition of the executable elements of the library, the conditions for their launch and the data flows between the inputs and outputs of the applied elements of the library. The set of these rules will be referred to as the "matrix". In fact, the core of the system is an interpreter that calls pre-compiled program blocks in accordance with the program created in a graphical way. The interpretation of the code (even if it operates with precompiled procedures) naturally somewhat slows down the execution of the program, storing the "matrix" in the data memory reduces its "useful" capacity, but all this can be compensated through the built-in mechanism for generating code in text form after the debugging process is completed.

Keywords: MexBIOS, matrix, system kernel, Dial field, graphical controls, programming subsystem, viewing/setting.

Introduction

The MexBIOS Development Studio environment is a software product for developing and debugging software for microprocessor controllers and supports many ways to create software:

– Text procedural programming in C;

– Graphical programming in the form of functional block diagrams;

- Graphical programming of algorithms, taking into account the branching and the sequence of execution of modules, in which the chains of functional block diagrams

are located;

– Automatic programming approach;

- Event programming.

In the microcontroller, the user periodically calls the system kernel, which, together with the library of program blocks, must first be loaded into the microcontroller's memory. The kernel executes the execution of program blocks in accordance with the information (bytecode) provided by the user. Program blocks are compiled by standard compilation tools and loaded by the user before starting the kernel. The bytecode is formed by the user through graphical programming, which indicates the events and conditions for starting algorithms, algorithms in the form of block diagrams, functional block diagrams, as well as the order of their execution and the addresses for reading / writing data of program blocks. As a result of the execution of program blocks in accordance with user algorithms, the data in the microcontroller memory changes as a function of incoming data and events.

The MexBIOS software package consists of the following main parts:

– Integrated environment MexBIOS Development Studio;

– Library of blocks;

- Nucleus;

– "Starter" project;

- A set of ready-made templates for design.

The integrated environment (IS) allows the development of a program for the microcontroller and its debugging. There are two operating modes of the integrated environment: simulation and data exchange with the microcontroller. Simulation mode is designed to simulate the work of the created program on a computer. For the modeling mode, a special library Models has been created, which is designed to create mathematical models of regulated objects. The mode of data exchange with the chip is designed to work with hardware loaded with the MexBIOS Kernel core. The core is a library of blocks (each series of processors has its own library) and the core of the MexBIOS real-time operating environment.

According to the program created and tested in the simulation mode, a configuration file is created, which is loaded into the chip's memory. After loading the configuration file into the memory of the MexBIOS Kernel chip, according to the loaded data, it creates a structure identical to the visual program compiled in the MexBIOS Development Studio. The operation of the programs is identical, after establishing a connection with the chip, you can read and set the parameters in the chip.

The main components of the integrated environment:

1. Dial field.

2. Component library: graphical controls, built-in components, model blocks, block library, data protocols.

3. Module for creating blocks.

4. Inspector of block properties.

5. Panel for simulating the operation of the microcontroller interrupt system.

6. Project manager. Block search function.

7. Simulation mode setting window.

8. Library compilation settings window.

9. Window for viewing/setting variable values.

10. Window for viewing project parameters.

The startup project launches the MexBIOS system in the microcontroller. The start-up project binds the system to the microcontroller and circuit solutions of the electronic board on which the microcontroller is installed. The startup project depends on the type of microcontroller used. The start project contains a call to the MexBIOS system kernel and provides communication with the IS. The manufacturer attaches an example of a start-up project to the system for supported types of microcontrollers. The MexBIOS system kernel can receive data from the start project and transfer data to the start project by means of special blocks. As a start project, the user can use his own project, including the MexBIOS core and the block library.

The kernel performs the execution of program blocks in accordance with the order and launch conditions predefined by users through graphical programming, and, in fact, is a bytecode interpreter. Because the kernel runs precompiled routines, overall system performance is comparable to projects created by full compilation. Block libraries are created by text programming followed by compilation. The library of blocks is loaded into the memory of the microcontroller, in parallel with this, a similar library of blocks is formed to simulate the operation of the system on a personal computer. Libraries contain standard groups of blocks that are most often used when designing software for microcontrollers, in particular, for control systems.

electric motors. The creation of blocks and the formation of libraries are carried out in IS. A set of ready-made templates is a package of ready-made application software projects for the MexBIOS system that greatly simplifies software development.

MexBIOS Development Studio provides two modes of program operation: simulation and work with a chip.

1) Simulation (simulation) is actually the process of simulating on a computer the operation of software in a controller with MexBIOS in the absence of a real controller. This active mode is characterized by the execution of the program in the address space

of the MexBIOS Development Studio environment on the embedded kernel. At the same time, MexBIOS has access to an array of global data.

2) Working with a chip is the process of controlling and monitoring the operation of a real device with a loaded MexBIOS program that controls a real technological process. This mode is characterized by the execution of the program in the control controller.

To switch to this mode, you must establish a connection with the microcontroller and start the update (Start Refreshing). In this mode, MexBIOS Development Studio does not have direct access to the global data array of the MexBIOS device. WatchWindow, oscilloscope and visual organs are updated using the transport protocol through regular requests to the device, while the user gets the opportunity to influence the operation of the program in the controller running MexBIOS in real time by changing the program variables.

References

1. Сосонкин, В.Л. Принцип подчиненного управления в логических системах управления /В.Л. Сосонкин, В.И. Клепиков // Приборы и системы управления.1995.- №12. - c.16-18.
2. Frank, D. Petruzella: Programmable Logic Controllers, 3rd edition 2005 copyright ISBN: 0-07-829852-0
3. Харазов, В.Г. Интегрированные системы управления технологическими процессами / В.Г. Харазов // СпБ.: Профессия, 2009. - 592 c.
4. Джураев Шерзод Собиржонович, Мамаханов Аъзам Абдумажидович, Шарибаев Носир Юсубжанович, Тухтасинов Даврон Хошимжон Угли, Тулкинов Мухамадали Эркинжон Угли Логическое реле Owen для автоматизированной системы управления // Universum: технические науки. 2020. №8-1 (77). URL: https://cyberleninka.ru/article/n/logicheskoe-rele-owen-dlya-avtomatizirovannoy-sistemy-upravleniya (дата обращения: 24.05.2022).