УДК 004.934:681.391

THE CHOICE OF OPTIMAL SOLUTION FOR THE MICROCONTROLLER SELECTION TO CREATE AN AUTONOMOUS MACHINE CONTROL SYSTEM

***Lysenko V.V.* –** group МгЧКІ-20, master, *lysenko.vv@knutd.edu.ua*
***Dmytryuk S.V.* –** senior instructor, *svitmartal73@gmail.com*
Kyiv National University of Technologies and Design

Дана доповідь присвячена пошуку і розгляду готових рішень, аналізу асортименту на ринку мікроконтролерів та оптимального вибору для автономної системи керування верстатом з числовим програмним управлінням.

The article deals with the search and consideration of ready-made solutions, the analysis of the range of microcontrollers at the market and the optimal choice for an autonomous machine control system with numerical program control.

Introduction. With the rapid development of technology, small CNC milling machines, laser engravers and inexpensive 3D printers are gaining popularity, they are bought as hobby machines, for children as toys, stimulating children's creativity, or more expensive to create souvenirs and opportunities to start their own business.

Building up the most budget machine with numerical program control I faced the problem of self-sufficiency, because the machine requires a computer with specialized software installed on it, it consumes electricity, takes up space, and must be constantly connected to both the machine and the power supply. Certainly, expensive machines have a system for reading the control program from a memory card while cheap machines have no alternative. Having not found a ready-made budget solution, we will try to offer our own.

Problem statement. It is necessary to understand what a microcontroller is, what types of them exist andto find an inexpensive functional microcontroller to build an autonomous system.

The issue of microcontrollers is studied by many scientista, in particular: Korolev N., Malakhov V., Plyasunov A., Yakovlev D., but they studied the microcontrollers themselves, and not autonomous systems for machines. Investigating the topic of autonomous systems for CNC machines, it turned out that there are very expensive ready-made solutions of dozens of thousandshryvnias, there are expensive machines with built-in autonomous systems, but there are not so many of them. Thus, the topiccontinues to berelevant today.

The purpose of the article is to study the available microcontrollers and to select the optimal one for building an autonomous control system for a CNC machine.

Research findings. In 1967, Texas Instruments released the first integrated circuit calculator, which launched the process of computer miniaturization.With the development of technology, microcontrollers began to develop rapidly in the industry and they became cheaper and more affordable.Nowadays, there are more than a dozen companies producing microcontrollers, such as: Intel, Motorola, Zilog, Altera, Infineon, Texas Instruments, etc. The most common are PIC controllers from Microchip, AVR controllers from Atmel, and the joker in the pack at the market is STM controllers from STMicroelectronics, they are rapidly displacing other manufacturers, producing cheap microcontrollers, with parameters superior to competitors.

Each company produces a variety of microcontrollers of different architectures, so the developer faces a serious problem of choice. I chose the AVR family, it is more common, and has a large number of fairly simple software development environments.

First of all, let's deal with the very concept of "Microcontroller". The microcontroller can be defined as a miniature computer based on a single chip that includes, in addition to the processor, a number of auxiliary elements, such as: random access memory "RAM", programmable read only memory "PROM", timer, etc. The microcontroller is designed to perform any predefined tasks [8].

The easiest way is to compare the Microcontroller with a personal computer, then just a PC. Like a PC, the Microcontroller has a processor, RAM and ROM. However, unlike a personal computer, all of these elements are located on a single chip.

The PC is designed to perform a large general purpose task. For example, you can use a computer to type a text, write programs, store and run video files, surf the Internet, etc. while microcontrollers are designed to perform specific tasks such as switching the air conditioner when the room temperature drops below a certain value, or vice versa rises above.

Atmel AVR Microcontrollers (MCs) are 8-bit microcontrollers designed for embedded applications. Microcontrollers are manufactured on a low-consumption CMOS technology, which in combination with the advanced RISC architecture allows to achieve the best ratio of performance / energy consumption. Due to the fact that most commands are executed in one clock cycle, the speed of these microcontrollers can reach 1 MIPS (millions of operations per second) per l MHz clock speed.

Within the framework of a single basic architecture, AVR MKs have three families:

– Classic - baseline MK;
– Mega -MK for complex applications;
– Tiny - cheap MK in the 8-output case.

Let's consider in more detail the distinctive features of each family.

There are features that are common to all three families:

– the ability to calculate with the speed up to 1 MIPS / MHz;
– the ability to protect against reading and modifying program memory and data (in EEPROM);
– various synchronization methods: built-in PC-generator, external synchronization signal or external resonator (piezoceramic or quartz);
– the availability of several modes of reduced energy consumption;
– the possibility to program directly in the system through the serial SRI-interface.

Distinctive features of MK of the Classic family:

– FLASH programs memory with the capacity of 1 to 8 KB (the number of erase cycles/ writing cycles not less than 1000);
– datamemorybasedonstatic RAM withthecapacityofupto 512 bytes;
– data memory based on EEPROM with the capacity of 64 to 512 bytes (the number of erase cycles/ writing cycles not less than 100000);
– the ability to protect against reading and modifying program memory and data (in EEPROM);
– programming in parallel (using a programmer) modes.

Distinctive features of MK of the Mega family:

– FLASH programs memory with the capacity of 8 to 128 KB (the number of erase cycles / writing cycles not less than 1000);
– data memory based on static RAM with the capacity of 1 ... 4 Kbytes;
– data memory based on EEPROM with the capacity of 512 bytes ... 4 Kbytes (the number of erase cycles/ writing cycles not less than 100000);
– the possibility of self-programming;
– the possibility of internal system debugging according to the IEEE1149.1 standard;

– the availability of a supply voltage reduction detector;

– the possibility of programmed reduction of frequency of the clock generator.

Distinctive features of MK of the Tiny family:

FLASH programs memory with the capacity of 1 to 2 Kbytes (number of erase cycles/ writing cycles not less than 1000);

– data memory based on static RAM with the capacity of 1 ... 2 Kbytes;

– data memory based on EEPROM with a capacity of 64 bytes (the number of erase cycles/ writing cycles not less than 100000);

– some MK models can work when the supply voltage is reduced to 1.8V [6].

First of all, the microcontrollers of this series are high-speed. Many instructions are performed in one cycle by the Microcontroller processor while AVR microcontrollers are about 4 times faster than the PIC. Besides, they consume much less energy and can operate in several energy saving modes.

Many AVR controllers are 8-bit, although there is a 32-bit version of the AVR32 controllers. In addition, as noted above, AVRs belong to the type of RISC-Microcontrollers.The RISC (Reduced Instruction Set Computer) architecture is the computer with a reduced set of instructions but at the same time it gives an advantage in speed. The analogue of the RISC architecture is the CISC (Complex Instruction Set Computers) architecture.

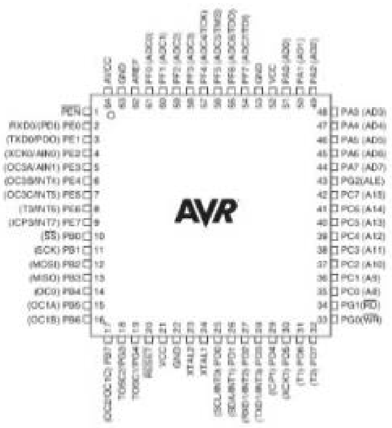

Fig. 1. 32-bit version of AVR32 controllers

The 8-bit controller implies that it is capable of transmitting and receiving 8-bit data. The provided Input / Output registers are also 8-bit. The controller architecture is based on registers.This means that registers are used to store the initial values of the operation and the result in the controller.

The controller processor receives data from the two input registers, it performs a logical operation and stores the result in the output register. All this takes 1 executable cycle.

In total, the AVR controller has 32 of 8-bit general-purpose registers. During the cycle, the processor takes data from two registers and uses them in an arithmetic-logic unit (ALU), which performs calculations with data and places them in a random register.The ALU can perform both arithmetic and logical operations with some registers.The ALU can also perform actions with one register.

In this case, the controller does not have an operand - the battery, unlike the controllers of the family 8051 - for which any operands can be used, and the result of the operation can also be placed in any operand.

The microcontroller corresponds to the Harvard computing architecture, according to which the computer has independent memory for programs and data. Therefore, while one operation is being performed, the next operation is pre-retrieved from the memory.

The controller is able to perform one operation per cycle. Due to this if the clock frequency of the microcontroller is 1 MHz, then its performance will be 1 million operations per second. The higher the clock speed of the controller, the higher its performance. However, when choosing the clock frequency of the controller, there must be a trade-off between its speed and power consumption.

In addition to flash memory and processor, the controller has peripherals such as Input / Output ports, analog - to - digital converter, timers, communication interfaces - I2C, SPI and serial port UART.All these peripherals can be controlled at the software level.

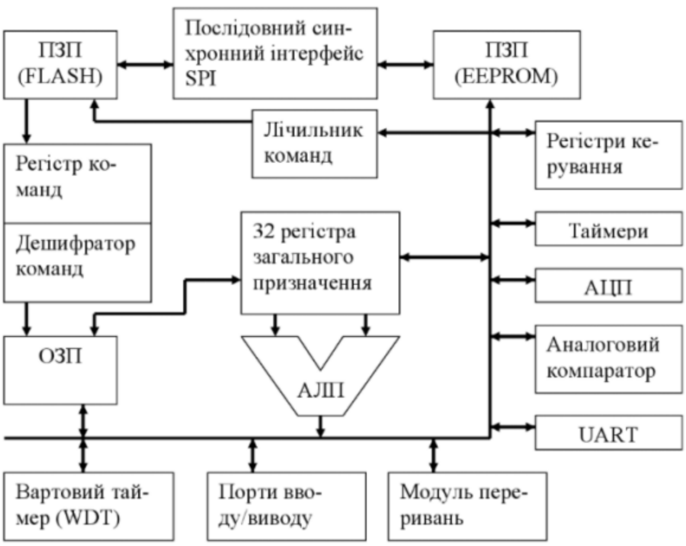

Fig. 2. Typical kernel architecture of AVR microcontrollers [6].

Conclusions. As a result of this work, various microcontrollers were considered, their structure and the principle of operation were clarified, and a family of microcontrollers was selected to be used for building up an autonomous system for CNC machine. We may conclude that to build up an autonomous system for CNC machines, the MegaAVR family is ideal. It is more than enough to read commands from a SD card and transmit them via the machine's serial port.

References

1. Korolev, N., Korolev, D. (2003). AVR-mikrokontrollery vtorogo pokoleniya: sredstva razrabotchika [AVR microcontrollers of the second generation: development tools]. *Komponentyitekhnologii - Components and technologies,* № 7. – P. 112 [in Russian].

2. Korolev N., Korolev, D. (2003). AVR-mikrokontrollery vtorogo pokoleniya:
novye apparatnye vozmozhnosti [AVR microcontrollers of the second generation: new hardware features]. *Komponentyitekhnologii - Components and technologies,* № 4. – P. 117 [in Russian].

3. Korolev, N., Korolev, D. (2001). AVR-mikrokontrollery vtorogo pokoleniya: bol'shoe v malom [AVR microcontrollers of the second generation: big to small]. *Skhemotekhnika – Circuitry,* № 5. – P. 8-17 [in Russian].

4. Korolev N., Korolev, D. (2000). AVR-mikrokontrollery: programmnye sredstva [AVR microcontrollers of the second generation: software]. *Komponentyitekhnologii - Components and technologies,* № 4. – P. 14 [in Russian].

5. Korolev N., Korolev, D. (1999). AVR: apparatny esredstva razrabotchika [AVR microcontrollers of the second generation: developer hardware]. *Komponentyitekhnologii - Components and technologies,* № 1. – P. 30 – 33 [in Russian].

6. Malakhov V. P., Yakovlev D. P. (2008). Mikrokontrolery: navch. posib. [Microcontrollers: teaching guide]. O: Naukaitekhnika- Science and Technology. 222p [in Ukrainian].

7. Plyasunov A. V. (2001). 0b odnompodkhode k resheniyu zadach dvukhurovnevogo programmirovaniya [On one approach to solving problems of two-level programming]. *Trudy XII Baykal'skoymezhdunar. konf. «Metody optimizatsii i ikh prilozhenii» - Proceedings of the XIIth Baikal international conference "Optimization methods and their applications."* Irkutsk, P. 227–231 [in Russian].

8. Sommer U. (2012). Programmirovanie mikrokontrollernykh plat Arduino/Freeduino [Microcontroller-Programming with Arduino/Freeduino]. SPb. : BKhV – Peterburg. 256p [in Russian].