Scientific Papers of the
Institute of Electrical Power Engineering of the
Wrocław University of Technology

PRESENT PROBLEMS OF POWER SYSTEM CONTROL

No 1

Wrocław 2011

Keywords:
microprocessor controller,
Programmable Logic Controller

Janusz STASZEWSKI*

MICROPROCESSOR CONTROLLERS VERSUS PLC

This paper describes microprocessor controllers and Programmable Logic Controllers for industry applications. The differences in hardware and in programming are presented.

1. INTRODUCTION

Ones upon a time… there were transistors and the first control systems with many problems, with noise and low level signals. Next were digital control systems using logical gates and memory elements like flip-flops, etc. These systems are more resistant to disturbance and signals can be sent to long distances. But there is still only one problem. Every time we want to change control system, we must add new logic elements and change connections between them. So, a man has thought "What can I do to change a control circuit without changing hardware? The solution was only one: a universal structure with many logical gates, flip-flops etc. and connections between them changed by special algorithm (program).

2. MICROPROCESSOR CONTROLLER

2.1. HARDWARE

This special programmable structure is called microprocessor. It sounds very simple but, in fact it is very complex. The typical structure of microprocessor is shown in

* Wroclaw University of Technology, Institute of Electrical Power Engineering, Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland.

Fig. 1. The main part CPU (Central Processing Unit) is called ALU (Arithmetic Logic Unit). In this unit all arithmetical and logical operations are executed. This, what is executed (program) is stored in the block called Memory. The Memory is connected with ALU by unidirectional Address Bus and bidirectional Data Bus. We can also use the Memory to read or write variables. Very important is a block called Input/Output. It is used for communication of ALU with external world. This block is connected with ALU in the same way as Memory. On the rising edge of the special signal called Read, when the appropriate address of external module is present on the Address Bus, all the signals on the microprocessor Inputs are read. On the rising edge of the special signal called Write, when the appropriate address of external module is present on the Address Bus, all signals from ALU are latched in the microprocessor Outputs. Signals Read and Write are the same to operate Memory and Input/Output Block. To distinguish these signals for external devices (by Input/Output Block), a special signal called Chip Enable is created. These signals (Read, Write, Chip Enable) are managed by Control Block. This block contains a crystal oscillator too. Program execution is synchronized by clock of Control Block.

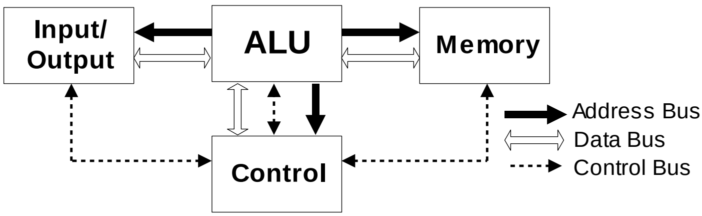

Fig. 1. The typical structure of microprocessor

2.2. SOFTWARE

First microprocessor programmes were very sophisticated. There were binary sequences (4 or 8 bits) - difficult to write and to remember. The next notation of a program is called assembler. Each binary sequence has its own mnemonic code. This code is similar to English words, for example: *move*, *ld* (load), *call*, *jmp* (jump) etc. But still programming is not so easy. There is one programme instruction for each hardware, arithmetical or logical operation. Only addition and subtraction operations are implemented in basic microprocessor. To obtain the other arithmetical operations for example multiplication, division or square rotation one must write a lot of basic instructions.

It is impossible to program well without perfect knowledge on microprocessor hardware. Many years ago somebody said: “To understand microprocessor it is necessary to know memory management and interrupt structure”. And that is true. There are two popular memory architectures: von Neumann architecture and Harvard architecture. The first one uses only one memory space, divided into three areas: instructions, data, and input/output. The second one uses separate memory space with almost the same addressing for instructions, data, and input/output areas, but with common memory and data buses. It means that there are three memory areas with exactly the same address. To distinguish these cells we have to use different instructions or different kinds of addressing. There are two ways of addressing: direct and indirect with many variants. For example in direct addressing we can use simply the address of register or the name of this register.

The most difficult in microprocessor management are interrupts. Interrupts service external events. The simplest example of interrupt using: we have to check voltage level of battery. Without interrupt, it is difficult to say how often we should check the voltage: once a week, but maybe once a minute. Once a week, it seems too seldom, because between two periods of checking, the battery can be discharged. Once a minute seems good, but it takes too much time of microprocessor working. Interrupt is in this example most convenient. The level of battery voltage is controlled by external analogue comparator with digital output and microprocessor is working normally without battery checking. When the voltage is too low, the comparator generates special digital signal to microprocessor interrupt system. And only now microprocessor breaks standard program to service this problem. There are many sources of interrupts and problems, when two (or more) interrupts are coming simultaneously. User should establish priority of interrupts. The interrupt service is the same like the subroutine service. So we should declare interrupt vector (assigning addresses in memory space) and write separate procedures for each interrupt. There is only one important difference: the place and time of interrupt calling are unpredictable.

2.3. INDUSTRIAL MICROPROCESSOR CONTROLLERS

Microprocessor controllers are utilized very often in industry. They are convenient and can substitute a lot of analogue control systems. They work more reliably, the construction is simpler and, finally, they are cheaper than analogue systems. But they have disadvantages too. To create even the simplest microprocessor controller, we have to know very good its hardware and software. The software is horrifying for some people. Assembler, often called “low level language”, is the difficult programming language. As a matter of fact we can use “high level languages” too, for example C language. Using of this language can simplify only arithmetical algorithms, because for example ”sine” function doesn’t exist in assembler. But the hardware programming is more difficult like in assembler, because in C language the bit operations, which are essential for input/output programming, are not implemented. And now a few words about hardware. The supply voltage is not typical, for example 3.3 V, and we have to design (or buy) special AC/DC adaptor. All digital outputs have very low level of current rating (about 1 mA), so it is impossible to operate external devices without amplifier. Maximum level of digital input voltage is also limited, most frequently to TTL signals (5V). The hardware of microprocessor controller is very fragile and a special case resistant to atmospheric conditions is needed.

So the microprocessor controllers should be design only by specialists. And what about the other engineers?

3. PROGRAMMABLE LOGIC CONTROLLERS

The designer’s answer could be only one: a special microprocessor controller adapted to industrial condition. This controller referred to as PLC (Programmable Logic Controller) is the standard microprocessor controller with special case resistant to industrial applications, with standard power supply 230 V AC or safe 24 V DC and with high level current rating outputs. There are two kinds of output: slow speed relay output with maximum current about 2…5 A or high speed open collector transistor output with maximum current about 1 A. The voltage range of digital input is also higher: 120 V AC or 24 V DC.

3.1. PLC SOFTWARE

There are more changes in software. Management of software is very simple. First of all, a user doesn’t need to know controller’s hardware to write a program. The main problems like memory addressing or interrupts in principle don’t exist. Memory is divided into several areas. Each area has the same way of addressing, for example:

- register addressing MB23, where first letter means memory type (M – general purpose memory), the second one means size of the memory register (B – byte, W – word, D – double word). The number at the end means decimal number of memory cell.
- bit addressing Q2.6, where first letter means memory type (Q- output image register), first digit means decimal number of memory cell, the second one – number of bit in byte register.

This addressing system allows mixing different sizes of memory registers at any time (Fig. 2). The essential convenience results from available three very simple programming methods:

STL (Statement List) – very high level text language basis of mnemonic codes,

FBD (Function Block Diagram) – very high level graphical language basis of symbols like logical gates, flip-flops, etc.

LAD (Ladder Logic) - very high level graphical language based on symbols used in electrical engineering.

The third one allows direct and very easy transformation of electric circuit diagrams into PLC program implementation. The user can choose any programming language and, at any time, change it with automatic translation to new language. This is very important particularly when the program is written by many people working in interdisciplinary areas.

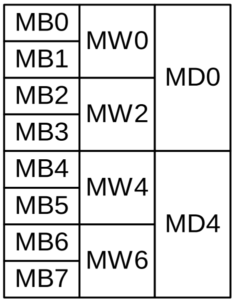

Fig. 2. PLC memory organisation

PLC executes programme in a continuous, cyclical series of tasks called a scan. The scan cycle for the CPU consists of the following tasks (Fig. 3):

- reading the inputs,
- executing the program,
- processing any communication requests,
- executing the CPU self-diagnostic test,
- writing to the outputs.

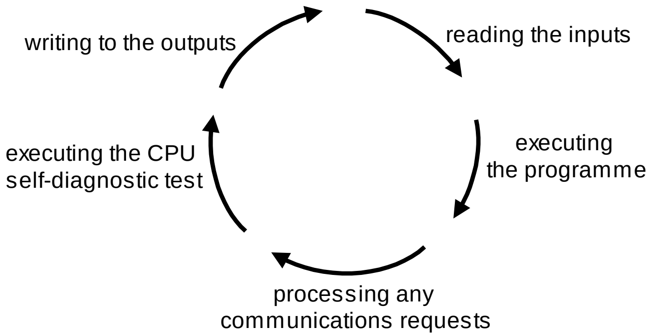

Fig. 3. Scan cycle of PLC

Self-diagnostic test, not implemented in classical microprocessor controllers, is very important. There are special immediate input/output instructions which allow direct access to the specified input or output signal excluding some scans like communication and diagnostic ones.

Power supply interruption handling is one of the biggest disadvantages of controller's operation. PLC offers three ways to avoid it. The first one is the application of the super capacitor with capacity about 1F which enables to store data in memory for about 50 hours. The second one is a dedicated lithium battery which enables to store data in memory for about 2 years. More important data can be stored in flash memory with unlimited time of data keeping. To make testing of production process without breaks possible some fragments of program can be exchanged without stopping the PLC. To test, for example, inactivated alarm system, some input or output can be frozen on specific value. PLC has possibility to communicate with external devices by a lot of standard communication protocols, for example: RS-485, Ethernet etc. The modular construction allows extending of PLC hardware very easily.

4. CONCLUSIONS

Classical microprocessor controllers are very convenient in automation process. However, very complex hardware and software discouraged potential users, in particular unskilled engineers, from applying them in industry. Very experienced specialists were needed for development and implementation of microprocessor controller programs.

Programmable Logic Controllers standardize almost all microprocessor controllers. Now even inexperienced engineer can use them.

REFERENCES

[1] KASPRZYK J., *Programowanie sterowników przemysłowych*, WNT, Warszawa, 2005.
[2] *ARM family microprocessors. User manual*, PHILIPS Publishing 2004.
[3] *Simatic S7-1200 microcontrollers. User manual,* SIEMENS Publishing 2010.

STEROWNIKI MIKROPROCESOROWE KONTRA PLC

W artykule opisano klasyczne sterowniki mikroprocesorowe oraz programowalne sterowniki logiczne PLC pod kątem ich użycia w aplikacjach przemysłowych. Zostały zaprezentowane podstawowe różnice zarówno w budowie sprzętowej jak i oprogramowaniu.