CONTROLLING AN AUTOMATED STORAGE BY USING THE BLUE EARTH 485 MICROCOMPUTER

T. BARABAS

Control Systems Engineering Department, University of Oradea, Str. Universităţii nr. 1, Oradea, Romania
E-mail: tbarabas@uoradea.ro

Abstract. *In this paper it is presented the control system of an automated storage which was conceived and achieved at the University of Oradea by using a Blue Earth 485 microcomputer. There are presented the main components of the control system: the microcomputer and the hardware interface for the drive, the control programs conceived as well as the implemented user interface on a PC computer.*

Keywords: storage, interfacing, control, PC, communication, user interface

1. INTRODUCTION

At the University of Oradea, was built a flexible manufacturing cell by the physical and informational integration of some adequate manufacturing devices (industrial robots, machine-tools, automated storage), Figure 1. This cell contain 2 industrial robots (RV-M1, AID), a CNC machining center: CP20UO and a milling and boring machine: GPR45NC, Figure 2 [1].

Figure 1. The flexible manufacturing cell

For the achievement of the storage function of the manufacturing parts, a storage buffer (with 8 working place) was introduced, Figure 3.

The control architecture of the manufacturing cell was established so to permit the implementation of some open control structures. Thus the serial connection of all control equipments (CNC, ROC1, ROC2) with one by one computer PC, then the interconnection of these PC's in a local area network (LAN), was foreseen [2-3].

To build a manufacturing cell in which the storage function of the manufacturing parts would be solved, the rotary table of the machine center was changed in a storage buffer and was placed between the two machines.
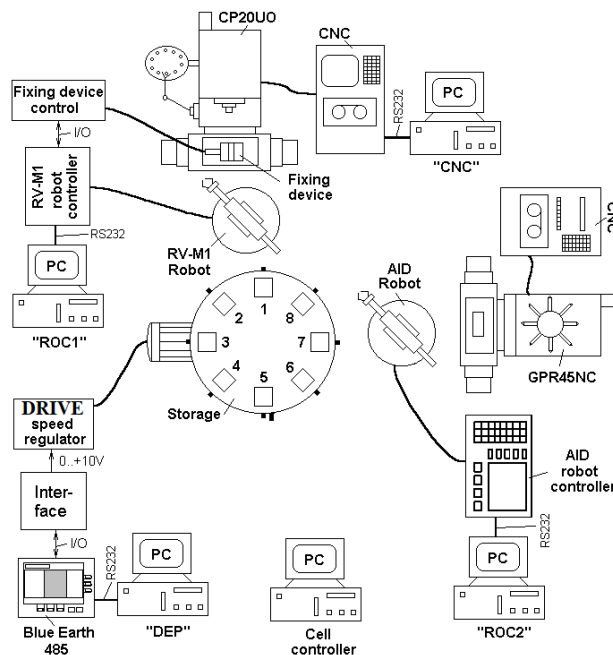

Figure 2. Components of the flexible manufacturing cell

To get a manufacturing cell with a high flexibility, it was achieved that the storage should use an independent control system so that it shouldn't be under the CNC equipment control.

Figure 3. The storage

Practically this storage buffer has at its basis a MR360NC rotary table which was initially used for B axis of the CP20UO machining center and thus it was

under the control of the CNC numerical control equipment of this.

In this paper are presented the results obtained at the achievement of an independent control system for the storage buffer by using a Blue Earth 485 microcomputer.

2. DESIGN AND ACHIEVEMENT OF THE CONTROL SYSTEM

The new control system must assure an appropriate control of the electric drive (DC motor), through an electronic speed regulator, to put in movement the storage disk as well as the programmed stopping of it (with a sufficient precision) in the 8 afferent positions, Figure 3.

To make easier the getting for the storage a control system as simple as possible, the measurement system was given up and the storage was endowed with two proximity sensors (noted with S1 and S2 in Figure 4.).

The conceived control system has at its basis a microcomputer Blue Earth 485 respectively a hardware interface through which the connection to it of the electronic speed regulator and of the proximity sensors is achieved. It has to be mentioned that the I/O interface of the Blue Earth 485 microcomputer is not have analogue outputs (only have analogue inputs and digital outputs) [4-5].

As a result for the control of the speed regulator, which needs an analogue signal in 0…+10V range, it had to be done a hardware interface which had the role of a Digital-Analog Converter (DAC). At the 0V value of the signal, the motor was stopped and at the +10V value the maximum speed was obtained.

By changing the level of the signal, in the respective range, can get a smaller or a bigger speed at the motor and at the storage disk.
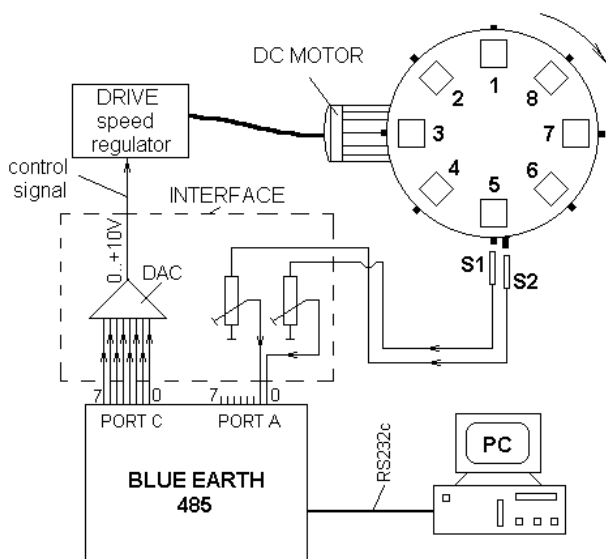
It was considered that it was sufficient the achievement of the one way rotation of the storage disk. As a result it wasn't necessary the use of control signals in –10V...0V negative range proper to the opposite rotation way.

The adoption of this variant of control with only one way of rotation has the advantage of getting an easier interface.

The achieved interface had also the role of supplying with 12Vdc the proximity sensors, respectively the taking over and the adjustment of the supplied signals, so that their level should be compatible with those supported by the input ports of the microcomputer.

Blue Earth 485 microcomputer have a RS232c serial interface, and it could be connected to a PC computer. Thus it could be solved not only the local control of the storage but also the informational connection, through an "interface computer", to a cell controller.

Blue Earth 485, used for the control of the storage, is a microcomputer of general use which has at its basis the 80C51 microcontroller chip. One can find several copies at the University of Oradea, for the control of the various didactical stands with electrical motors as well as pneumatics. It is also endowed with 2 digital ports, each of 8 lines, respectively with a port of 4 analogue input lines (Figure 5).
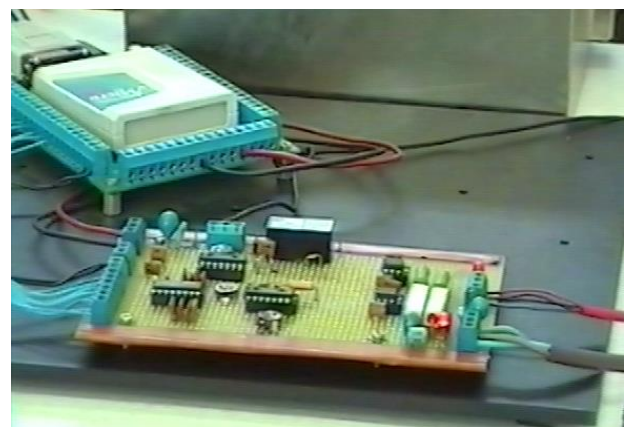

Figure 5. The Blue Earth 485 Microcomputer and the drive interface

It allows the loading and running of the application programs written in assembly language or in BASIC. For the loading of the programs, from a PC computer (to which is connected through a RS232c serial port) it is necessary a special communication software offered separately by the manufacturing company.

This software, due to the relatively high price, wasn't purchased by the University of Oradea. But in conformity with the documentation, we can use any other program of the "terminal" kind, for the programs editing in "command line" mode.

Figure 4. The scheme of the control system

The use of the "terminal" programs offer several disadvantages. For instance it is accessible only the incorporated BASIC interpreter, so consequently the application programs must be written in this BASIC language, and the running speed is relatively low (as compared to those written in the assembly language). Another disadvantage is the fact that the terminal programs don't allow the accomplishment of an automated data change, so without the presence of human operator (they were conceived so that they could be interactive). For instance an operator must be introduced from the keyboard the commands proper to the loading sentence by sentence of the application program and start program with the "RUN" command. At stopping (at the end of execution), for the restarting of the program the presence of the operator is needed also.

Considering that the control system of the storage must offer besides the operator and server state (characterized by an automatic change of information and data) the decision of a proper communication program is made. The protocol which has to be implemented is relatively easy: the PC computer sends a command or a sentence and the microcomputer responds through an acceptance message or in case of syntax error: with an error message. Thus the loading of an application programs are made through loading sentence by sentence of it. For the starting program is used the "RUN" command and for the delete of the memory the "NEW" command (Figure 6 and Figure 7).
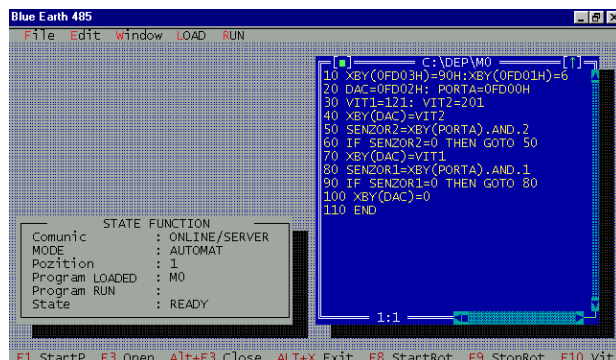

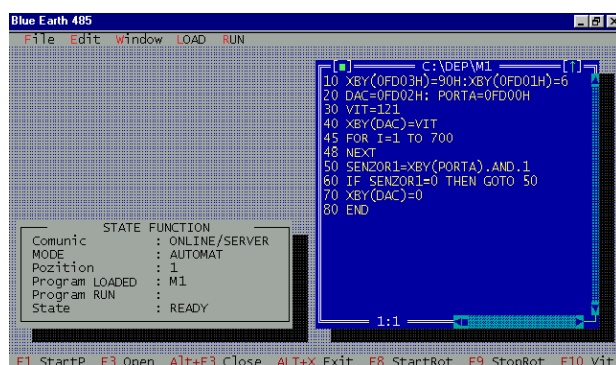**Figure 6. User interface of the control program**

Figure 7. User interface after running the M1 program

The achieved control program allows the loading and downloading of the application programs in/from the

microcomputer memory respectively the starting of these and deleting from memory when is needed. At a one moment only one program can be loaded in the microcomputer's memory, consequently it is not necessary the selection of the program before running. The user interface of the control program has incorporated a text editor for the creation and modification of the application programs.

Also it uses a window dedicated to the visualization of the operation states. In this window is also presented the position of the storage disk.

In operator mode the rotation of the storage disk can be controlled by the F8 and F9 functional keys and with F10 can be prescribed the level of the rotation speed through a value of 0 and 255 (the implicit value is 100). To control the rotation of the storage disk's in automatic mode, two programs can be used: M0 and M1 written in BASIC language.

The M0 program:

```
10 XBY(0FD03H)=90H:XBY(0FD01H)=6
20 DAC=0FD02H: PORTA=0FD00H
30 VIT1=121: VIT2=201
40 XBY(DAC)=VIT2
50 SENZOR2=XBY(PORTA).AND.2
60 IF SENZOR2=0 THEN GOTO 50
70 XBY(DAC)=VIT1
80 SENZOR1=XBY(PORTA).AND.1
90 IF SENZOR1=0 THEN GOTO 80
100 XBY(DAC)=0
110 END
```

The M1 program:

```
10 XBY(0FD03H)=90H:XBY(0FD01H)=6
20 DAC=0FD02H: PORTA=0FD00H
30 VIT=121
40 XBY(DAC)=VIT
45 FOR I=1 TO 700
48 NEXT
50 SENZOR1=XBY(PORTA).AND.1
60 IF SENZOR1=0 THEN GOTO 50
70 XBY(DAC)=0
80 END
```

The M0 program is used to place the storage disk in the origin position, when the real disk's position is unknown (at start up). The other program commands a rotation movement with "a step" by the current position.

Within the 2 programs, the starting of the rotation movement is achieved through the programming of certain control tension (in 0 … +10V range) at the input of the speed regulator, which moves the motor. Considering that the 485 Blue Earth microcomputer is not have analogue output ports, the C digital port is used for the release of a digital value of 8 bits (corresponding to those 8 out lines). This digital value is applied to the

DAC (Digital Analogue Converter) from the achieved interface, which changes it into an analogue signal. This signal then is applied to the speed regulator's.

The command through which the value of the rotation speed is prescribed is XBY(DAC)=VIT when the DAC variable contains the C port's address and VIT contains the value of the speed level. If the prescribed level is different from 0 then a rotation movement, is obtained a proportional speed with it. To stop the movement one can program the zero value to which correspond a command tension of 0V at the speed regulator.

In case of M1 program the movement is stopped at the first received impulse from the S1 sensor. This sensor (connected to line 0 of the A port) sends some impulses at each of the 8 positions of the storage disk. Consequently if the movement takes place until the first impulse appears to S1 then a single step is achieved. Within the program the variable SENSOR1 is used to follow the logic level of the signal from line 0 of the A port if the value is 1 then it is commanded the stopping of the motor through XBY(DAC)=0.

Concerning M0 program the movement is stopped also by the impulse given to S1 sensor but only after the receiving of an "origin impulse" from S2 sensor. Practically in this program are two kinds of speed are used: a normal speed (VIT1) and a "fast" speed (VIT2). Having received the origin impulse, the speed is reduced and the position (followed by stopping) is achieved with a normal speed.

3. CONCLUSIONS

In this paper was presented the control system accomplished for the storage buffer.

The control system assure an appropriate control of the storage by using the 485 Blue Earth microcomputer, and a hardware interface with an electronic drive and a PC computer. This system, within the achieved experiments proved to be precise enough for the accomplishment in good conditions of the parts transfer.

4. REFERENCES

[1] Barabas T., Structuri deschise de automatizare a fabricaţiei din cadrul hipersistemelor CIM robotizate, Editura Universităţii Oradea, ISBN 973-613-528-4, 212 pg., 2004;

[2] Mikell P. Groover, Automation, Production Systems and Computer Integrated Manufacturing, 3rd Edition, Pearson Education, Inc., 2008.

[3] Vesselenyi T., Barabas T., CNC and robot programming, Course book of Series of Advanced Mechatronics Systems, Publisher: Dr. habil Edit Szucs phD, dean of Faculty of Engineering, University of Debrecen, Debrecen, Hungary, HU ISBN 978-963-473-522-9, 305 pg., 2012;

[4] Blue Earth Research Inc.: Blue Earth MICRO-485 – USER'S MANUAL, Blue Earth Research, Mankuto, Minnesota USA, 1993

[5] Embedded Technology.com, A VertMarkets Marketplace for Industry Professionals. Product/Service: Micro-485 Programmable Controller. See also: https://www.embeddedtechnology.com/doc/micro-485-programmable-controller-0001.