Development of a Flexible Manufacturing Cell

ANTÓNIO FERROLHO
Electrical Engineering Department
Superior School of Technology of the Polytechnic Institute of Viseu
Campus Politécnico de Repeses, 3504-510 Viseu
PORTUGAL

MANUEL CRISÓSTOMO
Institute of Systems and Robotics
University of Coimbra
Polo II, 3030-290 Coimbra
PORTUGAL

Abstract: - The actual need for rising the productivity and flexibility of the manufacturing systems requires the use of new technologies and methodologies. Particularly, Flexible Manufacturing Cells (FMC) are the field where better results can be obtained. FMC are highly adaptable to changes in the market and in the product, due to their flexibility. This flexibility only is possible if we use programmable equipment and control it in a real-time fashion. The programmability must be understood in terms of processing equipment, materials handling equipment, use of tools (with the possibility of automatic tool changing, ATC), and the storing equipment. In a competitive market, where the product lifetime is smaller and smaller, the use of FMC is of a crucial importance.

This paper describes the implementation and operation of a FMC in the Institute of Systems and Robotics of the University of Coimbra, Portugal. The objectives of this project were to show the potentialities of such flexible system and a mean to introduce the production systems to the university students. The FMC is composed by a Computer Numerical Control (CNC) milling machine, a CNC lathe, two robots manipulators, two sliding bars for the robots, a conveyer belt, two raw material stores, a buffer, an alarm system, a packing system, and a supervisory computer. A package of software, “WinATS”, was developed in order to overcome the limitations of the robot’s manufacturer software.

The implemented cell has the ability to process different parts in a random way as well as processing new parts. The possibility of processing new parts is assured by the use of a CAD/CAM system. Tests of typical tasks, such as machining, material handling, storing and automatic packing, were performed on the FMC. All the operations on the FMC work in an integrated fashion. The system architecture allows the necessary information flux for command and supervision of all developed activities. Task coordination, synchronization and integration is performed by the developed software.

Key-Words: - Flexible Manufacturing Cells (FMC), Computer Aided Design and Manufacturing (CAD/CAM), Computer Numerical Control (CNC), Robot, Machine-tools.

1 Introduction

Dedicated automation is being replaced by flexible automation, allowing efficient mass production as well as job shop production, even lot sizes of one unit.

A flexible manufacturing cell (FMC) was designed and implemented in the Institute of Systems and Robotics of the University of Coimbra, Portugal. The FMC has the ability of automatically process different types of parts, in any sequence, and can be adapted to a new part demand.

The developed work includes the automatic loading of the CNC programs onto the CNC machine-tools, and the design and implementation of a automatic buffer near of the machine-tools. A CAD/CAM system is used to efficiently design a part and generate the respective CNC code for the CNC machine-tools, allowing the production of high quality parts.

A Flexible Manufacturing Cell can be defined as an “integrated configuration, computer controlled, of CNC machine-tools, auxiliary equipment for production, and a materials handling system, conceived to produce small and medium lot sizes of a vast range of high quality products, at a low cost” [11].

The FMC concept was introduced by the Production Engineering Laboratories of the University of Trondheim, Norway [10].

FMC are production systems adapted to the actual market system, due to the ability to quick and efficient reconfiguration and reprogramming, in order to produce different types of parts and different quantities. The flexibility is assured by the versatility of the processing and handling equipment, and in the real-time control of the entire system. Following are the main advantages of the FMC [9] and [10]:

- FMC produce “for the order”, not “for the stock”;
- They can produce lot sizes of only one unit;
- They allow 100% of inspection, rising the product quality to a pre-defined level;
- They can produce according the costumer specifications;
- The use of FMC rises the productivity;
- FMC produces with better quality;
- They are more reliable, due to the use of intelligent sensors;
- They are more secure for the operators;
- Etc.

The use of FMC is as more important as bigger is the parts variety or as more frequent are the changes in the design. Some equipment which can be included in a FMC are [9] and [10]:

- CNC Machine-tools;
- Robots;
- Automated Guided Vehicles (AGV);
- Rail Guided Vehicles (RGV);
- Conveyer belts;
- Automatic stores;
- Quality Control Systems;
- Etc.

2 Scorbot ER VII robot used in FMC

The Scorbot ER VII robot, shown in figure 1, is a medium dimension robot manipulator, specially designed for execution of manipulation tasks. It is equipped with an electrical gripper, but, by using another type of gripper, it is able to execute tasks such as painting, grinding and welding. This robot has 5 degrees of freedom, corresponding to 5 electrical joints [4] and [5].

The Scorbot ER VII robotic system is composed by the manipulator, a teach pendent, and the ATS (Advanced Terminal Software) and ACL (Advanced Control Language) software, as shown in figure 2. Both ACL and ATS run under MS-DOS [1] and [3]. The ATS runs on-line with the robot. The control is done directly from the PC to the controller, though the serial RS232 port. It is possible, for example, open and close the gripper, do the robot’s homing, visualise and execute the programmes resident in the robot’s controller memory, configure the robot, etc. [3].

Fig. 1 – Scorbot ER VII robot.

The ACL works in a off-line mode, allowing the development of programmes and its download to the robot’s controller, though the serial RS232 connection [1] and [2]. The robot programme runs in the robot’s controller, so that a robot programme must be downloaded to the controller, before its execution. Both ATS and ACL have several limitations concerning programming and robot’s control. Furthermore, they only run in MS-DOS environment.

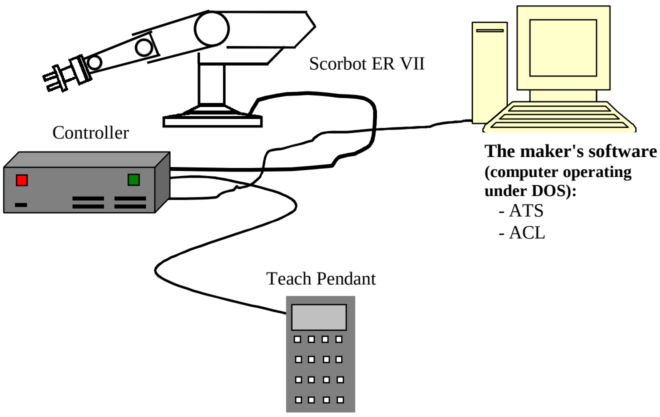


Fig. 2 – The Scorbot ER VII robotic system.

The controller has a Motorola 68020 CPU and 128 Kbytes de RAM, so that the development of more complex robot programmes is difficult.

Aiming to overcome these limitations the software “winATS” was developed. This users friendly software runs in the windows 95, 98, NT, 2000 and XP environments. Furthermore, the users programmes do not need to be in the robot’s controller memory, once the programmes can be run from the PC. It is possible to develop C++ robot’s programmes in the “winATS” and execute them on the PC. Nevertheless, “winATS” also allows the development of robot’s programmes to be run from the robot’s controller. It is also possible to have a hybrid solution, having some programmes running from the PC and others running from the robot's controller, either individually or simultaneously. In this way, the reduced processing power and memory of the controller are not an obstacle to the development of more complex robot's programmes. In this way, the robot's controller can be freed to other tasks.

2.1 The developed WinATS software

A thread process was created in order to poll the serial port. An event is generated each time data is presented in the serial port. In order to send data to the robot's controller, the message is placed in the serial port queue. Asynchronous processing will be used to send a message. For example, in order to open the robot's gripper, a data message is placed in the thread queue and further sent to the robot's controller trough the RS232 channel. After the delivery of the data, a waiting cycle is activated. This cycle is waiting a reply from the robot's controller. It ends when the robot's controller sends back a prompt ('>'), a timeout error occurs or a cancellation message is sent. Figure 3 shows the messages cycle of the thread process.

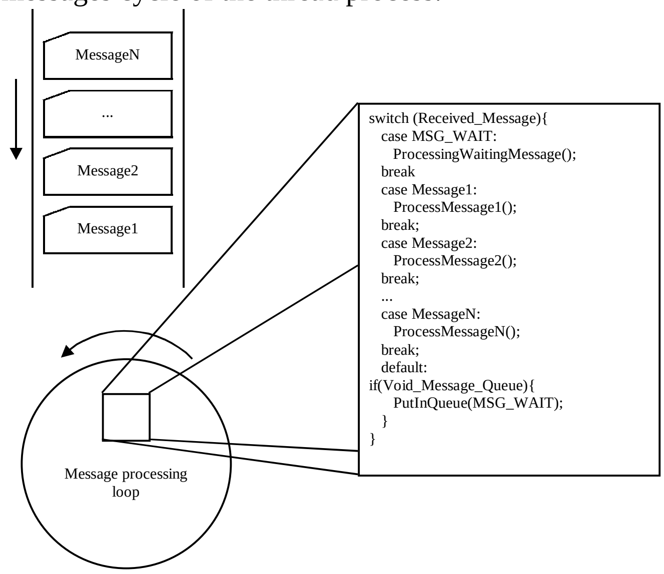


Fig. 3 – Messages Cycle of the parallel process.

The main programme communicates with the parallel process through the messages placed in the waiting queue, being this queue managed by the operating system. A message is immediately processed as soon as it arrives at the parallel process. In the case when there are no messages in the queue, the parallel process enters in a loop, waiting for data from the robot's controller. An event is generated when data arrives.

The parallel process is activated in the beginning of the communication with the robot's controller. When the communication with the robot's controller is finished, the resources allocated to the parallel process are returned to the operating system.

The developed Application Programming Interface (API), for the "winATS" application, is based in a thread process running simultaneously with the main programme [6] and [7].

The ScorbotAPI library was developed having in mind the access to the controller's functions, as shown in figure 4. This API allows us to communicate with the robot's controller.

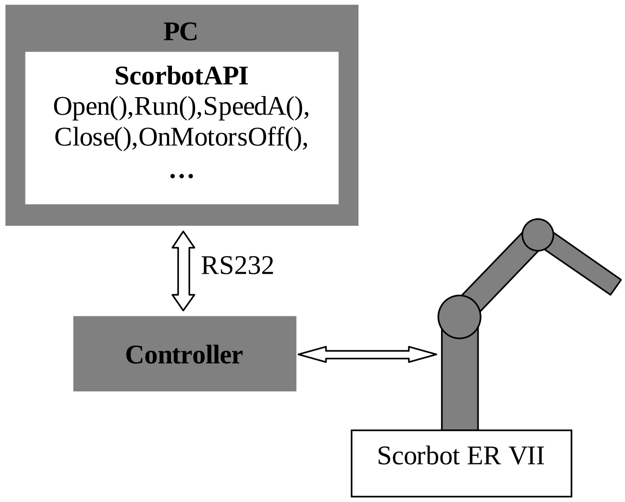


Fig. 4 – ScorbotAPI Library and the robot.

The available functions are divided in two groups. The first contains the public methods and the second contains the methods representing events. Table 1 shows some of these functions.

	Function	Description
Public Methods	SpeedA()	Changes the group A speed
	Home()	Homing the robot
	MotorsOff()	Switches off the robot's motors
	MotorsOn()	Switches on the robot's motors
	Close()	Closes the gripper
	Open()	Opens the gripper
	…	…
Events	OnEndHoming()	This event is activated after homing
	OnClose()	This event is activated after the execution of the Close() function
	OnOpen()	This event is activated after the execution of the Open() function
	OnMotorsOff()	This event is activated after the motors are switched off
	OnMotorsOn()	This event is activated after the motors are switched on
	…	…

Table 1 – Some available functions.

3 The developed FMC

The FMC developed at the Institute of Systems and e Robotics of the University of Coimbra, Portugal, is a totally automatic production cell. It includes:

- A CNC milling machine;
- A CNC lathe;
- A conveyer belt;
- A buffer system;
- Raw material stores;
- An alarm system;
- An automatic packing system;
- A CAD/CAM system;
- Two Scorbot ER VII robots manipulators and their controllers;
- Two linear sliding bars for the robots;
- A supervisor computer (PC).

The layout of the cell is shown in figure 5. Figure 5 also shows the physical connections of the cell devices.

The 5 degrees of freedom Scorbot ER VII robots are placed on linear sliding bars, in order to extend their working range. One Scorbot ER VII robot is used to load and unload the CNC machine-tools and the other is used to serve the stores.

4 The CAD/CAM/CNC Process

The CAD/CAM/CNC process is shown in figure 6. The CAD/CAM system runs in the PC. The design of the parts is done in the CAD system. The CAM system allows to simulate the machining operation of the designed part.

A post-processor is used after the CAD/CAM process, creating the CNC code for the CNC machine-tools. The post-processor was designed by our research team.

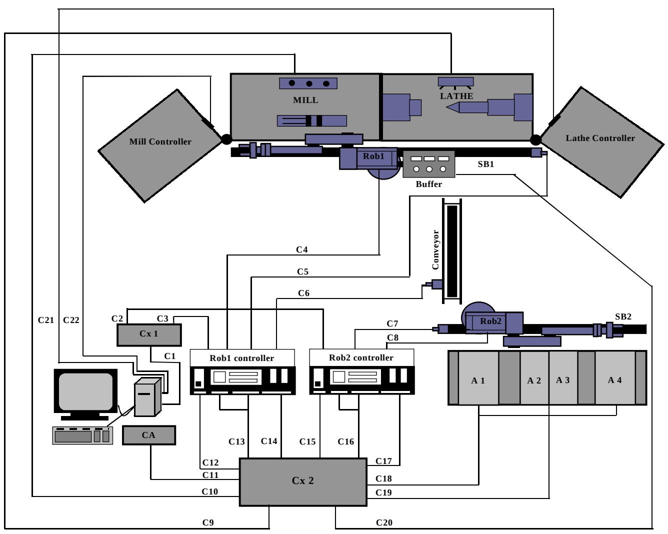


Fig. 5 - Layout and physical connections of the FMC.

A RS232 serial interface is used to download the file resulting of the post-processor, to the CNC controller.

The CNC process, shown in figure 6, comprises the CNC controller and the respective machine-toll.

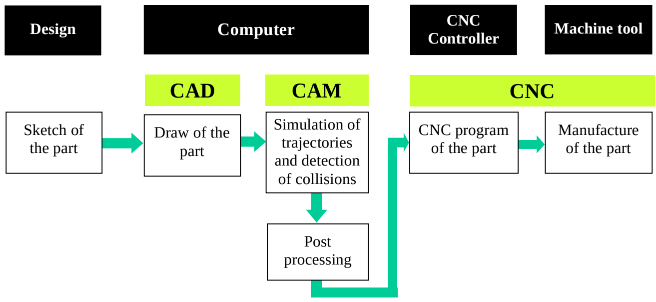


Fig. 6 - CAD/CAM/CNC process.

4.1 CAD/CAM process

The CAD/CAM systems allow the reduction of the design time in the product conception phase, as well as speeding up the data transfer to the production units. In order to design parts in this FMC the software "Mastercam" was used. Mastercam is a CAD/CAM software package running in the Windows environment. Its versatility and easy to use allow to the quick production of high quality parts [8].

4.1.1 Simulation of the Path and Collisions

One of the main objectives of a CAD/CAM system, beyond the geometric modelling, is the CNC programs generation. A CNC program, even if it is generated by a CAD/CAM system, can have some geometric errors. These errors, if they are not detected before, can cause severe economic losses in the manufacturing process, as well as delays in the manufacturing planning. In order to avoid such undesirable situations, it is a good practice to test the programs before they are sent to the CNC machine-tools.

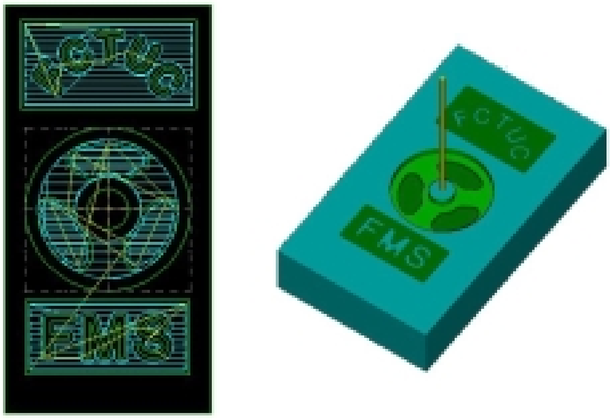
Simulation is good to reduce the number of programming errors by the detection of collisions, bad programmed speed of the tools, etc.

Figure 7 a) shows the simulation of the path of the toll used to mill the part shown in figure 7 b).

4.1.2 Post-processing

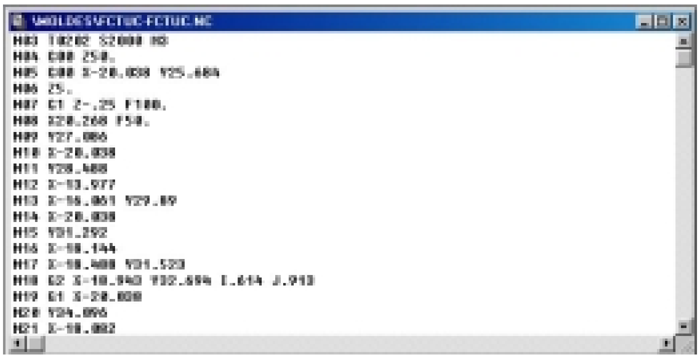
In the final phase of the CAD/CAM process, a post-processor is used in order to convert the generated data.

After the post-processing operation, a window appears showing the CNC data for a possible change. Figure 8 shows an example of a NC file resulting from a post-processing operation.



a) b)

Fig. 7 – Production of a part by a CNC milling machine.
a) Simulation of the toll path.
b) Machined part.



```
N03 T0202 S2000 M3
N04 G00 Z50.
N05 G00 X-20.038 Y25.686
N06 Z5.
N07 G1 Z-.25 F100.
N08 X20.268 F50.
N09 Y27.086
N10 X-20.038
N11 Y28.488
N12 X-13.977
N13 X-16.061 Y29.89
N14 X-20.038
N15 Y31.292
N16 X-10.144
N17 X-10.408 Y31.523
N18 G2 X-10.940 Y32.694 I.614 J.913
N19 G1 X-20.038
N20 Y34.096
N21 X-10.082
```

Fig. 8 - NC file resulting from a post-processing operation.

4.2 Computer Numerical Control - CNC

Computer Numerical Controller (CNC) is an electronic equipment that reads and interprets coded data (programs) and sends electric signals to the machine-toll's motors, according with the program, in order to manufacture a part. This system is composed by a microprocessor and memory.

The CNC controls the axes movement, the cutting speed of the tool, the tool changing, the working conditions of the machine (cooling, coordinates, position, etc.), as well as its state (warnings and alarms).

5 Communication among equipments

Communication among the several equipments assures the coordination, synchronization and integration of all activities of the FMC. The robot's 1 controller has the necessary real-time programs for cell operations, including the communication with machine-tools. It also communicates with robot's 2 controller, in order to activate a given process according to the actual cell activity.

6 The production process

Figure 9 shows the resident programmes in each one of the robot's controllers, needed for cell exploration.

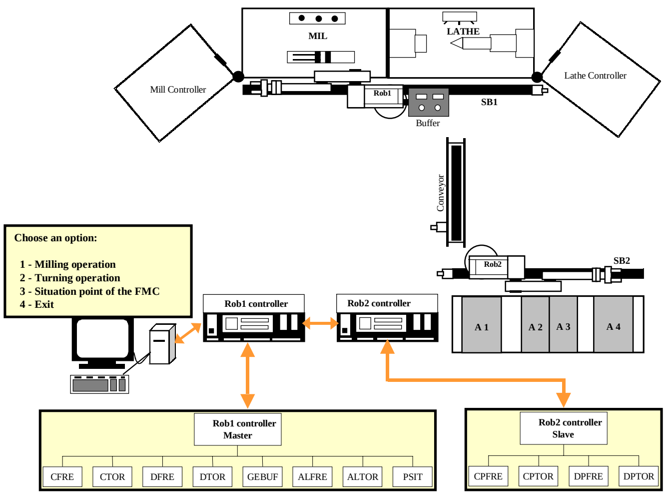


Fig. 9 - Resident controller programmes.

In order to initiate the production, the operator must choose, in the computer, the number and the type of parts to be machined. From now on, the process is totally automated, the operator has no further intervention. Total unmanned operation means the machine-tolls loading with raw material, machination and packing and storage of the parts.

The various tasks performed on the cell are executed by the resident programs in the robot's controllers. For example, the program CFRE is responsible to pick up a part from the buffer and load the milling machine to machine it. CTOR is a program that do the same but for the lathe. DFRE is used to unload the milling machine and DTOR is used to unload the lathe. Each program in each robot's controller was designed to perform a task in the automatic production process.

7 Master and Slave processes

The master process is the first to be executed in the controller of the robot 1 (Rob1). The respective flowchart is shown in figure 10. Its function is to start all processes necessary to operate the FMC. In this process the initialisation of all global variables used also by other processes resident in the controller Rob 1 is done. The synchronization with controller of the robot 2 (Rob2) is also done by the master process. At the starting, this process allows to verify if the buffer is full or not, allows to homing the robot 1 and the linear base 1, and allows to prepare the machine-tools to start the production. It also puts the processes GEBUF, ALFRE and ALTOR in auto-run.

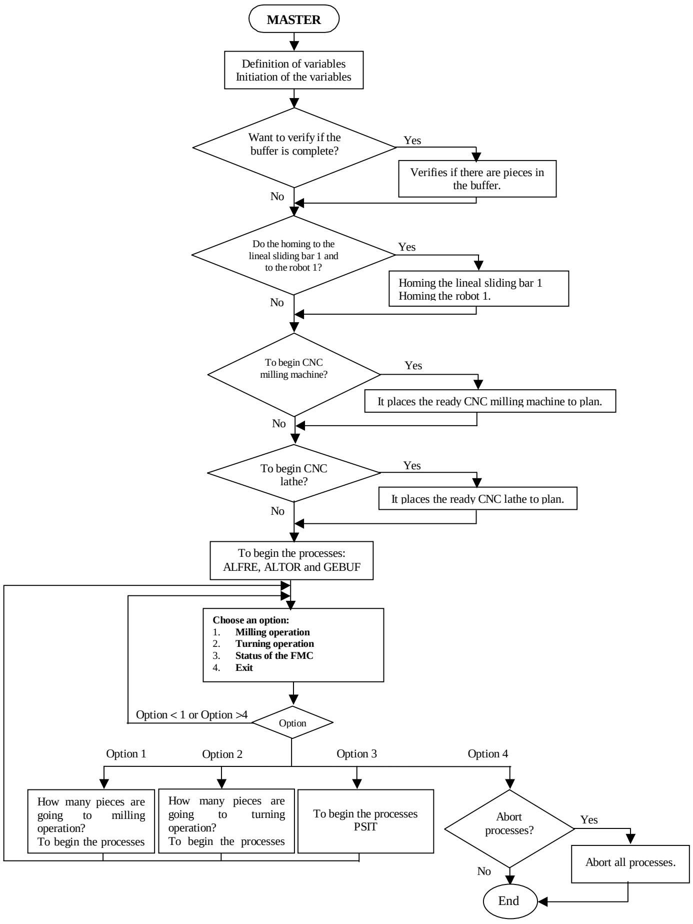


Fig. 10 - MASTER process flowchart.

The objective of the slave process is to home the linear base 2 and the robot 2, and to initialise the CPFRE, CPTOR, DPFRE and DPTOR processes, resident in the controller of the robot 2. The respective flowchart is shown in figure 11.

8 Conclusions

A Flexible Manufacturing Cell (FMC) was implemented and tested for typical production activities, particularly loading and unloading operations, machining, transportation, packing and storage.

All operations are integrated in the FMC. The cell architecture allows the flux of data necessary for command and supervision of all cell activities.

A package of software, “WinATS”, was developed in order to overcome the limitations of the robot’s manufacturer software.

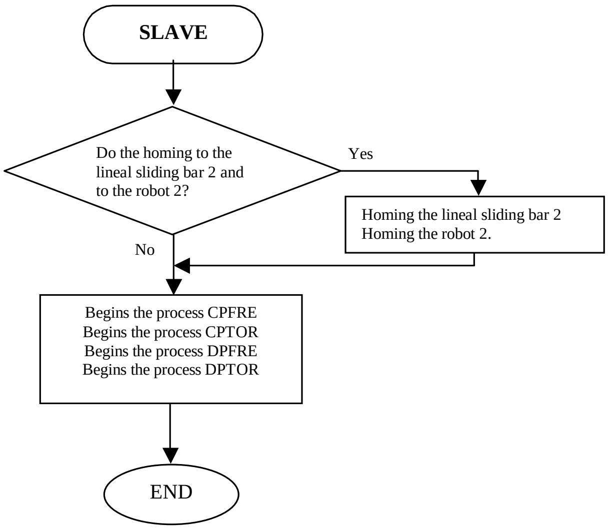


Fig. 11 - SLAVE process flowchart.

The use of an alarm system allows the detection of equipment malfunctions as well as the lack of raw material and empty buffer.

The buffer, located near the robot 1, allows better times of machine-tools load and unload operations.

References:

[1] “ACL–Advanced Control Language – Reference Guide”, Eshed Robotec, 4th Edition, 1995.

[2] “ACLoff-line – User's Manual”, Eshed Robotec, 2nd Edition, 1995.

[3] “ATS–Advanced Terminal Software - Reference Guide”, Eshed Robotec, 1st Edition, 1995.

[4] “Scorbot ER VII, User's Manual”, Eshed Robotec, 2nd Edition, 1996.

[5] Moshe Shoham, Textbook 1, *Fundamentals of Robotics*, Eshed Robotec, 1984.

[6] David J. Kruglinski, *Visual C++*, Microsoft Press, 1997.

[7] William P. Alves, *C++ Builder 6*, Erica, 2002.

[8] “Mastercam version 7.0, Reference Manual”, 1998.

[9] M. P. Groover, *Automation, Production Systems, and Computer Integrated Manufacturing*, Prentice-Hall, Inc., 1987.

[10] Y. Koren, *Computer Control of Manufacturing Systems*, Mc Graw Hill, 1983.

[11] N. Viswanadham and Y. Narahari, *Performance Modeling of Automated Manufacturing Systems*, Prentice-Hall, INC., 1992.