

WEB BASED VIRTUAL ROBOT PROTOTYPING AND MANUFACTURING

Raul Mihali, Tarek Sobh, ¹⁾ Vasyl Koval, Tymur Korkishko,
Anatoly Sachenko, Volodymyr Kochan ²⁾

1) University of Bridgeport, School of Engineering and Design, 169 University Avenue, Bridgeport, CT 06601, U.S.A. Phone: (203) 576 - 4116, Fax: (203) 576 - 4766 sobh@bridgeport.edu, <http://www.bridgeport.edu/~sobh>

2) Ternopil Academy of National Economy, Institute of Computer Information Technologies, 3 Peremoga Square, 46000, Ternopil, Ukraine Phone: +380 (352) 33-0810 Fax: +380 (352) 33-0024 (24 hrs) as@tanet.edu.te.ua, vko@tanet.edu.te.ua, <http://www.tanet.edu.te.ua>

Abstract - *Developing an environment that enables optimal and flexible design of robot manipulators using universal reconfigurable links, joints, controllers, actuators and sensors is an essential step for efficient robot design and prototyping. Robot control has to be made by a universal controller with remote hardware and software reconfigurations. This technique allows to create an optimal configuration of software and hardware components for designing of the specialized robot, its algorithmic control and rapid prototyping.*

I. INTRODUCTION

Now the large attention is given to a problem of robots creation and implementation. One of the important directions is the creation of cheapest and effective manipulators, which find more implementation in industry. There are a lot of various manipulator's types depending on functions which they have to perform. The scientific interest is represented by the tasks on designing and creation of reliable remote control tools by such manipulators. For efficiency of the tasks solution performed by manipulators it is necessary to create a framework for it designing.

The idea of web-based control has been always envisioned from the first days of networked computing. Being able to execute operations from remote locations [1], with only feedback of some sort, it is an active and desired choice in many fields, such as robotic manipulators. This paper presents a web based control solution for a manipulator.

A complex Application Programming Interface (API) for the control of the SIR-1 robot (fig. 1) has been developed [2]. Available functions include direct/inverse kinematics computation, serial port communication interfacing, and link speed control. The API can support an indefinite number of port connections and thus control a theoretically indefinite number of SIR-1 robots. Due to the high availability of serial ports on standard PCs, this API can be deployed virtually anywhere and in any environment, including the Internet, for almost any application.

The designed robot features base rotation, shoulder, and elbow and wrist motion with a functional

gripper along with two additional servos to provide different steering. The electronics and the servo controller are completely assembled.

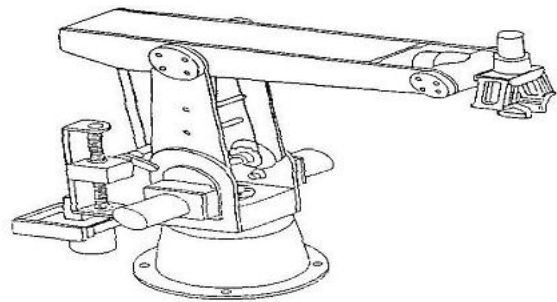


Fig. 1. *The SIR-1 robot*

II. THE ROBOT PROTOTYPING ENVIRONMENT

Designing and building an electro-mechanical system such as a robot manipulator requires solving different problems:

- (i) specifying the tasks and performance requirements,
- (ii) determining the robot configuration and parameters that are most suitable for the required tasks,
- (iii) developing the necessary software and hardware components,
- (iv) ordering and manufacturing of the parts and assembling the robot,
- (v) testing the robot and measuring its performance.

Our goal is to build a framework for the optimal and flexible design of robot manipulators with the

required software and hardware systems and modules that are independent of the design parameters, so that it can be used for different configurations and varying parameters. Fig. 2 shows a schematic view of the prototyping environment with its subsystems and interface. The proposed Prototyping and Manufacturing Environment (PME) consist of a central interface (CI) and subsystem interfaces (SSI). The several subsystems carry out certain tasks to build the prototype robot. These subsystems share parameters and information. To maintain the integrity and consistency of the whole system, a central interface (CI) is proposed with required rules and protocols for passing information. This interface will be the layer between the robot prototype and the subsystems, and it will also serve as a communication channel between the different subsystems.

The tasks of the central interface are to:

- maintain a global database of all the information needed for the design process;
- communicate with the subsystems to update any changes in the system. This requires the central interface to know which subsystems need to know these changes and send messages to these subsystems informing them of the required changes;
- receive messages and reports from the subsystems when any changes are required, or when any action has been taken (e.g., update complete);
- transfer data between the subsystems upon request;
- check constraints and apply some of the update rules;
- maintain a design history containing the changes and actions taken during each design process with date and time stamps;
- deliver reports to the customer with the current status and any changes in the system.

The subsystem interfaces are the interface layers between the CI and the subsystems. This makes the design more flexible and enables us to change any of the subsystems without much change in the CI - only the corresponding SSI need to be changed. The role of the SSIs is to:

- report any changes to the CI;
- receive messages from the CI with required updates;
- perform the necessary updates in the actual files of the subsystem;
- send acknowledgments or error messages to the CI.

The task will require various types of integration, including the development of a web interface, the design of software simulators for hardware

controllers, parts and automated (robotized) manufacturing facilities. To make the proposed model truly accessible and marketable, the set of required tasks, torques, dexterity, repeatability etc will be integrated into a web based portal, hence easily access through a URL.

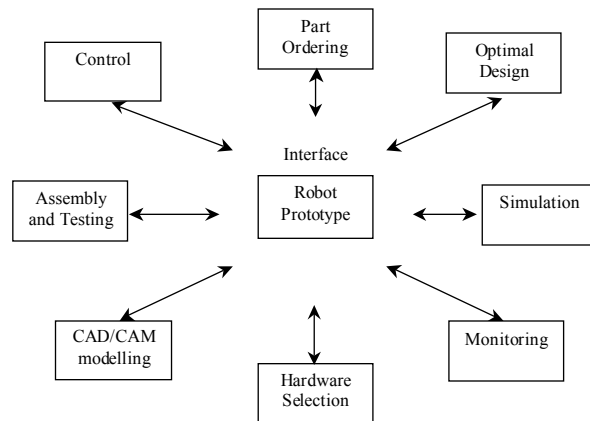


Fig. 2. The prototyping environment

The execution environment for heterogeneous systems is called “InterBase” [3]. It integrates pre-existing systems over a distributed, autonomous, and heterogeneous environment via a tool-based interface. In this environment each system is associated with a Remote System Interface (RSI) that enables the transition from the local features of each system to a uniform system-level interface. It uses object orientation and its applications to integrate heterogeneous, autonomous, and distributed systems to achieve such unification. Object-oriented distributed computing is a natural step forward from the client-server systems of today. Automated, flexible and intelligent manufacturing based on object-oriented design and analysis techniques and a system for design, process planning and inspection will be presented in this project. A prototyping environment for robot manipulators and a 3-link manipulator prototype have been developed in Utah (Utah Prototyping Environment UPE and Utah Robot Kit URK) [4]. Various controlling, simulation and optimization methods have been successfully presented and exercised on custom manipulators such as a tire changing manipulator, a generic 6 degrees of freedom manipulator, etc.

Rapid prototyping of the developed robots demands universal and flexible elements. Particular, the system of the robot control subsystem have to be basing on the controller, which must provide:

- sensors and actuators interacting with different interfaces;
- the possibility of easy changes in the on-line interfaces for the external devices;
- sufficient computational capability for sensor

signal processing and control under actuators. The main goal of such processing is to improve the accuracy of uncertain sensor information received from noisy inaccurate sensors using artificial intelligence methods, such as neural networks, sensor fusion and data mining;

- the possibility of on-line changes of the algorithms.

III. THE ROBOT CONTROL STRUCTURE

The common structure of robot control can be synthesized on bases of given manipulator functions in prototyping environment. Such structure is presented on fig. 3. The central element of the structure is universal controller. It can control a robot according to the commands, which are entered from the Internet or a phone line. Using of the phone lines have benefit if the robot structure is a constant and the number of performed commands are small. The web-based protocol is used if it is necessary frequently remote on-line robot reprogramming. For this one it is necessary to include a modem in the robot capacity (or hardware, or equipment).

The controller decodes the commands, which enter to its input and generates a corresponding control signals to the servomotors of the manipulators. The sensors provide feedback links. The sensors set depends on the performed task. The necessary servomotor set and sensor set for different manipulator types are different. It is defined by the tasks, which have to be performed. Thus, the controller structure has to be changed for different tasks. The individual controller design according to each task doesn't allow on-line robot reprogramming and is economically inefficient. Therefore the design of the universal controller, which could change remotely a structure of hardware and software, is needed.

The universal controllers expediently to create on the single chip microcontroller bases for the simple and cheapest manipulators. It is necessary to provide the controller reprogramming feature with minimum resources expenses in the case of

absence of the operation systems. The changes of hardware are possible by using field programmable gate array (FPGA) chip [5]. The necessary files for reprogramming have a size between 8 and 130 KB. Therefore a robot reprogramming is expediently to provide only through the Internet. The phone channel has relatively small traffic capacity, therefore a reprogramming time would be unacceptably large.

IV. THE UNIVERSAL CONTROLLER

The universal controller structure is presented on figure 4. The base of structure is single chip controller MC. The universal controller include: the RAM, input-output ports on the FPGA base, ADC, phone interface, reset circuit and address register RgA. Moreover, there are additional logical elements OR, AND, power switches SW and Schmit trigger.

The designed universal controller provide possibility of on-line changing of:

- the software;
- the hardware reconfiguration of the ports for the actuators control (by servomotors, relay etc.);
- the hardware reconfiguration of the inputs ports of the signal from sensors, which have a discrete output.

The application-specific controller's software is located in RAM in the usual conditions. When MC is powered on, the program for RAM loading is activated and is located in the internal MC memory. It refers to the server for the necessary robot-operating program. At the end of loading process, MC switches to the mode for operating with extended memory and runs the program written to the RAM [6]. If the robot works independent from the server, the part of RAM should be replaced with Flash RAM. At this case the robot control through the phone line is expedient [2].

The hardware configuring of the universal controller occurs simultaneously with changing of software. The FPGA configuration file is passed after application-specific program. If a working program is located in Flash RAM, the FPGA configuration file is located there too.

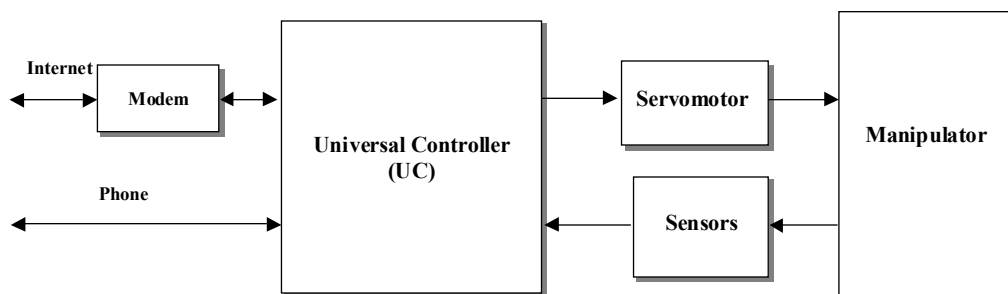


Fig. 3. The Common Structure of the Manipulator's Controller

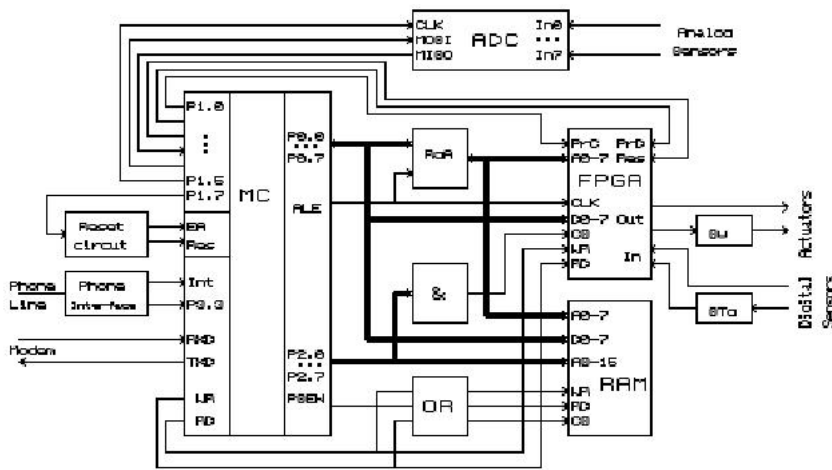


Fig 4 The Universal Controller Structure

The on-line reprogramming realized by the special command, which passes the control signal on the output P1.7. This one initiates the restart of MC and runs the program for RAM loading of the application-specific program. At this situation the FPGA configuration could be updated if it is necessary.

The main task of FPGA is formation of necessary quantity of inputs and outputs of discrete signals. Thus, the pulse-width modulators of the outputs are realized by FPGA tools if it is necessary [7]. In structure of the universal controller the powerful keys SW for control by the servomotors also included. Also in structure of the universal controller Schmitt trigger is included for the precise signals formation at presence of noises and contact bounce. For FPGA control last 255 addresses in MC address space allocated.

If necessary to use a sensors with an analog output signal, in structure of the universal controller is entered 8-channel 10-digit ADC such as MC145051. This ADC cooperates with MC through the SPI interface. This interface can be realized as a program module for MC which are based on the simple versions of microcontrollers or integrated in MC.

The developed controller can be widely used in simple and cheap manipulators taking into account whole set of its possibilities.

V. INTERNET CONTROLLED MANIPULATOR

The Internet-Controlled Robot is a robust and repeatable telerobotic vehicle that can be controlled from any Internet-enabled computer. In supervised telerobotics, an operator at a local site utilizes input devices and graphical visualization tools to command execution of a task at a remote site using a telerobotic vehicle.

The robot is controlled from a web page, where the operator handles the robot by using a graphical user interface. The operator can control each link of the robot separately, or move the robot to a given point in 3-D space by inputting the corresponding coordinates. The operator sees the movement of the robot on the computer screen in real-time via a live video camera broadcast (fig. 5).

Such a telerobotic vehicle can be used for controlling a robot that operates in a hazardous or inaccessible environment. In addition, this vehicle offers the benefit that the operator does not have to be in the vicinity of the robot, but can be located anywhere throughout the globe. Practical applications of the system include remote-controlled operations in radioactive or toxic areas, which are dangerous for human beings, as well as remote reconnaissance and sample collection.

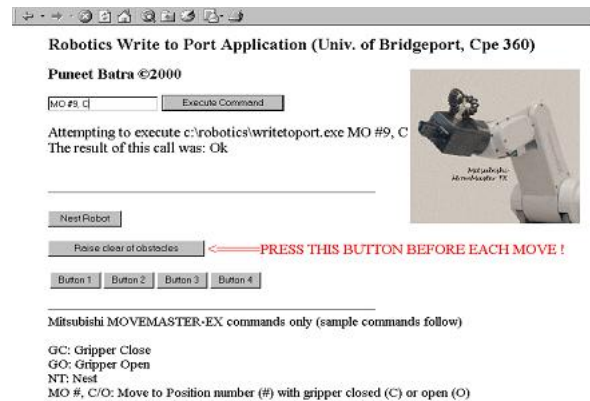


Fig 5 Screenshot of the main interface page

For the Internet control of the robot, we have created a distributed high performance client/server architecture [8]. In this architecture the client provides an easy control interface for the user and transmits control commands to the server over TCP/IP, encoding the data before the transmission. The system architecture is based on the thin client model where the client performs only data validation and data transmission routines. The server receives and decodes control commands from the client. Once the transmitted command has been decoded, the server then performs inverse kinematics calculations on the data. This is necessary since the user inputs values of x, y, z coordinates via the client and they have to be transformed into

angular values, which the robot's micro-controller can understand.

Telerobots have the benefit of human cognitive and perceptual abilities, since they are operated by a human being, they can perform more efficiently in unfamiliar and dynamic environments. Telerobots such as the Internet-Controlled Robot can be thought of as a robotic device that synergistically combines human being and machine, and such systems will undoubtedly have a great contribution towards the progress of humanity in the future.

VI. CONCLUSION

There is proposed a web interface based prototyping environment for robot manipulators with heterogeneous subsystems and interfaces between different components of this environment. It allows potential customers (located anywhere geographically) to input through a web interface a set of requests and design parameters that could be analyzed, simulated and converted to specific manufacturing information that can be ultimately used by an automated manufacturing plant.

The structure of robot's universal controller is proposed. The controller provides possibility of on-line changing of: the software, the hardware reconfiguration of the ports for the actuators control, the hardware reconfiguration of the inputs ports of the signal from sensors, which have a discrete output. This approach allows the wide usage of the controller in simple and cheap manipulators.

VII. ACKNOWLEDGEMENTS

Authors would like gratefully acknowledging INTAS for financial support of this work, granting reference number INTAS-OPEN-97-0606.



Raul C. Mihali received the B.Sc. and M.S. degree in Computer Programming degree with honors in Computer Science and Automatic Control from Department of Computer Science and Engineering, University of Bridgeport, CT, USA, in 1999 and 2000. He is currently completing the Ph.D. degree in Computer Engineering at University of Bridgeport.

His current research interests include CAD/CAM and active sensing under uncertainty, robots and electromechanical systems prototyping, modeling and applications, mobile robotic manipulation, algorithms and programming techniques, stealth and security oriented robots, polymorphic structures and robot actuators and sensors for unusual execution tasks.

VIII. REFERENCE

[1]. H. Latchman, Sanjeev Tothapilly, C. Salzmann and D. Gillet, "Hybrid Asynchronous and Synchronous Learning Networks in Distance and Local Education", in *Proceedings of the 1998 International Conference on Engineering Education*, Rio de Janeiro, August, 1998.

[2]. T.Sobh, R.Mihali, A.Rosca, B.Ghimire, K.Vovk, G.Gosine, P.Batra, A.Singh, S.Pathak, "Case Studies in Web-Controlled Devices and Remote Manipulation", in *Proceedings of the International Workshop on "Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications," IDAACS'2001, Foros, Crimea, Ukraine, July 1-4, 2001*, pp. viii-xiv.

[3]. Bukhres O. A., Chen J., Du W. and Elmagarmid A. K., "Interbase: An execution environment for heterogeneous software systems." *IEEE Computer Magazine* (Aug. 1993), pp. 57-69.

[4]. Dekhil M., Sobh T. M. and Henderson T. C., "URK: Utah Robot Kit - a 3-link robot manipulator prototype." In *IEEE Int. Conf. Robotics and Automation* (May 1994).

[5]. Yalamanchili, Sudhakar, *VHDL Starter's Guide*, Prentice Hall, 1998.

[6]. A.Sachenko, V.Kochan, V.Turchenko, V.Tymchyshyn, N.Vasylkiv, "Intelligent Nodes for Distributed Sensor Network", *Proceedings of the IEEE Instrumentation and Measurement Technology Conference IMTC/99, Venice, Italy, 1999*, pp.1479-1484.

[7]. Skahill, Kevin. *VHDL for Programmable Logic*, Addition-Wesley, 1996.

[8]. D. Gillet, G. F. Franklin, R. Longchamp, and D. Bonvin, "Introduction to Automatic Control via an Integrated Instruction Approach", *The 3rd IFAC Symposium on Advances in Control Education*, Tokyo, Japan, August 1994.



Tarek M. Sobh received the B.Sc. in Engineering degree with honors in Computer Science and Automatic Control from the Faculty of Engineering, Alexandria University, Egypt in 1988, and M.S. and Ph.D. degrees in Computer and Information Science from the School of Engineering,

University of Pennsylvania in 1989 and 1991, respectively. He is currently the Director of the School of Engineering and Design; director of external engineering programs; Professor of Computer Science and Computer Engineering; the founding director of the interdisciplinary Robotics, Intelligent Sensing, and Control (RISC) laboratory at the School of Engineering and Design, University of Bridgeport, Connecticut; and the Chairman of the Discrete Event

Dynamic Systems (DEDS) Technical Committee of the IEEE Robotics and Automation Society.

He was the Interim Chairman of Computer Science and Engineering at the University of Bridgeport in 1998 and was a Research Assistant Professor of Computer Science at the Department of Computer Science, University of Utah from 1992—1995, and a Research Fellow at the General Robotics and Active Sensory Perception (GRASP) Laboratory of the University of Pennsylvania during 1989—1991. Dr. Sobh's current research interests include reverse engineering and industrial inspection, CAD/CAM and active sensing under uncertainty, robots and electromechanical systems prototyping, sensor-based distributed control schemes, unifying tolerances across sensing, design, and manufacturing, hybrid and discrete event control, modeling, and applications, and mobile robotic manipulation.

He has published over 80 journal and conference papers, and book chapters in these and other areas. Dr. Sobh is a Licensed Professional Electrical Engineer (P.E.), a Certified Manufacturing Engineer (CMfgE) by the Society of Manufacturing Engineers, and a Certified Reliability Engineer (C.R.E.) by the American Society for Quality, a member of Tau Beta Pi, Sigma Xi, Phi Beta Delta, and Upsilon Pi Epsilon.

Dr. Sobh was the recipient of the Best Paper Award at the World Automation Congress Conference (WAC 98). Dr. Sobh is a member or senior member of several professional organizations including; ACM, IEEE, IEEE Computer Society, IEEE Robotics and Automation Society, the National Society of Professional Engineers (NSPE), the American Society of Engineering Education (ASEE), and the Society of Manufacturing Engineers (SME).



Vasyl Koval was born in Ternopil in 1975. In 1998 he received B.Sc. in "Management Information Systems" at Ternopil Academy of National Economy, Institute of Computer Information Technologies. In 1999 he obtained M.Sc. in "Economic Cybernetics" at Ternopil Academy of National Economy. Since 1999 he is Ph. D. Student in "Computing Machines, Systems and Networks" at Ternopil Academy of National Economy.

Now he works as lecturer of Department of Information Computing Systems and Control of the Institute of Computer Information Technologies of Ternopil Academy of National Economy, Ukraine; scientist of Research Lab. of Automation Systems and Networks of Ternopil Academy of National Economy.

His research area includes: Robotic Systems; Distributed Systems; Sensor Fusion Techniques; Sensor Fusion Algorithms; Data Acquisition Systems; Stereo Vision; Stereo Matching; Image Processing; Artificial Intelligence; Neural Networks; Robot Navigation Systems; Sensor Systems, etc.

Tymur A. Korkishko (20 Apr. 1975), Ukrainian, graduated in computer science at State University "Lviv Polytechnic", Lviv, Ukraine (1997). He is Ph.D. student at Ternopil academy of national economy. He has published more than 20 research papers on hardware cryptographic processors and systems.



Became lecturer at the Department of Computer Science, Ternopil academy of national economy (2002). He teaches courses: Programming Languages, Essentials of Computer Aided Design of digital systems.

Areas of special interests: high-speed cryptography; cryptoprocessors and cryptoaccelerators ASICs development methodologies; programming languages.

Anatoly Sachenko is Professor and Director of the Institute of Computer Information Technologies, Ternopil Academy of National Economy since 1994. At present he is Visiting Fulbright Professor at University of Maine, Department of Computer Science for the 2002/03 academic year.



He earned his B.Eng. Degree in Electrical Engineering at Lviv Polytechnic Institute in 1968 and his PhD Degree in Electrical Engineering at Lviv Physics and Mechanics Institute in 1978 and his Doctor of Technical Sciences Degree in Electrical and Computer Engineering at Leningrad Electrotechnic Institute in 1988. Since 1991 he has been Honored Inventor of Ukraine, since 1993 he has been IEEE Senior Member.

His main Areas of Research Interest are Implementation of Artificial Neural Network, Distributed System and Network, Parallel Computing, Intelligent Controllers for Automated and Robotics Systems. He has published over 300 papers in areas above.



Volodymyr Kochan, Ph.D. of Measurement Engineering, Associate Professor, was born in 1951 in Lviv. In 1973 he received B. Eng. Electrical Engineering at Lviv Polytechnic Institute, Ukraine. In 1989 he obtained Ph.D. of Electric and Magnetic Instrumentation at Kiev Polytechnic Institute, Ukraine.

Now he works as Associate Professor of Department of Specialized Computing Systems and Senior staff scientist of Research Lab. of Automation Systems and Networks of Institute of Computer Information Technologies of Ternopil Academy of National Economy; Department of Computer Science of Ternopil State Technical University; Department of Electrical devices of Technical college of Ternopil State Technical University; Instructor of the practical course on microprocessor application.

His research area includes: Sensor Intelligent System; Distributed Sensor Network; Computer based Intelligent Measurement and Control Systems; Intelligent Controllers for Automated and Robotic Systems in Industry; Sensor Systems Calibration and Verification.