PERIODICA POLYTECHNICA SER. MECH. ENG. VOL. 43, NO. 2, PP. 179–188 (1999)

NEW PC AND LABVIEW BASED ROBOT CONTROL SYSTEM

Sándor J. TÓTH

Department of Manufacture Engineering
Technical University of Budapest,
H–1521 Budapest, Hungary
Tel.: +36-1-463-3180, Fax: +36-1-463-3176,
Email: sanco@manuf.bme.hu

Received: June 5, 1999

Abstract

This paper presents a new economical solution of robot control system for different sophisticated robot applications. The control system is based on PC controlled servo motor control card and an intelligent control software, which has been developed using high level graphical programming language (LabVIEW). The basic development is an interface software for making connection between the control card and LabVIEW. LabVIEW gives a wide range of opportunity of utilisation of the developed control system at different robot applications. This paper shows a complete solution of robot control system for a ZIM 15 (KUKA license) type 6-axis robot. The DDA (Digital Differential Analysis) method and the closed loop control system of the servo motor control card are described. The programming of the control card and the time optimal trajectory planning method are presented, too.

Keywords: time optimal trajectory planning, robot control system, economical solution, LabVIEW, PCL-832 servo motor control card.

1. Introduction

The advent of new high-speed technology and the growing computer capacity provided realistic opportunity for new robot controls and realisation of new methods of control theory. This technical improvement together with the need for high-performance robots created faster, more accurate and more intelligent robots using new robot control devices, new drives and advanced control algorithms. On the other hand – parallel with the need for high-performance robots – another need has appeared: the need for low cost robot control. This paper describes a new economical solution of robot control systems. The presented robot control system can be used for different sophisticated robot applications. The control system consists of a PC, two servo motor control cards, a digital I/O card, drivers of the servo motors of the robot and an intelligent control software, which has been developed using high level graphical programming language (LabVIEW). A complete solution of a robot control system is presented in this paper. The control system was developed for a ZIM 15 (KUKA license) type 6-axis robot, but any kind of robots can be controlled with the presented control system, which utilises servomotors. The basic development is an interface software for making connection between the control card and

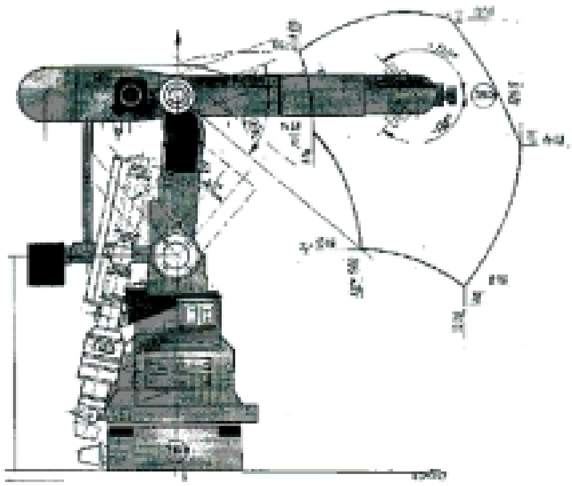

Fig. 1. The ZIM 15 type robot

LabVIEW software. This interface software has been developed using C language and integrated with LabVIEW using the Code Interface Node (CIN) function of LabVIEW. The advantage of this approach is that LabVIEW gives a wide range of opportunities of utilisation of the developed control system at different robot applications. Only few advanced features are mentioned, e.g.: easy use of the TCP-IP communication between the robots and the Flexible Manufacturing Cell (FMC), built-in functions allowing fast numerical calculations and easy implementation of different control methods and trajectory planning algorithms.

2. The Robot and the Control System

The robot control system was developed for ZIM 15 (KUKA license) type 6-axis robot. *Fig. 1* shows the ZIM 15-type robot and *Fig. 2* describes the structure of the robot control system. Focusing on the actuators and sensors of the robot hardware, each axis of the robot has its own servomotor, tacho-generator, encoder and two end-position sensors. It is important to mention that the sixth axis is not linked to the robot itself. The sixth axis is a rotating table located near the robot inside the robot workspace. The hardware of control system consists of a PC (an ordinary PC 486DX4/100), two PCL-832 servo motor control cards, a PCL-720 digital I/O card, six amplifier-drivers of the robot servomotors and a power supply.

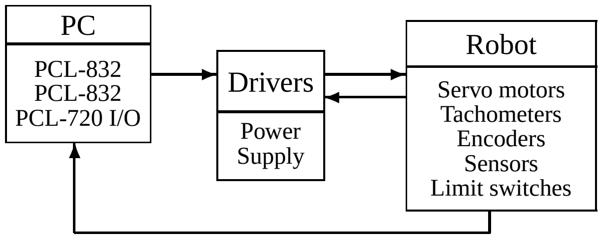

Fig. 2. The structure of the robot control system

3. PCL-832 Servo Motor Control Card

Two servo motor control cards (PCL-832) are inserted into the control PC. Each card can control 3 axes simultaneously, and each axis has its own position control chip, allowing completely independent control. The card can supply a simulated tachometer output to the servo motor driver. This signal makes a tachometer unnecessary in some applications, reducing overall system costs. A special synchronisation circuit synchronises all the axes, called Digital Differential Analysis (DDA). In the described control system two PCL-832 cards are used to control all 6 axes simultaneously. The two PCL-832 servo motor control cards work in such a way that one of them is selected as the master card and the other as the slave card.

4. Digital Differential Analysis (DDA)

In order to obtain synchronisation in a multi-joint servo motor system, all axes should move simultaneously. The principle of DDA method is that all axes start sending position commands simultaneously (T1), and stop sending these commands simultaneously (T2) as well. The duration (T2-T1) is defined as one DDA cycle (see in *Fig. 3*). The duration of the DDA cycle can be set (by the software) from 1 millisecond to 2 seconds as the application requires. For every pulse output the servo motor driver will advance the servo motor one step. One pulse therefore represents one position-command. In one DDA cycle the pulse number can be set from 0 to 4095. The number of pulses in one DDA cycle represents the total position change possible by that DDA cycle. The continuous pulse sequence output to the servo motor driver ensures that a smooth position response is obtained. After the determination of the direction of the motion and the pulse number, the chip itself will control the motion.

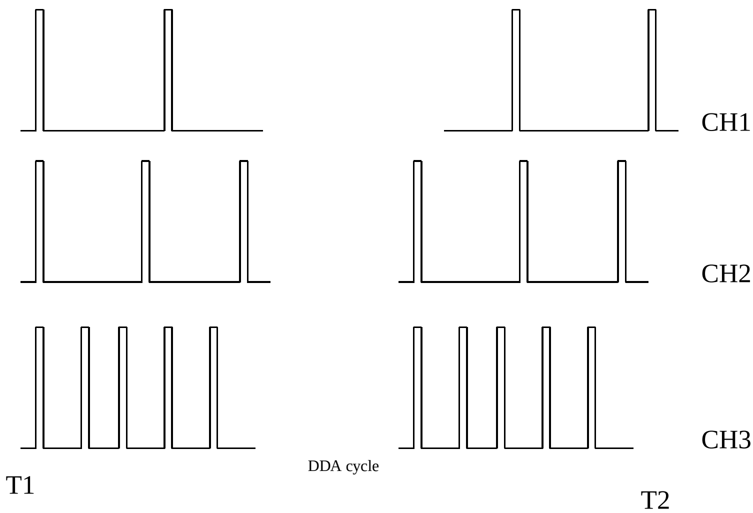

Fig. 3. DDA cycle

5. Closed-Loop Position Control of the Control Card

The servo motor control card uses proportional closed-loop position control to obtain reliable and accurate results. It features an internal velocity feedback loop and an offset technique to compensate for steady-state error that is caused by using small values on P controller. This is a general control technique in robot control, called PD control. The functional block of the closed loop control of the servo motor control card is shown in *Fig. 4*. Physically the servo motor control card can be found inside the dash-dot area. The control software calculates the pulse numbers, then allocates these numbers into the DDA pulse buffer. The DDA generates continuous command pulses, and these pulses are fed into a summing circuit, together with the pulses generated by the servo motor encoder device. The summing circuit determines the difference between both signals and feeds the result signal into the P pulse-offset controller. This module has a programmable gain (Kp). This is the proportional gain of the control loop. After this calculation the signals' pulses are fed into the error counter, which drives the DAC chip in real-time. The velocity block is provided in the motion control chip. Its purpose is to add a velocity feedback loop in the whole system through a frequency-to-voltage (F/V) converter. The maximum input frequency to the F/V converter is 250 kHz. The simulated tachometer output gives −10V to +10V. This internal loop improves the motion dynamics of the servo motor system. This block provides the derivative part of the closed-loop control system. Similar development work is outlined in SOKOLOV, A. (1999), where a LabVIEW based control system for PUMA 560 is described.

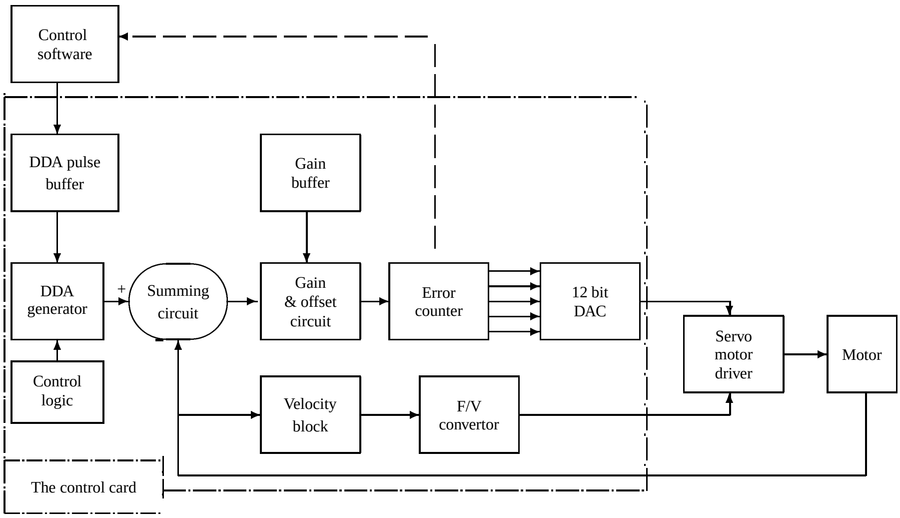

Fig. 4. The structure of closed-loop position control.

6. Modified Closed-Loop Position Control of the Control Card

Basically the described servo motor control card utilises a simple PD control as it is widely used in simple robot control. To improve the accuracy and the motion dynamics of the servomotor and to make the control system open for the user, another feedback is proposed by the author. In *Fig. 4* the dashed lines show the new proposed feedback. Physically the control card utilises two interrupts to make fast connection between the control card and the computer. One interrupt is used for the DDA pulses, and the other is used for the Error Counter. The Error Counter drives directly the DAC chip in real-time. The information about the pulse numbers in the Error Counter is very useful and important for control purposes. The modified control algorithm can receive the real pulse numbers from the Error Counter using the second interrupt and a new subroutine of the control software. This new subroutine has been developed for the control system to make modification – regarding to the transient motion – in pulse numbers. The modification has to be done to less than one DDA cycle that is determined beforehand. Then the modified pulses are fed into the DDA generator. This real time operation is still under development.

7. Programming of the Control Card

The basic development of the control software involves an interface software for making connection between the control card and the LabVIEW software. This

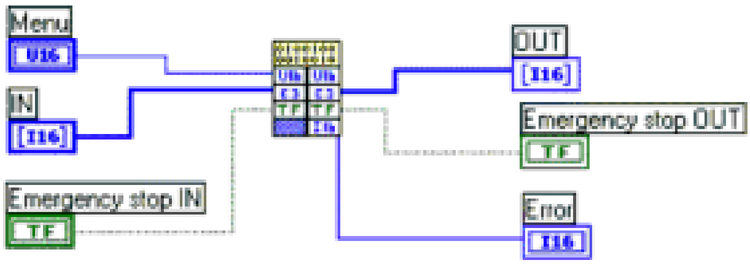

Fig. 5. The CIN function in LabVIEW

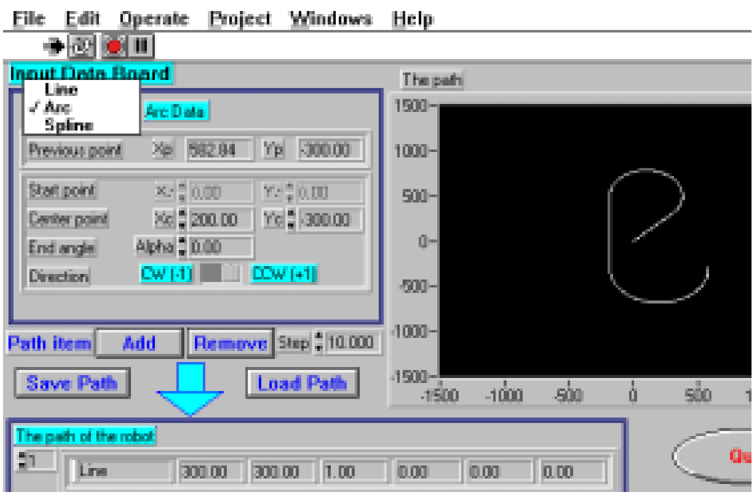

Fig. 6. The path planning module

software contains the basic functions of driving the control card. The LabVIEW software has a Code Interface Node function (CIN), which enables to implement compiled C code into LabVIEW. *Fig. 5* presents the utilisation of CIN in LabVIEW.

Once the driver-software of the control card is implemented in LabVIEW, the user can build up several sophisticated control algorithms utilising the advantages of LabVIEW programming. The control software consists of three main modules. The path-planning module is shown in *Fig. 6*, the trajectory-planning module is presented in *Fig. 7* and the control module is given in *Fig. 8*. These figures are presented here only for demonstration, showing the form of the communications windows.

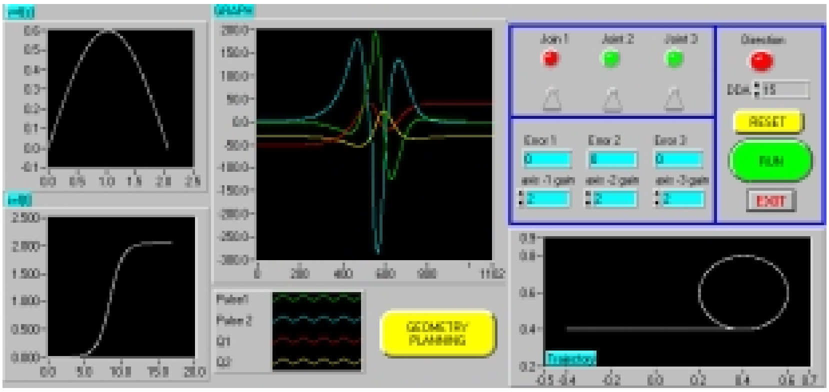

Fig. 7. The trajectory planning module

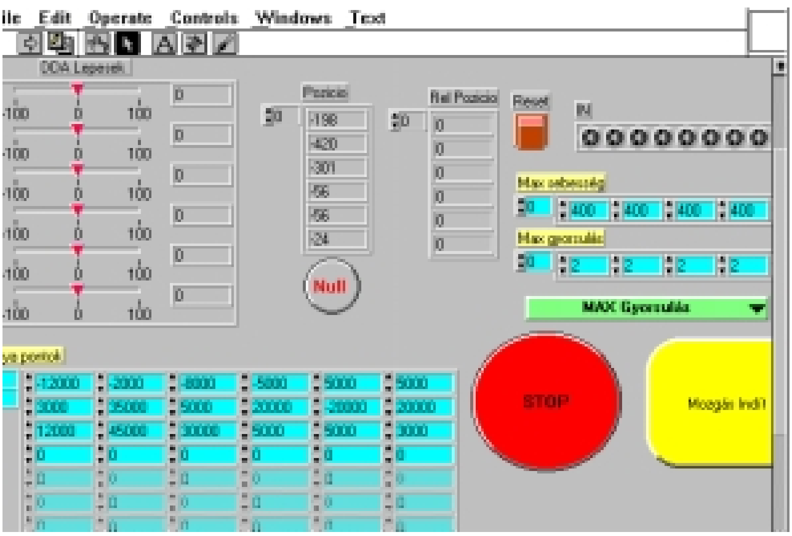

Fig. 8. The control panel of the control software

8. Optimal Cruising Trajectory Planning Method

The robot motion problems are extensively discussed in the literature. Some of the results are given e.g. in SHIN MC.KAY (1985) [1] and S.K. SINGH (1991) [2]. The presented control system is capable to use several sophisticated trajectory planning methods. This paper uses the time optimal trajectory planning method for CPC, which was introduced in SOMLÓ – PODURAJEV (1994) [3]. The time optimal trajectory planning method is used for the cruising part of the constrained path, where the change of speed is slow in comparison with the transient motion. On the transient part of the path the robot uses the maximum acceleration to reach the desired speed value that the cruising trajectory planning method determines. The minimum time cruising trajectory planning method is based on a simple idea. The method uses the maximum absolute value of the speed in every point of the path. This maximum is determined by the limit speed of the joints. Using the time-optimal trajectory planning method the time history of the motion along the path for the cruising part is obtained. This is extremely useful for application planning because in a simple way a clear picture can be obtained about the motion features. It is also important that the generalisation of the results is possible. There is a question which is decisive for the problem: what is the proportion, what is the role of the cruising and the transient motion in the motion as a whole. If the speed limits are low and/or acceleration is high, it is a good solution to use the minimum-time cruising trajectory planning as a basic procedure and complete it by the standard or special investigation of the transient part of the motion. For the transient part of the motion it turns out that using the parametric approach a similar procedure is possible as for the minimum-time cruising. For the transient part the absolute value of the acceleration in every point of the path should be maximised (SOMLÓ – PODURAJEV [4]). The maximum is determined by the maximum of the possible force(torque) at one of the joints.

Fig. 9 presents a path of the robot using 3rd-order Hermite type spline curve.

Using the minimum-time trajectory planning method for the given path, the algorithm results the speed profiles of the joints shown in *Fig. 10*. In *Fig. 10* the minimum-time trajectory planning method is only used for the first three joints (out of six). The equations of the inverse kinematics of the ZIM 15-type robot are presented by G. ERDŐS [9].

9. Conclusion

The presented robot control system utilises a PC based commercial servo motor control card and a LabVIEW based intelligent control software (developed by the author). This implementation provides high robot performance at an affordable price. The basic development of the software enables the user to develop sophisticated control method for different robot applications with the utilisation of the advantages of LabVIEW programming. The author proposed an additional feed-

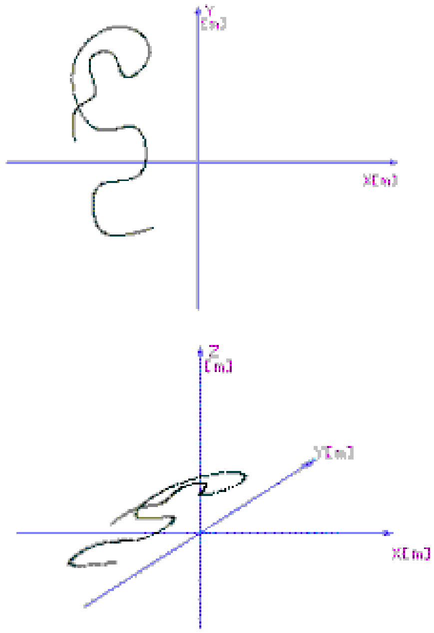

Fig. 9. A path of the robot using 3rd-order Hermite type spline curve

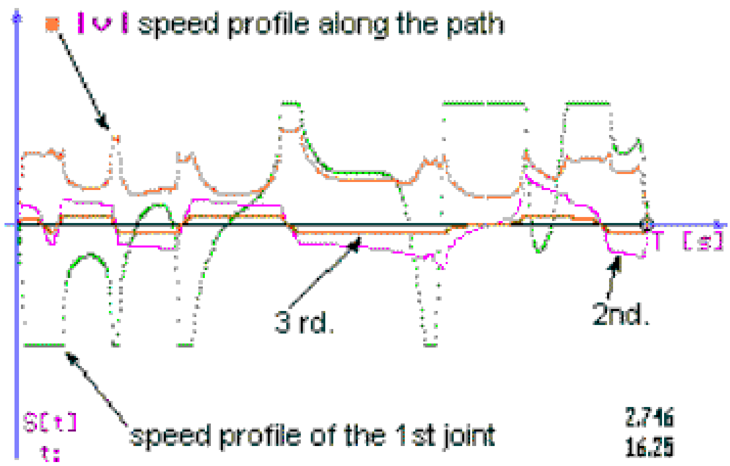

Fig. 10. The speed profiles of the first three joints of the robot, using the given path

back to improve the motion dynamics of the servomotors. The optimal cruising trajectory planning method is presented as one of the sophisticated trajectory planning methods that can be implemented using the control system.

Acknowledgement

The presented achievement was supported by the Research Fund of the Hungarian Academy of Sciences (OTKA T026407) and (OTKA T029072) the author expresses his thanks. The author express sincere thanks to his research advisor Professor János Somló and to his colleague Gábor Erdős.

References

[1] SHIN, K. G. – MCKAY, N. D. (1985): Minimum-Time Control of Robotics Manipulators with Geometric Path Constraints, *IEEE of Automatic Control*. AC-30, 6, pp.531–541.
[2] SHIN, K. G.: Trajectory Planning for Robot Control: A Control System Perspective, Control and Dynamic System Series, Advances in Robotics System, 1991, pp. 105–146 (Academic Press Inc.). e
[3] PODURAJEV, J. – SOMLÓ, J (1993): A New Approach to the Contour Following Problem in Robot Control (Dynamic Model in Riemann Space), *Mechatronics*, **3**, No.2, pp. 241–263.
[4] SOMLÓ, J. – PODURAJEV, J. (1994): Optimal Cruising Trajectory Planning for Robots, *Mechatronics*, **4**, No. 5, pp. 517–538.
[5] TÓTH, S. J: Optimal Trajectory Planning for Robots, MSc Thesis, 1994, Budapest Technical University.
[6] LIPOVSZKI, GY. (1994): TUBSIM Digital Simulation Program, http://www.rit.bme.hu/, /szam-szim/modsim.html
[7] ZENTAY, P. –ZOLLER, Z.: Time Optimal Trajectory Planning for Robots in LabView Programming System MicroCAD'99 (to be published).
[8] SOKOLOV, A.: Robot Motion Realisation Using LabVIEW (to be published).
[9] ERDŐS, G.: 'Ipari robotok geometriai, kinematikai és dinamikai modellezésének egy új szimbolikus módszere' Ph. D. thesis 1988, Budapest.
[10] Advantech (1995): User Manual of the PCL-832 Servo Motor Control Card.