Journal of Achievements in Materials and Manufacturing Engineering
VOLUME 36 ISSUE 1 September 2009

Introduction of serial architecture for small CNC facilities

R.A. Mahdavinejad*
School of Mechanical Engineering, Faculty of Engineering,
University of Tehran, Tehran, Iran
* Corresponding author: E-mail address: mahdavin@ut.ac.ir

Received 20.02.2009; published in revised form 01.09.2009

Analysis and modelling

ABSTRACT

Purpose: The concept of open-architecture control, which has been used as a main method of control in many branches such as automatic systems of machining, robotics, testing and control, causes a great increment in efficiency and precision of control systems and dynamic capability to industrial control networks.
Design/methodology/approach: The high execution costs in one hand and unnecessary usage of all OAC standard components on the other hand, makes the modeling of OAC concept impossible for such systems.
Findings: This paper tries to represent a simple system based on elimination of unessential elements of open-architecture to reach a limited and efficient control system.
Research limitations/implications: In hierarchical architecture systems, regarding to the synchronous processes of calculation and controlling, the total execution time is very low in comparison with similar serial system. In spite of this, the execution time of synchronous processes in serial architecture is lower than hierarchical one.
Practical implications: The utilization of one processor instead of multiple one, merging the industrial control network layer into system management layer and omission of processor execution layer in each axis and optimized using of multi-tasking capability of processor.
Originality/value: The analysis of simulated data in motions with higher degrees of freedom shows that the usage of CNC machines with serial architecture for higher than 3-degrees of freedom is unaccepted due to large increment in total machining time.
Keywords: Open-architecture control; Industrial networks; Controller

Reference to this paper should be given in the following way:
R.A. Mahdavinejad, Introduction of serial architecture for small CNC facilities, Journal of Achievements in Materials and Manufacturing Engineering 36/1 (2009) 57-64.

1. Introduction

Open-architecture control (OAC) is introduced the most efficient control architecture for CNC machines, robots, control and non-destructive test systems in 1990 decade [1,2]. For series of types of technology for one selected technology, the parameters of process are calculated. They are being calculated by: algorithms, analytical programmes, norms[3]. Executive applications of CNC systems in machining, inspections and testing, robotics and other manufacturing operations causes numerous architecture in controls regarding to their facility and capability in various branches of times to be established[4]. The systems based on open architecture, with the aim of control period minimization and accuracy optimization are quickly replaced the traditional control systems. Modulation with high flexibility, development capability, infinite minimization and improvement of control capabilities are contemporary specifications of control process [5]. OAC systems consist of various standard elements for software operations and data transmission protocols. The hardware with control operation tasks is also a section of this standardized system. The openness of these systems not only keeps the coordination to reach the desired accuracy in software and hardware sections, but also lets the designer to do some necessary variations in structure of system [6]. Although the open-architecture systems are convenient in control of complex processes with long distance and multiple axes, but, due to the large number of engaged components and the complexity of data transmission protocols, are not qualified to control small processes and projects. The extension of open-architecture control to separate CAD/CAM, interpolation, running and simulation layers confirms that the most of CNC manufacturers interest to high flexibility obtained from development of software and replacement of old hardware with powerful software [7]. Architechture of agent systems for industrial environment is presented making it possible to generate the particular agents customized for the specific tasks, based on the automatic analysis of its required features[8]. Due to the widespread use of highly automated machine tools in the industry, manufacturing requires reliable monitoring and optimization models and methods[9]. The technical adminstrative and human factors influencing on the quality products must be subjected to the system of control[10]. Control techniques that have been developped for machining traditionally requires some form parameter adaptive[11].

The meaning of small control processes is the control in which the tool positioning distance is short with the least number of synchronous engaged axes. Two-dimensional control cutting systems and non-contact non-destructive testing systems are the samples of these kinds of applications with general use in industry. This article presents a powerful serial – architecture control system, which is useful in small control applications.

As a principle, the shortening of control cycle and summarized processing of total layers with one processor is done. In addition, a sample of necessary software and hardware algorithms is presented to ensure the action of each layer.

2. General framework CNC machines

In a CNC machine to control tool along its path, various sections as interpolation unit, linear velocity generation unit, management and simulation units and the hardware to control the motion of each axis are needed [12]. Software interpolation and linear velocity generation are done independently from the type of system architecture. In addition, one or several safe data transmission protocols are needed for simulation and data management and other real-time tasks. Simulation and the management of data are done via a software independent from interpolation due to motion algorithms. In an open-architecture control system, the arrangement of the above units will be accomplished layer by layer to reach a logic connection of data/algorithm between layers [13]. Figure 1 shows open-architecture control systems with simultaneous interpolation and linear motion control. The connection between the main control program, application software and abstract model of machine at application layer is a perfect software connection. According to the layout and connection methods, the hierarchical structure is performed.

In object management layer, object manager software is used for data collection and transferring from/to local controllers. A Micro-Kernel-based OS system is used to control the network (to data /algorithm connection) and device drivers. The relation between object management program, network driver and control equipment is done with Micro – Kernel-based operational system

The hierarchical arrangement in the device driver layer is also performed. I/O connectors at the last layer are directly connected to object management program via network drivers. In general case, to program such an OAC system, following parts are needed:

- Mechanical hardware of CNC or robot system.
- A processor like Intel 80 *86/ Pentium™, IBM PowerPC™ or so on.
- The corresponding operation system with processor as; windows NT, MAC, or LINUX.
- Sensors and I/O interfaces. The industrial network controller with CAN, MPI, MODBUS, PROFIBUS standards or so on.
- The related software are used to interpolate, generate velocity and object management programmed independently from hardware.
- Standard controller of axes.

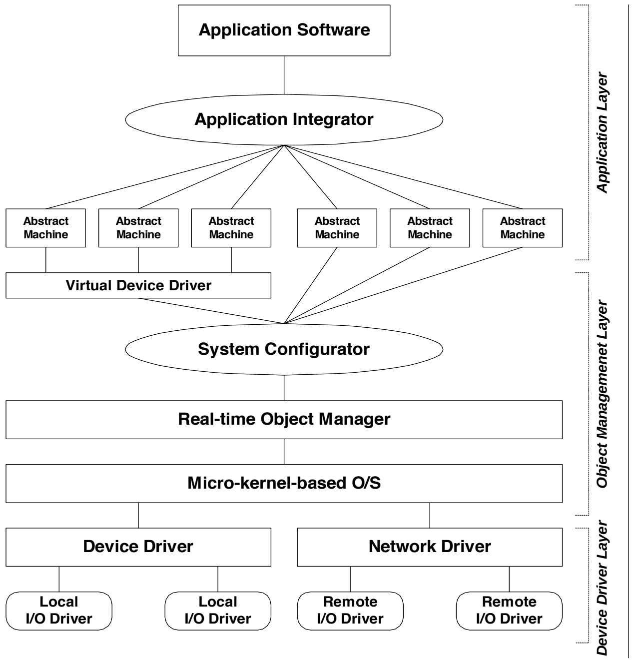


Fig. 1. Schematic of open-architecture control method

In a system consist of several machines; an industrial controller network is used for data collection and transmission in the middle and lower layers [14]. Using industrial network controllers let the designer to increase or decrease the network controlled nodes and hardware according to the industrial necessity.

3. CNC architecture and control cycle

The control cycle is determined as the time needed to system self-recovery so that it will be ready for the next task. The length of the control cycle is somehow, the result of the system inertia, so that, its increasing causes increasing of reaction time and decreasing in final accuracy of the control system. Therefore, increasing in the number of delay creating elements in CNC machine decreases the total accuracy of the system]. The main delay creation resources in a machine with open-architecture controller are as follow:

- Calculation and control time in execution layer (closed loop control period)- t_{CP}.
- The transmission data time via industrial network from trajectory generator to the execution layer processor- t_{RC}.
- The delay created from sensors during reading process-t_{PC}.
- The transmission data time from supervisory controller to the trajectory generator unit-t_{SR}.

According to the synchronous trajectory generation with minor time processing, the next coordinate and velocity generation is supposed to be the control target in each period. So, the time needed to interpolate and generate the desired velocity will not be considered in the total time calculation. Figure 2 shows an OAC system in the hierarchical form that delay resources with persist on their non-linear effect is exhibited.

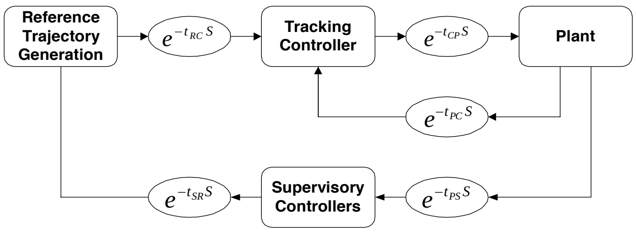


Fig. 2. A hierarchical CNC control with communication delays

We can decrease the length of the control period with the omission of some of the intermediate operations if it is possible to discrete the whole control processing of the trajectory into synchronous and non-synchronous groups. It is noticeable that for the connection and execution operations in CNC open-architecture systems, powerful processors with different process capabilities are used on all calculation layers and the transmission protocols joint processing network to each other. The elimination of these intermediate operations results often the omission of the related processors. Therefore, in addition to necessity of keeping the synchronous control operation and active modularity separation of these systems we should distribute the different unit tasks between limited numbers of processors. This approach guides the designer to the serial architecture.

4. Serial architecture for CNC machine

A serial architecture that is schematically shown in Figure 3, consists of the whole elements of hierarchical control systems with different communication methods and logical place for elements with unique processor.

Non-synchronous process in an OAC system with serial arrangement is as follows:

- Interpolation (set-point creation for positioning).
- Velocity generation (set-point creation for velocity control).
- Operations of high level motional coding creation with CAD/CAM software (G- Code).

Synchronous operations in such a serial system are as:

- Retrieval of velocity-positioning data from database.
- Correction of data in each stage by the error of previous stage via a cross-coupled controller.
- Single-axis control action.

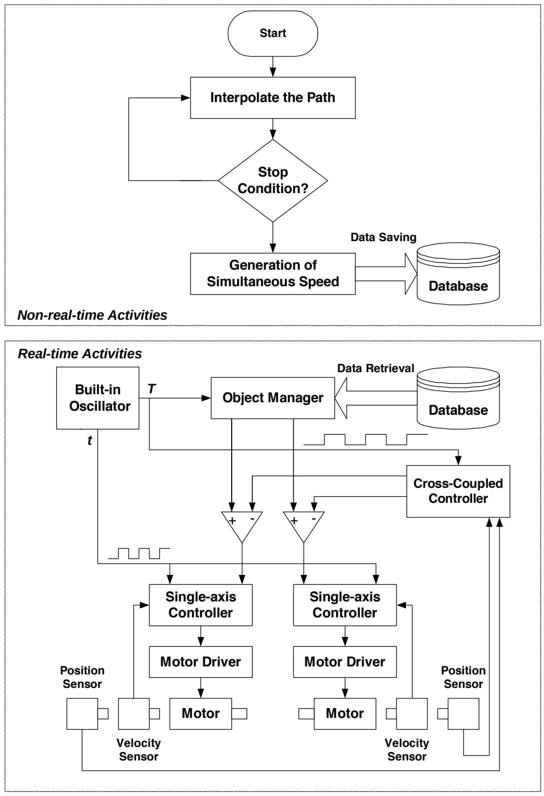


Fig. 3. Serial architecture to control a multi-axis system

It is obvious that there is no industrial network controller with its related data transmission delay. Indeed, the data, which are transmitted by these networks, are produced at non-synchronous process stage and after storing in database; they will be recycled in synchronous period again. Basically, an industrial single-board computer as executer of all control process based on software and hardware controller due to summarization processing operations in a processor can be used. The elimination of data transmission network and afterwards, the omission of the corresponding data transmission time cycle, is the first gain of this system.

In order to execute the basic concept of serial architecture, it is necessary to determine the software and hardware details.

4.1. Interpolation in serial architecture

Interpolation is used to generate the next positioning point along its direction via a mathematical algorithm with analytical and parametric input curves. Analytical curves in general line and circle's arc for the use in two different types of motion systems (stepwise and continuous motion systems) can be interpolated by two following methods:

Four Point Algorithm (FPA) Method

In this method, software interpolator needs the explicit form of motion equations. The interpolation of various curves has to be transferred into linear or circular form.

In four point method, the motive system can move from current position in 2-dimensional motion to one of up, down, left and right directions. Motion space is a matrix space with a network width of one BLU (basic length unit). The optimum path is the path with the least sum of square distances of interpolated points. The directions of motion from starting point to the end and vise versa for linear motions, and clockwise or counterclockwise for circular motions, can be determined by user.

Concavity and inclination of curves and motion directions define the next motion points. The final path information are stored as relative motions in a database of hard disk. Figure 4 shows the interpolation flowchart of the FPA method. The results of FPA can be developed to six points in 3-dimensional space.

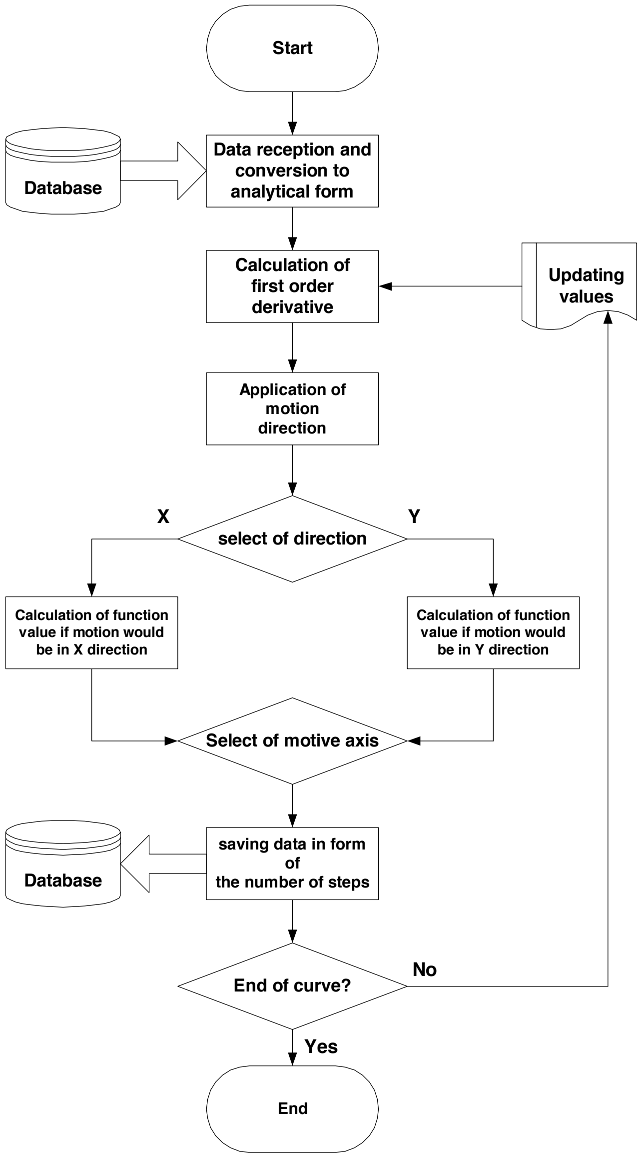


Fig. 4. Flowchart of a 2-dimensional FPA algorithm

Velocity-Based Interpolation

In this interpolation method, basically, the instantaneous linear speed in each axis is used to dynamic control of motor's rotational speed and afterwards, control of the velocity integral term (namely instantaneous position). The principle of this interpolation method is limited on making component tangent to the motion path curve along engaged axis and with constant linear velocity assumption along the path (Figure 5).

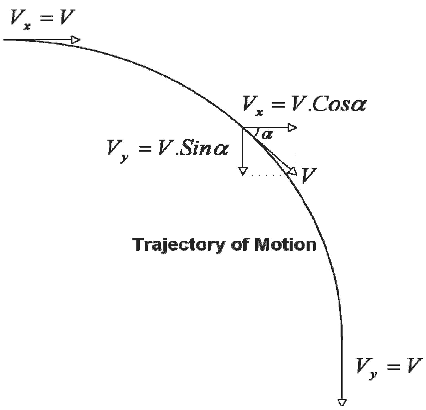


Fig. 5. Schematic expression for velocity-based interpolation

Suppose t_C is the length of control cycle and transmission of the data in CNC system and V is the motive velocity along the path, then length of the interpolation cycle namely new V components generated along engaged axis have to be equal or less than t_C. In this case the new data generated for interpolation can be generated before velocity/motion control at the previous time domain.

Suppose that T and ΔL are the length of the interpolation cycle and parts respectively, then:

$\Delta L = T.V$

In each ΔL bound, the motion is supposed to be linear, so the linear error will be very small in the short length of interpolation cycle. Finally, instantaneous velocity data of axis, the coordinate of each interpolated node and the coordinate of the node at the end of path are respectively stored in a database

Interpolation of Parametric Curves

Parametric curves like NURBS and BEZIER have many industrial applications. Regarding to the different nature of parametric free form curves, there is a simple data generation method for velocity/position in each time interval T. The parametric form of a curve is:

$$x = x(u) \ ; \quad y = y(u) \ ; \quad z = z(u) \tag{1}$$

where $0 \le u \le 1$ is the parameter of curve. $u = 0$ and $u = 1$ are corresponded to the start and the end points of the curve respectively.

In general point of view, the interpolation of a parametric curve is the determination of u variations in T intervals, so that, the result of velocity components in the direction of the engaged axis are constant. The linear velocity along the curve corresponds to the time variation rate of u, so that:

$$V(u)=\frac{ds}{dt}=\frac{ds}{du}\frac{du}{dt}\rightarrow\frac{du}{dt}=\frac{V}{ds/du} \tag{2}$$

ds is a 3-dimensional differential motion. On the other hand:

$$x'=\frac{dx}{du},\ y'=\frac{dy}{du},\ z'=\frac{dz}{du} \tag{3}$$

and:

$$\frac{ds}{du}=(x'^2+y'^2+z'^2)^{1/2} \tag{4}$$

$$\frac{du}{dt}=\frac{V}{(x'^2+y'^2+z'^2)^{1/2}} \tag{5}$$

where V is the desired linear velocity.

Using Taylor Expansion around the function $t=K.T$ (K is the number of repetition and T is the length of interpolation cycle). $u(t)$ can be approximately calculated so that:

$$u_{k+1}=u_k+T\dot{u}_k+(T^2/2k)\ddot{u}_k+\cdots \tag{6}$$

where:

$$\dot{u}=\frac{du}{dt},\ \ddot{u}=\frac{d^2u}{dt^2},\text{and}\ldots\ldots\ldots \tag{7}$$

If T is very small, then:

$$u_{k+1}=u_k+T\dot{u}_k \tag{8}$$

Substituting $\dot{u}_k$ from equation (5) to equation (8):

$$u_{k+1}=u_k+\frac{VT}{(x'^2+y'^2+z'^2)^{1/2}} \tag{9}$$

According to the possibility of $x'(u)$, $y'(u)$ and $z'(u)$ calculations from primary equations, the value of u in each time interval T will be determined.

The whole values of $x'(u)$, $y'(u)$, $z'(u)$, $x(u)$, $y(u)$ and $z(u)$ including the coordinate of the end point of curve will be stored in database for control operation.

4.2. Velocity generation

The velocity generation action is said as instantaneous set-point creation of velocity control in each axis. Obviously, in velocity-based and parametric curves interpolations, the instantaneous velocity is obtained during the execution of interpolation algorithm. In FPA method that the interpolation is completely based on positioning, it is necessary to create an algorithm to control of instantaneous velocity.
The instantaneous velocity in each motion step, created via FPA interpolation is as follows:

$$V\approx\bar{V}=\frac{ds}{dt}=\frac{(\Delta^2x+\Delta^2y)^{1/2}}{\Delta t} \tag{10}$$

where Δx and Δy are assumed to be the length of X and Y steps in each piece of path respectively and Δt is the time needed for passing through two pieces, namely:

$$\Delta t=\Delta t_x+\Delta t_y \tag{11}$$

$$\bar{V}=\frac{\Delta s}{\Delta t}\Rightarrow\Delta t=\frac{\Delta s}{V} \tag{12}$$

$$\Delta t_x+\Delta t_y=\frac{(\Delta^2x+\Delta^2y)^{1/2}}{\bar{V}} \tag{13}$$

If Δt is very small:

$$V=\lim_{\Delta t\to 0}\frac{\Delta s}{\Delta t}=\frac{ds}{dt}\approx\frac{\Delta s}{\Delta t} \tag{14}$$

The values of Δx and Δy as interpolation program output and $\bar{V}$ as feeding speed are available from the input of the program. The angle α is attributed as the angle of motion vector between the starting and the end points of imaginary motion path with X axis, so:

$$Cos\alpha=\frac{\Delta x}{\Delta s}=\frac{\Delta x}{(\Delta^2x+\Delta^2y)^{1/2}} \tag{15}$$

$$Sin\alpha=\frac{\Delta y}{\Delta s}=\frac{\Delta y}{(\Delta^2x+\Delta^2y)^{1/2}} \tag{16}$$

$$\bar{V}_x=\bar{V}Cos\alpha=\frac{\bar{V}\Delta x}{(\Delta^2x+\Delta^2y)^{1/2}} \tag{17}$$

$$\bar{V}_y=\bar{V}Sin\alpha=\frac{\bar{V}\Delta y}{(\Delta^2x+\Delta^2y)^{1/2}} \tag{18}$$

In this manner, the linear speed of axes to reach to a constant speed in each part of imaginary motion path will be obtained. In a 3-dimensional space:

$$\Delta s=(\Delta^2x+\Delta^2y+\Delta^2z)^{1/2} \tag{19}$$

In comparison between equations (17), (18) and (19), the velocity of triple axis will be determined:

$$\bar{V}_x=\frac{\bar{V}\Delta x}{(\Delta^2x+\Delta^2y+\Delta^2z)^{1/2}} \tag{20}$$

$$\bar{V}_y=\frac{\bar{V}\Delta y}{(\Delta^2x+\Delta^2y+\Delta^2z)^{1/2}} \tag{21}$$

$$\bar{V}_z=\frac{\bar{V}\Delta z}{(\Delta^2x+\Delta^2y+\Delta^2z)^{1/2}} \tag{22}$$

Moreover, in the same manner the rotational speed in each motion axis is as follows:

$$\theta_x = \frac{\bar{V}_x}{k_x} ; \theta y = \frac{\bar{V}y}{ky} ; \theta z = \frac{\bar{V}z}{kz} \qquad `(23)$$

k_i Coefficients are the product transform coefficients of rotational to linear motions in motional transforming process. θ_i Calculation means to determine the speed of motor in each step of motion trajectory.

4.3. Object management program

The object management in a serial architecture is done by a program that has the responsibility of database creation for speed and motion and the connection with hardware controller. The object management program uses a built-in oscillating source with a constant reference frequency for timing the process of correction of data interpolated. This program executes the main core of timing in synchronous processes. The jobs of OMP are as follow:

- Data retrieval of instantaneous speed/motion from databases in each T interval.
- Data correction of instantaneous speed/motion via cross-coupled controller software.
- Parallel sending of speed/motion data to single-axis controller.

Module of Cross-Coupled Controller

In order to correct the motion data, based on the error of previous motion execution, a software module that can access the relative position value at the end moment of sending motion command, is used. The value and direction of instantaneous velocity vectors changes so that the motion along the velocity results to reach the next interpolated point. Figure 6 shows the principle operations of cross-coupled controller.

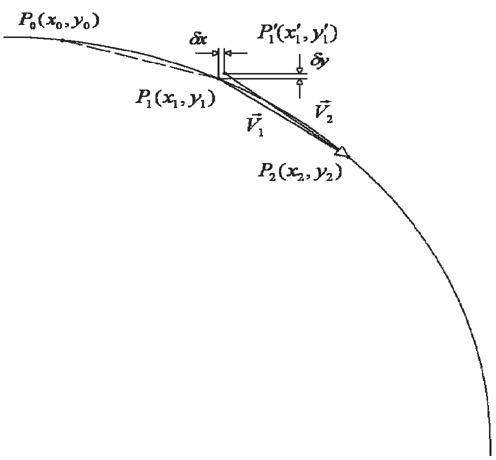


Fig. 6. Basis of cross-coupled controller operation

This method can be uses for error compensation in all interpolation methods. Suppose δx and δy are the errors along two motion axes. The V_2 vector with V_{2x} and V_{2y} components, as a new velocity vector, can be calculated as follows:

V_1 from (x_1, y_1) to (x_2, y_2) , V_2 from (x_1, y_1) to (x_2, y_2)

$$\left|V_{1x}\right| = \left|\frac{x_2 - x_1}{T}\right| , \left|V_{1y}\right| = \left|\frac{y_2 - y_1}{T}\right| \qquad (24)$$

$$\left|V_{2x}\right| = \left|\frac{x_2 - x_1 + \delta x}{T}\right| \qquad (25)$$

$$\left|V_{2y}\right| = \left|\frac{y_2 - y_1 + \delta y}{T}\right| \qquad (26)$$

δx and δy have + or – signs. The + sign shows that the direction of the error and motion are the same and vise versa. The cross-coupled control has clearly a little volume calculation and it can prevent the system against the large error from the motion control of external point.

4.4. Motion controller of a single axis

A motion controller of single axis based on computer consists of a number of modular software and hardware to receive the position and speed data from digital and analog sensors and to catch set-points from object management program. The output of this unit is the signal pulses related to the motor's driver and the data related to the motors instantaneous torque. Figure 3 shows the connection between sensors with cross-coupled and single axis controllers. There are the voltage-current drivers of motors at the layers after single axis controller.

4.5. Timing in serial architecture

In serial architecture of a CNC machine a built-in internal oscillator is used to determine time units with square waveforms. The timing controller unit of single axis motion, which is shown with symbol t, is a very short amount between .001 to .002 second range. In spite of this, the basic interpolation time (T), which is a multiple integer coefficient of single axis control cycle (t), is between .05 to .1 second. In order to create a sufficient opportunity for controlling process, the interpolation cycle time deliberately is considered a large amount indeed. It is noticeable that, all clock receivers have to be software or hardware elements of interrupt-oriented in order to multi-tasking processor and operational system have no incorrect effect on control execution process.

4.6. Processor in serial architecture

According to what mentioned previously, a single processor is used in serial architecture to execute total calculation and control processing. Naming serial architecture is basically due to serially position of processor during interpolation process, linear velocity generation, retrieval and data correction and control of single axis process. It is tried to optimize utilization of multi-tasking processor and operational system, so that, the timing system not to be disordered.

5. Testing of the system

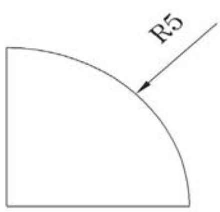
In order to analyze the effects of control architecture variation on the length of control cycle and the quality of machining surface, a 3-axes CNC milling machine with the following possibilities and serial architecture are used:

- Single board industrial computer with Intel Pentium ™ 133MHz processor and PCI slot.
- 16 bits digitally data transmitter board of shaft encoder with PCI standard.
- Data transmission and conversion board for analog tachometer with PCI standard.
- Free-running pulse width modulation driver system to drive two DC servo-motors and instantaneous revolution-torque control.
- Digital data transmission board with PCI standard and 20 MHz built-in oscillator.
- Interpolation and velocity generator written in Visual Basic software under Win. NT operational system.
- Object management software written in Delphi under Win. NT operational system.
- Cross-coupled controller written in Delphi under Win. NT operational system.

A similar system with hierarchical open-architecture in contrary used to compare the length of control cycle:

- A motive system with three degrees of freedom.
- An industrial single board computer with Intel i486/33 MHz processor.
- The sensors of motion, velocity and converter circuits related to computer base.
- Robotool commercial CNC controller.
- Industrial network controller with controller area network (CAN) standard, Synchronous interpolator based on velocity.

First, the delayed values in various processing stages on machining of a 2-dimensional contour (Figure 7) in two systems are measured and final product is dimensionally and geometrically tested on 3-dimensional CMM with 1μm accuracy.



Simple milling part to test contouring system:
A quarter-circle with radius of 5mm
and thickness of 5mm
Material: Mild Steel (St37)

Fig. 7.Test part used for accuracy determination

Figures 8 and 9 show the dimensional accuracy and the total machining process time from start point of machining execution command of the G-code written.

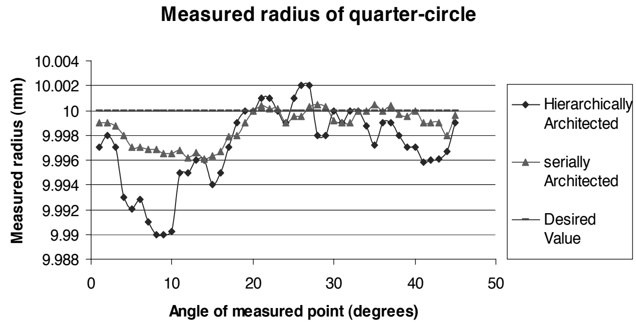


Fig. 8. Dimensional precision of the test part machined via two CNC systems

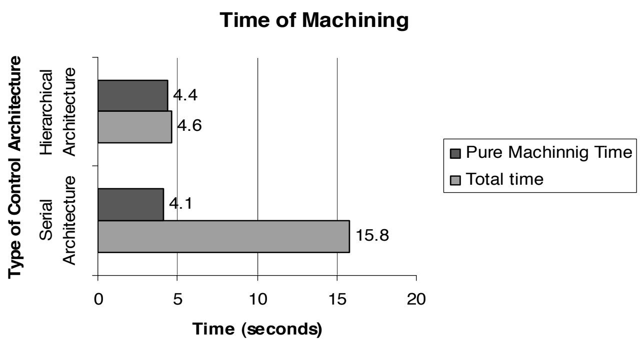


Fig. 9. Total and pure machining times of the test part

The effect of control cycle length on the maximum error of contour in hierarchical and serial architectures has also been analyzed (Figure 10).

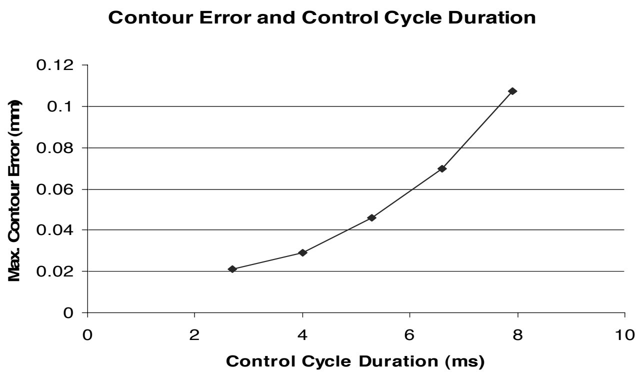


Fig. 10. Relation between contour error and control cycle duration

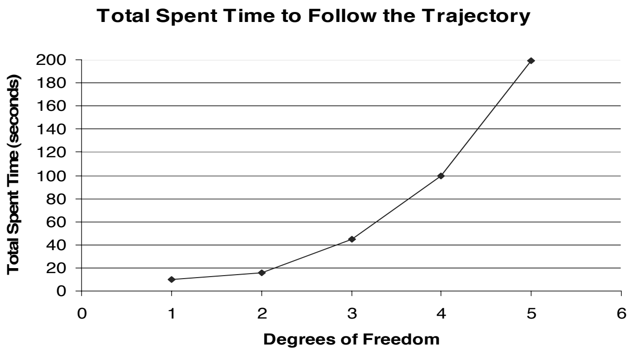


Fig. 11.Relation between total spent time to standard trajectory and degrees of freedom controlled

In addition, the time execution of synchronous and non-synchronous processes in a numerical control system with serial architecture according to the motion simulation for the higher degrees of freedom has been analyzed (Figure 11).

6. Conclusions and results

A simple and skilled architecture for small control applications in robots and CNC machines has been introduced. The following results from error diagrams, testing and simulation analysis has been obtained:

- The open-architecture control is noun as an efficient and safe method for arrangement of working sections in CNC machines. But in spite of this, the independency to the number of engaged axes, flexibility and extendibility are caused its high execution costs, so that, its application for small control jobs is not justified.
- In serial architecture for short control processes with the small number of engaged axes, in spite of noticeable reduction costs in execution due to the hardware concentration of control in a PC and simple circuits based on PC, the control cycle length in noticeably reduced due to its serially operations.
- Analysis the parts of produced by hierarchical and serial CNC systems, shows the dimensional and geometrical accuracy reduction with increasing in control cycle length. From this point of view, serial architecture usage instead of similar hierarchical control systems is noticeably approved the quality of production.
- The total execution times of resemble motions in hierarchical architecture and serial systems are quite different. In hierarchical architecture systems, regarding to the synchronous processes of calculation and controlling, the total execution time is very low in comparison with similar serial system. In spite of this, the execution time of synchronous processes in serial architecture is lower than hierarchical one.
- The analysis of simulated data in motions with higher degrees of freedom shows that the usage of CNC machines with serial architecture for higher than 3-degrees of freedom is unaccepted due to large increment in total machining time.

References

[1] L. Luscombe, D.W. Thompson, R.Dluzniak, A new type of machine control system to replace traditional CNC, International Journal of Advanced Manufacturing Technology 9 (1998) 369-374.

[2] J. Park, S. Birla, K.G. Shin, Z.J. Pasek, G. Ulsoy, Y. Shan, Y. Koren, An open-architecture test bed for real-time monitoring and control of machining process, Proceedings of the American Control Conference, 1995, Seattle, Washington.

[3] P. Gendar, R. Rzasinski, P. Chyra, Technological similarity in process of series of type technology creating, Journal of Achievement in Materials and Manufacturing Engineering 29/1 (2008) 159-162.

[4] I. Budak J. Hodolic, M. Sokovic, Development of a Programme System for Data-Point Pre-Processsing in Inverse Engineering, Proceedings of the 13th International Scientific Conference "Achievements in Mechanical and Materials Engineering" AMME'2005, Gliwice − Wisła, 2005, (CD-ROM).

[5] M.Y. Park, J.H. Yang ,A study on an open-architecture CNC system with a NURBS interpolator for WEDM, International Journal of Advanced Manufacturing Technology 19 (2002) 664-668.

[6] A. Buchacz, Dynamical flexibility of torsionally vibrating mechatronic system, Journal of Achievements in Materials and Manufacturing Engineering 26/1 (2008) 33-40.

[7] F. Cus, M. Milfelner, J. Bulic, An Overview of Data Acquisition System for Cutting Force Measuring and Optimization in Milling, Proceedings of the 13th International Scientific Conference "Achievements in Mechanical and Materials Engineering" AMME'2005, Gliwice − Wisła, 2005, (CD-ROM).

[8] J. Madejski, Agent architecture for intelligent manufacturing systems, Journal of Achievements in Materials and Manufacturing Engineering 29/2 (2008) 167-170.

[9] T. Dzitkowski, A Dymarek, Design and examining sensitivity of machine driving systems with required frequency spectrum, Journal of Achievements in Materials and Manufacturing Engineering 26/1 (2008) 49-56.

[10] M. Dudek-Burlikowska, K. Szewieczek, Customer's satisfaction the element of proquallity strategies of organization, Journal of Achievements in Materials and Manufacturing Engineering 28/1 (2008) 91-94.

[11] F. Cus, U. Zuperl, Model reference-based machining force and surface roughness control, Journal of Archives of Materials Science and Engineering 31/2 (2008) 117-120.

[12] C. Grabowik, K. Kalinowski, Z. Monica, Integration of the CAD/CAPP/PPC Systems, Proceedings of the 13th International Scientific Conference "Achievements in Mechanical and Materials Engineering" AMME'2005, Gliwice − Wisła, 2005, (CD-ROM).

[13] Y. Altintas, W. K.Munasinghe, A hierarchical open-architecture CNC system for machine tools.Annals of CIRP, 43/1 (1994) 349-354.

[14] J. Kopac, High precision machinig on high speed machines, Journal of Achievements in Materials and Manufacturing Engineering 24/2 (2007) 405-412.