Synthesis of Multi-core Architecture for Optimal Control

Radu MELNIC, Ion BALMUS, Victor ABABII, Mariana CATRUC
Technical University of Moldova
rmelnic@mail.utm.md

***Abstract* — This paper offers a solution to the problem of optimal control system synthesis based on uniform multi-core architecture. For this purpose the equations that describe the control system are decomposed into a set of homogeneous equations with the same complexity and the search for the minimum value of the functional optimization is done as a sequential iterative search using a set of one-dimensional equations. In order to ensure maximum efficiency, computational power of the optimal solutions management, is uniformly distributed on a set of computing architecture controllers. AVR microcontrollers and industrial network standard I2C series were used to study the behavior of the control system. In the design process there where developed: math models, the structure of the control system, data protocol, the behavioral algorithm and the circuit of the control system based on four microcontrollers.**

***Index Terms* — control system, optimal control, multi-controller architecture, I2C bus.**

I. INTRODUCTION

Currently, the sphere of the multiprocessor systems is continuously expanding, covering new domains in various fields of science, business and production. One of such areas is solving optimal control problems of technological or manufacturing processes in real time. The classical problem of optimal control is the system of equations (1), where it is necessary to find the minimum value of the functional $\boldsymbol{Q}$ (2) when maintaining restrictions $\boldsymbol{x(t)}$ (3) and controlling $\boldsymbol{u(t)}$ (4) [1, 2]:

$$\frac{\boldsymbol{dx}}{\boldsymbol{dt}} = \boldsymbol{f}\left(\boldsymbol{x},\boldsymbol{u}\right) \qquad (1)$$

$$\boldsymbol{Q}\left(\boldsymbol{x},\ \boldsymbol{f}\left(\boldsymbol{x}\right),\boldsymbol{u},\boldsymbol{g}(\boldsymbol{u});\boldsymbol{t}\right) \to \min \qquad (2)$$

$$\boldsymbol{x} \in \mathrm{X}\text{, where } \boldsymbol{x} = \left(\boldsymbol{x}_i, \boldsymbol{i} = \overline{1,\boldsymbol{N}}\right)^{\boldsymbol{T}} \qquad (3)$$

$$\boldsymbol{u} \in \mathrm{U}\text{, where } \boldsymbol{u} = \left(\boldsymbol{u}_i, \boldsymbol{i} = \overline{1,\boldsymbol{N}}\right)^{\boldsymbol{T}} \qquad (4)$$

The solution of the functional (2) reduces to successive iteration method for solving system of equations (5).

$$\boldsymbol{Q}_i\left(\boldsymbol{x},\frac{\partial \boldsymbol{x}_i}{\partial \boldsymbol{t}},\boldsymbol{u},\boldsymbol{g}\left(\boldsymbol{u}\right),\boldsymbol{t}\right) < 0, \Rightarrow \boldsymbol{Q} \to \min,\ \forall\, \boldsymbol{i} = \overline{1,\boldsymbol{N}} \qquad (5)$$

The solution of the system of equations (5) is to find the minimum value of the functional (2) through a gradient method in $\boldsymbol{N}$ - dimensional space.

The purpose of this paper is to adapt the solution of optimal control for distributed multi-controllers computing architecture while minimizing the amount of information transmitted between the controllers.

II. PROBLEM STATEMENT OF THE CONTROL SYSTEM

In order to ensure a uniform distribution of computing power the system of equations (1) on optimal condition (2) is decomposed into $\boldsymbol{N}$ homogeneous equations of the form:

$$\frac{\boldsymbol{dx}_i}{\boldsymbol{dt}} = \boldsymbol{f}\left(\boldsymbol{x}^*(\boldsymbol{t}),\boldsymbol{u}^*(\boldsymbol{t})\right),\ \forall \boldsymbol{i} = \overline{1,\boldsymbol{N}}, \qquad (6)$$

Where: $\boldsymbol{x}^*(\boldsymbol{t}) = \left(\boldsymbol{x}_i^*(\boldsymbol{t}),\ \boldsymbol{i} = \overline{1,\boldsymbol{N}}\right)$ - state of the process and $\boldsymbol{u}^*(\boldsymbol{t}) = \left(\boldsymbol{u}_i^*(\boldsymbol{t}),\ \boldsymbol{i} = \overline{1,\boldsymbol{N}}\right)$ - control signals at the time $\boldsymbol{t}$.

Starting from the expression (5) and (6), in order to maximize the effectiveness of multi-computing architecture you have to perform the condition of uniform distribution of computer power as follows:

$$\begin{cases} \dfrac{\boldsymbol{dx}_i}{\boldsymbol{dt}} = \boldsymbol{f}\left(\boldsymbol{x}^*(\boldsymbol{t}),\boldsymbol{u}^*(\boldsymbol{t})\right), \\ \boldsymbol{Q}_i\left(\boldsymbol{x}^*,\dfrac{\partial \boldsymbol{x}_i}{\boldsymbol{dt}},\boldsymbol{u}^*,\boldsymbol{g}(\boldsymbol{u}^*),\boldsymbol{t}\right) < 0, \end{cases} \text{"}\boldsymbol{i} = \overline{1,\boldsymbol{N}}. \qquad (7)$$

III. CONTROL SYSTEM SYNTHESIS

To solve the problem (6) considering (5), a MIMD class (Multiple Instruction Multiple Data) [3] multi-controller computing system (MCS) is proposed (Figure 1). MCS is a homogeneous distributed computing architecture based on the industrial network $\boldsymbol{I^2C}$ [4] and set of processing elements $PE_i, i = \overline{1,N}$. A controlled process is represented by many technological or manufacturing operations $TO_i, i = \overline{1,N}$, where $DI_i, i = \overline{1,N}$ - a state signal of the process operation and $DO_i, i = \overline{1,N}$ - the control signal of the process operation.

The functionality of a multi-computing system is based on parallel calculation of the following operations: data input about the process state $DI_i, i=\overline{1,N}$, data exchange between the computational elements $PE_i, i=\overline{1,N}$, optimization and calculation of control signals according to expressions (7) and the impact on controlled process signals $DO_i, i=\overline{1,N}$.

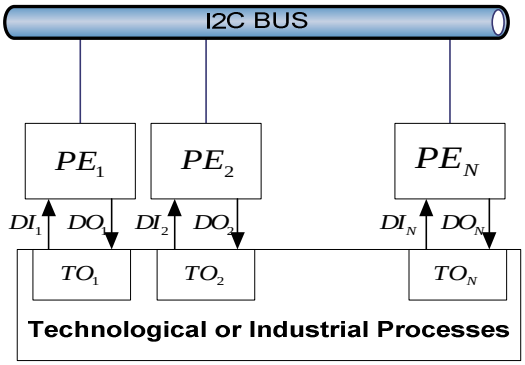

Figure 1. The structure of a computing system.

IV. COMMUNICATION PROTOCOL SYNTHESIS

To exchange data between the computational elements $PE_i, i=\overline{1,N}$ a special protocol is developed, which is a bit-vector $\boldsymbol{BV_x}$ change in the state of the controlled process and state of the process $\boldsymbol{BV_u}$. Figure 2 shows the method of calculating the values of $\boldsymbol{BV_x}$ and $\boldsymbol{BV_u}$.

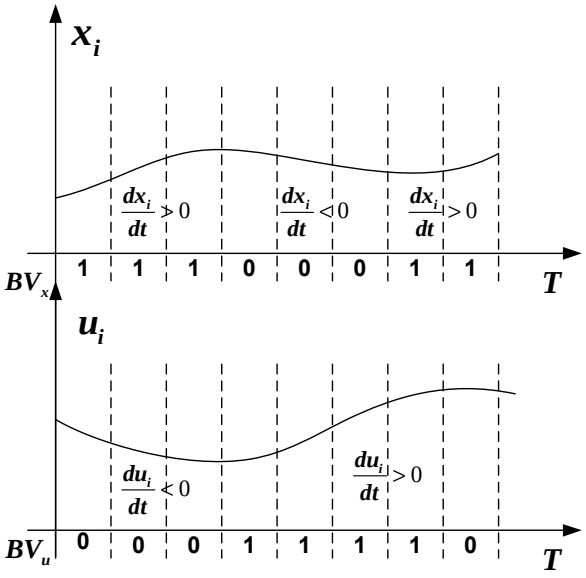

Figure 2. The method of calculating the bit-vectors.

To determine the values of bit-vectors the sign of the first derivative signal of the state of the process (8) and control is calculated (9) from the following conditions:

$$\begin{cases} \textbf{If } \dfrac{dx_i}{dt} > 0, \textbf{ then } \boldsymbol{BV}_{x_i} = 1, \\ \textbf{If } \dfrac{dx_i}{dt} \le 0, \textbf{ then } \boldsymbol{BV}_{x_i} = 0. \end{cases} \quad \forall \boldsymbol{i} = \overline{1, N}, \qquad (8)$$

$$\begin{cases} \textbf{If } \dfrac{du_i}{dt} > 0, \textbf{ then } \boldsymbol{BV}_{u_i} = 1, \\ \textbf{If } \dfrac{du_i}{dt} \le 0, \textbf{ then } \boldsymbol{BV}_{u_i} = 0. \end{cases} \quad \forall \boldsymbol{i} = \overline{1, N}. \qquad (9)$$

The purpose of bit-vector use is to decrease the amount of sending information between the computational elements $PE_i, i=\overline{1,N}$.

Recovery of standing $\boldsymbol{f^*(x)}$ and process control $\boldsymbol{g^*(u)}$ is carried out from the expression (10).

$$\begin{cases} \boldsymbol{f^*(x_i)} = \int_0^T (\boldsymbol{BV}_{x_i})\, \boldsymbol{dt}, \\ \boldsymbol{g^*(u_i)} = \int_0^T (\boldsymbol{BV}_{u_i})\boldsymbol{dt}. \end{cases} \quad \forall \boldsymbol{i} = \overline{1, N} \qquad (10)$$

V. CONTROL SYSTEM ALGORITHM

Algorithm for optimal functioning of the control system is shown in Figure 3, where computing elements PE_1 … PE_N are displayed, containing the same sequence of operations and operating in parallel.

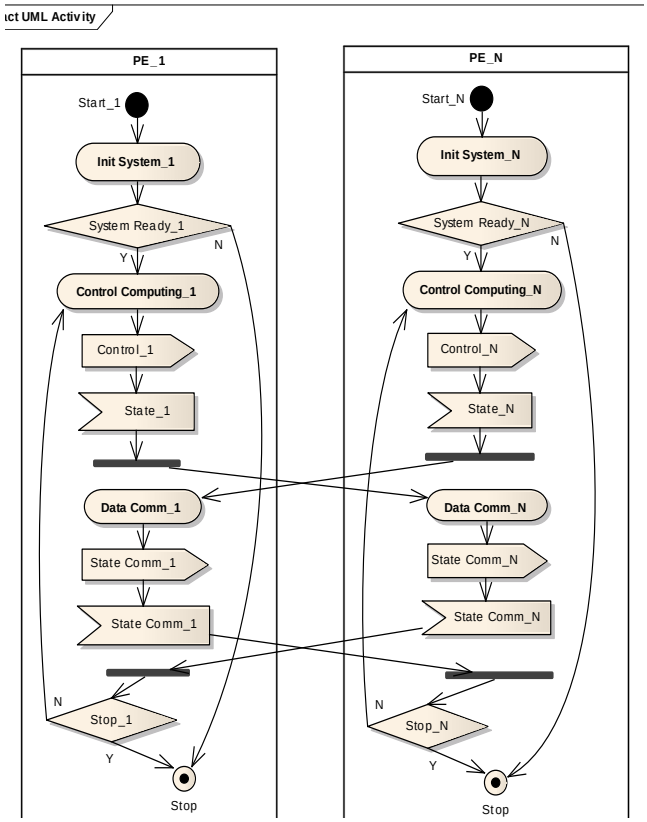

Figure 3. The algorithm of computing elements operation.

Specification of a sequence of computational operations to processing elements $PE_i, i = \overline{1, N}$:

Start – beginning of the sequence of operations; ***Init System*** – initialization processing elements; ***System Ready*** – determination of availability of computational elements, if the device is ready to work ***Y*** then continue to execute sequences of operations of the algorithm, if the device is not ready to work ***N*** then blocked and is derived from the multi-controller computing architecture; ***Control Computing*** – computation of control signals; ***Control*** – impact of control signals $DO_i, i = \overline{1, N}$ a controlled process; ***State*** – input the state $DI_i, i = \overline{1, N}$ of operation of the controlled process; ***Data Comm*** - data exchange between computing elements $\boldsymbol{BV}_x$ and $\boldsymbol{BV}_u$; ***State Comm*** – exchange of data between computing elements via an $\boldsymbol{I^2C}$ network; ***Stop*** – end of the sequence of operations.

VI. EXPERIMENTAL MULTI-CONTROLER SYSTEM SYNTHESIS

To test the functionality of the control system is proposed embodiment of the processing elements $PE_i, i = \overline{1, N}$ based on microcontrollers Atmel AVR ATmega16 [5,6].

Figure 4 shows the experimental scheme of multiprocessor architecture, which includes four microcontrollers (U1-U4) ATmega16 connected to the network through a standard $\boldsymbol{I^2C}$ signals ***SCL*** and ***SDA***. Connectors (J1-J4) are intended to enter the state of the controlled process (***ADC1***), and process control signals (***PA3***).

The principle of operation of the control system is to solve the system of equations (7) using an iterative method.

As a result of modeling a multi-functional control system (Figure 4) results that prove the correctness of the solution have been obtained for this purpose we used design environment Proteus 7.6 [7]. For more complex control systems (up to 30 MCU) functionality and precision of the results were checked by analytical methods.

VII. CONCLUSION

The optimal control problems include increased complexity and the solution of this problem based on single-processor (single-controller) computer architectures is inefficient.

This paper offers a solution to the problem of optimal control based on multi-controller architecture with a uniform distribution of computing power and minimizes the amount of information transmitted between the computing elements.

REFERENCES

[1] A. A. Agrachev, Yu. L. Sachkov, Lectures on geometric control theory, Trieste, SISSA preprint, 2001.

[2] A. A. Agrachev, Introduction to Optimal Control Theory, Trieste, SISSA preprint, 2001.

[3] Henry, John L., and David A. Patterson. “Multiprocessing and Thread level Parallelism”. Computer Architecture. ed. 3rd. Singapore: Elsevier Pte. Ltd, 2003.

[4] http://easyelectronics.ru/interface-bus-iic-i2c.html (citation 20.09.2014).

[5] http://www.atmel.com/products/microcontrollers/ (citation 20.09.2014).

[6] http://www.avr.ru (citation 20.09.2014).

[7] http://www.labcenter.com/ (citation 28.09.2014).

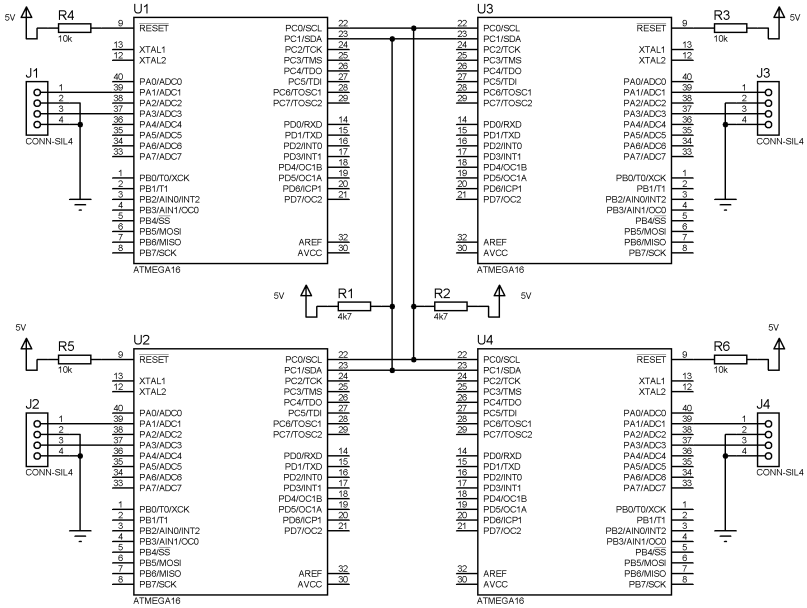

Figure 4. Experimental multi-controller system.