World Applied Sciences Journal 29 (8): 1004-1009, 2014
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2014.29.08.13966

Distributed Streaming Microcomputer Systems for Collecting and Processing Data from Dynamic Objects Sensors

Oleg Nikolaevich Pyavchenko

Southern Federal University, Taganrog, Russia

Abstract: The present work is investigating the issues of establishing high-performance streaming local information microcomputer systems intended for real-time monitoring (defining trajectory, state, control and monitoring) of complex dynamic objects. Collection and processing of sensors signals values is based on streaming four-tier processing of solution sampling intervals. With that, signals are digitized in parallel way by groups of signals, each of which is sequentially formed in a loop. To build such systems, information micro-controller modules are used that make it possible to create variety of schemes for stream-oriented sensors signals processing. Consequent and consequent-and-parallel schemes with group processing of sensors data and the scheme for parallel solving triads of tasks have been considered and assessed. Prospective viability of consequent-and-parallel schemes with group processing of sensor signals was shown when these schemes are used at inputs of fourth tier in multichannel communication devices for collecting data packets.

Key words: Streaming distributed system · Computation processes schemes · Time of collecting and processing sensors information

INTRODUCTION

Almost half a century have passed since the publication of the concept of high capacity homogeneous computing systems (HCS) developed in the Siberian Division of the USSR Science of Academy by E.V. Evreinov and Y.G. Kosarev, based on the principle of computation distribution [1, 2]. Over the past years, staff of the Siberian Division of the USSR Science of Academy, including V.G. Horoshevsky and his students [3], have developed the concept of HCS into a recognized trend in the field of computer technology, having developed not only the theory, but also having built many HCS type specimens. Generalized definition and description of features of distributed computing systems belongs to G. Lorin [4] who claims the following: "The main feature of a computing system lies in the fact that its various components can be divided into separate parts and can be arranged in different logical areas interconnected by respective protocols..."

The principle of computing distribution became wide-spread not only in computer systems, but in automation systems of technical objects and technological processes as well. "A Distributed Control System (DCS) may be defined as a system consisting of many devices distributed in space, each being independent from the others, but interacting with them for fulfilling a common task" [5]. As a typical example of one of many implementations of a distributed information systems, which are problem-oriented to meet challenges of managing complex technological processes, let us refer to a distributed system based on OWEN modules [6]. The system has a complex bus-based hierarchical structure. As in the majority of other systems, data exchange between modules and programmable logic controller (PLC) is realized by field buses on the master-slave principle. At the end of the step after processing information, PLC establishes connections and sends calculation results to output modules. This way of implementation is flexible but time consuming.

High-performance distributed information systems are required in order to automate highly dynamic objects / processes. One promising area of building high-performance automation tools is creating distributed information systems that use streaming of business solutions for Collection and Processing Sensor Data (CPSD).

Corresponding Author: Pyavchenko, Southern Federal University, Petrovskay, 81, 347900, Taganrog, Russia.

Main: Features of high-performance streaming local information-processing microcomputer systems.

The principle of streaming organization is in forming a resolution of digital signals of analogue sensors at the beginning of each step, their preservation and further processing in predefined manner without participation of the dispatcher program.

Most fully, the streaming principle of organizing CPSD task-solving is implemented in Local Information-processing architecture Micro-Computer Systems (LIMCS) intended for monitoring (defining trajectory, state, control and monitoring) of complex dynamic objects/processes [7]. Features of LIMCS are the following: streaming organization of business processes solving CPSD, availability of modules collecting and processing information in a given set of tasks; hardware and software support for CPSD problem-solving processes at the level of modules and network configuration of their connections; use of network channels for transferring information between modules; dependence of resolution step duration on maximum frequency of signals; binding generating signals values of all sensors to start of CPSD task solving step; synchronization of parallel CPSD processes.

High accuracy of LIMCS is defined by choice of physical quantities precision sensors, method of collecting values of sensor signals, primary processing and measurement methods, digital methods of computing functions, the bit number used in calculations, a slight delay of results calculation with respect to the beginning of solution step. This work discusses organization of high-speed computational schemes that can ensure minimum phase error.

The title "Systems for collection and processing sensors data" focuses on the content of ongoing processes. Processes are viewed as some action sequences that establish procedures for data collection, processing, measurement and monitoring.

In a generalized sense, the CPSD task solution scheme is shown in a tiered form where the following designations are introduced: Z_{jd} - tasks d = 1,2,3, ..., D that can be solved independently of each other at each tier $_j$ ($_j$ = 1,2,3) (number of tasks corresponds to the number of sensors D); IO_j - data object generated at j tier;T_{zj} - time of task solution at tier j; Z_4 - tasks of the fourth (top) tier.

At the lower, first tier, tasks $Z_1 = \{Z_{1d}\}$ for analog processing of sensors signals and analog-to-digital conversion (ADC) are solved. The higher, second tier is assigned tasks $Z_2 = \{Z_{2d}\}$ of Primary Digital Processing (PDP) of digital values of D sensors signals formed at the 1st tier. Located on the third tier are tasks of first level of secondary digital processing (1 S_{DP}) of group $Z_3 = \{Z_{3d}\}$, which make up tasks of measuring signal values in physical units, defining state of physical quantities and their assessment, etc. At the top, fourth tier, tasks of second level of secondary digital processing (2 S_{DP}) are resolved - Z_4:

- Concentration of input data, including sorting and storage of results of first level of secondary digital processing, etc.;
- Computing trajectories of state change, evaluation of current and forecast states of the local object, variables of signals of which are processed in LIMCS;
- Formation of an information object and sending it to a higher level of an informational microcomputer system.

The adopted as a basis tier scheme is an ordered decomposition of tasks solution by priority levels of their implementation, from tasks on the lower Z_1 (junior) level to tasks to the top Z_4 (senior) level. Aforementioned tasks Z_1-Z_4 are characterized by different objective functions, mathematical methods, increasing from bottom to top computational complexity, completeness consisting in full realization of a corresponding objective function.

Problem orientation (objective function) is reflected in CPSD architecture and content of processing tasks. Organization of analog sensors information collection considerably impacts LIMCS characteristics [8]. In practice, cyclic organization is widely used, whereby in each cycle according to readings of a counter of sensor address (number) STd = d, analog switch (multiplexer) selects the sensor signal d for time T_{DS}. After analog processing signal is digitized during T_{AD} and obtained data is stored. Then address of sensor is changed as STd = CTd+ 1 and the next signal value (d +1) of sensor is formed.

Start of polling and conversion of first sensor d = 1 is linked to moment ti that identifies the beginning of task solving in CPSD. Collection procedure is completed after formation of signal d = D.

At point ti, only value of the first analog signal is determined. Values of the remaining signals are generated with time shift at points

$$t_{id} = t_i + (T_{DS} + T_{AS}) (d - 1) \ (d = \overline{1, D}) \quad (1)$$

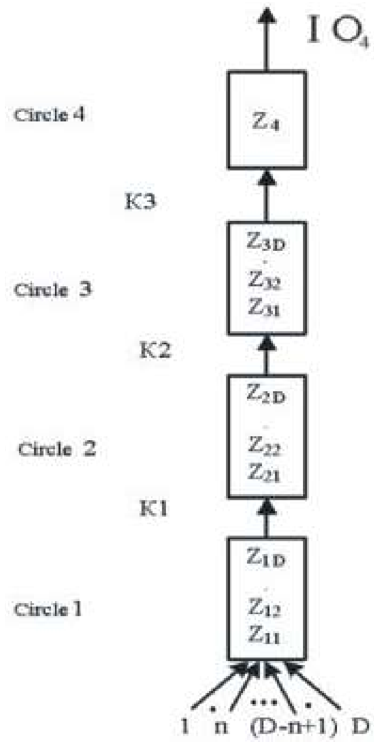

Fig. 1: Sequential scheme S1

Only the value of the first sensor signal is formed without time offset errors. Signal values of other sensors contain collection errors. The later the sensor is polled, the greater is the error. For example, collection error may become unacceptable about 0.345% already for d = 2 with increasing signal frequency f_{dmax} up to 5 Hz and higher. In such cases, collection error of 0.01% is ensured during independent digitization of each sensor signal. In such cases necessary accuracy of collection is ensured by parallel digitizing of sensor signals. However, this method of signal digitizing organization is costly as it requires implementation of D ADC and memory sections. Interesting is the use of group method cyclic for digitizing sensors signals. In this case, D sensors of the object being monitored are divided into G groups, n sensors in each, n = D/G. Each group of signals is cyclically processed by own ADC. At the beginning of each step, solutions in each group are generated in parallel and saved are values n of signals that are sent for processing in order of priority.

Basic components of CPSD systems are functionally and structurally completed Informational Microprocessor Modules (IMM) integrated into information and control relationships, as well as a real-time operating system. Basic IMM models are determined by their specialization for tasks assigned to different tiers and for solving admissible sets of tasks belonging to different tiers.

Time id task solving in CPSD largely depends on LIMCS scheme of processing organization, including timing of data transfer between tiers. Maximum time is required for solving CPSD tasks in organization of group collection and stepwise processing of signals D>>1 in sequential [9] distributed LIMCS (Fig. 1).

Sending signals to AD converters starts with the first signals in groups at the beginning of ti resolving step. Remaining signals are sent according to the counter of group sensors. From output of each G AD converter, digital value of the signal is sent for storage in appropriate G section buffer memory. After recording values of D signals of sensor they undergo PDP at tier 2. Results of task solutions in the second tier are used at the third tier, where values of signals are defined in units of physical quantities and of their status. From the third level, generated signals values of variables and their states are transferred to the fourth tier. At the input to this tier they are stored until last piece of data is obtained. After that, Z4 tasks are solved.

The main disadvantage of sequential scheme of CPSD process is its low performance. Computation time according to S1 schema is

$$T_{S1} = \sum_{j=1}^{3}\sum_{d=1}^{D} T_{Zjd} + T_{Z4} + \sum_{j=1}^{3} T_{Kj}, \qquad (2)$$

whereas TK_j is duration of transmitting information from j-th to (j +1)-th tier.

Natural step in improving LIMCS performance is the organization of multisequencing [10] of batch processing of sensor signals not only at the first tier, but at the second and third tiers. In a series-and-parallel scheme S2 implemented are G = D/n branches, in each of which information from n sensors is processed. At each j tier tasks in g (g = 1,2,…,G) group Z_{jd} (d = [(g – 1)n+1, (g–1) n+2,…,gn]) are solved consequently. Solution results are passed from tier j to tier (j+1) in G packets. Tach packet of tiers 1, 2 contains minimum N_j = n (j=1,2) data, of tier 3 - N3 ≥ 2n. Parallel-and consequent scheme with the collection and processing of the G groups of sensors is shown in Figure 2. Feature of scheme topology is presence of G channels for collecting and preliminary processing of sensor groups signals, as well as a network switch K (G) [11] that provides collection of solution results from the third tier at input to the fourth tier Z_4.

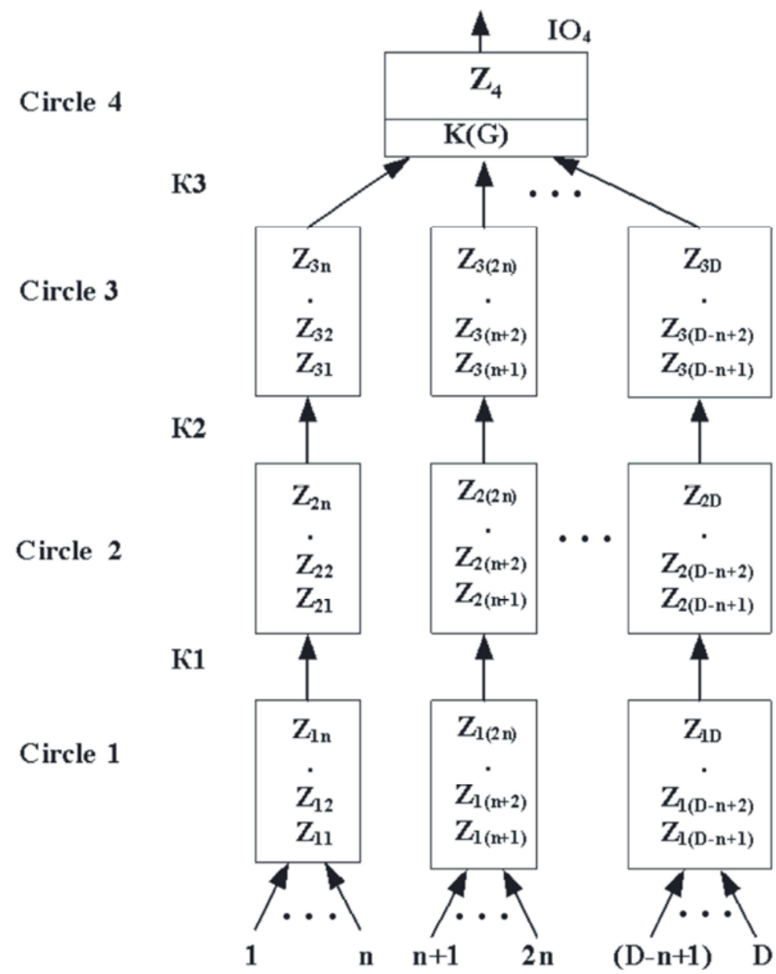

Fig. 2: Scheme with group collection and processing sensor signals

Time of solving CPSD tasks on a consequent-and-parallel scheme is approximately assessed by expression,

$$T_{S2} = \sum_{j=1}^{3}\sum_{d=1}^{n} T_{Zjd} + T_{Z4} + \sum_{j=1}^{3} T_{Kj} + T_K(G), \tag{3}$$

whereas TK(G) is time of forming of data packet for solving tasks Z4 in communicator K(G). Relation TS2 < TS1 is valid if

$$\frac{T_K(G)}{\sum_{j=1}^{3}\sum_{d=n+1}^{D} T_{Zjd}} < 1. \tag{4}$$

Inequality (4) holds when a network communicator K (G) for receiving and storing information from the G channels uses fractional memory and a simple scheme of integral message composition that does not need time consumption. At the same time, the condition (4) may be valid also when from G packets Q packets are formed (Q <G) [9].

Highest performance LIMCS are built during implementation of scheme S3 with parallelization of triads <Z1d, Z2d, Z3d> (Fig. 3). $(d = \overline{1,D})$

In the scheme, each sensor signal is converted and processed in its own channel. Results of processing are sent via communication channel $K3_d$ to communicator K(D) located at the input of tier four. After obtaining all data, tasks Z_4 are solved. $(d \in \overline{1,D})$

Solving CPSD tasks according to scheme S3 takes time,

$$T_{S3} = \sum_{j=1}^{3} T_{Zj1} + T_{Z4} + T_{K3} + T_K(G) \tag{5}$$

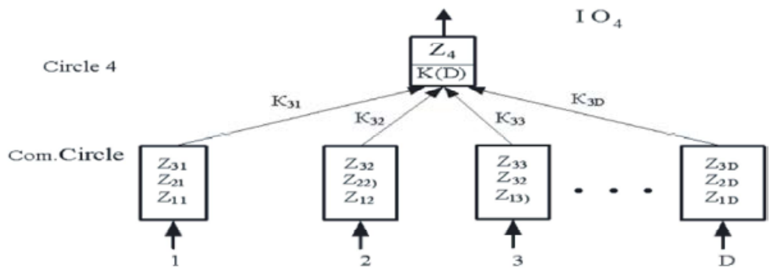

Fig. 3: Ungrouped parallel-and-consequent scheme S3

whereas $\sum_{j=1}^{3} T_{Zj1}$ is time for solving tasks of triad <Z11, Z12, Z13> taken as equal for all D triads, T_{K3} is time for sending results of signal processing in course of solving tasks of triad <Z1d, Z2d, Z3d> from tier 3 to tier 4, TK(D) is time of forming in communicator K(D) of data packet for solving tasks Z4. $(d.=\overline{1,D})$

Let's define a condition when

$T_{S3} < T_{S2}$.

After substituting (4) and (6) in this inequality and not complex transformations, we shall obtain that the scheme S3 (Fig. 3) is more productive than S2 (Fig. 3) when

$$\frac{T_K(D) - T_K(G)}{\sum_{j=1}^{3}\sum_{d=1}^{n} T_{Zjd} - \sum_{j=1}^{3} T_{Zj1} + \sum_{j=1}^{2} T_{Kj}} < 1. \quad (6)$$

Condition (6) is satisfied if in communicators K(D), K(G) integral messages are arranged according to simple algorithms. Indeed, in such cases, the difference between values TK(D) and TK(G) is small and can be divided into a number greater than unity. Condition (6) is also satisfied when linkers of communicators K (D), K (G) use hardly discernible sophisticated algorithms, difference value [TK (D) - TK(G)] of which is within the margin of error.

Opinion: To ensure high performance in architecture of distributed systems that are intended to collect and process sensor information, the following features should be implemented:

- Signals digitizing should be organized in accordance with their maximum change rate and permissible magnitude of phase error;
- Streaming principle of collection and preprocessing of groups of sensor signals should be implemented;
- Parallel topologies of start type with the group collecting and processing sensor signals should be implemented;
- High-speed packet signal processing results communicators should be used.

CONCLUSIONS

Currently, task solution automation makes use of distributed systems based on principles of hierarchical bus network configurations, organization of functioning on master-slave principle, etc. Automation for more dynamic objects should make use of parallel LIMCS streaming ensuring high performance and accuracy of solving CPSD tasks. High-performance streaming CPSD LIMCS are based on consequent-and-parallel processes schemes with group collection and processing of sensor signals with introduction of high-speed network communicators.

REFERENCES

1. Evreinov, E.V. and Y.G. Kosarev, 1962. About Possibility of Building High-Performance Computation Systems. Novosibirsk: Siberian Department of Academy of Sciences of the USSR, pp: 40.
2. Evreinov, E.V. and Y.G. Kosarev, 1966. Homogeneous Universal High-Performance Computation System. Novosibirsk: Nauka Publishers, pp: 308.
3. Khoroshevsky, V.G., 2008. Architecture of Computational Systems: Educational Textbook, 2[nd] Ed., Revised and Amended. Moscow: Publishing House of MGKU n.a. N.E. Bauman, pp: 520.
4. Lorin, G., 1984. Distributed Computing Systems: Translation from English. Moscow: Radio and Communications, pp: 296.

5. Bonastre, A., J.V. Capella and R. Ors, 2002. A New Generic Architecture for the Implementation of Intelligent and Distributed Control Systems. In the Proceedings of IECON 02 [IEEE 2002 28th Annual Conference of the Industrial Electronics Society], 3: 1790-1795.
6. Distributed System Based on OWEN Modules. Date Views 01.10.2013 www.owen.ru/text/28868129.
7. Pyavchenko, O.N., 2011. Distributed Systems for Collecting and Processing Data from Dynamic Objects Sensors. SFedU Bulletin: Engineering Sciences. Taganrog: Publishing House of SFedU TTI, 5(118): 9-15.
8. Park, J. and S. Mackay, 2003. Practical Data Acquisition for Instrumentation and Control Systems. Newnes, pp: 432.
9. Bagad, V.S., 2010. Computer Networks. Technical Publications, pp: 516.
10. Kurose, J.F. and K.W. Ross, 2010. Computer Networking: A Top-Down Approach. Pearson Education, Limited, pp: 888.
11. Pyavchenko, O.N., 2013. Communication Modules of High Performance Distributed Information Microcomputer Systems. SFedU bulletin: Engineering Sciences. Taganrog: Publishing House of SFedU TTI, 5(142): 9-14.