We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



11

A Computer-Aided Control and Diagnosis of Distributed Drive Systems

Jerzy Świder and Mariusz Hetmańczyk
The Silesian University of Technology, Poland

1. Introduction

Modern systems of the industrial processes automation most often are composed of several basic components, among others: controllers (PLCs, regulators, industrial computers, etc.), actuators (electric, pneumatic, hydraulic and their combinations), control and measuring equipment (sensors and detectors), industrial networks and SCADA/HMI systems (Świder & Hetmańczyk, 2009c). Functional subassemblies have been divided into several groups (mechanical elements, electrical and electronic parts, control equipment and software packets). All of the enumerated components are susceptible to failures, therefore, the main problem is an estimation of the trouble-free service period. In the case of individual components the described task is easy, but an analysis of distributed applications defined as a collection of dependent subsystems the reliability function is dependent on all reliability factors assigned to all components (Dhillon, 2002).

Sequential distributed systems include most often: automated machining centres, assembly lines, conveyors etc. Specified examples (for safety reasons) involve extended monitoring procedures strictly connected with requirements of the employed maintenance model.

The preventive maintenance model includes several meaningful assumptions, among other things: risk reduction, failure avoidance, the continuous improving of the reliability of the considered system, defects elimination (with attention focused on minimization of the time needed for the detection of error sources and their elimination), maximization of a productivity combined with decreasing of the total production costs, with continuous improvement of the quality level. An elaboration of the method that fulfils all the specified requirements with enumerated factors will be the most important task for engineers derived from different domains. A utilization of more advanced devices allows for a combination of diagnostic and control functions at various levels. Furthermore, in the most of contemporary applications, an integration was omitted for many reasons (economic premises, time and work efforts, the complexity of the algorithm etc.).

2. Problems connected with the maintenance of distributed systems

The maintenance of distributed systems is a very important task in many engineering applications (Ryabinin, 1976). Industrial control systems are described by several qualities:

control system organization (centralized or distributed), application size, type of measuring equipment (sensors and detectors), technical orientation, etc.

Industrial productive systems run at a risk of impact by many external factors. Among meaningful impacts occur two main groups in the form of: command signals (purposefully dispatched by users) and disturbances. Progress at a complexity of manufacturing departments, productive systems, as well as machines, causes a continuous increase in the number of failure sources (De Silva, 2005). Nowadays the diagnosis of automated industrial productive systems is a significant issue.

Design, implementation and operation of machines without taking into consideration aspects related to the monitoring of operating conditions and planning of the duty are not possible in applications taking advantage of mechatronics devices.

The described problem is especially apparent in the case of the monitoring and control of the states of electric drives equipped with AC asynchronous motors that are subjected to extensive investigation in many research institutes. However, tests on a single drive treated as an autonomous machine constitute a predominating tendency nowadays.

Such an approach does not meet the needs of operational and maintenance engineers, as industrial conditions compel the requirements of complying with several rules, including: fast detection and removal of failures, minimization of the number of configuration procedures run on the diagnostic equipment (a universality securing the operation of numerous communication protocols and methods of access to the diagnosed object) as well as a wide range of available tools (Bloch & Geitner, 2005).

Up to this moment, devising a concept of diagnosing distributed drive systems operated by frequency converters coupled with the ProfiBus industrial network (Mackay & Wright, 2004) and a centralized PLC control has not been developed.

2.1 A structural analysis of the electric drive unit

Electric motors dominate in industrial drive systems of modern machines. The reasons for such wide spread popularity are their functional features such as: high efficiency of the individual drive unit, wide ranges of torque and speed adjustments, small dimensions, ease of a power supply, good prices regarding functional features.

The availability of a wide selection of the components belonging to the group of the automatic control equipment has made it possible to construct new technical devices that have not previously corresponded to the traditional definition of a machine (Fig. 1a). The determination of the state of the machine on the grounds of control, energy, disturbances and mass fluxes does not consider many factors to ensure a proper course of work of distributed units. The problems of control and identification of failures of units that have a different degree of technological advancement are still important tasks to be addressed.

Automation involves the necessity of providing man-free operation of the duty, for which a two-directional information flow is required (enabling both control of the parameters and change in the states).

The specification of the components of the machine block (Fig. 1b) enables a complex identification of the internal information fluxes (signals exchanged at the level of individual

control of drives by means of frequency inverters) and external ones (a composite flux, corresponding to the operational state of the machine as a whole).

Each drive subsystem has features that predispose it to a group of autonomous machines. In each subsystem were extracted an electric drive, extended by a reduction gear and a control subsystem (in the form of a frequency inverter or an I/O module). Detailed relationships between functional subsystems of the drive system are shown in Figure 1c.

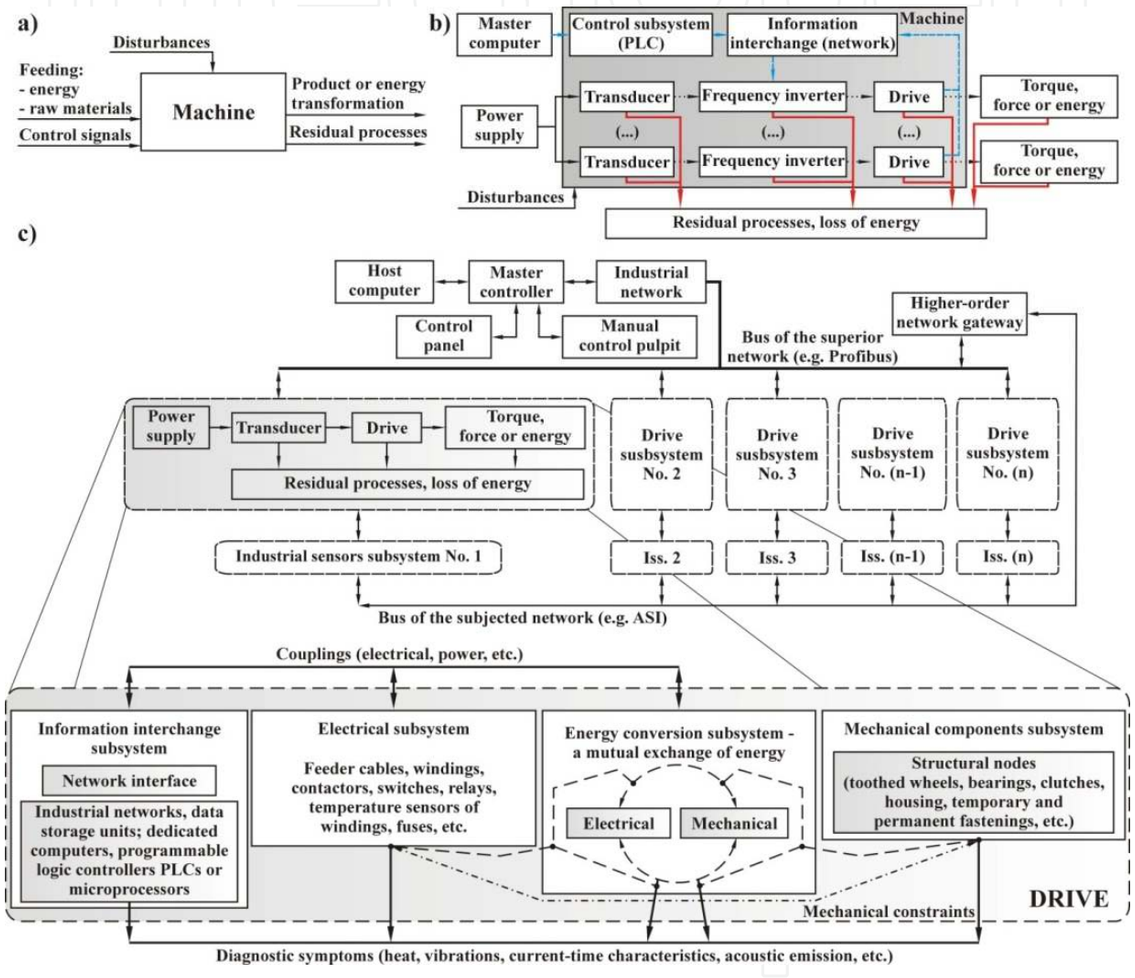


Fig. 1. Definitions of the machine: a) a traditional approach (Korbicz et al., 2002), b) the approach adjusted to the requirements of the analysed systems, c) simplified view of the drive subsystem in the form of coupled components collection

The proposed structure refers to, among others, conveyors, production lines treated as a one cell, in which the proper course of the work cycle depends on all the components.

Distributed systems with the appropriately formed information flux may be subjected to an analysis in view of the system parameters at different levels (from the values recorded by sensors, through the state of the control elements, to production data).

2.2 An overview of the most often used approaches for the monitoring of distributed systems

The supervision (Boyer, 2004) over industrial processes and the change of the parameters of the states of distributed systems are most frequently performed by three means (Fig. 2).

Portable analysers (Fig. 2a) make it possible to make measurements of the operational parameters of machines at a work place, yet, in view of the nature of their construction (specialized units enabling the determination of parameters, mostly for the one type of phenomenon), the measurements are conducted only at planned time intervals or in the case of visible symptoms of damage. An additional disadvantage of such an approach is the need to provide the service using experienced diagnostics staff.

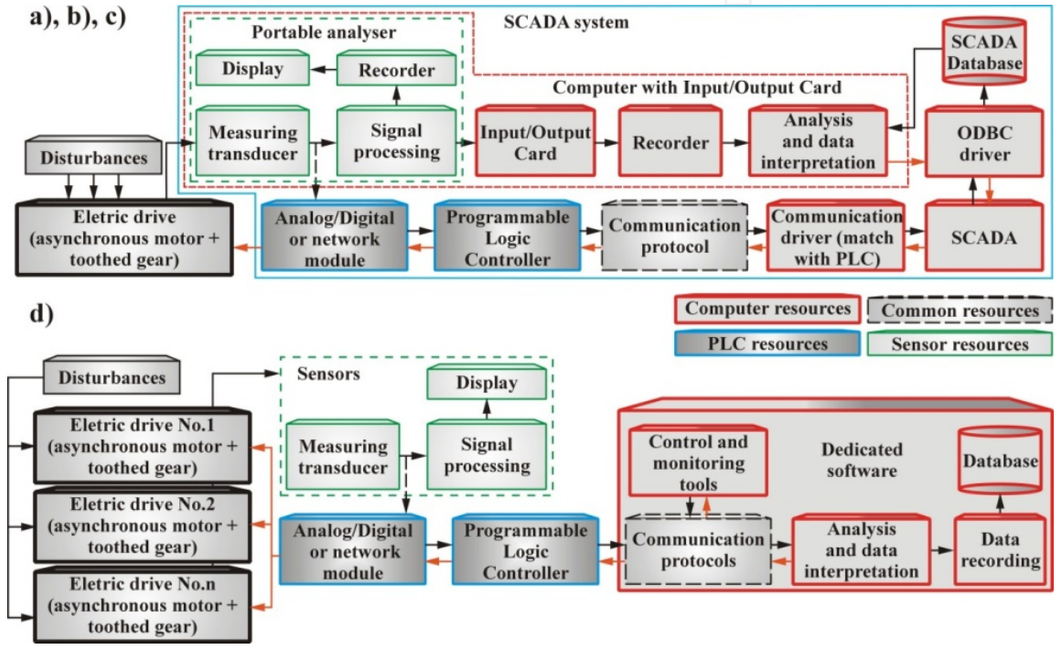


Fig. 2. Methods of controlling the states of technical systems: a) analysis of the machine under operation made by the analyser, b) use of the input/output card, c) implementation of SCADA system and data collector, d) a concept of the customized computer environment

Computer input/output cards (Fig. 2b) are used in systems with a small number of measuring or executive devices. SCADA systems (Fig. 2c) do not facilitate access to special functions of the supervised equipment and are, at the same time, relatively expensive (Boyer, 2004). Furthermore, their satisfactory recording resolution requires the configuration of an optional collector. The remaining two means have many limitations and are most frequently used as elements supporting exact diagnostic activities.

The last of the concepts (Fig. 3d) is based on the application of customized software, which combines benefits of presented supervisory and control systems, expanding simultaneously opportunities for additional features such as: dedicated diagnostic and control blocks, utilization of the ProfiBus main line as a carrier of the diagnostic information, advanced features of the ProfiBus protocol. The main advantage of the proposed concept is the ability of the implementation in any process (or machine) operated by the PLC controller.

3. Integration of diagnostic and control functions

The developed method is characterized by some degree of universality (all applications of control and diagnostic activities of electric drives supported by PLC controllers), but at the design stage the authors have adopted significant limitations related to the organization of hardware and software structures (Kwaśniewski, 1999).

Usage of programmable logic controllers and distributed drives equipped in industrial network interfaces is a common case in industrial applications. PLC's popularity results from a fact that exchanged data most often have a status of control commands (Legierski, 1998).

In the majority of cases diagnostic and control functions are separated. The traditional structure of an organization of control algorithms is oriented to the failure-free execution of the course of work, but the diagnostic information is most often omitted. The main cause of this situation arises from a substantial complexity of the control algorithm extended by diagnostic functions. Integrated diagnostic and control functions compound specified functional qualities, which combine presented domains.

A set of diagnostic and control functions creates a program for realizing self-diagnostic functions of the controlled system. A fault detection function should be prepared for every unit (Świder & Hetmańczyk, 2010). A superior level function is performed by the computer system, which consists of dedicated screens that allow interpretation of current and historical data. To primary parameters belonged, among other things current, minimum and maximum bus cycles, diagnostic information, a current work status, error codes, reactive and active powers, current status of I/Os, etc.

3.1 The model of the information flow

The proposed solution is intended for the real time analysis in the case of industrial applications that use automatically controlled equipment and are mainly operated by means of customized controllers with properly selected modules and communication standards.

A preventive maintenance of programmable logic controllers contains several simple actions. On the basis of this statement the PLC unit could be assigned to the group of the most reliable elements of automated productive systems (Prasad, 2004).

The reliability of constituent elements and a whole system depends on a fast assessment of operating conditions and prediction of failures on the grounds of the current state.

So that execution of detailed results could be undertaken, research objects were restricted to Mitsubishi PLC controllers. A functional extension can be obtained on most PLCs available in engineering applications.

In the diagram of a distributed system shown in Fig. 3a, the system has been divided into particular functional subsystems. The PLC holds a superior function as it enables the execution of the control parameters for all drives (Świder & Hetmańczyk, 2009d).

The basic assumption of the decentralized system, in the form of distributed actuators and centrally located PLC (or separated control systems connected by the communication main line, facilitating access to the data in the entire system) leads to the conclusion that the

system components are connected hierarchically. An elaborated method in the form of Computer Control and Diagnostic System of Distributed Drives (CCaDSoDD) enables the control of the states of the PLCs, actuators and industrial sensors without interferences into the operation of the distributed system, and a simultaneous running of other software connected with the actuators or control units (Świder & Hetmańczyk, 2009a). The structure of the described system, based on connection relationships, conforms to an automation hierarchy (Fig. 3):

- electric drives and frequency inverters are placed in the lower level (a primary technology level),
- a Programmable Logic Controller performs the role of a supervised unit (within the algorithm defined in the PLC memory) and is placed at the group control level (Świder & Hetmańczyk, 2009b),
- the elaborated expert system is located on the highest level (the PLC and the ProfiBus DP interface mediate at the stage of data exchange between frequency inverters and the expert system),
- work of the CCaDSoDD does not interfere with an operation of the PLC controller, but it is possible to enforce certain conditions of drives (performance of active diagnostic tests).

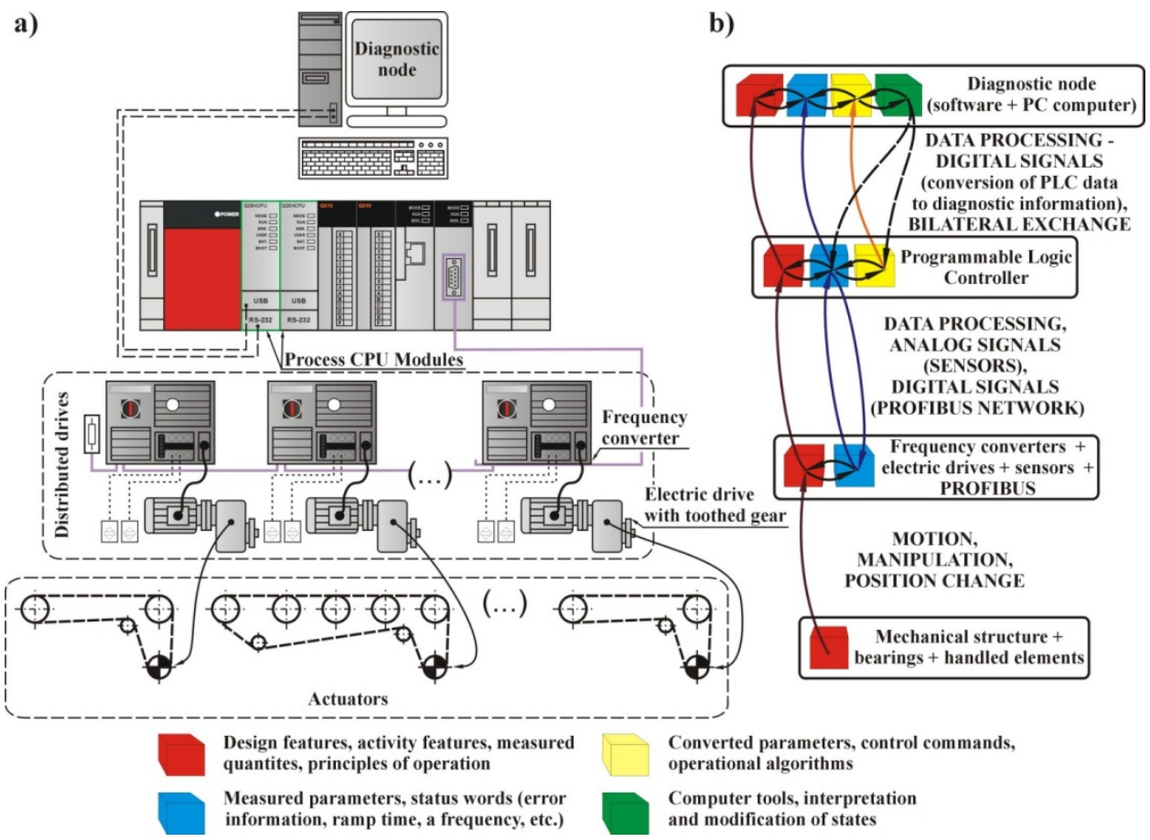


Fig. 3. Schematic diagrams of: a) the considered system, b) a data flow

Despite a precise division, a problem with an inspection of distributed mechatronic systems still exists. The maintenance of big systems requires utilization of advanced tools.

The proposed approach is based on making direct or indirect connections by means of a coupled network. The topology of the connection depends on the type of the communication network.

The choice of Mitsubishi controllers has been made in view of the ease of connection with selected logical units at each level of the defined hierarchy of the system (Fig. 4).

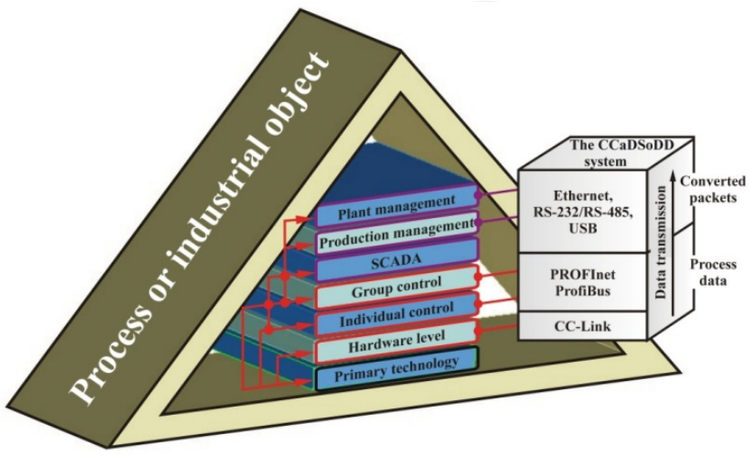


Fig. 4. A schematic diagram of dependences of the considered system taking into account the automation hierarchy

The acquisition of data from remote units is possible by means of the access via gate function (for example, the HMI panel - Fig. 5a).

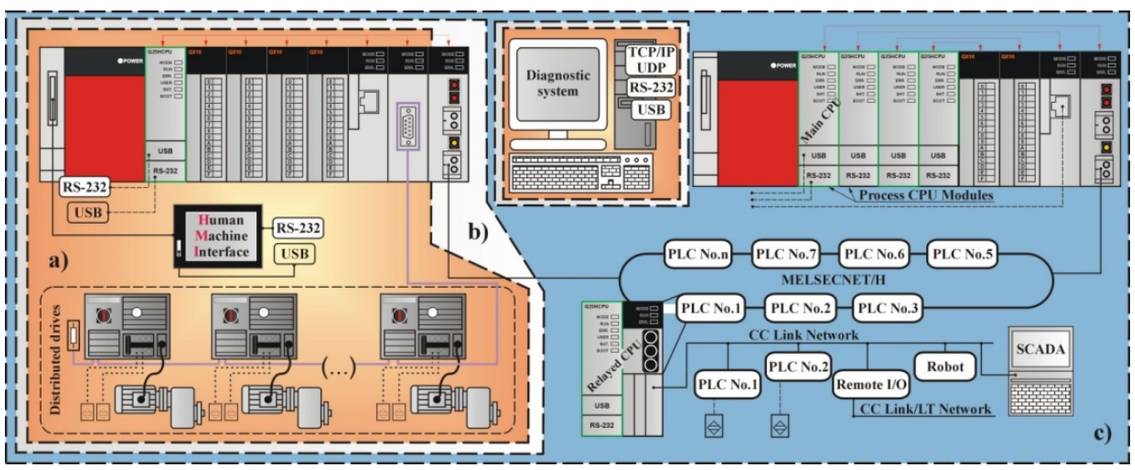


Fig. 5. Manners of access to the discussed distributed system: a) 1:1 connection (direct or by means of the operator's panel), b) and c) by means of 1:n connection

The software provided by producers must be connected directly (Fig. 6) to the drive section (frequency converter) by means of a network coupling the entire manufacturing company (the Industrial Ethernet is very popular). It is also possible to use the communication socket and the USB port of a PC, however this is still 1:1 type connection. Moreover, the contents of the data file are limited to control signals of the electric motor.

The classification of industrial networks in view of their possible applications and the configuration of the system in accordance with the assumed hierarchy of a control enables access from any level (Fig. 6). Such a solution offers many advantages, as it facilitates the configuration of the connection in a coupled or in a direct mode using any unit extended by a selected network module.

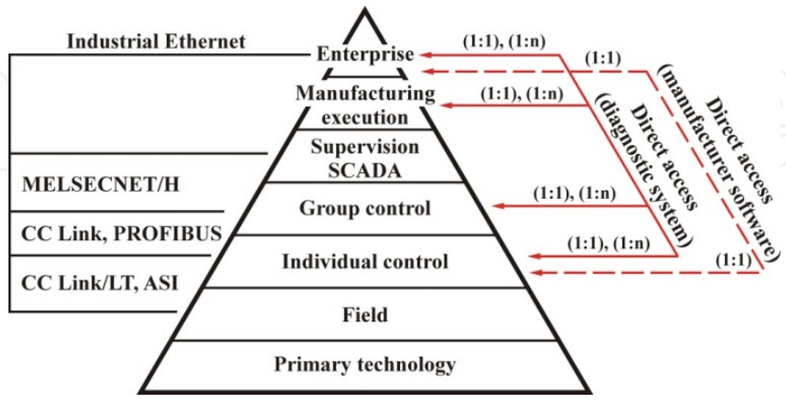


Fig. 6. The pyramid of access methods to the resources of control units

Knowing the address or the relations of the connections among the system elements, it is possible to utilize a direct access to the resource (a coupled network), both for the ascending graph trend (from lower to higher level networks) and for the descending one.

3.2 Integrated control and diagnostic blocks

In the discussed case, the communication and control of the operational parameters of each drive are performed by means of a control algorithm, saved in a PLC memory (the sequence of operating conditions). Correct course of work results from a faultless control algorithm definition. Elimination of errors leads to acquiring a faultless control algorithm, but is rather time consuming. A better approach is elimination of roots of problems and that method has been used in elaboration of integrated control and diagnostic blocks. An essential drawback of industrial networks is difficulty in direct access to transmitted data.

Integrated control and diagnostic blocks have been divided, at an angle of possibilities of applications, into several groups (Fig. 7):

- primary control blocks (for: the Movimot frequency inverter, the Movifit frequency inverter, distributed I/O modules),
- universal diagnostic blocks (of: the SLAVE unit, an interpretation of diagnostic information, the interpretation of extended diagnostic information, the interpretation of alarm information, a distribution of control data).

Basic blocks are necessary for correct system operation and are required (in case of the elementary configuration) to proper realization of the control process, however, maintenance of diagnostic functions is unfeasible. Achievement of the complete functionality requires utilization of optional blocks. Error elimination, based on internal algorithms, permits a programmer to abandon of the PLC memory allocation and algorithm testing stages (Świder & Hetmańczyk, 2010, 2009c).

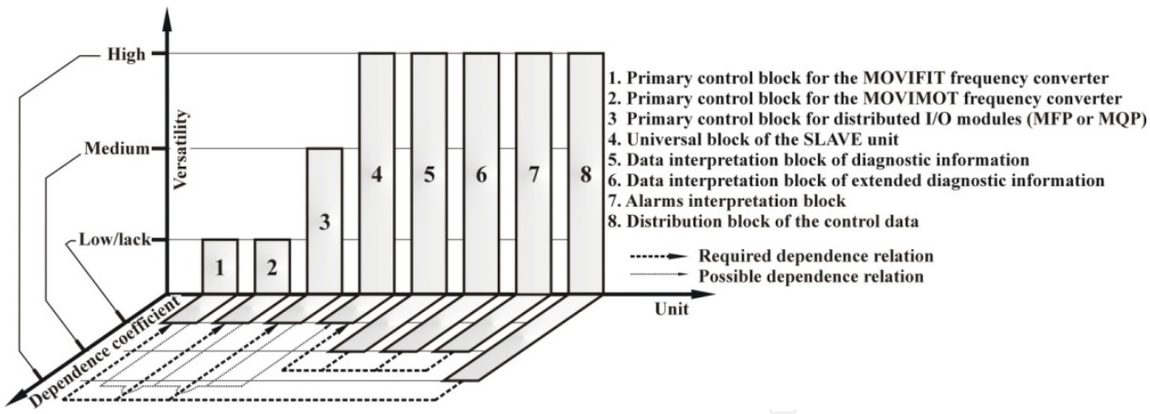


Fig. 7. A schematic representation of relationships between types of blocks, degrees of universality and the signal flow dependency

Moreover the algorithm structure for each SLAVE unit is unequivocally defined, without possibilities of mistakes. Each presented block has different tasks in the control algorithm.

An elaboration of integrated control and diagnostic blocks has been based on characteristic features of Programmable Logic Controllers. Principal assumptions of the presented method can be characterized in several points:

- each of the SLAVE unit is identified by an unique ordinal number (dependent on the ProfiBus network address),
- for proper system operation, the configuration of minimum three primary control blocks is necessary,
- additional functions (extended diagnostic functions, alarm control, etc.) demand an extension at the level of diagnostic blocks,
- exists the possibility of programming without the knowledge about a network configuration and a system structure,
- presented universal blocks have been characterized by general-purpose internal structure (implementation to control functions of all SLAVE devices with ProfiBus interfaces),
- control blocks have been elaborated at an angle of individual units.

The presented scheme of the signal flow and blocks arrangement is universal for all units (not only for the presented devices). Accommodation for an arbitrary configuration requires an implementation of the control block dedicated to the concrete unit.

Up to now, the method of traditional programming with algorithms acquiring the faultless data exchange between PLCs networks modules and SLAVE units requires:

- a continuous inspection of addresses defined in options of a network module and addresses assigned by a programmer,
- multiple testing connected with the task of a proper memory allocation,
- a definition of the amount of input and output words of each SLAVE unit (most often on the basis of manuals),
- realization of time consuming additional operations (in the case of requirements concerning diagnostic information).

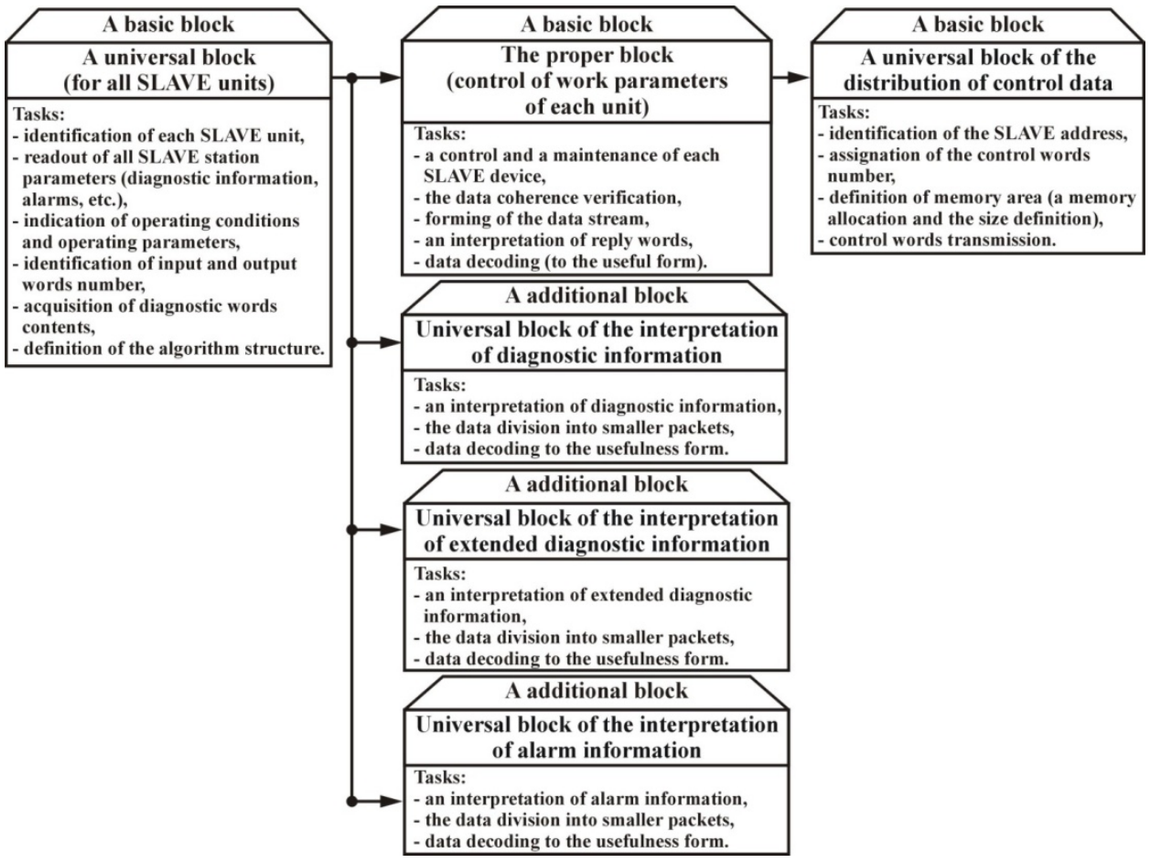


Fig. 8. A scheme of the signal flow and the blocks configuration for the single Slave unit

The proposed solution is based on the memory stack (Fig. 9) which is the primary principle of operation of all devices that convert digital data. The main problem arises from difficulties with an identification of registers which store corresponding data.

The first step is checking of statistical data of a concerned system, at an angle of number of devices, number of input (control) words and output words (diagnostic words). On the grounds of those information a programmer is able to identify the type of the SLAVE unit, without a knowledge of the functional structure of a distributed system. All described functions are performed by the universal block (Fig. 8).

The main difference between a traditional programming method and the proposed solution might be defined as a direct data exchange between the PLC processor and a special network module.

An internal algorithm of the universal block is presented in the Figure 11. The described program block belongs to the group of main elements of the control algorithm. Basic features of this block are presented in Figure 8, but the most important task is the dispatching of reply words from the SLAVE unit. A contrary task characterizes a distribution block of the control data. The block located at the centre (Fig. 8) is dependent on two described blocks and must be matched to the individual features of each SLAVE unit.

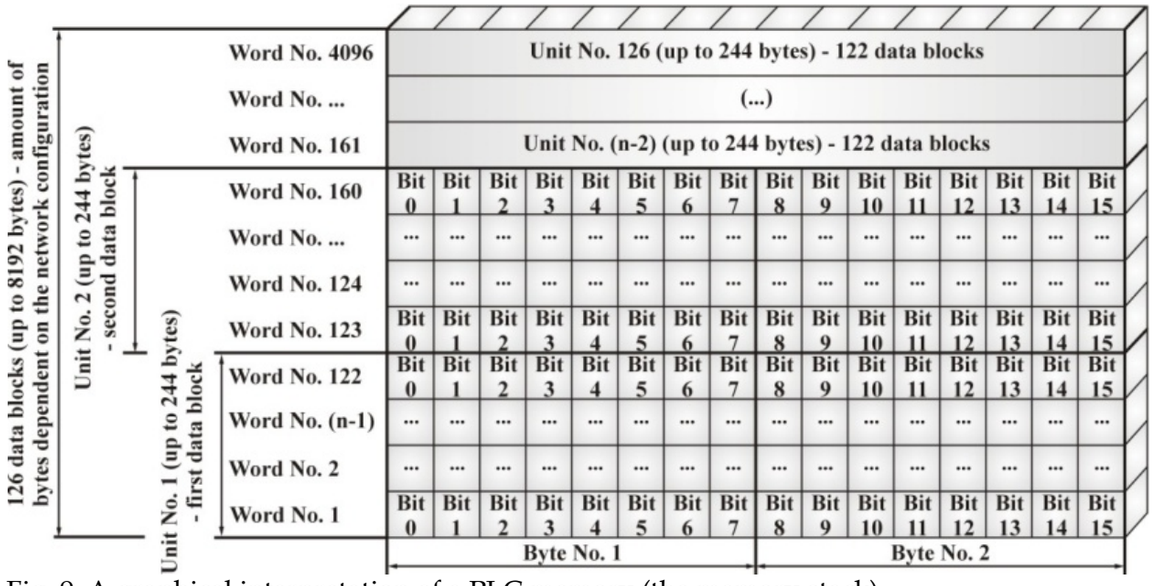


Fig. 9. A graphical interpretation of a PLC memory (the memory stack)

An internal algorithm formulates contents of control words dispatched to a block of distribution of the control data. Due to required organization of each control and diagnostic blocks, the set must be arranged in accordance with the presented scheme.

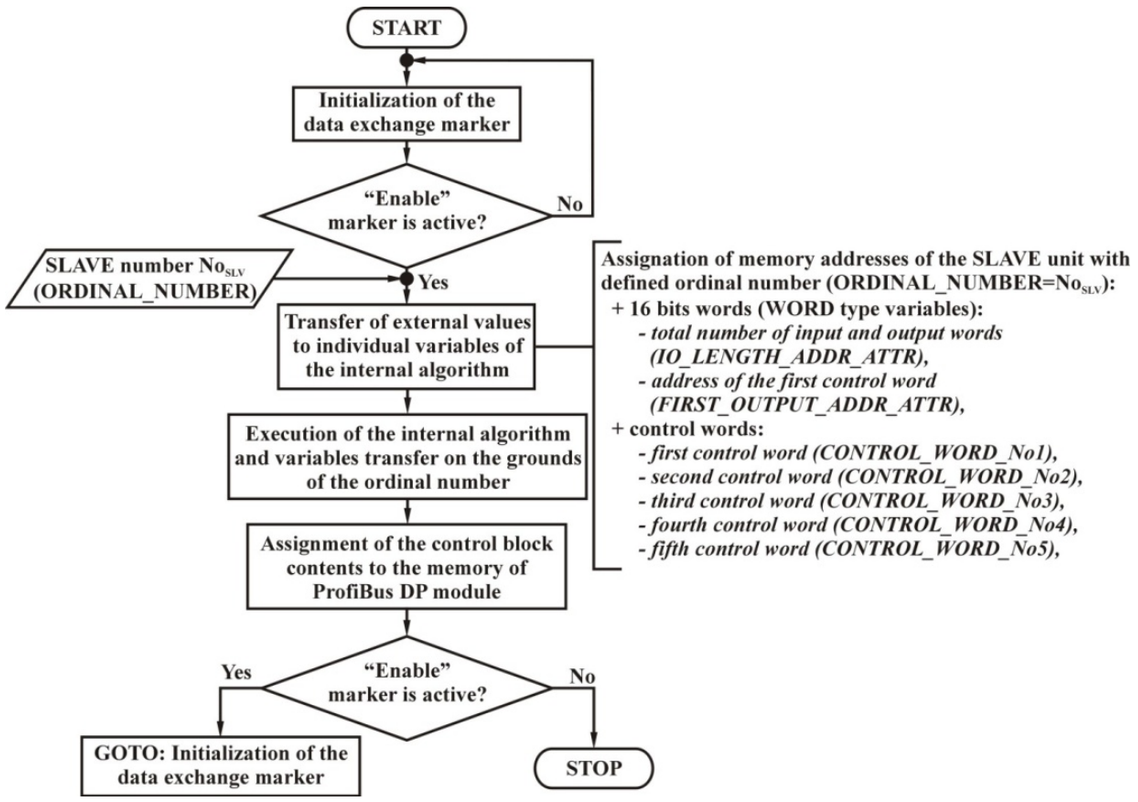


Fig. 10. A block diagram of the block of a distribution of the control data

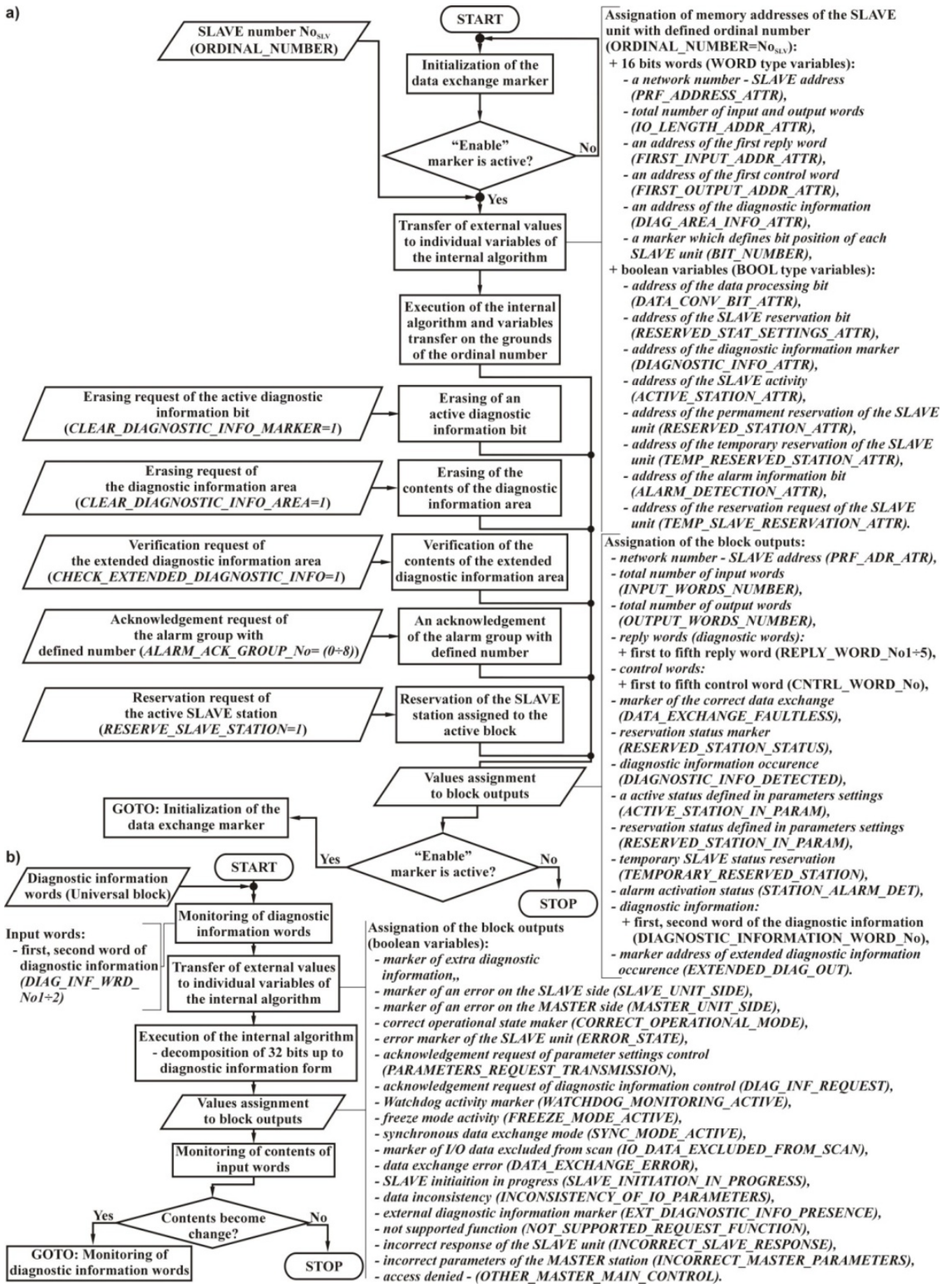


Fig. 11. Block diagrams of: a) the main universal block, b) the data interpretation block of the diagnostic information

The principle of an operation of other blocks is similar to the data interpretation block of diagnostic information and its description has been neglected.

4. Embedded functions of identification and interpretation of errors

4.1 The binary states vector method – a manual definition of failure states

A common malfunction of systems with implemented industrial networks is a loss of the communication. In consequence, often parameter reading of the inverters (or I/O modules) becomes impossible, resulting mostly from improper configuration, electromagnetic interferences, interruptions or damages in bus structures, etc.

The CCaDSoDD system allows a definition of the binary diagnostic vector (Świder & Hetmańczyk, 2011). The vector is defined on the basis of damage or failure symptoms indicated by external elements of control devices (status diodes placed on front plates).

In the case of integrated frequency inverter of the MOVIFIT unit, contents of the state's vector are defined according to the following formula:

$$W_i=[\ S_t\ \ T_r\ \ K_{zi}\ \ K_{zl}\ \ K_c\ \ P_r\ \ S_{rp}\ \ P_{dk}\ \ P_{zp}\ \ K_r2x\ \ K_r3x\ \ K_r4x\ \ K_r5x\ \ K_r6x\ ] \quad (1)$$

where:

i – the unique ordinal number of the diagnosed unit,
S_t – the current state of a status diode (0 - passive, 1 – active),
T_r – a work mode (0 – a continuous signal, 1 – the pulse signal),
K_{zi}, K_{zl}, K_c – colours (accordingly: green, yellow or red; marker equals logical true corresponds to the current colour),
P_r, S_{rp}, P_{dk}, P_{zp} – accordingly: a uniform pulsation, the fast uniform pulsation, the pulsation in two colours, the pulsation with a pause,
K_r – times of pulsation (3, 4, 5, 6-times)

Each failure has been assigned to the unequivocally defined diagnostic vector, which is the pattern used for comparison of functions that define errors.

In the case of determination of bus work parameters, the described vector has been divided into components, with the aim of easier identification of damaged states:

$$W_i=[\ [W_{RUN}]\ [W_{BUS}]\ \ [W_{SYS}]\ ] \quad (2)$$

where:

W_{RUN} – a six-elements vector of the hardware components diagnosis,
W_{BUS} – the three-elements vector of the communication status diagnosis,
W_{SYS} – the five-elements vector of the configuration diagnosis, a data exchange management program, etc.

Table 1 shows forms of vectors constituting the main vector. The result of the submission is the fourteen-elements diagnostic vector, whose components were determined as:

- S_{tA}, S_{tP}, S_{tpls} – current states, accordingly: active, passive or pulse,
- K_{zi}, K_{zl}, K_c – as previously.

Vector denotation	Contents
W_{RUN}	W_{RUN}=[S_{tA} K_{zi} K_{zl} K_c S_{tp} S_{tpls}]
W_{BUS}	W_{BUS}=[S_{tA} S_{tp} S_{tpls}]
W_{SYS}	W_{RUN}=[S_{tA} S_{tp} S_{tpls} K_c K_{zl}]

Table 1. Vectors corresponded to isolated components

The encoded state's vector is unreadable for the end user. The transformation of the vector content W_i into a type of damage U_n is realized through the use of rules of an inference in the form of a simple conjunction, for example, the first vector:

$$\text{If } (S_t=s_j) \cap (T_r=s_j) \cap (K_{zi}=s_j) \cap \ldots \cap (K_r6x=s_j) \text{ Then } \cap U_n) \tag{3}$$

where:

s_j – a current state (0 or 1, j=2),
U_n- n-th damage (defined as a unique tag of a data record).

An error description is identified on the basis of the unique damage marker Un (syntax consistent with the equation 3). The selection function of corrective actions is defined in the following form:

$$\text{If } ZU=U_n \text{ Then } [Z_{AKR}= k[z]] \tag{4}$$

where:

ZU – a damage marker,
Z_{AKR} – the z-th element of the corrective actions vector.

The correct definition of the vector can lead to two situations: determined as:

- the corrective actions means the vector is strictly assigned to one type of damage,
- a given state vector corresponds to n failures.

In the second case the selection function of corrective actions takes the form of a logical alternative and its form of notation can be presented in the following form:

$$\text{If } ZU=U_n \text{ Then } [Z_{AKR1}= k[z] \text{ or } Z_{AKR2}= k[s] \text{ or } \ldots \text{ or } Z_{AKRm}= k[p]] \tag{5}$$

A size of corrective action vectors is variable and depends on the type of error, impact degree at the diagnosed system and number of steps required in order to restore the correct status.

It is worth mentioning that the presented system is complete (contains all possible combinations of fault states of frequency inverters). An implementation of the method was performed using a high order language. Figure 12 shows a screen of Computer Control and Diagnostic System of Distributed Drives used to determine the status of the integrated frequency inverter of a MOVIFIT unit. Representation of the diagnostic vector is realized by a state's definition in subsequent dialog windows (Fig. 13).

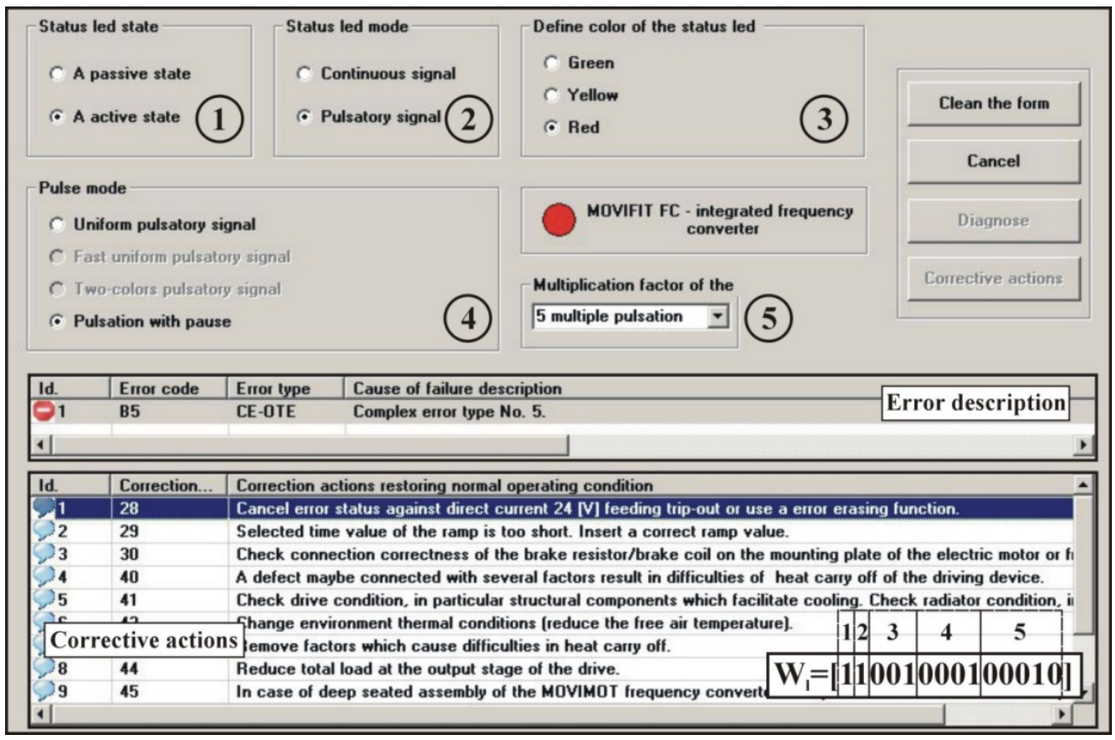


Fig. 12. A view of the diagnosis window of an integrated frequency converter of the MOVIFIT unit

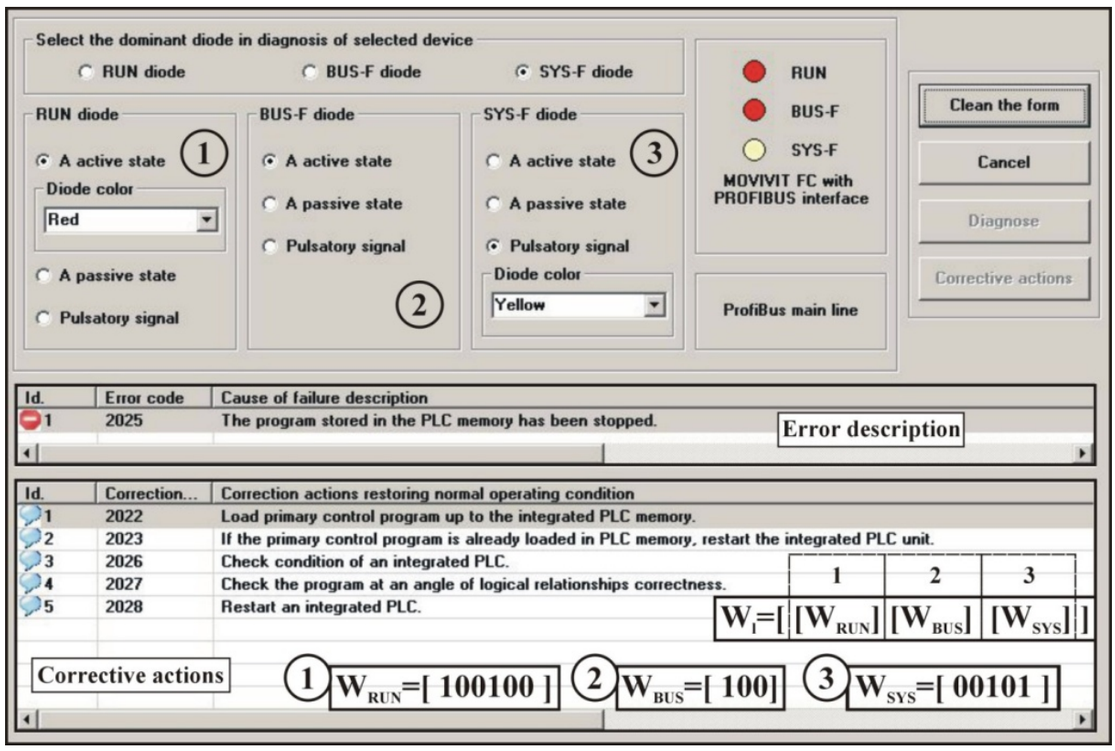


Fig. 13. A view of the MOVIFIT communication bus diagnosis window

In the case of diagnosis of the MOVIFIT bus, the exemplary diagnostic vector is presented in Figure 13. The division into three sub-vectors were introduced by defining the weight marker of diagnosed subsystems. The weight is introduced to define the subsystem, among which exist an irregularity. Selecting a tag "RUN LED" will search for solutions among the hardware components of the application. The other markers work similarly. The whole advisory system has been elaborated in fourteen windows.

4.2 An automatic mode of the definition of the failure states

In the case of the automatic failure detection, a phase of diagnostic vector definition is omitted. Content of the failure vector U_n is directly defined by using internal functions of frequency inverters and I/O modules. The error is identified on the basis of the individual 8-bit code contained in the status word of SLAVE units.

The primary disadvantage of this approach is dependence on the correct condition of the PLC controller and the ProfiBus network. The communication disturbance or incorrect PLC operational status is indicated by the diagnosing system of the PLC controller. The structure of the CCaDSoDD is hierarchical; that allows a reciprocal supplementation of defined subsystems.

It is necessary to present the main assumption, which confirms the presented stand of the authors, in the following form: entrusting to the advanced mechatronics devices of the control functions (which require a high level of reliability) allows functionality extension at additional functions and data connected with diagnostic functions without loss of the accepted quality factor.

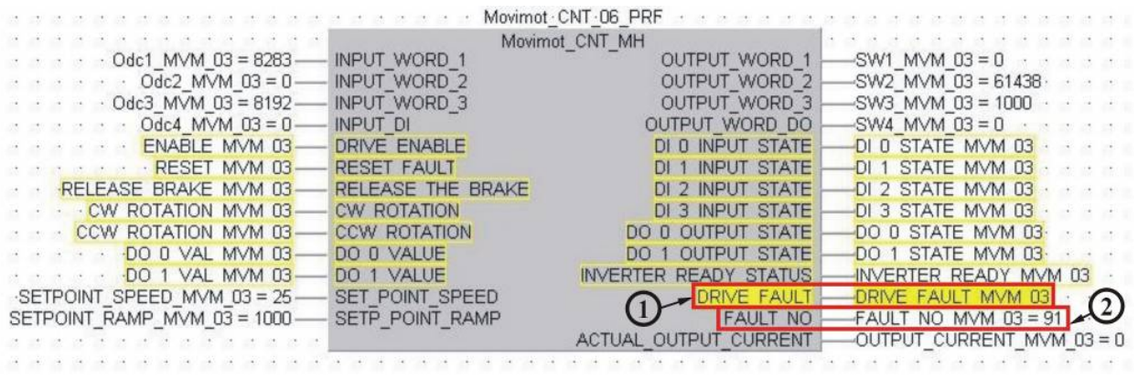


Fig. 14. A general view of the MOVIMOT functional block, dedicated to the fault identification of frequency inverter, where: 1 – a fault marker, 2 – a number of the identified error

The group of frequency inverters coupled by the ProfiBus DP network are controlled and monitored via integrated control and diagnostic blocks (placed in PLC memory) and elaborated expert system, at an angle of functional features of integrated self-diagnostic algorithms of frequency inverters (acquiring error codes) which are utilized in the CCaDSoDD system (a set of implication rules that converts numerical codes to text strings with user directions).

4.3 Self-diagnostic functions of frequency inverters

The presented example of frequency inverters allows the identification of a domain which could be used for computer-aided failures detection and their elimination. It is worth stressing that modern mechatronic devices offer very reliable and expanded functions connected with monitoring of internal subsystems.

Utilization of inner self-diagnostic algorithms coverage by specialized electronic circuits, integrated inside mechatronic devices, brings many benefits, among others:

- a separation of system functions (discharging of resources),
- shortening of time assigned to detection and elimination of identified failures,
- increasing application reliability.

A result of self-diagnostic functions is given in the form of individual error codes. Error codes contain 8-bits and are easy to gather by the PLC connected with the frequency inverters by the ProfiBus DP network. Additionally, the PLC performs several functions at the same time:

- processing of the control algorithm determined by dependences defined in accordance with the cyclogram,
- gathering the diagnostic data from frequency inverters circuits,
- checking current parameters of additional elements (like sensors, electric drives, frequency inverters, etc.) within the defined dependences (relations between states of constituent elements) determining the current state.

Ordinal number	Assigned error set	Description of sources
1.	AE-PTE	• incorrect configuration of hardware resources, • causes generated in the stage of loading of programs, settings or parameters, • an overflow of a memory stack, • disturbances of the beginning or an execution of program actions, • abnormalities of data conversion, settings or an active mode change.
2.	EE-HTE	• the ProfiBus DP bus failures, • incorrect matching of the hardware layer, • an abnormal work of hardware devices (reading errors, incorrect hardware configurations, etc.), • diagnostic information, • failures of hardware components like electronic circuits, feeding interfaces, • abnormal conditions of supervised units (i.e. decay of feeding voltage, phases, etc.).
3.	CE-OTE	• complex relationships between identified failures belonged to the AE-PTE and EE-HTE collections

Table 2. Types of identified failure states with description of their sources

All identified failure conditions were grouped by the authors into three main domains:

- AE-PTE – application errors (connected with programming activities, configuration parameters, etc.),
- EE-HTE – hardware errors,
- CE-OTE – complex errors (coupled by a conjunction of previously specified groups).

The end user has not got the possibility of any influence over the form and size of the error set notified by the internal resources of frequency inverters, because access to the embedded circuits and functions in many cases is blocked by several methods (lack of sufficiently documentation, resource interlocking and simplification of accessible functions).

A complete exploitation of advantages of AE-PTE and EE-THE collections enables continuous monitoring of values stored in variables included in the control algorithm.

A designer of the control algorithm decides how to use codes and how to define mutual dependences which define complex errors CE-OTE. Error codes are collected and gathered in the PLC memory. On the basis of their current values and conditions of other variables, a user realizes the diagnostic inference. Each of the frequency inverter error codes is treated like an independent diagnostic premise.

An identification of complex errors is performed by a comparison of two (or more) simultaneously identified errors belonging to different groups within the defined relations placed in the statements of the expert system of the CCaDSoDD application. Data gathered in the CCaDSoDD expert system are transmitted in on a package in every cycle of the PLC controller.

The identified error code is sent to a PLC controller in the form of one byte, transformed up to the hexadecimal number. For the considered groups of frequency inverters produced by the SEW Eurodrive group, the authors isolated more than 70 failures identified only by embedded self-diagnostic functions.

Taking into consideration functional constraints of considered devices, the final results are satisfactory, because a defined number describes attributes referred to the hardware subsystems (a group of diagnostic premises identified directly). The identification of residual conditions could be realized as a result of an elaboration of decision rules matched to the concrete application of distributed drives. A diagnostic inference is possible in two independent modes:

- directly on the basis of premises delivered in the form of frequency inverters error codes,
- indirectly by connection of frequency inverters error codes and additional statements implemented in the expert system.

Isolated errors (used in the direct diagnostic inference) expanded by relations connected with monitored values of the control algorithm allow defining a group of corrective actions with a quantity exceeding 270. The accepted inference method is based on dynamic checking of a diagnosed subsystem and a failures localization model at an assumption supported by the serial inference algorithm.

The method of identification and definition of reciprocal dependences of complex errors were defined in accordance with the following assumptions:

- a complex error is implicated by two (or more) errors belonging to the set of AE-PTE or EE-HTE errors coupled with a conjunction condition (definition of used implications shows Figure 15),
- in the case of the occurrence of two or more errors from the same set (AE-PTE or EE-HTE), errors are treated as individual diagnostic premises (adequately $u_{AE\text{-}PTE}$ and $_{uEE\text{-}HTE}$); occurrence of many errors in the domain of one collection do not implicate a complex error (identified diagnostic premises take part in the inference process in accordance with defined rules in the CCaDSoDD system),
- each of the errors (AE-PTE or EE-HTE) can belong to many rules implicating complex errors.

The mediate form of the inference has a lot of advantages, among others:

- a possibility of an unbounded forming of statements, fitted to the concrete applications,
- shortening of time connected with an elaboration of expert systems,
- an open structure and feasibility of development.

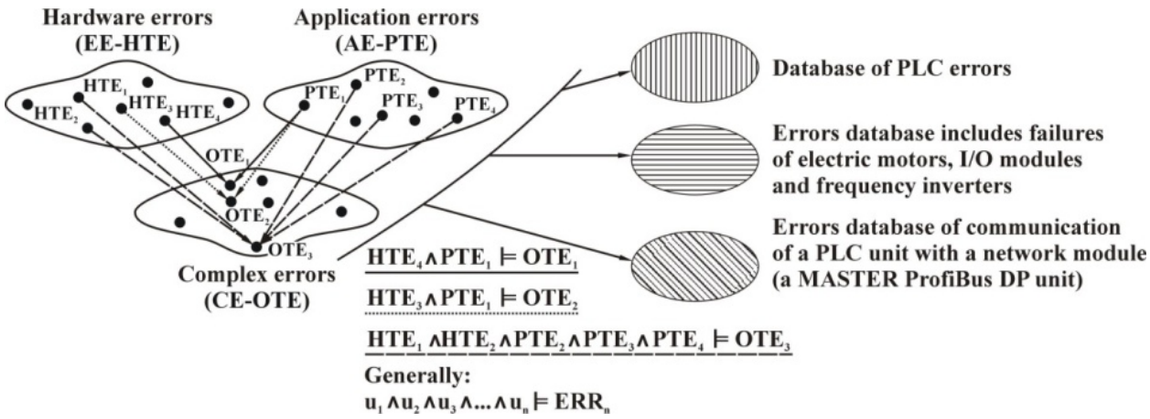


Fig. 15. A definition of the complex error implication (CE-OTE type), where:
$AE\text{-}PTE_n$, $EE\text{-}HTE_n \subset u$, u – diagnostic premises ($AE\text{-}PTE_n$ or $EE\text{-}HTE_n$, $n \subset N$), ERR_n – the logical implication of a conjunction of u premises

Defined relational databases are part of the expert system. On the basis of rules, the CCADSoDD system chooses descriptions of the identified error and corrective actions (steps enabling a restoration of correct operating conditions). All relational databases based on the SQL language (Structured Query Language) were divided into tables (the main key is equivalent to the error code).

At the stage of an error identification, the authors accepted a few meaningful assumptions:

- the monitoring and collection of all data are realized within PLC tasks,
- maximal time of a symptom generation is equal to the PLC cycle, increased at the time connected with ProfiBus DP bus refreshing, within the equation (Korbicz et al., 2002):

$$\theta_j = \max_{k:f_k \in F(s_j)} \left\{ \theta_{jk}^2 \right\} \tag{6}$$

where:

F(s_j) - a failures set detected by the diagnostic signal s_j,

θ^2_{jk} - maximal time from the moment of occurrence of the k failure to generation of the j symptom.

- a set of diagnostic tests does not include elements corresponding to symptoms with time occurrence exceeding the time predicted to a diagnosis period,
- protective functions (safety actions) of the analysed case are executed in the time equal to the value of one PLC cycle,
- dynamic properties of the diagnostic symptoms formation is dependent on the PLC cycle,
- in the elaborated expert system, the authors do not improve a states detection on the basis of the time and detection sequence of diagnostic symptoms (a packet block transfer).

Implication rules are referred to all groups of diagnosed units and databases of the CCaDSoDD system. Recording of error values are stored within a defined standard (Fig. 16).

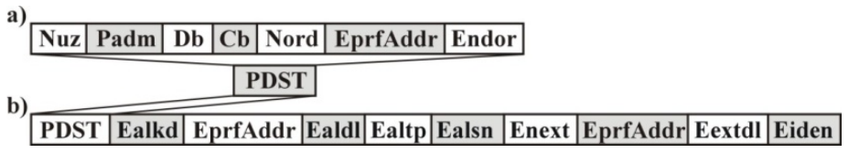


Fig. 16. A view of entry error patterns of the frequency inverter: a) basic, b) extended, where: Nuz – a user name, Padm – a authorization level, Db – date, Cb – current time, Nord – the ordinal number of the SLAVE unit, EprfAddr – network address error, Endor – ordinal number error of the SLAVE unit, Ealkd – alarm code, Ealdl – length of the alarm information, Ealtp– alarm type, Ealsn – socket number with an active error state, Enext – extended diagnostic information code, Eextdl – data length of the extended diagnostic information, Eiden – identification number of SLAVE unit

The pattern of error entries has been divided into two main subgroups:

- basic - main errors of the SLAVE (frequency inverter) unit,
- extended - includes all data of the basic pattern widen of diagnostic information of the ProfiBus DP network and hardware devices.

5. An investigation of the impact of volume and syntax of the algorithm at the performance of the PLC controller

From the standpoint of the operation of the control unit, a main parameter is refresh time values stored in registers of the control device. The assumption of the entrusting of control and diagnostic tasks to one device requires a high reliability factor of the data processing element (in this case the PLC controller).

During the study, the influence of the complexity of the integrated control and diagnostic algorithm at processing parameters of the PLC was specified (Świder & Hetmańczyk, 2009d). Table 3 summarizes the configuration parameters and their numerical values, adapted to the verification needs of the PLC algorithm.

Networks configuration parameters	Configuration	Physical unit
Data transmission type	Digital differential signal compatible with RS485 standard, NRZ coding type	-
Data rate	variable	kbits
Data protection type	HD=4, parity bit, start/end transmission separator	-
Data carrier	Two-wire twisted shielded pair (A type)	-
Power supply type	With additional cable	-
Network topology	Main line	-
Bus length	12	[m]
Number of stations	9 (MASTER + 8 x SLAVE)	-

Table 3. Basic configuration parameters of the tested network

Refresh time values of the Profibus DP bus were recorded on the basis of parameters recorded by the MASTER QJ71PB92V_ContDiagBLK dedicated block. A set of variable parameters of realized verification contains the data transfer rate of the ProfiBus DP bus and organization of the syntax of the control and diagnostic algorithm.

Time values of maximum and minimum cycles were subjected to an exceptional reading, while the current time is calculated as the arithmetic mean of samples, obtained within 10-minutes cycles of observation (recorded with a sampling resolution equal to the PLC cycle, labelled as TC_PLC).

The presented graphs also show values of the PLC cycle time. It should be noted that the numerical value of the first cycle of the PLC controller is not known. The only value registered during the first cycle is the maximum refresh time value of the bus network, given in the second cycle of the controller.

The first series of measurements were performed using the following assumptions:

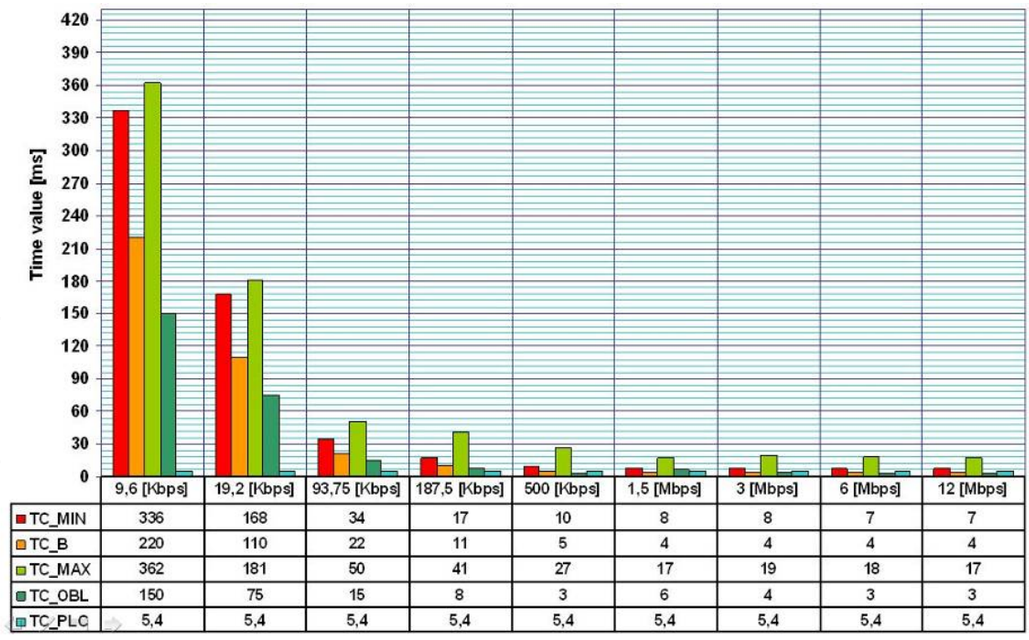
- program blocks were placed in the main program,
- control blocks of drives are supported from the calling subroutine placed in the main program,
- additional blocks of a control algorithm (determining a course of work of controlled objects) were positioned inside individually called subprograms.

Figure 17 shows measured values of PLC refresh times, with a characteristic of work specified in the form of assumptions defined with respect to the first measurement cycle and configuration of slave units shown in Table 3.

The second of the considered cases assumed placement of all blocks in the main program window (Fig. 18). This configuration of the algorithm affects the growth of the recorded maximal value of the refresh time of the ProfiBus network bus. The identified phenomenon is related to the need for processing a larger number of variables, their support and assignment of parameters contained in the main program.

SLAVE/MASTER unit type	Category	Network address	Unit configuration
QJ71PB92V	M	0	-
-	NU	1÷2	-
MFP/MOVIMOT	S	3	[000] 3PD+DI/DO (MFP2x)
MFP	S	4	[000] 0PD+DI/DO (MFP2x/3x)
MFP/MOVIMOT	S	5	[000] 3PD+DI (MFP2x/3x)
MFP	S	6	[000] 0PD+DI (MFP2x/3x)
MFP/MOVIMOT	S	7	[000] 3PD+DI/DO (MFP2x)
MFP	S	8	[000] 0PD+DI/DO (MFP2x/3x)
MFP/MOVIMOT	S	9	[000] 3PD+DI (MFP2x/3x)
MOVIFIT	S	10	[000] Slot unused [001] MOVIFIT Status [002] FC 3PD [003] Slot unused [004] FC/SC: 6/8 DI [005] 2/4 D0 [006÷008] Slot unused

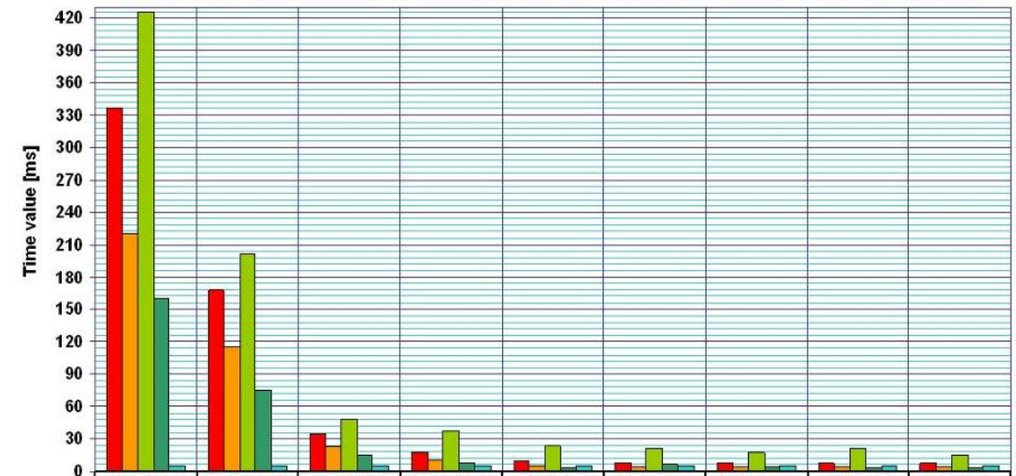
Table 4. A matching of units and configuration parameters of the considered application



	9,6 [Kbps]	19,2 [Kbps]	93,75 [Kbps]	187,5 [Kbps]	500 [Kbps]	1,5 [Mbps]	3 [Mbps]	6 [Mbps]	12 [Mbps]
TC_MIN	336	168	34	17	10	8	8	7	7
TC_B	220	110	22	11	5	4	4	4	4
TC_MAX	362	181	50	41	27	17	19	18	17
TC_OBL	150	75	15	8	3	6	4	3	3
TC_PLC	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4

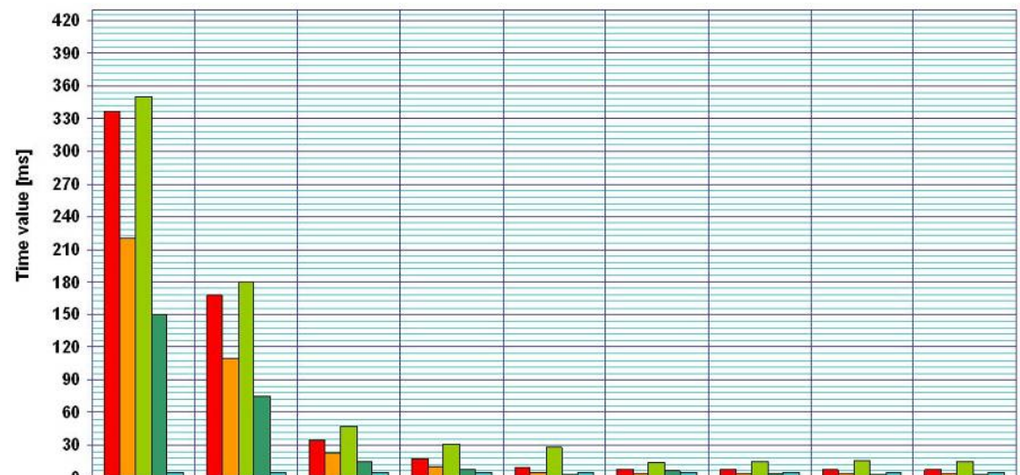
Fig. 17. The dependence of characteristic time periods of the ProfiBus DP network (case of the complete configuration including all diagnostic and control blocks called from subroutines), where: TC_MIN – minimum value [ms], TC_B – current value [ms], TC_MAX – maximum value, TC_OBL – calculated value (predicted value), TC_PLC – PLC cycle time (onefold time of an algorithm processing)

In the first configuration case a deficiency of variable that initiate subroutine execution causes an omission of the algorithm execution contained in its structure, which significantly limits the data amount subjected to processing in the first cycle of the driver.



	9,6 [Kbps]	19,2 [Kbps]	93,75 [Kbps]	187,5 [Kbps]	500 [Kbps]	1,5 [Mbps]	3 [Mbps]	6 [Mbps]	12 [Mbps]
TC_MIN	336	168	34	17	10	8	8	8	7
TC_B	220	115	23	11	5	4	4	4	4
TC_MAX	426	201	48	38	24	22	17	21	15
TC_OBL	160	75	15	8	3	6	4	3	3
TC_PLC	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4

Fig. 18. The dependence of characteristic time periods of the ProfiBus DP network (case of the complete configuration including all diagnostic and control blocks placed in the main program)



	9,6 [Kbps]	19,2 [Kbps]	93,75 [Kbps]	187,5 [Kbps]	500 [Kbps]	1,5 [Mbps]	3 [Mbps]	6 [Mbps]	12 [Mbps]
TC_MIN	336	168	34	17	10	8	7	7	7
TC_B	220	110	23	11	5	4	4	4	4
TC_MAX	350	179	47	31	28	14	15	16	15
TC_OBL	150	75	15	8	3	6	4	3	3
TC_PLC	5,5	5,5	5,5	5,5	5,5	5,5	5,5	5,5	5,5

Fig. 19. The dependence of characteristic time periods of the ProfiBus DP network (case of the configuration based on main blocks)

Because of an application of the rapid CPU unit (Q25HCPU) there were no significant differences in the recorded values of current bus cycle times and refresh times of the PLC resources.

Estimated (computed) values of the refresh time of the bus network, obtained from the MELSOFT GX CONFIGURATOR DP, are on average two times smaller than the actual recorded values. The presented fact must be taken into account at the design stage of the industrial network structure (step of matching network module and a data transfer rate choice).

A third case of measurements (Fig. 19) was carried out using the configuration shown in Table 4 and the PLC algorithm, without loading of the control code (only control blocks of the data exchange, i.e. QJ71PB92V_ContDiagBLK and PROFIBUS_NET).

In the described test cycle a slight decrease in values of maximum refresh time of bus was noticed, both in relation to the previously described configuration. However, current bus refresh times oscillate around previously identified values.

It can be concluded that the current cycle time of a ProfiBus DP bus does not depend on the configuration architecture of a control and diagnostic algorithm, which confirms the usefulness of the application in complex appliances extended at Fieldbus network buses.

Increasing data transmission speed of the network bus does not cause significant problems in the recorded times, despite changes in the structure of the configuration of the control and diagnostic algorithm.

The proposed form of the structure organization of the developed control and diagnostic system is configuration organized in subroutines called from the main program.

The configuration is beneficial for reasons arising from the need to fulfil the requirements of smooth operation of the PLC processor and exchange of data with remote units.

Small time values of the current bus cycle (TC_B) confirm the validity of the correct implementation of the described diagnostic functions.

6. Conclusions

The presented expert system of the CCaDSoDD software is one of the modules of the computer diagnostic system of distributed drives and it is equipped with the following embedded functions:

- on-line diagnosis,
- off-line diagnosis,
- preliminary diagnosis.

The computer environment is convenient to use, but an elaboration of the expert system is rather complicated. Significant difficulties can arise for the following reasons:

- the process of data gathering involves participation of high-class engineers,
- a database designer must identify all possible failures,
- the expert system requires performing of the additional database with stored corrective actions,

- the readable user interface is necessary,
- work stability depends on the specified protocol.

Self-diagnostic functions are frequently used in removal of states of failures by qualified staff, but an intention of the authors was to turn the attention to the utilization of embedded diagnostic functions at the stage of design of the final form of applications included in modern mechatronic devices.

Advantages of the presented approach are meaningful and can be summarized by several points:

- increasing reliability of the considered application,
- a possibility of preparation of fault detection functions for every mechatronic unit (taking into account its functional constraints and embedded functions),
- utilization of more advanced mechatronic units allowing a widening of the amount of possible identified operating conditions by embedded subsystems,
- to primary parameters belong, among other things: current, minimal and maximal bus cycles (detected by a industrial network module of the PLC unit), the diagnostic information, a current work status, error codes, reactive and active powers, etc.

Beyond the specified advantages a group of functional constraints also exist, resulting in an enlargement of expenditure connected with:

- a preparation of the control algorithm widened by memory cells which include values of diagnostic premises,
- an elaboration of the expert system and inference rules (diversified at an angle of the concrete application).

The application of developed control and diagnostic blocks with the proposed structure of the algorithm allows for:

- reduction of the time needed for an elaboration of the complex control and diagnostic system,
- resignation of the phase of copying the contents of global memory addresses to memory buffers (a direct manipulation on memory buffer registers),
- the implementation of different configuration types:
  - with an implementation of the function corresponding to the control algorithm,
  - using diagnostic functions,
  - combining these functions into the one system extended with additional capabilities of an expert system.

7. References

Ryabinin, I. (1976). *Reliability of engineering systems*, MIR Publishers, ISBN: 0714709492, Moscow

Legierski, T. (1998). *PLC Controllers Programming*, Jacek Skalmierski Publishing, ISBN: 838664416-8, Gliwice

Kwaśniewski, J. (1999). *Industrial programmable controllers in control systems*, AGH Publishers, ISBN: 83-86320-45-1, Kraków

Dhillon, B.S. (2002). *Engineering Maintenance - A Modern Approach*, CRC Press, ISBN 978-1-58716-142-1, New York

Korbicz, J. & Kościelny, J. & Kowalczuk, Z. & Cholewa, W. (2002) *Processes diagnostic: models, methods of the artificial intelligence, applications*, WNT, ISBN 83-204-2734-7, Warsaw

Boyer, S. (2004). *SCADA: Supervisory Control and Data Acquisition*, ISA, ISBN: 1556178778, 2004, New York Fraden J. (2004) *Handbook of modern sensors –physics, design, applications*, Springer, ISBN 1441964657, New York

Prasad, K.V. (2004). *Principles of Digital Communication Systems and Computer Networks*, Charles River Media, ISBN 158450-3297, New York

Mackay, St. & Wright, E. (2004). *Practical Industrial Data Networks: Design, Installation and Troubleshooting*, Elsevier Linacre House, ISBN 07506-5807X , Oxford

Bloch, H.P. & Geitner, F. K. (2005). *Practical Machinery Management for Process Plants – Machinery Component Maintenance and Repair*, Elsevier Linacre House, ISBN 0750677260, Oxford

De Silva, C.W. (2005). *Mechatronics: an integrated approach*, CRC Press, ISBN 0849312744, New York

Świder, J. & Hetmańczyk, M. (2009). Hardware and software integration of mechatronic systems for an example measurement path for temperature sensors, *Solid State Phenomena*, Vol. 147-149, 2009, pp. 676-681, ISSN 1012-0394

Świder, J. & Hetmańczyk, M. (2009). Method of indirect states monitoring of dispersed electric drives, [in:] Proceedings of the The Sixth International Conference on Condition Monitoring and Machinery Failure Prevention Technologies, Dublin, (June 2009), pp. 1171-1179

Świder, J. & Hetmańczyk, M. (2009). The visualization of discrete sequential systems, *Journal of Achievements in Materials and Manufacturing Engineering*, Vol. 34, No. 2, (April 2009), pp. 196-203, ISSN 1734-8412

Świder J. & Hetmańczyk, M. (2009) Computer-aided diagnosis of frequency inverter states, *Problems of Working Machines*, pp. 57-66, Vol. 33, ISSN 1232-9304

Świder, J. & Hetmańczyk, M. (2010). Adaptation of the expert system in diagnosis of the connection of the PLC user interface system and field level, *Solid State Phenomena*, Vol. 164, 2010, ISSN 1012-0394

Świder, J. & Hetmańczyk, M. (2011). *The computer integrated system of control and diagnosis of distributed drives*, Silesian Technical University Publishing, ISBN: 978-83-7335-844-7, Poland

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the Creative Commons Attribution 3.0 License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen