UDC 62.932.2

I. NEVLIUDOV, V. YEVSIEIEV, N. DEMSKA, S. NOVOSELOV

DEVELOPMENT OF A SOFTWARE MODULE FOR OPERATIONAL DISPATCH CONTROL OF PRODUCTION BASED ON CYBER-PHYSICAL CONTROL SYSTEMS

The **subject** of research in the article is production management processes based on cyber-physical systems. The **purpose** of the work is to automate the management of production processes using the cybernetic component and analyze the results. To achieve this goal, it is necessary to solve the following **tasks**: develop an analytical and logical model of the CPPS development management process taking into account the requirements specified in the terms of reference for the development CPPS; to present the analytical and logical structure of connections in the form of a model of interaction of the main windows and all the necessary graphical elements for the implementation of the full-featured interface of the developed CPPS; to carry out experimental researches of efficiency and practical approbation of the received theoretical results, by comparison of classical methods and the developed ones. **Results**: The targets, physical and cybernetic components of the CPPS development management process were combined in a single information space from the beginning of development to its implementation. The process of checking the achievement of the main goal of CPPS development has been automated, which makes it possible to make changes and manage the CPPS development process at any level and stage of the proposed technology. The process of managing the development of the cybernetic component is automated on the basis of synthesized algorithms of functioning using the GUI elements of object-oriented programming languages, which made it possible to increase the flexibility of the proposed architecture and technologies for automating the CPPS development process and to reduce the development time of the cybernetic component. **Conclusions**: the proposed models and methods will not only automate the process of managing the development of CPPS from scratch, but will also make it possible to automate the process of managing the development of the cybernetic component for the modernization and improvement of existing CPPS.

Keywords: Industry 4.0; Cyber-physical Production Systems; technological process; analytical-logical structure of relations; control; "OSCEM".

Introduction

Trends in the development of the modern world require a revision of approaches to the use of high technology and their role in areas of human activity. Increasing the amount of information, increasing the requirements for its accuracy and timely submission for analysis and decision-making, in real time, are forcing changes in approaches to industrial technologies. Current trends in the field of production automation and data exchange in industrial technologies have different names, but they all belong to the same process [1-2]. We are talking about a new, more rational production [3-4], which uses revolutionary for modern industry cyber-physical systems, Internet of Things technology and cloud computing.

"Smart production", related to Industry 4.0, involves the optimization of production processes through innovation, control of physical processes, creating a virtual copy of the physical world, as well as more accurate and timely decisions [5-9]. Processes in such production are guided by cyber-physical connections that allow real-time decision-making and feedback between planning and actual production [10]. Due to the rapid development of Cyber-physical Production Systems (CPPS), the world is paying significant attention to various aspects of their formation and operation [11-12].

These systems provide tools to overcome complexity and provide flexibility, integration with existing management systems, but there are no appropriate management architectures that would provide flexible adaptation during their operation, as well as theoretical provisions [13] that would allow moving from the analysis of these features to the synthesis of cyber physical systems. Consequently, in order to achieve the practical implementation of CPPS, a systematic methodology of the process of management, collection, processing and application of data for CPPS is being developed. This is due to the fact that CPPS can be successfully implemented only when all the processing criteria and application methods for various data are developed, the sequence of the development process is determined taking into account their existence in real time due to the nature of the production process.

Problem statement and research purpose

In this paper, we propose to interpret the CMMI (Capability Maturity Model Integration) model with a consistent decomposition of the main goal of CPPS development, as a set of sub-goals and objectives for each hierarchical level of CPPS development management architecture, which must ultimately meet the goal developing CPPS.

To conduct experimental research and verify the correctness of the proposed solutions in this work, based on the developed automation system for managing the development of complex CPPS, it is necessary to develop fragments of cyber-physical production systems and compare the results with existing standard approaches to cyber-physical production systems or their fragments.

Research and analysis of the obtained results

Based on the requirements set by the customer to the developed "Integrated system of operational and control of production of the enterprise (OSCEM)" we will define the "main goal" (Aim_i) and tasks ($Task_j$) of CPPS development, which are presented in table 1.

156

ISSN 2522-9818 (print)
ISSN 2524-2296 (online)   *Innovative technologies and scientific solutions for industries. 2020. No. 4 (14)*

Table 1. *Definition of Aim_i and $Task_j$ at the initial stage of the process of managing the development of the cybernetic component "OSCEM"*

Aim_i	$Task_j$	Sub_S_k	$G(AEofS)_j$
Integrated system of operational dispatching control management of the enterprise's production	Production	Current orders	Order archive
			Changes in orders
			Final product
		TCD Acceptance	Adding materials
			Acceptance of economic works
			Reception of overtime
	Warehousing operations	Purchase of materials	Purchase of goods and materials
			Return to suppliers
		Sale of goods	Sale of goods
			Return to buyers
		Warehouse management	Transfer to the warehouse
			Admission from the warehouse
			Write-off of goods and materials
			Write-off of unusable goods and materials
	Guides	Enterprise	The structure of the enterprise
		Employee map	Position
			Discharge
		Inventory guide	Trademark
			Dimensions
			Unit
		CDT guide	Format
			Devices
			Technological operations
			Descriptions of technological operations
			Changes in technology
		Counterparty guide	State
			Region
			City
			Type of enterprise
	Analysis	Order	Execution of current orders
			Completion of the order
			Filter by dates
		Selling	Sales made
			Open sales
			Filter by dates
		Storage	Quantity in stock
			The number of defects in the warehouse
			Filter on products

When analyzing, we can decompose Aim_i on $Task_j$ by levels: multisystem (MS''_0), subsystems (Sub_S_k) and groups ($G(AEofS)_j$)), while the atomic element ($AEofS)_j$ for $G(AEofS)_j$ will be the elementary graphic element / or event that is caused by the user as a reaction of the cybernetic component to an action or flow of input / output information.

Based on the results of Aim_i decomposition (table 1) it is possible at the initial stage to form the analytical-logical structure of the developed CPPS, which determines the main functions and properties that will determine the future number of visual graphics ($Form_1^{master}$, $Form_2^{slave}$ of CF_1 elements) and connection between them (fig. 1).

At the next stage of "OSCEM" development, it is need to define the necessary and sufficient set of $AEofS_j$ for each $G(AEofS)_j$, which describes Sub_S_k and allows to achieve Taskj in accordance with the requirements of the customer's TT. To do this, the developer, based on the analytical-logical structure (fig. 1), describes all $AEofS_j$ and their relationships. Here is an example of a fragment

157

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2020. № 4 (14)   *ISSN 2524-2296 (online)*

of the choice of $AEofS_j$ and $Task_j$ for = "Guides", Sub_S_k = "CDT handbook" for $G(AEofS)_j$ = "Device 1" (fig. 2).

Based on the results obtained in Table1 and the analytical-logical structure of connections "OSCEM", presented in fig.1, for $Task_j$ = "Guides", Sub_S_k = "CTD Guide" for $G(AEofS)_j$ = "Device 1", namely the fragment

of selection $AEofS_j$ for $G(AEofS)_j$ = "Device 1" → "Part 1" → "Design documentation" and "Technology", we define that the following hierarchy of HMI (Human machine interface) of communications for "OSCEM" will be logical (fig. 2).
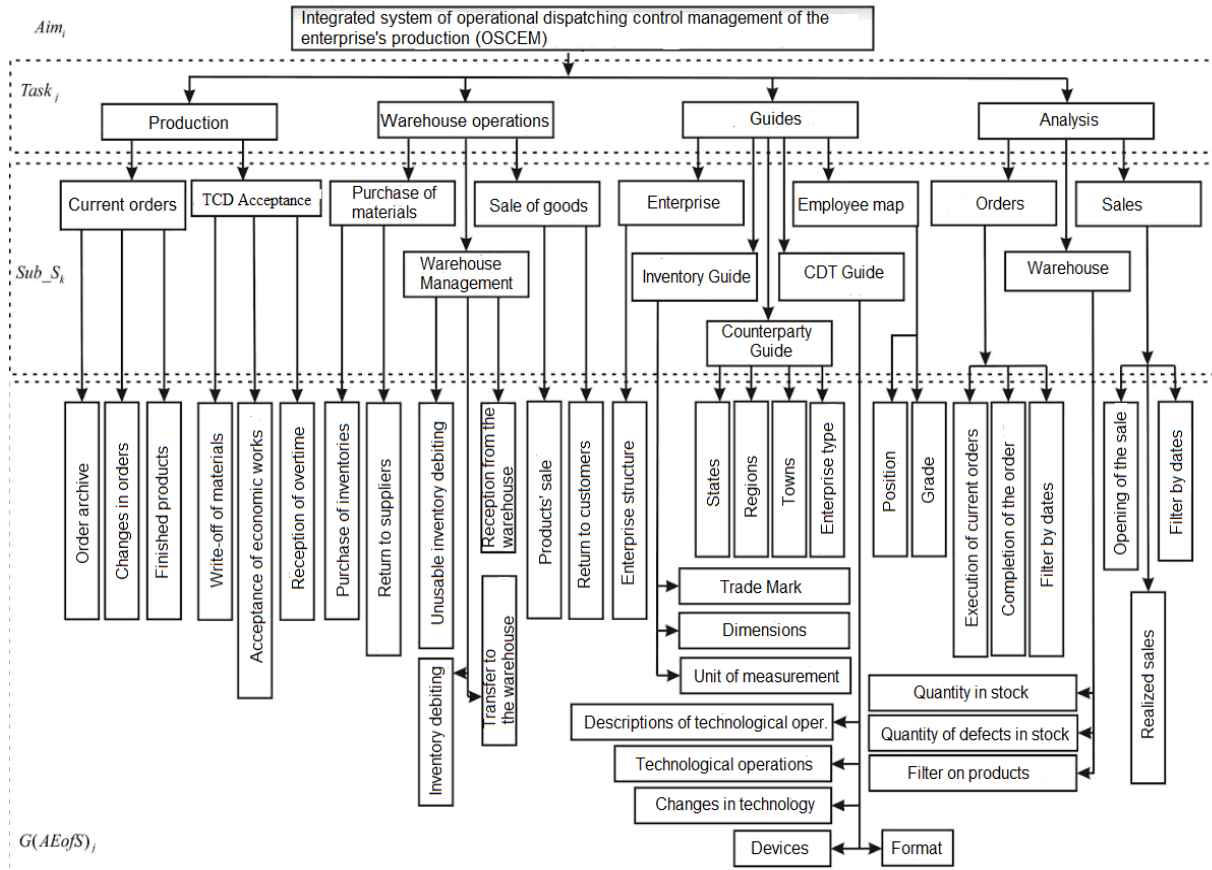
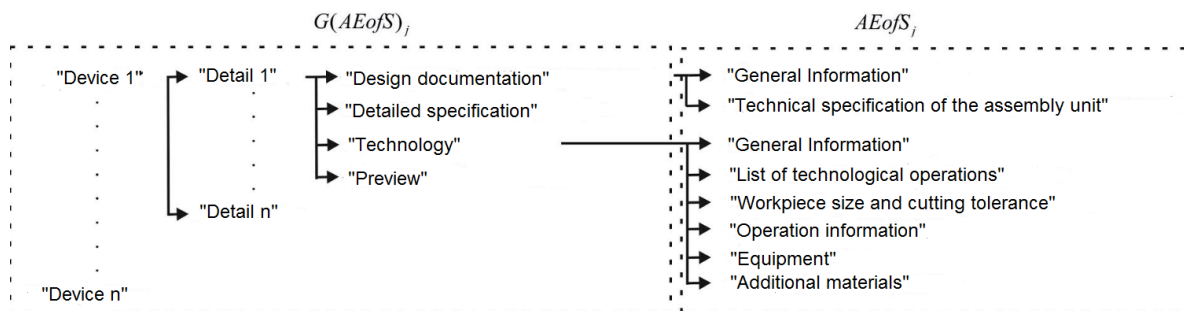**Fig.1.** Analytical and logical structure of "OSCEM" relations

Fig. 2. The fragment of selection $AEofS_j$ for $G(AEofS)_j$ = "Device 1" → "Part 1" → "Design documentation" and "Technology"

According the requirements of the customer's TT, it is necessary to determine the "parameter name" and "values" included in each $AEofS_j$. Table 2 shows the parameters selected by the developer and the type of value for $G(AEofS)_j$ = "Device 1" → "Part 1" → $AEofS_j$ = "Technology", which were approved by the customer and allow to achieve the goal and requirements of TT for the developed "System of operational dispatching management of the production enterprise "OSCEM".

The next step in the development management process is to present the analytical and logical structure of

relations "OSCEM" (fig. 1) in the form of a model of interaction (table 2) of the main HMI windows, it should be noted that in this case it is necessary to use industrial design requirements:

- convenience and ease of providing access to basic functions and forms of data (placement of controls, content, design of output messages and input formats);

- minimization of steps of access to the internal functionality of the developed system by means of the user interface.

Table 2. *Interaction models G(AEofS)ⱼ = "Device 1" → "Part 1" → AEofSⱼ = "Technology"*

AEofSⱼ	Parameters	Value type
"General information"	Order name	Texts
	Part / assembly name	Text numbers
	The main material	Numbers
	Number of parts (pcs)	Numbers
	Number of billits per 1 workpiece	Numbers
	Part weight (kg)	Numbers
	Workpiece weight (kg)	Numbers
"List of technological operations"	Area	Texts
	Operation	Texts
	Equipment	Texts
	Development	Numbers
	Number of works	Numbers
	Party	Logically
	Unit of norm	Numbers
	Preparatory / final time, Tpf	Numbers
	Time per unit, Tun	Numbers
"Workpiece size and cutting tolerance"	Length (mm)	Numbers
	Width (mm)	Numbers
"Transaction information"	Date	Date
"Equipment"	Name	Texts
	Number	Numbers
"Additional materials"	Name	Texts
	Unit of measurement	Numbers
	Number	Numbers

Based on the above requirements, it was decided to implement HMI using the main window of the application ($Form_{ASUP}^{master}$) and auxiliary windows ($Form_{techn}^{directoty-CAD}, ..., Form_{TMC}^{ASUP}$), which will be called through graphical elements $CD_1^{ASUP}, ..., CD_x^{ASUP}$ located on $Form_{ASUP}^{master}$ by the event initiated by the user or system. For convenience of representation of structure of communications "OSCEM" between the main window $Form_{ASUP}^{master}$ and auxiliary windows

($Form_{techn}^{directoty-CAD}, ..., Form_{TMC}^{ASUP}$) the developer builds the graphic model of communications of Windows Form on the basis of the improved methodology Constant (tab. 1-2). Let's present in table 3 the designation and description of auxiliary (called) Windows Form, all other elements will be implemented within the main window $Form_{ASUP}^{master}$ based on GUI elements (Graphical user interface) – Panel, GroupBox, etc.

Table 3. *Designation and description of Windows auxiliary Forms from the main window $Form_{ASUP}^{master}$*

Marking	Description
$Form_{accept}^{ASUP}$	TCD Acceptance
$Form_{purchase}^{ASUP}$	Purchase of inventory
$Form_{sale}^{ASUP}$	Sale of goods
$Form_{transfer}^{ASUP}$	Transfer to the warehouse
$Form_{reception}^{ASUP}$	Admission from the warehouse
$Form_{write\text{-}off}^{ASUP}$	Write-off of inventory
$Form_{write\text{-}off\_TMC}^{ASUP}$	Write-off of unusable inventory
$Form_{TMC}^{ASUP}$	Inventory guide
$Form_{directory\_CAD}^{ASUP}$	CDT guide
$Form_{contractors}^{ASUP}$	Guide of contractors
$Form_{techn}^{directory\_CAD}$	Part manufacturing technology

Therefore, all $Task_j$, Sub_S_k, $G(AEofS)_j$, which are not shown in the graphical structure model (fig. 3), will be implemented in the main GUI window "OSCEM" ($Form_{ASUP}^{master}$) according to the analytical-logical structure of relationships (fig. 1) using the following graphic elements, presented in table 4.

The graphical model of the Windows Forms relationship structure in "OSCEM" is presented in fig. 3.
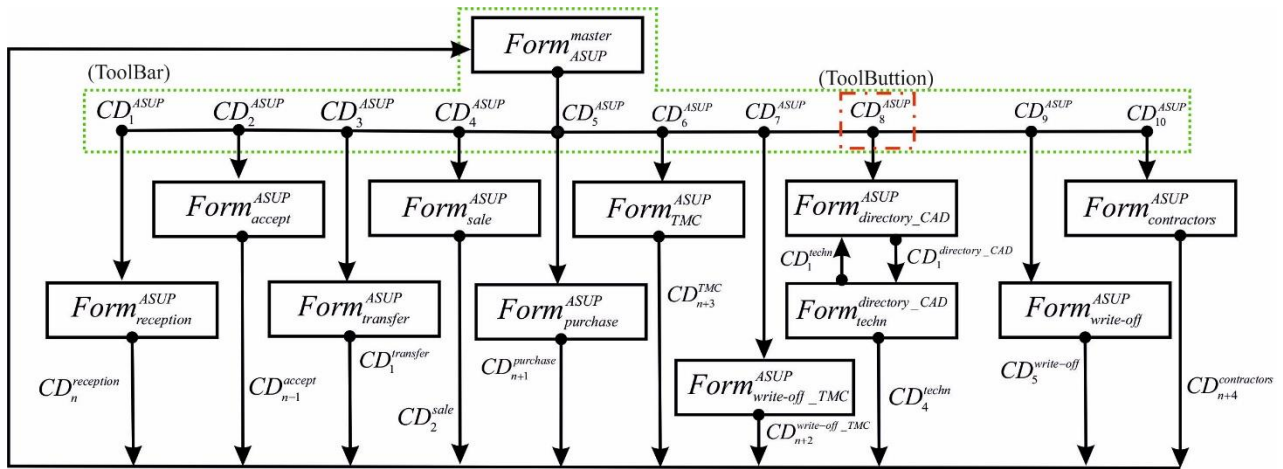

Fig. 3. The graphical model of the Windows Forms relationship structure in "OSCEM"

Table 4. *Selecting graphical GUI elements to implement user access to Taskj, Sub_Sk ma G(AEofS)j with the form $Form_{ASUP}^{master}$*

Marking	Name of the graphic element		Description
$Form_{ASUP}^{master}$	Windows Forms		Main GUI window "OSCEM"
$Task_j$	TabControl		The control is used to display multiple tabs (TabPage), similar to the separators in the notebook or labels in a set of folders of the file block Graphical user interface element designed to accommodate several other elements (buttons, menus) for quick access
Sub_S_k	ToolBar (ToolStrip)		Graphical user interface element designed to accommodate several other elements (buttons, menus) for quick access
$G(AEofS)_j$	ToolButton (ToolStripButton)		The child element ToolBar (ToolStrip) is highlighted as a button with an image
$AEofS_j$	Panel	Label	Displays text that cannot be modified by the user
		Edit	Designed to enter user data and is a single-line field
		DBGrid	Displays data from the database and allows to make and save changes
	GroupBox	DBMemo	Is a multi-line text editor that allows you to edit the text of the window, the information of which is stored in the database
		DBListBox	Allows the user to select one or more items from a predefined list of information stored in the database
$AEofS_j$	Button		Represents a standard button that the user can press to perform an action

Graphic element *Panel* serves as a container (parent component), which combines a group of controls, input / output and playback of information.

GroupBox is a container with a frame and an inscription, which unites a group of related controls.

Based on the developed analytical and logical structure of connections (fig. 1) and selected graphical elements of the GUI (table 4), we present the implementation of the GUI "Main Control Menu" ($Form_{ASUP}^{master} \rightarrow Task_j \rightarrow Sub\_S_k \rightarrow G(AEofS)_j \rightarrow AEofS_j$) in the form of a graph (fig. 4).

ImageList is a non-visual component that is a set of images of the same size that can be referenced by indexes to display them in ToolButton.

Based on the above solutions (fig. 1-2), as well as selected graphical elements (table 4) and the graph of the implementation of the GUI of the main window "OSCEM" (fig. 3), you can make a mathematical description of the structure of the main window "OSCEM" ($Form_{ASUP}^{master}$). A fragment of the mathematical representation of the implementation of Windows Forms ($Form_{ASUP}^{master}$) is presented in (1).
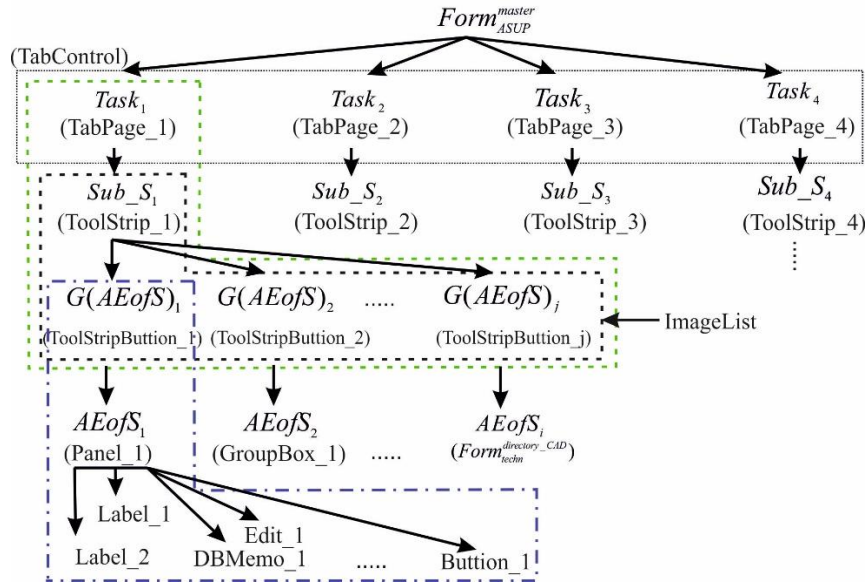
160

ISSN 2522-9818 (print)
ISSN 2524-2296 (online)   *Innovative technologies and scientific solutions for industries. 2020. No. 4 (14)*

Fig. 4. Graph of implementation of GUI "Main control menu" of the main window "OSCEM" ($Form_{ASUP}^{master}$)

Based on MP_{ASUP}^{master} the developer sets the values that should belong to the parameters that specify all the necessary properties of the main GUI window "OSCEM" and specifies the name "*Linguistic Variable*" in

(*Open_BD*, *Close_BD*), which allows to set the action on the event (*Events*) "*Load*" and "*Form Closed*", i.e. to connect to the database (DB) and disconnect from the database, respectively.

$$Form_{ASUP}^{master} (Form_{ASUP} PE)[(MP_{ASUP}^{master}) \xrightarrow{mp_3^{ASUP} = Form\_ASUP; mp_4^{ASUP}, mp_5^{ASUP}, mp_8^{ASUP},} \wedge$$

$$\wedge \xrightarrow{mp_9^{ASUP}, mp_{46}^{ASUP} = true; mp_6^{ASUP}, mp_{10}^{ASUP}, mp_{12}^{ASUP}, mp_{13}^{ASUP} = false; mp_{32}^{ASUP} = ./img/icon.png; mp_{62}^{ASUP},} \wedge$$

$$\wedge \xrightarrow{mp_{64}^{ASUP} = Default; mp_{14}^{ASUP} = 100; mp_{63}^{ASUP} = 708,582; mp_{71}^{ASUP} = ASUP;} (PP_{ASUP}^{master})] \qquad (1)$$

$$[(ME_{ASUP}^{master}) \xrightarrow{me_1^{ASUP} = "Open\_BD" me_{12}^{ASUP} = Close\_BD} (EA_{ASUP}^{master})]$$

The next step is to add in (1) a mathematical description of the GUI parameters of the TabControl element (CD1ASUP) according to the graph of the

implementation of the GUI "Main Control Menu" of the main window "OSCEM" (fig. 4).

$$CF_{ASUP}[(CD_1^{ASUP})(PC_1^{ASUS}) \xrightarrow{pc_2^1 = TabControl\_ASUP, pc_{14}^1, = Top; pc_{19}^1, pc_3^1, pc_{43}^1 = false;} \wedge$$

$$\wedge \xrightarrow{pc_{13}^1, pc_{27}^1, pc_{41}^1 = true; pc_{20}^1 = 81,25; pc_{51}^1 = 680,237; pc_{51}^1 = Ilist\_ASUP;} (PP_1^{ASUP})] \qquad (2)$$

Based on the proposed example of a mathematical description [13-15] and similarly (1) and (2), the developer can specify all the parameters, values and events necessary for the implementation of the analytical-logical structure of relations "OSCEM" (fig.1) for all GUI elements are presented in table 4 according to the developed graph of implementation (fig. 4).

Let's consider an example of the presentation of parameters, values and events $MP_{techn}^{directory\_CAD}, PP_{techn}^{directory\_CAD}$, for

$ME_{techn}^{directory\_CAD}, EA_{techn}^{directory\_CAD}$ creation a visual window "Technology" ($Form_{techn}^{directory\_CAD}$) and all the necessary graphic objects $CD_1^{techn}, ..., CD_x^{techn}$ for input / output and display the necessary information in accordance with table 2 and fig. 2. Expression (3) describes the parameters for creating an empty visual window "Technology" ($Form_{techn}^{directory\_CAD}$) "OSCEM".

$$Form_{techn}^{directory\_CAD} (Form_{techn} PE)[(MP_{techn}^{directory\_CAD}) \xrightarrow{mp_3^{ASUP} = Form\_ASUP; mp_4^{ASUP},} \wedge$$

$$\wedge \xrightarrow{mp_5^{ASUP}, mp_8^{ASUP}, mp_9^{ASUP}, mp_{46}^{ASUP} = true; mp_6^{ASUP}, mp_{10}^{ASUP}, mp_{12}^{ASUP}, mp_{13}^{ASUP} = false; mp_{32}^{ASUP} = ./img/techn.png;} \wedge$$

$$\wedge \xrightarrow{mp_{62}^{ASUP}, mp_{64}^{ASUP} = Default; mp_{14}^{ASUP} = 100; mp_{63}^{ASUP} = 708,582; mp_{71}^{ASUP} = Деталь;} (PP_{techn}^{directory\_CAD})] \qquad (3)$$

$$[(ME_{techn}^{directory\_CAD}) \xrightarrow{me_1^{techn} = "PostBD"} (EA_{techn}^{directory\_CAD})]$$

Element *AEofS$_j$* "General information" is a visual element which belongs to $Form_{techn}^{directory-CAD}$, and contains a set of necessary interface elements ($CD_1^{techn},…, CD_x^{techn}$)

$$CF_{techn}[(CD_1^{techn})(PC_1^{techn}) \xrightarrow{pc_2^1,pc_6^1,pc_{12}^1,pc_{14}^1,pc_{19}^1=alNone;pc_3^1,pc_{43}^1,pc_{53}^1=false;pc_7^1=Close;pc_{10}^1=crDefault;} \wedge$$

$$\wedge \xrightarrow{pc_{13}^1,pc_{27}^1,pc_{41}^1,pc_{42}^1,pc_{46}^1=-1;pc_{20}^1,pc_{35}^1,pc_{36}^1,pc_{37}^1,pc_{38}^1,pc_{39}^1,pc_{48}^1,pc_{51}^1=false;pc_{51}^1=true;pc_{15}^1=crDrag;pc_{16}^1=dkDrag;} \wedge$$

$$\wedge \xrightarrow{pc_{17}^1=dmManual;pc_{44}^1='Copy';pc_{50}^1=328;pc_{52}^1=75;pc_3^1=button\_copy}(PP_1^{techn}) \wedge$$

$$\wedge(CE_1^{techn}) \xrightarrow{ce_6^1="Copy\_to\_clipboard"}(EA_1^{techn})$$

$$\text{(4)}$$

(4) describes the parameters (visualization) and events (the reaction of the graphic object to the user's interaction with it) of reaction to a single click on the Booton interface element, which initializes the "Linguistic variable" - *Copy_to_clipboard*, in the name of which in the knowledge base (KB) the search for a code fragment to implement the functions of copying data to the clipboard is done.

Thus, we can present the structure of any visual window and all the necessary graphical elements for the implementation of a full-featured interface developed by CPPS.

for data management. Let's add (3) a mathematical description of the element (4):

To simplify the manipulation of large volumes of records and connections of all GUI elements, all the logic and sequence of constructing an example of mathematical descriptions (1-4) including: parameters, structure and events of graphical interface elements hidden from the developer and implemented at the kernel level "Automation of CPPS development management process". Consequently, (1-2) can be represented as the following record, based on the mathematical model presented below.

```
<Form_ASUP>
{?** Creating the main form with the specified parameters and events for creating and closing the form **?
[Name=Form_ASUP; ControlBox, MaximizeBox, MinimizeBox, ShowIcon, ShowInTaskbar =true; TopMost, Locked,
Localizable, AutoSize = false; Icon=../img/icon.png; Cursor, Language = Default;Opacity=100; Size=708,515; Text=
ASUP] [Load= Open_BD; FormClosed=Close_BD]
}
# TabControl_ASUP
{?** Creating a TabControl GUI with the specified parameters using the non-visual ImageList component **?
[Name=TabControl_ASUP; Alignment=top; AllowDrop, Locked, Multiline=false; Enabled, TabStop, Visible=true;
ItemSize=81,25; Size=680,237, ImageList=Ilist_ASUP]}
…
#TabPage_directory
?** Description of bookmarks "Guides" **?
{[ImegeIndex=12;Text=Guides; Size=192,67;Locked=false]}
 # ToolStrip _CAD
  ?** Explanation **?
            {[ ]}
                    # ToolStripButton _dir_CAD
?** Explanation **?
                    {[ ]  [ ]]}
}
…
#PictureBox_logo
{?** Explanation **?
[Name=PicBox_ OSCEM; BorderStyle=None, ImageLocation= ../img/logo.jpg; Location=10,98; Size= 200,200]
}
….
}
</ Form_ASUP>
```

The results of the compilation of the generated program code in Visual Studio 2019, based on the developed system "Automation of the CPPS development management process" of the main window "OSCEM", are presented in Fig. 5. As can be seen from fig. 5 this implementation of HMI on the basis of the selected GUI elements allows to implement the convenience of access to the required functionality, in accordance with the analytical and logical structure of connections "OSCEM" (fig. 1). Fig. 6 shows the use of GUI elements from table 4 to implement *Task$_j$*, *Sub_S$_k$*, *G(AEofS)$_j$* of the access levels to the functionality "OSCEM".
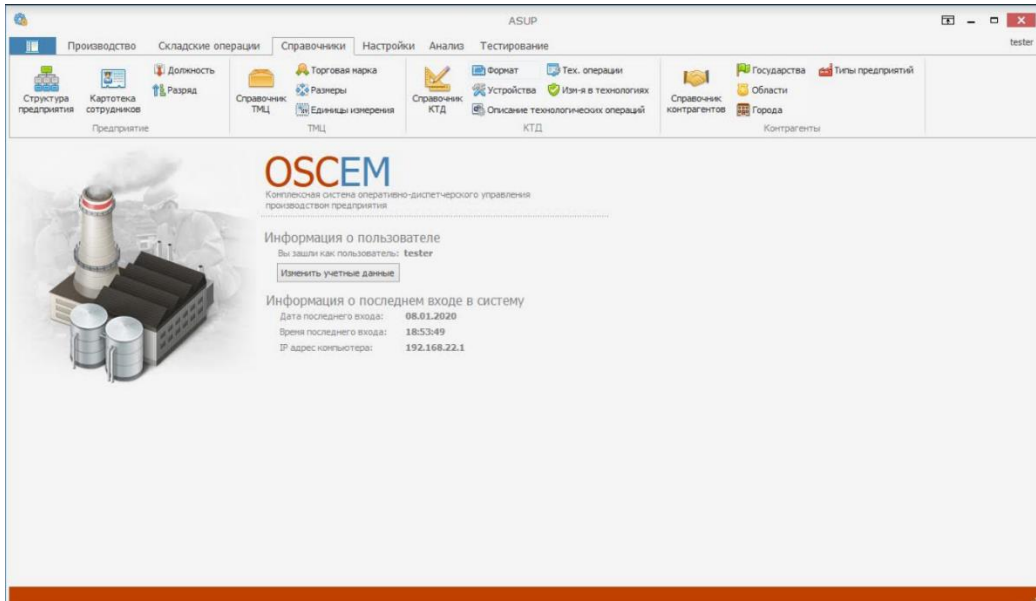
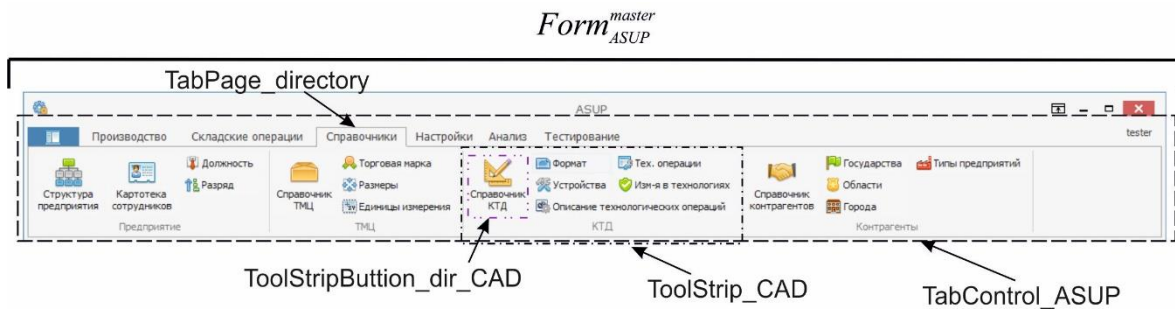**Fig. 5.** Compiled result of main window generation $Form_{ASUP}^{master}$ "OSCEM"

Fig.6. Fragment of implementation of $Task_j$, Sub_S_k, $G(AEofS)_j$ levels of access to the "OSCEM" functionality

Based on the results obtained (fig. 5) and the functionality requirements in TT, this implementation of the GUI "Main Control Menu" of the main window "OSCEM" ($Form_{ASUP}^{master}$) will be unchanged and fixed, and all other elements $AEofS_j$ will be placed on the *Panel* within the framework $Form_{ASUP}^{master}$, or called by separate Windows Form in accordance with the graphical model of the link structure (fig. 4).

Here are examples of implementing the HMI interface "OSCEM" "Guides"→"CDT Guide", based on the GUI element *Panel*, which contains all the visual elements in the framework $Form_{ASUP}^{master}$ (fig. 7).
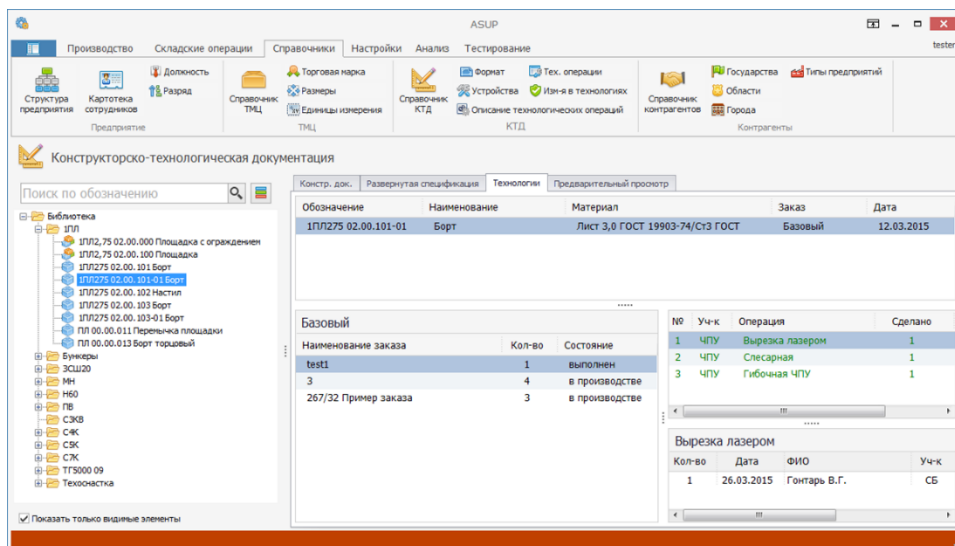

Fig. 7. Example of implementing the HMI interface "OSCEM" "Guides" → " CDT Guide"

163

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2020. № 4 (14)   *ISSN 2524-2296 (online)*

Let's give, as an example, the result of a compiled window (Windows Form) implementation of an interface based on GUI elements (fig. 4, tables 2, 4) for "Technology" ($Form_{techn}^{directory\_CAD}$). For example (fig. 8)

Windows with "CDT Guide" are called separately when the user selects TabControl "Technology" and information in DBGrid (table 4).

Fig. 8. Example of implementing the "Technology" window ($Form_{techn}^{directory\_CAD}$)

Based on the obtained scientific results, to check the correctness of the decisions made, studies of the development management process of the "Integrated System of operational dispatching control management of production of the enterprise" (OSCEM) were conducted on the basis of the system "automation of development management processes CPPS" and the classical method of the development management process, which were conducted in parallel with another group of developers, as part of the implementation of a scientific topic commissioned by the enterprise.

For clarity of the received results of the conducted experiment by classical process of management of development "OSCEM" the comparison table (table 5) on the following parameters and criteria was constructed:

- stages, sub-stages;
- time of the beginning of each substage (from the beginning of initialization of process of management of development);
- duration of each stage in days;
- the amount of work in hours, provided that the length of the working day is 8 hours.

Table 5. *Classic "OSCEM" development management process*

Project stages	Substages	Start time, days	Duration, days	Volume, hours
Initialization	General statement of project tasks	0	15	120
	Definition of requirements	0	10	80
	Definition of specification	3	15	120
	Definition of TT	4	40	320
Planning	Goal setting	15	30	240
	Clarification of results	25	34	272
	Determining the composition of works	32	40	320
	Formation of the calendar plan	40	47	376
	Risk assessment	5	70	560
Development	Configuration definition	40	64	512
	Technical ways to achieve the goal	53	20	160
Realization	Creating an interface	65	40	320
	Writing code	70	70	560
	Metric control	80	60	480
Testing	Testing	120	30	240
	Verification of compliance with TT requirements	140	34	272
	Correction of comments and errors	170	30	240
Monitoring and completion of the project	Implementation	200	23	184
	Correction of comments and errors	220	30	240
	Support of project results	240	40	320

Based on the obtained experimental data during the development of "OSCEM" by the classical method of the control process (table 5), a Gant chart was constructed, which is shown in fig. 9.
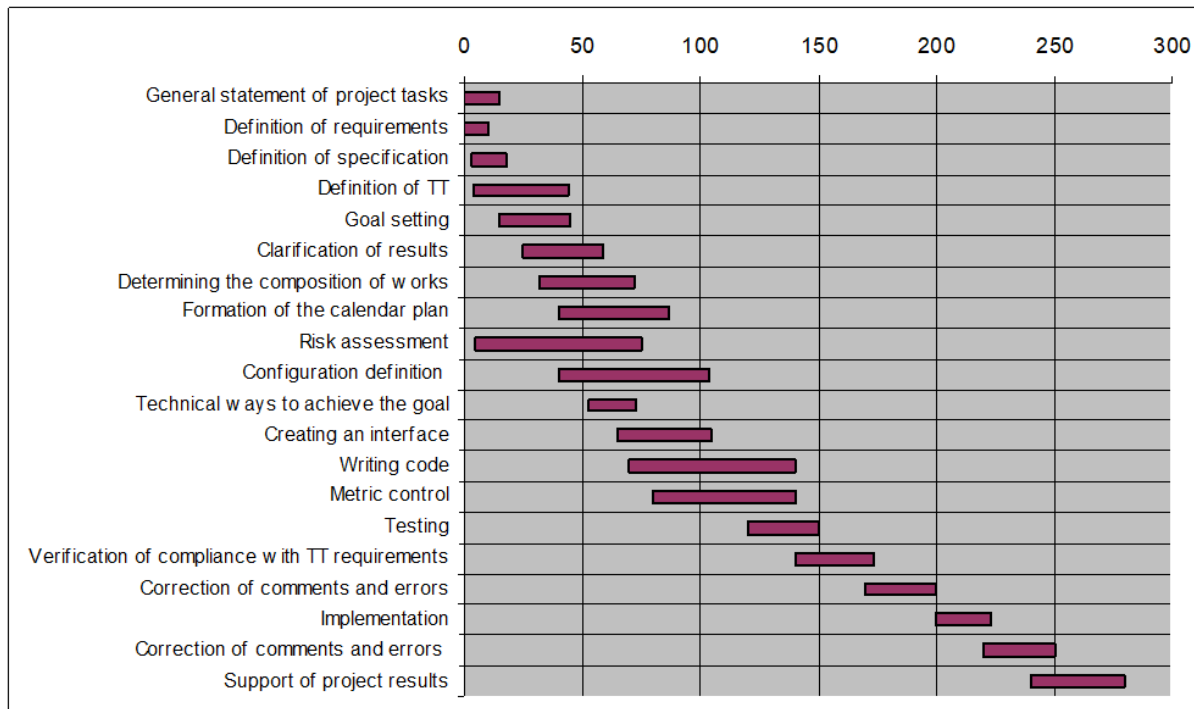

Fig. 9. The process of managing the development of "OSCEM" by the classical method

Accordingly, for the convenience of presenting the results, the process of managing the development of "Integrated system of operational and control of enterprise production" (OSCEM), based on the proposed models implemented in the system "Automation of CPPS development management processes", was built table 6 according to the following parameters and criteria:

- stages, sub-stages;

- time of the beginning of each substage (from the beginning of initialization of process of management of development);

- duration of each stage in days;

- the amount of work in hours, provided that the working day is 8 hours.

On the basis of the received experimental data (table 6) the Gant chart (fig. 10) of process of management of development "OSCEM" by means of the developed system is constructed.

Table 6. *"OSCEM" development management process based on the system "Automation of CPPS development management processes"*

Project stages	Substages	Start time, days	Duration, days	Volume, hours
Target	Setting the main goal	0	15	120
	Defining sub-goals	15	20	160
	Defining tasks for sub-goals	30	31	248
	Mathematical description of elementary problems	59	50	400
Physical	Defining the functions of each level	108	20	160
	Development of an organizational and technical structure	120	8	64
Cybernetic	Development of an infological structure	128	10	80
	Development of an information structure	138	14	112
	Development of control algorithms	149	19	152
Development	Developing a CPPS description in the modeling language	165	10	80
	Updating the code and HMI interface	173	30	240
	Checking whether the main goal has been achieved	200	5	40
Testing and monitoring	Testing	205	15	120
	Implementation and monitoring	219	10	80
	Support service	220	40	320

165

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2020. № 4 (14)   ISSN 2524-2296 (online)
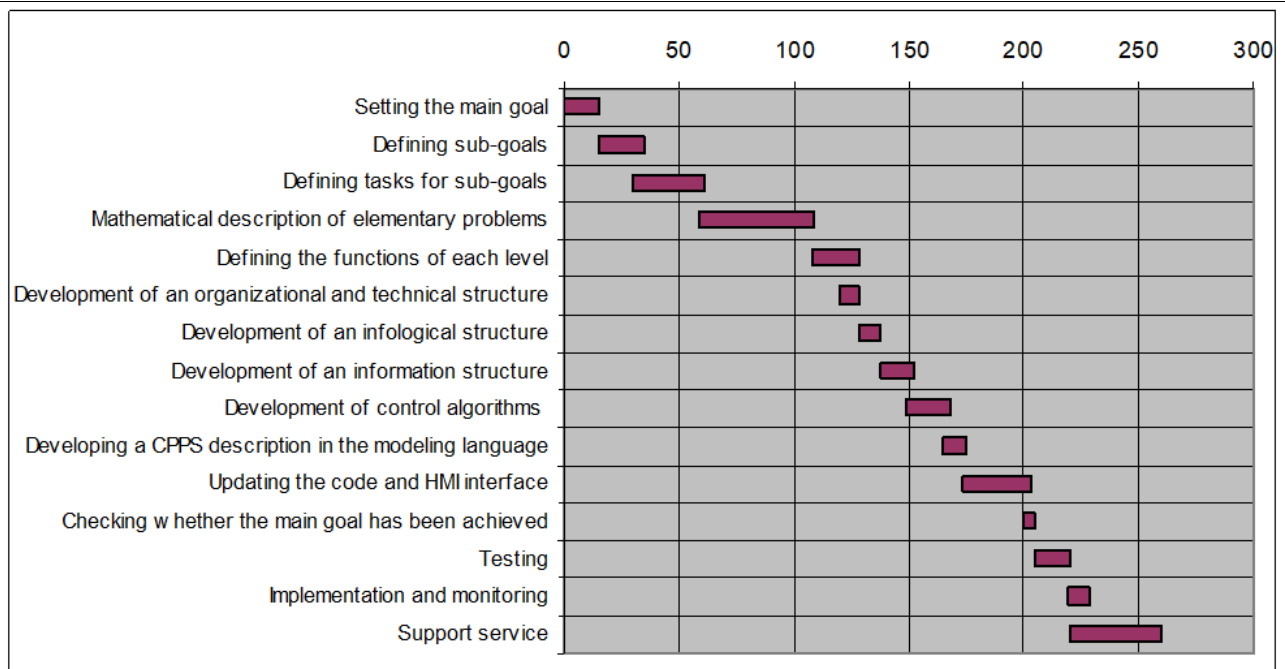
Fig. 10. "OSCEM" development management process based on the system "Automation of CPPS development management processes"

For convenience of comparison of the data received during experimental researches during OSCEM development we will construct the petal diagram in which we will designate through 1,2,…, n substages of development and volume of hours spent on each of them for the developed management process presented in fig. 11a, and the classical (fig. 11 b).

As can be seen from fig. 11 the developed automated control process, implemented in the system "Automation of control processes for the development of CPPS", has a number of advantages over the classical method of control, that is:

- the number of stages and sub-stages in the developed technological process of CPPS development management is less than in the classical method;
- the developed system "Automation of CPPS development management processes" can reach the maximum amount of time required to solve problems in the sub-stage "Mathematical description of elementary problems" - 400 hours.

We will combine the obtained experimental results for the classical and developed control method, by superimposing the obtained data in the form of a petal diagram presented in fig. 12.
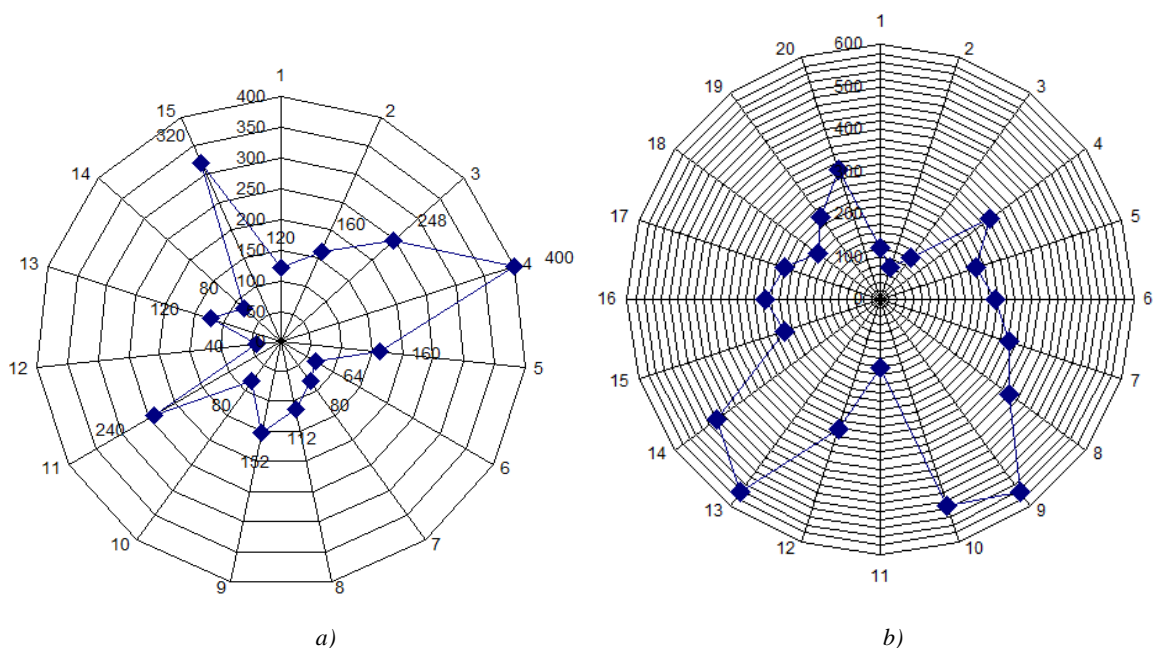

a) the developed CPPS development management process; b) the classic method of managing the development of CPPS
Fig. 11. The amount of hours spent at each sub-stage of the "OSCEM" development management process

166

ISSN 2522-9818 (print)
ISSN 2524-2296 (online)   *Innovative technologies and scientific solutions for industries. 2020. No. 4 (14)*
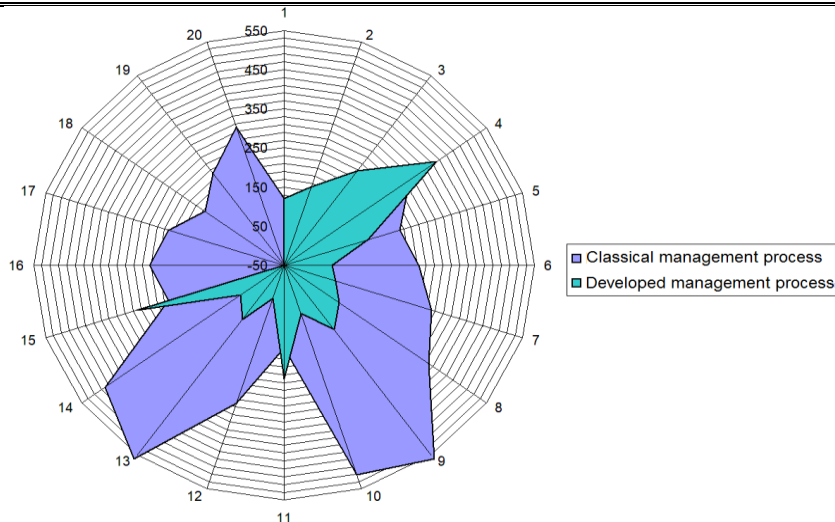
Fig. 12. Diagram of comparison of costs of volume of hours at use of classical and developed process of management of development "OSCEM"

Also, on the basis of the system "Automation of CPPS development management processes" the following solutions were developed for:

- automation of TP management in production:

1) "Voice control of the robot RM-01" (author's certificate №57666 dated 17.12.2014);

2) Programming and remote control of the mobile robot "Programming robots" (author's certificate №59439 dated 24.04.2015);

3) Automated system for designing the technological process of manufacturing accelerometers "AsSAM" ("AsSAM") (author's certificate № 65348 from 16.05.2016);

4) "Module of automated design of technological schemes of assembly of robots" Mach-CAM "" (author's certificate № 74619 from 13.11.2017);

5) "Automation of computer vision and video stream processing for mobile robots" (author's certificate № 80306 dated 16.07.18);

- automated systems of TP design and standardization: Standardization system "NORM" (author's certificate №57667 dated 17.12.2014);

- automated systems of design training: "Module of automated design of robots" Max-Robotics "(copyright certificate № 74642 from 13.11.2017);

- automated non-destructive testing systems: "Program for determining the synchronous control of temperature regimes of boards in the production of" QUAcontrol "(copyright certificate №59980 from 04.06.2015);

- software development system for computer-integrated system of technological preparation of production: "Automated software design system for corporate information systems of technological preparation of production" "CAD-Programming Code" (author's certificate № 74576 from 09.11.2017).

Conclusions

Based on the proposed models, in this work, to test and verify the correctness of research decisions, a system "Automation of CPPS development management processes" was developed. Given that the developed system allows not only to automate the process of managing the development of CPPS from scratch, but also allows you to automate the process of managing the development of the cyber component for the modernization and improvement of existing CPPS.

Introduction of technologies of automation of processes of management of development of CPPS will allow:

- to combine the target, physical and cybernetic components of the CPPS development management process in a single information space from the beginning of development to its implementation;

- automate the process of verifying the achievement of the main goal (TT requirements) of CPPS development, which allows to make changes and manage the process of CPPS development at any level and stage of the proposed technology;

- automate the process of managing the development of the cybernetic component (HMI) based on synthesized algorithms using GUI elements of object-oriented programming languages.

- increase the flexibility of the proposed architecture and technologies for automating the CPPS development process, which reduces the development time of the cybernetic component by expanding the database "ContainerSolutions", which contains ready-made program code.

During the study, a number of comparative experiments were conducted, which showed:

- time for system development, based on the system "Automation of CPPS development management processes", was reduced by 2.5 times compared to the classical method;

- the number of stages of the CPPS development management process based on the classical method is 25% more than in the proposed one;

- the amount of hours spent on the development of the system "Automation of CPPS development management processes" decreased by 2.4 times.

The obtained scientific results can be useful to developers of CPPS, software and hardware modules in the field of SCADA, ERP, MES systems, departments of ACS of enterprises for modernization of existing CPPS, as well as specialists in automation of complex CPPS development management processes.

The results of scientific and theoretical research are implemented in the educational process, in the production process of a number of public and private enterprises, as well as the stated solutions are protected by copyright certificates.

References

1. Sony, M., Naik, S. (2019), "Key ingredients for evaluating Industry 4.0 readiness for organizations: a literature review", *Benchmarking: An International Journal*, Vol. 27, No. 7, P. 2213–2232. DOI: https://doi.org/10.1108/BIJ-09-2018-0284
2. Thoben, K. D., Wiesner, S., Wuest, T. (2017), "Industrie 4.0" and smart manufacturing-a review of research issues and application examples", *International journal of automation technology*, Vol. 11, No.1, P. 4–16. DOI: https://doi.org/10.20965/ijat.2017.p0004
3. Wagner, T., Herrmann, C., Thiede, S. (2017), "Industry 4.0 impacts on lean production systems", *Procedia Cirp*, No. 63, P. 125–131. DOI: https://doi.org/10.1016/j.procir.2017.02.041
4. Perico, P., Arica, E., Powell, D. J., Gaiardelli, P. (2019), "MES as an Enabler of Lean Manufacturing", *IFAC-PapersOnLine*, Vol. 52, Issue 13, P. 48–53. DOI: https://doi.org/10.1016/j.ifacol.2019.11.306
5. Frank, A. G., Dalenogare, L. S., Ayala, N. F. (2019), "Industry 4.0 technologies: Implementation patterns in manufacturing companies", *International Journal of Production Economics*, Vol. 10, P. 15–26. DOI: https://doi.org/10.1016/j.ijpe.2019.01.004
6. Oztemel, E., Gursev, S. (2020), "Literature review of Industry 4.0 and related technologies", Journal of Intelligent Manufacturing, Vol. 1(1), P. 127–182. DOI: https://doi.org/10.1007/s10845-018-1433-8
7. Alcácer, V., Cruz-Machado, V. (2019), "Scanning the industry 4.0: A literature review on technologies for manufacturing systems", *Engineering Science and Technology, an International Journal*, Vol. 22, Issue 3, P. 899–919. DOI: https://doi.org/10.1016/j.jestch.2019.01.006
8. Castelo-Branco, I., Cruz-Jesus, F., Oliveira, T. (2019), "Assessing Industry 4.0 readiness in manufacturing: Evidence for the European Union", *Computers in Industry*, Vol. 107, P. 22–32. DOI: https://doi.org/10.1016/j.compind.2019.01.007
9. Liu, Y., Peng, Y., Wang, B., Yao, S., Liu, Z. (2017), "Review on cyber-physical systems", *IEEE/CAA Journal of Automatica Sinica*, Vol. 4, Issue 1, P. 27–40. DOI: 10.1109/JAS.2017.7510349
10. Zanero, S. (2017), "Cyber-physical systems", *Computer*, Vol. 50, Issue 4, P. 14–16. DOI: 10.1109/MC.2017.105
11. Hehenberger, P., Vogel-Heuser, B., Bradley, D., Eynard, B., Tomiyama, T., Achiche, S. (2016), "Design, modelling, simulation and integration of cyber physical systems: Methods and applications", *Computers in Industry*, Vol. 82, P. 273–289. DOI: https://doi.org/10.1016/j.compind.2016.05.006
12. Berger, C., Zipfel, A., Braunreuther, S., Reinhart, G. (2019), "Approach for an event-driven production control for cyber-physical production systems", *Procedia CIRP*, Vol. 79, P. 349–354. DOI: https://doi.org/10.1016/j.procir.2019.02.085
13. Yevsieiev, V. (2018), "Conceptual scheme and basic concepts graphic representation of software and modules visual elements description in CIS TPP design automation problem solution", *Mizhvuzivskyi zbirnyk "Naukovi notatky"*, Lutsk, No. 61. P. 40-47.
14. Nevliudov, I., Yevsieiev, V., Maksymova, S., Filippenko, I. (2020), "Development of an architectural-logical model to automate the management of the process of creating complex cyber-physical industrial systems", *Eastern-European Journal of Enterprise Technologies*, Vol. 4, No. 3 (106). P. 44–52. DOI: 10.15587/1729-4061.2020.210761
15. Yevsieiev V. (2018), "Visual objects interaction mathematical presentation to solve the problem of software design automation for computer information systems of technological production preparation", *Zbirnyk Naukovykh prats Donetskoho natsionalnoho tekhnichnoho universytetu. Seriia: "Obchysliuvalna tekhnika ta avtomatyzatsiia"*, No. 1 (31), P. 24–31. DOI: 10.31474/2075-4272-2018-1-31-24-31

Відомості про авторів / Сведения об авторах / About the Authors

Невлюдов Ігор Шакирович – доктор технічних наук, професор, Харківський національний університет радіоелектроніки, завідувач кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, Харків, Україна; email: igor.nevliudov@nure.ua; ORCID: https://orcid.org/0000-0002-9837-2309.

Невлюдов Игорь Шакирович – доктор технических наук, профессор, Харьковский национальный университет радиоэлектроники, заведующий кафедрой компьютерно-интегрированных технологий, автоматизации и мехатроники, Харьков, Украина.

Nevliudov Igor – Doctor of Sciences (Engineering), Professor, Kharkiv National University of Radio Electronics, Head of the Department of Computer-Integrated Technologies, Automation and Mechatronics, Kharkiv, Ukraine.

Євсєєв Владислав В'ячеславович – кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, Харків, Україна; e-mail: vladyslav.yevsieiev@nure.ua; ORCID ID: https://orcid.org/0000-0002-2590-7085.

Евсеев Владислав Вячеславович – кандидат технических наук, доцент, Харьковский национальный университет радиоэлектроники, профессор кафедры компьютерно-интегрированных технологий, автоматизации и мехатроники, Харьков, Украина.

Yevsieiev Vladyslav – PhD (Engineering Sciences), Associate Professor, Kharkiv National University of Radio Electronics, Professor of the Department of Computer-Integrated Technologies, Automation and Mechatronics, Kharkiv, Ukraine.

Демська Наталія Павлівна – кандидат технічних наук, Харківський національний університет радіоелектроніки, старший викладач кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, м. Харків, Україна; email: demska.nataliia@nure.ua; ORCID: https://orcid.org/0000-0002-9931-9964.

Демская Наталия Павловна – кандидат технических наук, Харьковский национальный университет радиоэлектроники, старший преподаватель кафедры компьютерно-интегрированных технологий, автоматизации и мехатроники, Харьков, Украина.

Demska Nataliia – PhD (Engineering Sciences), Kharkiv National University of Radio Electronics, Senior Lecturer of the Department of Computer-Integrated Technologies, Automation and Mechatronics, Kharkiv, Ukraine.

168

ISSN 2522-9818 (print)
ISSN 2524-2296 (online)   *Innovative technologies and scientific solutions for industries. 2020. No. 4 (14)*

Новоселов Сергій Павлович – кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, Харків, Україна; email: sergiy.novoselov@nure.ua; ORCID: https://orcid.org/0000-0002-3190-0592.

Новоселов Сергей Павлович – кандидат технических наук, доцент, Харьковский национальный университет радиоэлектроники, профессор кафедры компьютерно-интегрированных технологий, автоматизации и мехатроники, Харьков, Украина.

Novoselov Sergiy – PhD (Engineering Sciences), Associate Professor, Kharkiv National University of Radio Electronics, Professor of the Department of Computer-Integrated Technologies, Automation and Mechatronics, Kharkiv, Ukraine.

РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ОПЕРАТИВНО-ДИСПЕТЧЕРСЬКОГО КОНТРОЛЮ ВИРОБНИЦТВА НА БАЗІ КІБЕР-ФІЗИЧНИХ СИСТЕМ КЕРУВАННЯ

Предметом дослідження в статті є процеси керування виробництвом на базі кібер-фізичних систем. **Мета** роботи – автоматизація керування виробничими процесами за допомогою кібернетичної складової і аналіз отриманих результатів. Для досягнення поставленої мети необхідне вирішення наступних **завдань**: розробити аналітично-логічну модель процесу управління розробкою CPPS з урахуванням вимог зазначених в технічному завданні на розроблювальний CPPS; представити аналітично-логічну структуру зв'язків у вигляді моделі взаємодії основних вікон і всіх необхідних графічних елементів для реалізації повнофункціонального інтерфейсу розроблюваної CPPS; провести експериментальні дослідження ефективності та практичну апробацію отриманих теоретичних результатів, шляхом порівняння класичних методів та розроблених. **Результати**: Об'єднано цільові, фізичні та кібернетичні складові процесу управління розробки CPPS в єдиному інформаційному просторі від початку розробки до його впровадження. Автоматизовано процес перевірки досягнення головної мети розробки CPPS, який дає можливість вносити зміни і керувати процесом розробки CPPS на будь-якому рівні і етапі запропонованої технології. Автоматизовано процес управління розробки кібернетичної складової на базі синтезованих алгоритмів функціонування з використанням GUI елементів об'єктно-орієнтованих мов програмування, що дало можливість збільшити гнучкість запропонованої архітектури і технологій автоматизації процесу розробки CPPS та скоротити час розробки кібернетичної складової. Результати наукових і теоретичних досліджень впроваджено в освітній процес, у виробничий процес ряду державних і приватних підприємств, також викладені рішення захищені авторськими свідоцтвами. **Висновки**: запропоновані моделі та методи дозволять не тільки автоматизувати процес управління розробкою CPPS з "нуля", а й дасть можливість автоматизувати процес управління розробкою кібернетичної складової для модернізації і удосконалення вже існуючих CPPS.

Ключові слова: Industry 4.0; Розумне виробництво; Cyber-physical Production Systems; технологічний процес; аналітично-логічна структура зв'язків; управління; "OSCEM".

РАЗРАБОТКА ПРОГРАММНОГО МОДУЛЯ ОПЕРАТИВНО-ДИСПЕТЧЕРСКОГО КОНТРОЛЯ ПРОИЗВОДСТВА НА БАЗЕ КИБЕР-ФИЗИЧЕСКИХ СИСТЕМ УПРАВЛЕНИЯ

Предметом исследования в статье являются процессы управления производством на базе кибер-физических систем. **Цель** работы – автоматизация управления производственными процессами с помощью кибернетической составляющей и анализ полученных результатов. Для достижения поставленной цели необходимо решение следующих **задач**: разработать аналитически-логическую модель процесса управления разработкой CPPS с учетом требований указанных в техническом задании на разрабатываемый CPPS; представить аналитически-логическую структуру связей в виде модели взаимодействия основных окон и всех необходимых графических элементов для реализации полнофункционального интерфейса разрабатываемой CPPS; провести экспериментальные исследования эффективности и практической апробации полученных теоретических результатов, путем сравнения классических методов и разработанных. **Результаты**: Объединены целевые, физические и кибернетические составляющие процесса управления разработки CPPS в едином информационном пространстве от начала разработки до его внедрения. Автоматизирован процесс проверки достижения главной цели разработки CPPS, который дает возможность вносить изменения и управлять процессом разработки CPPS на любом уровне и этапе предлагаемой технологии. Автоматизирован процесс управления разработки кибернетической составляющей на базе синтезированных алгоритмов функционирования с использованием GUI элементов объектно-ориентированных языков программирования, что позволило увеличить гибкость предложенной архитектуры и технологий автоматизации процесса разработки CPPS и сократить время разработки кибернетической составляющей. **Выводы**: предложенные модели и методы позволят не только автоматизировать процесс управления разработкой CPPS с "нуля", но и даст возможность автоматизировать процесс управления разработкой кибернетической составляющей для модернизации и усовершенствования уже существующих CPPS.

Ключевые слова: Industry 4.0; Cyber-physical Production Systems; технологический процесс; аналитически-логическая структура свіязей; управление; "OSCEM".

Бібліографічні описи / Bibliographic descriptions

Невлюдов І. Ш., Євсєєв В. В., Демська Н. П., Новоселов С. П. Розробка програмного модуля оперативно-диспетчерського контролю виробництва на базі кібер-фізичних систем керування. *Сучасний стан наукових досліджень та технологій в промисловості*. 2020. № 4 (14). С. 155–168. DOI: https://doi.org/10.30837/ITSSI.2020.14.155

Nevliudov, I., Yevsieiev, V., Demska, N., Novoselov, S. (2020), "Development of a software module for operational dispatch control of production based on cyber-physical control systems", *Innovative Technologies and Scientific Solutions for Industries*, No. 4 (14), P. 155–168. DOI: https://doi.org/10.30837/ITSSI.2020.14.155