# What's new in MULTIPROG



Issue date: October 14, 2005



### About this Readme File

This README file provides further information about the new features in MULTIPROG.

For each feature the corresponding section of the MULTIPROG help provides further details. Please use the 'Index' of the help to find the corresponding chapters.

#### Table of contents

New features in MULTIPROG 4.6	3
New features in MULTIPROG 4.5	5
Additional modules of MULTIPROG 4.5	8
New features in MULTIPROG 4.1	9
Automation Interface Extensions	. 10
New features in MULTIPROG 4.0 and ProConOS 4.0	. 11
Additional Modules of MULTIPROG 4.0	. 13
New features in MULTIPROG 3.5 and ProConOS 3.3	. 16
New features in MULTIPROG 3.3 and ProConOS 3.3	. 21
New features in MULTIPROG 3.02	. 27
Appendix 1: ProConOS 3.2	. 42



## New features in MULTIPROG 4.6

#### **Toggle Boolean Values**

Graphical worksheets (programming languages FBD, LD and SFC) in online mode can be switched to an enhanced online mode called **'Toggle Boolean Value'** mode.

In this mode, it is possible to toggle the values of Boolean variables by simply clicking and releasing the left mouse button. Depending on the type of the toggled variable, this click causes an **overwriting or forcing** of the variable. When the left mouse button is released the variable returns to its original value.

To allow toggling for a particular variable, a specific flag must be set for this variable in the variables grid worksheet.

This mode is useful during implementation of projects since Boolean values can be operated easily and quickly direct in the online worksheet without calling the 'Debug' dialog.

#### **Network Templates**

A network template is a code body - an LD/FBD network or a code sequence written in ST or IL - which has been saved as template before and which can then be inserted into POU of the same programming language in any projects.

Within this "template code", variables, instance names, address terms and comments can be substituted by placeholders which have to be replaced by the actual terms when inserting the template code into a code worksheet.

#### **Use DIN Qualifiers in Identifiers**

According to the IEC standard, variable names can consist of letters, digits and underscores. The identifier has to begin with a letter or an underscore. The use of any other character causes the compiler error "Illegal identifier".

This naming convention has been expanded in your programming system in a way that IEC variable names can also contain DIN qualifiers:

The characters – + < > can be used at any position *inside* the name and as *last character*. However, they cannot be used as first character of a variable name.

The DIN qualifiers / \* # and the numbers 0 to 9 can be used at every position of the variable name, i.e. as first character as well as inside the name and at the end.

In order to use this extension please check the option 'Use DIN identifiers' on the tab 'Build' of the 'Options' dialog.



#### **Tooltips on Variables in ST including Online Values**

Tooltips have been added to the textual editors (IL and ST). When moving the mouse pointer onto a variable or FB instance in the text editor (without clicking the object), a tooltip appears showing properties of the element under the cursor. Tooltips are available in online mode and in offline mode. Handling and setting up tooltips is identical in both modes.

To set the content of the tooltips in code worksheets

- select 'Extras > Options',
- open the dialog page 'Tooltips',
- and mark the checkboxes of the attributes you want to see inside the tooltip.

In online mode additionally the value of the corresponding variable is displayed.

#### **Comparing the POU of two projects**

This existing feature has been extended:

- More than one difference can be detected per each POU.
- The difference can be located in the current project and also in the referenced project. Therefore each difference is described by several line in the 'Info' tab of the message window.

For details refer to MULTIPROG online help and enter 'Comparison of projects' to the 'Index' tab.



## **New features in MULTIPROG 4.5**

#### Variables Management

#### Direct Copy and Paste between Excel and Variables Grid

This item enables the easy exchange of cell contents between Excel and programming system's variables grid using the clipboard.

The variables grid supports contiguous multiple selection of cells, i.e. the selected cells must be neighbors to each other and the shape of the selection must be a rectangle.

- If the target cell contains Boolean values (e.g. cell data shall be inserted into the grid column 'RETAIN') a text "0" or an empty text indicates false (checkbox not set), all other values indicate true (checkbox set).
- If the target cell contains Integer values the text is converted to a number. If the text is empty or cannot be converted no value will be filled into that cell.
- If the target cell contains textual values, the text from the clipboard is copied. Columns containing cells with textual values are 'Name', 'Type', 'Description', 'Init' and extended properties. In case of 'Name' and 'Type' the value is limited (cut) to 30 bytes. No checks will be done regarding valid identifiers or existing data types.
- If the target cell is a cell of the column 'Usage' (list box) and the value of the clipboard is a legal keyword this keyword is used for the cell. If no legal keyword is used a default usage ('VAR' or 'VAR\_GLOBAL') will be inserted.

#### Excel-like editing operations in variables grids

The following Excel-like editing features have been implemented into the variables grid:

- Copying one cell of the Variables Grid to the clipboard and pasting this data to several variables grid cells.
- Selecting several cells in one column and editing the current (most recently selected) cell. When
  finishing the edit operation by pressing <Ctrl> + <Enter>, the content of the edited cell is copied to
  all selected cells.
- A dialog has been implemented to create several variables by one operation. In this dialog the name of the variables is entered using a name prefix, a start number and an end number. Furthermore, the data type, usage, comment, Retain-, PDD- and OPC-flag can be set; these entries are the same for all inserted variables.
- Using <F2> for in-place editing. Note: In previous versions, <F2> is used to switch on/off the display of the project tree. This command is now assigned to <Shift><F8>.



#### Updating external variables according to the global variables

This feature allows to update all external variables of all POUs used in the resource for which the update command is called. By executing the update command, the (changed) properties of the global variables will be automatically copied to the related external variables in all POUs in the specific resource.

**Example**: A global variable is used in two different function blocks (each function block has its own external declaration of this variable. The user has changed the data type of the global variable using the variable properties dialog in one function block. So the global definition and one external declaration have been changed. The second external declaration has not been changed resulting in a compile error.

**Feature description**: By executing this new command from the context menu of a resource in the project tree, the data type and comment of the global variable are updated at the external declarations of all POU, which are used in this resource. Only existing declarations are updated, no new declarations are inserted.

#### Printing

#### Printing a range of pages from a project

Similar to Microsoft Word you can select a range of pages to be printed. This way, you can print single pages, separated by semicolon, and range of pages separated by hyphen, e.g. "1;3;5-12".

#### Printing variables grid using multiple lines

When printing the variables grid and the contents of a cell cannot be printed completely into the cell area, several lines can be used for this variable. This qay, the complete content of each cell is printed. The user can switch on and off this behavior by a checking the option 'Enable word wrapping during printing' in the tab 'Variables Grid' of the 'Options' dialog.



#### **Miscellaneous**

#### PLCopen XML Import/Export

PLCopen has defined a new tool-independent XML Exchange Format. MULTIPROG 4.5 supports this exchange format for IL, ST, LD (Graphic Editor), FBD. For more information please refer to: <u>http://plcopen.org/TC6/XML\_Intro.htm</u>.

#### **Opening a zipped project**

This feature is related to opening a zipped project (\*.zwt). The procedure is:

- If a project is currently open, this project is closed first.
- After selecting a zwt file, the following message box appears:

MULTIP	ROG 🛛 🔀
1	Do you want to unzip the project to the folder containing the zwt file? Press 'Yes' to unzip the project to the same folder containing the zwt file. Press 'No' to select another folder.
	Yes No Cancel

Clicking 'Yes' unzips the project into the folder of the zwt file. Pressing 'No' opens the 'File Open' dialog for selecting another folder. Here, the default project folder, as defined in the 'Options' dialog, is proposed.

• The zipped project is unzipped and opened.

Creating a new project (from a template) and uploading a zipped project from the PLC results in "Untitled" (same as previous versions).

#### Drag & Drop in project tree

Currently, the project tree supports Cut/Copy/Paste. Additionally, Drag & Drop is implemented to move or copy objects (worksheets, POUs, resources ...) within the project tree.

#### Implicit addressing of DWORD, WORD, BYTE variables

This feature enables direct access to parts of BIT-STRING variables without using the BIT\_UTIL functions. Examples:

- myDWORDvar.x22 (Bit 22)
- myDWORDvar.b1 (Byte 1)
- mYDWORDvar.w0 (Word 0)

This feature is supported by PLC adaptations to ProConOS 3.2 (and higher).

#### **1000 Elements in one STRUCT Definition**

The number of elements per each STRUCT definition has been increased from 200 to 1000.



## Additional modules of MULTIPROG 4.5

#### Integrated Source Code Control system

This feature is an extension of MULTIPROG MultiUser module.

It is quite often useful to transfer data of a MULTIPROG project into an SCM system. This is done by using an interface which is supported by several different SCM systems.

Microsoft uses such an interface (called SCCAPI) to integrate Visual SourceSafe® with Visual Studio®.

This SCCAPI interface is also supported by PVCS Dimensions® from Merant and several other SCM programs. The existing MULTIPROG MultiUser-functionality is usable with the SCCAPI. Based on the documentation of the SCCAPI interface, the MS Visual SourceSafe® can be used to store MULTIPROG data.

For more detailed information's refer to the 'FDS integration of SCC systems in MULTIPROG'.

#### Fixed Format LD Editor

The Fixed Format LD Editor is grid based. This means objects are placed into a cell like spreadsheet. Auto-routing is implemented by which the connection lines are placed automatically by the system. Different display modes (layouts) allow displaying more or less details for each variable and consequently display smaller or larger parts of the worksheet. Comments can either be assigned to network blocks (left power rails) or positioned freely.



## **New features in MULTIPROG 4.1**

#### Enlarge %I/%Q > 64/64 k (IPC\_40)

The size of the I/O area has been increased for more than 64 kB each. This is an optional feature. If you are interested please ask for details.



### Automation Interface Extensions

The automation interface which is an optional module of MULTIPROG, has been extended by several functions which were available by DDE. For details refer to the documentation of the automation interface.

#### **Create Global Variables from Externals**

In the project tree the menu item 'Create Global Variables from Externals' has been added to the context menu. This menu item is available if a resource is selected. When executing this new feature all programs and function blocks which are used in this resource will be scanned for external variables. These externals are compared with the global variables. If not yet existing, an appropriate global variable will be generated for each external variable. For details please refer to the MULTIPROG help (page 'Creating global variables from external variables').

#### Number of recent projects increased in the 'File' menu

The number of recently used projects in the 'File' menu has been increased from 4 to 9.

Additionally the complete path of the project (selected in the menu) is displayed in the status bar.

#### Step Comment in SFC

In SFC it is possible to assign a comment to a step. In the properties dialog of the step a new button 'Comment' is available to create or edit the comment. Furthermore, it is possible to edit an existing comment by double clicking it. When moving the step the comment is also moved. The comment is displayed above the step (centered). Comment may overlap other objects.

#### IntelliSense with Data Types

In the text editors the IntelliSense feature is used to display components of a structured variable. In addition to the name of the component also its data type is displayed (same display as in the variables dialogs).



## New features in MULTIPROG 4.0 and ProConOS 4.0

#### **Download Changes (IPC\_40)**

Download Changes enables you to perform nearly any kind of changes in a project and download it while the PLC keeps running. The performed changes may affect the whole project. This feature is intended to ensure the real-time control by the PLC while the changes are downloaded.

In spite of its high requirements this feature is easy to use. The project development can be continued normally. For example, the project can be compiled at any time to check it for errors. Or, the changes may be tested with a simulation before sending them to the control system.

To send the compiled project to the running PLC, open the 'Download' dialog as usual. This dialog has been extended by the 'Download Changes' button which you just have to click instead of 'Download'.

Download	×
Project	Bootproject
Download	Download
Download Changes	Activate
Ensure real-time for Download Changes     Include Bootproject	Delete on Target
☐ Include <u>S</u> ources ☐ Include <u>OP</u> C data	J
Do <u>w</u> nload Source	
Include User-Libraries Include PageJayouts Include Backend-Code	
Delete Source on Target	Download <u>F</u> ile
Close	Help

Due to the amount of changes and a heavy CPU load it may happen that Download Changes cannot be performed and at the same time the PLC controls the process in real-time conditions. In such cases the controlled process must be changed to a state where real-time violations are uncritical. Then, Download Changes can be performed by explicitly allowing real-time violations.

The main advantages of Download Changes are:

- Any changes in code and data are allowed for several POU simultaneously including changing formal parameters of FB or functions.
- New POUs can be created and used.
- New libraries can be inserted and used.
- Task cycle times or watchdogs can be modified.
- The PLC continues controlling the process in real-time.



#### Data Type LREAL (IPC\_40)

The data type LREAL is available with ProConOS 4.0 for Intel targets. This allows high-precision calculation with decimal values.

LREAL can be used with the 'Usages' VAR, VAR\_GLOBAL, VAR\_EXTERNAL, VAR\_INPUT, VAR\_OUTPUT, VAR\_IN\_OUT. Initial values and literals (constants) are supported. This data type can be used for retentive as well as non-retentive variables. Furthermore, components of user-defined data types can use the data type LREAL.

The following IEC standard functions support the data type LREAL:

ADD, SUB, MUL, DIV, MOD, the most important type conversion functions (LREAL\_TO\_REAL, LREAL\_TO\_DINT, LREAL\_TO\_UDINT, REAL\_TO\_LREAL, DINT\_TO\_LREAL, UDINT\_TO\_LREAL), ABS, SQRT, LN, LOG, EXP, EXPT, NEG, trigonometric functions, MIN, MAX, comparison functions.

LREAL variables is can be monitored in debug mode (online) with powerflow. Overwriting is possible. LREAL variables can be included in the PDD.

Logic Analyzer and directly represented variables (%IL xxx, %QL xxx, %ML xxx) of data type LREAL are currently not supported. This also includes access to shared memory (%ML3.xxx) and system memory (%ML 1.xxx).



## Additional Modules of MULTIPROG 4.0

#### **MultiUser**

Short programming times by parallel development with help of the 'multi user' functionality.

The MultiUser feature is a powerful tool to provide safe access to project source files when several users are working on the same project at the same time. To achieve a safe and fast MultiUser development environment, the project is stored as server project on a network server PC and each user has to create a client project on his PC.

The MultiUser feature provides several functions to handle a MultiUser project in a fast and easy way, e.g. automated server and client project creation, updating the client and server projects, checking in and out source files or inserting and deleting parts of the projects.

In the client project all nodes (files) have read-only status by default. To be able to work on the MultiUuser project, the individual node(s) have to be checked out in the project tree of the client project. After this, the checked out nodes are locked in the server project, i.e. no other user has write access. Due to this, source files can not be changed by several users at the same time.

In addition, client projects can be compiled as known from single user projects and downloading and debugging is possible if the corresponding resource is checked out.



Please ask for the detailed specification of the MultiUser feature!

Example of MULTIPROG (client project) with checked out POU



## Automation Interface Toolkit enables a closer Integration of MULTIPROG into your automation software

The open architecture of MULTIPROG allows to follow a new direction in the creation of automation software. It is possible to create tools which lead to real competitive advantages in the engineering sector - beyond PLC programming itself.

#### Automation Interface guarantees consistent engineering data

Modern automation suites include software tools for nearly all projection phases and tasks: E-CAD, fieldbus/network configurator, PLC programming, SCADA, etc. They all access partly the same objects. However, there is often a lack of data exchange, i.e. a common pool which is available equally for all components. The solution for this problem is the automation interface.

- Via the automation interface, MULTIPROG<sup>®</sup> offers its data to other tools
- MULTIPROG<sup>®</sup> allows that its own data are externally created, modified or get additional attributes.
- As all essential data can be displayed in MULTIPROG<sup>®</sup>, a frequent switching among the different tools is inapplicable during PLC programming and commissioning.
- Observers guarantee data consistence with other tools.

Please ask for the detailed feature description of the Automation Interface!





#### Automation Interface Extension with MULTIPROG 4.0

The Automation Interface is available since MULTIPROG 3.02. With MULTIPROG 4.0 the Automation Interface has been extended to support the 'Extended IEC61131-3 Import and Export'.

#### Integrated Automation Software by ActiveX and Add-Ins

Different tools of different manufacturers are often used in different automation tasks. ActiveX Controls can be integrated as control bars which allow extending MULTIPROG by other automation software.

The MULTIPROG Add-In interface allows the integration of ActiveX Controls with menus and toolbars or the implementation of additional customer-specific functionalities.



Example of the Integrated SyCon.net witch is plugged in to MULTIPROG via the automation interface and Active X control with an own toolbar.



## New features in MULTIPROG 3.5 and ProConOS 3.3

#### Faster and easier use of variables in the editors

The variable dialogs have been redesigned. Now, all dialog elements are shown on a one dialog (not on various property sheets). This results in three main advantages:

- All properties can be seen at a glance, there is no need to click through various dialog tabs.
- Operation is easier and more intuitive.
- Operation is faster, as there is no need to switch between the dialog tabs.

Variable Properties			×
Name: Drain_Q Usage: VAR_GLOBAL ↓ Data Type: BOOL Initial value: //O address: %QX0.2 Description: The drain is opened		Scope C Local C Global Local ⊻ariable Groups: VARIABLES_OF_CONTAINER Global Vgriable Groups: Global Vgriable Groups: CIPC C	OK Cancel Help
	✓ OP <u>C</u>	Show all variables of worksheet	

Operations in the variable dialog:

- Insert global or external variables. The dialog automatically generates the IEC declarations.
- Auto-complete mechanism within the comboboxes 'Name', 'Usage', 'Data Type' and 'Local Variable Groups' facilitate declaration and prevent from searching in long lists.
- Variables are displayed according to the local or global scope radio button.
- The checkbox 'Show all variables of worksheet' determines whether only the selected group or all variables of a worksheet are displayed. Anyhow, this checkbox does not influence variable creation since a new variable will always be inserted into the selected variables group.
- The settings made in 'Usage' and 'Data Type' are stored when inserting a new variable. The saved settings become active when the next new variable is inserted.



- The settings of the radio button 'Scope', the combo box 'Local Variable Groups', the tree control 'Global Variable Groups' and the checkbox 'Show all variables of worksheet' are stored in the POU and used when the dialog is displayed the next time.
- The comboboxes 'Name', 'Data Type' and 'Local Variable Groups' are expanded to the height of the dialog. This way, more entries are visible without scrolling.

Variable Properties		×
Name:          Coil_out1_0       ✓         Coil_out1_0       ▲         Coil_out1_1       Coil_out1_1         Coil_out1_2       △         Coil_out1_4       △         Coil_out1_6       △         Contact_in1_0       △         Contact_in1_3       △         Contact_in1_5       ○         Contact_in1_6       ○         Contact_in1_7       ○         Contact_in1_6       ○         Contact_in1_7       ○         Contact_in1_8       ○         Contact_in1_1       ○         Contact_in1_5       ○         Contact_in1_6       ○         Contact_in1_7       ○         Container_SOK       ○         Container_States       ○         CUNNTER_1       ○         Diagnose_Kette_1       ✓	Scope Local Global Local Variable Groups: Default Global Variable Groups: Physical Hardware CIPC RES_SIM BW Global Variables Machine SFC ProConDS system va Show all variables of worksheet	OK Cancel <u>H</u> elp

#### IntelliSense for structured data types

The variables dialog now offers IntelliSense for structured data types.

Variable Properties		×
Variable Properties          Name:         [aStruct.         Usage:           bool_var (* BOOL *)          VAR          constVal (* REAL *)          Data T3          constVal (* REAL *)          LA_TY          counter (* INT *)          ILA_TY          counter (* INT *)          Initial vz          grad (* REAL *)          0 counter          (* INT *)          I/O adc          zigZag (* INT *)          Description:           LogicAnalyzer with Structure	Scope	Cancel
	Sho <u>w</u> all variables of worksheet	



#### Filtering variables according to formal parameter data type

The variables list is filtered according to the data type of a formal parameter of a function or function block.



#### New dialog for functions and function blocks

A common dialog for functions and function blocks has been implemented.

me:	Sco	ppe Local C Global	OK
	~	cal Variable Groups:	Cancel
R 🔽 🗖	BETAIN NA		Help
ta Type: DN_STRU		Show all variables of worksheet	
scription:		nction / Function Block	
	He	ight:  36	
or <u>m</u> al Parameters:	Data tupe	ight:  36	
or <u>m</u> al Parameters: Name	Data type	ight:  36	Delete
or <u>m</u> al Parameters: Name Drain Heater	Data type BOOL BOOL	ight: 36 Negated Edge	Delețe
ormal Parameters: Name Drain Heater	Data type BOOL BOOL BOOL	ight: 36	Delete
ormal Parameters: Name Drain Heater Intake Level	Data type BOOL BOOL BOOL INT	Negated Edge	Dele <u>t</u> e D <u>u</u> plicate
ormal Parameters: Name Drain Heater Hintake Level Temp	Data type BOOL BOOL INT INT	Negated Edge	Deleţe Dyplicate
ormal Parameters: Name Drain Heater Intake Level Temp State_of_container	Data type BOOL BOOL BOOL INT INT STRING	ight: 36	Delețe D <u>u</u> plicate
ormal Parameters: Name Drain Heater Intake Level Temp State_of_container Values_of_container	Data type BOOL BOOL BOOL INT INT STRING STRING STRING	ight: 36	Delețe Duplicate



#### Changing the properties of variables in the variables grid

The properties of variables marked in the variables grid can be changed in one step. For example, the 'Retain' or 'OPC' flag can be set for all selected variables.

RULTIPROG - EXAMPLE - [PRG_LDV:C_IPC	.RES_SIM]				- 🗗 X
Ele Edit View Project Build Layout Online E	⊻tras Window ?			_	8 ×
□ 🗲 🖬   <b>b, 45   3, 46   5</b> 2 2   €	R Q 🖪 🗆 🔨 🔞 🖬	19 (* 2 🕮 🗶 💠 🗊 (*	•	유 🖬 🚾 🗌 란 방 📾	1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1			190 - 18-	111	
Project Tree Window					
	Name           ELD_VAR_PCH           Contact_inf_0           Contact_inf_0           Contact_inf_3           Contact_inf_3           Contact_inf_6           Contact_inf_6           Contact_inf_6           Contact_inf_6           Contact_inf_6           Contact_inf_6           Contact_inf_6           Cot_outf_12           Cot_outf_4           Cot_outf_6	Type     Usage       Properties       Source       C       Variables Collection >>       Usage:       VAR, GLOBAL, PG       VAR, GLOBAL, PG       Data Type:       BOOL       Initial value:       C       Different Values >>       Double Source	OPC Description	Addess         Init           30x10	
	4				×
· · · · · · · · · · · · · · · · · · ·	PRG_LDV:C				
Xariable 🛆 POU/Worksheet	Access Command I/O Ad	Global Path Type In	it. Va Comment	Line/Position(X/Y)	•
Contact_in1_0 C_IPC.RES_SIM.PRG_LDV	%IX1.0	C_IPC.RES_SIM BOOL		1420	_
Contact_n1_U LADDER.LADDER	кеad -  - %IX1.0 Read -  - %IX1.0	C_IPC.RES_SIM.PRG BOOL C_IPC.RES_SIM.PRG BOOL		14/9 14/23	
Contact_in1_0 LADDER.LADDER	Read -  - %IX1.0	C_IPC.RES_SIM.PRG BOOL		14/62	
Contact_in1_0 LADDER.LADDER	Read -  - %IX1.0	C_IPC.RES_SIM.PRG BOOL		14/108	_
Contact_in1_0 LADDER.LADDER	Read -  - %IX1.0	C_IPC.RES_SIM.PRG BOOL		14/116	-
					E: >2GB
🛃 Start 🛛 😂 🙆 🚞 3 Windows	🝷 1 2 Sim32 🚽 🦉 U	nbenannt 🖻 Release note	🕅 New_feature 🕻	MULTIPROG DE 🔇	22:09

#### Pagelayouts on project level

This feature enables the use of pagelayouts also on project level. If you select project-specific pagelayouts in the dialog tab 'Pagelayouts', these settings are stored together with the project and are still available when reopening the project.

'Project'	×
Plc/Processor Language Pa	gelayouts Security
🔲 Use global pagelayout sett	ings
<u>I</u> ext:	portrait.plt
<u>G</u> raphic:	portrait.plt
Optimized Printing:	portrait.plt
Logic Analyzer:	landscap.plt
ОК	Cancel <u>Apply</u> <u>Help</u>



## New features in MULTIPROG 3.3 and ProConOS 3.3

#### Organizing POUs in groups improves project clearness

POU groups can be used in the project tree for project organization purposes. This is especially useful if many POU are used in one project.





## Exchanging POU data with external tools using the Extended POU Import/Export

Based on the PLCopen export, now graphical POUs (FBD, LD, SFC) and the Physical Hardware subtree (global variables lists) can as well be exported and imported.





#### **Display name of control bars**

To get a better overview on the visible control bars, the name of the control bar is displayed.

×	衛 Variable	△ POU/Worksheet	Access	Command	I/O Ad	Global Path	Ту 🔨
-	🖉 A_ST_S2	C_IPC.RES_SIM.Global_V				C_IPC.RES_SIM	вс —
1	CACTION_INIT	C_IPC.RES_SIM.Global_V				C_IPC.RES_SIM	BC
8	Action_INIT	FBD.FBDV				C_IPC.RES_SIM	BC
Ę.	Action_init	FBD.SIMULATION_OF_CO	Read			C_IPC.RES_SIM	BC
N SS	Action_INIT	IL.IL	Read	OR		C_IPC.RES_SIM	BC
ĕ	Action_INIT	IL.IL	Write	ST		C_IPC.RES_SIM	BC
fere	Action_INIT	IL.ILV				C_IPC.RES_SIM	BC
Ъ.	Action_INIT	LD.LD	Read	-   -		C_IPC.RES_SIM	BC
88	Action_INIT	LD.LDV				C_IPC.RES_SIM	BC 🛩
Ę	K						>

#### Copying the contents of the output window

The contents of the output window can be copied to the clipboard and pasted in any document.

#### New PDF manual available

The HTML Help system has been converted to PDF format. So, a complete PDF manual is available again.



#### Force list (ProConOS adaptation)

The force list displays all forced variables which can be reset en-bloc or individually by selecting them in the list. The list is displayed on a separate dialog tab of the 'Resource Info' dialog. During online monitoring in the programming editors, forced variables are marked by a different background color.

🚊 MULTIPROG	- EXAMPLE										_ & ×
Eile Edit View	Project Build La	yout Ogline E≥	tras <u>W</u> indow	2							
	3 A X B			- 		a a m	× 🔺 🕅			N 📰 kaz 📗 et	- 2- Au
					••••		····   •• •	1201251260			
	1   ]]Q Q	(rd) the contract of the contr	物書書	]♥■ @	[에]] 월	нн но нн Т	기요누ㅋ	나라 나라-			
								1			1
🗎 RES_SIM			<u>- 🗆 ×</u>	🔠 IL:IL - C			PRG 🗕 🗖 🔰	🖬 🖬 LADD			<b>_ _ ×</b>
M	Name C	)nline value	Type 🔺	1	FALSE	LD %IX	0.2 (**		idder r	network	
in_1	FAL	SE BC	OL	2	FALSE	AND %IX	0.3 -	000		_	
out	FAL	.SE BC	OL	3	FALSE	OR Act	ion_INIT	002	uniaci_ni_u		
out_time	0.0	DO TIN	E	14 E	FALSE	<sup>51</sup> <sup>11</sup> -	VAR				
Input_ix0_u		E BC		6	TRUE	LD Inn	ut IXO O				
Intake Q	FAI	SF BC		7		JMPC MAN	UAL				
Person DE	e em					·	1	. 🗖	Contact_in1_1	Contact_in	h1_4
Resource: RE	.5_5IM						TON*)		Contact_in1_2		
Resource P	OUs Force						r_start				
· · · ·		,		,			ON IL			• • • •	INT#3-
Variables n	iame 🖂	Instance Path		Address	Value		IL.PT	III			
PRG_LL	D.Contact_in1_0	Global variable	\$	%IX 1.0	TRUE		IL				
PRG_LL	D.Contact_in1_2	Global variable	\$	%⊠ 1.2	TRUE		IL.Q				
PRG_L	D.Contact_in1_5	Global variable	\$	%⊠ 1.5	TRUE		on_INIT	(*1	addor	notwork	· 3*)
	D.Col_out1_2	Global variable	\$	%UX 1.2	FALSE		r_start	11 ( -	auder	iletwork	· • • • • • • • • • • • • • • • • • • •
	D.Loi[out1_5	Global variable	s -	44X 1.5	TRUE		r value				
Heater	0_0	Global variable	s	202.00	TRUE			003	Contact in1 0	0.50	
	,u	Chobar Validade	4	180/10.0	more		-			RES_SIM	- <b>X</b>
								1 · · ·		State: Ru	in [Debug]
									Contact_in1_0	Stop	Cold
							1			Beset	Warm
							ing				
											Hot
							ut_1	· ·		Download	Upload
										Error	
,							uc_1			Clean	
	Reset force	1		Reset for	ce list		1			Cluse	
				Γ	ОК	Abbrechen					
For contract of the											26.108 E: >268
4 Start	10 10 A	C KW-Softwar	e + Sycon	( MILL TIPRO	S - EXAMPLE	10 Pc5m32			DRIVER	DE	21:19
Jolan		Concernation Concernation		MULTIPRO	S - CAAMPLE	Teo Acomos		IN CLINOIO			

#### User-defined data types in the Logic Analyzer

Elements of arrays or structures can now be recorded using the Logic Analyzer. When selecting an array or a structure for recording, a dialog appears. Using this dialog the element to be recorded can be selected.

Image: Second	🔁 Structure Component / Array Element Selection	×
	Image: Struct : LA_TYPE         Image: Counter : INT         Image: Simple : Simp	OK Cance



## Downloading a boot project or project archive together with the project (ProConOS adaptation)

The 'Download' dialog of the resource (invoked from the resource control dialog) has been extended by two additional checkboxes. Using these options you can additionally download the boot project and/or the zipped project when downloading the project. The setting is stored with the project at the resource. When executing the download using the 'Project Control' dialog (in case of several resources in the same project) these settings are as well available.

Download	×
Project	Bootproject
(Download	Download
✓ Include Bootproject Include Sources ✓ Include OPC data	Activate
	Dele <u>t</u> e on Target
Download Source	
✓ Include User-Libraries ✓ Include Page ayouts ✓ Include Backend-Code	
Delete Source on Target	Download <u>F</u> ile
	Help

#### Confirm when starting/stopping the PLC

This option can be defined in the 'Options' dialog. Having set the option, you have to confirm an appearing message dialog each time you start or stop the PLC.



#### **Displaying resource load (ProConOS adaptation)**

The display of the resource load has been improved. In the 'POU' tab of the resources info dialog the following features have been added:

- For each POU the number of instances is displayed.
- The size of the PDD is displayed.
- The code size of the init code is displayed also as percentage.
- The information given in this dialog can be
  - copied to the clipboard.
  - exported to a CSV file.

Name	Code	NonRetain: Data	Reserve	Retain: Data	Reserv 🔺
CON_STRU	1059	970	500	0	
CTD	273	10	0	0	
CTU	270	10	0	0	
CTUD	433	12	0	0	
FBD	1302	5598	500	0	
IL	418	1060	500	0	
INIT_STR	495	502	500	0	
LD	727	512	500	0	
Logic_Analyzer	1048	1760	500	0	
RO_BOT	359	1532	500	0	
ROBOT	604	516	500	0	
SFC	1786	694	500	0	
TLC	295	514	500	0	
TOF	392	26	0	0	
TON	370	26	0	0	
IP	402	26	U	U	
•					•
PDD Size		E	1		
Total: 102		Export to USV file			
Polotiuo: 0%			. 1		

#### PDD > 64 kB with ProConOS 3.3

The size of the PDD has been increased. This feature is available with ProConOS 3.3 and the PLCadaptation IPC\_33. Now, up to 16,384 variables can be stored at the PDD with an average name length of 63 characters.

#### **New retain handling with ProConOS 3.3**

The new retain handling ensures that the PLC can be started with a hot or a warm start after changing retain data. Up to now a cold start was necessary.



## **New features in MULTIPROG 3.02**

#### Extended online help system

Many existing parts of the MULTIPROG help system have been extended from MULTIPROG 2.1 to MULTIPROG 3.02. The MULTIPROG 3.02 help system now additionally includes parts of the MULTIPROG 2.1 manual. With MULTIPROG 3.02 the former "MULTIPROG Manual" has been replaced by the extended help system and the new "MULTIPROG Quickstart Manual".

In particular, the tutorial "7 minutes for an easier startup" has been added, containing videos (English) which support new users.

#### Table oriented variables management

To make the declaration of variables easier, the textual declaration of variables has been replaced by a table oriented variables management.





All variables are managed within a grid control which has a similar look and feel as the cross references list. Editing features are:

- easy sorting by clicking on the title bar
- grouping of the variables
- filtering
- hiding and showing columns
- cut/copy/paste
- drag & drop
- indication of multiple declarations
- extendable properties

Reference to MULTIPROG help system: Please refer to the help chapter 'Editing and developing a project/Declaring variables and instances in variables worksheets'.

#### **Pictures inside FBs in the graphical editor**

User-defined pictures inside FBs make it easier to comprehend the functionality and purpose of FBs.





Scalable enhanced meta files (\*.EMF) can be shown inside a function block or function for better documentation purposes. These pictures can be added for firmware and user blocks.

Windows Enhanced meta files (\*.EMF) are vector oriented so that they can be zoomed to any size. EMF files also give the possibility to include simple bitmaps.

A new feature in the graphical editor even allows showing the content of a FB itself inside of the green representation of the FB call.

Reference to MULTIPROG help system: Please refer to the topic 'Text editor, graphic editor and edit wizard – General description/Graphic editor – General description/General editing operations in the graphic editor/Pictures inside the graphical representation of FUs and FBs'.

#### Edit Wizard with descriptions

Descriptions in the Edit Wizard make it easier to comprehend the functionality and purpose of functions and FBs.

🐺 MULTIPI	ROG F	0X 2	.11 - m	y_proj	1 - [M	ain]																				_ E	X
Eile Edi	t <u>V</u> iew	<u>P</u> roje	ect <u>B</u> ui	ld <u>O</u> bje	ects <u>L</u>	ayout	0 <u>n</u> lin	ie <u>W</u>	indow	2																_ 6	' ×
Dpen Project	Save	,	⊕ Zoom Ir	Zoi	⊙ om Out	Proj	ect Tree	Edit	Wizard	Mess	ages	(Refer	) ences	Declara	tions	Nt:	* ike	Pat	🕍 sh POU	Debug on	/off PLI	C Contro					
Mark Mark Connect						· · · · · · · · · · · · · · · · · · ·	PLC 1								·					ik Wizard Favorites Favorites FAND FAND FAND FOTU FOTU FOTU FOTU FOTU FOTU FOTU FOTU	Addi Bitwi Cour Cour Divis Equu Divis Equu Cour Eatir Gree Gree Less Mod Assig	cription ise ANE ise ANE ther Dop ison al: = Than fi Than fi T	 on >=				
										_															10	4,69 D::	2GB

A textual description can be added to each firmware function and firmware FB.

Reference to MULTIPROG help system: Please refer to the topic 'Text editor, graphic editor and edit wizard – General description/Edit Wizard – General description' and open the context menu within the Edit Wizard.

KW-Software GmbH



#### Alignment of Power Rails

Power rails can be horizontally aligned easily.

🍘 MULTIPROG wt - Example - [Ladder:LADD	ER*]	
Eile Edit ⊻iew Project Build Objects Layo	ut O <u>n</u> line E <u>x</u> tras <u>W</u> indow <u>?</u>	<u>_8×</u>
D 📽 🖬 🖪 🎒 🕷 🛍 🛍 🗠 🕾	@ @ □ べ @ 5 = 10   0 2 2 2 2 4 0 1 1   0 =   + 5 5	
	·월송송송 · · · · · · · · · · · · · · · · ·	
		1
Project Proje	(*Ladder network 2 with a Counter FB*)	
LADDER     LADDER     LADDER     LADDER     LA     IL     IL     IL     ISFC     ROBOT     ROBOT     ILOGIC_ANALYZER     INIT_STR     Physical Hardware     I_PC:IPC_30     IDEM_SIMT     DEM_SIMT     RES_SIM : PCOS_NT     IT RES_SIMT     RES_SIMT     ID_Configuration	Сти Стерия	Gen 3 Gen 3
	TIC:TLC ECONTAINE E CON_stru:C E Ladder.LA	
		D:>2GB
🏽 🚮 Start 🛛 💋 🈂 🔍 🎽 🖤 🖆 🚽 🔯 E	plorer 🔁 Posteing 👖 PcSim32 🛛 🤬 DEMOIO 🕅 Microsoft 🔄 mwt30 🖉 🚛 MULTI	17:16

Reference to MULTIPROG help system: Please refer to the topic 'Editing and developing a project/Editing LD using the graphic editor/Aligning power rails.



#### IntelliSense in the text editor

While editing a textual program, IntelliSense makes it easier to insert variables, FBs or data types.



"IntelliSense" is a technology known from Microsoft Visual Basic.

While entering the first characters of a variable or FB instance the IntelliSense pop-up window shows all the corresponding declarations. Pressing enter is enough to insert the completed name.

In the same way, entering a '.' after the name of a STRUCT or FB instance, opens the IntelliSense pop-up window with a list of all related components or parameters.

Reference to MULTIPROG help system: Please refer to the topic 'Text editor, graphic editor and edit wizard – General description/Text editor – General description/IntelliSense function in the text editor'.



#### **Source conversion**

The source conversion functionality allows "switching" POU code between IL, FBD and LD in order to re-use applications written in other programming languages or to support a customer's preferred programming language.

🙆 MULTIPROG wt - Example 📃 🗗 🔀
<u>File Edit View Project Build Objects Layout Online Extras Window 2</u>
◨┦゠◈▏◙▯◈ ◣▯& *
1 LD Contact_in1_0 2 AND Contact_in1_1 3 AND Contact_in1_2 4 OR Contact_in1_3 5 AND Contact_in1_4 Contact_in1_4 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1 Contact_in1_1
6     ST Coll_out1_0       7     7       8     LD Contact_in1_0       9     AND Contact_in1_2       10     AND Contact_in1_3       11     AND Contact_in1_4
12       OR       Action_INIT         13       ANDN       Contact_in1_6         14       AND Contact_in1_5       Contact_in1_6         15       ST       Coil_out1_1         16       Contact_in1_6       Contact_in1_7
17       18     LD     Contact_in1_5       19     AND     Contact_in1_6       20     AND     Contact_in1_7       21     ST     Coil_out1_2
י
45,21 E: 110.4MB

Possible conversions:

- The code of a POU written in FBD can be automatically converted into an IL POU or an LD POU.
- POUs written in LD can be automatically converted into IL POUs or FBD POUs.
- POUs written in IL can be automatically converted into FBD POUs or LD POUs.

The logic and execution order of the code remain the same after source conversion.

Reference to MULTIPROG help system: Please refer to the topic 'Editing and developing a project/Editing in IL using the text editor/Source conversion for IL, FBD and LD'.

#### **Compare POUs**

The 'Compare POUs' functionality makes it easier to compare different versions of a project or to compare projects on a Laptop with a project on a file server.



Compare POUs bases on executable code level.

- Changes in comments do not influence the result
- Differences are shown in the message window
- Easily jump to a difference by double-click
- Comparison of POUs on PC and PLC possible

Reference to MULTIPROG help system: Please refer to the topic 'User interface and general project handling/Handling a project/Comparing the POUs of two projects'.



#### **Cross references**

The powerful cross reference tool has been further extended and optimized.

Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Opine Extra Window 2         Image: File Edit View Project Build Objects Layout Objects Projects Project	×1
□       □	<b>3</b>
Image: Second	3
Project         Image: State of the st	3  
Project         19         0.300         ST         Timer_value           Bibliotheken         20         20         NANUAL:           Bibliotheken         21         NANUAL:           Datentypen         23         34 LD         Temp           Date:         24         775         ADD         Level	
Bibliotheken         20           Ligg Bit UTIL         21           Detertypen         22           Detertypen         23           Detertypen         23           Ligg Bit UTIL         24           TYPES         24           Types         24           Types         27	-
Image: Site of the set of the s	
□         □	
□ □ TYPES 24 775 ADD Level	8 H -
EBD 61 600	
B TLC - 26 TRUE ST Warning	
CON_STRU 28 25/88 51 Output_1	11
(*Function block TLC is called*)	
(*The TLC Functionblock is a two level controller*)	a h
Actual value	4
	۲,
🗏 🕼 🍸 Variable 🔺 POU/Worksheet Access Command 1/0 Add Global Path Type Init. Value Comment Line/Position(X/Y)	•
To the container 35/33	
Level IL.IL Read ADD C_IPC.RES_SIM_INT 1 Level of the Container 24	
LEVEL_MAX FBD.CONTAINER_IS_CONT Read C_IPC.RES_SIM INT 800 Maximum level of the Container 0/23	
LEVEL MAX FBD.SIMULATION_OF_CON Read C_IPC.RES_SIM_INT_800 Maximum level of the Container 1/71	
CLYPELMIN PROJUCTIAINER IS CUNI Read CUPPERESSIM INI SUU Minimum level of the Container 0/19	
CLEVEL_MIN FBU.SIMULATION_OF_CON Head CL_PC.RES_SIM_INT_SUU Minimum level of the Container 1767	
1 marticle international (1)	
Naximum ILIL Bead LD INT 90 38	
Imaximum         IL.IL         Read         LD         INT         90         38           Iminimum         IL.IL         Read         LD         INT         50         36	
III Maximum         ILIL         Read         LD         INT         90         38           III Minimum         ILIL         Read         LD         INT         50         36           III Minimum         ILIL         Read         LD         INT         50         36           III COLLEVEL_ARRAY         CON_STRU.CON_STRU         Write         :=         C_IPC.RES_SIM         OPC         AUTOINSERT         19	
III Maximum         ILIL         Read         LD         INT         90         38           III Minimum         ILIL         Read         LD         INT         50         36           III Minimum         ILIL         Read         LD         INT         50         36           III DPC_LEVEL_ARRAY CON_STRU.CON_STRU         Write         :=         C_JPC.RES_SIM         OPC         AUTOINSERT         19           III Optimum_1         ILIL         Read         LD         %QW3         C_IPC.RES_SIM         INT         28	
Image: Maximum         ILIL         Read         LD         INT         90         38           Image: Minimum         ILIL         Read         LD         INT         50         36           Image: Minimum         ILIL         Read         LD         INT         50         36           Image: Minimum         ILIL         Read         LD         INT         50         36           Image: Minimum         Cliptical         Cliptical         INT         50         36           Image: Minimum         Cliptical         Cliptical         INT         50         36           Image: Minimum         Lill         Read         LD         %GW3         C_IPC:RES_SIM         NT         28           Image: Minimum         Lill         Write         ST         %GW3         C_IPC:RES_SIM         INT         30	
Image: Maximum         ILIL         Read         LD         INT         90         38           Image: Maximum         ILIL         Read         LD         INT         50         36           Image: Maximum         ILIL         Read         LD         INT         50         36           Image: Maximum         ILIL         Read         LD         INT         50         36           Image: Maximum         CON_STRU.CON_STRU         Write         :=         C_IPC.RES_SIM         OPC         AUTOINSERT         19           Image: Maximum         ILIL         Read         LD         %QW3         C_IPC.RES_SIM         INT         28           Image: Maximum         ILIL         Write         ST         %QW3         C_IPC.RES_SIM         INT         30           Image: PT_TON_IL         ILIL         Read         LD         TIME         #1s500ms         12	
III.IL         Read         LD         INT         90         38           Minimum         ILIL         Read         LD         INT         50         36           OPC_LEVEL_ARRAY         CON_STRU.CON_STRU         Write         :=         C_IPC.RES_SIM         OPC         AUTOINSERT         19           Output_1         ILIL         Read         LD         XQW3         C_IPC.RES_SIM         OPC         AUTOINSERT         28           Output_1         ILIL         Read         LD         XQW3         C_IPC.RES_SIM         INT         20           IPT_TON_IL         ILIL         Read         LD         XQW3         C_IPC.RES_SIM         INT         30           IPT_TON_IL         ILIL         Read         LD         TIME         11s00ms         12           IPT_TON_IL         ILIL         Read         LD         C_IPC.RES_SIM         INT         15000ms         12/16	-
Image: Maximum         ILIL         Read         LD         INT         90         38           Image: Maximum         ILIL         Read         LD         INT         90         36           Image: Maximum         ILIL         Read         LD         INT         50         36           Image: Maximum         ILIL         Read         LD         ZQW3         C_IPCRES_SIM         OPC         AUTOINSERT         19           Image: Maximum         ILIL         Read         LD         ZQW3         C_IPCRES_SIM         INT         28           Image: Maximum         ILIL         Write         ST         ZQW3         C_IPCRES_SIM         INT         30           Image: PTTON_L         ILIL         Write         ST         ZQW3         C_IPCRES_SIM         INT         30         12           Image: PTTON_L         ILIL         Read         C_IPCRES_SIM         INT         15000         12/16           Image: PTTON_L         ILIL         Read         C_IPCRES_SIM         INT         15000         12/16           Image: PTTON_L         ILIL         Read         C_IPCRES_SIM         INT         15000         12/16           Image: PTTON_L         Im	- GB

The extended cross references support:

- Display of unused variables
- Order of the columns can be changed via drag & drop
- New column 'Initial Value'
- New icons to indicate a filter applied and to display if the cross references are up to date.

Reference to MULTIPROG help system: Please refer to the topic 'User interface and general project handling/User interface – General description/Cross reference window'.



#### **Password protection**

A password protection has been added allowing restricted access to entire projects or parts of projects or libraries.

MULTIPROG wt - Example
File Edit View Project Build Opline Estras 2
▶ ☞ ■ ほう ★ 物能 모임 옷의 ▶ ■ ★ 및 ■ ■ ★ 및 ■ ■ ★ ■ ● ■ ₩ ■ ₩ ■ ₩ ■ ₩ ■ ₩ ■ ₩ ■ ₩ ■ ₩ ■ ₩
Project
Eigenschaften von 'RES_SIM'
Name Pic/Processor Security
Rights:
OK Abbrechen Ügemehmen Hille
E war reported and the second se
- C RES_SIM

The password protection mechanism allows to:

- lock the entire project or library
- protect single POUs against unauthorized change
- restrict actions like downloading or starting/stopping PLC

Reference to MULTIPROG help system: Please refer to the topic 'User interface and general project handling/Handling a project/Protecting a project using a password'.



#### **Dialog-based I/O configuration**

The dialog-based I/O configuration makes it easy to configure I/Os and drivers for ProConOS. Specific drivers can be easily plugged in by an open DLL interface.

🚃 170 Configuration				×	
INPUT OUTPUT VARCONF					
1/0 Group / Peard / 1/0 k	ladula	Panao	Commont		
		1 Manye 12 MiBO 12 MiB7	Comment		
	Add I/O Group			×	
	Name:		ОК		
	Iask: TASK	•	Cancel	]	
	– Logical addresses – Start address:	%IB 8	Des <u>c</u> ription		
	 Length:	8	Driver informatio	n of device IBSG	4 🗙
	End address:	%IB 15	Drivername:	INTERBUS	ОК
	Refresh	Device © Driver	Bus Terminal:		Cancel
	O man <u>u</u> al	C Memory	<u>M</u> odule Nr.:	0 *	
	Board / 10 Module:		Module ID:	0	
	Hilscher CIF INTERBUS G4 SST DRL		Communication Reference:		
	User defined			escription	
	<u>C</u> omment			Hilfe	

Reference to MULTIPROG help system: Please refer to the topic 'PLC help"/"I/O configuration'.

#### Recipes

The recipe handling allows defining parameter sets which can be downloaded to a PLC in order to reconfigure a machine or production process.



The easy-to-use recipe handling is integrated into the watch window and allows to:

- Overwrite a set of variables with self-defined values
- Read a set of values from the PLC into a recipe
- Load and save recipes from/to a file
- Edit a recipe file with EXCEL or a text editor

Reference to MULTIPROG help system: Please refer to the topic 'PLC help/Watch window/General information about recipe files'.



#### Saving Logic Analyzer values into a file

The values recorded by the Logic Analyzer can be stored to a standard CSV file for further processing with EXCEL or your own tool.



Reference to MULTIPROG help system: Please refer to the topic 'PLC help/Logic analyzer/Export logic analyzer data into a \*.csv file'.



#### Enhanced variable selection in the Logic Analyzer

Variables in the Logic Analyzer can be marked for display by setting a checkbox inside the variable list.



Reference to MULTIPROG help system: Please refer to the topic 'PLC help/Logic analyzer/Dialog connected variables'.



#### Display of ProConOS code size

The code size of POUs loaded into ProConOS can be displayed.

Name	Code	NonRetain: Data	Reserve	Retain: Data	Reserve
CON_STRU	753	250	50	0	0
CON_TEXT	1047	574	50	0	0
СТU	270	10	0	0	0
CTUD	433	12	0	0	0
DRAIN	235	58	50	0	0
FBD	1261	1218	50	0	0
IL	418	160	50	0	0
INIT_STR	495	52	50	0	0
LADDER@000	727	62	50	0	0
LOGIC_ANALYZER	474	72	50	0	0
RO_BOT	356	182	50	0	0
ROBOT	488	66	50	0	0
SFC	1742	242	50	0	0
TLC	295	64	50	0	0
TON	370	26	0	0	0
Global variables		856	50	0	0
INIT_CODE(internal)	1078				
TASK_MODULE(internal)	262				
Total	10704	3080		0	

Reference to MULTIPROG help system: Please refer to the topic 'PLC help/Configuration specific dialogs/Dialog resource *resource\_name*'.

#### Changes in the project tree

The project tree allows to insert elements via the context menu and to edit the new task type DEFAULT.

Reference to MULTIPROG help system: Please refer to the topic 'Editing and developing a project/Defining the project structure in the project tree/General features of the project tree/Inserting objects in the project tree'.

#### PLCopen X-Reusability Level

Exchange POUs between different programming packages. X-Reusability level also allows data exchange with Siemens STEP 7.

Reference to MULTIPROG help system: Please click on 'File/Export/Export IEC 61131-3 file' or on 'File/Import/Import IEC 61131-3 file'. Currently there is no help available for this MULTIPROG Add-in.

What's new in MULTIPROG



#### Keywords in lower case

IEC 61131 keywords can be typed in lowercase.

Reference to MULTIPROG help system: Please refer to the topic 'User interface and general project handling/User interface – General description/Customizing the user interface – dialog options'.

#### Optimized handling of PLC networks

One control dialog allows full control of all the PLCs in a network.



Instead of having one control dialog for each PLC in a network, the new 'Project Control' dialog allows the control of all or a set of PLCs in a network with one command. The Project Control dialog shows the state of all PLCs when they are stopped or running. Any set of PLCs may be started, stopped or a download may be started.

If a fast network like Ethernet is used to connect the PLCs, a download to all PLCs becomes much faster, because each download to each PLC is done by an own Windows-Thread inside MULTIPROG.

Reference to MULTIPROG help system: Please refer to the topics 'PLC Help/Configuration specific dialogs/Dialog Project Control' and 'PLC Help/Downloading – General description/Downloading a project/Downloading a project if two or more resources are available'.



## Appendix 1: ProConOS 3.2

Please refer to chapter 22 "ProConOS Changes List" of the ProConOS Developers Manual, since especially the API of the memory registration of shared memory, retain and flag area have been simplified.

#### **EN/ENO for INTEL32 and SH3**

Functions can be used with an EN input and an ENO output.