

S7A Driver / OPC Server V 8.60

Version History

© 2024 INCOSOL
Jürgen Stähler

1.	What is new...	5
1.1	Version 8.60, Build 108	5
1.2	Version 8.60, Build 107	5
1.3	Version 8.60, Build 106	5
1.4	Version 8.60, Build 105	5
1.5	Version 8.60, build 104	6
1.5.1	Export of a TIA Symbol Node	8
1.6	Version 8.60, build 103	9
1.6.1	Alias names for TIA symbols	10
1.6.2	Secured connection to S7-1200/1500	13
1.6.3	Alarms sent by the PLC	15
1.7	Version 8.50, build 109	16
1.7.1	New Tool Window for unresolved I/O addresses	16
1.8	Version 8.50, build 108	16
1.8.1	Automatic fail-over for (TIA) PLC connections	17
1.8.2	Extended logging and debug options	18
1.9	Version 8.50, build 106	18
1.10	Version 8.50, build 105	20
1.11	Version 8.50, build 104	20
1.12	Version 8.50, build 102	20
1.12.1	Support for iFIX 2022 (6.8)	21
1.12.2	Re-arranged grid columns on TIA symbol grid	21
1.13	Version 8.50, build 101	22
1.13.1	New Setup	22
1.13.2	Support for iFIX secure mode introduced with iFIX version 6.5	22
1.13.3	Symbol Editor with watch feature	23
1.13.4	OPC UA Server	24
1.13.5	Automatic iFIX block generation of TIA symbols	26
1.13.6	Extended data block diagnosis and client access monitoring	27
2.	Fixed defects...	28
2.1	Version 8.60, Build 108	28
2.1.1	Background Process	28
2.1.2	iFIX Interface	28
2.2	Version 8.60, Build 107	28
2.2.1	Background Process	28
2.2.2	iFIX Interface	29
2.3	Version 8.60, Build 106	29
2.3.1	Background Process	29
2.3.2	iFIX Interface	29

2.4	Version 8.60, Build 105	29
2.4.1	Background Process	29
2.4.2	Power Tool	30
2.4.3	iFIX Interface	30
2.5	Version 8.60, build 104	30
2.5.1	Background process	30
2.5.2	Power Tool	31
2.5.3	iFIX Interface	32
2.5.4	OPC UA Server	32
2.6	Version 8.60, build 103	33
2.6.1	Background process	33
2.6.2	Power Tool	33
2.7	Version 8.50, build 109	34
2.7.1	Background process	34
2.7.2	Power Tool	34
2.7.3	iFIX interface	34
2.8	Version 8.50, build 108	35
2.8.1	Background process	35
2.8.2	Power Tool	35
2.8.3	OPC DA/UA	36
2.9	Version 8.50, build 106	36
2.9.1	Program installation (Setup)	36
2.9.2	Background process	36
2.9.3	Native CIMPLICITY interface	37
2.10	Version 8.50, build 105	37
2.10.1	Power Tool	37
2.10.2	Background process	37
2.11	Version 8.50, build 104	38
2.11.1	Program installation (Setup)	38
2.11.2	Power_Tool	38
2.11.3	Native iFIX interface	39
2.11.4	Native CIMPLICITY interface	39
2.12	Version 8.50, build 102	39
2.12.1	Offset addressing via iFIX DR block on analog TIA symbols	39
2.12.2	High CPU load caused by S7ADRV process	39
2.12.3	Power Tool failed to connect to background process	39
2.13	Version 8.50, build 101	40
2.13.1	Lower case I/O addresses or OPC Item Ids	40
2.13.2	Import of version 8.10 CSV import files	40
2.13.3	Poll overrun counting	40
2.13.4	Fragmented polling on demand	40
2.13.5	Limitation of Slot number range	41

2.13.6	iFIX Register Element Offset	41
2.13.7	Control Tag addresses for symbolic (TIA) driver objects	41
2.13.8	Simulation mode	41
2.13.9	OPC UA Security Policy	41
2.13.10	Limitation of Cimplicity point name length	41
2.13.11	Software license and lower case computer name	42
2.14	Known Issues	42

List of Figures	0
------------------------	----------

1 What is new...

The following topics describe all new functions and enhancements added to the respective version and build numbers.

1.1 Version 8.60, Build 108

This chapter and the sub-chapters describe the new features of S7A Version 8.60, Build 108 in compact form.

Enhancements in V 8.60, Build 108:

- Support for Cimplicity HMI version 2024.
- If the 'Latch Data' option and the 'Suppress COMM Alarms' option are activated for an S7A data block, in the case of a communication error the data quality of the S7A data block is set to 'Bad Last Known' instead of 'Bad Comm Failure' (if the 'Latch Data' is not activated). In iFIX, the 'Bad Last Known' data quality means that the last valid value is kept, so no COMM error occurs in the iFIX alarm system.

1.2 Version 8.60, Build 107

This topic and its sub-topics briefly describes all new features added to S7A version 8.60, build 107

Build 107 does not contain any new functions, except the **release for iFIX 2024** and two bug fixes (see topic [Fixed defects in V 8.60, build 107](#)^[28]).

1.3 Version 8.60, Build 106

This topic and its sub-topics briefly describes all new features added to S7A version 8.60, build 106

Build 106 does not contain any new functions, only a few bug fixes (see topic [Fixed defects in V 8.60, build 106](#)^[29]).

1.4 Version 8.60, Build 105

This topic and its sub-topics briefly describes all new features added to S7A version 8.60, build 105

Build 105 does not contain any new functions, only a few bug fixes (see topic [Fixed defects in V 8.60, build 105](#)^[29]).

1.5 Version 8.60, build 104

This chapter and the sub-chapters describe the new features of S7A Version 8.60, Build 104 in compact form.

Major enhancements in V 8.60, Build 104:

- [Export of a TIA symbol node](#) and its sub-nodes to a CSV file.
- With symbolic addressing, the latest S7-1200/1500 firmware versions 3.0.3 (S7-1500) and 4.6.1 (S7-1200) are now supported.
- Symbolic addressing now supports TIA version V19
- Full redundancy support for the S7-1500R/H controllers, compatible with the S7-400H. The S7A software constantly checks which of the two PLC CPUs is the active one and connects to it. If the network connection to the active CPU is interrupted, the S7A software automatically switches to the backup CPU. Depending on the configuration of the S7A device, the S7A software checks the availability of the primary (active) CPU and automatically switches back as soon as the active CPU is available again. To support this functionality, a PLC function developed by INCOSOL must be called in OB1 on the S7-1500R/H. A TIA demo project for this can be requested via the INCOSOL support form.

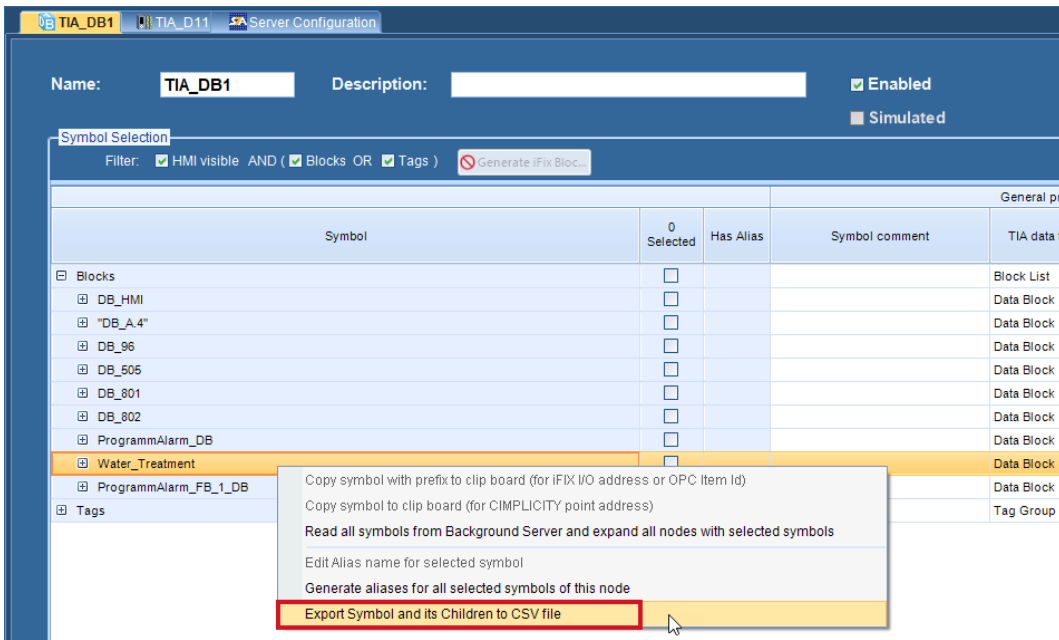
Minor enhancements and changes in V8.60, Build 104:

- In iFIX, TIA array symbols can now be specified as I/O addresses for DR/AR blocks. In combination with the auto-select option, the complete array (i.e. all array elements) is selected when importing an iFIX database for DR/AR tags that refer to TIA array symbols in the I/O addresses. If, on the other hand, the first array element of the array is specified as the I/O address, only this first array element is automatically selected.
- The shared memory footprint of the S7A background process (S7ADrv.exe) has been significantly reduced. The previous versions up to 8.60, Build 103 allocated 300 MB in the virtual address space of the S7A background process for the shared memory in the default installation. In S7A configurations with a large number of PLCs that were addressed symbolically and contained a very large number (several hundred thousand) of symbols, a memory shortage or a complete memory allocation of up to 2 GB could occur. With this improvement or reduction in memory consumption, more memory is now available for TIA symbols. The software now determines and allocates the required size of the shared memory based on the size of the loaded configuration file, including a reserve in case the configuration should be expanded.

- The input length of a physical I/O address of a symbol or of a symbol path of a TIA alias was limited to 50 characters and has been extended to 256 characters.
- Symbolic (TIA) array variables larger than 1024 bytes can now be written.
- The TIA data type ULLint is now supported.
- Improved read performance when reading TIA addresses. The number of shared memory read calls has been optimized/minimized. The access time or runtime of a read call from iFIX to the S7A driver has been reduced by 50%.

1.5.1 Export of a TIA Symbol Node

An CSV export function is now available via the context menu in the symbol tree of an S7A data block of a symbolic TIA device.



The export takes place in a file in CSV format.



In this version, the structure (the number and sequence of the exported fields) is fixed. In a later S7A version, the fields and their sequence will be freely selectable.

TreeSequenceNr	A	B	C	D	E	F	G	H	I
Symbol	Has Alias	Symbol comment	TIA data type	Path	Access	HMI accessible	HMI visible		
1	Water_Treatment	No	Data Block	Blocks.Water_Treatment	NONE	Yes	Yes		
2	Circ_Pump1_Main_Pool	No	UDT/Drive_Fl	Blocks.Water_Treatment.Circ_Pump1_Main_Pool	NONE	Yes	Yes		
3	Cmd_Man	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Cmd_Man	READ_WRITE	Yes	Yes		
4	Cmd_Off	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Cmd_Off	READ_WRITE	Yes	Yes		
5	Cmd_Auto	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Cmd_Auto	READ_WRITE	Yes	Yes		
6	Cmd_Manual_On	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Cmd_Manual_On	READ_WRITE	Yes	Yes		
7	Cmd_Manual_Off	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Cmd_Manual_Off	READ_WRITE	Yes	Yes		
8	Status_Man_Auto	No	Status Hand (1), 0, Automatic (2)	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Status_Man_Auto	READ_WRITE	Yes	Yes		
9	Status_On_Off	No	S7_Byte	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Status_On_Off	READ_WRITE	Yes	Yes		
10	RPM_Set	No	S7_Int	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.RPM_Set	READ_WRITE	Yes	Yes		
11	RPM_Act	No	S7_Int	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.RPM_Act	READ_WRITE	Yes	Yes		
12	Temp_Act	No	S7_Int	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Temp_Act	READ_WRITE	Yes	Yes		
13	Temp_Limit_Warn	No	S7_Int	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Temp_Limit_Warn	READ_WRITE	Yes	Yes		
14	Temp_Limit_Alarm	No	S7_Int	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Temp_Limit_Alarm	READ_WRITE	Yes	Yes		
15	Current_Act	No	S7_Real	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Current_Act	READ_WRITE	Yes	Yes		
16	Current_Limit_Warn	No	S7_Real	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Current_Limit_Warn	READ_WRITE	Yes	Yes		
17	Current_Limit_Alarm	No	S7_Real	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Current_Limit_Alarm	READ_WRITE	Yes	Yes		
18	Alarm_Overcurrent	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Alarm_Overcurrent	READ_WRITE	Yes	Yes		
19	Alarm_Temperature	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Alarm_Temperature	READ_WRITE	Yes	Yes		
20	Alarm_Fl	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Alarm_Fl	READ_WRITE	Yes	Yes		
21	Error_Code_Fl	No	S7_Boolean	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Error_Code_Fl	READ_WRITE	Yes	Yes		
22	Operation_Hours	No	S7_Word	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Operation_Hours	READ_WRITE	Yes	Yes		
23	Current_Act	No	UDT/Maintenance	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Current_Act	NONE	Yes	Yes		
24	Interval	No	Maintenance interval (days)	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Srv.Interval	READ_WRITE	Yes	Yes		
25	Last	No	Date of last maintenance	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Srv.Last	READ_WRITE	Yes	Yes		
26	Next	No	Next maintenance date	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Srv.Next	READ_WRITE	Yes	Yes		
27	ID	No	ID of the service technician who carried out last maintenance	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Srv.ID	READ_WRITE	Yes	Yes		
28	Fl	No	UDT/Drive_Fl	Blocks.Water_Treatment.Circ_Pump1_Main_Pool.Fl	NONE	Yes	Yes		

1.6 Version 8.60, build 103

This topic and its sub-topics briefly describes all new features added to S7A version 8.50, build 109

Major extensions in V 8.60, Build 103:

- Release for Windows 11
- Release for Cimplicity HMI 2023 (12.0)
- [Alias names for TIA Symbol paths](#)^[10]
- [Secured \(encrypted\) connections](#)^[13] to S7-1200/1500 for symbolic addressing mode
- The S7A now can receive [program alarms from TIA PLCs](#)^[15]

Minor extensions and changes in V8.60, Build 103:

- Symbol comments for array element of a TIA array symbol are now displayed
- Support for TIA data type 'S7_WString'
- The length of the channel and device names has been increased from 12 to 20 characters and aligned with the length of the data block names, which already allowed 20 characters length.
- The version is prepared for the use of loadable iFIX blocks, which can be created by the user with the soon to be released 'Block Generation Wizard'. This wizard automatically generates loadable iFIX blocks from TIA PLC data types.

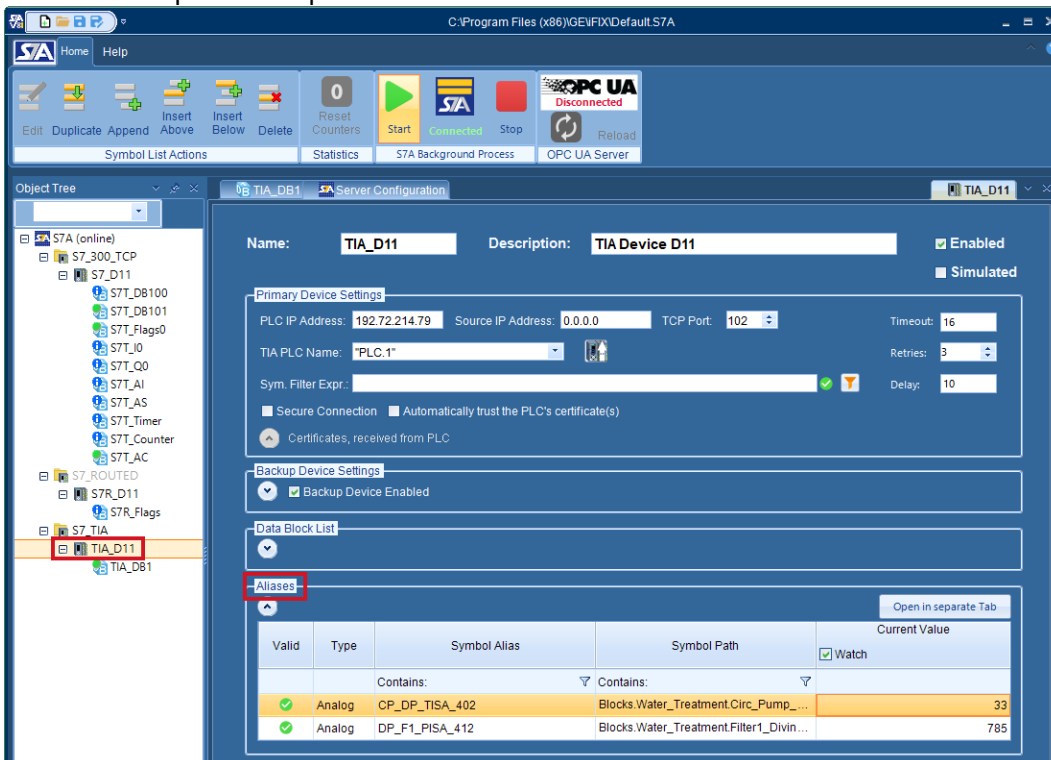
1.6.1 Alias names for TIA symbols

Version 8.60 of the S7A software now allows the definition of alias names for symbol paths. The use of alias names instead of symbol paths may be necessary or useful if the symbol paths become very long and thus difficult to read due to a large nesting depth. The use of short 'speaking' alias names instead of long symbol paths can increase and/or clarify the function and meaning of the symbols on which the alias names are based.

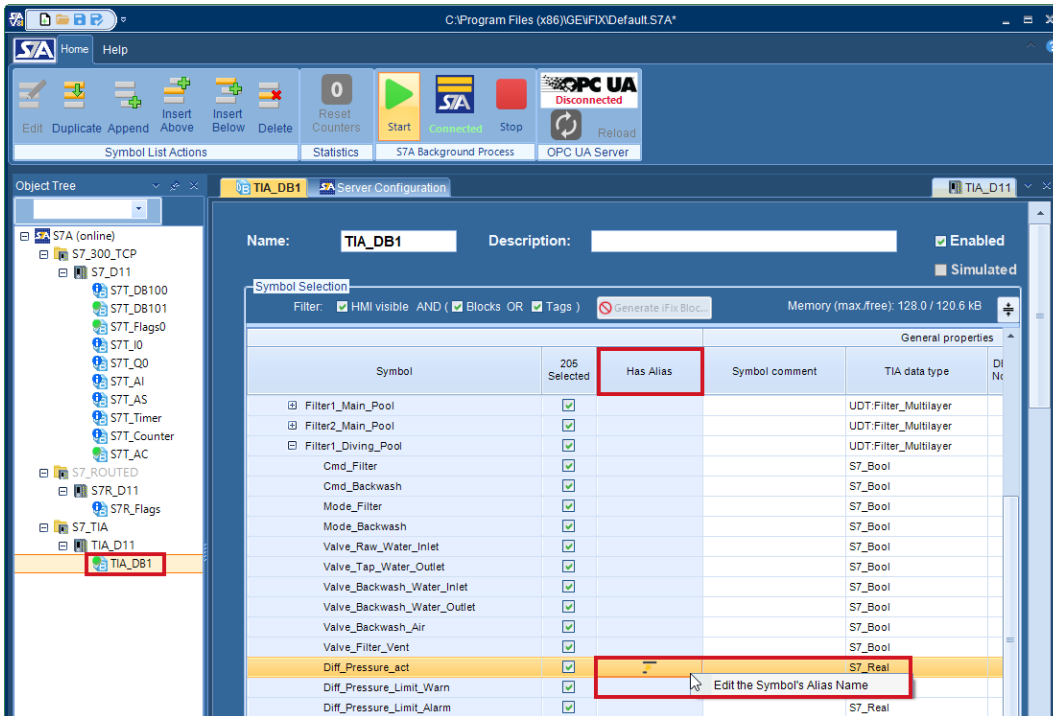
Alias names can be used in the client applications (iFIX, OPC DA and OPC UA) instead of the regular symbol paths.


Alias names can be created in two ways in the Power Tool:

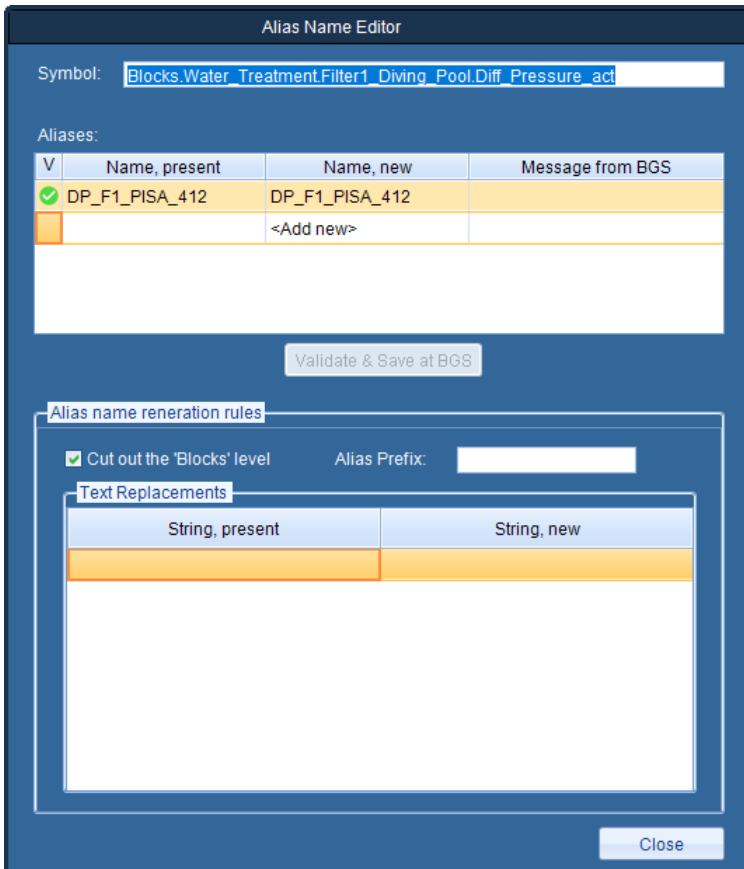
1. Via the expandable panel 'Aliases' on the TIA device level:



2. In the data block's symbol tree:



The new column 'Alias def.' indicates via the symbol  that an alias name is defined for the symbol path. You can open the dialogue for editing the symbol name via the context menu.



In this dialog you can enter an alias name individually or have it generated automatically. With automatic generation you can, for example, remove irrelevant symbol path elements (e.g. the group name 'Blocks') or replace long symbol path elements with shorter ones. In addition, you can define a prefix to be placed in front of the alias name.

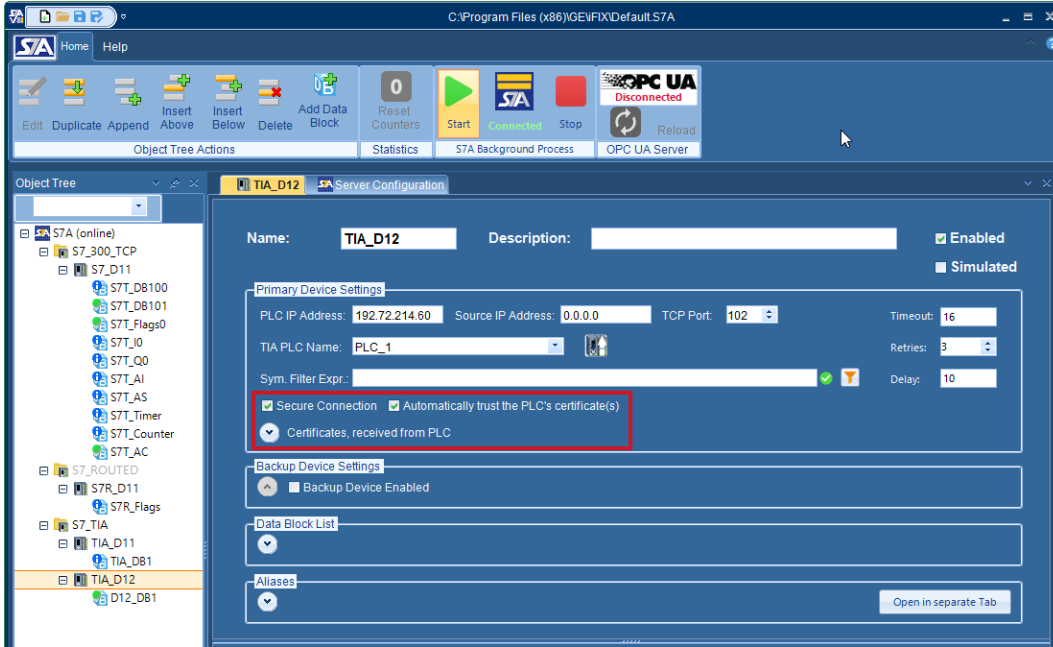
The alias names are not stored in the binary S7A configuration file, but in a separate symbol file that follows an XML syntax. This symbol file has the file extension 'S7AS' and is located in the same directory as the S7A configuration file. The very simple XML format of the symbol file also allows you to create it 'manually', i.e. to create the alias names outside the S7A-Power Tool dialog.

The following example shows the rather simple structure of the symbol file:

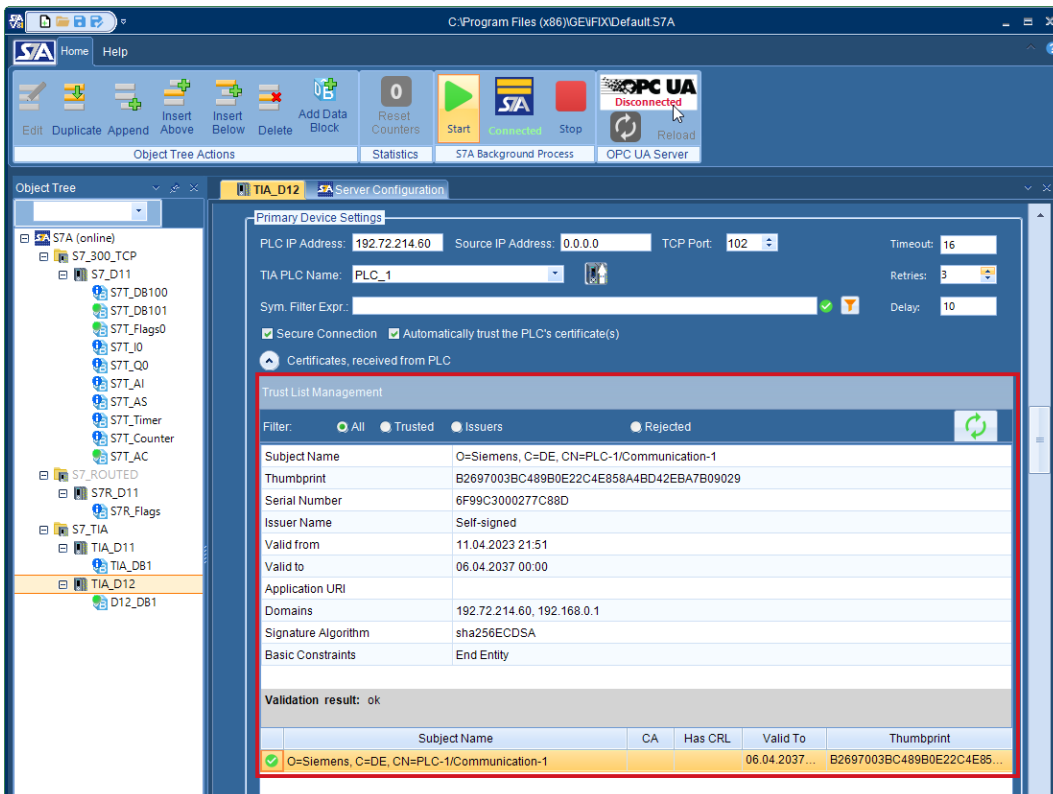
```
<DRIVER>
<ITEM Symbol="Global_Symbol_1" PhysAddress="S7_D11:DB100.DBW 0"/>
<CHANNEL name="S7_300_TCP">
<DEVICE name="S7_D11">
<ITEM Symbol="Device_Symbol_1" PhysAddress="DB100.DBW 0"/>
</DEVICE>
</CHANNEL>
<CHANNEL name="S7_ROUTED">
<DEVICE name="S7R_D11"/>
</CHANNEL>
<CHANNEL name="S7_TIA">
<DEVICE name="TIA_D11">
<ITEM Alias="CP_DP_TISA_402"
SymbolPath="Blocks.Water_Treatment.Circ_Pump_Diving_Pool.Temp_Act"
/>
<ITEM Alias="DP_F1_PISA_412"
SymbolPath="Blocks.Water_Treatment.Filter1_Diving_Pool.Diff_Pressu
re_act"/>
</DEVICE>
</CHANNEL>
</DRIVER>
```

1.6.2 Secured connection to S7-1200/1500

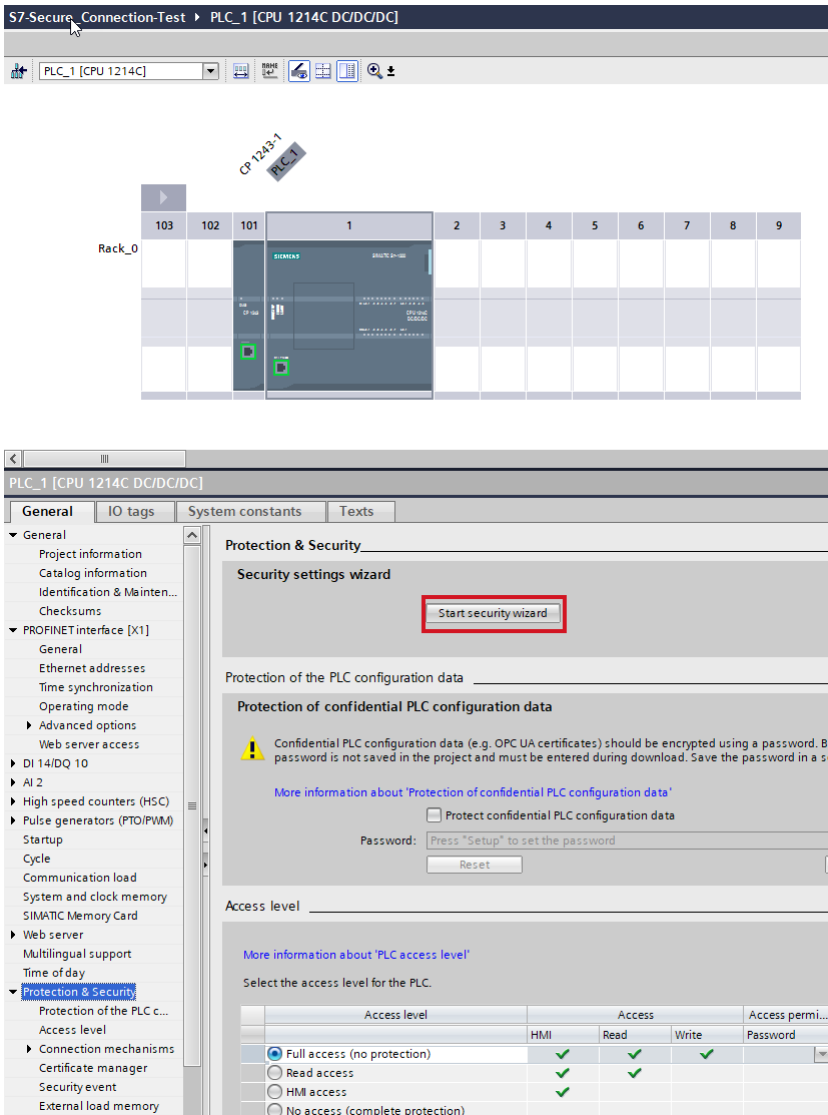
The secure (encrypted) communication via the TSL protocol available from TIA version 17 can now also be used for the symbolic communication to S7-1200/1500. For this purpose, the check boxes 'Secure connection' and 'Automatically trust the PLC's certificate' have been added to the parameter group 'Primary Device Settings' in the configuration dialog for the device:



In the expandable panel 'Certificates received from PLC', you can view detailed information about the certificates received by the PLC.



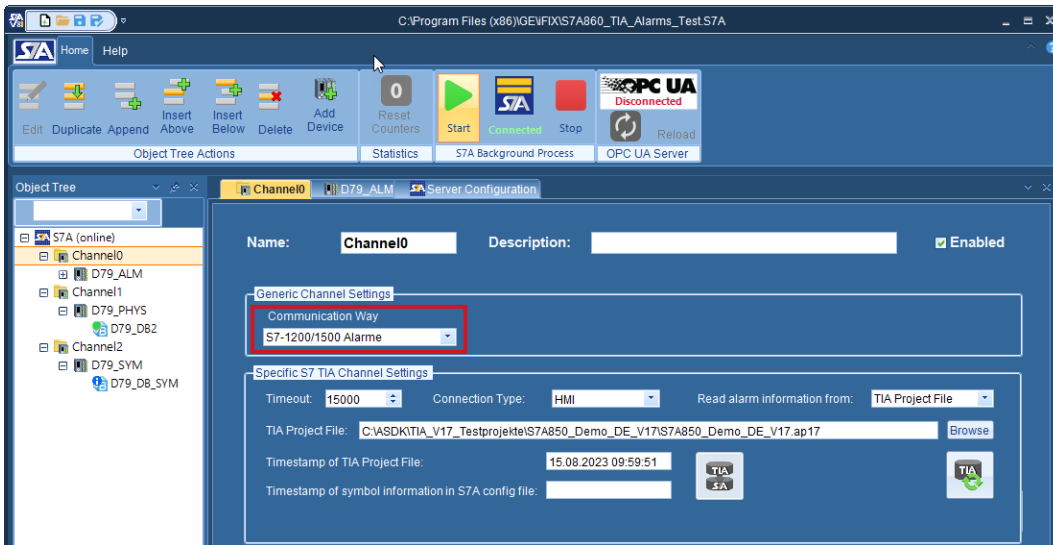
Please note that secure communication must be explicitly activated on the PLC side. The TIA Portal offers the 'Security Wizard' in the 'Protection & Security' settings for the PLC CPU for this purpose:



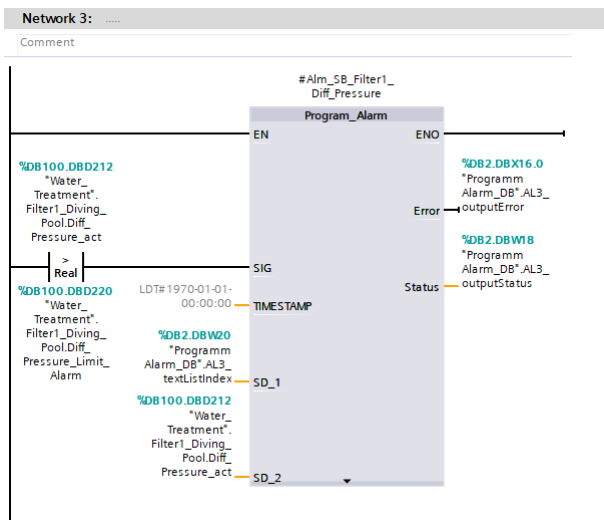
The S7A driver can only establish a secure connection once the PLC has been downloaded with the certificate required for encryption via the TIA Portal.

1.6.3 Alarms sent by the PLC

Digital program alarms and their associated values sent by an S7-1500 via the function block 'Program_Alarm' can now be received by the S7A driver and are available to the clients (iFIX, OPC) for processing. The new communication path 'S7-1500 TCP/IP Alarms' has been added for this purpose:



The following screen shot shows an example of a 'Program_Alarm' call in the PLC program:



This new S7A function also enables the alarm texts defined in the TIA project (from PLC alarm text lists), which are also transmitted via an alarm associated value (parameters SD_1 to SD_10), to be made available to the S7A clients. Combined with the INCOSOL DAX block (extended digital alarm block for iFIX) the PLC message texts can then be sent directly as alarm texts to the iFIX alarm system.

An application note describing the function and configuration of this new feature in detail will be released soon.

1.7 Version 8.50, build 109

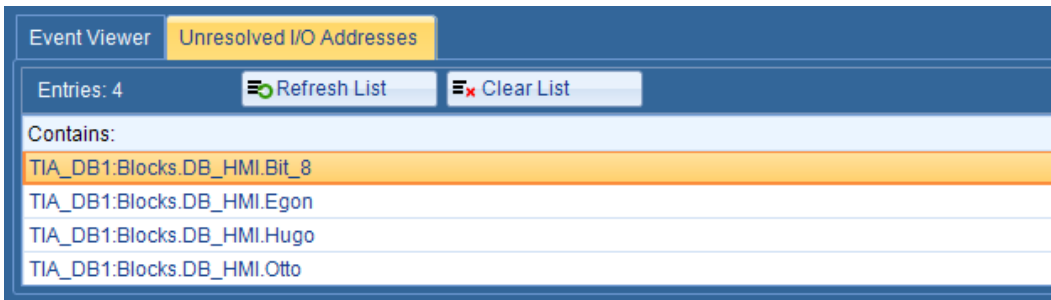
This topic and its sub-topics briefly describes all new features added to S7A version 8.50, build 109

The main new features are:

- Release for iFIX 2023 (7.0)
- Release for use under Windows Server 2022
- The Tool Windows Area has been extended by a new windows which shows all unresolved I/O addresses which are not covered by a S7A data block. See the sub-topic next to this topic for detailed information.

All bug fixes are described in topic [Fixed defects in version 8.50, build 109](#)³⁴.

1.7.1 New Tool Window for unresolved I/O addresses



This new window shows all iFIX I/O addresses and OPC Item Ids which could not be resolved by the background server. An unresolved I/O address means that

- the physical I/O address is not covered by a data block since the data block is missing or too short.
- the symbolic (TIA or SIMOTION) I/O address (the symbol) is unknown (not loaded from the project data or the PLC) or not selected in the data block's symbol tree.

In this first version the list is not automatically updated yet. To refresh or delete the list you have to click on the respective 'Refresh List' and 'Clear List' buttons.

1.8 Version 8.50, build 108

This topic and its sub-topics briefly describes all new features added to S7A version 8.50, build 108

The main new features are:

- Automatic fail-over for symbolic (TIA) PLC connections is now implemented.
- The logging has been extended by detailed debug flag options.

All bug fixes are described in topic [Fixed defects in V 8.50, build 108](#)³⁵.

1.8.1 Automatic fail-over for (TIA) PLC connections

With build 108 the automatic fail-over from the primary device IP address to the back device IP address is now available for symbolic TIA) devices.

The following screen shot shows the device settings for the primary and the backup IP connection:



When a backup device IP address is configured and the **Backup Device Enabled** check box is checked, the driver automatically switches the IP connection to this backup address in case that the primary connection becomes disturbed.

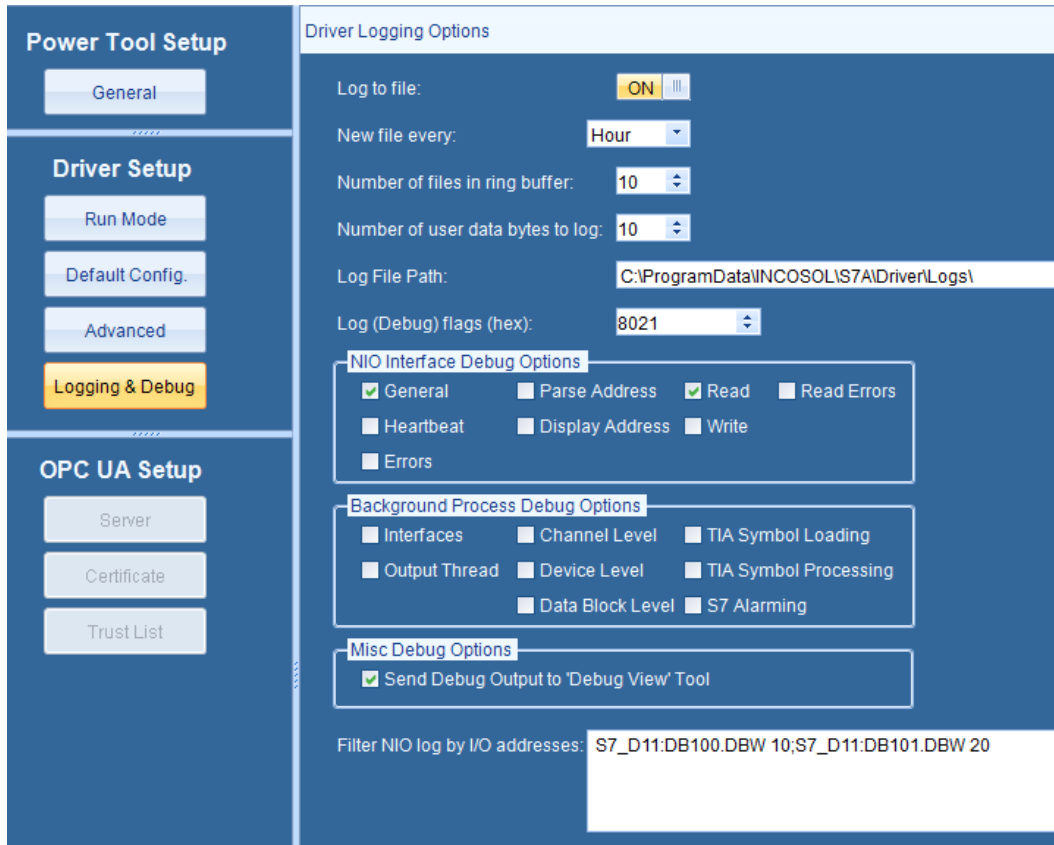
If the **Automatically switch back to primary device** check box is activated, the driver cyclically checks (via Ping) the availability of the primary connection and once the PLC becomes reachable again by this primary IP address the driver automatically changes back the communication to the primary IP address.

The currently active connection (primary or backup) can be retrieved by an analog tag (AR or AI) with the so-called Control Tag Address **!Switch:<Device name>** whereby <Device name> is the logical name of the S7A device. For the above screen shot the correct Control Tag Address is **!Switch:TIA_D11**. Returned values are 0 (zero) for primary and 1 for the backup connection. By means of writing these two values such a Control Tag also can be used to control the active connection. Note: This manual switching just makes sense when the **Automatically switch back to primary device** is deactivated!

1.8.2 Extended logging and debug options

Especially to get detailed information about data traffic in the interface between the client processes (iFIX WSACTask, TCPTask, Workspace etc.) and the S7A background process and its shared process data memory, the logging abilities have been extended.

The following screen shot shows the new/extended dialog in the S7A Power Tool's backstage view:



1.9 Version 8.50, build 106

This topic and its sub-topics briefly describes all new features added to S7A version 8.50, build 106

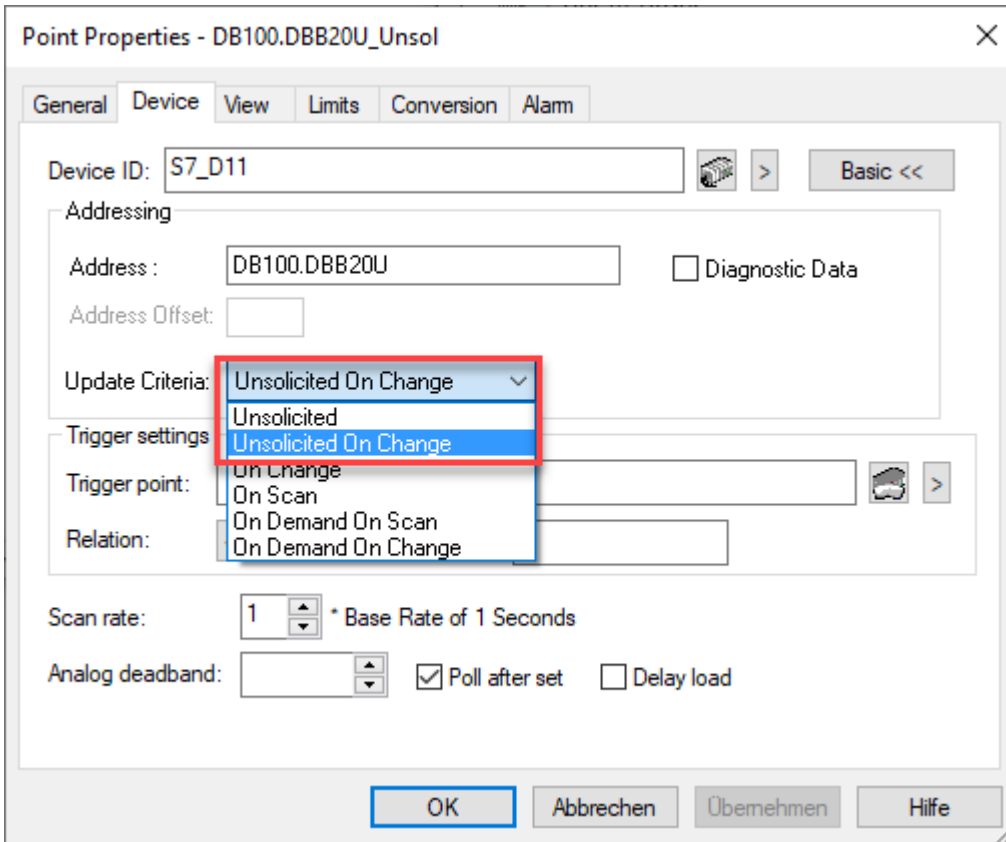
The build 106 does not contain new main features but the following minor changes/improvements and some bug fixes (see topic [Fixed defects in V 8.50, build 106](#)⁽³⁶⁾).

Minor changes and improvements:

1. To achieve compatibility with the old S7M driver two new signal conditioning function have been added. The BWRD function reads the two bytes starting at the iFIX tag's I/O address and provides an unsigned 16 bit value in the range between 0 and 65535. The BINT function reads the two bytes starting at the iFIX

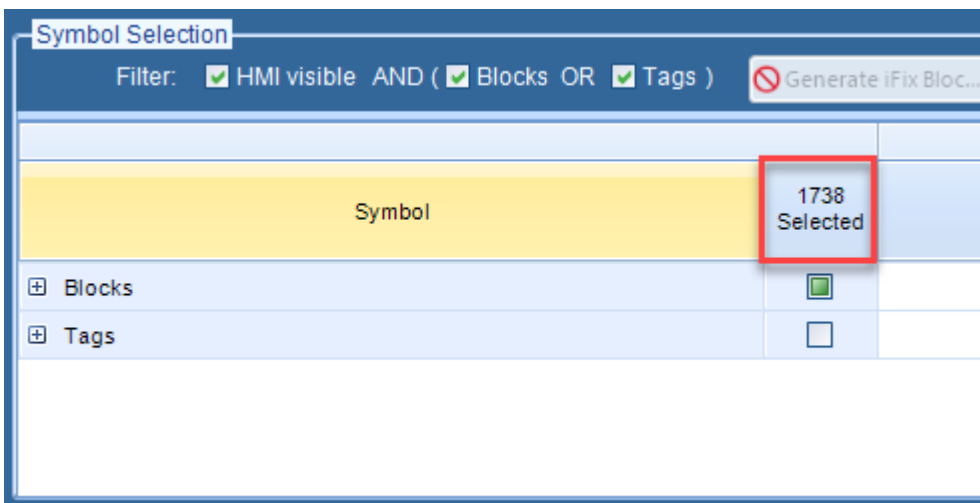
tag's I/O address and provides a signed 16 bit value in the range between -32768 and 32767.

2. Das Cimplicity Interface now supports the Point Update Criteria 'Unsolicited' and 'Unsolicited On Change'.



3. The shared memory size of a S7A data block has been increased from 64k byte to 128k byte. As a result, the number of selectable TIA symbols in a S7A data block now is twice as many as in former versions.

4. The 'Selected' header text of the TIA symbol grid now shows the number of selected symbols:



1.10 Version 8.50, build 105

The build 105 does not contain new main features but some minor changes/improvements and bug fixes (see topic [Fixed defects in V 8.50, build 105](#)^[37]).

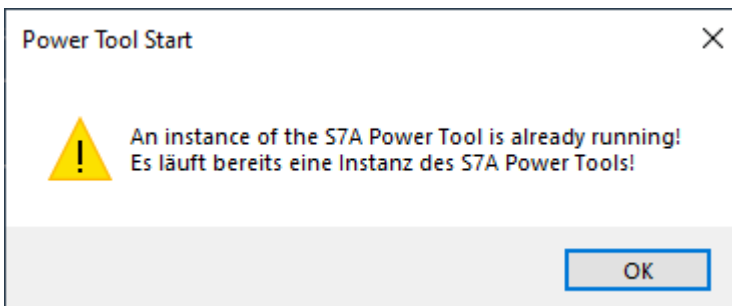
1.11 Version 8.50, build 104

This topic and its sub-topics briefly describes all new features added to S7A version 8.50, build 104

The build 104 does not contain new main features but the following minor changes/improvements and some bug fixes (see topic [Fixed defects in V 8.50, build 104](#)^[38]).

Minor changes and improvements:

- When the S7A Power Tool is started while another instance of the program is already running, the following message box will be displayed:



Since at this stage the specific language information is not yet available the message is displayed in English and German language.

1.12 Version 8.50, build 102

This topic and its sub-topics briefly describes all new features added to S7A version 8.50, build 102

The build 102 does not contain new main features but the following minor changes/improvements and some bug fixes (see topic [Fixed defects in V 8.50, build 102](#)^[39]).

Minor changes and improvements:

- Support for [iFIX 2022](#)^[21] (6.8)
- [Re-arranged grid columns](#)^[21] in TIA symbol grid.

1.12.1 Support for iFIX 2022 (6.8)

The S7A driver now supports the new iFIX version 2022 respectively 6.8 in both modes, means with or without access controls.

1.12.2 Re-arranged grid columns on TIA symbol grid

The columns of the TIA symbol grid have been re-arranged / re-sorted. Furthermore the two left-most columns 'Symbol' and 'Selected' are now 'pinned', means they are anchored to the left side of the grid. When the grid is scrolled horizontally these two columns will not be scrolled.

The screenshot shows a 'Symbol Selection' window with a table of symbols. The 'Symbol' and 'Selected' columns are highlighted with a red box and labeled 'Pinned columns'. The table includes columns for Symbol, Selected, Symbol comment, TIA data type, DB No., Access, HMI accessibility, HMI visible, and Current value (with a Display checkbox).

Symbol	Selected	Symbol comment	General properties				Current value	
			TIA data type	DB No.	Access	HMI accessibility	HMI visible	<input type="checkbox"/> Display
Blocks	<input checked="" type="checkbox"/>		Block List					
DB_HMI	<input checked="" type="checkbox"/>		Data Block	1				
Bit_0	<input checked="" type="checkbox"/>	Single Bit 0	S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Bit_1	<input checked="" type="checkbox"/>		S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Bit_2	<input checked="" type="checkbox"/>		S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Bit_3	<input checked="" type="checkbox"/>		S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Bit_4	<input checked="" type="checkbox"/>		S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Bit_5	<input checked="" type="checkbox"/>		S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Bit_6	<input checked="" type="checkbox"/>		S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Bit_7	<input checked="" type="checkbox"/>	Single Bit 0	S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
TenSignals	<input checked="" type="checkbox"/>	Ten Bits in an array	ARRAY [0..9] of S7_Bool		RW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

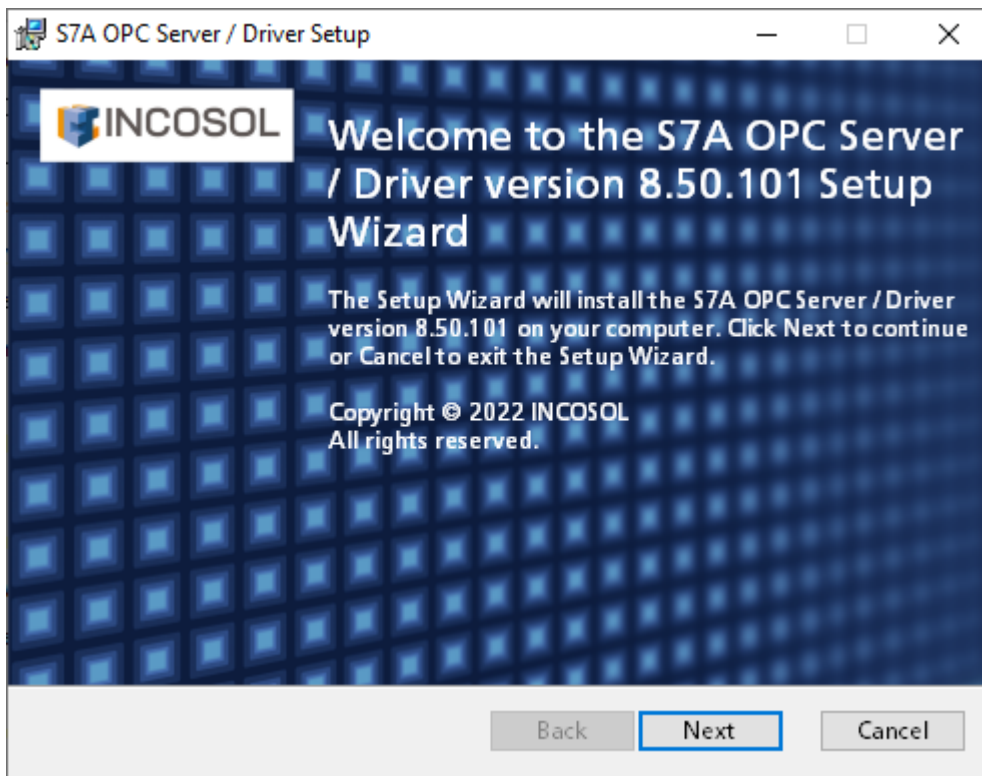
1.13 Version 8.50, build 101

This topic and its sub-topics briefly describes all new features added to S7A version 8.50, build 101

The main new features are:

- New [setup](#)^[22] routine
- Support for [iFIX secure mode](#)^[22] introduced with iFIX version 6.5
- Support for [CIMPLICITY HMI](#)^[23] versions 9.5 up to 11.5 (2022)
- OPC UA Server^[24]
- Symbol editor for all communication ways with physical (classic) addressing
- iFIX [block generation](#)^[26] from TIA symbolic
- Extended data block [diagnosis and client access monitoring](#)^[27]

1.13.1 New Setup



The Setup of the S7A OPC Server has to be redesigned to address the additional requirements for the OPC UA Server installation.

1.13.2 Support for iFIX secure mode introduced with iFIX version 6.5

In iFIX version 6.5 GE introduced a new secure mode which even has influence on the S7A driver.

One aspect of this secure mode is that all iFIX processes (SCU, Workspace, WSACTask etc.) now runs with a reduced so-called Integrity Level *Medium*. on all former iFIX versions the program run on Integrity Level *High*.

For proper interoperability between the S7A driver and iFIX it was necessary to adapt/downgrade the integrity level of the S7A process (S7ADrv.exe) to the same *Medium* level as iFIX works on.

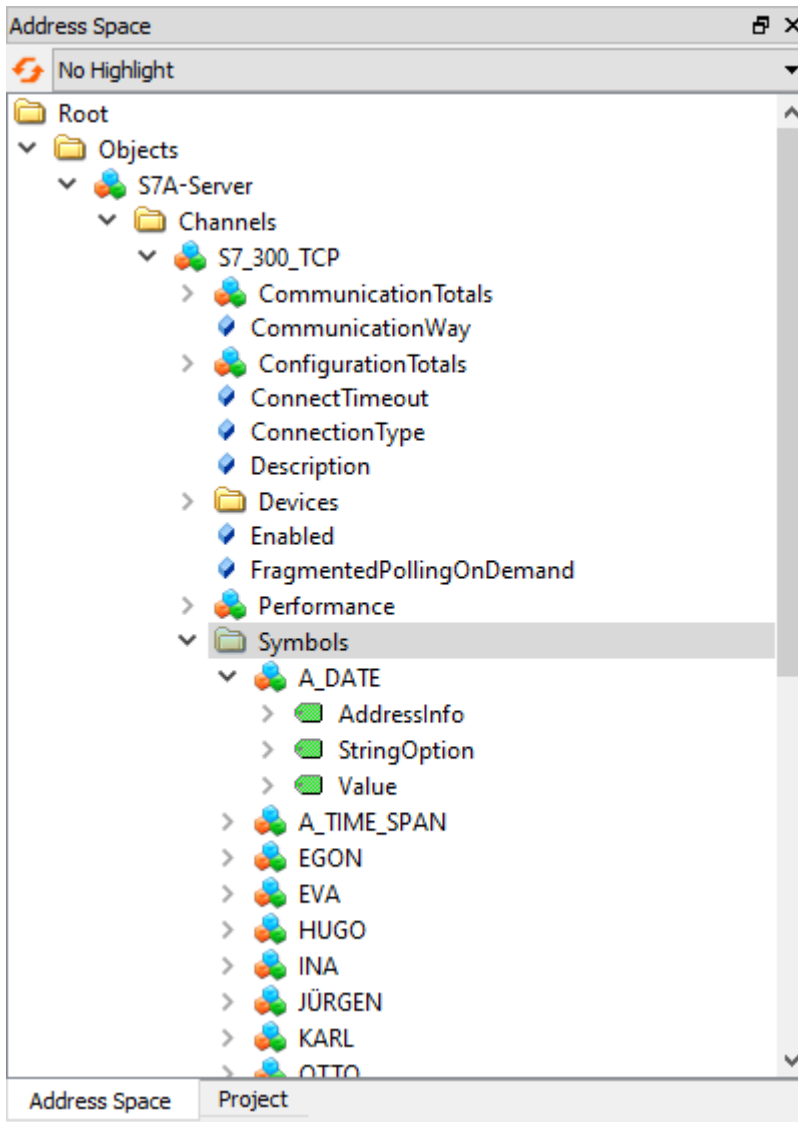
1.13.3 Symbol Editor with watch feature

The S7A driver now supports all CIMPLICITY HMI versions from 9.5 up to latest version 11.5, also know as CIMPLICITY 2022.

The native CIMPLICITY driver interface has been fully refactored with the goal of improving the performance and usability.

Now all control tags the S7A driver provides can be access by CIMPLICITY points. Furthermore all signal conditionig functions and options, the S7A driver provides, can be used in CIMPLICITY. E.g. S7 data and time data types now can be configured as CIMPLICITY text points in various, language specific display formats.

1.13.4 OPC UA Server



The S7A driver is now a full functioning OPC UA Server. This functionality is not available in the base license but a license option.

All physical communication ways (via symbol editor) as well as the two symbolic communication ways for S7-1200/1500 series and SIMOTION controllers can provide its data as OPC UA nodes.

Furthermore most of the driver's, channel's, device's and data block's configuration, runtime, performance and diagnostics properties are browsable and readable from OPC UA clients.

Server / OPC UA Base Settings

TCP Port:	<input type="text" value="48030"/>
Network Address:	<input type="text" value="DESKTOP-QQAKSBS"/>
Logical Host Name:	<input type="text" value="desktop-qqaksbs"/>
Organization Name:	<input type="text" value="INCOSOL"/>
Instance Name:	<input type="text" value="S7AUAServer"/>
Endpoint URL:	<input type="text" value="opc.tcp://DESKTOP-QQAKSBS:48030/"/>
Application URI:	<input type="text" value="urn:desktop-qqaksbs:INCOSOL:S7AUAServer"/>
Application Name:	<input type="text" value="S7AUAServer@DESKTOP-QQAKSBS"/>

Server / OPC UA Security Settings

Allow communication with no security (None):	<input checked="" type="checkbox"/>		
Allow secure communication with data privacy (SignAndEncrypt):	<input checked="" type="checkbox"/>		
Allow secure communication without data privacy (SignOnly):	<input checked="" type="checkbox"/>		

OPC UA Server Security Policies / (Encrytion modes)

Basic256Sha256 (Mandatory, always ON):	<input checked="" type="checkbox"/>		
Aes128-Sha256-RsaOaep (Recommended - Fastest):	<input type="checkbox"/>	OFF	
Basic256 (Not recommended):	<input type="checkbox"/>	OFF	
Basic128Rsa15 (Not recommended):	<input type="checkbox"/>	OFF	

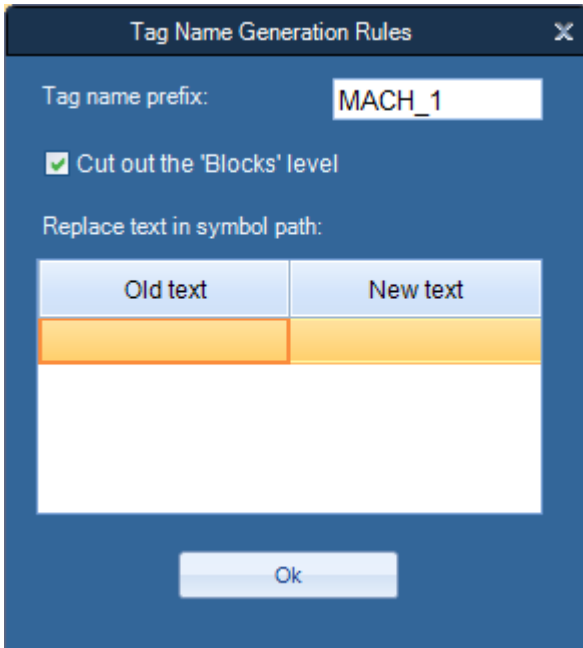
The configuration of the OPC UA Server and the entire certificate management (even the request from a GDS) is integrated in the S7A Power Tool.

1.13.5 Automatic iFIX block generation of TIA symbols

General properties						iFix Auto Gen. Properties							
Symbol	TIA data type	DB No.	Selected	Symbol comment	Status	Gen.	Block type	Tag Name	Description	I/O Address	Scan time	On scan	Enable output
Blocks	Block List		<input checked="" type="checkbox"/>										
DB_HMI	Data Block	1	<input checked="" type="checkbox"/>										
Bit_0	ST_Boolean		<input checked="" type="checkbox"/>	Single Bit 0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	DR	Blocks_DB_HMI_Bit_0	Single Bit 0	TIA_D11.Blocks.DB_HMI.Bit_0		<input type="checkbox"/>	<input checked="" type="checkbox"/>
Bit_1	ST_Boolean		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	DR	Blocks_DB_HMI_Bit_1		TIA_D11.Blocks.DB_HMI.Bit_1		<input type="checkbox"/>	<input checked="" type="checkbox"/>
Bit_2	ST_Boolean		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	DI	Blocks_DB_HMI_Bit_2		TIA_D11.TIA_DB1.Blocks.DB_HMI.Bit_2	3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Bit_3	ST_Boolean		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	DA	Blocks_DB_HMI_Bit_3		TIA_D11.TIA_DB1.Blocks.DB_HMI.Bit_3	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Bit_4	ST_Boolean		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	DO	Blocks_DB_HMI_Bit_4_XXX	Eigene Beschreibung	TIA_D11.Blocks.DB_HMI.Bit_4		<input type="checkbox"/>	<input type="checkbox"/>
Bit_5	ST_Boolean		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
Bit_6	ST_Boolean		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
Bit_7	ST_Boolean		<input type="checkbox"/>	Single Bit 0		<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
TenSignals	ARRAY (0.....		<input type="checkbox"/>	Ten Bits in an array		<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
Byte_0	ST_Byte		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	AA	Blocks_DB_HMI_Byte_0		TIA_D11.TIA_DB1.Blocks.DB_HMI.Byte_0	1	<input type="checkbox"/>	<input type="checkbox"/>
Byte_1	ST_Byte		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
USInt_1	ST_USInt		<input type="checkbox"/>	Unsigned short int (8 Bit)		<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
Word_0	ST_Word		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	AI	Blocks_DB_HMI_Word_0		TIA_D11.Blocks.DB_HMI.Word_0	1	<input type="checkbox"/>	<input type="checkbox"/>
Word_1	ST_Word		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
Int_0	ST_Int		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	AO	Blocks_DB_HMI_Int_0		TIA_D11.Blocks.DB_HMI.Int_0	1	<input type="checkbox"/>	<input type="checkbox"/>
Int_1	ST_Int		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
DWord_0	ST_DWord		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	AA	Blocks_DB_HMI_DWord_0		TIA_D11.Blocks.DB_HMI.DWord_0	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>
DWord_1	ST_DWord		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
DInt_0	ST_DInt		<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	AI	Blocks_DB_HMI_DInt_0		TIA_D11.Blocks.DB_HMI.DInt_0	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DInt_1	ST_DInt		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
Lint_0	ST_LInt		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
Lint_1	ST_LInt		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>
LWord_0	ST_LWord		<input type="checkbox"/>			<input type="checkbox"/>						<input type="checkbox"/>	<input type="checkbox"/>

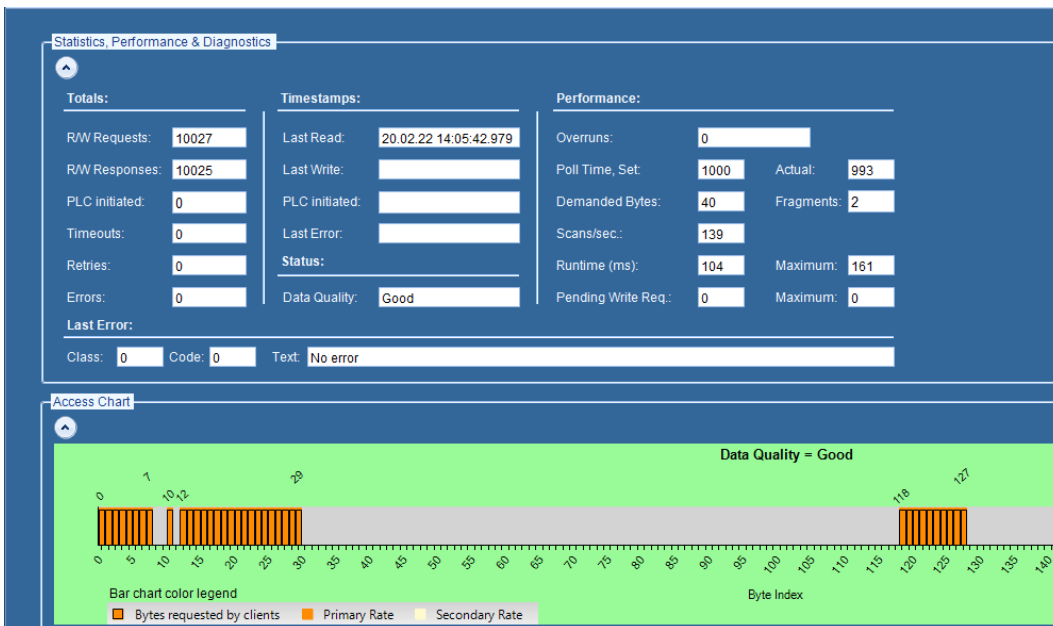
The symbol selection tree in the Power Tool's TIA data block configuration dialog has been extended by addition columns which allows to directly configure and generate iFIX database blocks from within the Power Tool. The following data block fields can be configured by this way:

- Block Types DI, DA, DO, DR, AI, AA, AO, AR and TX
- *Tag Name* can be generated automatically, derived from the TIA symbol path or can be entered individually
- *Tag Description* can be generated automatically, derived from the TIA symbol comment or can be entered individually
- *I/O Address*, derived from *STA* device name and TIA symbol path
- *Startup mode On Scan or Off Scan*
- *Scan Time*
- *Enable Output Option*
- *Signal Conditioning*
- *Hardware Option*
- *EGU Low Limit and High Limit*



The automatic iFIX tag name generation can be controlled by various options.

1.13.6 Extended data block diagnosis and client access monitoring



The Power Tool now provides a tool which allows the monitoring of the client access to the data byte of a single driver data block. A bar chart, shown on the new expandable panel 'Access Chart' in the Power Tool's data block dialog visualizes the access state of all bytes of the data block with different colors and thus provides a direct view of the current 'occupancy' of the data block by the clients (iFIX, OPC etc.).

2 Fixed defects...

The following sub-topics describe all defects, fixed in the respective versions and builds.

2.1 Version 8.60, Build 108

The following chapters describe all errors that have been fixed in version 8.60, build 108 compared to version 8.60, build 107.

2.1.1 Background Process

- Error in redundancy switching when connected to S7-400H controllers fixed, which appeared with version 8.60, build 104.
- Writing a symbolic (TIA) array variable with a size of more than 1024 bytes did not work correctly when written repeatedly.
- When writing several values that are located in different S7A data blocks in immediate succession, individual values were not written correctly to the PLC.
- When writing several values that are located in different S7A data blocks of a device in rapid succession, the order in which the values were transferred to the PLC was not the same as the order in which the values were transferred to the S7A.
- Under certain conditions, loading TIA symbols (online or from a TIA project file) could fail if 'Software Units' were configured in the TIA project.

2.1.2 iFIX Interface

In specific time zones, the date string was incorrectly formatted (to the previous day) when using the DATE_ENG hardware option.

2.2 Version 8.60, Build 107

The following chapters describe all errors that have been fixed in version 8.60, build 107 compared to version 8.60, build 106.

2.2.1 Background Process

The S7ADrv process could crash when writing a symbolic TIA address.

2.2.2 iFIX Interface

When NIO simulation mode was active, string variables (iFIX TX blocks) were not written correctly to symbolic TIA strings and TIA character arrays.

2.3 Version 8.60, Build 106

The following chapters describe all errors that have been fixed in version 8.60, build 106 compared to version 8.60, build 105.

2.3.1 Background Process

After the general query of all open alarms of an S7-1200/1500 controller, the data quality was only set to GOOD for the alarm data blocks for which an alarm was received. For all other data blocks, the data quality was set to BAD. As a result, COMM alarms were generated for all iFIX database blocks that referred to S7A alarm data blocks with data quality BAD. Now, after the general query, all S7A alarm data blocks are set to data quality GOOD. All iFIX data base blocks for which no PLC alarm is pending thus receive the initial, valid F_CV value 0.

2.3.2 iFIX Interface

1. Writing strings via iFIX could cause the WSACTask, TCPTask or workspace process to crash.
2. If the string to be written contained language-specific characters (e.g. German umlauts ä, ö or ü), too many characters may have been sent to the PLC if the string options SPACEFILL, NULLTERM or SIEMENS were activated.

2.4 Version 8.60, Build 105

The following chapters describe all errors that have been fixed in version 8.60, build 105 compared to version 8.60, build 104.

2.4.1 Background Process

The background process S7ADrv.exe could crash with an application error due to a malfunction implemented in build 104.

2.4.2 Power Tool

The Trust List of OPC UA certificates was created without the buttons for moving the certificates and without a context menu. An error message was displayed when trying to open the context menu.

2.4.3 iFIX Interface

The formatting of an S7_Date value into a string returned an incorrect string if the system time zone was set to a negative value (e.g. UTC - 8).

2.5 Version 8.60, build 104

The following chapters describe all errors that have been fixed in version 8.60, build 104 compared to version 8.60, build 103.

2.5.1 Background process

1. Memory leak if database blocks with I/O addresses to unresolved TIA addresses exist in an iFIX application.
2. The default value of the configuration parameter '**Scans per second**' was reduced from 50 to 15. It was discovered that for very large configurations with several thousand data blocks, the CPU load caused by the S7ADrv process was unusually high and this also slowed down the processing of the iFIX database by the WSACTASK. For S7A configurations that require low poll times (< 500ms) with the most accurate possible compliance with these poll times, the configuration parameter '**Scans per second**' should be increased. In such configurations, a '**Scans per second**' value that is too low would lead to a greater deviation of the actual poll time compared to the configured value.
3. Writing an empty string (from iFIX or OPC) led to the application being blocked for a few seconds, which triggered the write request.
4. The list of selected symbols of a TIA data block was not saved if no connection to the PLC was established and the configuration was saved. As a result, all previously selected symbols were no longer selected at the next start.
5. A byte twister when writing bits to S5 data block addresses has been fixed.
6. Fixed an error with the **!SyncPoll** command on a physical data block. If **Optimized Polling on Demand** was active and the primary and secondary poll rate was disabled for the data block and a **!SyncPoll** was executed, the data quality of the block was set to **Bad, Out of Service** immediately after polling. Now, if neither primary nor secondary rate is set but an access time

and the latch data flag is *not* set, the quality is set to **GOOD** after the !SyncPoll. Once the access time has expired, the quality changes to **BAD, Out of service**. If, on the other hand, no access time is set or the latched data flag is set, the quality remains **GOOD**.

7. Selected TIA array symbols with a very large number of array elements (> 5000) led to a long loading time when starting the S7A background process. This time delay has been significantly reduced.
8. Writing TIA arrays with a size of more than 1024 bytes failed. TIA arrays of any size can now be written.
9. Fixed a memory leak that occurred when TIA symbols were read or written that were not present in the PLC. This situation could occur if the S7A driver read the symbols via the TIA project file, but the TIA project had not (yet) been loaded onto the PLC, i.e. it may have contained new symbols that were not yet present on the PLC.
10. If TIA symbols are to be read that are not present on the PLC, the iFIX tags associated with them are now set to Off-Scan. Previously, the value of the tag was set to 0 (analog tag) or False (digital tag).
11. Although the poll time of TIA blocks could be set to 0, only a minimum of 500 ms was actually achieved. Now poll times of less than 100 ms can also be achieved.
12. If the number of symbols to be read (selected) in a TIA data block was greater than the value in the registry value **TIA_MaxCollectiveSymbolsInReadRequest**, the S7ADrv process crashed.
13. Block writing could lead to a crash of the S7ADrv process.
14. When writing to a TIA array element of a fully selected array variable, the wrong array element was written to in the PLC.
15. The display of the process values in the TIA symbol grid led to an increased CPU load on the S7ADIDW process.

2.5.2 Power Tool

1. Copying the TIA symbols from the TIA project to the S7A configuration reported an error.
2. With TIA symbol source 'S7A Config', the TIA project was not loaded when the button for transferring the symbols from the TIA project to the S7A configuration was clicked.

3. The own OPC UA certificate was not displayed.
4. A symbol filter expression with the string operation 'DataBlock Like' and a wildcard character in front of the search term led to incorrect results. Data blocks were filtered that did not match the condition.
5. The validation of the filter expression has been improved at channel level. If a filter expression was defined with a string operator '**DataBlock like**' or '**DataBlock contains**' without a (mandatory) preceding string operator for the PLC name (i.e. '**PLCName like**' or '**PLCName contains**'), no explicit error message was issued.
6. Selecting a TIA array with a large number of elements (several thousand) led to a very long delay/blocking of the Power Tool. This delay time has been significantly reduced.
7. After creating a new configuration via the Power Tool (File->New), no I/O addresses that were sent to the driver for validation via iFIX or OPC were accepted but rejected as 'invalid address'.
8. The error status of a channel node in the driver object structure tree was not displayed correctly. If a device summary alarm was set once, the error status of the channel node was not deleted after all alarms of the subordinate devices had been resolved. The channel node continued to display an error.
9. When the service account was validated, the incorrect check was carried out in the 4th step in which it is checked whether the account is authorized to start a service or to be used as a service account. As a result, an account that had no service authorization could pass the validation, but the subsequent reconfiguration of the service then failed since the user account did not have the authorization to do that.

2.5.3 iFIX Interface

1. Writing an unsigned 64-bit value (LWORD) failed for values greater than 9,223,372,036,854,775,807. Such values were incorrectly converted to a signed negative value and written to the PLC.

2.5.4 OPC UA Server

1. The ring buffer management of the OPC UA server log files did not work. Log files were generated in more than the configured number of log files.

2.6 Version 8.60, build 103

The following chapters describe all bugs fixed with version 8.60, build 103 compared to version 8.50, build 109.

2.6.1 Background process

1. With certain data block configurations of a symbolic TIA device it could happen that some data blocks were 'stuck', that means they were no longer polled, when cyclic write requests to the data blocks were active.
2. When stopping the driver, the data quality of certain symbolic TIA data blocks was not set to 'BAD' but remained at 'GOOD'.
3. Writing of digital signals to a S5 data block address does not work.
4. The sporadic error fff50012 (NO_INFORMATION) on AG state requests is now completely suppressed.
5. By installing an S7A update, an existing configuration file in the current iFIX PDB directory with the name <iFIX node name>.S7A was overwritten by an empty (dummy) file.

2.6.2 Power Tool

1. On channel and device level, the count for retries and timeouts did not work properly. As soon as one of the two values became greater than zero on the data block level, the counters on the device and channel level were continuously incremented.
2. When inserting a channel, device or data block below/above a selected entry in the object tree, the new object was always appended to the bottom of the list instead of being inserted at the selected position.
3. Setting an individual symbol file name in the Power Tool's options did not work.
4. With a large number (> 1000) of symbols or alias names, the Power Tool caused a high CPU load and the update of the symbol grid was very slow. Here the performance was increased significantly and the CPU load was reduced considerably.
5. Write requests to data blocks which were set to manual or automatic block write mode were not counted in the write statistics.

2.7 Version 8.50, build 109

The following sub-topics describe all defects which have been fixed in build 109.

2.7.1 Background process

1. Clients (iFIX WSACTask, Workspace, OPC Clients etc.) of the S7A Server background process could crash when the heartbeat logging was enabled.
2. Some On-the-fly changes of the debug and logging parameters were ignored by the S7A Server background process.
3. The S7A Server background process could crash when user deselected TIA symbols in the Power Tool's symbol tree.
4. The S7A Server background process could crash when a control tag I/O address (e.g. !Poll) was read or written but the referring driver object does not exist anymore in the current driver configuration.
5. The maximum size of a S7ADrv log file is now limited to 1 GB.
6. Switch back from the backup to the primary device failed in version 8.50, build 108.
7. Once user created a new, empty configuration in Power Tool, the driver rejected all symbolic TIA addresses.
8. Under rare occasions the S7A background process could crash after the Power Tool was started.

2.7.2 Power Tool

1. The Power Tool's log viewer window cyclically showed the error message 'The communication with the OPC UA Server failed with an exception in code path 4' even though the OPC UA server was neither installed nor running.

2.7.3 iFIX interface

1. Under specific conditions all non-register blocks have been set to off-scan after iFIX start.
2. An initial iFIX register (DR or AR) read or write operation with a register offset greater 0 failed. The specified offset value was not recognized. On a

read operation the driver returned the value of the register's base address, on a write operation the driver has written the write value to the register's base address.

3. An iFIX data base which contained non-register blocks with symbolic TIA addresses and the 'Exception based' scan flag set, could corrupt the internal structure of the driver data block's shared memory when the iFIX block was stored in the upper half of the driver data block's shared memory.

2.8 Version 8.50, build 108

The following sub-topics describe all defects which have been fixed in build 108.

2.8.1 Background process

1. When 'Auto create', respectively 'Auto select' for symbolic TIA channels, was active, the top most element of a TIA array variable was not automatically selected and rejected by the I/O address parser.
2. By an export of the driver configuration to CSV file, the primary rate of all data block was overwritten by the value of the secondary rate.
3. When the data quality of a data block changed from GOOD to BAD the driver sent a 'communication recovered' message to the iFIX alarm system instead of 'communication failed'.

2.8.2 Power Tool

1. The channel setting 'iFIX Register Offset is Element Offset' was not properly set on driver start. The driver always worked with 'Byte Offset' even though 'Element Offset' was set for the channel.
2. Undocking and moving the 'Log Viewer' dock window led to an exception message. Now the 'Log Viewer' window can be undocked, moved and re-docked properly.
3. The drop down area of the 'Communication Way' drop down list has been extended to show all communication way items without the need to scroll the list.

2.8.3 OPC DA/UA

1. On a connect of an OPC DA client to the Server an empty event message was sent to the Windows event log system.
2. Under certain circumstances the Power Tool's internal OPC UA client interface cyclically sent 'connection failed' messages to the 'Log Viewer' window.

2.9 Version 8.50, build 106

The following topics describe all defects which have been fixed in build 106.

2.9.1 Program installation (Setup)

1. When S7A driver was installed as a Windows Service, any configuration changes made in the Backstage Setup View of the S7A Power Tool (e.g. change of default configuration file name) only became active once the system was restarted or the S7A driver service was restarted. With this new build the S7ADrv service start-up mode has been changed from 'Automatic' to 'Manual' which means that the S7ADrv Service is not started automatically with the system start-up. Instead it is started by the first client (iFIX processes, OPC client or Cimplicity S7A_DC process) which connects to the S7ADrv background server. In turn, the S7ADrv service is stopped once the last client is disconnected from the S7ADrv background service. Changes in the setup parameters (made before via S7A Power Tool) now become effective upon the S7ADrv service has been automatically has terminated and started next time (by a new client connection).

2.9.2 Background process

1. When automatic symbol selection was active (*Auto create 'ON'* in Advanced driver setup) symbols which contained language-specific characters in its symbol path were not selected automatically.
2. When exception-based data transfer between S7ADrv process and a client (iFIX WSACTask, OPC UA Server or Cimplicity driver) was configured the S7ADrv process generated duplicate exceptions for digital signals (boolean addresses).

2.9.3 Native CIMPLICITY interface

1. The S7A_DC driver process crashed when more than 100 Cimplicity points referred to the same S7A driver data block.
2. Memory leak in S7A_DC driver process has been fixed.
3. Point values have been displayed as zero values instead as '***' when the PLC was disconnected.

2.10 Version 8.50, build 105

The following topics describe all defects which have been fixed in build 105.

2.10.1 Power Tool

1. The settings for *I/O Address Syntax* and *iFIX Register Offset* in the channel configuration were not persistently stored in the configuration file. Once the *I/O address syntax* was set to 'S7M comp.' and/or the *iFIX Register Offset* was set to 'is element offset' and the configuration was saved and the driver was restarted, the settings changed back to the default settings ('S7 Not. IEC' and 'is byte offset').

2.10.2 Background process

1. Once the connection to a PLC dropped, all the iFIX database blocks which referred to the PLC went persistently Off-Scan. Now they stay On-Scan. Only the data quality of the database block changes to 'BAD' thus all data links in iFIX pictures changes from regular value to displaying question marks. As soon as connection is re-established the data links become valid again and displaying their regular value.
2. Loading an empty S7A configuration file could cause a crash of the S7ADrv.exe process.
3. Large configurations with hundreds of S7A data blocks could cause a high CU load on the S7ADrv.exe process.
4. Symbolic (TIA) data blocks not started cyclical polling when a *Primary Rate* and an *Access Time* but no *Secondary Rate* were configured for the data block
5. When disabling a S7A device e.g. via Power Tool or by a !Mode:<device name> control tag, the data quality of some of the subordinated data blocks remained at 'GOOD' instead of changing to 'BAD'.

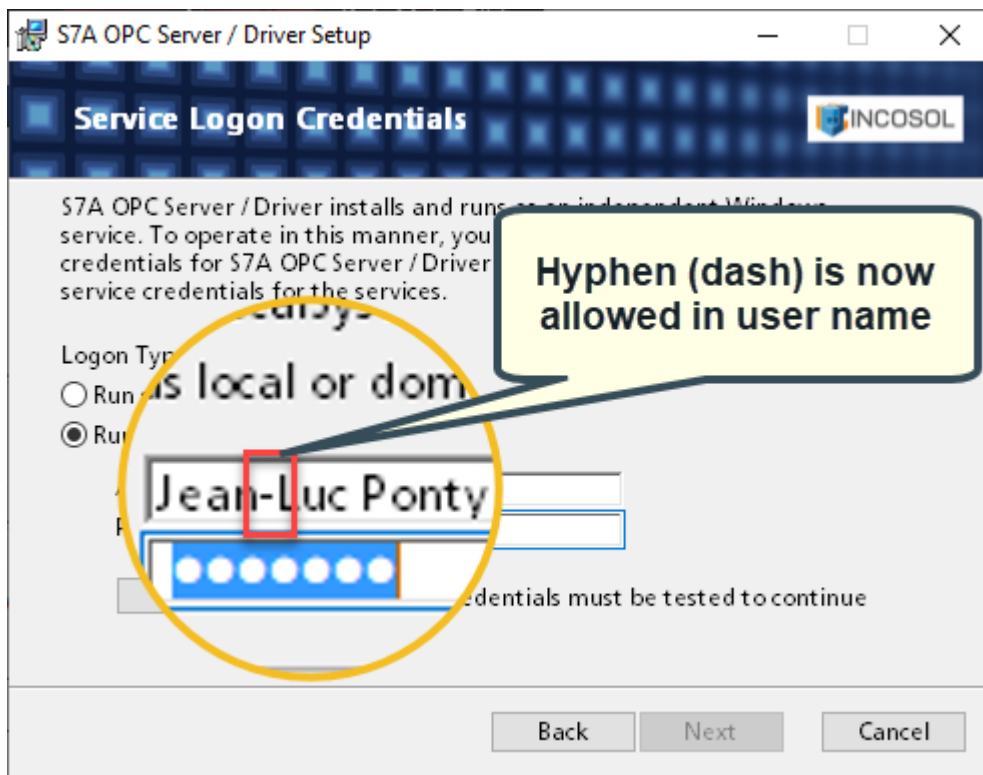
6. When symbol source 'TIA project' was selected and a PLC name within the TIA project contained language specific characters then the selection of symbols in the S7A data block dialog failed. The auto-selection of symbols when the iFIX database was loaded failed as well.
7. When Auto-create mode was activated and iFIX was started and a S7A data block was deleted via the Power Tool, the S7ADrv.exe process crashed.

2.11 Version 8.50, build 104

The following topics describe all defects which have been fixed in build 104.

2.11.1 Program installation (Setup)

The 'Account' field in the 'Service Logon Credentials' dialog did not accept hyphens (dashes). Now hyphens within the account name are allowed.



2.11.2 Power_Tool

1. In the device configuration dialog of a S7 TCP channel the selection of the PLC type has not been persistently stored. Instead it has been reset to the default selection 'S7-300/400' once the configuration dialog has been closed and opened again.

2. For data blocks with memory area unequal to 'Data Block' the start index could not be set to a value greater than zero.

2.11.3 Native iFIX interface

1. When the data block option 'Auto Create' was activated the S7A background process (S7ADrv) could possibly crash when the iFIX data base manager program was launched.
2. A memory leakage in the S7A background process (S7ADrv), the iFIX WSACTask and the iFIX data base manager might occur in cases where the iFIX data base contained blocks with unresolved S7A I/O addresses.

2.11.4 Native CIMPLICITY interface

1. The native CIMPLICITY driver process S7A_DC.EXE possibly could crash.
2. Writing a value to a single element of a CIMPLICITY Array point failed.

2.12 Version 8.50, build 102

The following topics describe all defects which have been fixed in build 102.

2.12.1 Offset addressing via iFIX DR block on analog TIA symbols

The indexed addressing (F_x) via iFIX DR block of a bit within an analog TIA symbol value failed. Only the first bit's value was read no matter which offset was defined in the iFIX data link.

2.12.2 High CPU load caused by S7ADRV process

On large S7A configuration with multiple symbolic TIA channels and devices the S7A driver background process S7ADrv.exe could cause a high CPU load of more than 50%.

2.12.3 Power Tool failed to connect to background process

After the driver was installed as a service the S7A Power Tool and all other clients which run on integrity level 'Medium' failed to connect to the

background process (S7ADrv.exe service). After a restart of the computer or a restart of the S7A Driver service the issue disappeared.

2.13 Version 8.50, build 101

The following topics describe all defects which have been fixed in build 101.

2.13.1 Lower case I/O addresses or OPC Item Ids

The I/O address portion (the part right to the colon) of an iFIX I/O address or an OPC Item Id now accept lower case characters for the memory area and data type specifier. Example: 'D11:db100.dbw0' now is a valid I/O address.

2.13.2 Import of version 8.10 CSV import files

- The import of a CSV configuration file exported by version 81.0 failed due to an invalid or missing symbol filter expression string value (CSV file property **TCPTIASymbolFilterExpr**) in the channel portion of a symbolic TIA channel.
- The import of a CSV configuration file exported by version 81.0 failed due to an invalid or missing TIA project file time stamp value (CSV file property **TCPTIAProjectTSSaved**) in the channel portion of a symbolic TIA channel. The driver now tolerates such missing property values.
- CSV import failed when the CSV file was modified and save in MS Excel. Now the driver accepts CSV files which were modified and saved by Excel.
- The Channel property **TCPMaxGap** has been added to achieve compatibility with CSV files generated with version 8.10.

-

2.13.3 Poll overrun counting

The detection of the polling overruns counted much more overruns than actually occurred.

2.13.4 Fragmented polling on demand

The '**S7-300/400/1500 TCP/IP Absolute**' communication way channel feature '**Fragmented polling on demand**' now works properly.

2.13.5 Limitation of Slot number range

The **Slot** number field in the TCP/IP device settings dialog was limited to a range between 0 and 10. The upper limit has been extended to 64.

2.13.6 iFIX Register Element Offset

The **iFIX Register Offset is Element Offset** channel setting failed in S7A 8.50 RC14. This option is working properly now.

2.13.7 Control Tag addresses for symbolic (TIA) driver objects

All control tag addresses (e.g. !Mode:<device name>) with driver objects (Channel, Device or Data Block) of symbolic communication ways (TIA and SIMOTION) failed and were rejected by the S7ADrv.

2.13.8 Simulation mode

In simulation mode of symbolic communication ways (TIA and SIMOTION) the data quality 'BAD' was returned to the clients (iFIX or OPC). Now the driver returns the proper data quality 'GOOD'.

2.13.9 OPC UA Security Policy

Setting the OPC UA Security Policy 'None' on or off via the Power Tool's OPC UA options dialog had no effect on the OPC UA Server.

2.13.10 Limitation of Cimplicity point name length

The length of a CIMPLICITY point address was limited to 32 characters, which is too less for symbolic addresses for TIA or SIMOTION communication ways. The address length has been extended to 296 characters.

2.13.11 Software license and lower case computer name

Under certain circumstances the comparison of the computer name against the name encoded in the software license file failed when the computer name contained lower case characters.

2.14 Known Issues

The following issues or functional restrictions are still pending and are going to be address by the next version or build.

- The TCP port number of the EDA GRPC server is not configurable. Currently it is solely changeable in the `S7ADIDW.exe.config` file.
- The EDA GRPC Server's log files are stored in the installation folder but not in the program data path (where the other S7A program store their log files).
- The change of the user interface language does not work when S7A is installed as a service and Power Tool is started with integrity level 'Medium'.
- Due to an issue in iFIX 6.1 (the `S7ASTSR.DLL` is improperly unloaded by the Databasemanager) the `S7ADrv` background process does not automatically shut down while iFIX is shutting down. This issue seems to be fixed with iFIX version 6.5.