



SIEMENS

Function manual

SIMATIC

S7-1500, ET 200SP, ET 200pro, SIMATIC Drive Controller

Structure and Use of the CPU Memory

Edition

11/2019

support.industry.siemens.com

SIEMENS

SIMATIC

S7-1500, S7-1500R/H, SIMATIC Drive Controller, ET 200SP, ET 200pro Structure and Use of the CPU Memory

Function Manual

Preface

Function manuals
Documentation Guide **1**

Memory areas and retentive
memory **2**

Memory usage and
application examples **3**

SIMATIC memory card **4**

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER
indicates that death or severe personal injury will result if proper precautions are not taken.
⚠ WARNING
indicates that death or severe personal injury may result if proper precautions are not taken.
⚠ CAUTION
indicates that minor personal injury can result if proper precautions are not taken.
NOTICE
indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Preface

Purpose of the documentation

This documentation describes the various memory areas of the following systems:

- SIMATIC S7-1500 automation system
- SIMATIC S7-1500R/H redundant system
- SIMATIC Drive Controller
- The CPUs of the ET 200SP distributed I/O system
- The CPUs of the ET 200pro distributed I/O system based on SIMATIC S7-1500

The book also shows:

- How to make the best use of the memory areas
- How to reduce work memory utilization by:
 - Using recipes and data logs
 - Saving data blocks on the SIMATIC memory card

Basic knowledge required

The following knowledge is required in order to understand the documentation:

- General knowledge of automation technology
- Knowledge of the SIMATIC industrial automation system
- Knowledge about the use of computers
- Proficiency with STEP 7

Conventions

STEP 7: in this documentation, "STEP 7" is used as a synonym for all versions of the "STEP 7 (TIA Portal)" configuring and programming software.

Please also observe notes marked as follows:

Note

A note contains important information on the product described in the documentation, on the handling of the product or on the section of the documentation to which particular attention should be paid.

Scope of the documentation

This documentation is valid for the central modules of the SIMATIC S7-1500, SIMATIC Drive Controller and ET 200SP systems and for ET 200pro.

The CPUs of the redundant system S7-1500R/H do not support all the memory objects described in this function manual. The limitations of the redundant S7-1500R/H system are pointed out where appropriate in the manual. You can find a list of the unsupported functions in the Redundant System S7-1500R/H (<https://support.industry.siemens.com/cs/ww/en/view/109754833>) system manual.

What's new in edition 11/2019 as compared to edition 10/2018?

What's new?		What are the customer benefits?	Where can I find information?
Changed contents	Scope of the function manual expanded to include the CPUs of the SIMATIC Drive Controller	SIMATIC Drive Controller and SINAMICS Integrated support the memory structure familiar from the S7-1500 automation system. Information on the special features of the SIMATIC Drive Controller is explained at the relevant points in the manual.	<ul style="list-style-type: none"> In the corresponding sections of the manual SIMATIC Drive Controller (https://support.industry.siemens.com/cs/ww/en/view/109766665) system manual

What's new in Edition 10/2018 compared to Edition 09/2016

What's new?		What are the customer benefits?	Where can I find information?
Changed contents	Scope of the function manual expanded to include the CPUs of the redundant system S7-1500R/H.	The CPUs of the redundant system S7-1500R/H support the memory structure familiar from the S7-1500 automation system. Information on the particularities of the redundant S7-1500R/H system is provided where appropriate in the manual.	<ul style="list-style-type: none"> In the corresponding sections of the manual System manual Redundant System S7-1500R/H (https://support.industry.siemens.com/cs/ww/en/view/109754833)

What's new in Edition 09/2016 compared to Edition 01/2013

What's new?		What are the customer benefits?	Where can I find information?
New contents	Analyzing memory requirements and memory usage	<p>You have various options for analyzing the memory requirements and the memory usage of the CPU:</p> <ul style="list-style-type: none"> • With STEP 7 • With the display of the CPU • With the web server of the CPU 	Section Memory requirements and memory usage (Page 15)
	Memory requirements in load memory for downloading software changes	<p>When loading software changes to the SIMATIC memory card, the files in question are only deleted after creation of the new files. For this reason, the CPU requires adequate free memory space on the SIMATIC memory card.</p> <p>You have various options for creating memory space; these will be explained.</p>	Section Memory requirements for downloading software changes (Page 28)
	Service life of SIMATIC memory cards	Using calculation examples of the service life of a SIMATIC memory card, you can estimate which SIMATIC memory card is required for your automation task.	Section Service life of the SIMATIC memory card (Page 63)
Changed contents	Scope of the function manual expanded to include the CPUs of the ET 200SP distributed I/O system and the CPU 1516pro-2 PN	Functions that you will be familiar with from the SIMATIC S7-1500 CPUs are implemented in CPUs in other designs (ET 200SP) and in the CPU 1516pro-2 PN (degree of protection IP65, IP66 and IP67).	<ul style="list-style-type: none"> • Manual CPU 1510SP-1 PN (https://support.industry.siemens.com/cs/ww/en/view/90157130) • Manual CPU 1512SP-1 PN (https://support.industry.siemens.com/cs/ww/en/view/90157013) • Operating instructions CPU 1516pro-2 PN (https://support.industry.siemens.com/cs/ww/en/view/109482416)

Recycling and disposal

For environmentally friendly recycling and disposal of your old equipment, contact a certified electronic waste disposal company and dispose of the equipment according to the applicable regulations in your country.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For additional information on industrial security measures that may be implemented, please visit (<https://www.siemens.com/industrialsecurity>).

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customers' exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed visit (<https://www.siemens.com/industrialsecurity>).

Siemens Industry Online Support

You can find current information on the following topics quickly and easily here:

- **Product support**

All the information and extensive know-how on your product, technical specifications, FAQs, certificates, downloads, and manuals.

- **Application examples**

Tools and examples to solve your automation tasks – as well as function blocks, performance information and videos.

- **Services**

Information about Industry Services, Field Services, Technical Support, spare parts and training offers.

- **Forums**

For answers and solutions concerning automation technology.

- **mySupport**

Your personal working area in Industry Online Support for messages, support queries, and configurable documents.

This information is provided by the Siemens Industry Online Support in the Internet (<https://support.industry.siemens.com>).

Industry Mall

The Industry Mall is the catalog and order system of Siemens AG for automation and drive solutions on the basis of Totally Integrated Automation (TIA) and Totally Integrated Power (TIP).

You can find catalogs for all automation and drive products on the Internet (<https://mall.industry.siemens.com>).

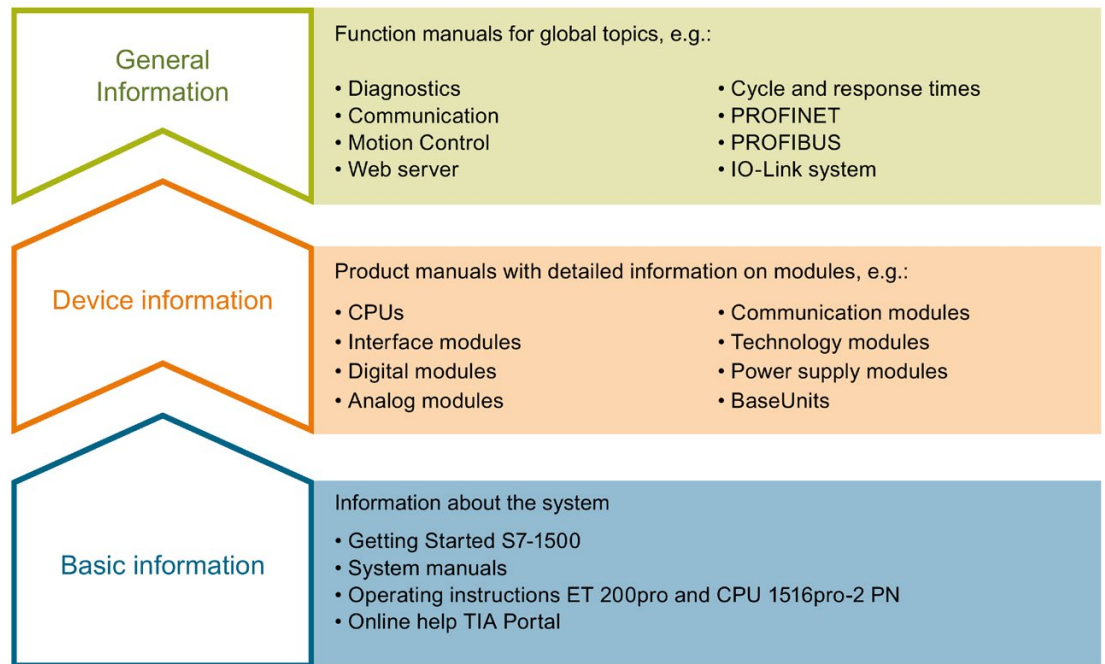
Table of contents

	Preface	3
1	Function manuals Documentation Guide	9
2	Memory areas and retentive memory	11
2.1	Memory areas	11
2.1.1	Specifics of the CPUs of the redundant system S7-1500R/H	14
2.2	Memory requirements and memory usage	15
2.3	Retentive memory areas	21
2.4	Summary of retentive behavior	25
2.4.1	Retentive behavior of the memory objects	25
2.5	Memory behavior when loading software changes.....	26
2.6	Memory requirements for downloading software changes	28
3	Memory usage and application examples	33
3.1	Memory usage for data blocks.....	33
3.2	Memory usage for recipes	35
3.3	Memory usage for data logging	37
3.3.1	Overview of data logging	37
3.3.2	Data structure of the data logs.....	39
3.3.3	Instructions for data logging.....	40
3.3.4	Example program for data logging.....	41
3.3.5	Calculation of the data log size	47
4	SIMATIC memory card	51
4.1	SIMATIC memory card - Overview	51
4.2	Setting the card type	58
4.3	Data transfer with SIMATIC memory cards	60
4.4	Service life of the SIMATIC memory card.....	63
4.5	Expanding the load memory of the CPUs of the redundant system S7-1500R/H.....	68
	Glossary	69
	Index	75

Function manuals Documentation Guide

The documentation for the SIMATIC S7-1500 automation system, for CPU 1516pro-2 PN based on SIMATIC S7-1500, and for the distributed I/O systems SIMATIC ET 200MP, ET 200SP and ET 200AL is divided into three areas.

This division allows you easier access to the specific information you require.



Basic information

System manuals and Getting Started manuals describe in detail the configuration, installation, wiring and commissioning of the SIMATIC S7-1500, ET 200MP, ET 200SP and ET 200AL systems; use the corresponding operating instructions for CPU 1516pro-2 PN. The STEP 7 online help supports you in configuration and programming.

Device information

Product manuals contain a compact description of the module-specific information, such as properties, terminal diagrams, characteristics and technical specifications.

General information

The function manuals contain detailed descriptions on general topics such as diagnostics, communication, Motion Control, Web server, OPC UA.

You can download the documentation free of charge from the Internet (<https://support.industry.siemens.com/cs/ww/en/view/109742705>).

Changes and additions to the manuals are documented in product information sheets.

You will find the product information on the Internet:

- S7-1500/ET 200MP (<https://support.industry.siemens.com/cs/us/en/view/68052815>)
- ET 200SP (<https://support.industry.siemens.com/cs/us/en/view/73021864>)
- ET 200AL (<https://support.industry.siemens.com/cs/us/en/view/99494757>)

Manual Collections

The Manual Collections contain the complete documentation of the systems put together in one file.

You will find the Manual Collections on the Internet:

- S7-1500/ET 200MP (<https://support.industry.siemens.com/cs/ww/en/view/86140384>)
- ET 200SP (<https://support.industry.siemens.com/cs/ww/en/view/84133942>)
- ET 200AL (<https://support.industry.siemens.com/cs/ww/en/view/95242965>)

"mySupport"

With "mySupport", your personal workspace, you make the best out of your Industry Online Support.

In "mySupport", you can save filters, favorites and tags, request CAx data and compile your personal library in the Documentation area. In addition, your data is already filled out in support requests and you can get an overview of your current requests at any time.

You must register once to use the full functionality of "mySupport".

You can find "mySupport" on the Internet (<https://support.industry.siemens.com/My/ww/en>).

Application examples

The application examples support you with various tools and examples for solving your automation tasks. Solutions are shown in interplay with multiple components in the system - separated from the focus on individual products.

You will find the application examples on the Internet (<https://support.industry.siemens.com/sc/ww/en/sc/2054>).

Memory areas and retentive memory

2.1 Memory areas

The automation data is located in the automation system in different memory areas.

The offline data of the project created in STEP 7 is located on the hard disk of the programming device. The online data of the project is located in the load memory on the SIMATIC memory card. In addition, the work memory, retentive memory and other memory areas are located on the CPU.

The following figure shows an overview of the memory areas of the CPUs:

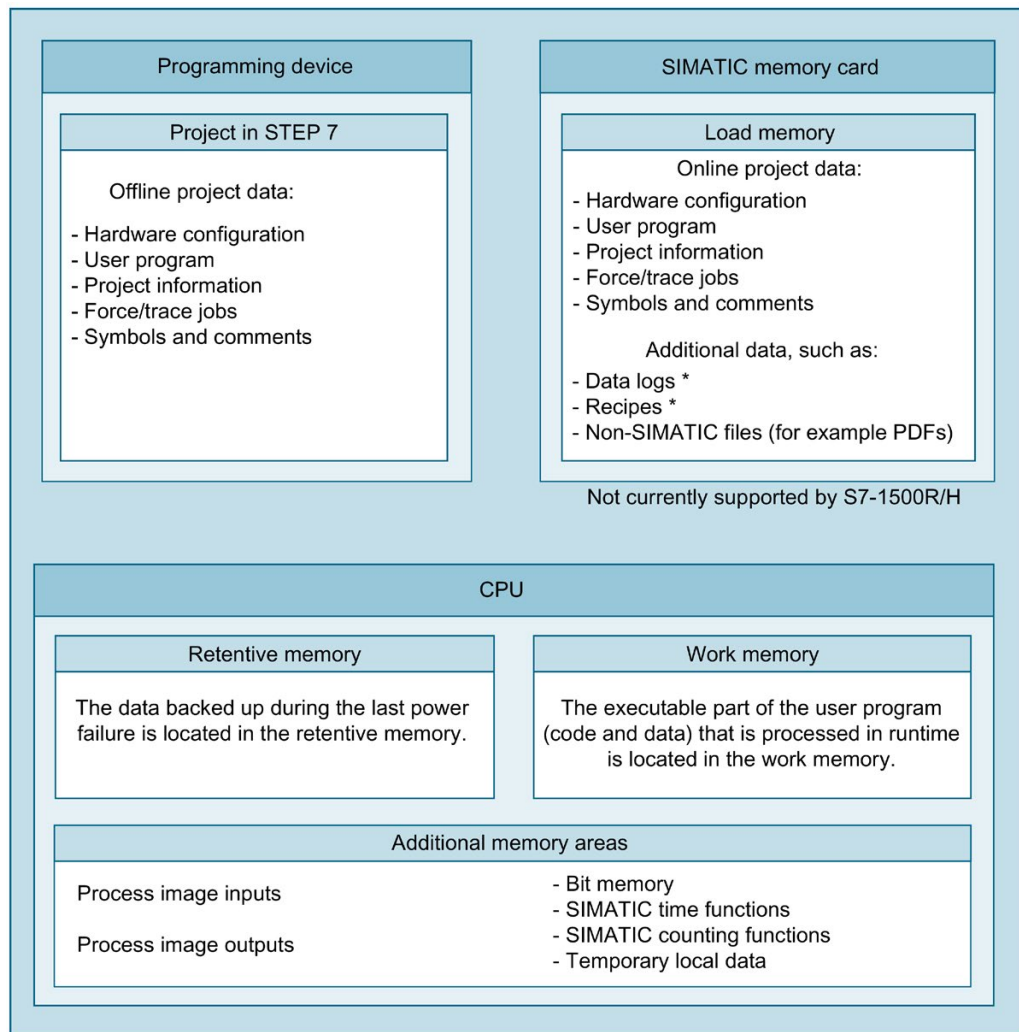


Figure 2-1 Memory areas

Load memory

Load memory is non-volatile memory for code blocks, data blocks, technology objects and the hardware configuration. This load memory is located on the SIMATIC memory card. STEP 7 transfers the project data from the programming device to the load memory.

You can copy additional data (for example HMI backups and other files) to the SIMATIC memory card using the web server or Explorer. This data can then be found in the load memory on the SIMATIC memory card.

Note

An inserted SIMATIC memory card is required to operate the CPU.

Load memory: CPU 1518-4 PN/DP MFP and CPU 1518F-4 PN/DP MFP

On these CPUs additional memory space is used in the load memory by:

- Linux runtime, which runs in parallel with the CPU runtime
- C/C++ runtime applications
- Files that are needed for C/C++ runtime applications
- CPU function libraries

Note

CPU runtime

In the context of the CPU 1518-4 PN/DP MFP and the CPU 1518F-4 PN/DP MFP as well as the Open Development Kit (ODK), the term "CPU runtime" has the following meaning:

The CPU runtime is the runtime environment in which a CPU runtime application can be executed. CPU 1518-4 PN/DP MFP and CPU 1518F-4 PN/DP MFP run a Linux runtime in parallel with the CPU runtime. The Linux runtime is the runtime environment for applications that are executable in Linux, e.g. C/C++ runtime applications.

For additional information on the CPUs, refer to the CPU 1518-4 PN/DP MFP (<https://support.industry.siemens.com/cs/ww/en/view/109749061>) manual and the CPU 1518(F)-4 PN/DP MFP (<https://support.industry.siemens.com/cs/ww/en/view/109756478>) product information. You can find information on creating C/C++ runtime applications in the SIMATIC S7-1500 ODK 1500S (<https://support.industry.siemens.com/cs/ww/en/view/109752683>) manual.

Work memory

The work memory is volatile memory that contains the code and data blocks. The work memory is integrated into the CPU and cannot be extended. The work memory is only used in operation of the CPU.

In the CPUs, the work memory is divided into two areas:

- Code work memory: The code work memory contains runtime-relevant parts of the program code.
- Data work memory: The data work memory contains the runtime-relevant parts of the data blocks and technology objects.
Tags of global data blocks, instance data blocks and technology objects are initialized with their start values at the operating states changes below. Retentive tags receive their actual values saved in the retentive memory.
 - STOP → STARTUP
 - POWER ON → STARTUP
 - POWER ON → STOP

Work memory of CPU 1518-4 PN/DP MFP and CPU 1518F-4 PN/DP MFP

Additional work memory is needed for use of the CPU function libraries and C/C++ runtime applications.

Retentive memory

The retentive memory is non-volatile memory for saving a limited amount of data in the event of power failure.

The following actions delete certain memory objects of the retentive memory:

- Memory reset
- Reset to factory settings

You can find an overview of the retentive behavior of the individual memory objects in section Retentive behavior of the memory objects (Page 25).

You can find additional information on the memory reset and reset to factory settings functions in the following manuals:

- System manual S7-1500, ET 200MP Automation System
(<http://support.automation.siemens.com/WW/view/en/59191792>)
- SIMATIC Drive Controller
(<https://support.industry.siemens.com/cs/ww/en/view/109766665>) system manual
- System manual ET 200SP Distributed I/O System
(<http://support.automation.siemens.com/WW/view/en/58649293>)
- Operating instructions ET 200pro CPU 1513pro-2 PN
(<https://support.industry.siemens.com/cs/ww/en/view/109769507>) and
ET 200pro CPU 1516pro-2 PN
(<https://support.industry.siemens.com/cs/ww/en/view/109482416>)

Additional memory areas

Besides the memory areas that have been described for the user program and data, the CPU has additional memory areas available.

The additional memory areas include the following:

- Process images
- Temporary local data

The CPU-specific sizes can be found in the technical specifications for the respective CPU.

2.1.1 Specifics of the CPUs of the redundant system S7-1500R/H

The redundant system S7-1500R/H consists of two CPUs. There must be a SIMATIC memory card in each of the CPUs. We recommend the same memory size for each SIMATIC memory card. During redundant operation, both CPUs execute the user program in parallel. In so doing, one CPU takes the role of the leading CPU (Primary CPU) and one CPU takes the role of the following CPU (Backup CPU). If one CPU fails, the second CPU maintains control over the process.

Memory areas

The two CPUs of the redundant system S7-1500R/H have the same memory areas as the CPUs of non-redundant systems.

You create the hardware configuration of the offline project data only once in STEP 7. STEP 7 loads the offline project data into the current primary CPU. The system synchronizes all data required for redundant operation from the primary CPU to the backup CPU.

Both CPUs receive identical online project data. The online project data contains the hardware configuration of the H-system.

Particularities of the retentive memory

Like for non-redundant CPUs, the retentive memory is a non-volatile memory. The retentive memory saves a limited amount of data in the event of power failure.

Each of the CPUs of the redundant system S7-1500R/H has its own retentive memory. If one of the CPUs changes to POWER OFF and the second CPU is still in RUN, the retentive data of the CPU in RUN continues to be updated. If the remaining CPU also goes from RUN to POWER OFF, the retentive memory of this CPU contains the more up-to-date retentive data. Note the following for when you switch the CPUs back to RUN afterwards.

Note

Characteristics of retentive data after a STOP or POWER OFF of both CPUs

After a STOP or POWER OFF of both CPUs, switch the CPU with the more up-to-date data to RUN first. The more up-to-date data is located in the CPU that controlled the process before STOP or POWER OFF. This procedure gives you certainty that you are using the more up-to-date retentive data. The precondition for this is that you did not change any data using STEP 7 or the HMI while the CPUs were in STOP.

Redundancy ID

Unlike in a non-redundant CPU, an additional memory object is contained in the retentive memory of the respective redundant CPU. Each CPU saves its respective redundancy ID in this memory object. The redundancy IDs can assume the values 1 and 2. Different redundancy IDs are required for redundant operation in order to clearly identify the CPUs and assign the project data to the respective CPUs. For additional information on redundancy IDs of the CPUs, refer to the Redundant System S7-1500R/H (<https://support.industry.siemens.com/cs/ww/en/view/109754833>) system manual.

You can find an overview of the retentive behavior of the individual memory objects in section Retentive behavior of the memory objects (Page 25).

2.2 Memory requirements and memory usage

You can access information regarding the memory areas of the CPUs in the following ways depending on product family used:

Product family	Information regarding the memory areas of the CPU is accessible via:		
	STEP 7	Web server	Display
S7-1500	✓	✓	✓
S7-1500R/H	✓	--	✓
SIMATIC Drive Controller	✓	✓	--
ET 200SP	✓	✓	--
ET 200pro CPU 151xpro-2 PN	✓	✓	--

Memory requirements of the program in the offline project

During creation or modification of a project, the display of memory utilization in STEP 7 shows you the size of the project in the following memories:

- Load memory
- Work memory
- Retentive memory

You can find this information for the CPU under "Program info" in the project tree, "Resources" tab.

- Total size of the memory areas of the respective CPU project (in the figure below line "Total:")
- Memory requirements of the program elements (blocks, data types, objects for motion technology and PLC tags)
- Memory amounts in the respective memory area of the offline project (in figure below, line "Used:")
- Used inputs and outputs

The following figure shows an overview of the utilization of the various memory areas of the "Resources" tab:

Resources of PLC_1					
	Objects	Load memory	Code work-memory	Data work-memory	Retain memory
1		??%	0 %	0 %	0 %
2					
3	Total:	not specified	1048576 bytes	5242880 bytes	484000 bytes
4	Used:	not specified	164 bytes	0 bytes	0 bytes
5	Details	2 MB			
6	▶ OB	4 MB	>164 bytes		
7	FC	12 MB			
8	▶ FB	24 MB	-		
9	▶ DB	256 MB	?		
10	▶ DB	2 GB		-	0 bytes
11	▶ Objects for Motion Technology	32 GB	?	-	0 bytes
12	▶ Data types	43941 bytes			
13	▶ PLC tags	1426 bytes			0 bytes

Figure 2-2 Display of the utilization of the various memory areas

For a CPU, you can select the total size of the load memory in a drop-down list. Select the size of the load memory in accordance with the size of the SIMATIC memory card you are using. The percentage shown in the Load memory column depends on the selected size of the load memory. As soon as the memory size exceeds the size of the load memory of the memory card you are using, the sizes indicated turn red.

Note

Determination of the memory utilization

Note that the SIMATIC memory card also contains data other than the user program that you cannot determine using "Resources". This data includes:

- Hardware configuration
- Recipes, data logs and HMI backups (not supported for S7-1500R/H)
- Non-SIMATIC files, such as PDF, etc.

Therefore, the drop-down list with the size of your SIMATIC memory card only serves as a visual orientation aid.

Also refer to the FAQ "How do you estimate the memory requirements of your project in the load memory of a SIMATIC S7-1500" on the Internet (<https://support.industry.siemens.com/cs/ww/en/view/97553417>).

Note

Display of the memory utilization under "Program info"

The display of the memory utilization in the program information is an offline display in STEP 7 and only shows the memory requirements of the program in the project. The program on the memory card of the CPU may differ, however, for example if the program:

- is more up to date
 - contains blocks generated by other projects
 - contains blocks generated on the CPU
-

Data on the SIMATIC memory card

In addition to the program and the associated program elements (blocks, data types, objects for motion technology and PLC tags), the following data is also stored on the memory card:

- Hardware configuration
- Project information
- Force jobs
- Trace recordings (not supported for S7-1500R/H)
- Symbols and comments

The following further data may also be found on the memory card:

- Recipes, data logs and HMI backups (not supported for S7-1500R/H)
- Non-SIMATIC files that were copied to the memory card via the web server of the CPU or offline in Windows Explorer (for example PDF files, etc.)

Display of the memory utilization in STEP 7

In online mode, the online function "Memory" provides you with the following up-to-date memory information:

- Size of the total free and already allocated load memory on the SIMATIC memory card.
- Size of the total free and already allocated work memory, separated by code and data.
- Size of the total free and already allocated retain memory.

The online function "Memory" can be found in Online & Diagnostics under "Diagnostics > Memory". You can access the functions under Online & Diagnostics in various ways:

- In the project tree under each configured CPU.
- In the project tree under Online access > Accessible devices, in order to display the memory utilization of CPUs that were not configured in the project.
- In all views of the device configuration (topology view, network view, device view) by selecting a CPU with the right mouse button.

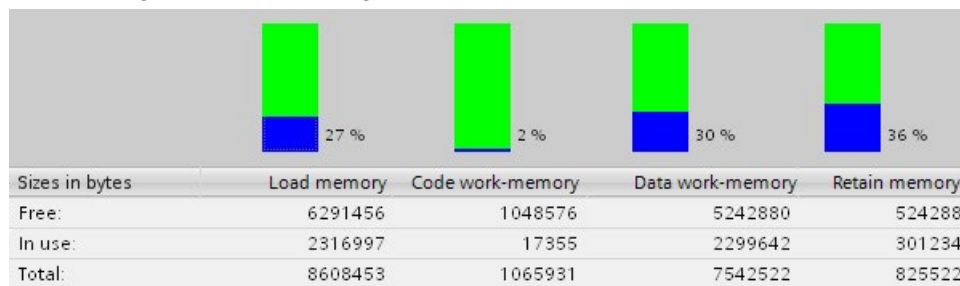


Figure 2-3 "Memory" online function

Note

Fill levels of the CPUs of the redundant system S7-1500R/H

The CPUs of the redundant system S7-1500R/H can have CPU-specific fill levels in non-redundant operation.

The fill levels of the load memory of the CPUs can differ in redundant operation and non-redundant operation (e.g. due to stored PDF files or SIMATIC memory cards of different sizes).

You can have the memory utilization of both CPU 1 and CPU 2 displayed in STEP 7.

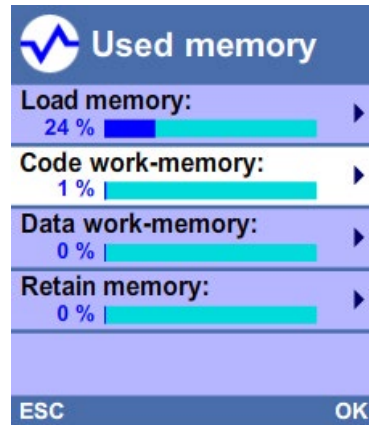
Alternatively to the "Memory" online function, you will also find a display of the current memory functions on the "Online tools" task card in the "Memory" section.

Display of the memory utilization on the display of the CPU

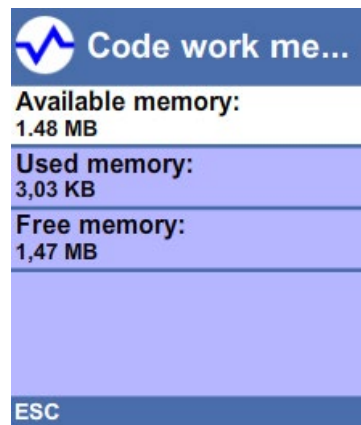
To obtain information about the available memory via the display, proceed as follows:

- Select the "Diagnostics" menu on the display with the help of the arrow keys.
- Select the "Used memory" command from the "Diagnostics" menu.

Under the "Used memory" menu item, you can find information about the utilization of the various memory areas (see following figure). Note that the memory usage is a snapshot of the memory used at the time of the request and is not continuously updated.



To find out details about the respective memory areas (e.g. code work memory), select the required memory area with the help of the arrow keys (see following figure).



In the detail view, e.g. of the code work memory, the display provides you with the following information:

- Memory space which is still available in the code work memory.
- Memory space which is already allocated in the code work memory.
- Total available memory space in the code work memory.

Display of the memory usage in the Web server

On the Web server, you can find information about the current usage of the individual memory areas on the Web page "Diagnostics" in the "Memory" tab.

Detailed information about the use of the Web server can be found in the S7-1500 Web server (<https://support.industry.siemens.com/cs/ww/en/view/59193560>) function manual.



Figure 2-4 Display of the memory usage in the Web server

Note

Redundant system S7-1500R/H

The web server is not supported by the CPUs of the redundant system S7-1500R/H.

2.3 Retentive memory areas

Introduction

The CPUs have a memory for storing retentive data at POWER OFF. You can find the size of the retentive memory in the technical specifications of the respective CPU.

You can find the utilization of the retentive memory of the configured CPU in STEP 7 offline under "Program info > Resources" or online under Online & diagnostics with "Diagnostics > Memory".

If you define data as retentive, its content is retained for the startup of a program after STOP or a power failure.

You can define the following data and objects as retentive:

- Tags of global data blocks
- Tags of instance data blocks of a function block
- Bit memories, timers and counters

Tags of technology objects are retentive, e.g. calibration values of absolute encoders. STEP 7 manages the retentivity of the technology object tags automatically. Hence, you do not have to configure retentivity.

The retentive tags of technology objects are unaffected by a memory reset. You can only delete these retentive tags with a reset to factory settings.

Tags of a global data block

In a global data block, you can define either individual tags from a block or all of its tags collectively as retentive, depending on the setting for the "Optimized block access" attribute:

- "Optimized block access" activated: In the declaration table of the data block, you can define individual tags as retentive.

	Name	Data type	Start value	Retain	Visible in HMI	Comment
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	
2	var1	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	var2	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
4	var3	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	var4	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
6	var5	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figure 2-5 Retentivity setting "Optimized block access" activated

- "Optimized block access" not activated: In the declaration table of the data block, you can only define the retentivity of all tags collectively.

	Name	Data type	Offset	Start value	Retain	Visible in HMI	Comment
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	
2	var6	Bool	...	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	var7	Bool	...	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	var8	Bool	...	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	var9	Bool	...	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	var10	Bool	...	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figure 2-6 Retentivity setting "Optimized block access" not activated

You can find additional information on optimized and non-optimized data blocks in the Programming guideline for S7-1200/S7-1500

(<https://support.industry.siemens.com/cs/de/de/view/90885040/en>).

Tags of an instance data block of a function block

You can define the tags of the instance data block of a function block as retentive in STEP 7. Depending on the setting for the "Optimized block access" attribute, you can define retentivity either for individual tags from a block or for all of its tags collectively:

- "Optimized block access" activated: In the interface of the function block, you can define individual tags as retentive.
- "Optimized block access" not activated: In the instance data block, you can only define the retentivity of all tags collectively.

Creation of a data block in the user program

The instruction "CREATE_DB" is used to create a new data block in the load and/or work memory. For data blocks which you create in the load memory, depending on the selection for the ATTRIB parameter, the generated data block either has the property "retentive" or the property "non-retentive". Setting the retentivity for individual tags is not possible here. The "Optimized block access" attribute is disabled.

You can find additional information on the "CREATE_DB" instruction in the STEP 7 online help under "Programming a PLC > Instructions > Extended instructions > Data block control > CREATE_DB: Create data block".

Note

Redundant S7-1500R/H system

The "CREATE_DB" instruction is not supported by the CPUs of the redundant system S7-1500R/H.

Tags of technology objects

Tags of technology objects are retentive, e.g. calibration values of absolute encoders. STEP 7 manages the retentivity of the technology object tags automatically. Hence, you do not have to configure retentivity.

The retentive tags of technology objects are unaffected by a memory reset. They can be deleted only by a reset to factory settings.

Note

Redundant S7-1500R/H system

Technology objects are not supported by the CPUs of the redundant system S7-1500R/H.

Bit memories, timers and counters

You can define the number of retentive bit memories, timers and counters in STEP 7 in the PLC tag table using the "Retain" button.

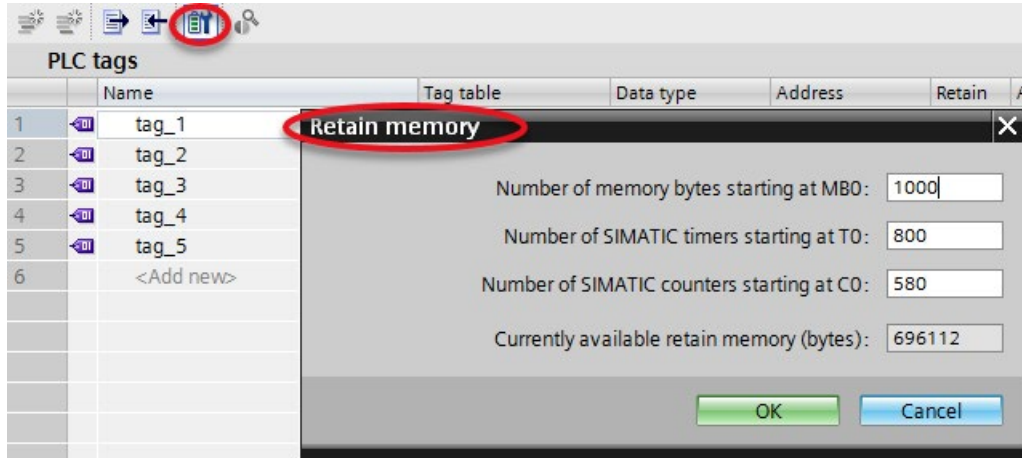


Figure 2-7 Definition of the number of retentive bit memories, timers and counters (beginning at 0, continuing without gaps) using the "Retain" button

Reference

You can find additional information on setting the retentivity in the STEP 7 online help.

2.4 Summary of retentive behavior

2.4.1 Retentive behavior of the memory objects

This section gives an overview of the retentive behavior of the memory objects of the CPUs.

In addition to the retentive memory areas described, there are other objects with retentive characteristics, for example, the diagnostics buffer. These objects do not occupy any storage space in the retentive memory.

The following table shows the retentive behavior of the memory objects in the case of the following:

- STOP → STARTUP
- POWER ON → STARTUP
- POWER ON → STOP
- "Memory reset"
- "Reset to factory settings"

Table 2- 1 Retentive behavior of the memory objects

Memory object	STOP → STARTUP POWER ON → STARTUP POWER ON → STOP	Memory reset	Reset to factory settings
Actual values of the data blocks, instance data blocks	Can be set in the properties of the DB in STEP 7 ¹⁾	-	-
Bit memories, timers and counters - configured as retentive	x	-	-
Bit memories, timers and counters - configured as non-retentive	-	-	-
Retentive tags of technology objects (e.g. calibration values of absolute encoders) ²⁾	x	x	-
Diagnostics buffer entries	x	x	-
Operating hours counter	x	x	-
Clock time	x	x	-
Redundancy ID ³⁾	x	x	-
x = content is retained - = object is initialized			

¹⁾ For DBs with optimized access the retentive behavior is configurable for specific tags.

²⁾ Not for the CPUs of the redundant system S7-1500R/H

³⁾ Only for the CPUs of the redundant system S7-1500R/H

Diagnostics buffer

The 500 most recent entries in the diagnostics buffer are retained after power failure and are not affected by a memory reset. The diagnostics buffer can be cleared only by a reset to factory settings. The entries in the diagnostics buffer do not occupy any memory space in the retentive memory.

Operating hours counter

The operating hours counters of the CPUs are retentive and are not affected by a memory reset. By resetting to factory settings, the operating hours counters are set to zero.

Clock time

The clock time of the CPUs is retentive and is not affected by a memory reset. A reset to factory settings resets the system time to 01.01.2012 00:00:00.

2.5 Memory behavior when loading software changes

Introduction

You can download software changes in STOP and RUN without affecting the actual values of previously loaded tags.

You load changes to the software in STEP 7 (in the project tree and with selected PLC station) under "Download to device > Software (only changes)".

Impact of software changes on PLC tags

You can load the following software changes without affecting the actual values of PLC tags that have already been loaded:

- Name change
- Comment change
- Addition of new tags
- Deletion of tags
- Change of retentivity settings for Retentive memory areas (Page 21)

The actual values are affected by loading the following software changes:

- Data type change
- Change to addresses

Memory reserve of global DBs and instance DBs

Each function block or data block with the "Optimized block access" attribute activated contains, by default, a memory reserve which you can use for subsequent interface changes. The memory reserve is initially not used. When you have compiled and loaded the block, and then observe that you want to reload interface changes, activate the memory reserve. All tags that you subsequently declare will be placed in the memory reserve. During the next load, the new tags are then initialized to their start values. Tags which have already been loaded are not reinitialized.

The setting of the memory reserve can be found in STEP 7 under the data block properties in the "Download without reinitialization" category.

Impacts of software changes on data blocks without memory reserve

If you are not using memory reserve, you can load the following software changes without this reinitializing the actual values of DB tags that have already been loaded:

- Start value change
- Comment change

Impacts of software changes on data blocks with memory reserve

If you are using the memory reserve for data blocks ("Optimized block access" attribute and "Enable download without reinitialization for retentive tags" button selected), you can load the following software changes. The actual values of previously downloaded DB tags are not reinitialized by this.

- Start value change
- Comment change
- Addition of new tags

If the button "Enable download without reinitialization for retentive tags" is deactivated, then all actual values of the data block are reinitialized on the next loading of the following software changes:

- Name change
- Data type change
- Retentivity change
- Deletion of tags
- Changes to the memory reserve settings
- Addition of new tags

Reference

You can find additional information on setting and activating the memory reserve and on downloading block changes in the STEP 7 online help under "Programming a PLC > Compiling and downloading PLC programs > Downloading blocks for S7-1200/1500 > Loading block extensions without reinitialization".

2.6 Memory requirements for downloading software changes

Memory requirements in RUN state

For the consistent and atomic execution of the complete download operation, the CPU requires adequate free memory space in the work memory and on the SIMATIC memory card. The files affected by loading the software changes to the CPU are only deleted after the new files have been created. This SIMATIC memory card therefore requires free memory space corresponding approximately to the space required for all program objects to be loaded on the memory card.

If this amount of memory is not available on your SIMATIC memory card, the following message is displayed in STEP 7 during the download operation to the CPU: "There is insufficient memory on the memory card for this amount of data."

In order to still enable the loading of changes to the CPU in such a case, we recommend one or more of the following options:

- Download in RUN state
 - Delete any files no longer required (e.g. CSV files, panel backups, etc.) on the memory card using the Web server.
 - Preferably, you should download extensive changes in RUN state in multiple steps or perform the download after each change step.

Note

RUN-Redundant Loading to the CPUs of the S7-1500R/H redundant system in the RUN-Redundant system state

Before a write function is performed, the system **does not** check whether there is enough free space on the SIMATIC memory cards of the CPUs for the function. Writing functions are online functions with the PG/PC, for example loading/deleting a block, test functions, loading a modified user program in the RUN-Redundant system state.

If insufficient memory is available on the SIMATIC memory card of a CPU:

- The CPU affected switches to STOP mode.
 - If insufficient memory is available on the SIMATIC memory card of the selected CPU (to which you wish to load), that CPU switches to STOP mode. The other CPU switches to RUN with the previous user program (redundant system → RUN-Solo system state).
 - If insufficient memory is available on the other CPU, that CPU switches to STOP mode. The selected CPU (to which you have loaded) switches to RUN with the modified user program (redundant system → RUN-Solo system state).
- The ERROR LED flashes red (temporary error)
- A corresponding error message is entered in the diagnostics buffer

If after that there is not enough free memory on the SIMATIC memory card of the other CPU either, this CPU remains in RUN. The CPU then responds like a standard CPU.

- Download in STOP state
 - If downloading in several steps is not possible, load extensive changes in STOP state.
- Use a larger memory card
 - In order to perform extensive downloads in RUN state of the CPU in the future, use a larger memory card. A description of how to change the memory card can be found in the section "Changing the memory card without loss of retentive data".

Impact of minor program changes on the load procedure

Under the objects of a program, there are dependencies, e.g.

- Of code blocks to called code blocks
- Of code blocks to data blocks
- Of data blocks to data types (PLC data types, FB types)

This means that the load procedure for a minor change may be very time-consuming if the change affects a large number of dependent objects.

Example:

A STEP 7 program contains an organization block (OB), 20 functions (FC) and a data block (DB). The OB calls the 20 FCs. All the FCs access the DB. If you change the program code in one of the FCs, the following load procedure only contains the changed FC. If you change the data type of a tag in the DB, however, the following load procedure then contains all the FCs and the DB.

The following figure shows the objects in the load procedure in a preview.

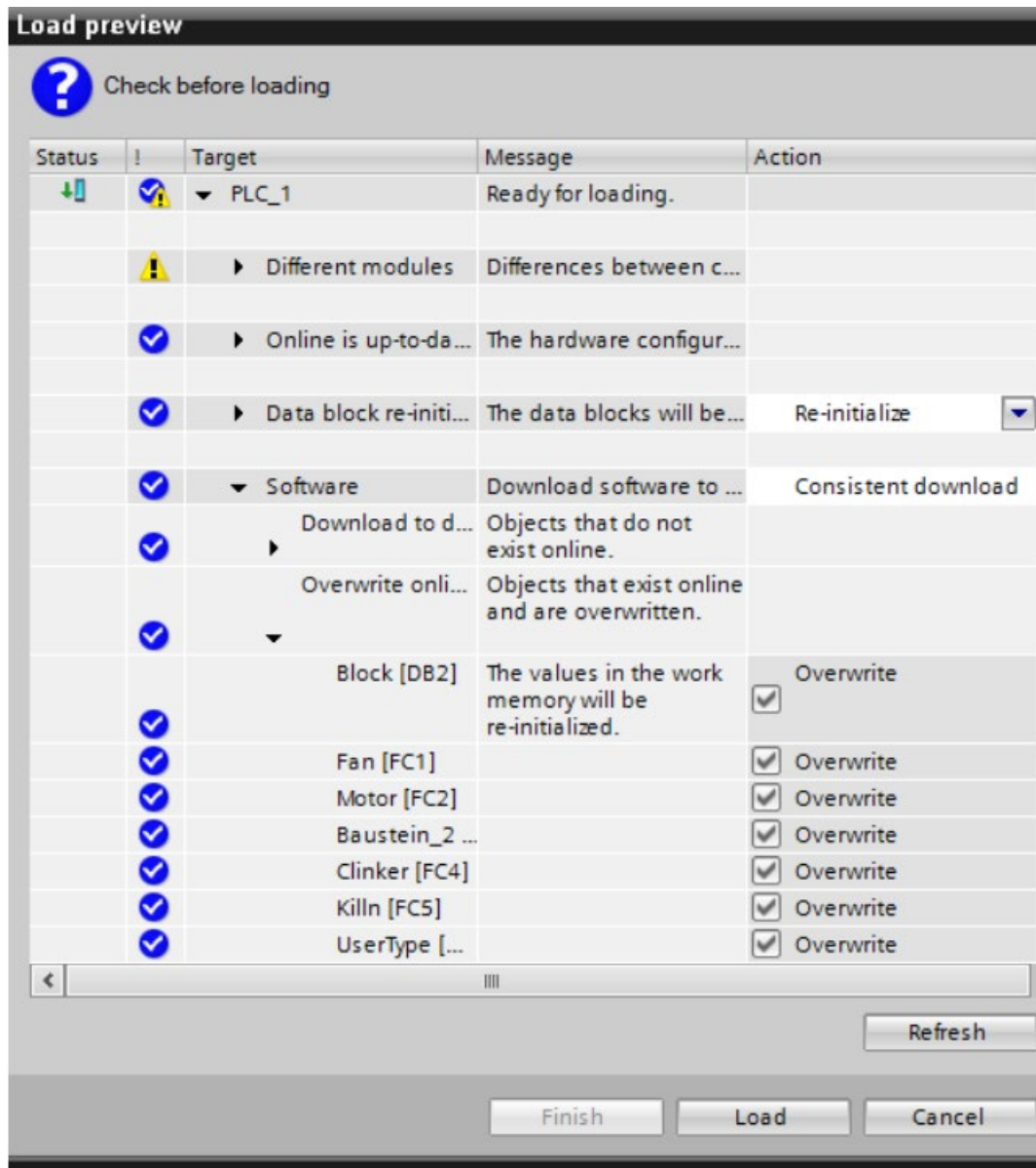


Figure 2-8 Preview for loading

To find the interdependencies of the individual objects, double-click "Program info" in the project tree. Change to the "Dependency structure" tab in the "Program info" dialog.

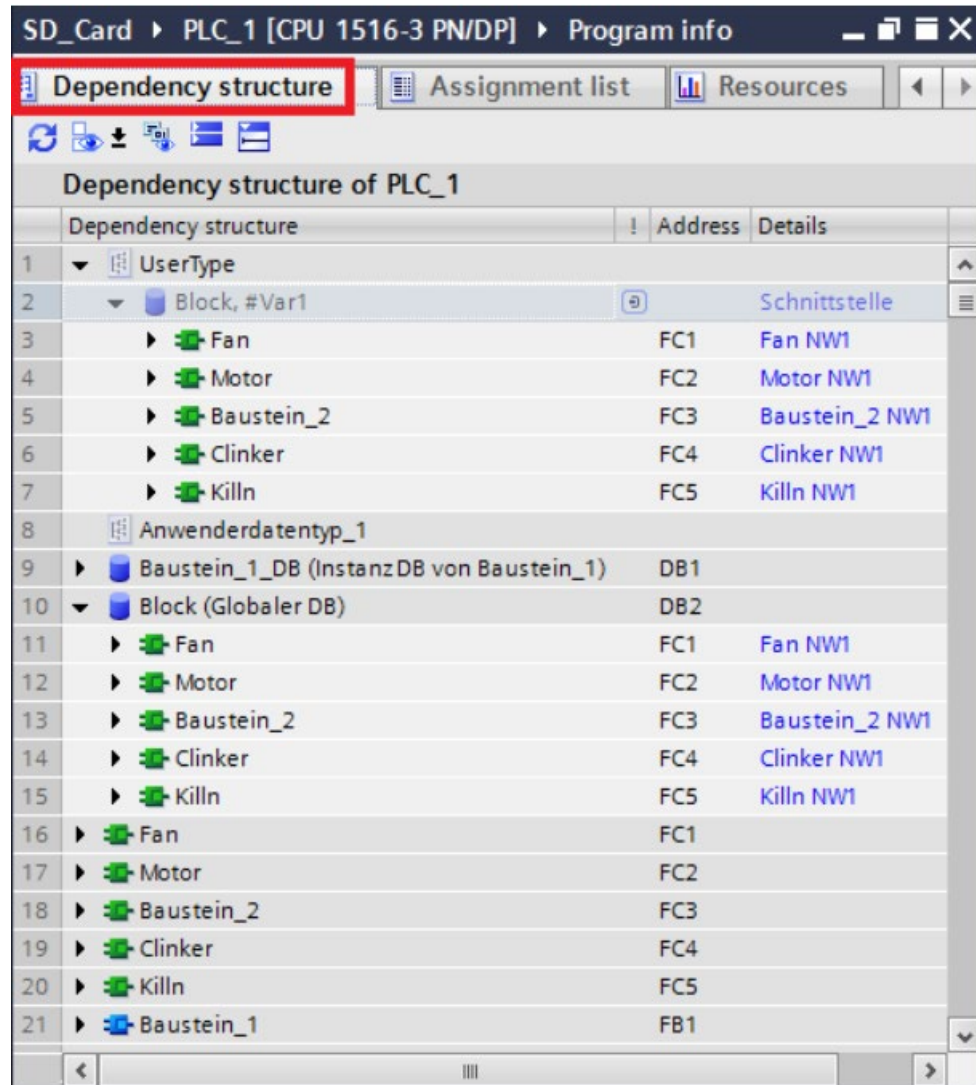


Figure 2-9 Dependency structure

Memory requirements in STOP state

Even when downloading in STOP state, a certain amount of reserve memory is required, because your CPU needs sufficient free memory on the memory card for consistent downloading of individual data blocks. The files affected by loading the data blocks are only deleted after the new files have been created. Therefore, for the modifications you must have available at least the memory requirement of the largest data block.

If insufficient reserve memory is available on your SIMATIC memory card when loading in STOP state, the following message is displayed in STEP 7 during the download to the CPU: "There is insufficient memory on the memory card for this amount of data."

In order to still enable the loading of changes to the CPU in such a case, we recommend one or more of the following options:

- Delete any files no longer required (e.g. CSV files, panel backups, etc.) from the memory card using the Web server.
- Use a larger memory card. A description of how to change the memory card can be found in the section "Changing the memory card without loss of retentive data".

Note

Please note that retentive data and possibly also project data is lost with the following three options. Therefore only use the following options described if the two options described previously did not lead to the desired result.

- In STEP 7, download your program to the CPU with menu command "Online > Download and reset PLC program".
- Remove the memory card from the slot of the CPU. Delete the content that is longer needed from the memory card with your programming device.
- Delete the entire contents, e.g. by formatting the memory card. A description of how to format the memory card can be found in the section "Formatting a SIMATIC memory card".

Note also the following FAQs:

- "When downloading to the S7-1500 CPU, why is the message "There is insufficient memory on the memory card for this amount of data" displayed although there is still enough memory available?" on the Internet (<https://support.industry.siemens.com/cs/ww/en/view/107108015>).
- "Why can you not load the project data into the load memory of the S7-1500 CPU when the number of alarms and messages is too large?" on the Internet (<https://support.industry.siemens.com/cs/ww/en/view/109751485>).

Memory usage and application examples

3.1 Memory usage for data blocks

Processing sequence

- **Storing data records in the load memory**

STEP 7 stores individual data records in a non-runtime-relevant DB and downloads the DB to the CPU. In order to configure a non-runtime-relevant DB, you must enable the "Only store in load memory" block attribute. The data records then only use memory space in the load memory and not in the work memory.

- **Working with data record data in the user program**

You use the "READ_DBL" instruction to copy a data record from the DB in the load memory to a runtime-relevant DB in the work memory. As a result, the work memory only has to accommodate the data for the data record currently required. The user program can now access the data of the current data record.

- **Saving back changed recipe data records**

The "WRIT_DBL" instruction writes new or changed data records from the user program back to the load memory. The data written to the load memory is portable and not affected by a memory reset. To back up changed data records, you need to upload the data blocks and back them up on the PG/PC.

You can find information on uploading data blocks in the STEP 7 online help under "Programming a PLC>Compiling and downloading blocks>Downloading blocks for S7-1200/1500>Uploading blocks from a memory card".

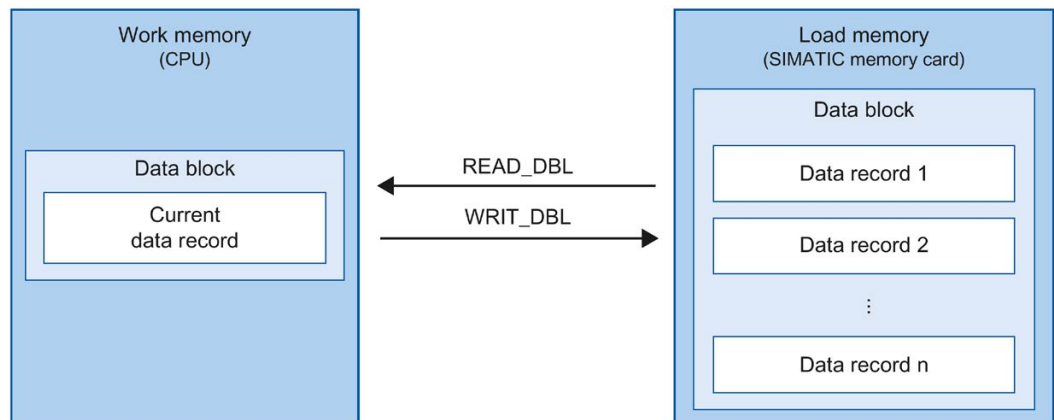


Figure 3-1 Processing sequence with "READ_DBL" and "WRIT_DBL"

3.1 Memory usage for data blocks

Note

Redundant system S7-1500R/H

The data block functions are not supported by the CPUs of the S7-1500R/H redundant system. You therefore cannot use non-runtime-relevant DBs.

Please also note the FAQ "How do you configure data blocks with the "Only store in load memory" attribute for the S7-1200/S7-1500?" on the Internet (<https://support.industry.siemens.com/cs/ww/en/view/53034113>).

Note

Instructions that access the SIMATIC memory card have a lower performance than instructions that access the work memory. The associated blocks (e.g. READ_DBL and WRIT_DBL) are therefore asynchronous. Their execution extends if necessary over several cycles.

NOTICE

Service life of the SIMATIC memory card
--

Only a limited number of delete and write operations are possible on the SIMATIC memory card. After expiration of the service life, there is a risk that the card can no longer be used.
--

You can find additional information on the service life of the SIMATIC memory card in Service life of the SIMATIC memory card (Page 63).
--

Note

Memory size of the SIMATIC memory card

Use a SIMATIC memory card with sufficient memory for your particular purposes.

3.2 Memory usage for recipes

Introduction

A recipe is a collection of parameter sets with the same structure. These recipe data records are located in a data block in the load memory, and do not occupy any storage space in the work memory. You have the option of reading individual recipe data records to a data block in the work memory and accessing the data in the user program. You can write a recipe data record that has been changed in the user program back to the recipe data block.

Recipes contain, for example, the related data of a particular batch in production. You can export recipe data records of a recipe DB as a csv file. A web browser can read data via the web server of the CPU, even when the CPU is in STOP state. You can also directly access the data of the SIMATIC memory card using a card reader on the programming device.

Note**Redundant system S7-1500R/H**

Recipes are not supported by the CPUs of the S7-1500R/H redundant system.

Import and export of recipe data

You have the option of exporting recipe data records of a recipe DB as a CSV file, and importing them from a CSV file into a DB. The CSV file is located in the "\recipes" directory on the SIMATIC memory card. You can open and process this file further with a spreadsheet program such as Microsoft Excel.

You can easily work with CSV files on the SIMATIC memory card via the web server of the CPU (e.g. rename, save to hard disk, delete, etc.). To avoid undesired manipulation, set access rights for the web server in STEP 7. You can find additional information on the Web server in the Web Server (<http://support.automation.siemens.com/WW/view/en/59193560>) function manual.

- **Export of recipe data**

The "RecipeExport" instruction exports all the recipe data records of a recipe DB from the load memory to a CSV file on the SIMATIC memory card. The CSV file has the same name as the recipe of the DB. The CSV file is stored in the "\recipes" directory on the SIMATIC memory card.

The "RecipeExport" instruction only exports valid and unencrypted recipe data records.

- **Import of recipe data**

The "RecipeImport" instruction imports all recipe data records from the CSV file into the recipe DB in the load memory. The name of the CSV file must match the name of the recipe DB.

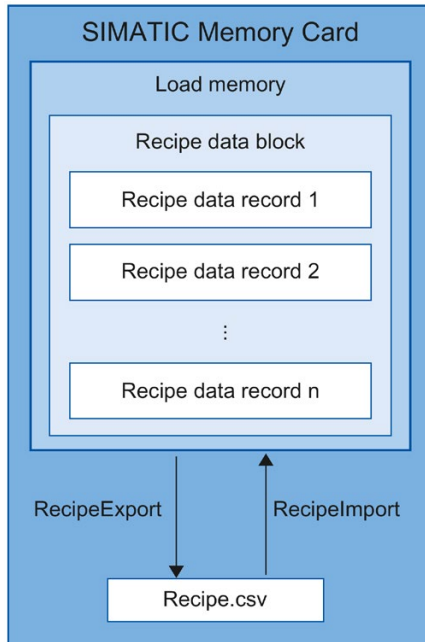


Figure 3-2 Import and export of recipe data

Note

Asynchronous instructions

Please note that the "RecipeExport" and "RecipeImport" instructions are asynchronous instructions.

In contrast to synchronous instructions, this means the execution of an asynchronous instruction can extend over multiple calls before execution is completed. The CPU processes asynchronous instructions in parallel with the cyclic user program.

A CPU can process several asynchronous instruction jobs in parallel. The CPU can process a maximum of 10 jobs of the instructions listed in parallel.

You can find additional information about asynchronous instructions in the following manuals:

- System manual S7-1500, ET 200MP (<http://support.automation.siemens.com/WW/view/en/59191792>)
- System manual Redundant System S7-1500R/H (<https://support.industry.siemens.com/cs/ww/en/view/109754833>)
- SIMATIC Drive Controller (<https://support.industry.siemens.com/cs/ww/en/view/109766665>) system manual
- System manual ET 200SP Distributed I/O System (<http://support.automation.siemens.com/WW/view/en/58649293>)
- Operating instructions ET 200pro CPU 1513pro-2 PN (<https://support.industry.siemens.com/cs/ww/en/view/109769507>) and ET 200pro CPU 1516pro-2 PN (<https://support.industry.siemens.com/cs/ww/en/view/109482416>)

Additional information

You can find additional information on the instructions for recipes in the STEP 7 online help under "Programming a PLC > Instructions > Instructions (S7-1200, S7-1500) > Extended instructions > Recipes and data logging > Recipe functions".

Note also the application example "Using recipe functions for persistent data with SIMATIC S7-1200 and S7 1500" on the Internet (<https://support.industry.siemens.com/cs/ww/en/view/109479727>).

3.3 Memory usage for data logging

3.3.1 Overview of data logging

With data logging, you save selected process values from the user program in a file, the data log. The data logs are saved on the SIMATIC memory card in csv format and stored in the "\datalogs" directory. A web browser can read data via the web server of the CPU, even when the CPU is in STOP state. You can also directly access the data of the SIMATIC memory card using a card reader on the programming device.

NOTICE

Service life of the SIMATIC memory card

Only a limited number of delete and write operations are possible on the SIMATIC memory card. Cyclic write operations to the SIMATIC memory card by the user program reduce the service life of the SIMATIC memory card. After expiration of the service life, there is a risk that the card can no longer be used. Therefore, use a SIMATIC memory card with sufficient memory for your particular purposes.

Information on the service life of the SIMATIC memory card can be found in the section Service life of the SIMATIC memory card (Page 63).

Note

Memory size of the SIMATIC memory card

Use a SIMATIC memory card with sufficient memory for your particular purposes.

The "data logging" instructions can be used in your program to create, open, write, close or delete data logs. You decide which tags are logged by creating a data block that defines a single data log data record. Your data block is used as temporary storage for a new data log data record. New current values for the tags must be transferred into the data block during runtime by means of user program instructions. If all tag values have been updated, you can execute the "DataLogWrite" instruction, in order to transfer data from the data block into the data log.

3.3 Memory usage for data logging

You manage your data logs with the integrated web server. You can download data logs from the standard "File Browser" website. After you have transferred a data log to your PC, you can analyze the data using popular spreadsheet programs, e.g. Microsoft Excel.

Note

Redundant system S7-1500R/H

Data logging is not supported by the CPUs of the redundant system S7-1500R/H.

The following figure shows the basic sequence for creating a data log:

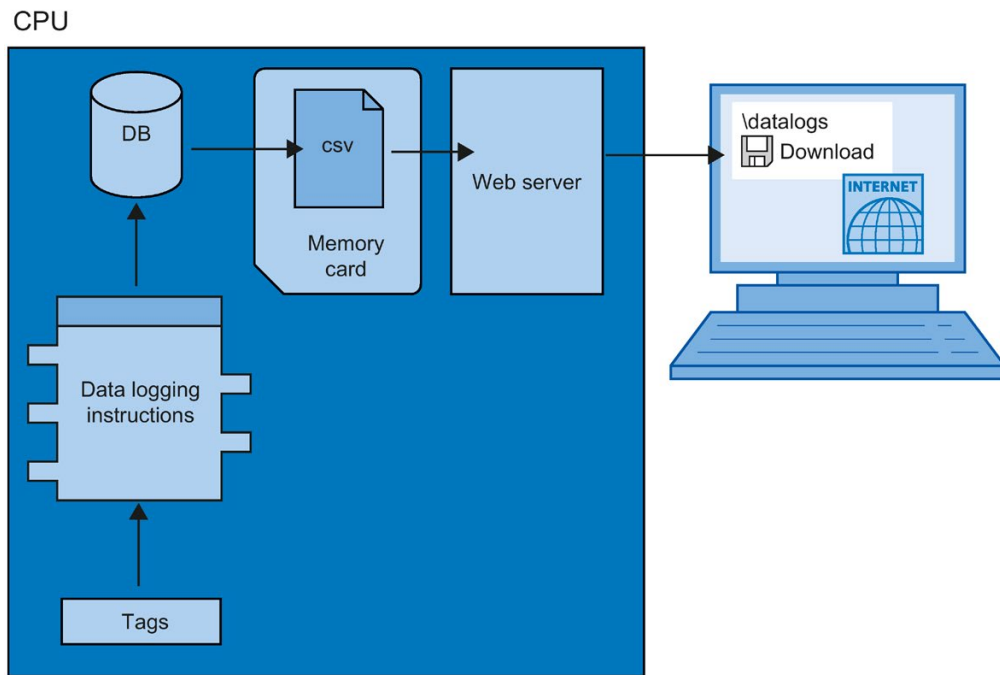


Figure 3-3 Basic sequence during the creation of a data log

3.3.2 Data structure of the data logs

Introduction

You use the "DataLogCreate" instruction to create a data log in STEP 7. The NAME parameter assigns the data log a name. The DATA and HEADER parameters specify the data type of all data elements in a data log data record, and the header line of the data log. The RECORDS parameter indicates the maximum number of records in the data log.

NAME parameter for the "DataLogCreate" instruction

You use the NAME parameter to assign a name for the data log. This is the name under which the data log is saved in the "\datalogs" directory of the SIMATIC memory card.

DATA parameter for the "DataLogCreate" instruction

The DATA block parameter specifies the structure of the data log's records. The columns and data types of a data record in the data log are determined by the elements of the structure declaration or array declaration of this data buffer. Each element of a structure or array corresponds to a column in a row in the data log.

HEADER parameter for the "DataLogCreate" instruction

Using the HEADER block parameter, you can assign a heading in the header row to each column in the data log.

RECORDS parameter for the "DataLogCreate" instruction

The RECORDS parameter specifies the maximum number of records that can be stored in a data log. When the specified maximum number of data records in a data log is reached, the next write operation overwrites the oldest data record.

3.3.3 Instructions for data logging

Overview

The following table gives an overview of the instructions for data logging. You will find the data logging instructions in "STEP 7" in the "Instructions" task card, under "Extended instructions > Recipe and data logging > Data Logging".

Table 3- 1 Overview of the data logging instructions

Name of the instruction	Description
"DataLogCreate": Create data log	With the "DataLogCreate" instruction you create a data log. The data log is saved on the SIMATIC memory card in the "\datalogs" directory. You can use the data logging instructions to save process data. The amount of data in a data log depends on the available memory space on the SIMATIC memory card.
"DataLogOpen": Open data log	With the "DataLogOpen" instruction, you open an existing data log on the SIMATIC memory card. A data log must be open before you can write new data records to it. The data log opens automatically when the "DataLogCreate" and "DataLogNewFile" instructions are executed. A maximum of 10 data logs can be open at any one time. The data log to be opened can be selected using the ID or name of the data log. The maximum file size of data logs is 2 GB. A maximum number of 1000 data log files is possible with firmware version V2.0.
"DataLogWrite": Write data log	With the "DataLogWrite" instruction you write a data record into an existing data log. Use the ID parameter to select the data log to which the data record is to be written. To write a new data record, the data log must be open.
"DataLogClose": Close data log	With the "DataLogClose" instruction, you close an open data log. You select the data log using the ID parameter. When the CPU changes to STOP state, all open data logs are closed.
"DataLogNewFile": Data log in new file	With the "DataLogNewFile" instruction you create a new data log. The new data log has the same properties as an already existing data log. By creating a new data log, you prevent cyclic overwriting of existing data records. When the instruction is called it creates a new data log on the SIMATIC memory card using the name defined in the NAME parameter. You use the ID parameter to specify the ID of the old data log whose properties you want to apply to the new data log. The ID parameter then outputs the ID of the new data log.
"DataLogClear": Empty data log	The "DataLogClear" instruction deletes all data records in an existing data log. The header of the data log is not deleted (see description of parameter Data structure of the data logs (Page 39)).
"DataLogDelete": Delete data log	The "DataLogDelete" instruction is used to delete a data log from the SIMATIC memory card. Select the data log to be deleted using the NAME and ID parameters.

Note**Asynchronous instructions**

Note that the instructions in the table are asynchronous instructions.

In contrast to synchronous instructions, this means the execution of an asynchronous instruction can extend over multiple calls before execution is completed. The CPU processes asynchronous instructions in parallel with the cyclic user program.

A CPU can process several asynchronous instruction jobs in parallel. The CPU can process a maximum of 10 jobs of the instructions listed in the table in parallel.

You can find additional information about asynchronous instructions in the following manuals:

- System manual S7-1500, ET 200MP
(<http://support.automation.siemens.com/WW/view/en/59191792>)
 - System manual ET 200SP Distributed I/O System
(<https://support.industry.siemens.com/cs/ww/en/view/58649293>)
 - SIMATIC Drive Controller
(<https://support.industry.siemens.com/cs/ww/en/view/109766665>) system manual
 - Operating instructions ET 200pro CPU 1513pro-2 PN
(<https://support.industry.siemens.com/cs/ww/en/view/109769507>) and
ET 200pro CPU 1516pro-2 PN
(<https://support.industry.siemens.com/cs/ww/en/view/109482416>)
-

3.3.4 Example program for data logging

This example program shows the storing of process values for counter content, temperature, and pressure in a data log.

The example shows the basic functioning of the instructions for data logs. The complete program logic is not shown.

Note**General use of data logs**

- Data logs are automatically opened after execution of the "DataLogCreate" and "DataLogNewFile" instructions.
 - Data logs are automatically closed at a change of the CPU from RUN to STOP or a restart of the CPU.
 - A data log must be open so that data can be written to the data log with the "DataLogWrite" instruction.
 - A maximum of 10 data logs can be open at any one time, even when more than 10 data logs exist.
-

Tags of the data block

The following figure shows the tags of the "My_Datalog_Vars" data block. These tags are used by the "Data logging" instructions "DataLogCreate" and "DataLogNewFile". The "MyDataLogName" and "MyNEWDataLogName" tags are called in the NAME block parameter, and give the data logs a name. The "MyData" structure is called in the DATA block parameter and specifies the structure of the csv file. The three MyData tags temporarily store new values. The tag values at these DB addresses are transferred to a data log using the "DataLogWrite" instruction. The "MyDataLogHeaders" tag is called in the HEADER block parameter and specifies a header for the data log.

My_Datalog_Vars			
	Name	Data type	Start value
1	Static		
2	MyNEWDataLogName	String	'MyNEWDataLog'
3	MyDataLogName	String	'MyDataLog'
4	MyDataLogID	DWord	0
5	MyDataLogHeaders	String	'Count.Temperature.Pressure'
6	MyData	Struct	
7	MyCount	Int	0
8	MyTemperature	Real	0.0
9	MyPressure	Real	0.0

Figure 3-4 Declaration table with the data block's tags

Network 1

A rising edge at REQ starts the creation of the data log.

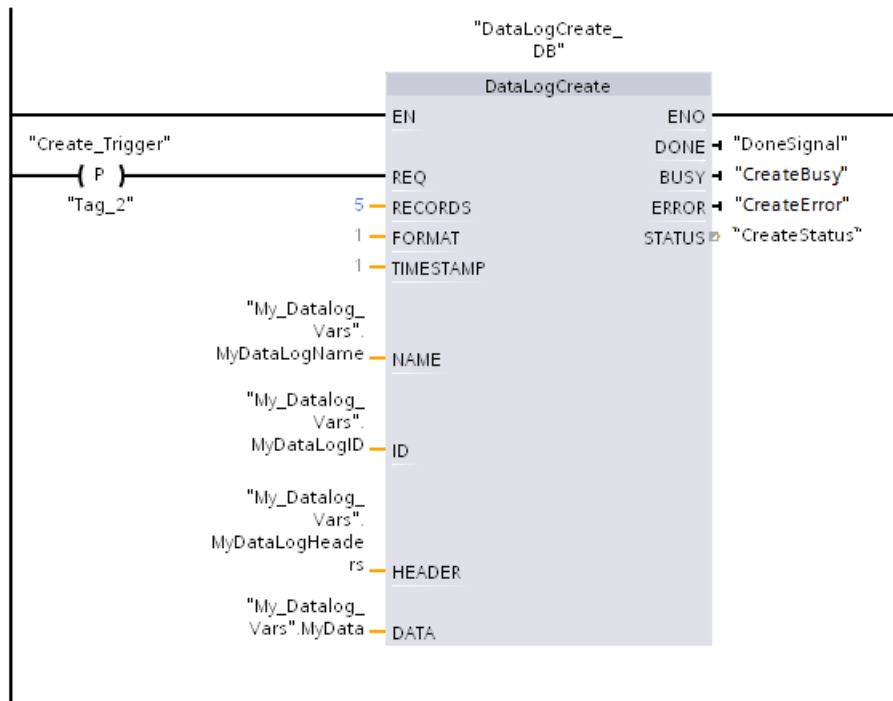


Figure 3-5 Network 1

Network 2

Detect the output DONE of "DataLogCreate", because after the execution of "DataLogCreate" it is only set to 1 for one cycle.

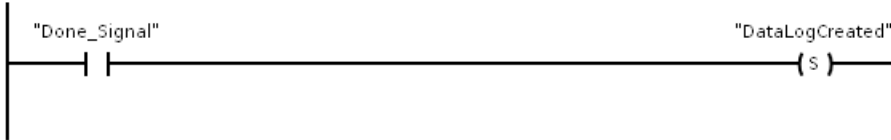


Figure 3-6 Network 2

Network 3

A rising edge triggers the point in time at which new process values are stored in the MyData structure.

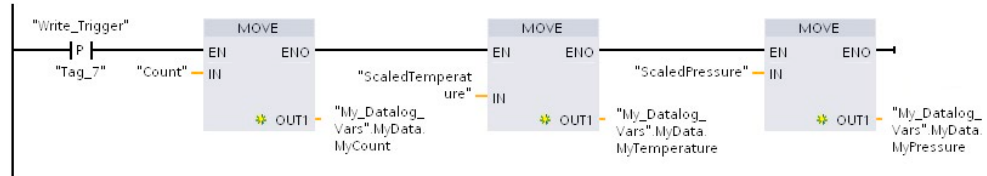


Figure 3-7 Network 3

Network 4

The state of the input EN is based on the point in time at which the execution of "DataLogCreate" was completed. One execution of "DataLogCreate" extends over multiple cycles, and must be completed before a write operation is executed. The rising edge at input REQ is the event that triggers an activated write operation.

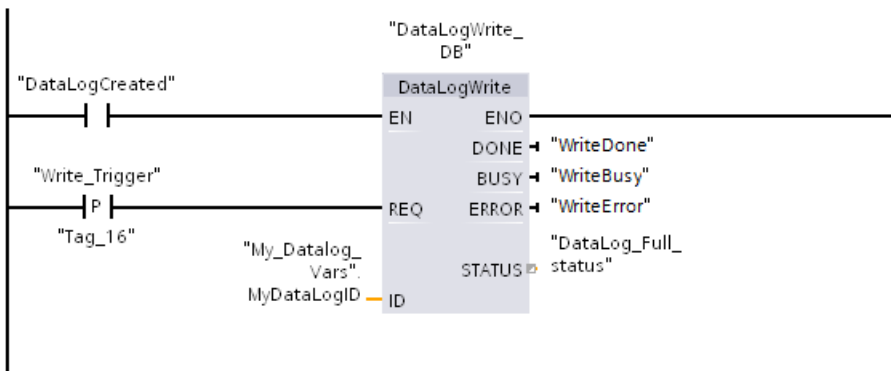


Figure 3-8 Network 4

3.3 Memory usage for data logging

Network 5

Close the data log after the last data record has been written. After execution of the "DataLogWrite" instruction, which writes the last data record, the STATUS output is set to "1".

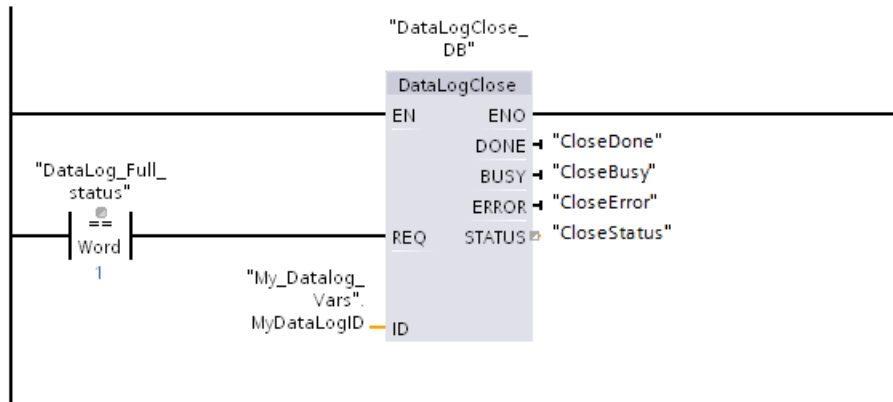


Figure 3-9 Network 5

Network 6

A rising edge at the input REQ of the instruction "DataLogOpen" simulates that the user presses a button on an HMI device, which opens a data log. If you open a data log in which all records are occupied by process data, then the next execution of the "DataLogWrite" instruction overwrites the oldest data record. You can however also preserve the old data log, and create a new data log instead. This is shown in network 7.

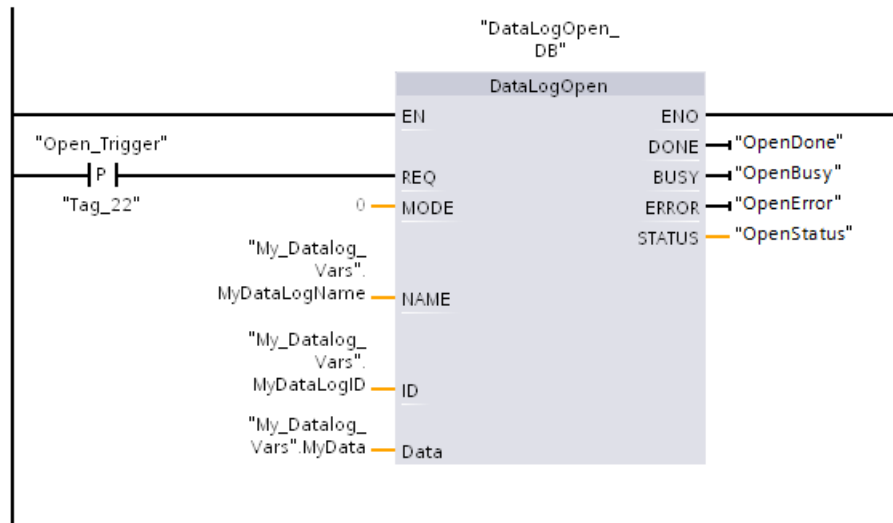


Figure 3-10 Network 6

Network 7

The ID parameter is an IN/OUT type. First you indicate the ID value of the existing data log whose structure you want to copy. After the "DataLogNewFile" instruction has been executed, a new and unique ID value for the new data log is written back into the address of the ID reference. The required detection DONE bit = TRUE is not shown. An example for the logic of the DONE bit can be found in networks 1, 2 and 4.

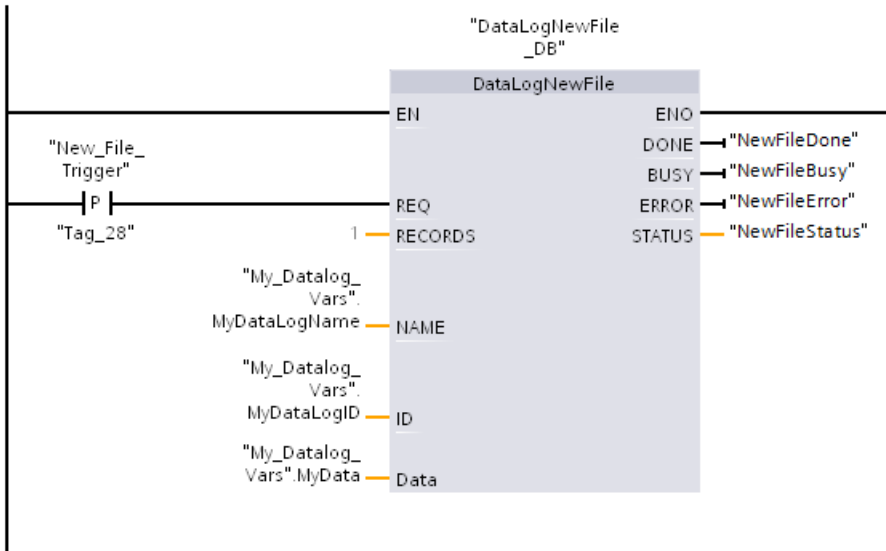


Figure 3-11 Network 7

The data logs created in the example program can be found on the CPU Web server's standard "File browser" Web page in the "datalogs" folder.

The following figure shows the standard Web page of the Web server using the example of the CPU 1516-3 PN/DP.

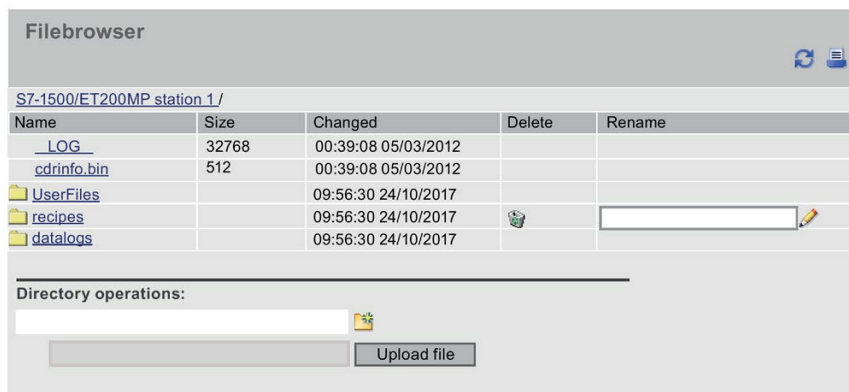



Figure 3-12 Standard "File browser" web page of the Web server

3.3 Memory usage for data logging

In the file browser, you can download the data logs created in the example program. It is not possible to delete or rename the data logs in the Web server. To delete a data log you either use the "DataLogDelete" instruction or you format the SIMATIC memory card.

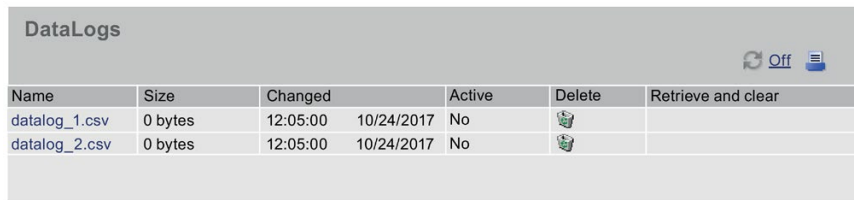
On the DataLogs web page, you can have all the data logs that you created displayed. You can call and empty the relevant data log file by clicking the icon .

Note

Manipulation of the data logs using a card reader

Do not delete or change the data logs using a card reader on the PG/PC. You can copy the data logs on the SIMATIC memory card, however, using a card reader on the PG/PC.

The recommended medium for viewing, downloading (copying) and deleting data logs is the file browser of the web server, however. Direct file access via Windows Explorer brings with it the risk of inadvertent deletion or modification of data logs or system files. This can cause the files to be damaged or the SIMATIC memory card to become unusable.



The screenshot shows a web interface titled "DataLogs" with a refresh button and "Off" status. Below is a table with columns: Name, Size, Changed, Active, Delete, and Retrieve and clear. Two rows of data are shown: "datalog_1.csv" and "datalog_2.csv", both 0 bytes, changed at 12:05:00 on 10/24/2017, and marked as "No" in the Active column. Each row has a trash icon in the Delete column.



Name	Size	Changed	Active	Delete	Retrieve and clear
datalog_1.csv	0 bytes	12:05:00 10/24/2017	No		
datalog_2.csv	0 bytes	12:05:00 10/24/2017	No		

Figure 3-13 Example - Data logs in the folder "datalogs" of the file browser

Table 3- 2 Downloaded examples of data logs displayed in Microsoft Excel

Two data records written in a data log which contains a maximum of five data records.	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>1</td> <td>1/3/2012</td> <td>10:16:47</td> <td>5</td> <td>5,00E+00</td> <td>5,00E+00</td> </tr> <tr> <td>3</td> <td>2</td> <td>1/3/2012</td> <td>10:16:48</td> <td>5</td> <td>5,00E+00</td> <td>5,00E+00</td> </tr> <tr> <td>4</td> <td>//END</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>5</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	Time	Count	Temperature	Pressure	2	1	1/3/2012	10:16:47	5	5,00E+00	5,00E+00	3	2	1/3/2012	10:16:48	5	5,00E+00	5,00E+00	4	//END						5																				
	A	B	C	D	E	F																																																			
1	Record	Date	Time	Count	Temperature	Pressure																																																			
2	1	1/3/2012	10:16:47	5	5,00E+00	5,00E+00																																																			
3	2	1/3/2012	10:16:48	5	5,00E+00	5,00E+00																																																			
4	//END																																																								
5																																																									
Five data records written in a data log which contains a maximum of five data records.	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>1</td> <td>1/3/2012</td> <td>10:08:23</td> <td>1</td> <td>9,86E+01</td> <td>3,52E+01</td> </tr> <tr> <td>3</td> <td>2</td> <td>1/3/2012</td> <td>10:08:39</td> <td>2</td> <td>1,00E+02</td> <td>3,73E+01</td> </tr> <tr> <td>4</td> <td>3</td> <td>1/3/2012</td> <td>10:08:54</td> <td>3</td> <td>9,99E+01</td> <td>3,68E+01</td> </tr> <tr> <td>5</td> <td>4</td> <td>1/3/2012</td> <td>10:09:11</td> <td>4</td> <td>9,95E+01</td> <td>3,64E+01</td> </tr> <tr> <td>6</td> <td>5</td> <td>1/3/2012</td> <td>10:09:28</td> <td>5</td> <td>9,92E+01</td> <td>3,74E+01</td> </tr> <tr> <td>7</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	Time	Count	Temperature	Pressure	2	1	1/3/2012	10:08:23	1	9,86E+01	3,52E+01	3	2	1/3/2012	10:08:39	2	1,00E+02	3,73E+01	4	3	1/3/2012	10:08:54	3	9,99E+01	3,68E+01	5	4	1/3/2012	10:09:11	4	9,95E+01	3,64E+01	6	5	1/3/2012	10:09:28	5	9,92E+01	3,74E+01	7						
	A	B	C	D	E	F																																																			
1	Record	Date	Time	Count	Temperature	Pressure																																																			
2	1	1/3/2012	10:08:23	1	9,86E+01	3,52E+01																																																			
3	2	1/3/2012	10:08:39	2	1,00E+02	3,73E+01																																																			
4	3	1/3/2012	10:08:54	3	9,99E+01	3,68E+01																																																			
5	4	1/3/2012	10:09:11	4	9,95E+01	3,64E+01																																																			
6	5	1/3/2012	10:09:28	5	9,92E+01	3,74E+01																																																			
7																																																									
After another data record has been written, the 6th write operation overwrites the oldest data record (record 1) with data record 6. Another write operation overwrites data record 2 with data record 7, etc.	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>6</td> <td>1/3/2012</td> <td>10:09:42</td> <td>6</td> <td>9,87E+01</td> <td>3,58E+01</td> </tr> <tr> <td>3</td> <td>2</td> <td>1/3/2012</td> <td>10:08:39</td> <td>2</td> <td>1,00E+02</td> <td>3,73E+01</td> </tr> <tr> <td>4</td> <td>3</td> <td>1/3/2012</td> <td>10:08:54</td> <td>3</td> <td>9,99E+01</td> <td>3,68E+01</td> </tr> <tr> <td>5</td> <td>4</td> <td>1/3/2012</td> <td>10:09:11</td> <td>4</td> <td>9,95E+01</td> <td>3,64E+01</td> </tr> <tr> <td>6</td> <td>5</td> <td>1/3/2012</td> <td>10:09:28</td> <td>5</td> <td>9,92E+01</td> <td>3,74E+01</td> </tr> <tr> <td>7</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	Time	Count	Temperature	Pressure	2	6	1/3/2012	10:09:42	6	9,87E+01	3,58E+01	3	2	1/3/2012	10:08:39	2	1,00E+02	3,73E+01	4	3	1/3/2012	10:08:54	3	9,99E+01	3,68E+01	5	4	1/3/2012	10:09:11	4	9,95E+01	3,64E+01	6	5	1/3/2012	10:09:28	5	9,92E+01	3,74E+01	7						
	A	B	C	D	E	F																																																			
1	Record	Date	Time	Count	Temperature	Pressure																																																			
2	6	1/3/2012	10:09:42	6	9,87E+01	3,58E+01																																																			
3	2	1/3/2012	10:08:39	2	1,00E+02	3,73E+01																																																			
4	3	1/3/2012	10:08:54	3	9,99E+01	3,68E+01																																																			
5	4	1/3/2012	10:09:11	4	9,95E+01	3,64E+01																																																			
6	5	1/3/2012	10:09:28	5	9,92E+01	3,74E+01																																																			
7																																																									

3.3.5 Calculation of the data log size

The maximum memory size is allocated when you create the data log. Besides ensuring enough memory space for all the data records, you must take into account the memory space of the following elements for the memory allocation:

- Data log header (if used)
- Time stamp header (if used)
- Data record index header
- Minimum block size

The formula below represents a method to help you determine the estimated size of your data log. Make sure that you observe the rule for the maximum size.

Data bytes of the data log = ((data bytes in a data record + time stamp bytes + 12 bytes) * number of data records)

Header

Header bytes of the data log = header character bytes + 2 bytes

Header character bytes

- No data header and no time stamp = 7 bytes
- No data header and time stamp (with time stamp header) = 21 bytes
- Data header and no time stamp = Number of character bytes in all column headers including separating commas
- Data header and time stamp (with time stamp header) = Number of character bytes in all column headers including separating commas + 21 bytes

Data

Data bytes of the data log = ((data bytes in a data record + time stamp bytes + 12 bytes) * number of data records)

Data bytes in a data record

The DATA parameter of the "DataLogCreate" instruction points to a structure. The structure assigns the number of data fields and the data type of each data field for a data log data record.

Multiply the number of the respective data type with the number of bytes required for this data type. Repeat this step for each data type in a data record. Add all the data bytes to obtain the sum of all the data elements in a data record.

Size of the individual data types

The data in data logs is saved as character bytes in csv format (comma separated values). The table below shows the number of bytes that are required to save each data type.

Data type	Bytes
Any	10
Bool	1
Byte	4
Char	1
Date	10
DInt	12
DTL	31
DWORD	11
Int	7
LDT	31
LReal	25
Real	16
Sint	5

Data type	Bytes
String	<p>Example 1: MyString String[10] The maximum character string size is specified with 10 characters.</p> <ul style="list-style-type: none"> Text character + automatic filling with spaces = 10 bytes Quotation marks at the start and end + comma character = 3 bytes <p>10 + 3 = 13 bytes total</p> <p>Example 2: Mystring2 String If no size is specified in square brackets, 254 bytes are assigned by default.</p> <ul style="list-style-type: none"> Text character + automatic filling with spaces = 254 bytes Quotation marks at the start and end + comma character = 3 bytes <p>254 + 3 = 257 bytes total</p>
Time	14
Tod	12
UDInt	12
UInt	7
USInt	5
WChar	1
Word	6

Number of data records in a data log

The RECORDS parameter of the "DataLogCreate" instruction specifies the maximum number of data records that can be stored in a data log.

Time stamp bytes in a data record

- No time stamp = 0 bytes
- Time stamp = 22 bytes

3.3 Memory usage for data logging

Example for size of a CSV file

The figure "Open CSV file" shows a CSV file opened in a spreadsheet program with five written data records in one data log.

The figure "Size of the header and the data records" shows the size of the header used in the CSV file and the size of the individual data records on the SIMATIC memory card.

The figure "Column size" shows the size of the respective columns depending on the data type used.

Open CSV file							Size of the header and the data records	
1								
2	Record	Date	UTC Time	Count	Temperature	Pressure		
3	1	9/30/2010	20:26:56	1	9.86E+01	3.52E+01	49 (42 characters + 5 commas + 2 line ending)	
4	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01	79 (72 + 5 commas + 2 line ending)	
5	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01	79 (72 + 5 commas + 2 line ending)	
6	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01	79 (72 + 5 commas + 2 line ending)	
7	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01	77 (72 + 5 commas)	
8							Total = 442	
Column size								
9	Size of each column:							
10	index	Date	Time	int	real	real		
11	11	10	12	7	16	16		

Each data record also has a separator comma. Therefore, the calculation of the total size per column also includes 1 byte for each separator comma.

Note that the calculation of the size of the header, the data records and the columns is not an integral component of the open CSV file. The size information was added manually to illustrate the elements that make up the size of a data log. You can have the total size of a data log you have created as CSV file displayed in the web server on the "DataLogs" page.

SIMATIC memory card

4.1 SIMATIC memory card - Overview

Introduction

The automation system uses a SIMATIC memory card as the program memory. The SIMATIC memory card is a preformatted memory card compatible with the Windows file system. The memory card is available in different memory sizes and can be used for the following purposes:

- Transportable data carrier
- Program card
- Firmware update card
- Service data card

When you transfer the user program to the CPU via an online connection, it is written to the SIMATIC memory card. The SIMATIC memory card must be in the card slot of the CPU for this.

You can also write the SIMATIC memory card in the programming device or PC. A commercially available SD card reader is needed to read or write the SIMATIC memory card with the programming device or PC. You use this, for example, to copy files directly to the SIMATIC memory card with Windows Explorer.

Note

The SIMATIC memory card is mandatory for operation of the CPU.

Note

SIMATIC memory cards of the redundant system S7-1500R/H

For the redundant system S7-1500R/H you need a SIMATIC memory card for each of the two CPUs. During redundant operation, both CPUs access the memory cards.

There must be a SIMATIC memory card in each of the CPUs. We recommend the same memory size for each SIMATIC memory card. Both memory cards must have enough memory reserve.

Note

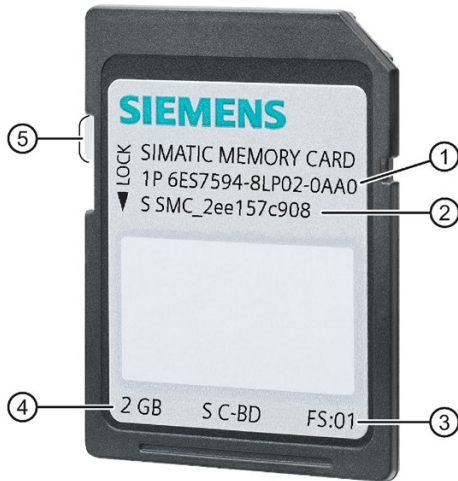
SIMATIC memory card of the SIMATIC Drive Controller

In addition to the CPU data of the SIMATIC Drive Controller, you also store the configuration of the SINAMICS Integrated on the SIMATIC memory card.

A description of how to store the online and offline data can be found in the SIMATIC memory card section of the SIMATIC Drive Controller

(<https://support.industry.siemens.com/cs/ww/en/view/109766665>) system manual.

Labeling of the SIMATIC memory card



- ① Article number
- ② Serial number
- ③ Production version
- ④ Memory size
- ⑤ Slider for setting the write protection:
 - Slider up: not write-protected
 - Slider down: write-protected

Figure 4-1 Labeling of the SIMATIC memory card

Folders and files on the SIMATIC memory card

The following folders and files can be found on the SIMATIC memory card:

Table 4- 1 Folder structure

Folder	Description
FWUPDATE.S7S	Firmware update files for CPU, I/O modules or SINAMICS Integrated**
SIMATIC.S7S	User program, i.e. all blocks (OBs, FCs, FBs, DBs) and system blocks, project data of the CPU
SINAMICS.S7S**	SINAMICS Integrated project data The folder structure below the SINAMICS.S7S folder is based on the structure of a SINAMICS S120 memory card.
SIMATIC.HMI	HMI-relevant data
DataLogs*	DataLog files
Recipes*	Recipe files
UserFiles*	You have to manually create the folder named "UserFiles" on the memory card for your user data. Only files in this folder (*.pdf, *.txt, *.csv, etc) are also loaded into the STEP 7 project with the "Upload device as new station" function.
Backups	Files for backing up and restoring via the display
DUMP.S7S	Service data files

* The content of these folders is also loaded into the STEP 7 project with the "Upload device as new station" function.

** Only with SIMATIC Drive Controllers

Table 4- 2 File structure

File type	Description
S7_JOB.S7S	Job file
SIMATIC.HMI\Backup*.psb	Panel backup files
SIMATICHMI_Backups_DMS.bin	Protected file (necessary for use of panel backup files in STEP 7)
__LOG__	Protected system file (necessary for use of card)
crdinfo.bin	Protected system file (necessary for use of card)
*.pdf, *.txt, *.csv, etc.	Additional file with different formats that you can also store in folders of the SIMATIC memory card If you store the files in the "UserFiles" folder, the files are stored in the STEP 7 project with "Upload device as new station" and can be used, for example, for restoring files in the event of a defective SIMATIC memory card.

Note

Supported file/directory names and hierarchy levels on the SIMATIC memory card

The following rules apply to file names, directory names and hierarchy levels on the SIMATIC memory card:

- Do not use any umlauts (ö, ä, ü, Ö, Ä, Ü) in file and/or directory names.
- Use a maximum of 60 characters for file and/or directory names.
- Use a maximum of 6 hierarchy levels for the directory structure of the SIMATIC memory card (for example: /mountpoint/1/2/3/4/5/6/file.txt).

The operating system of the CPU supports directories of no more than 8 levels, where one level is reserved for the mount point and one for the actual file.

Additional folders and files: CPU 1518-4 PN/DP MFP and CPU 1518F-4 PN/DP MFP

The SIMATIC memory card of the CPU 1518-4 PN/DP MFP and CPU 1518F-4 PN/DP MFP also contain the following folders and files:

- The following C/C++ runtime containers are located in the "/CppEnv1.MFP" directory on the SIMATIC memory card and are mounted in the file system in Linux:
 - System.img → Mount point: "/etc/mfp" (system files)
 - User.img → Mount point: "/home" (home directories of users, for C/C++ runtime application, for example)
 - Data.img → Mount point: "/var/userdata" (e.g. log data)
- RAM-Disk → Mount point: "/var/volatile"

Note

First-time startup of CPU 1518-4 PN/DP MFP and CPU 1518F-4 PN/DP MFP with empty SIMATIC memory card

When the CPU starts up for the first time with an empty SIMATIC memory card, the card is prepared for use with C/C++ runtime. This process takes up to three minutes. Do not turn off the CPU during this phase; the STOP LED flashes.

Note

Impact on performance of the CPU 1518-4 PN/DP MFP and CPU 1518F-4 PN/DP MFP

Depending on the type of programming, C/C++ runtime applications, such as mass storage accesses to the SIMATIC memory card, can affect the performance of the CPU.

You can find additional information on these CPUs in the CPU 1518-4 PN/DP MFP (<https://support.industry.siemens.com/cs/ww/en/view/109749061>) manual and the CPU 1518(F)-4 PN/DP MFP (<https://support.industry.siemens.com/cs/ww/en/view/109756478>) product information.

You can find information on creating C/C++ runtime applications in the SIMATIC S7-1500 ODK 1500S (<https://support.industry.siemens.com/cs/ww/en/view/109752683>) manual.

Using the serial number for copy protection

You can set up copy protection for CPUs that binds execution of the block to a specific SIMATIC memory card. You can then only execute the block if it is on the SIMATIC memory card with the specified serial number.

You can find additional information about copy protection in the following manuals:

- System manual S7-1500, ET 200MP Automation System
(<http://support.automation.siemens.com/WW/view/en/59191792>)
- SIMATIC Drive Controller
(<https://support.industry.siemens.com/cs/ww/en/view/109766665>) system manual and SINAMICS S120 Drive Functions
(<https://support.industry.siemens.com/cs/ww/en/view/109763287>) function manual
- System manual ET 200SP Distributed I/O System
(<http://support.automation.siemens.com/WW/view/en/58649293>)
- Operating instructions ET 200pro CPU 1513pro-2 PN
(<https://support.industry.siemens.com/cs/ww/en/view/109769507>) and ET 200pro CPU 1516pro-2 PN
(<https://support.industry.siemens.com/cs/ww/en/view/109482416>)

Note

Redundant system S7-1500R/H

The CPUs of the S7-1500R/H redundant system do not support the copy protection function.

Removing a SIMATIC memory card from the CPU

Note

Do not remove the SIMATIC memory card while a write operation is in progress.

If you remove the memory card from the CPU during a write operation, the contents of the memory card may become invalid. The retentivity of the memory areas can then be lost. It may be necessary to delete the memory card from the PG and reload the program.

To remove the memory card, follow these steps:

- Turn off the line voltage.

Note

If there are no write accesses via the PG, you can even remove the memory card when the power is on and the operating state is STOP. In this case, first disconnect all communication connections as a precaution.

Inserting the SIMATIC memory card in the CPU in STOP state triggers a re-evaluation of the SIMATIC memory card. The CPU compares the content of the configuration on the SIMATIC memory card with the backed-up retentive data. If the backed-up retentive data matches the data of the configuration on the SIMATIC memory card, the retentive data is retained. If the data differs, the CPU automatically performs a memory reset. A memory reset deletes the retentive data on the CPU. After the memory reset, the CPU goes to STOP.

With regard to removal of the SIMATIC memory card, note also the following FAQ on the Internet (<https://support.industry.siemens.com/cs/ww/en/view/59457183>).

Note

Using the SIMATIC memory card as a firmware update card

If you use the SIMATIC memory card as a firmware update card, removing and inserting the card will not result in the loss of retentive data.

Removing a SIMATIC memory card from Windows computers

If you are using the card in a commercially available card reader under Windows, use the "Eject" function before you remove the card from the card reader. If you remove the card without using the "Eject" function, you may lose data.

Deleting contents of the SIMATIC memory card

You have the following options for deleting the content of the SIMATIC memory card:

- Delete files with Windows Explorer
- Format with STEP 7

Note

Deletion of files and folders is permitted, with the exception of the "__LOG__" and "crdinfo.bin" system files. The CPU needs these system files. When you delete the files, you will no longer be able to use the SIMATIC memory card with the CPU.

If you have deleted the "__LOG__" and "crdinfo.bin" system files, format the SIMATIC memory card as described in the following section.

Formatting a SIMATIC memory card

NOTICE
Formatting the SIMATIC memory card
Do not format the memory card with Windows tools. Formatting with Windows renders the memory card unusable by a CPU initially.
You can find information on how to repair an inconsistent or incorrectly formatted card in the following FAQ on the Internet (https://support.industry.siemens.com/cs/ww/en/view/69063974).

To free up memory space on your SIMATIC memory card, you have the option of formatting the SIMATIC memory card. During formatting, the entire content of the memory card is deleted.

The SIMATIC memory card may only be formatted in the CPU. With a SIMATIC memory card inserted, follow these steps:

Formatting with STEP 7:

- Establish an online connection.
- Open the online and diagnostics view of the CPU (either from the project context or via "Accessible devices").
- In the dialog window select "Functions > Format memory card" and then select the "Format" button.

Formatting via the display of the CPU

- In the display of the CPU, select the menu "Settings" > "Card functions" > "Format card" and confirm with OK.

Changing the memory card without loss of retentive data

You can change the memory card or use a larger SIMATIC memory card without losing the retentive data. When you switch off the CPU, the retentive data is backed up in the retentive memory of the CPU. While the CPU is switched off, you can remove the memory card and copy its content to a larger memory card. After switching on the CPU, the data backed up in the CPU when the CPU was switched off is restored.

4.2 Setting the card type

Introduction

You can use the SIMATIC memory card as a program card or as a firmware update card.

Procedure using STEP 7

1. To set the card type, insert the SIMATIC memory card into the card reader of the programming device.
2. Select the "SIMATIC Card Reader" folder in the project tree.
3. In the properties of the selected SIMATIC memory card, specify the card type:

- **Program card**

You use a program card as an external load memory for the CPU. It contains the complete user program for the CPU. The CPU transfers the user program from the load memory to the work memory. The user program runs in the work memory.

The following folders are created on the SIMATIC memory card:

- SIMATIC.S7
- SINAMICS.S7S (only with SIMATIC Drive Controller)

- **Firmware update card**

You can save firmware for a CPU and for I/O modules on a SIMATIC memory card. This enables you to perform a firmware update with the help of a specially prepared SIMATIC memory card.

The following folder is created on the SIMATIC memory card: FWUPDATE.S7S

Reference

You can find additional information in the STEP 7 online help.

Procedure using the job file

You can also set whether the SIMATIC memory card works as a program card or a firmware update card via the job file S7_JOB.S7S on the SIMATIC memory card.

Entry in the job file:

- PROGRAM: The SIMATIC memory card is used as a program card
- FWUPDATE: The SIMATIC memory card is used as a firmware update card

Proceed as follows:

1. Open the job file S7_JOB.S7S with an editor.
2. Overwrite the PROGRAM entry with FWUPDATE using the editor (or vice versa).
Do not use spaces, line breaks, or quotation marks.
3. Save the file under the existing file name.

Program card incl. firmware update files

If, for example, you want to deliver your project with a firmware that you have validated, you can store program data and firmware update files together on the memory card. Your customers then have the opportunity to upgrade the CPU to exactly this firmware version.

Note

Note that depending on the tool used (e.g. STEP 7, Display, Web server), the firmware update files are also deleted when the "Program card" is reset.

4.3 Data transfer with SIMATIC memory cards

Saving objects from the project to the SIMATIC memory card

When the SIMATIC memory card is inserted in the programming device or in the external card reader, you can save the following objects from the project tree (STEP 7) to the SIMATIC memory card:

- Individual blocks (multiple selection possible)
In this case, the transfer is consistent, i.e. the function takes dependencies between blocks due to block calls into account.
- CPU folder
In this case, all the runtime-relevant objects, such as blocks and the hardware configuration, are saved to the SIMATIC memory card, just as with downloading.
- Service data
In this case, the service data saved beforehand is saved to the SIMATIC memory card. You will find additional information on the service data in the following manuals:
 - System manual S7-1500, ET 200MP Automation System
(<http://support.automation.siemens.com/WW/view/en/59191792>)
 - System manual Redundant System S7-1500R/H
(<https://support.industry.siemens.com/cs/ww/en/view/109754833>)
 - SIMATIC Drive Controller
(<https://support.industry.siemens.com/cs/ww/en/view/109766665>) system manual
 - System manual ET 200SP Distributed I/O System
(<http://support.automation.siemens.com/WW/view/en/58649293>)
 - Operating instructions ET 200pro CPU 1513pro-2 PN
(<https://support.industry.siemens.com/cs/ww/en/view/109769507>) and
ET 200pro CPU 1516pro-2 PN
(<https://support.industry.siemens.com/cs/ww/en/view/109482416>).

You have the following options for performing the save operation:

- Save the objects using drag-and-drop.
- Use the "Card Reader/USB memory > Write to memory card" command in the "Project" menu.

Saving trace recordings to the SIMATIC memory card

The "Save measurements on device (memory card)" function allows you to save trace recordings on your SIMATIC memory card.

Note

Redundant system S7-1500R/H

The CPUs of the redundant system S7-1500R/H do not support the saving of measurements on the SIMATIC memory card.

To access the corresponding dialog, proceed as follows:

1. In the project tree, select the "Traces" > "Trace".
2. In the work area, select "Configuration" > "Recording conditions" > "Measurements in the device (memory card)"

Response when number reached

"Deactivate recording" repeats the measurements until the configured "Number of measurements" is reached.

"Overwrite oldest recording" replaces the oldest measurement with the latest measurement when the configured "Number of measurements" is reached. Note, however, that continuously writing data to the SIMATIC memory card shortens its service life.

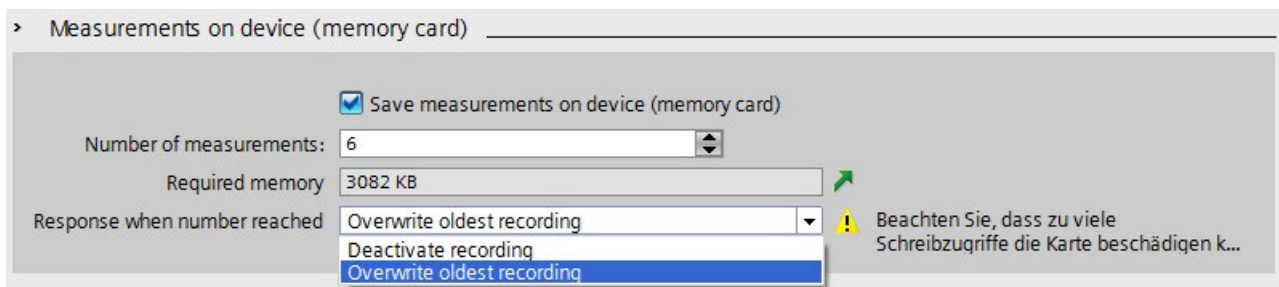


Figure 4-2 Dialog of settings for saving measurements on the memory card in STEP 7

Number of measurements

The CPU supports a maximum of 999 measurements. The CPU writes the trace recordings to the load memory of the memory card. Meanwhile, the CPU pauses the checking of trigger conditions for the trace job. Once the CPU finishes saving the trace recordings, the CPU continues checking the trigger conditions.

NOTICE

Memory required on the SIMATIC memory card

When the trace function "Measurements on device (memory card)" requires more memory than is available on the SIMATIC memory card, undesired effects may result. Ensure there is always sufficient free memory space to use the "Measurements on device (memory card)" function.

In addition to the "Measurements on device (memory card)" trace function, other functions such as storing data logs use memory space on the SIMATIC memory card. Ensure there is always sufficient memory space available for all functions that use memory.

You can find additional information about trace measurements and trace recordings in the following:

- Function manual Using the Trace and Logic Analyzer Function (<http://support.automation.siemens.com/WW/view/en/64897128>)
- Function manual Web Server (<http://support.automation.siemens.com/WW/view/en/59193560>)
- STEP 7 online help

Firmware update using SIMATIC memory card

You can find information on how to perform a firmware update in the following:

- System manual S7-1500, ET 200MP Automation System (<http://support.automation.siemens.com/WW/view/en/59191792>)
- System manual Redundant System S7-1500R/H (<https://support.industry.siemens.com/cs/ww/en/view/109754833>)
- In the SIMATIC Drive Controller (<https://support.industry.siemens.com/cs/ww/en/view/109766665>) system manual
- System manual ET 200SP Distributed I/O System (<http://support.automation.siemens.com/WW/view/en/58649293>)
- Operating instructions ET 200pro CPU 1513pro-2 PN (<https://support.industry.siemens.com/cs/ww/en/view/109769507>) and ET 200pro CPU 1516pro-2 PN (<https://support.industry.siemens.com/cs/ww/en/view/109482416>)

Reference

You can find additional information about the SIMATIC memory card in the STEP 7 online help.

4.4 Service life of the SIMATIC memory card

Calculation of the theoretical service life of a SIMATIC memory card serves as a decision-making aid for selecting which card you need for your automation task. The following examples only return a guide value, however. A precise calculation of the service life is not possible due to the fact that the description cannot cover all the theoretically possible scenarios.

Influences on the service life

You can influence the service life of SIMATIC memory cards by the following factors:

- Card size and number of guaranteed write operations
- Number of actual write operations

The number of physical write operations to the memory blocks of the card results from the number of write operations from the application.

Structure of a SIMATIC memory card

The internal flash memory of the SIMATIC memory card is organized in memory blocks. A memory block is a memory area of a fixed size. A write operation always addresses entire memory blocks on the SIMATIC memory card. When a memory block has been written once it must be deleted before it can be written again. The number of delete/write operations per memory block is limited. The service life of the SIMATIC memory card is measured by the maximum number of supported delete or write operations per memory block.

In contrast to delete or write operations, read operations have a negligible impact on the service life. Therefore, the impact of the read operations on the service life is not included in this calculation. A very high number of read operations can, however, influence the service life to a small extent.

Maximum number of write/delete operations

The internal controller of the memory card ensures that the available memory blocks are evenly used. In this way, a maximum number of write operations is possible on the SIMATIC memory card. Internal algorithms distribute the write accesses to the same logical memory area over changing physical memory areas in order to use the memory blocks evenly.

The following table shows the maximum possible number of write/delete operations as a function of the SIMATIC memory card used. The number of maximum write/delete operations of the respective SIMATIC memory card is also available online in the technical specifications of the respective SIMATIC memory card.

Memory size of the SIMATIC memory card *	Article number	Max. number of write/delete operations per memory block
4 MB	6ES7954-8LCxx-0AA0	500 000
12 MB	6ES7954-8LExx-0AA0	500 000
24 MB	6ES7954-8LFxx-0AA0	500 000
256 MB	6ES7954-8LL03-0AA0	200 000
2 GB	6ES7954-8LP01-0AA0	100 000
2 GB	6ES7954-8LP02-0AA0	60 000
32 GB	6ES7954-8LT03-0AA0	100 000

* The memory size figures named in the table are theoretical values. The actual existing memory size in practice is below the theoretical value. The reason for this is that the internal controller of the card and the file system reserve part of the existing memory for internal memory management.

Note

Write or delete operations

Write or delete operations, particularly repeated (cyclic) write/delete operations by the user program on the SIMATIC memory card reduces its service life.

Cyclic execution of the following instructions reduces the service life of the memory card depending on the number of write operations and data:

- "CREATE_DB" (with ATTRIB "Create DB in load memory")
- "DataLogWrite"
- "RecipeExport"
- "RecipeImport" (if target DB in load memory)
- "WRIT_DBL"
- "SET_TIMEZONE"

Note also that, in addition to the cyclic write/delete operations, the writing or deleting of very large amounts of data also adversely affects the service life of the SIMATIC memory card.

Note

Redundant system S7-1500R/H

The CPUs of the S7-1500R/H redundant system do not support the instructions specified in the "Write or delete operations" note.

Guaranteed data retention time

If you do not use your SIMATIC memory card for an extended period of time, there is the risk that data contained on the memory card may no longer be readable after a certain amount of time.

The guaranteed data retention time of a SIMATIC memory card is 10 years on delivery and with proper storage. With a number of $\leq 10\%$ of the maximum write/delete operations, the data stored on the card has a retention time of 10 years.

Please note that increasing numbers of write/delete operations to the card reduces its data retention time. If 90% of the maximum write/delete operations is reached, the guaranteed data retention time is reduced to 1 year. If 100% of the maximum write/delete operations is reached, the retention time of the saved data can no longer be guaranteed.

Determining the current usage level of a SIMATIC memory card in STEP 7

When you activate the "Aging of the SIMATIC memory card" option, you enter a threshold value as a percentage in the text box below. As soon as the service life of the SIMATIC memory card has reached the specified threshold value (e.g.: 80 %), the CPU outputs a diagnostics alarm and the MAINT LED lights up in yellow.

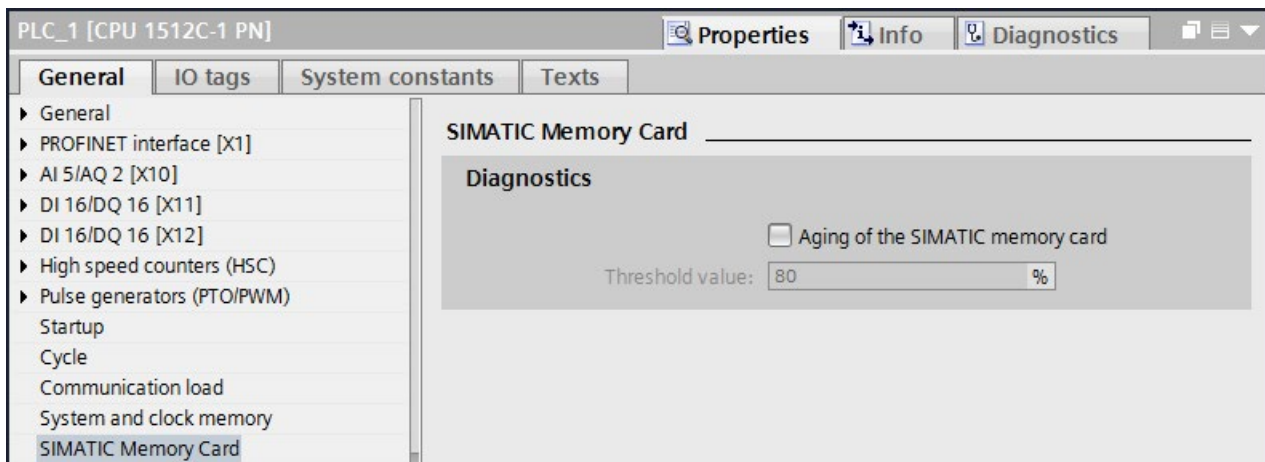


Figure 4-3 Enabled option "Aging of the SIMATIC memory card"

Calculation of the theoretical service life of a SIMATIC memory card

We will use the following example as a basis for calculation:

The user is using a new **256 MB** memory card. In accordance with the table, this memory card type supports **200 000** write operations. After parameter changes, the user would like to write **200 DBs** of **5 KB** each to the SIMATIC memory card at a frequency of **50** times per day with the "RecipeExport" instruction.

Step 1: Calculating the write operations

First use the following formula to calculate the service life of the SIMATIC memory card:

$$\text{Write operations} = \frac{\text{Size of the memory card} * \text{Maximum number of write operations}}{\text{Number of written bytes}}$$

We first use the sizes from the example in the formula "Write operations" as a basis for calculating the service life:

- Size of the memory card: **256 MB = 268 435 456 bytes**
- Maximum number of write operations: **200 000**
- Number of written bytes: **1024 000 bytes (200 x 5 KB)**

If we use the sizes from the example in the formula, we obtain the following result:

$$\text{Write operations} = \frac{268\,435\,456 \text{ byte} * 200\,000}{1\,024\,000 \text{ byte}} = 52\,428\,800 \text{ write operations}$$

Step 2: Calculating the service life

Use the following formula to calculate the service life in years:

$$\text{Service life} = \frac{\frac{\text{Write operations}}{\text{Write operations per day}}}{365 \text{ days}} : \text{Net-gross factor} = \text{years}$$

Note

Net-gross factor

Internal data (metadata) are also written to the SIMATIC memory card with each write operation. Due to this additional data, include the net-gross factor **100** when calculating the service life.

If we use the sizes from the example in the formula, we obtain the following result:

$$\text{Service life} = \frac{\frac{52\,428\,800}{50}}{365 \text{ days}} : 100 = 28.7 \text{ years}$$

Calculation with more frequent write accesses and a higher number of bytes written

If the frequency of write accesses and the number of bytes written per day increases, the service life of the SIMATIC memory card is reduced.

The following table shows, based on empirical values, how the service life of a SIMATIC memory card with a size of 256 MB is reduced

Write accesses per day	Number of bytes written per instruction	Service life of the SIMATIC memory card in years
50	1 024 000	28.7
100	1 024 000	14.3
400	1 024 000	3.6
400	2 048 000	1.8
400	4 096 000	0.9

The following table shows how the same values impact the service life of a SIMATIC memory card with a size of 2 GB (6ES7954-8LP01-0AA0).

Write accesses per day	Number of bytes written per instruction	Service life of the SIMATIC memory card in years
50	1 024 000	114.9
100	1 024 000	57.5
400	1 024 000	14.4
400	2 048 000	7.2
400	4 096 000	3.6

The result shows that a high number of write accesses together with a high number of written bytes significantly shortens the service life of the SIMATIC memory card.

Reference

You can find an alternative method for calculating the service life of a SIMATIC memory card in an FAQ on the Internet (<https://support.industry.siemens.com/cs/ww/en/view/109482591>).

GetSMCinfo instruction

When the SIMATIC memory card inserted, you can read out the following information in STEP 7 (TIA Portal) using the GetSMCinfo instruction:

- Memory size in (1 KB = 1 024 bytes)
- Memory space occupied in (1 KB = 1 024 bytes)
- Maintenance information: Previously used up portion of the service life in %
- Configured percentage of service life after which the CPU creates a diagnostics buffer entry.

You can find additional information on the GetSMCinfo instruction in the STEP 7 online help.

4.5 Expanding the load memory of the CPUs of the redundant system S7-1500R/H

Memory requirements

If the memory space on one of the two SIMATIC memory cards is not sufficient, you can replace this card during operation of the redundant system S7-1500R/H.

Note

To avoid errors on the SIMATIC memory card due to insufficient memory, use memory cards with enough memory space.

Expanding the load memory during operation

To expand the load memory of the CPUs of the redundant system S7-1500R/H during operation, follow these steps:

Action		System response
1.	Switch the first CPU to STOP while in redundant operation.	The system changes to system state RUN-Solo.
2.	Remove the existing SIMATIC memory card from the CPU that is in STOP. Insert a larger SIMATIC memory card.	The CPU performs a memory reset.
3.	After the memory reset switch the CPU back to RUN.	The CPU performs a SYNCUP.
4.	Wait until the SYNCUP of the CPU is complete and then switch the second CPU to STOP.	The system changes to system state RUN-Solo again.
5.	Remove the existing SIMATIC memory card from the CPU that is in STOP. Insert a larger SIMATIC memory card.	The CPU performs a memory reset.
6.	After the memory reset, switch the CPU back to RUN.	The CPU performs a SYNCUP.
7.	Wait until the SyncUp of the CPU is complete.	The CPUs now have a larger load memory and are in system state RUN-Redundant again.

Glossary

Backup CPU

When the R/H system is in RUN-Redundant system state, the Primary CPU controls the process. The Backup CPU synchronously executes the user program and can take over the process control at a failure of the Primary CPU.

Bit memory

Bit memory is a component of the system memory of the CPU for saving intermediate results. You access the bit memory from the user program in bit, byte, word or double-word mode.

Code block

In SIMATIC S7, a code block contains a portion of the STEP 7 user program.

Consistent data

Consistent data is data whose content belongs together. Consistent data is read and written contiguously.

Counter

Counters are components of the system memory of the CPU. You can modify the content of the "counter cells" using STEP 7 instructions. Example: Count up or down.

Data block

Data blocks (DBs) are data areas in the user program that contain user data. The following data blocks are available:

- Global data blocks that you access from all code blocks.
- Instance data blocks that are assigned to a particular FB call.

Data log

Data logs are CSV files for the saving of tag values. The data logs are stored on the SIMATIC memory card in the "\datalogs" directory. Instructions in the user program write data records of tag values to a data log.

Diagnostics

Monitoring functions include:

- Detection, localization, classification of errors, faults and alarms.
- Display and further evaluation of errors, faults and alarms.

The monitoring functions run automatically during system operation. This increases the availability of systems because commissioning times and downtimes are reduced.

Diagnostics buffer

The diagnostics buffer is a buffered memory area in the CPU in which diagnostics events are stored in their order of occurrence.

Firmware of the CPU

In SIMATIC, a distinction is made between the firmware of the CPU and user programs.

Firmware is software that is embedded in electronic devices, i.e. functionally connected permanently to the hardware. It is usually saved in a flash memory, EPROM, EEPROM or ROM and cannot be replaced by the user or can only be replaced with special tools or functions.

User program: see glossary entry "User program"

Firmware update

You use a firmware update to update the firmware of modules. A firmware update is performed, e.g. for functional extensions of a CPU or interface module.

Function block

A function block (FB) is a code block with static data. An FB allows you to pass parameters in the user program. Function blocks are thus suited for programming frequently recurring complex functions, such as closed-loop controls or operating states selection.

Global data block (DB)

Every function block, every function, and every organization block can read the data from a global data block, or write its own data to a global data block. This data is retained in the data block, even when the data block is exited.

I/O module

Device of the distributed I/O that is used as an interface between the controller and the process.

Instance data block (DB)

Each call of a function block in the STEP 7 user program is assigned a data block that is automatically generated. Values of the input, output and in/out parameters, as well as local block data, are stored in the instance data block.

IP address

The IP address is made up of four decimal numbers, each with a value range of 0 to 255. The decimal numbers are separated by a dot (e.g. 192.162.0.0).

The IP address consists of the following:

- Address of the network
- Device address (PROFINET interface of the IO controller/IO devices)

Local data

This memory area accepts the temporary local data of a block for the duration of processing.

Memory reset

Procedure for setting the memories of the CPU to a defined initial state.

Operating states

Operating states describe the behavior of an individual CPU at any given point in time.

The CPUs of the SIMATIC standard systems feature the STOP, STARTUP and RUN operating states.

The primary CPU of the redundant system S7-1500R/H has the operating states STOP, STARTUP, RUN, RUN-Syncup and RUN-Redundant. The backup CPU has the operating states STOP, SYNCUP and RUN-Redundant.

Optimized block access

Data blocks with optimized access have no fixed structure. In the declaration, the data elements only receive a symbolic name, and no fixed address within the block. The elements are automatically arranged in the block's available memory area in such a way that its capacity is optimally exploited.

In these data blocks, you can only address tags symbolically. For example, you would access the "FillState" tag in the "Data" DB as follows:

```
"Data".FillState
```

Optimized access offers the following advantages:

- The data is structured and saved in a manner that is optimal for the CPU used. This allows you to increase CPU performance.
- Access errors, e.g. from the HMI, are not possible.
- You can selectively define individual tags as retentive.

Organization block

Organization blocks (OBs) form the interface between the operating system of the CPU and the user program. The organization blocks determine the order in which the user program is executed.

Parameter

- Variable of a STEP 7 code block
- Variable for setting the behavior of a module (one or more per module). In delivery state, each module has an appropriate basic setting that can be changed by configuring in STEP 7. There are static and dynamic parameters.

Parameters, dynamic

In contrast to static parameters, you can change dynamic parameters of modules during operation by calling an SFC in the user program, e.g. limit values of an analog input module.

Parameters, static

In contrast to dynamic parameters, you cannot change static parameters of modules with the user program but only by configuring in STEP 7, e.g. input delay of a digital input module.

Primary CPU

When the R/H system is in RUN-Redundant system state, the Primary CPU controls the process. The Backup CPU synchronously executes the user program and can take over the process control at a failure of the Primary CPU.

Process image (I/O)

The CPU transfers the values from the input and output modules to this memory area. At the start of the cyclic program, the CPU transfers the process image output as a signal state to the output modules. The CPU then reads the signal states of the input modules to the process image of the inputs. Then the CPU executes the user program.

Redundant systems

Redundant systems are characterized in that important automation components are present multiple times (redundantly). Process control is maintained if a redundant component fails.

Reset to factory settings

Resetting to factory settings restores the CPU settings to the delivery state.

Restart

A restart (warm restart) deletes all non-retentive bit memory and resets non-retentive DB contents to the start values from the load memory. Retentive bit memory and retentive DB contents are retained. Program execution begins at the call of the first startup OB.

Retentivity

A memory area whose content is retained after power failure and after a transition from STOP to RUN is retentive. The non-retentive bit memory area, timers and counters are reset after power failure and after a STOP to RUN transition.

SIMATIC memory card

Memory for the user program for programmable modules and communications processors. You can also use the SIMATIC memory card for exchange of user software and user data.

Standard access

Data blocks with standard access have a fixed structure. In the declaration, the data elements contain both a symbolic name and a fixed address within the block. The address is displayed in the "Offset" column.

In these data blocks, you can address tags both symbolically and absolutely:

"Data".FillState

DB1.DBW2

System states

The redundant S7-1500R/H system has various system states. The system states result from the operating states of the Primary and Backup CPUs. The concept of the system state is used to obtain a simplified expression that characterizes the simultaneously occurring operating states of the two CPUs. The following system states are available for the redundant system S7-1500R/H: STOP, STARTUP, RUN-Solo, SYNCUP and RUN-Redundant.

Timer

Timers are components of the system memory of the CPU. The operating system automatically updates the content of the "timer cells" asynchronously to the user program. STEP 7 instructions specify the precise function of the timer cell (e.g. on delay) and trigger its execution.

User program

In SIMATIC, a distinction is made between user programs and the firmware of the CPU.

The user program contains all instructions, declarations and data that enable a plant or process to be controlled. The user program is assigned to a programmable module (e.g. CPU, FM) and can be structured in smaller units.

Firmware: see glossary entry "Firmware of the CPU"

Index

B

Bit memory, 24

C

C/C++, 12, 54

Counter, 24

D

Data block, 22, 27

Data logging

 Data structure, 39

 DataLogCreate, 39

 Example program, 42

 Overview of data logging, 37

Data retention time, 65

Diagnostics buffer, 26

Display, 19, 53, 57

Disposal, 5

F

FAQ

 Calculate service life SMC, 67

 Configure data blocks, 34

 Estimating memory requirements, 17

 Formatting a SIMATIC memory card, 57

 Insufficient load memory, 32

 Load project data in load memory, 32

 Removing a SIMATIC memory card, 56

 Repairing the SIMATIC memory card, 57

 Using recipe phases, 37

Function block, 22

I

Instructions

 Asynchronous, 36, 41

 CREATE_DB, 23

 DataLogClear, 40

 DataLogClose, 40

 DataLogCreate, 39, 40, 48

 DataLogDelete, 40

 DataLogNewFile, 40

 DataLogOpen, 40

 DataLogWrite, 40

 GetSMCinfo, 67

 READ_DBL, 33

 RecipeExport, 35

 RecipeImport, 36

 WRIT_DBL, 33

L

Linux, 12, 54

Load memory, 12

Load memory S7-1500R/H-CPUs, 68

M

Memory reserve, 27

O

Offline project, 14, 16

Offline project data, 11

Online project data, 11

Operating hours counter, 26

P

PLC tags, 26

R

Redundancy ID, 15

Redundant S7-1500R/H system, 68

Retentive data, 15, 56, 58

Retentive memory, 11, 13, 21

 S7-1500R/H-CPUs, 15

Runtime, 12, 54

S

Scope, 4

SIMATIC memory card, 51, 58, 60

 Basics, 51

 Firmware card, 58

- Possible applications, 60
- Program card, 58
- Repairing, 57
- Updating firmware, 62
- Software change, 26

T

- Technology objects, 23
- Timer, 24
- Trace recordings, 61

W

- Web server, 12, 20, 45, 62
- Work memory, 11, 13