S7-400 Instruction List

CPU 412, 414, 416, 417

This Instruction List has the order number:

6ES7498-8AA03-8BN0

Edition 12/2002

A5E00069511-07

Copyright © Siemens AG 2002 All rights reserved

The reproduction, transmission or use of this document or its contents is not permitted without express written authority. Offenders will be liable for damages. All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

Siemens AG
Bereich Automation and Drives
Geschaeftsgebiet Industrial Automation Systems
Postfach 4848, D- 90327 Nuernberg

Disclaimer of Liability

We have checked the contents of this manual for agreement with the hardware and software described. Since deviations cannot be precluded entirely, we cannot guarantee full agreement. However, the data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcomed.

© Siemens AG 2002
Subject to change without prior notice

Siemens Aktiengesellschaft

6ES7498-8AA03-8BN0

Excellence in Automation & Drives: Siemens

Contents

Applicability

This list of instructions applies to the CPUs listed below.

Name	Order number	subsequently described as*
CPU 412-1	6ES7 412-1XF03-0AB0	CPU 412
CPU 412-2	6ES7 412-2XG00-0AB0	
CPU 412-2 PCI	6ES7 612-2QH00-0AB4	
CPU 414-2	6ES7 414-2XG03-0AB0	CPU 414
CPU 414-3	6ES7 414-3XJ00-0AB0	
CPU 414-4H	6ES7 414-4HJ00-0AB0	CPU 414
CPU 416-2	6ES7 416-2XK02-0AB0	CPU 416
CPU 416-2 PCI	6ES7 616-2QL00-0AB4	
CPU 416-3	6ES7 416-3XL00-0AB0	
CPU 417-4	6ES7 417-4XL00-0AB0	CPU 417
CPU 417-4 H	6ES7 417-4HL01-0AB0	

* except in the tables, where a detailled differentiation is necessary

Address Identifier and Parameter Ranges

Addr. ID	Parameter Range				Description
	CPU 412	CPU 414	CPU 416	CPU 417	
Q[2)]	0.0 to 127.7	0.0 to 255.7	0.0 to 511.7	0.0 to 1023.7	Output (in PIQ)
QB[2)]	0 to 127	0 to 255	0 to 511	0 to 1023	Output byte (in PIQ)
QW[2)]	0 to 126	0 to 254	0 to 510	0 to 1022	Output word (in PIQ)
QD[2)]	0 to 124	0 to 252	0 to 508	0 to 1020	Output double word (in PIQ)
DBX	0.0 to 65533.7[1)]	0.0 to 65533.7	0.0 to 65533.7	0.0 to 65533.7	Data bit in data block
DB	1 to 511	1 to 4095	1 to 4095	1 to 8191	Data block
DBB	0 to 65533[1)]	0 to 65533	0 to 65533	0 to 65533	Data byte in DB
DBW	0 to 65532[1)]	0 to 65532	0 to 65532	0 to 65532	Data word in DB
DBD	0 to 65530[1)]	0 to 65530	0 to 65530	0 to 65530	Data double word in DB
DIX	0.0 to 65533.7[1)]	0.0 to 65533.7	0.0 to 65533.7	0.0 to 65533.7	Data bit in instance DB
DI	1 to 511	1 to 4095	1 to 4095	1 to 8191	Instance data block
DIB	0 to 65533[1)]	0 to 65533	0 to 65533	0 to 65533	Data byte in instance DB
DIW	0 to 65532[1)]	0 to 65532	0 to 65532	0 to 65532	Data word in instance DB
DID	0 to 65530[1)]	0 to 65530	0 to 65530	0 to 65530	Data double word instance DB

[1)] Also restricted by the size of the working memory.

[2)] Default setting can be changed, see Technical Specifications

Address Identifier and Parameter Ranges, continued

Addr. ID	Parameter Range				Description
	CPU 412	CPU 414	CPU 416	CPU 417	
I[2)]	0.0 to 127.7	0.0 to 255.7	0.0 to 511.7	0.0 to 1023.7	Input bit (in PII)
IB[2)]	0 to 127	0 to 255	0 to 511	0 to 1023	Input byte (in PII)
IW[2)]	0 to 126	0 to 254	0 to 510	0 to 1022	Input word (in PII)
ID[2)]	0 to 124	0 to 252	0 to 508	0 to 1020	Input double word (in PII)
L[2)]	0.0 to 4095.7	0.0 to 8191.7	0.0 to 16383.7	0.0 to 32767.0	Local data
LB[2)]	0 to 4095	0 to 8191	0 to 16383	0 to 32767	Local data byte
LW[2)]	0 to 4094	0 to 8190	0 to 16382	0 to 32766	Local data word
LD[2)]	0 to 4092	0 to 8188	0 to 16380	0 to 32764	Local data double word
M	0.0 to 4095.7	0.0 to 8191.7	0.0 to 16383.7	0.0 to 16383.7	Bit memory
MB	0 to 4095	0 to 8191	0 to 16383	0 to 16383	memory byte
MW	0 to 4094	0 to 8190	0 to 16382	0 to 16382	memory word
MD	0 to 4092	0 to 8188	0 to 16380	0 to 16380	memory double word

[2)] Default setting can be changed, see Technical Specifications

Address Identifier and Parameter Ranges, continued

Addr. ID	Parameter Range				Description
	CPU 412	CPU 414	CPU 416	CPU 417	
PQB	0 to 4095	0 to 8191	0 to 16383	0 to 16383	Peripheral output byte (direct I/O access)
PQW	0 to 4094	0 to 8190	0 to 16382	0 to 16382	Peripheral output word (direct I/O access)
PQD	0 to 4092	0 to 8188	0 to 16380	0 to 16380	Peripheral output double word (direct I/O access)
PIB	0 to 4095	0 to 8191	0 to 16383	0 to 16383	Peripheral input byte (direct I/O access)
PIW	0 to 4094	0 to 8190	0 to 16382	0 to 16382	Peripheral input word (direct I/O access)
PID	0 to 4092	0 to 8188	0 to 16380	0 to 16380	Peripheral output double word (direct I/O access)
T	0 to 255	0 to 255	0 to 511	0 to 511	Timer
C	0 to 255	0 to 255	0 to 511	0 to 511	Counter

Constants and Ranges

Constant	Range	Description
B(b1,b2) B(b1,b2,b3,b4)	–	Constant, 2 or 4 bytes
D# Date	–	IEC date constant
L# Integer	–	32-bit integer constant
P# Bit pointer	–	Pointer constant
S5T# Time	–	S7 time constant [1]
T# Time	–	Time constant
TOD# Time	–	IEC time constant
C# Count	–	Counter constant (BCD code)
2#n	–	Binary constant
W#16# DW#16#	–	Hexadecimal constant

[1] For loading of S7 timers.

Abbreviations and Mnemonics

The following abbreviations and mnemonics are used in the Instruction List:

Abbrev.	Description	Example
k8	8-bit constant 0 to 255	32
k16	16-bit constant 256 to 32 767	28 131
k32	32-bit constant 32 768 to 999 999 999	127 624
i8	8-bit integer -128 to +127	-113
i16	16-bit integer -32768 to +32767	+6523
i32	32-bit integer -2 147 483 648 to +2 147 483 647	-2 222 222
m	Pointer constant	P#240.3
n	Binary constant	1001 1100
p	Hexadecimal constant	EA12
LABEL	Symbolic jump address (max. 4 characters)	DESTINATION
a	Byte address	

Abbrev.	Description	Example
b	Bit address	
c	Address area	I, Q, M, L, DBX, DIX
d	Address in: MD, DBD, DID or LD	
e	Number in: MW, DBW, DIW or LW	
f	Timer/counter No.	
g	Address area	IB, QB, PIB, PQB, MB, LB, DBB, DIB
h	Address area	IW, QW, PIW, PQW, MW, LW, DBW, DIW
i	Address area	ID, QD, PID, PQD, MD, LD, DBD, DID
q	Block No.	

Registers

ACCU1 to ACCU4 (32 Bits)

The accumulators are registers for processing bytes, words or double words. The address identifiers are loaded into the accumulators, where they are logically gated. The result of the logic operation (RLO) is in ACCU1 and can be transferred from there to a memory cell.

The accumulators are 32 bits long.

Accumulator designations :

ACCU	Bits
ACCUx (x = 1 to 4)	Bits 0 to 31
ACCUx-L	Bits 0 to 15
ACCUx-H	Bits 16 to 31
ACCUx-LL	Bits 0 to 7
ACCUx-LH	Bits 8 to 15
ACCUx-HL	Bits 16 to 23
ACCUx-HH	Bits 24 to 31

Address Registers AR1 and AR2 (32 Bits)

The address registers contain the area-internal or area-crossing pointers for instructions using indirect addressing. The address registers are 32 bits long.

The area-internal and/or area-crossing pointers have the following syntax:

- Area-internal pointer `00000000 00000bbb bbbbbbbb bbbbbxxx`
- Area-crossing pointer **`yyyyyyyy`** `00000bbb bbbbbbbb bbbbbxxx`

Legend:

`b`	Byte address
`x`	Bit number
`y`	Area identifier (see “Examples ofAddressing”)

Status Word (16 Bits)

The status word bits are evaluated or set by the instructions.

The status word is 16 bits long.

Bit	Assignment	Description
0	$\overline{FC}$	First check bit
1	RLO	Result of logic operation
2	STA	Status
3	OR	Or (AND before OR)
4	OS	Stored overflow
5	OV	Overflow
6	CC 0	Condition code 0
7	CC 1	Condition code 1
8	BR	Binary result
9 to 15	Unassigned	–

Examples of Addressing

Addressing Examples	Description
Immediate Addressing	
L +27	Load 16-bit integer constant “27” into ACCU1
L L#-1	Load 32-bit integer constant “-1” into ACCU1
L 2#1010101010101010	Load binary constant into ACCU1
L DW#6#A0F0BCFD	Load hexadecimal constant into ACCU1
L ’ENDE’	Load ASCII character into ACCU1
L T#500 ms	Load time value into ACCU1
L C#100	Load count value into ACCU1
L B#(100,12)	Load 2-byte constant
L B#(100,12,50,8)	Load 4-byte constant
L P#10.0	Load area-internal pointer into ACCU1
L P#E20.6	Load area-crossing pointer into ACCU1
L -2.5	Load real number into ACCU1
L D#1995-01-20	Load date
L TOD 13:20:33.125	Load time of day

Examples of Addressing, continued

Addressing Examples	Description
Direct Addressing	
A I 0.0	ANDing of input bit 0.0
L IB 1	Load input byte 1 into ACCU1
L IW 0	Load input word 0 into ACCU1
L ID 0	Load input double word 0 into ACCU1
Indirect Addressing of Timers/Counters	
SP T [LW 8]	Start timer; the timer number is in local data word 8
CU C [LW 10]	Count upwards; the counter number is in local data word 10
Area-Internal Memory-Indirect Addressing	
A I [LD 12] Example: L P#22.2 T LD 12 A I [LD 12]	AND operation: The address of the input is in local data double word 12 as pointer
A I [DBD 1]	AND operation: The address of the input is in data double word 1 of the open DB as pointer
A I [DID 12]	AND operation: The address of the output is in data double word 12 of the open instance DB as pointer
A I [MD 12]	AND operation: The address of the output is in memory double word 12 as pointer

Examples of Addressing, continued

Addressing Examples	Description
Area-Internal Register-Indirect Addressing	
A I [AR1,P#12.2]	AND operation: The address of the input is calculated as follows: "pointer value contained in address register 1 + pointer P#12.2"
Area-Crossing Register-Indirect Addressing	

For area-crossing register-indirect addressing, the address must also contain an area identifier. The address is in the address register. The area identifiers are as follows:

Area identifier	Coding (binary)	hex.	Area
P	1000 0**000**	80	I/O area
I	1000 0**001**	81	Input area
Q	1000 0**010**	82	Output area
M	1000 0**011**	83	Bit memory area
DB	1000 0**100**	84	Data area
DI	1000 0**101**	85	Instance data area
L	1000 0**110**	86	Local data area
VL	1000 0**111**	87	Predecessor local data area (access to local data of invoking block)

L B [AR1,P#8.0]	Load byte into ACCU1: The address is calculated from the "pointer value in AR 1 + P#8.0"
A [AR1,P#32.3]	AND operation: The address of the operand is calculated from the "pointer value in AR 1 + P#32.3"
Addressing Via Parameters	
A Parameter	Addressing via parameters

Examples of how to calculate the pointer

- **Example for sum of bit addresses ≦7:**

  LAR1 P#8.2
  A I [AR1,P#10.2]

Result: Input 18.4 is addressed (by adding the byte and bit addresses)

- **Example for sum of bit addresses >7:**

  L P#10.5
  LAR1
  A I [AR1,P#10.7]

Result: Input 21.4 is addressed (by adding the byte and bit addresses with carry over)

Execution Times with Indirect Addressing

When using indirect addresses statement consists of two parts:

Part 1: Load the address of the instruction

Part 2: Execute the instruction

In other words, when working with indirect addresses, you must calculate the execution time of an instruction from these two parts.

Calculating the Execution Time

The total execution time is calculated as follows:

	Time required for loading the address
+	execution time of the instruction
=	Total execution time of the instruction

The execution times listed in the chapter entitled “List of Instructions” apply to the execution times of the second part of an instruction, i.e. for the actual execution of an instruction.

You must then add the time required for loading the address of the instruction to this execution time (see following Table).

The execution time for loading the address of the instruction from the various areas is shown in the following table.

Address is in ...	Execution Time in µs			
	CPU 412	**CPU 414**	**CPU 416**	**CPU 417**
Bit memory area M Word Double word	 0.2 0.3	 0.2 0.3	 0.16 0.24	 0.2 0.3
Data block DB/DX Word Double word	 0.2 0.3	 0.2 0.3	 0.16 0.24	 0.2 0.3
Local data area L Word Double word	 0.2 0.3	 0.2 0.3	 0.16 0.24	 0.2 0.3
AR1/AR2 (area-internal)	0.0 [1)]	0.0 [1)]	0.0 [1)]	0.0 [1)]
AR1/AR2 (area-crossing)	0.0 [1)]	0.0 [1)]	0.0 [1)]	0.0 [1)]
Parameter (word) ... for: • Timers • Counters • Block calls	 0.2 0.2 0.2	 0.2 0.2 0.2	 0.16 0.16 0.16	 0.2 0.2 0.2
Parameter (double word) ... for Bits, bytes, words and double words	0.3	0.3	0.24	0.3

[1)] Address registers AR1/AR2 do not need to be loaded in separate cycles for addressing.

The pages that follow contain examples for calculating the instruction run time for the various indirectly addressed instructions.

Examples of Calculations

You will find a few examples here for calculating the execution times for the various methods of indirect addressing.

Calculating the Execution Times for Area-Internal Memory-Indirect Addressing

Example: A I [DBD 12] with CPU 414

Step 1: Load the contents of DBD 12 (time required is listed in the table on page 20)

Address is in ...	Execution Time in µs
Bit memory area M	
Word	0.2
Double word	0.3
Data block DB/DX	
Word	0.2
Double word	0.3

Step 2: AND the input addressed in this way (you will find the execution time in the tables in the chapter entitled “List of Instructions” on page 25)

Typical Execution Time in µs	
Direct Addressing	**Indirect Addressing**
0.2	Time for A I **0.1+**
:	**:**

Total execution time:

```
      0.3 ms
+     0.1 ms
=     0.4 ms
```

Execution Time for Area-Crossing Register-Indirect Addressing

Example: A [AR1, P#23.1] ... with I 1.0 in AR1 with CPU 416

Step 1: Load the contents of AR1, and increment them by the offset 23.1 (the time required is in the table on page 20)

Address is in ...	Execution Time in µs
:	:
AR1/AR2 (area-crossing)	0.00
:	:

Step 2: AND the input addressed in this way (you will find the execution time in the tables in the chapter entitled “List of Instructions”)

Typical Execution Time in µs	
Direct Addressing	**Indirect Addressing**
0.08 :	Time for A I **0.08+** **:**

Total execution time:

0.00 ms
+ 0.08 ms
= 0.08 ms

Execution Time for Addressing Via Parameters

Example: A Parameter ... with I 0.5 in the block parameter list with CPU 414

Step 1: Load input I 0.5 addressed via the parameter (the time required is in the table on page 20)

Address is in ...	Execution Time in µs
:	:
:	:
Parameter (double word)	0.3

Step 2: AND the input addressed in this way (you will find the execution time in the table in the chapter entitled “List of Instructions” on page 25)

Typical Execution Time in µs	
Direct Addressing	**Indirect Addressing**
0.2 :	Time for A I **0.1+** **:**

Total execution time:

```
      0.3 ms
+     0.1 ms
=     0.4 ms
```

List of Instructions

This chapter contains the complete list of instructions for the S7-400 CPUs. The descriptions have been kept as concise as possible. You will find a detailed functional description in the various STEP 7 reference manuals.
Please note that, in the case of indirect addressing (examples see page 15), you must add the time required for loading the address of the particular instruction to the execution times listed (see page 20).

Bit Logic Instructions

All logic instructions generate a result (new RLO). The first instruction in a logic string generates the new RLO from the signal state scanned. The subsequent logic instructions generate the new RLO from the signal state scanned and the old RLO. The logic string ends with an instruction which limits the RLO (e.g. a memory instruction); that is, the /FC bit is set to zero.

Instr.	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
A/AN		AND/AND NOT					
	I/Q a.b	Input/output	1*/2	0.2/0.3	0.1	0.08	0.1
	M a.b	Bit memory	1**/2	0.2/0.3	0.1	0.08	0.1
	L a.b	Local data bit	2	0.3	0.1	0.08	0.1
	DBX a.b	Data bit	2	0.3	0.1	0.08	0.1
	DIX a.b	Instance data bit	2	0.3	0.1	0.08	0.1
	c [d]	Memory-indirect, area-internal	2	0.3+	0.1+	0.08+	0.1+
	c [AR1,m]	Register-ind., area-internal (AR1)	2	0.3+	0.1+	0.08+	0.1+
	c [AR2,m]	Register-ind., area-internal (AR2)	2	0.3+	0.1+	0.08+	0.1+
	[AR1,m]	Area-crossing (AR1)	2	0.3+	0.1+	0.08+	0.1+
	[AR2,m]	Area-crossing (AR2)	2	0.3+	0.1+	0.08+	0.1+
	Parameter	Via parameter	2	0.3+	0.1+	0.08+	0.1+

Status word for: **A/AN**	BR	CC 1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction depends on:	–	–	–	–	–	Yes	–	Yes	Yes
Instruction affects:	–	–	–	–	–	Yes	Yes	Yes	1

\+ Plus time required for loading the address of the instruction (see page 20)
* With direct instruction addressing;Address area 0 to 127
** With direct instruction addressing;Address area 0 to 255

Bit Logic Instructions, continued

Instr.	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
O/ON		OR/OR NOT					
	I/Q a.b	Input/output	1*/2	0.2/0.3	0.1	0.08	0.1
	M a.b	Bit memory	1**/2	0.2/0.3	0.1	0.08	0.1
	L a.b	Local data bit	2	0.3	0.1	0.08	0.1
	DBX a.b	Data bit	2	0.3	0.1	0.08	0.1
	DIX a.b	Instance data bit	2	0.3	0.1	0.08	0.1
	c [d]	Memory-indirect, area-internal	2	0.3+	0.1+	0.08+	0.1+
	c [AR1,m]	Register-ind., area-internal (AR1)	2	0.3+	0.1+	0.08+	0.1+
	c [AR2,m]	Register-ind., area-internal (AR2)	2	0.3+	0.1+	0.08+	0.1+
	[AR1,m]	Area-crossing (AR1)	2	0.3+	0.1+	0.08+	0.1+
	[AR2,m]	Area-crossing (AR2)	2	0.3+	0.1+	0.08+	0.1+
	Parameter	Via parameter	2	0.3+	0.1+	0.08+	0.1+

Status word for: **O, ON**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	Yes
Instruction affects:	–	–	–	–	–	0	Yes	Yes	1

\+ Plus time required for loading the address of the instruction (see page 20)
* With direct instruction addressing; Address area 0 to 127
** With direct instruction addressing; Address area 0 to 255

Bit Logic Instructions, continued

Instr.	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
X/XN		Exclusive OR/ Exclusive OR NOT					
	I/Q a.b	Input/output	2	0.3	0.1	0.08	0.1
	M a.b	Bit memory	2	0.3	0.1	0.08	0.1
	L a.b	Local data bit	2	0.3	0.1	0.08	0.1
	DBX a.b	Data bit	2	0.3	0.1	0.08	0.1
	DIX a.b	Instance data bit	2	0.3	0.1	0.08	0.1
	c [d]	Memory-indirect, area-internal	2	0.3+	0.1+	0.08+	0.1+
	c [AR1,m]	Register-ind., area-internal (AR1)	2	0.3+	0.1+	0.08+	0.1+
	c [AR2,m]	Register-ind., area-internal (AR2)	2	0.3+	0.1+	0.08+	0.1+
	[AR1,m]	Area-crossing (AR1)	2	0.3+	0.1+	0.08+	0.1+
	[AR2,m]	Area-crossing (AR2)	2	0.3+	0.1+	0.08+	0.1+
	Parameter	Via parameter	2	0.3+	0.1+	0.08+	0.1+

Status word for: **X, XN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	Yes
Instruction affects:	–	–	–	–	–	0	Yes	Yes	1

+Plus time required for loading the address of the instruction (see page 20)

Bit Logic Instructions with Parenthetical Expressions

Saving the RLO and OR bits and the relevant function identifier (A, AN, ...) to the nesting stack. Seven nesting levels are possible per block. After the right parenthesis, the logic operation indicated by the function identifier is performed on the saved RLO and the current RLO; the current OR is overwritten with the saved OR.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
A(		AND left parenthesis	1	0.1	0.1	0.08	0.1
AN(		AND NOT left parenthesis	1	0.1	0.1	0.08	0.1
O(		OR left parenthesis	1	0.1	0.1	0.08	0.1
ON(		OR NOT left parenthesis	1	0.1	0.1	0.08	0.1
X(		Exclusive OR left parenthesis	1	0.1	0.1	0.08	0.1
XN(		Exclusive OR NOT left parenthesis	1	0.1	0.1	0.08	0.1

Status word for: **A(, AN(, O(, ON(, X(, XN(**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	Yes	–	Yes	Yes
Instruction affects:	–	–	–	–	–	0	1	–	0

Bit Logic Instructions with Parenthetical Expressions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
)		Right parenthesis, removing an entry from the nesting stack.	1	0.1	0.1	0.08	0.1

Status word for:)	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	Yes	1	Yes	1

ORing of AND Instructions

The ORing of AND instructions is implemented according to the rule: AND before OR.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
O		ORing of AND operations according to the rule: AND before OR	1	0.1	0.1	0.08	0.1

Status word for: **O**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	Yes
Instruction affects:	–	–	–	–	–	Yes	1	–	Yes

Logic Instructions with Timers and Counters

Examining the status of the addressed timer/counter and gating the result with the RLO according to the appropriate logic function.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
A/AN		AND/AND NOT					
	T f	Timer	1[1)]/2	0,3	0.1	0.08	0.1
	T [e]	Timer, memory-indirect addressing	2	0.3+	0.1+	0.08+	0.1+
	C f	Counter	1[1)]/2	0,3	0.1	0.08	0.1
	C [e]	Counter, memory-indirect addressing	2	0.3+	0.1+	0.08+	0.1+
	Timer para. Counter para.	Timer/counter (addressing via parameter)	2	0.3+ 0.3+	0.1+ 0.1+	0.08+ 0.08+	0.1+ 0.1+

Status word for: **A, AN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	Yes	–	Yes	Yes
Instruction affects:	–	–	–	–	–	Yes	Yes	Yes	1

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing ;Address area 0 to 255

Logic Instructions with Timers and Counters, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
O/ON		OR/OR NOT					
	T f	Timer	1[1)]/2	0.3	0.1	0.08	0.1
	T [e]	Timer, memory-indirect addr.	2	0.3+	0.1+	0.08+	0.1+
	C f	Counter	1[1)]/2	0.3	0.1	0.08	0.1
	C [e]	Counter, memory-indirect addressing	2	0.3+	0.1+	0.08+	0.1+
	Timer para. Counter para.	Timer/counter (addressing via parameter)	2	0.3+ 0.3+	0.1+ 0.1+	0.08+ 0.08+	0.1+ 0.1+
X/XN		EXCLUSIVE OR/EXCLUSIVE OR NOT					
	T f	Timer	2	0.3	0.1	0.08	0.1
	T [e]	Timer, memory-indirect addr.	2	0.3+	0.1+	0.08+	0.1+
	C f	Counter	2	0.3	0.1	0.08	0.1
	C [e]	Counter, mem.-indirect addr.	2	0.3+	0.1+	0.08+	0.1+
	Timer para. Counter para.	EXCLUSIVE OR timer/counter (addressing via parameter)	2	0.3+ 0.3+	0.1+ 0.1+	0.08+ 0.08+	0.1+ 0.1+

Status word for: **O, ON, X, XN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	Yes
Instruction affects:	–	–	–	–	–	0	Yes	Yes	1

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing; address area 0 to 255

Word Logic Instructions with the Contents of Accumulator 1

Gating the contents of ACCU1 and/or ACCU1-L with a word or double word according to the appropriate function. The word or double word is either specified in the instruction as an address or is in ACCU2. The result is in ACCU1 and/or ACCU1-L.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
AW		AND ACCU2-L	1	0.1	0.1	0.08	0.1
AW	W#16#p	AND 16-bit constant	2	0.2	0.1	0.08	0.1
OW		OR ACCU2-L	1	0.1	0.1	0.08	0.1
OW	W#16#p	OR 16-bit constant	2	0.2	0.1	0.08	0.1
XOW		EXCLUSIVE OR ACCU2-L	1	0.1	0.1	0.08	0.1
XOW	W#16#p	EXCLUSIVE OR 16-bit constant	2	0.2	0.1	0.08	0.1

Status word for: **AW, OW, XOW**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	0	0	–	–	–	–	–

Word Logic Instructions with the Contents of Accumulator 1, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
AD		AND ACCU2	1	0.1	0.1	0.08	0.1
AD	DW#16#p	AND 32-bit constant	3	0.3	0.15	0.12	0.15
OD		OR ACCU2	1	0.1	0.1	0.08	0.1
OD	DW#16#p	OR 32-bit constant	3	0.3	0.15	0.12	0.15
XOD		EXCLUSIVE OR ACCU2	1	0.1	0.1	0.08	0.1
XOD	DW#16#p	EXCLUSIVE OR 32-bit constant	3	0.3	0.15	0.12	0.15

Status word for: **UD, OD, XOD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	0	0	–	–	–	–	–

Evaluating Conditions Using AND, OR and EXCLUSIVE OR

All logic instructions generate a result (new RLO). The first instruction in a logic string generates the new RLO from the signal state scanned. The subsequent logic instructions generate the new RL from the signal state scanned and the old RLO. The logic string ends with an instruction which limits the RLO (e.g. a memory instruction); that is, the $\overline{FC}$ bit is set to zero.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
A/AN O/ON X/XN	==0	AND/AND NOT OR/OR-NOT EXCLUSIVE OR/ EXCLUSIVE-OR-NOT Result=0 (A1=0 and A0=0)	1	0.1	0.1	0.08	0.1
	>0	Result>0 (CC1=1 and CC0=0)	1	0.1	0.1	0.08	0.1
	<0	Result<0 (CC1=0 and CC0=1)	1	0.1	0.1	0.08	0.1
	<>0	Result≠0 ((CC1=0 and CC0=1) or (CC1=1 and CC0=0))	1	0.1	0.1	0.08	0.1

Status word for: **A/AN/O/ON/X/XN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	Yes	Yes	–	–	Yes	–	Yes	Yes
Instruction affects:	–	–	–	–	–	Yes	Yes	Yes	1

Evaluating Conditions Using AND, OR and EXCLUSIVE OR, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
A/AN O/ON X/XN	>=0	Result>=0 ((CC1=1 and CC0=0) or (CC1=0 and CC0=0))	1	0.1	0.1	0.08	0.1
	<=0	Result<=0 ((CC1=0 and CC0=1) or (CC1=0 and CC0=0))	1	0.1	0.1	0.08	0.1

Status word for: **A/AN/O/ON/X/XN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	Yes	Yes	–	–	Yes	–	Yes	Yes
Instruction affects:	–	–	–	–	–	Yes	Yes	Yes	1

Evaluating Conditions Using AND, OR and EXCLUSIVE OR, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
A/AN O/ON X/XN	UO	AND/AND-NOT OR/OR-NOT EXCLUSIVE-OR/ EXCLUSIVE-OR-NOT Unordered math instruction (CC1=1 and CC0=1)	1	0.1	0.1	0.08	0.1
	OS	AND OS=1	1	0.1	0.1	0.08	0.1
	BR	AND BR=1	1	0.1	0.1	0.08	0.1
	OV	AND OV=1	1	0.1	0.1	0.08	0.1

Status word for: **A/AN/O/ON/X/XN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	Yes	Yes	Yes	Yes	Yes	Yes	–	Yes	Yes
Instruction affects:	–	–	–	–	–	Yes	Yes	Yes	1

Edge-Triggered Instructions

The current RLO is compared with the status of the instruction or “edge bit memory”. FP detects a change from “0” to “1”; FN detects a change from “1” to “0”.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
FP/FN	I/Q a.b	The positive/negative edge is indicated by RLO = 1. The bit addressed in the instruction is the auxiliary edge bit memory.	2	0.4	0.2	0.16	0.2
	M a.b		2	0.4	0.2	0.16	0.2
	L a.b[1)]		2	0.4	0.2	0.16	0.2
	DBX a.b		2	0.4	0.2	0.16	0.2
	DIX a.b		2	0.4	0.2	0.16	0.2
	c [d]		2	0.4+	0.2+	0.16+	0.2+
	c [AR1,m]		2	0.4+	0.2+	0.16+	0.2+
	c [AR2,m]		2	0.4+	0.2+	0.16+	0.2+
	[AR1,m]		2	0.4+	0.2+	0.16+	0.2+
	[AR2,m]		2	0.4+	0.2+	0.16+	0.2+
	Parameter		2	0.4+	0.2+	0.16+	0.2+

Status word for: **FP, FN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	Yes	Yes	1

\+ Plus time required for loading the address of the instruction (see page 20)

1) Unnecessary if the bit being monitored is in the process image (local data of a block are only valid while the block is running).

Setting/Resetting Bit Addresses

Assigning the value “1” or “0” to the addressed instruction when RLO = 1. The instructions can be dependent on the MCR (see page 93).

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
S		Set addressed bit to “1”					
R		Set addressed bit to “0”					
	I/Q a.b	Input/output	1[1]/2	0.3/0.4	0.2	0.16	0.2
	M a.b	Bit memory	1[2]/2	0.3/0.4	0.2	0.16	0.2
	L a.b	Local data bit	2	0.4	0.2	0.16	0.2
	DBX a.b	Data bit	2	0.4	0.2	0.16	0.2
	DIX a.b	Instance data bit	2	0.4	0.2	0.16	0.2
	c [d]	Memory-indirect, area-internal	2	0.4+	0.2+	0.16+	0.2+
	c [AR1,m]	Register-indirect, area-internal (AR1)	2	0.4+	0.2+	0.16+	0.2+
	c [AR2,m]	Register-indirect, area-internal (AR2)	2	0.4+	0.2+	0.16+	0.2+
	[AR1,m]	Area-crossing (AR1)	2	0.4+	0.2+	0.16+	0.2+
	[AR2,m]	Area-crossing (AR2)	2	0.4+	0.2+	0.16+	0.2+
	Parameter	Via parameter	2	0.4+	0.2+	0.16+	0.2+

Status word for: **S, R**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	Yes	–	0

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing; Address area 0 to 127
2) With direct instruction addressing; Address area 0 to 255

Setting/Resetting Bit Addresses, continued

The RLO is written to the address of the instruction. The instructions can be dependent on the MCR (see page 93).

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
=		Assign RLO					
	I/Q a.b	To input/output	1[1])/2	0.3/0.4	0.2	0.16	0.2
	M a.b	To bit memory	1[2])/2	0.3/0.4	0.2	0.16	0.2
	L a.b	To local data bit	2	0.4	0.2	0.16	0.2
	DBX a.b	To data bit	2	0.4	0.2	0.16	0.2
	DIX a.b	To instance data bit	2	0.4	0.2	0.16	0.2
	c [d]	Memory-indirect, area-internal	2	0.4+	0.2+	0.16+	0.2+
	c [AR1,m]	Register-indirect, area-internal (AR1)	2	0.4+	0.2+	0.16+	0.2+
	c [AR2,m]	Register-indirect, area-internal (AR2)	2	0.4+	0.2+	0.16+	0.2+
	[AR1,m]	Area-crossing (AR1)	2	0.4+	0.2+	0.16+	0.2+
	[AR2,m]	Area-crossing (AR2)	2	0.4+	0.2+	0.16+	0.2+
	Parameter	Via parameter	2	0.4+	0.2+	0.16+	0.2+

Status word for: =	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	Yes	–	0

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing; Address area 0 to 127
2) With direct instruction addressing; Address area 0 to 255

Instructions Directly Affecting the RLO

The following instructions have a direct effect on the RLO.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
CLR		Set RLO to “0”	1	0.1	0.1	0.08	0.1

Status word for: **CLR**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	0	0	0	0

SET		Set RLO to “1”	1	0.1	0.1	0.08	0.1

Status word for: **SET**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	0	1	1	0

NOT		Negate RLO	1	0.1	0.1	0.08	0.1

Status word for: **NOT**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	Yes	–	Yes	–
Instruction affects:	–	–	–	–	–	–	1	Yes	–

SAVE		Save RLO to the BR bit	1	0.1	0.1	0.08	0.1

Status word for: **SAVE**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	Yes	–	–	–	–	–	–	–	–

Timer Instructions

Starting or resetting a timer. The time value must be in ACCU1-L. The instructions are triggered by an edge transition in the RLO; that is, when the status of the RLO has changed between two calls.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
SP	T f T [e]	Start timer as pulse on edge change from “0” to “1”	1[1)]/2	0.3/0.4 0.3+0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Timer para.		2	0.4+	0.2+	0.16+	0.2+
SE	T f T [e]	Start timer as extended pulse on edge change from “0” to “1”	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Timer para.		2	0.4+	0.2+	0.16+	0.2+
SD	T f T [e]	Start timer as ON delay on edge change from “0” to “1”	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Timer para.		2	0.4+	0.2+	0.16+	0.2+

Status word for **SP, SE, SD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	–	–	0

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing Timer No.: 0 to 255

Timer Instructions, continued

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
SS	T f T [e]	Start timer as retentive ON delay on edge change from “0” to “1”	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Timer para.		2	0.4+	0.2+	0.16+	0.2+
SF	T f T [e]	Start timer as OFF delay on edge change from “0” to “1”	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Timer para.		2	0.4+	0.2+	0.16+	0.2+

Status word for **SS, SF**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	–	–	0

Timer Instructions, continued

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
FR	T f T [e]	Enable timer for restarting on edge change from “0” to “1” (reset edge bit memory for starting timer)	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Timer para.		2	0.4+	0.2+	0.16+	0.2+
R	T f T [e]	Reset timer	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Timer para.		2	0.4+	0.2+	0.16+	0.2+

Status word for: **FR, R**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	–	–	0

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing Timer No.: 0 to 255

Counter Instructions

The count value must be in ACCU1-L in the form of a BCD number (0 - 999).

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
S	C f C [e]	Presetting of counter on edge change from “0” to “1”	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Counter para.		2	0.4+	0.2+	0.16+	0.2+
R	C f C [e]	Reset counter to “0” when RLO = “1”	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Counter para.		2	0.4+	0.2+	0.16+	0.2+
CU	C f C [e]	Increment counter by 1 on edge change from “0” to “1”	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Counter para.		2	0.4+	0.2+	0.16+	0.2+

Status word for: **S, R, CU**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	–	–	0

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing Counter No.: 0 to 255

Counter Instructions, continued

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
CD	C f C [e]	Decrement counter by 1 on edge change from “0” to “1”	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Counter para.		2	0.4+	0.2+	0.16+	0.2+
FR	C f C [e]	Enable counter on edge change from “0” to “1” (reset edge bit memory for up and down counting and setting the counter)	1[1)]/2	0.3/0.4 0.4+	0.2 0.2+	0.16 0.16+	0.2 0.2+
	Counter para.		2	0.4+	0.2+	0.16+	0.2+

Status word for: **CD, FR**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	–	–	0

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing Counter No.: 0 to 255

Load Instructions

Loading address identifiers into ACCU1. The contents of ACCU1 are first saved to ACCU2. The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L		Load ...					
	IB a	Input byte	1[1])/2	0.2/0.3	0.1	0.08	0.1
	QB a	Output byte	1[1])/2	0.2/0.3	0.1	0.08	0.1
	PIB a	Peripheral input byte [2]	2	0.3	0.1	0.08	0.1
	MB a	Bit memory byte	1[3])/2	0.2/0.3	0.1	0.08	0.1
	LB a	Local data byte	2	0.3	0.1	0.08	0.1
	DBB a	Data byte	2	0.3	0.1	0.08	0.1
	DIB a	Instance data byte	2	0.3	0.1	0.08	0.1
		... into ACCU1					
	g [d]	Memory-indirect, area-internal	2	0.3+	0.1+	0.08+	0.1+
	g [AR1,m]	Register-indirect, area-internal (AR1)	2	0.3+	0.1+	0.08+	0.1+
	g [AR2,m]	Register-indirect, area-internal (AR2)	2	0.3+	0.1+	0.08+	0.1+
	B[AR1,m]	Area-crossing (AR1)	2	0.3+	0.1+	0.08+	0.1+
	B[AR2,m]	Area-crossing (AR2)	2	0.3+	0.1+	0.08+	0.1+
	Parameter	Via parameter	2	0.3+	0.1+	0.08+	0.1+

\+ Plus time required for loading the address of the instruction (see page 20)
1) With indirect instruction addressing; Address area 0 to 127
2) The following peripheral acknowledgement time must be observed with CPU 414-4H and CPU 417-4H: solo 34 µs, redundant 64 µs
3) With direct instruction addressing; Address area 0 to 255

Load Instructions, continued

With direct addressing, access to odd word addresses takes an additional 0.08 µs for all CPU 416 operations and an additional 0.1 µs for all CPU 414 operations.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L		Load ...					
	IW a	Input word	1[1]/2	0.2/0.3	0.1	0.08	0.1
	QW	Output word	1[1]/2	0.2/0.3	0.1	0.08	0.1
	PIW a	Peripheral input word [2]	2	0.3	0.1	0.08	0.1
	MW a	Bit memory word	1[3]/2	0.2/0.3	0.1	0.08	0.1
	LW a	Local data word	2	0.3	0.1	0.08	0.1
	DBW a	Data word	2	0.3	0.1	0.08	0.1
	DIW a	Instance data word	2	0.3	0.1	0.08	0.1
		... into ACCU1-L					
	h [d]	Memory-indirect, area-internal	2	0.3+	0.1+	0.08+	0.1+
	h [AR1,m]	Register-indirect, area-internal (AR1)	2	0.3+	0.1+	0.08+	0.1+
	h [AR2,m]	Register-indirect, area-internal (AR2)	2	0.3+	0.1+	0.08+	0.1+
	W[AR1,m]	Area-crossing (AR1)	2	0.3+	0.1+	0.08+	0.1+
	W[AR2,m]	Area-crossing (AR2)	2	0.3+	0.1+	0.08+	0.1+
	Parameter	Via parameter	2	0.3+	0.1+	0.08+	0.1+

\+ Plus time required for loading the address of the instruction (see page 20)
1) With indirect instruction addressing; Address area 0 to 127
2) The following peripheral acknowledgement time must be observed with CPU 414-4H and CPU 417-4H: solo 37 µs, redundant 67 µs
3) With direct instruction addressing; Address area 0 to 255

Load Instructions, continued

The execution times given in the table for direct addressing are extended by 0.1 µs (CPU 412), 0.1 µs (CPU 414), 0.8 µs (CPU 416) or 0.1 µs (CPU 417) when accessing odd-numbered addresses.

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L		Load ...					
	IDa	Input double word	1[1)]/2	0.3/0.4	0.2	0.16	0.2
	QD a	Output double word	1[1)]/2	0.3/0.4	0.2	0.16	0.2
	PID a	Peripheral input double word [2)]	2	0.3/0.4	0.2	0.16	0.2
	MD a	Bit memory double word	1[3)]/2	0.3/0.4	0.2	0.16	0.2
	LD a	Local data double word	2	0.4	0.2	0.16	0.2
	DBD a	Data double word	2	0.4	0.2	0.16	0.2
	DID a	Instance data double word	2	0.4	0.2	0.16	0.2
		... in ACCU1					
	i [d]	Memory-indirect, area internal	2	0.4+	0.2+	0.16+	0.2+
	i [AR1,m]	Register-ind., area internal (AR1)	2	0.4+	0.2+	0.16+	0.2+
	i [AR2,m]	Register-ind., area internal (AR2)	2	0.4+	0.2+	0.16+	0.2+
	D[AR1,m]	Area-crossing (AR1)	2	0.4+	0.2+	0.16+	0.2+
	D[AR2,m]	Area-crossing (AR2)	2	0.4+	0.2+	0.16+	0.2+
	Parameter	Via parameter	2	0.4+	0.2+	0.16+	0.2+

\+ Plus time required for loading the address of the instruction (see page 20)
1) With indirect instruction addressing; Address area 0 to 127
2) The following peripheral acknowledgement time must be observed with CPU 414-4H and CPU 417-4H: solo 41 µs, redundant 71 µs
3) With direct instruction addressing; Address area 0 to 255

Load Instructions, continued

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L		Load ...					
	k8	8-bit constant into ACCU1-LL	2	0.2	0.1	0.08	0.1
	k16	16-bit constant into ACCU1-L	2	0.2	0.1	0.08	0.1
	k32	32-bit constant into ACCU1	3	0.3	0.15	0.12	0.15
	Parameter	Load constant into ACCU1 (addressed via parameter)	2	0.2/0.3+	0.1+	0.08+	0.1+
L	2#n	Load 16-bit binary constant into ACCU1-L	2	0.2	0.1	0.08	0.1
		Load 32-bit binary constant into ACCU1	3	0.3	0.15	0.12	0.15
	B#16#p	Load 8-bit-hexadecimal constant into ACCU1-L	1	0.1	0.1	0.08	0.1
L	W#16#p	Load 16-bit hexadecimal constant into ACCU1-L	2	0.2	0.1	0.08	0.1
	DW#16#p	Load 32-bit hexadecimal constant into ACCU1	3	0.3	0.15	0.12	0.15

Load Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L	'x'	Load 1 character	2	0.2	0.1	0.08	0.1
	'xx'	Load 2 characters	2	0.2	0.1	0.08	0.1
	'xxx'	Load 3 characters	3	0.3	0.15	0.12	0.15
	'xxxx'	Load 4 characters	3	0.3	0.15	0.12	0.15
L	D# time value	Load IEC date	3	0.3	0.15	0.12	0.15
L	S5T# time value	Load S7 time constant (16 bits)	2	0.2	0.1	0.08	0.1
L	TOD# time value	Load IEC time constant	3	0.3	0.15	0.12	0.15
L	T# time value	Load 16-bit time constant	2	0.2	0.1	0.08	0.1
		Load 32-bit time constant	3	0.3	0.15	0.12	0.15
L	C# count value	Load counter constant (BCD code)	2	0.2	0.1	0.08	0.1
L	B# (b1, b2)	Load constant as byte (b1, b2)	2	0.2	0.1	0.08	0.1
	B# (b1, b2, b3, b4)	Load constant as 4 bytes (b1, b2, b3, b4)	3	0.3	0.15	0.12	0.15

Load Instructions, continued

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L	P# bit pointer	Load bit pointer	3	0.3	0.15	0.12	0.15
L	L# integer	Load 32-bit integer constant	3	0.3	0.15	0.12	0.15
L	Real number	Load floating-point number	3	0.3	0.15	0.12	0.15

Load Instructions for Timers and Counters

Loading a time value or count value into ACCU1. The contents of ACCU1 are first saved to ACCU2. The bits of the status word are not affected.

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L	T f T (e)	Load time value	1[1])/2 2	0.2/0.3 0.3+	0.1 0.1+	0.08 0.08+	0.1 0.1+
	Timer para.	Load time value (addressed via parameter)	2	0.3+	0.1+	0.08+	0.1+
L	C f C (e)	Load count value	1[1])/2 2	0.2/0.3 0.3+	0.1 0.1+	0.08 0.08+	0.1 0.1+
	Counter para.	Load count value (addressed via parameter)	2	0.3+	0.1+	0.08+	0.1+
LC	T f T (e)	Load time value in BCD	1[1])/2 2	0.3 0.3+	0.3 0.3+	0.24 0.24+	0.3 0.3+
	Timer para.	Load time value in BCD (addressed via parameter)	2	0.3+	0.3+	0.24+	0.3+
LC	C f C (e)	Load count value in BCD	1[1])/2 2	0.3 0.3+	0.3 0.3+	0.24 0.24+	0.3 0.3+
	Counter para.	Load count value in BCD (addressed via parameter)	2	0.3+	0.3+	0.24+	0.3+

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing; Timer/counter No.: 0 to 255

Transfer Instructions

Transferring the contents of ACCU1 to the addressed operand. Note that some instructions are affected by the MCR (see page 93). The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
T		Transfer contents of ACCU1-LL to ...					
	IB a	input byte	1[1])/2	0.2/0.3	0.1	0.08	0.1
	QB a	output byte	1[1])/2	0.2/0.3	0.1	0.08	0.1
	PQB a	peripheral output byte[2])	2	0.3	0.1	0.08	0.1
	MB a	bit memory byte	1[3])/2	0.2/0.3	0.1	0.08	0.1
	LB a	local data byte	2	0.3	0.1	0.08	0.1
	DBB a	data byte	2	0.3	0.1	0.08	0.1
	DIB a	instance data byte	2	0.3	0.1	0.08	0.1
	g [d]	Memory-indirect, area internal	2	0.3+	0.1+	0.08+	0.1+
	g [AR1,m]	Register-ind., area internal (AR1)	2	0.3+	0.1+	0.08+	0.1+
	g [AR2,m]	Register-ind., area internal (AR2)	2	0.3+	0.1+	0.08+	0.1+
	B[AR1,m]	Area-crossing (AR1)	2	0.3+	0.1+	0.08+	0.1+
	B[AR2,m]	Area-crossing (AR2)	2	0.3+	0.1+	0.08+	0.1+
	Parameter	Via parameter	2	0.3+	0.1+	0.08+	0.1+

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing; Address area 0 to 127
2) The following peripheral acknowledgement time must be observed with CPU 414-4H and CPU 417-4H: solo 29 µs, redundant 58 µs
3) With direct instruction addressing; Address area 0 to 255

Transfer Instructions, continued

With direct addressing, access to add word addresses requires an additional 0.1 µs for all CPU 417 operations, an additional 0.08 µs for all CPU 416 operations, an additional 0.1 µs for all CPU 414 operations, an additional 0.1 µs for all CPU 412 operations.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
T		Transfer contents of ACCU1-L to ...					
	IW a	input word	1[1)]/2	0.2/0.3	0.1	0.08	0.1
	QW a	output word	1[1)]/2	0.2/0.3	0.1	0.08	0.1
	PQW a	peripheral output word[2)]	2	0.3	0.1	0.08	0.1
	MW a	bit memory word	1[3)]/2	0.2/0.3	0.1	0.08	0.1
	LW a	local data word	2	0.3	0.1	0.08	0.1
	DBW a	data word	2	0.3	0.1	0.08	0.1
	DIW a	instance data word	2	0.3	0.1	0.08	0.1
	h [d]	Memory-indirect, area internal	2	0.3+	0.1+	0.08+	0.1+
	h [AR1.m]	Register-ind., area internal (AR1)	2	0.3+	0.1+	0.08+	0.1+
	h [AR2,m]	Register-ind., area internal (AR2)	2	0.3+	0.1+	0.08+	0.1+
	W[AR1.m]	Area-crossing (AR1)	2	0.3+	0.1+	0.08+	0.1+
	W[AR2,m]	Area-crossing (AR2)	2	0.3+	0.1+	0.08+	0.1+
	Parameter	Via parameter	2	0.3+	0.1+	0.08+	0.1+

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing; Address area 0 to 127
2) The following peripheral acknowledgement time must be observed with CPU 414-4H and CPU 417-4H: solo 32 µs, redundant 61 µs
3) With direct instruction addressing; Address area 0 to 255

Transfer Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
T		Transfer contents of ACCU1 to ...					
	ID a	Input double word	1[1]/2	0.3/0.4	0.2	0.16	0.2
	QD a	Output double word	1[1]/2	0.3/0.4	0.2	0.16	0.2
	PQD a	periph. output double word [2]	2	0.4	0.2	0.16	0.2
	MD a	Bit memory double word	1[3]/2	0.3/0.4	0.2	0.16	0.2
	LD a	Local data double word	2	0.4	0.2	0.16	0.2
	DBD a	Data double word	2	0.4	0.2	0.16	0.2
	DID a	Instance data double word	2	0.4	0.2	0.16	0.2
T	i [d]	Memory-indirect, area internal	2	0.4+	0.2+	0.16+	0.2+
	i [AR1,m]	Register-ind., area internal (AR1)	2	0.4+	0.2+	0.16+	0.2+
	i [AR2,m]	Register-ind., area internal (AR2)	2	0.4+	0.2+	0.16+	0.2+
	D[AR1,m]	Area-crossing (AR1)	2	0.4+	0.2+	0.16+	0.2+
	D[AR2,m]	Area-crossing (AR2)	2	0.4+	0.2+	0.16+	0.2+
	Parameter	Via parameter	2	0.4+	0.2+	0.16+	0.2+

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction addressing; Address area 0 to 127
2) The following peripheral acknowledgement time must be observed with CPU 414-4H and CPU 417-4H: solo 36 µs, redundant 65 µs
3) With direct instruction addressing; Address area 0 to 255

Load and Transfer Instructions for Address Registers

Loading a double word from a memory area or register into address register 1 (AR1) or address register 2 (AR2). The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
LAR1		Load contents from ...					
	–	ACCU1	1	0.2	0.2	0.16	0.2
	AR2	Address register 2	1	0.2	0.2	0.16	0.2
	DBD a	Data double word	2	0.4	0.3	0.24	0.3
	DID a	Instance data double word	2	0.4	0.3	0.24	0.3
	m	32-bit constant as pointer	3	0.3	0.2	0.16	0.2
	LD a	Local data double word	2	0.4	0.3	0.24	0.3
	MD a	Bit memory double word	2	0.4	0.3	0.24	0.3
		... into AR1					
LAR2		Load contents from ...					
	–	ACCU1	1	0.2	0.2	0.16	0.2
	DBD a	Data double word	2	0.4	0.3	0.24	0.3
	DID a	Instance data double word	2	0.4	0.3	0.24	0.3
	m	32-bit constant as pointer	3	0.3	0.2	0.16	0.2
	LD a	Local data double word	2	0.4	0.3	0.24	0.3
	MD a	Bit memory double word	2	0.4	0.3	0.24	0.3
		... into AR2					

Load and Transfer Instructions forAddress Registers, continued

Transferring a double word from address register 1 (AR1) or address register 2 (AR2) to a memory area or register. The contents of ACCU1 are first saved to ACCU2. The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
TAR1		Transfer contents from AR1 in ...					
	–	ACCU1	1	0.1	0.1	0.08	0.1
	AR2	Address register 2	1	0.2	0.2	0.16	0.2
	DBD a	Data double word	2	0.4	0.2	0.16	0.2
	DID a	Instance data double word	2	0.4	0.2	0.16	0.2
	LD a	Local data double word	2	0.4	0.2	0.16	0.2
	MD a	Bit memory double word	2	0.4	0.2	0.16	0.2
TAR2		Transfer contents from AR2 in ...					
	–	ACCU1	1	0.1	0.1	0.08	0.1
	DBD a	Data double word	2	0.4	0.2	0.16	0.2
	DID a	Instance data double word	2	0.4	0.2	0.16	0.2
	LD a	Local data double word	2	0.4	0.2	0.16	0.2
	MD a	Bit memory double word	2	0.4	0.2	0.16	0.2
CAR		Exchange the contents of AR1 and AR2	1	0.4	0.4	0.32	0.4

Load and Transfer Instructions for the Status Word

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L	STW	Load status word into ACCU1		0.1	0.1	0.08	0.1

Status word for: **L STW**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Instruction affects:	–	–	–	–	–	–	–	–	–

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
T	STW	Transfer ACCU1 (bits 0 to 8) to the status word		0.1	0.1	0.08	0.1

Status word for: **T STW**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Load Instructions for DB Number and DB Length

Loading the number/length of a data block into ACCU1. The old contents of ACCU1 are saved to ACCU2. The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
L	DBNO	Load number of data block	1	0.1	0.1	0.08	0.1
L	DINO	Load number of instance data block	1	0.1	0.1	0.08	0.1
L	DBLG	Load length of data block into byte	1	0.1	0.1	0.08	0.1
L	DILG	Load length of instance data block into byte	1	0.1	0.1	0.08	0.1

Integer Math (16 Bits)

Math instructions on two 16-bit words. The result is written to ACCU1 and/or ACCU1-L. ACCU3 and ACCU4 are then transferred to ACCU2 and ACCU3.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
+I		Add 2 integers (16 bits) (ACCU1-L)=(ACCU1-L)+(ACCU2-L)	1	0.1	0.1	0.08	0.1
–I		Subtract 1 integer from another (16 bits) (ACCU1-L)=(ACCU2-L)-(ACCU1-L)	1	0.1	0.1	0.08	0.1
*I		Multiply 1 integer by another (16 bits) (ACCU1)=(ACCU2-L)*(ACCU1-L)	1	0.8	0.8	0.64	0.8
/I		Divide 1 integer by another (16 bits) (ACCU1-L)=(ACCU2-L):(ACCU1-L) The remainder is in ACCU1-H	1	0.8	0.8	0.64	0.8

Status word for: **+I, –I,*I, /I**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	Yes	–	–	–	–

Integer Math (32 Bits)

Math instructions on two 32-bit words. The result is written to ACCU1. ACCU3 and ACCU4 are then transferred to ACCU2 and ACCU3.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
+D		Add 2 integers (32-bit) (ACCU1)=(ACCU2)+(ACCU1)	1	0.1	0.1	0.08	0.1
–D		Subtract 1 integer from another (32 bits) (ACCU1)=(ACCU2)-(ACCU1)	1	0.1	0.1	0.08	0.1
*D		Multiply 1 integer by another (32 bits) (ACCU1)=(ACCU2)*(ACCU1)	1	1.3	1.3	1.04	1.3
/D		Divide 1 integer by another (32 bits) (ACCU1)=(ACCU2):(ACCU1)	1	1.3	1.3	1.04	1.3
MOD		Divide 1 integer by another (32 bits) and load the remainder into ACCU1: (ACCU1)=remainder of [(ACCU2):(ACCU1)]	1	1.3	1.3	1.04	1.3

Status word for: **+D, –D,*D, /D, MOD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	Yes	–	–	–	–

Floating-Point Math (32 Bits)

The result of the math instruction is in ACCU1. ACCU3 and ACCU4 are then transferred to ACCU2 and ACCU3.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
+R		Add 2 real numbers (32 bits) (ACCU1)=(ACCU2)+(ACCU1)	1	0.6	0.6	0.48	0.6
–R		Subtract 1 real number from another (32 bits) (ACCU1)=(ACCU2)-(ACCU1)	1	0.6	0.6	0.48	0.6
*R		Multiply 1 real number by another (32 bits) (ACCU1)=(ACCU2)*(ACCU1)	1	1.4	1.4	1.12	1.4
/R		Divide 1 real number by another (32 bits) (ACCU1)=(ACCU2):(ACCU1)	1	2.1	2.1	1.68	2.1

Status word for: **+R, –R, *R, /R**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	Yes	–	–	–	–

Floating-Point Math (32 Bits), continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
NEGR		Negate the real number in ACCU1	1	0.1	0.1	0.08	0.1
ABS		Form the absolute value of the real number in ACCU1	1	0.1	0.1	0.08	0.1

Status word for: **NEGR, ABS**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

Square Root and Square Instructions (32 Bits)

The result of the instruction is in ACCU1. The SQRT instruction can be interrupted.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
SQRT		Calculate the square root of a real number in ACCU1	1	72	40	37 - 39	40
SQR		Form the square of the real number in ACCU1	1	1.4	1.4	1.12	1.4

Status word for: **SQRT, SQR**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	Yes	–	–	–	–

Logarithmic Function (32 Bits)

The result of the logarithmic function is in ACCU1. The instructions can be interrupted.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
LN		Form the natural logarithm of a real number in ACCU1	1	63	35	33	35
EXP		Calculate the exponential value of a real number in ACCU1 to the base e (= 2.71828)	1	63	35	32 - 34	35

Status word for: **LN, EXP**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	Yes	–	–	–	–

Trigonometrical Functions (32 Bits)

The result of the instruction is in ACCU1. The instructions can be interrupted.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
SIN		Calculate the sine of a real number	1	56	31	30	31
ASIN		Calculate the arcsine of a real number	1	117 - 133	65 - 74	62 - 70	65 - 74
COS		Calculate the cosine of a real number	1	58	32	30	32
ACOS		Calculate the arccosine of a real number	1	122 - 139	68 - 77	65 - 72	68 - 77
TAN		Calculate the tangent of a real number	1	58 - 63	32 - 35	30 - 33	32 - 35
ATAN		Calculate the arctangent of a real number	1	43 - 58	24 - 32	23 - 30	24 - 32

Status word for: **SIN, ASIN, COS, ACOS, TAN, ATAN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	Yes	–	–	–	–

Adding Constants

Adding integer constants and storing the result in ACCU1. The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
+	i8	Add an 8-bit integer constant	1	0.1	0.1	0.08	0.1
+	i16	Add a 16-bit integer constant	2	0.2	0.1	0.08	0.1
+	i32	Add a 32-bit integer constant	3	0.3	0.15	0.12	0.15

Adding Using Address Registers

Adding a 16-bit integer to the contents of the address register. The value is either specified as an address in the instruction or is in ACCU1-L. The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
+AR1		Add the contents of ACCU1-L to those of AR1	1	0.2	0.2	0.16	0.2
+AR1	m (0 to 4095)	Add a pointer constant to the contents of AR1	2	0.2	0.2	0.16	0.2
+AR2		Add the contents of ACCU1-L to those of AR2	1	0.2	0.2	0.16	0.2
+AR2	m (0 to 4095)	Add pointer constant to the contents of AR2	2	0.2	0.2	0.16	0.2

Comparison Instructions (16-Bit Integers)

Comparing the 16-bit integers in ACCU1-L and ACCU2-L. RLO = 1 if the condition is satisfied.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
==I		ACCU2-L=ACCU1-L	1	0.1	0.1	0.08	0.1
<>I		ACCU2-L≠ACCU1-L	1	0.1	0.1	0.08	0.1
<I		ACCU2-L<ACCU1-L	1	0.1	0.1	0.08	0.1
<=I		ACCU2-L<=ACCU1-L	1	0.1	0.1	0.08	0.1
>I		ACCU2-L>ACCU1-L	1	0.1	0.1	0.08	0.1
>=I		ACCU2-L>=ACCU1-L	1	0.1	0.1	0.08	0.1

Status word for: **==I, <>I, <I, <=I, >I, >=I**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	0	–	0	Yes	Yes	1

Comparison Instructions (32-Bit Integers)

Comparing the 32-bit integers in ACCU1 and ACCU2. RLO = 1 if the condition is satisfied.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
==D		ACCU2=ACCU1	1	0.1	0.1	0.08	0.1
<>D		ACCU2≠ACCU1	1	0.1	0.1	0.08	0.1
<D		ACCU2<ACCU1	1	0.1	0.1	0.08	0.1
<=D		ACCU2<=ACCU1	1	0.1	0.1	0.08	0.1
>D		ACCU2>ACCU1	1	0.1	0.1	0.08	0.1
>=D		ACCU2>=ACCU1	1	0.1	0.1	0.08	0.1

Status word for: **==D,< >D, <D, <=D, >D, >=D**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	0	–	0	Yes	Yes	1

Comparison Instructions (32-Bit Real Numbers)

Comparing the 32-bit real numbers in ACCU1 and ACCU2. RLO = 1 if the condition is satisfied.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
==R		ACCU2=ACCU1	1	0.1	0.1	0.08	0.1
<>R		ACCU2≠ACCU1	1	0.1	0.1	0.08	0.1
<R		ACCU2<ACCU1	1	0.1	0.1	0.08	0.1
<=R		ACCU2<=ACCU1	1	0.1	0.1	0.08	0.1
>R		ACCU2>ACCU1	1	0.1	0.1	0.08	0.1
>=R		ACCU2>=ACCU1	1	0.1	0.1	0.08	0.1

Status word for: **==R, <>R, <R, <=R, >R, >=R**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	Yes	0	Yes	Yes	1

Shift Instructions

Shifting the contents of ACCU1 and ACCU1-L to the left or right by the specified number of places. If no address identifier is specified, the contents of ACCU2-LL are used as the number of places. The last bit shifted is loaded into condition code bit CC 1.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
SLW[1)]		Shift the contents of ACCU1-L to the left. Positions that become free are provided with zeros.	1	0.1	0.1	0.08	0.1
SLW	0 ... 15						
SLD		Shift the contents of ACCU1 to the left. Positions that become free are provided with zeros.	1	0.1	0.1	0.08	0.1
SLD	0 ... 32						
SRW[1)]		Shift the contents of ACCU1-L to the right. Positions that become free are provided with zeros.	1	0.1	0.1	0.08	0.1
SRW	0 ... 15						

Status word for: **SLW, SLD, SRW**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	0	0	–	–	–	–	–

1) No. of places shifted: 0 to 16

Shift Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
SRD		Shift the contents of ACCU1 to the right. Positions that become free are provided with zeros.	1	0.1	0.1	0.08	0.1
SRD	0 ... 32						
SSI[1)]		Shift the contents of ACCU1-L with sign to the right. Positions that become free are provided with with the sign (bit 15).	1	0.1	0.1	0.08	0.1
SSI	0 ... 15						
SSD		Shift the contents of ACCU1 with sign to the right. Positions that become free are provided with with the sign (bit 31).	1	0.1	0.1	0.08	0.1
SSD	0 ... 32						

Status word for: **SRD,SSI, SSD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	0	0	–	–	–	–	–

1) No. of places shifted: 0 to 16

Rotate Instructions

Rotate the contents of ACCU1 to the left or right by the specified number of places. If no address identifier is specified, the contents of ACCU2-LL are used as the number of places. The last bit shifted is loaded into condition code bit CC1.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
RLD		Rotate the contents of ACCU1 to the left	1	0.1	0.1	0.08	0.1
RLD	0 ... 32						
RRD		Rotate the contents of ACCU1 to the right	1	0.1	0.1	0.08	0.1
RRD	0 ... 32						

Status word for: **RLD, RRD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	–	–	–	–	–

Rotate Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
RLDA		Rotate the contents of ACCU1 one bit position to the left through condition code bit CC 1		0.1	0.1	0.08	0.1
RRDA		Rotate the contents of ACCU1 one bit position to the right through condition code bit CC 1		0.1	0.1	0.08	0.1

Status word for: **RLDA, RRDA**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	0	0	–	–	–	–	–

Accumulator Transfer Instructions, Incrementing and Decrementing

The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
CAW		Reverse the order of the bytes in ACCU1-L.	1	0.1	0.1	0.08	0.1
CAD		Reverse the order of the bytes in ACCU1.	1	0.1	0.1	0.08	0.1
TAK		Swap the contents of ACCU1 and ACCU2	1	0.1	0.1	0.08	0.1
ENT		The contents of ACCU2 and ACCU3 are transferred to ACCU3 and ACCU4.	1	0.1	0.1	0.08	0.1
LEAVE		The contents of ACCU3 and ACCU4 are transferred to ACCU2 and ACCU3.	1	0.1	0.1	0.08	0.1
PUSH		The contents of ACCU1, ACCU2 and ACCU3 are transferred to ACCU2, ACCU3 and ACCU4	1	0.1	0.1	0.08	0.1
POP		The contents of ACCU2, ACCU3 and ACCU4 are transferred to ACCU1, ACCU2 and ACCU3	1	0.1	0.1	0.08	0.1
INC	k8	Increment ACCU1-LL	1	0.1	0.1	0.08	0.1
DEC	k8	Decrement ACCU1-LL	1	0.1	0.1	0.08	0.1

Program Display and Null Operation Instructions

The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
BLD	k8	Program display instruction: Is treated by the CPU as a null operation instruction.	1	0.1	0.1	0.08	0.1
NOP	0 1	Null operation instruction	1	0.1	0.1	0.08	0.1

Data Type Conversion Instructions

The results of the conversion are in ACCU1.

Instruction	Addr. ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
BTI		Convert contents of ACCU1-L from BCD (0 to +/- 999) to integer (16 bits) (**B**CD **T**o **I**nt)	1	0.1	0.1	0.08	0.1
BTD		Convert contents of ACCU1 from BCD (0 to +/-9 999 999) to double integer (32 bits) (**B**CD **T**o **D**oubleint)	1	0.1	0.1	0.08	0.1
DTR		Convert contents of ACCU1 from double integer (32 bits) to real number (32 bits) (**D**oubleint **T**o **R**eal)	1	0.3	0.3	0.24	0.3
ITD		Convert contents of ACCU1 from integer (16 bits) to double integer (32 bits) (**I**nt **T**o **D**oubleint)	1	0.1	0.1	0.08	0.1

Status word for: **BTI, BTD, DTR, ITD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

Data Type Conversion Instructions, continued

Instruction	Addr. ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
ITB		Convert contents of ACCU1-L from integer (16 bits) to BCD from 0 to +/- 999 (**I**nt **T**o **B**CD)	1	0.1	0.1	0.08	0.1
DTB		Convert contents of ACCU1 from double integer (32 bits) to BCD from 0 to +/- 9 999 999 (**D**oubleint **T**o **B**CD)	1	0.2	0.2	0.16	0.2

Status word for: **ITB, DTB**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	Yes	Yes	–	–	–	–

Data Type Conversion Instructions, continued

The real number to be converted is in ACCU1.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
RND+		Convert a real number into a 32-bit integer. The number is rounded up to the next whole number.	1	0.4	0.4	0.32	0.4
RND		Convert a real number into a 32-bit integer.	1	0.4	0.4	0.32	0.4
RND-		Convert a real number into a 32-bit integer. The number is rounded down to the next whole number.	1	0.4	0.4	0.32	0.4
TRUNC		Convert a real number into a 32-bit integer. The places after the decimal point are truncated.	1	0.4	0.4	0.32	0.4

Status word for: **RND, RND-, RND+, TRUNC**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	Yes	Yes	–	–	–	–

Forming the Ones and Twos Complements

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
INVI		Form the ones complement of ACCU1-L	1	0.1	0.1	0.08	0.1
INVD		Form the ones complement of ACCU1	1	0.1	0.1	0.08	0.1

Status word for: **INVI, INVD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

NEGI		Form the twos complement of ACCU1-L (integer)	1	0.1	0.1	0.08	0.1
NEGD		Form the twos complement of ACCU1 (double integer)	1	0.1	0.1	0.08	0.1

Status word for: **NEGI, NEGD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	Yes	Yes	Yes	Yes	–	–	–	–

Block Call Instructions

The runtimes of the SFCs are specified in the chapter entitled “System Functions” as of page 102.

The information on the status word only relates to the block call itself and not to the commands called in this block.

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
CALL	FB q, DB q	Unconditional call of an FB, with parameter transfer	1[1)]/2	8.2 [3)]	3.2 [3)]	2.56 [3)]	XX
CALL	SFB q, DB q	Unconditional call of an SFB, with parameter transfer	2	8.2 [3)]	3.2 [3)]	2.56 [3)]	XX
CALL	FC q	Unconditional call of a function, with parameter transfer	1[1)]/2	4.6 [3)]	1.8 [3)]	1.44 [3)]	XX
CALL	SFC q	Unconditional call of an SFC, with parameter transfer	2	4.6 [3)]	1.8 [3)]	1.44 [3)]	XX

Status word for: **CALL**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	0	0	1	–	0

1) With direct instruction (DB) addressing; Block No. 0 to 255
3) Plus time required for supplying parameters

Block Call Instructions, continued

In-struc-tion	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
UC	FB q FC q FB [e] FC [e] Parameter	Unconditional call of blocks, without parameter transfer Memory-indirect FB call Memory-indirect FC call FB/FC call via parameter	1 [1]/2 2 2 2	2.1/2.2 2.1/2.2 2.2+ 2.2+ 2.2+	1.4 1.4 1.4+ 1.4+ 1.4+	1.12 1.12 1.12+ 1.12+ 1.12+	1.4 1.4 1.4+ 1.4+ 1.4+
CC	FB q FC q FB [e] FC [e] Parameter	Conditional call of blocks, without parameter transfer Memory-indirect FB call Memory-indirect FC call FB/FC call via parameter	1 [1]/2 2 2 2	2.3/2.4/0.4 [4] 2.3/2.4/0.4 [4] 2.4+/0.4 [4] 2.4+/0.4 [4] 2.4+/0.4 [4]	1.4/0.4 [4] 1.4/0.4 [4] 1.4+/0.4 [4] 1.4+/0.4 [4] 1.4+/0.4 [4]	1.12/0.32 [4] 1.12/0.32 [4] 1.12+/0.32 [4] 1.12+/0.32 [4] 1.12+/0.32 [4]	1.4/04 [4] 1.4/0.4 [4] 1.4+/0.4 [4] 1.4+/0.4 [4] 1.4+/0.4 [4]

Status word for: **UC, CC [2]**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	0	0	1	–	0

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction (DB) addressing; Block No. 0 to 255
2) Depending on RLO, sets RLO = 1
4) If call is not executed

Block Call Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs Direct Addressing			
				CPU 412	CPU 414	CPU 416	CPU 417
OPN		Open:	1[1])/2				
	DB q	Data block		0.6/0.7	0.3	0.24	0.3
	DI q	Instance data block		0.7	0.3	0.24	0.3
	DB [e]	Data block, memory-indirect		0.7+	0.3+	0.24+	0.3+
	DI [e]	Instance DB, memory-indirect		0.7+	0.3+	0.24+	0.3+
	Parameter	Data block using parameters		0.7+	0.3+	0.24+	0.3+

Status word for: **OPN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

\+ Plus time required for loading the address of the instruction (see page 20)
1) With direct instruction (DB) addressing; Block No. 0 to 255

Block End Instructions

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
BE		End block	1	2.8	2.0	1.60	2.0
BEU		End block unconditionally	1	2.8	2.0	1.60	2.0

Status word for: **BE, BEU**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	0	0	1	–	0

BEC		End block conditionally if RLO = “1”		3.0 0.4[1)]	2.2 0.4[1)]	1.76 0.32[1)]	2.2 0.4[1)]

Status word for: **BEC**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	Yes	0	1	1	0

1) If jump is not executed

Exchanging Shared Data Block and Instance Data Block

Exchanging the two current data blocks. The current shared data block becomes the current instance data block, and vice versa. The status word is not affected.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
CDB		Exchange shared data block and instance data block	1	0.2	0.2	0.16	

Jump Instructions

Jumping as a function of conditions.

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
JU	LABEL	Jump unconditionally	1[1]/2	0.5/0.6	0.5	0.4	0.5

Status word for: **JU**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

JC	LABEL	Jump if RLO = “1”	1[1]/2	0.5/0.6 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]
JCN	LABEL	Jump if RLO = “0”	2	0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]

Status word for: **JC, JCN**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	1	1	0

1) 1 word long for jump widths between -128 and +127
2) If jump is not executed

Jump Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
JCB	LABEL	Jump if RLO = “1”. Save the RLO in the BR bit	2	0.6/0.2 2)	0.5/0.2 2)	0.4/0.16 2)	0.5/0.2 2)
JNB	LABEL	Jump if RLO = “0”. Save the RLO in the BR bit	2	0.6/0.2 2)	0.5/0.2 2)	0.4/0.16 2)	0.5/0.2 2)

Status word for: **JCB, JNB**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	Yes	–	–	–	–	0	1	1	0

JBI	LABEL	Jump if BR = “1”	2	0.6/0.2 2)	0.5/0.2 2)	0.4/0.16 2)	0.5/0.2 2)
JNBI	LABEL	Jump if BR = “0”	2	0.6/0.2 2)	0.5/0.2 2)	0.4/0.16 2)	0.5/0.2 2)

Status word for: **JBI, JNBI**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	Yes	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	0	1	–	0

2) If jump is not executed

Jump Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
JO	LABEL	Jump on stored overflow (OV = "1")	1[1]/2	0.5/0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]

Status word for: **JO**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	Yes	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

JOS	LABEL	Jump on stored overflow (OS = "1")	2	0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]

Status word for: **JOS**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	Yes	–	–	–	–
Instruction affects:	–	–	–	–	0	–	–	–	–

1) 1 word long for jump widths between -128 and +127
2) If jump is not executed

Jump Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
JUO	LABEL	Jump if "unordered math instruction" (CC1=1 and CC0=1)	2	0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]
JZ	LABEL	Jump if result = 0 (CC1=0 and CC0=0)	1[1]/2	0.5/0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]
JP	LABEL	Jump if result > 0 (CC1=1 and CC0=0)	1[1]/2	0.5/0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]
JM	LABEL	Jump if result < 0 (CC1=0 and CC0=1)	1[1]/2	0.5/0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]
JN	LABEL	Jump if result ≠ 0 (CC1=1 and CC0=0) or (CC1=0 and CC0=1)	1[1]/2	0.5/0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]
JMZ	LABEL	Jump if result ≤ 0 (CC1=0 and CC0=1) or (CC1=0 and CC0=0)	2	0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]
JPZ	LABEL	Jump if result ≥ 0 (CC1=1 and CC0=0) or (CC1=0 and CC0=0)	2	0.6/0.2 [2]	0.5/0.2 [2]	0.4/0.16 [2]	0.5/0.2 [2]

Status word for: **JUO, JZ, JP, JM, JN, JMZ, JPZ**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	Yes	Yes	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

1) 1 word long for jump widths between -128 and +127
2) If jump is not executed

Jump Instructions, continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
JL	LABEL	Jump distributor This instruction is followed by a list of jump instructions. The address identifier is a jump label to subsequent instructions in this list. ACCU1-LL contains the number of the jump instruction to be executed (max. 254). The number of the first jump instruction is 0.	2	0.8	0.7	0.56	0.7
LOOP	LABEL	Decrement ACCU1-L and jump if ACCU1-L ≠ 0 (loop programming)	2	0.6/0.2 [1]	0.5/0.2 [1]	0.4/0.08 [1]	0.5/0.2 [1]

Status word for: **JL, LOOP**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

[1] If jump is not executed

Instructions for the Master Control Relay (MCR)

MCR=1→MCR is deactivated
MCR=0→MCR is activated; “T” and “=” instructions write zeros to the

corresponding address identifiers if RLO = “0”; “S” and ”R” instructions leave the memory contents unchanged.

Instruction	Address ID	Description	Length in Words	Execution Time in µs		
				CPU 412	CPU 414	CPU 416	CPU 417
MCR(		Open an MCR zone. Save the RLO to the MCR stack.	1	0.1	0.1	0.08	0.1

Status word for: **MCR(**	CC1	BR	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	Yes	–
Instruction affects:	–	–	–	–	–	0	1	–	0

)MCR		Close an MCR zone. Pop an entry off the MCR stack.	1	0.1	0.1	0.08	0.1

Status word for: **)MCR**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	0	1	–	0

Instructions for the Master Control Relay (MCR), continued

Instruction	Address ID	Description	Length in Words	Execution Time in µs			
				CPU 412	CPU 414	CPU 416	CPU 417
MCRA		Activate the MCR	1	0.1	0.1	0.08	0.1
MCRD		Deactivate the MCR	1	0.1	0.1	0.08	0.1

Status word for: **MCRA, MCRD**	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC
Instruction evaluates:	–	–	–	–	–	–	–	–	–
Instruction affects:	–	–	–	–	–	–	–	–	–

Oganization Blocks (OB)

A user program for the S7-400 is made up of blocks containing the statements, parameters and data for the relevant CPU. The number of blocks you can create or which are provided by the operating system is different for each of the S7-400 CPUs. You will find a detailed description of the OBs and their use in the *STEP 7 Programming Manual*.

Organization Blocks	CPU 412	CPU 414	CPU 414-4H	CPU 416	CPU 417	CPU 417-4H	Start Events (Hexadecimal Values)
Free cycle							
OB 1	x	x	x	x	x	x	1101, 1102, 1103, 1104, 1105
Time-of-day interrupts							
OB 10	x	x	x	x	x	x	1111
OB 11	x	x	x	x	x	x	1112
OB 12		x	x	x	x	x	1113
OB 13		x	x	x	x	x	1114
OB 14				x	x	x	1115
OB 15				x	x	x	1116
OB 16				x	x	x	1117
OB 17				x	x	x	1118

Oganization Blocks (OB), continued

Organization Blocks	CPU 412	CPU 414	CPU 414-4H	CPU 416	CPU 417	CPU 417-4H	Start Events (Hexadecimal Values)
Time-delay interrupts							
OB 20	x	x	x	x	x	x	1121
OB 21	x	x	x	x	x	x	1122
OB 22		x	x	x	x	x	1123
OB 23		x	x	x	x	x	1124
Timed interrupts							
OB 30				x	x	x	1131
OB 31				x	x	x	1132
OB 32	x	x	x	x	x	x	1133
OB 33		x	x	x	x	x	1134
OB 34		x	x	x	x	x	1135
OB 35	x	x	x	x	x	x	1136
OB 36				x	x	x	1137
OB 37				x	x	x	1138
OB 38				x	x	x	1139

Organization Blocks (OB), continued

Organization Blocks	CPU 412	CPU 414	CPU 414-4H	CPU 416	CPU 417	CPU 417-4H	Start Events (Hexadecimal Values)
Hardware interrupts							
OB 40	x	x	x	x	x	x	1141, 1142, 1143, 1144, 1145
OB 41	x	x	x	x	x	x	1141, 1142, 1143, 1144, 1145
OB 42		x	x	x	x	x	1141, 1142, 1143, 1144, 1145
OB 43		x	x	x	x	x	1141, 1142, 1143, 1144, 1145
OB 44				x	x	x	1141, 1142, 1143, 1144, 1145
OB 45				x	x	x	1141, 1142, 1143, 1144, 1145
OB 46				x	x	x	1141, 1142, 1143, 1144, 1145
OB 47				x	x	x	1141, 1142, 1143, 1144, 1145
Interrupt OBs for DPV1:							
OB 55	x	x	x	x	x	x	1155
OB 56	x	x	x	x	x	x	1156
OB 57	x	x	x	x	x	x	1157

Organization Blocks (OB), continued

Organization Blocks	CPU 412	CPU 414	CPU 414-4H	CPU 416	CPU 417	CPU 417-4H	Start Events (Hexadecimal Values)
Multicomputing interrupts							
OB 60	x	x		x	x		1161, 1162
Synchronous cycle interrupt:							
OB 61	x	x	x	x	x	x	1164
OB 62	x	x	x	x	x	x	1165
OB 63	x	x	x	x	x	x	1166
OB 64	x	x	x	x	x	x	1167
Redundancy error interrupts:							
OB 70			x			x	73A2, 73A3, 72A3
OB 72			x			x	7301, 7302, 7303, 7320, 7321, 7322, 7323, 7331, 7333, 7334, 7335, 7340, 7341, 7342, 7343, 7344, 7950, 7951, 7952, 7852, 7953, 7954, 7955, 7855, 7956, 73C1, 73C2
Asynchronous error interrupts:							
OB 80	x	x	x	x	x	x	3501, 3502, 3505, 3506, 3507, 350A
OB 81	x	x	x	x	x	x	3821, 3822, 3823, 3825, 3826, 3827, 3831, 3832, 3833, 3921, 3922, 3923, 3925, 3926, 3927, 3931, 3932, 3933
OB 82	x	x	x	x	x	x	3842, 3942

Organization Blocks	CPU 412	CPU 414	CPU 414-4H	CPU 416	CPU 417	CPU 417-4H	Start Events (Hexadecimal Values)
OB 83	x	x	x	x	x	x	3267, 3367, 3861, 3863, 3864, 3865, 3961, 3968
OB 85	x	x	x	x	x		35A1, 35A2, 35A3, 38B3, 38B4, 39B1, 39B2, 39B3, 39B4
OB 86	x	x	x	x	x	x	38C1, 38C2, 39C1, 38C6, 38C7, 38C8 38C4[1)], 38C5[1)], 39C3[1)], 39C4[1)], 39C5[1)]
OB 87	x	x	x	x	x	x	35D2, 35D3, 35D4, 35D5, 35E1, 35E2, 35E3, 35E4, 35E5, 35E6
OB 88	x	x	x	x	x	x	3571, 3572, 3573, 3574, 3575, 3576, 3578, 357A
Background:							
OB 90	x	x		x	x		1191, 1192, 1193, 1195
Warm restart:							
OB 100	x	x	x	x	x	x	1381, 1382, 138A, 138B
Hot restart:							
OB 101	x	x		x	x		1383, 1384
Cold restart:							
OB 102	x	x	x	x	x	x	1385, 1386, 1387, 1388
Synchronous error interrupts:							
OB 121	x	x	x	x	x	x	2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 253A, 253C, 253D, 253E, 253F
OB 122	x	x	x	x	x	x	2942, 2943, 2944, 2945

1) not CPU 412-1

Function Blocks (FB)

The following tables list the quantities, numbers and maximum sizes of the function blocks you can create for the various S7-400 CPUs.

Function Blocks	CPU 412-1	CPU 412-2	CPU 414	CPU 416	CPU 417
Quantity	256	256	1024	2048	6144
Permissible numbers	0 to 255	0 to 255	0 to 1023	0 to 2047	0 to 6143
Maximum size of a function block (code required for execution)	48 Kbytes	64 Kbytes	64 Kbytes	64 Kbytes	64 Kbytes

Functions (FC) and Data Blocks

The following tables list the quantities, numbers and maximum sizes of the functions and data blocks you can create for the various S7-400 CPUs.

Functions	CPU 412-1	CPU 412-2	CPU 414	CPU 416	CPU 417
Quantity	256	256	2048	2048	6144
Permissible numbers	0 to 255	0 to 255	0 to 2047	0 to 2047	0 to 6143
Maximum size of a function (code required for execution)	48 Kbytes	64 Kbytes	64 Kbytes	64 Kbytes	64 Kbytes

Data Blocks	CPU 412-1	CPU 412-2	CPU 414	CPU 416	CPU 417
Quantity	511	511	4095	4095	8191
Permissible numbers	1 to 511	1 to 511	1 to 4095	1 to 4095	1 to 8191
Maximum size of a data block (number of data bytes)	48 Kbytes	64 Kbytes	64 Kbytes	64 Kbytes	64 Kbytes

System Functions

The following tables show the system functions which are provided by the operating system of the S7-400 CPUs and the execution times for the various CPUs. (X: function available, execution times not yet available before printing).

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
0	SET_CLK	Set clock	340	249	215	249	289	288
1	READ_CLK	Read clock	40	29	23	29	29	54
2	SET_RTM	Set run-time meter	35	26	20	26	25	26
3	CTRL_RTM	Start and stop run-time meter	30	23	18	23	22	22
4	READ_RTM	Read run-time meter	41	30	23	30	29	58
5	GADR_LGC	Find logical address of a channel Rack 0	55	39	31	39	38	38
		internal DP	66	46	36	46	46	46
6	RD_SINFO	Read start information of current OB	54	38	30	38	39	39
7	DP_PRAL	Trigger a process interrupt at the DP master First call	294	208	166	208	--	--
		Intermediate call	43	30	24	30	--	--
		Last call	46	32	25	32	--	--

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
9	EN_MSG	Enable block-related, symbol-related, and group status messages. First call, REQ = 1	176	122	97	122	128	232
		Last call	61	44	34	44	39	62
10	DIS_MSG	Disable block-related, symbol-related, and group status messages. First call, REQ = 1	176	122	97	122	128	232
		Last call	61	44	34	44	39	63

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
11	DPSYC_FR	Synchronize groups of DP Slaves First call, internal DP interface, REQ = 1	170	110	90	110	--	--
		Intermediate call, internal DP interface, BUSY = 1 [1)]	51 + n* 4	36 + n* 3	28 + n* 2	36 + n* 3	--	--
		Last call, internal DP interface, BUSY=0 [1)]	51 + n* 4	36 + n* 3	28 + n* 2	36 + n* 3	--	--
11	DPSYC_FR	First call, external DP interface, REQ=1	94	71	60	71	--	--
		Intermediate call, external DP interface, BUSY = 1 [1)]	64 + n* 4	50 + n* 3	39 + n* 2	50 + n* 3	--	--
		Last call, external DP interface, BUSY= 0 [1)]	64 + n* 4	50 + n* 3	39 + n* 2	50 + n* 3	--	--

[1)] n = number of active jobs with the same logic address

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
12	D_ACT_DP	Deactivate and activate DP slaves via integrated DP interface, MODE = 0	117	76	61	76	--	--
12	D_ACT_DP	Deactivate and activate DP slaves via integrated DP interface, MODE = 1 First call	269	179	142	179	--	--
		Intermediate call	114	73	59	73	--	--
		Last call	231	167	121	167	--	--
12	D_ACT_DP	Deactivate and activate DP slaves via integrated DP interface, MODE = 2 First call	378	268	202	268	--	--
		Intermediate call	113	72	58	72	--	--
		Last call	119	76	62	76	--	--
12	D_ACT_DP	Deactivate and activate DP slaves via external DP interface, MODE = 0	X	X	X	X	--	--
12	D_ACT_DP	Deactivate and activate DP slaves via external DP interface, MODE = 1 First call	X	X	X	X	--	--
		Intermediate call	X	X	X	X	--	--
		Last call	X	X	X	X	--	--

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
12	D_ACT_DP	Deactivate and activate DP slaves via external DP interface, MODE = 2 First call	X	X	X	X	--	--
		Intermediate call	X	X	X	X	--	--
		Last call	X	X	X	X	--	--
13	DP_NRMDG	Read slave diagnostic data First call	300	200	165	200	210	290
		Intermediate call	--	--	--	--	79	79
		Last call (28 bytes)	180	125	100	125	101	101
14	DPRD_DAT	Read consistent user data (n bytes) via integrated DP interface 3 bytes	83	56	45	56	70	96
		via integrated DP interface 32 bytes	94	67	54	67	88	122
		via external DP interface 3 bytes	86	62	50	62	76	99
		via external DP interface 32 bytes	181	156	137	156	152	209

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
15	DPWR_DAT	Write consistent user data (n bytes) via integrated DP interface 3 bytes	84[1]/91[2]	57[1]/ 61[2]	45[1]/ 49[2]	57[1]/ 61[2]	72[1]/ 76[2]	94[1]/ 98[2]
		via integrated DP interface 32 bytes	96[1]/127[2]	67[1]/ 97[2]	53[1]/ 78[2]	67[1]/ 97[2]	88[1]/ 119[2]	110[1]/ 142[2]
		via external DP interface 3 bytes	88[1]/94[2]	62[1]/ 67[2]	50[1]/ 54[2]	62[1]/ 67[2]	77[1]/83[2]	100[1]/ 105[2]
		via external DP interface 32 bytes	178[1]/ 209[2]	150[1]/ 181[2]	130[1]/ 154[2]	150[1]/ 181[2]	171[1]/ 201[2]	193[1]/ 224[2]
17	ALARM_SQ	Generate acknowledgeable block-related messages. First call, SIG = 0 –> 1	440	305	240	305	266	358
		Empty call	130	90	72	90	90	156
18	ALARM_S	Generate unacknowledgeable block-related messages. First call, SIG = 0 –> 1	460	310	250	310	275	365
		Empty call	140	90	75	90	97	163

[1] without data transmission to the process image

[2] with data transmission to the process image

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
19	ALARM_SC	Acknowledgment status of the last ALARM_SQ entering state message.	85	60	46	60	56	82
20	BLKMOV	Copy variable within the work memory (n = number of bytes to be copied)	60 + n * 0.3	41 + n * 0.13	32 + n * 0.23	41 + n * 0.13	42 + n * 0.17	42 + n * 0.17
		Source = Load memory	1400 + n* 1.0	1160 + n* 0.7	1100 + n* 0.7	1160 + n * 0.7	1124 + n * 1.0	2065 + n * 1.98
21	FILL	Set array default variables within the work memory (n = length of target variables in bytes)	60 + n * 0.15	44 + n * 0.13	34 + n * 0.1	44 + n * 0.13	45 + n * 0.12	45 + n * 0.12
22	CREAT_DB	Create data block n = DB length [bytes]	142	94	72	94	155 + n * 0.1	424 + n * 0.1
		Occupy last free DB No. from a field of 100 DBs	606	400	320	400	2877	13601
23	DEL_DB	Delete data block	122	81	64	81	179	625
24	TEST_DB	Test data block	47	32	25	32	68	248
25	COMPRESS	Compress user memory First call (trigger)	112	78	63	78	93	173
		Intermediate call (active)	32	23	18	23	22	22

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
26	UPDAT_PI	Update process image input table (run-time entry for 1 DI 32 in the central rack)	45	35	29	35	67	103
		AI 8* 13Bit	70	59	51	59	155	192
27	UPDAT_PO	Update process image output table (run-time entry for 1 DO 32 in the central rack)	45	35	29	35	54	81
		AO 8* 13Bit	66	55	48	55	122	149
28	SET_TINT	Set time-of-day interrupt	108	75	60	75	74	98
29	CAN_TINT	Cancel time-of-day interrupt	40	29	22	29	34	34
30	ACT_TINT	Activate time-of-day interrupt	73	53	41	53	51	75
31	QRY_TINT	Query time-of-day interrupt	44	34	27	34	33	34
32	SRT_DINT	Start time-delay interrupt	65	46	36	46	44	44
33	CAN_DINT	Cancel time-delay interrupt	41	30	23	30	36	36
34	QRY_DINT	Query time-delay interrupt	43	33	26	33	32	32
35	MP_ ALM	Trigger multicomputing interrupt	240	171	138	171	--	--
36	MSK_FLT	Mask synchronous faults	30	22	17	22	21	21
37	DMSK_FLT	Demask synchronous faults	31	23	18	23	22	23
38	READ_ERR	Read error register	32	23	18	23	23	23

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
39	DIS_IRT	Discard new events Block all events (MODE = 0)	555	535	580	535	731	732
		Block all events of a priority class (MODE = 1)	70 - 190	50 - 145	40 - 160	50 - 145	42 - 194	42 - 194
		Block one event (MODE = 2)	40 - 50	30 - 37	24 - 28	30 - 37	31 - 36	31 - 37
40	EN_IRT	Stop discarding events Enable all events (MODE = 0)	555	535	580	535	736	737
		Enable all events in a priority class (MODE = 1)	70 - 190	50 - 145	40 - 160	50 - 145	42 - 197	42 - 197
		Enable an event (MODE = 2)	40 - 50	30 - 37	24 - 28	30 - 37	31 - 37	31 - 37
41	DIS_AIRT	Delay interrupt events the first time delay is activated[1)]	248	166	132	166	165	165
		if the delay is already activated	26	19	14	19	18	18

1) When activating the delay for the first time, the SFC 41 runtime depends on the priority class in which the SFC 41 is called. The specified runtime refers to the call in OB 1. It decreases while the priority class number increases.

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
42	EN_AIRT	Stop delaying interrupt events when canceling the last delay [2]	26	19	14	19	18	18
		if other delays are present	435	320	272	320	343	343
43	RE_TRIGR	Retrigger watchdog monitoring	155	104	84	104	118	307
44	REPL_VAL	Transfer substitute value to ACCU1	30	21	16	21	20	20
46	STP	Force CPU into STOP mode cannot be measured	—	—	—	—	—	—
47	WAIT	Delay program execution in addition to waiting time	13 - 18	7 - 15	4 - 11	7 - 15	6 - 13	6 - 13
48	SNC_RTCB	Synchronize slave clocks	25	19	14	19	18	41
49	LGC_GADR	Find slot with logical address	55	40	31	40	41	41
50	RD_LGADR	Find all logical addresses of a block (run-time entry for 1 DI 32 in the central rack)	146	101	80	101	104	104

[2] When cancelling the last delay, the SFC 42 runtime depends on the priority class in which the SFC 42 is called. The specified runtime refers to the call in OB 1. It decreases while the priority class number increases.

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
51	RDSYSST	List of all system status list information (0000)	618	493	395	493	477	477
		List of all system status list information (0F00)	140	97	77	97	97	97
51	RDSYSST	”Module Identification” partial list Display all data records (0011)	224	170	135	170	169	168
		Display one data record (0111)	175	125	100	125	123	122
		Display header information (0F11)	145	100	80	100	99	99
51	RDSYSST	“CPU Characteristics” partial list Display all data records (0012)	317	235	187	235	233	232
		Display one data record (0112)	190 - 215	135 - 155	108 - 123	135 - 155	135 - 155	135 - 154
		Display header information (0F12)	145	100	80	100	99	98
51	RDSYSST	“Save” partial list Display all data records (0013)	185	134	105	134	134	133
		Display one data record (0113)	185	134	105	134	134	133
		Display header information (0F13)	145	100	80	100	100	99

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
51	RDSYSST	"System Areas" partial list Display all data records (0014)	220	145	120	145	145	144
		Display one data record (0114)	170	117	93	117	117	117
		Display header information (0F14)	745	480	480	99	99	99
51	RDSYSST	"Block Types" partial list Display all data records (0015)	196	425	425	145	145	144
		Dispaly one data record (0115)	165 - 185	118 - 128	94 - 102	118 - 128	119 - 128	118 - 127
		Display header information (0F15)	142	100	78	100	98	98
51	RDSYSST	"Priority Classes" partial list Display all data records (0016)	858	740	765	740	947	947
		Display one data record (0116)	196 - 347	110 - 250	110 - 135	110 - 250	137 - 291	137 - 290
		Display header information (0F16)	153	106	85	106	107	106

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
51	RDSYSST	"Status of Module LEDs" partial list Display status of all LEDs (0019)	322	216	175	216	225	--
		Display status of one LED (0119)	225	150	120	150	151	--
		Display header information (0F19)	206	136	110	136	136	--
51	RDSYSST	"Interrupt/Error Assignment" partial list Display all data records (0021)	1225	1010	1055	1010	1298	1297
		Display all data records of one interrupt class (0121)	210 - 590	145 - 410	115 - 330	145 - 410	145 - 365	144 - 364
51	RDSYSST	Display one data record (0221)	195 - 215	135 - 150	110 - 120	135 - 150	135 - 152	135 - 151
		Display all assigned interrupts of one class (0921)	225 - 640	155 - 440	125 - 390	155 - 440	155 - 485	155 - 485
		Alternative: n= number of loaded OBs (0921)	(225/ 375)+ n*34	(155/ 260)+ n*23	(125/ 245)+ n*18	(155/ 260)+ n*23	(155/ 305)+ n*23	(155/ 305)+ n*23
51	RDSYSST	"Interrupt/Error Assignment" partial list Display all assigned interrupts (0A21)	930 - 1510	795 - 1285	835 - 1390	795 - 1285	1037 - 1697	1037 - 1697
		Display header information (0F21)	155	107	85	107	108	107

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
51	RDSYSST	"Interrupt Status" partial list Display all data records of one interrupt class (0122)	225 - 660	160 - 490	125 - 390	160 - 490	157 - 432	160 - 450
		Display one data record (0222)	210 - 225	148 - 158	118 - 128	148 - 158	148 - 155	148 - 158
		Display all assigned interrupts of one class (0822)	235 - 720	165 - 515	130 - 470	165 - 515	165 - 560	165 - 560
		Alternative: n = number of loaded OBs	(235/ 375)+ n*45	(165/ 260)+ n*35	(130/ 245)+ n*25	(165/ 260)+ n*35	(165/ 305)+ n*35	(165/ 305)+ n*35
		Display header information (0F22)	158	110	87	110	44	108
51	RDSYSST	"Status of Priority Classes" partial list Display one data record (0123)	210	147	117	147	147	147
		All priority classes in process (0223) (n= number of priority classes)	535 + n*52	450 + n*35	443 + n*28	450 + n*35	540 + n*36	540 + n*36
		Display header information (0F23)	145	100	80	100	101	100

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
51	RDSYSST	"Operating Mode" partial list Display the last operating mode transition (0124)	200	140	111	140	139	138
		Display the current operating mode	175	125	100	125	125	125
51	RDSYSST	"Status Information Communication" partial list Display status information of a communication unit (0132) INDEX = 5	205	150	120	150	157	181
51	RDSYSST	"Status Information Communication" partial list Display status information of a communication unit (0232) INDEX = 4	–	–	–	–	235	425

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
51	RDSYSST	"Start Information List" partial list All synchronization error start information of one priority class (0281)	190 - 225	128 - 155	102 - 135	128 - 155	127 - 168	127 - 167
		All start information of one priority class (0381)	210 - 395	128 - 305	102 - 255	128 - 305	128 - 318	127 - 317
		All synchronization error start information of one priority class before processing (0681)	190 - 225	128 - 155	102 - 135	128 - 155	127 - 168	127 - 167
		All start information of one priority class before processing (0781)	190 - 390	145 - 295	115 - 235	145 - 295	142 - 293	141 - 293
		All synchronization error start information of one priority class in process (0A81)	190 - 225	130 - 160	102 - 135	130 - 160	129 - 170	128 - 169
		All start information of one priority class in process (0B81)	190 - 240	130 - 170	102 - 145	130 - 170	129 - 179	129 - 179
		Display one header information (0F81)	160	112	90	112	112	111

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
51	RDSYSST	"Start Information List" partial list All synchronization error start events of one priority class (0282)	190 - 220	128 - 150	102 - 125	128 - 150	128 - 156	128 - 155
		All start events of one priority class (0382)	210 - 305	128 - 225	102 - 185	128 - 225	129 - 230	128 - 229
		All synchronization error start events of one priority class before processing (0682)	190 - 220	128 - 150	102 - 125	128 - 150	128 - 156	128 - 155
		All start events of one priority class before processing (0782)	210 - 310	145 - 225	115 - 180	145 - 225	143 - 225	142 - 223
		All synchronization error start events of one priority class in process (0A82)	190 - 220	130 - 150	102 - 125	130 - 150	130 - 158	129 - 161
		All start events of one priority class in process (0B82)	190 - 225	130 - 155	102 - 130	130 - 155	130 - 162	130 - 161
		Display one header information (0F82)	160	112	90	112	113	112

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
51	RDSYSST	"Module Status Information" partial list Display the status information of all inserted modules (n = number of the data records) (0091)	660 + n* 22	508 + n* 19	408 + n* 16	508 + n* 19	--	--
		Display the status information: of all modules/racks with incorrect type ID (0191)	570 + n* 70	427 + n* 60	365 + n* 40	405 + n* 24	--	--
		of all faulty modules (0291)	580 + n* 138	428 + n* 22	344 + n* 18	428 + n* 22	--	--
		of all unavailable modules (0391)	585 + n* 72	430 + n* 60	370 + n * 40	430 + n* 60	--	--
		Display the status information: of all submodules of the host module in the specified rack (0991)	354 + n* 30	250 + n* 26	200 + n* 21	250 + n* 26	--	--
		centralized of one module with logical base address (0C91)	200 - 315	180	145	180	177	242
		distributed of one module with logical base address (0C91)	315	225	180	225	224	289

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
51	RDSYSST	”Module Status Information” partial list of a module (distributed) with a logical base address (4C91) first call	200 - 315	145 - 240	130 - 190	145 - 240	242	305
		”Module Status Information” partial list of a module (distributed) with a logical base address (4C91) intermediate call	--	--	--	--	148	148
		”Module Status Information” partial list of a module (distributed) with a logical base address (4C91) last call	--	--	--	--	167	167
		local of all modules in the specified rack (n = number of the DS) (0D91)	377 + n* 13	275 + n* 16	240 + n* 10	275 + n * 16	260 + n * 20	405 + n * 23
		distributed of all modules in the specified distributed I/O station (0D91)	330 - 390	250 - 300	200 - 240	250 - 300	305	408 - 420
		Display one header information (0F91)	560	435	350	435	--	--

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
51	RDSYSST	"Rack/Station Status Information" partial list, local, Display the setpoint status of the racks 0 (0092)	180	127	100	127	130	154
		distributed, Display the setpoint status of distributed I/O system 1 (0092)	900	725	585	725	712	743
		local, Display the actual status of the racks 0 (0292)	180	127	103	127	131	155
		distributed, Display the actual status of distributed I/O system 1 (0292)	940	745	600	745	725	757
		Display header information (0F92)	160	113	90	113	113	113
51	RDSYSST	"Diagnostic Buffer" partial list Display all available current operating mode event information (max. 23) (00A0)	195 - 525	138 - 410	110 - 330	138 - 410	140 - 412	140 - 412
		Display the n newest entries (n = 1-23) (01A0)	195 + n* 14.5	138 + n* 12	110 + n* 9.5	138 + n* 12	140 + n* 12	140 + n* 12
		Display the standard OB start information (04A0). Max value of -04A0 is calculated	195 - 1270	138 - 1530	110 - 1095	138 - 1530	140 - 1540	140 - 1540

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
51		Display all communication information (05A0) Display all object management system information (06A0) Display all test and startup information (07A0) Display all operating mode information (08A0) Display all asynchronous error start information (09A0) Display all synchronous error start information (0AA0) Display all STOP/cancel/operating mode transition information (0BA0) Display all fault-tolerant/failsafe information (0CA0) Display all diagnostic information (0DA0) Display all user information (0EA0)	195 - 1270	138 - 1530	110 - 1095	138 - 1530	140 - 1540	140 - 1540
		Display header information (0FA0)	167	--	90	--	114	114
		Display header information (0FA0)	167	--	90	--	114	114

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
51	RDSYSST	“Diagnostic Data DS 0”, partial list Display via logical address (00B1) local	406	286	233	286	300	360
		distributed First call	392	270	217	270	278	356
		distributed Intermediate call, REQ = 0	215	150	120	150	153	152
		distributed Last call	405	165	132	165	170	169
51	RDSYSST	”Diagnostic Data DS 1” partial list Display via graphical address (00B2) Display a 16-byte long DS 1	408	300	250	300	313	375
51	RDSYSST	”Diagnostic Data DS 1” partial list Display via logical address (00B3) Display a 16-byte long DS 1 local	447	324	268	324	340	402
		distributed First call	395	270	218	270	272	356
		distributed Intermediate call	218	150	120	150	153	153
		distributed Last call	257	178	142	178	182	182

DS = Data record

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
51	RDSYSST	”Diagnostic Data DP Slave” partial list Display via configured diagnostic address (00B4) First call	385	266	213	266	272	351
		Intermediate call, REQ = 0	--	--	115	--	149	148
		Last call (6 - 240 bytes)	246	170	135	170	174	173
52	WR_USMSG	Write user entry in diagnostic buffer write with message	186	128	102	128	75	100
		without message	107	75	60	75	74	98
54	RD_DPARAM	Read dynamic parameters local AI 8*13 bits	180	125	95	125	126	153
		distributed AI 8*12 bits (DS1 = 14 bytes)	200	135	105	135	121	121
55	WR_PARM	Write dynamic parameters local AI 8*13 bits	485	345	280	345	360	418
		distributed First call AI 8*12 bits (14 - 240 bytes)	370	260	210	260	268	347
		distributed Intermediate/last call, REQ = 0	175	115	90	115	122	122

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
56	WR_DPARM	Write predefined dynamic parameters AI 8*13 bits local	445	336	280	336	353	411
		distributed First call AI 8*12 bits (2 - 240 bytes)	300	205	165	205	217	296
		Intermediate/last call	145	100	80	100	106	106
57	PARM_MOD	Assign module parameters local Module/DS number/DS lengths in bytes AI 8*13 bits	770	580	490	580	609	695
		distributed AO 8*12 bits First call (16 - 240 bytes)	300	205	165	205	215	295
		distributed Intermediate/last call	145	100	80	100	104	104

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
58	WR_REC	Write parameter data record local (n = number of bytes)	390 + n* 2.87	267 + n* 2.71	217 + n* 2.52	267 + n* 2.71	282 + n* 2.68	311 + n* 2.71
		First call, integrated DP interface module (n = number of bytes)	334 + n* 0.42	228 + n* 0.35	182 + n* 0.30	228 + n* 0.35	222 + n* 0.39	276 + n* 0.32
		Intermediate call, REQ = 0 integrated DP interface module	138	90	70	90	94	94
		Last call, integrated DP interface module	138	90	70	90	95	94
		First call, external DP interface module (n = number of bytes)	332 + n* 0.32	215 + n* 0.26	171 + n* 0.23	215 + n* 0.26	208 + n* 0.26	208 + n* 0.29
		Intermediate call, REQ = 0 external DP interface module	139	90	72	90	95	94
		Last call, external DP interface module	140	91	72	91	95	95

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
59	RD_REC	Read data record local (n = number of bytes)	390 + n* 3.13	267 + n* 2.90	218 + n* 2.71	267 + n* 2.90	282 + n* 2.97	342 + n* 3.13
		First call, integrated DP interface module	322	217	172	217	212	264
		Intermediate call, REQ = 0 integrated DP interface module	138	90	70	90	95	94
		Last call, integrated DP interface module (n = number of bytes)	198 + n* 0.35	132 + n* 0.33	106 + n* 0.27	132 + n* 0.33	138 + n* 0.33	138 + n* 0.33
		First call, external DP interface module	304	204	163	204	198	197
		Intermediate call, REQ = 0 external DP interface module	139	91	72	91	95	94
		Last call, external DP interface module (n = number of bytes)	200 + n* 0.33	132 + n* 0.2	105 + n* 0.2	132 + n* 0.2	136 + n* 0.33	136 + n* 0.27
60	GD_SND	Send GD packet 1 byte	295	215	175	215	--	--
		32 bytes	910	640	515	640	--	--
61	GD_RCV	Receive GD package (1 - 32 Byte)	145	105	85	105	--	--
62	CONTROL	Check status of the connection belonging to a local communication-SFB-instance	116	87	69	87	107	136

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
64	TIME_TCK	Display millisecond timer	24	19	15	19	19	47
65	X_SEND	Transmit data to external partner First call, establish a connection (1 - 76 bytes) REQ = 1	860 - 910	710 - 740	765 - 795	710 - 740	--	--
		First call, connection present (1-76 bytes)	590 - 635	400 - 430	320 - 345	400 - 430	--	--
		Intermediate call (1-76 bytes)	180	130	100	130	--	--
		Last call, BUSY = 0	285	195	155	195	--	--
66	X_RCV	Receive data from external partner Test reception (1-76 bytes)	92	65	55	65	--	--
		Read data (1-76 bytes)	275 - 315	190 - 220	150 - 175	190 - 220	--	--

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
67	X_GET	Read data from external partner First call, establish a connection (1-76 bytes) REQ = 1	760	645	715	645	--	--
		First call, connection present (1-76 bytes)	490	335	265	335	--	--
		Intermediate call (1-76 bytes)	195	135	110	135	--	--
		Last call BUSY = 0	450 - 490	310 - 340	245 - 270	310 - 340	--	--
68	X_PUT	Write data to external partner First call, establish a connection (1-76 bytes) REQ = 1	880 - 925	725 - 755	780 - 810	725 - 755	--	--
		First call, connection present (1-76 bytes)	610 - 655	415 - 445	330 - 360	415 - 445	--	--
		Intermediate call (1-76 bytes)	195	135	110	135	--	--
		Last call, BUSY = 0	300	205	162	205	--	--

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
69	X_ABORT	Abort connection to external partner First call, REQ = 1	220	160	125	160	--	--
		Intermediate call	125	90	70	90	--	--
		Last call, BUSY = 0	365	375	75 - 500	375	--	--
72	I_GET	Read data from internal partner First call, establish a connection (1-76 bytes) REQ = 1	815	680	745	680	--	--
		First call, connection present (1-76 bytes)	505	345	275	345	--	--
		Intermediate call (1-76 bytes)	205	145	115	145	--	--
		Last call, BUSY = 0	460 - 505	315 - 345	250 - 275	315 - 345	--	--
73	I_PUT	Write data to internal partner First call, establish a connection (1-76 bytes) REQ = 1	690 - 980	430 - 800	340 - 840	430 - 800	--	--
		First call, connection present (1-76 bytes)	625 - 665	425 - 455	340 - 365	425 - 455	--	--
		Intermediate call (1-76 bytes)	205	145	115	145	--	--
		Last call, BUSY = 0	310	215	170	215	--	--

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
74	I_ABORT	Abort connection to internal partner First call, REQ = 1	225	160	125	160	--	--
		Intermediate call	125	90	75	90	--	--
		Last call, without / with connection BUSY = 0	365	380	70 / 503	380	--	--
79	SET [1)]	Set bit array in I/O area n = number of bits to set at 1	43 + n * 0.39	28 + n * 0.32	23 + n * 0.26	28 + n * 0.32	53 + n * 1.35	80 + n * 1.32
80	RSET [1)]	Delete bit array in I/O area n = number of bits to set at 0	43 + n * 0.39	28 + n * 0.32	23 + n * 0.26	28 + n * 0.32	53 + n * 1.35	80 + n * 1.32
81	UBLKMOV	Copy variable without interruption n = number of bytes to copy	62 + n* 0.30	44 + n* 0.17	33 + n* 0.17	44 + n* 0.17	43 + n* 0.17	42 + n* 0.17
87	C_DIAG	Determine current connection status MODE = 0	42	28	22	27	36	36
		Mode = 1, 2, 3	187	189	276	346	249	249
90	H_CTRL	Influence processes involving fault-tolerant systems	--	--	--	--	19 - 21	19 - 21
100	SET_CLKS	Set time-of-day and clock status MODE = 1	370	263	227	263	439	1169
		MODE = 2	125	84	67	84	192	403
		MODE = 3	375	266	232	266	442	1167

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
103	DP_TOPOL	Detemine bus topology in a DP master system first call, REQ = 1	325	213	170	213	227	345
		Intermediate call	63	45	36	45	51	51
		Last call BUSY = 0	66	47	38	47	53	56
104	CIR	Controls the CiR procedure MODE = 0, information	23	18	14	18	19	–
		MODE = 1, Enable CiR procedure	24	18	15	18	19	–
		MODE = 2, Disable CiR procefure entirely	24	18	15	18	19	–
		MODE = 3, Disable CiR procedure partially	26	19	16	19	20	–

1) Measured with I/O modules of the type “Binary Simulator C79459-A1002-A1, Release 1” in the central rack

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
105	READ_SI	Read dynamically assigned system resources MODE = 0	176 - 1807 0)	117 - 3574 0)	94 - 2859 0)	117 - 3574 0)	117 - 3205 0)	117 - 3206 0)
		MODE = 1	204 - 2098 1)	138 - 4128 1)	110 - 3302 1)	138 - 4128 1)	136 - 3802 1)	303 - 3971 1)
		MODE = 2	205 - 1478 1)	140 - 2868 1)	111 - 2294 1)	140 - 2868 1)	137 - 2901 1)	304 - 3069 1)
		MODE = 3	206 - 2152 2)	140 - 4129 2)	111 - 3303 2)	140 - 4129 2)	137 - 3802 2)	305 - 3970 2)
106	DEL_SI	Enable dynamically assigned system resources MODE = 1	176 - 1289 1)	125 - 2666 1)	97 - 2131 1)	125 - 2666 1)	145 - 6954 1)	507 - 23875 1)
		MODE = 2	180 - 1272 1)	127 - 2580 1)	99 - 2061 1)	127 - 2580 1)	147 - 2668 1)	510 - 3033 1)
		MODE = 3	177 - 1350 2)	125 - 2705 2)	98 - 2162 2)	125 - 2705 2)	145 - 6974 2)	507 - 23906 2)

0) Depending on the size of the SYS_INST target area and on the number of the system resources to be read

1) Depending on the number of active messages (assigned system resources)

2) Depending on the number of active messages (assigned system resources) and on the number of assigned instances with the desired CMP_ID.

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
107	ALARM_DQ	Acknowledgeable block-related messages create first call, SIG = 0 -> 1	497	336	267	336	349	566
		Call (without message)	145	98	78	98	101	157
108	ALARM_D	Not acknowledgeable block-related messages create first call, SIG = 0 -> 1	499	337	266	337	350	548
		Call (without message)	146	98	78	98	101	156
126	SYNC_PI	Update the process image partition of the inputs in a synchronous cycle						
127	SYNC_PO	Update the process image partition of the outputs in a synchronous cycle						

System Function Blocks

The following table lists the system function blocks provided with the operating system of the S7-400 CPUs as well as the execution times of the individual CPUs (X: function exists, execution times were not available when manual was printed).

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
0	CTU	Count up	26	16	13	16	17	16
1	CTD	Count down	25	17	13	17	17	17
2	CTUD	Count up and down	29	19	15	19	19	19
3	TP	Generate pulse	34	23	18	23	24	52
4	TON	Generate on-delay	34	23	18	23	24	52
5	TOF	Generate off-delay	36	24	19	24	20	53
8	USEND	Send data without coordination (one send parameter supplied) JOB activated (1 - 440 bytes)	473 - 737	318 - 509	253 - 407	317 - 509	330 - 436	425 - 542
		JOB checked	159	107	86	108	115	145
		JOB finished (DONE = 1)	152	103	82	104	107	137

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
9	URCV	Receive data without coordination (one receive parameter supplied) JOB activated	137	93	74	94	100	130
		JOB checked	137	93	74	94	100	130
		JOB finished (NDR = 1; 1 - 440 bytes)	345 - 610	232 - 421	186 - 337	233 - 421	243 - 363	314 - 435
12	BSEND	Send data block by block JOB activated (1 - 3000 bytes)	386	258	207	258	264	323
		JOB checked	171	115	92	116	122	152
		JOB finished (DONE = 1)	165	110	88	111	115	145
13	BRCV	Receive data block by block JOB activated (1 - 3000 bytes)	203	138	110	139	145	175
		JOB checked	161	110	88	111	117	147
		JOB finished	162	109	87	110	113	143

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
14	GET	Read data from remote CPU (one area specified) JOB activated	336	227	183	228	227	297
		JOB checked	161	109	87	110	116	146
		JOB finished (NDR = 1; 1 - 450 bytes)	344 - 626	231 - 431	185 - 345	232 - 432	243 - 369	314 - 441
15	PUT	Write data to remote CPU JOB activated (1 - 404 bytes)	498 - 748	337 - 513	269 - 410	337 - 515	349 - 458	443 - 552
		JOB checked	161	108	87	109	116	146
		JOB finished (DONE = 1)	154	104	83	105	108	138
16	PRINT	Send data to a printer JOB activated, REQ = 1	513 - 757	338 - 516	271 - 414	339 - 518	354 - 462	449 - 545
		JOB checked	160	107	86	108	115	145
		JOB finished, DONE = 1	153	103	82	104	107	137

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
19	START	Start remote device JOB activated, REQ = 1	497	333	265	333	339	408
		JOB checked	169	114	91	115	121	151
		JOB finished, DONE = 1	164	110	88	111	115	146
20	STOP	Stop remote device JOB activated, REQ = 1	472	314	251	314	322	384
		JOB checked	169	114	91	115	121	151
		JOB finished, DONE = 1	164	110	88	111	115	146
21	RESUME	Restart remote device JOB activated, REQ = 1	496	334	265	332	339	399
		JOB checked	169	114	91	115	121	151
		JOB finished, DONE = 1	164	110	88	111	115	145
22	STATUS	Query status of remote partner JOB activated, REQ = 1	268	183	146	184	188	258
		JOB checked	161	108	87	109	116	146
		JOB finished, NDR = 1	604	404	323	404	415	486

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
23	USTATUS	Receive status of remote device without coordination JOB activated, NDR = 1	137	93	74	94	100	131
		JOB checked	137	93	74	94	100	130
		JOB finished	604	404	323	404	415	486
31	NOTIFY_8P	Generate block–related message without acknowledgment JOB activated, SIG = 0–> 1 (1 - 420 bytes)	561 - 823	373 - 580	301 - 462	373 - 580	384 - 510	519 - 644
		JOB checked	186	125	100	125	133	163
		JOB finished, DONE = 1	191	128	102	128	130	160
32	DRUM	Implement sequencer	52	33	26	33	35	62
33	ALARM	Generate block-related message with acknowledgment JOB activated, SIG = 0–> 1 (1 - 420 bytes)	581 - 843	386 - 587	307 - 470	385 - 589	392 - 518	527 - 652
		JOB checked	205	136	109	137	141	171
		JOB finished, DONE = 1	207	137	110	138	136	166

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
34	ALARM_8	Generate block-related message without accompanying values for 8 signals JOB activated, SIG = 0–> 1 (1 - 420 bytes)	416	278	222	279	278	372
		JOB checked	203	135	108	136	140	170
		JOB finished, DONE = 1	206	137	109	138	135	166
35	ALARM_8P	Generate block-related message with accompanying values for 8 signals JOB activated, SIG = 0–> 1 (1 - 420 bytes)	580 - 842	384 - 587	308 - 469	385 - 597	392 - 517	526 - 651
		JOB checked	204	136	108	137	140	170
		JOB finished, DONE = 1	207	137	110	138	135	166

SFC No.	SFC Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
36	NOTIFY	Generate block-related message without acknowledgment JOB activated, SIG = 0–> 1 (1 - 420 bytes)	561 - 823	373 - 580	301 - 462	379 - 578	384 - 510	519 - 644
		JOB checked	186	125	100	126	133	163
		JOB finished, DONE = 1	191	128	102	129	130	160
37	AR_SEND	Send archive data JOB activated, REQ = 1 (1 - 3000 bytes)	388	258	208	258	265	328
		JOB checked	173	116	92	116	123	155
		JOB finished, DONE = 1	167	111	88	112	115	147

SFB No.	SFB Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
52	RDREC	Read data record from a DP slave via integrated DP interface, First call (2-16 bytes)	341	221	177	221	228	269
		Intermediate call	173	111	89	111	117	114
		Last call	236	157	127	157	164	161
52	RDREC	Read data record from a DP slave via external DP interface, First call (4-16 bytes)	323	211	170	211	213	210
		Intermediate call	174	112	90	112	117	114
		Last call	238	154	124	154	161	158

SFB No.	SFB Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redun-dant)
53	WRREC	Write data record in a DP slalve via integrated DP interface, First call (1-10 bytes)	354	234	187	234	241	281
		Intermediate call	170	110	88	110	116	112
		Last call	171	110	89	110	116	113
53	WRREC	Write data record in a DP slave via external DP interface, First call (2-14 bytes)	339	224	180	224	226	223
		Intermediate call	170	110	89	110	116	113
		Last call	172	111	89	111	117	113
54	RALRM	Receive interrupt from a DP slave Runtime measurement for non-I/O-dependent OBs, MODE = 1, OB 1	133	81	70	81	83	83
54	RALRM	Receive interrupt from a DP slave Runtime measurement at integrated DP interface, MODE = 1, OB 40, OB 83, OB 86	250	164	135	164	245	245
		OB 55 to OB 57, OB 82	257	171	140	171	251	251
		OB 70	--	--	--	--	242	242

SFB No.	SFB Name	Function	Execution Time in µs					
			CPU 412	CPU 414	CPU 416	CPU 417	CPU 414-4H 417-4H (solo)	CPU 414-4H 417-4H (redundant)
54	RALRM	Receive interrupt from a DP slave Runtime measurement at external DP interface, MODE = 1, OB 40, OB 83, OB 86	429	290	234	290	458	458
		OB 55 to OB 57, OB 82	704	499	413	499	747	747
		OB 70	--	--	--	--	460	460
54	RALRM	Receive interrupt from a DP slave Runtime measurement at central I/O, MODE = 1, OB 40, OB 82, OB 83, OB 86	215	138	111	138	143	143
		OB 55 to OB 57	619	472	414	472	567	567

Sublist of the System Status List (SSL)

SSL-ID	Information Functions
	Module Identification
0111	One ident. data record only
	CPU Characteristics
0012	CPU features, all features
0112	Features of a group
	MC7 processing unit
	Time system
	System behavior
	MC7 language description
	Availability of SFCs
	User Memory Area
0F12	Only partial list header information
	User Memory Area
0113	Data record for specified memory area
	Work memory
0F13	Only partial list header information

Sublist of the System Status List (SSL), continued

SSL-ID	Information Functions
	System Areas
0014	System areas, all system areas
0F14	Only partial list header information
	Block Types
0015	Block types, data records for all block types
	Status Module LEDs
0019	Status of all module LEDs
0F19	Only partial list header information
	Component Identification
001C	Identification of all components
011C	Identification of one component
0F1C	Only SSL partial list header information
	Interrupt Status
0222	Data record for specified interrupt
	Free cycle
	Clock interrupt

Sublist of the System Status List (SSL), continued

SSL-ID	Information Functions
	Time-delay interrupt
	Timed interrupt
	Process interrupt
	DP interrupt
	Multicomputing or synchronous cycle interrupt
	Redundancy interrupt
	Asynchronous error interrupts
	Background
	Startup
	Synchronous error interrupts
	Assignment between process image partitions and OBs
0025	Assignment betweeen all process image partitions and OBs within the CPU
0125	Assignment between a process image partition and the corresponding OB
0225	Assignment between an OB and the corresponding process image partitions
0F25	Only partial list header information

Sublist of the System Status List (SSL), continued

SSL-ID	Information Functions
	Communication Status Data
0132	Status data for a communication unit
	Diagnostics
	Time system
0232	Status data for a communication unit
	CPU protection level and operator switch settings
	H CPU Group Information
0071	Information on the current status of the H system
0F71	Only partial list header information
	Status of the Module LEDs
0174	Status of one LED
	Switched DP Slaves in the H System
0C75	Communication status between the H system and a switched DP slave

Sublist of the System Status List (SSL), continued

SSL-ID	Information Functions
	DP Master System Information
0090	Information about all the DP master systems known to the CPU
0190	Information about a DP master system
0F90	Only SSL partial list header information
	Module Status Information (A maximum of 27 data records are supplied)
0091	Module status information of all inserted modules/submodules
0191	Status information of all modules/racks with incorrect type IDs.
0291	Module status information of all faulty modules
0391	Module status information of all unavailable modules
0591	Module status information of all submodules of the host module
0991	Status information of all submodules in the host module in the rack
0C91	Status information of a module in the central rack or connected to an integrated DP interface module via the logical base address
4C91	Status information of a module connected to an external DP interface module via the logical base address
0D91	Status information of all modules in the specified rack
0E91	Status information of all assigned modules

Sublist of the System Status List (SSL), continued

SSL-ID	Information Functions
	Rack/Station Status Information
0092	Expected status of the central racks/stations of a DP master system
4092	Expected status of the stations of a DP master system which is connected via an external DP interface module
0192	Activation status of the stations of a DP master system which is connected via an external DP interface module
0292	Actual status of the central racks/stations of a DP master system
0392	Status of the back–up battery of a CPU rack if at least one battery fails
0492	Status of the entire back–up batteries of all racks of the a CPU
0592	Actual status of the racks in the central configuration/stations of DP master system which is connected via an external DP interface module.
4292	Actual status of the stations of a DP master system which is connected via an external DP interface module
0692	OK status of the expansion units in the central configuration/stations of a DP master system which is connected via an integrated DP interface module.
4692	OK status of the stations of a DP master system which is connected via an external DP interface module.
	Addtional DP Master System Information
0195	Additional information on a DP master system
0F95	Only partial list header information

Sublist of the System Status List (SSL), continued

SSL-ID	Information Functions
	Diagnostic Buffer (A maximum of 21 data records are supplied)
00A0	All current diagnostic entries available in current operating mode
01A0	Last x entries. X is listed in index
0FA0	Only partial list header information
	Module Diagnostic Data
00B1	First four diagnostic bytes of a module (DS0)
00B2	All diagnostic data of a module (≤ 220 bytes, DS1) (no DP module)
00B3	All diagnostic data of a module (≤ 220 bytes, DS1)
00B4	Diagnostic data of a DP slave with logical base address

Alphabetical Index of Instructions