

# Data Blocks



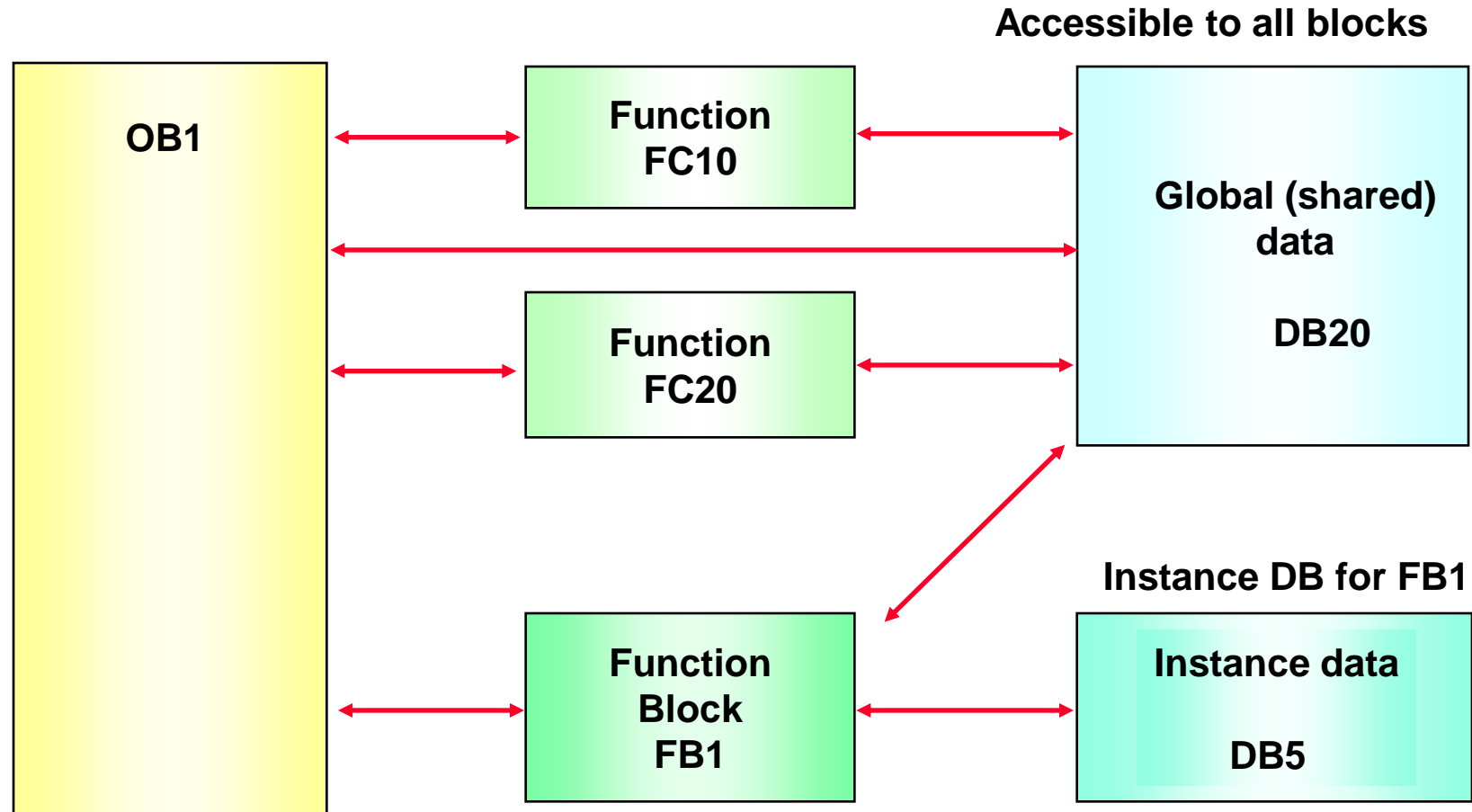
## Objectives

**Upon completion of this chapter the participant will ...**

- ... understand the purpose of global data blocks
- ... be familiar with elementary and complex data types
- ... be able to monitor a data block
- ... be familiar with the possibilities for addressing data block variables



# Data Blocks (DBs)



## Overview of Data Types in STEP 7

Elementary  
data types  
(up to 32 bits)

- Bit data types (BOOL, BYTE, WORD, DWORD, CHAR)
- Mathematical data types (INT, DINT, REAL)
- Time types (S5TIME, TIME, DATE, TIME\_OF\_DAY)

Complex  
data types  
(longer than 32 bits)

- Time type (DATE\_AND\_TIME)
- Array (ARRAY)
- Structure (STRUCT)
- Character chain (STRING)

User-defined data types  
(longer than 32 bits)

UDT data type (**U**ser **D**efined  
**D**ata **T**ype)

## Elementary Data Types in STEP 7

Keyword	Length (in bits)	Constants	Variables
BOOL	1		1 or 0 I 1.0
BYTE	8		B#16#A9
WORD	16	MB70	W#16#12AF
DWORD	32	MW72	DW#16#ADAC1EF5
CHAR	8		'w' DBB4
S5TIME	16		S5T#5s_200ms
INT	16	MW30	123 #Value
DINT	32		L#65539
REAL	32	MD80	1.2 or 34.5E-12

## Complex Data Types

Keyword	Length (in bits)	Example	
<b>DATE_AND_TIME</b>	64	DT#01-08-24-12:14:55:234-1	
<b>STRING</b> (character string with max. 254 characters)	8 * (number of characters +2)	`This is a string` `SIEMENS`	
<b>ARRAY</b> (Group of elements of the same data type)	user-defined	Measured values: ARRAY[1..20] INT	
<b>STRUCT</b> (Group of elements of different data types)	user-defined	Motor: STRUCT Speed : INT Current : REAL END_STRUCT	
<b>UDT</b> (User Defined Data Type = "Template" consisting of elementary or complex	user-defined	UDT as block	UDT as array element
		STRUCT  Speed : INT  Current : REAL	Drive: ARRAY[1..4]

UDT1

END\_STRUCT

## Creating a Data Block

The screenshot shows the SIMATIC Manager interface. The 'Insert' menu is open, and 'S7 Block' is selected, which has opened a sub-menu with the following options:

- 1 Organization Block
- 2 Function Block
- 3 Function
- 4 Data Block
- 5 Data Type
- 6 Variable Table

The 'Properties - Data Block' dialog box is open, showing the following configuration:

- General - Part 1 | General - Part 2 | Calls | Attributes
- Name and type: DB99, Shared DB
- Symbolic Name: HMI\_Interface
- Symbol Comment: Transfer interface for HMI-System in general
- Created in Language: DB
- Project path: (empty)
- Storage location of project: D:\S7-Courses\My\_proje
- Date created: 27/04/2004 08:21:25
- Last modified: 27/04/2004 08:21:25
- Comment: (empty text area)

Buttons: OK, Cancel, Help

Inserts Data Block at the cursor position.

# Entering, Saving, Downloading and Monitoring a Data Block

The image shows two overlapping windows from the SIMATIC Manager software. The top window, titled 'LAD/STL/FBD - [DB1 -- My\_project\test station\CPU 315-2 DP]', displays the 'Declaration View' of a data block. The bottom window, with the same title, displays the 'Data View' of the same data block.

**Declaration View Table:**

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Motor_data	STRUCT		
+0.0	speed	INT	0	motor speed
+2.0	rated_current			
+6.0	started_current			
+10.0	direction			
=12.0				
+12.0	measuring_point			
+2.0				
=32.0				

**Data View Table:**

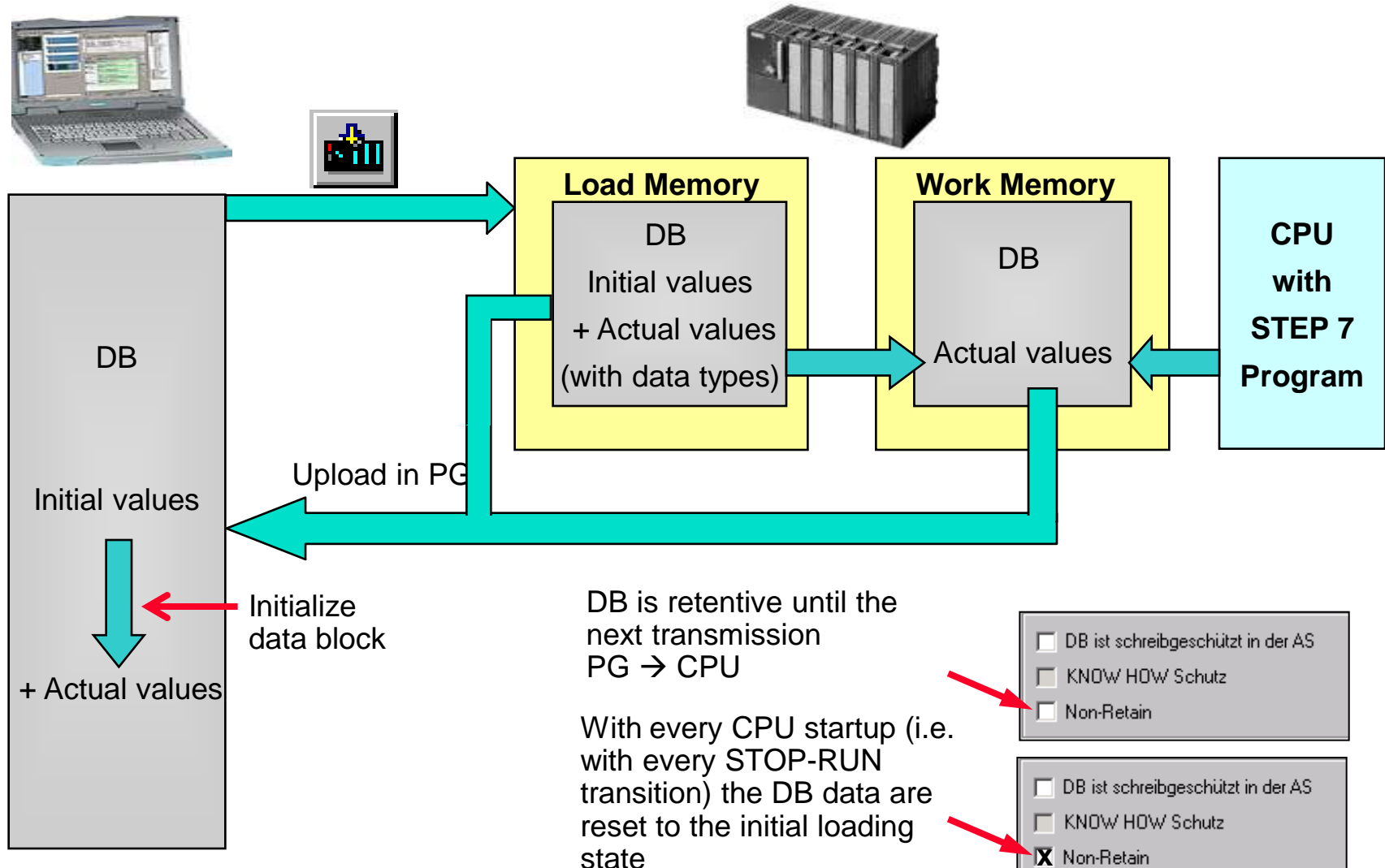
Address	Name	Type	Initial value	Actual value
0.0	Motor_data.speed	INT	0	0
2.0	Motor_data.rated_current	REAL	0.000000e+000	0.000000e+000
6.0	Motor_data.started_current	REAL	0.000000e+000	0.000000e+000
10.0	Motor_data.direction	BOOL	FALSE	FALSE
12.0	measuring_point[1]	INT	0	0
14.0	measuring_point[2]	INT	0	0
16.0	measuring_point[3]	INT	0	0
18.0	measuring_point[4]	INT	0	0
20.0	measuring_point[5]	INT	0	0
22.0	measuring_point[6]	INT	0	0
24.0	measuring_point[7]	INT	0	0
26.0	measuring_point[8]	INT	0	0
28.0	measuring_point[9]	INT	0	0
30.0	measuring_point[10]	INT	0	0

File/Block saved.

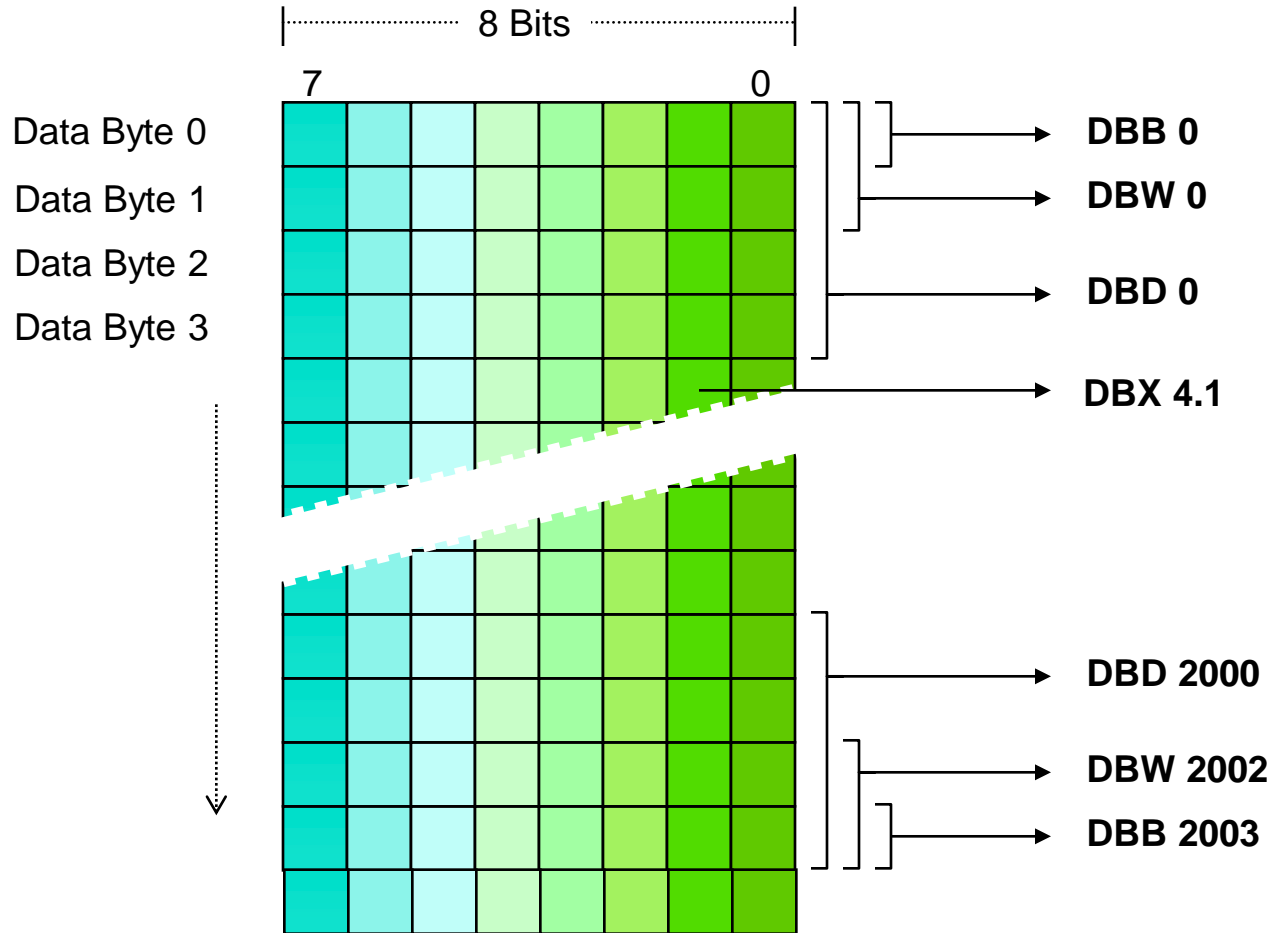
Press F1 to get Help. offline Sym as of 5. Insert



# Initial Value, Actual Value, Initialization, Retentivity

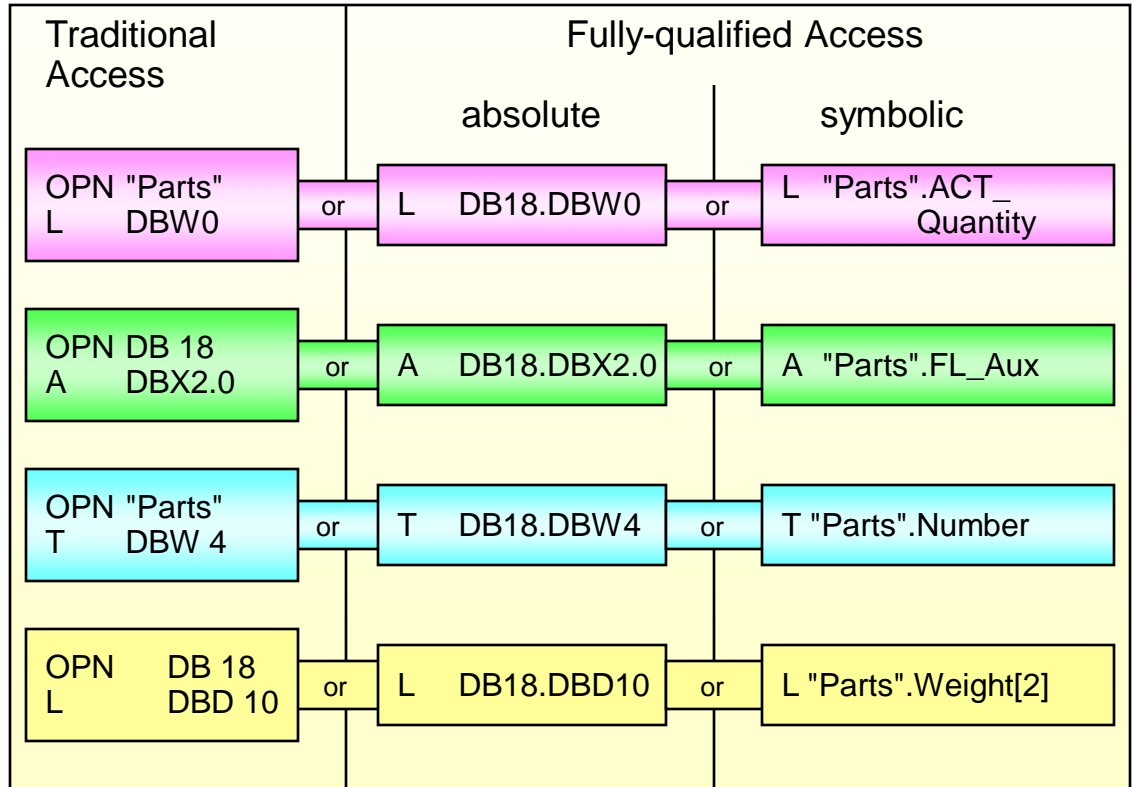


# Addressing Data Elements



# Accessing Data Elements

DB 18 "Parts"		
Addr.	Name	Type
0.0	Act_Quantity	INT
2.0	FL_Aux	BOOL
4.0	Number	INT
6.0	Weight[1]	REAL
10.0	Weight[2]	REAL



# Exercise 1: Replacing Bit Memory Variables with DB Variables

DB19 -- "DB\_PartsData" -- SERV2\_5\Ch03\_DB\...\DB19

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Act_quant	INT		
+2.0	Setp_quant	INT		
+4.0	aux_bit	BOOL		
=6.0	END_ST			

FC19 -- "FC\_Count\_Add" -- SERV2\_5\Ch03\_DB\...\FC19

FC19 : Count parts

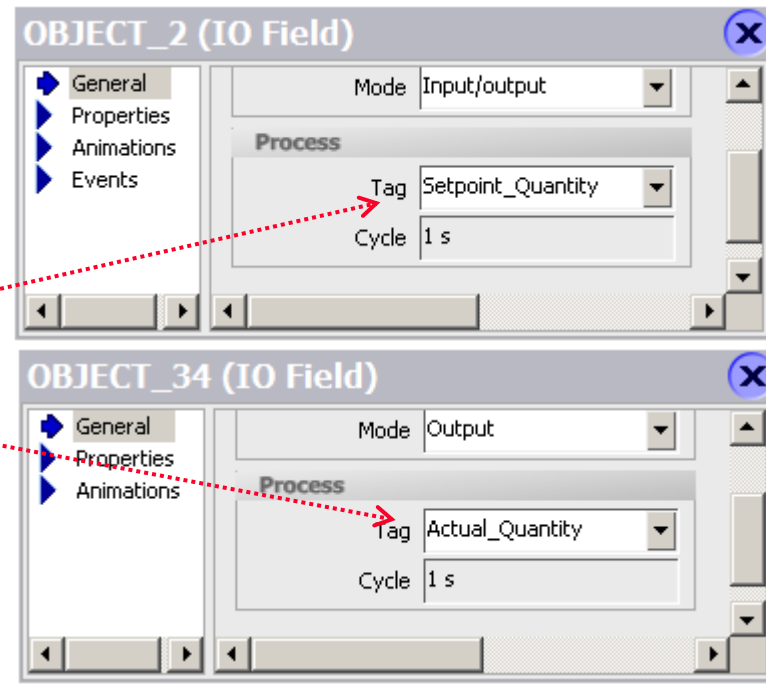
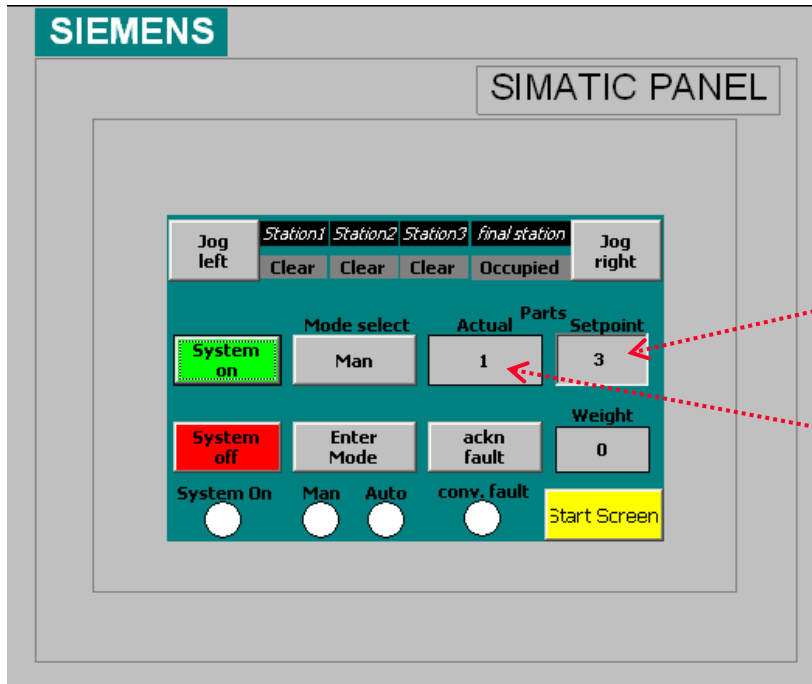
**Network 1:** Reset Counter to 0

**Symbol information:**

L_SYSTEM	Q4.1	-- Indicator light System ON
L_Bay-LB	Q8.4	-- Indicator light Light barrier bay
T_Bay-LB	I8.4	-- Momentary contact Light barrier bay
"DB_PartsData".Act_quant	DB19.DBW0	-- Quantity of transported parts

**Network 2:** Counting the Parts Transported while in AUTO Mode

# Exercise 2: Connecting WinCC flexible Tags to DB Variables



Name	Data type	Connection	Address	Comment	Symbol	Acquisition mode
Actual_Quantity	Int	2 Connection_1	DB 19 DBW 0	1 Quantity of transported parts	Act_quant	Cyclic on use
Setpoint_Quantity	Int	2 Connection_1	.. DB 19		Setp_quant	
MW_Messages	Word	2 Connection_1	.. MW 32		aux_bit	

Icon	Name	Info
	Act_quant	DB19.DBW0.0...
	Setp_quant	DB19.DBW2.0...
	aux_bit	DB19.DBX4.0...

Tags in WinCC flexible

Variables in STEP7

## If You Want to Know More



## Example of an ARRAY

### Measuring\_point

- 1. Measuring\_point, data type Real
- 2. Measuring\_point, data type Real
- 3. Measuring\_point, data type Real
- 
- 
- 
- 10. Measuring\_point, data type Real

Array with the name "Measuring\_point"  
(several elements of the same data type)

### Display in the Program Editor (Data Block DB 2):

The screenshot shows a window titled "DB2 -- My\_Project\My\_Station\CPU 314". It contains a table with the following data:

Address	Name	Type	Initial value	Comment
*0.0		STRUCT		
+0.0	Measuring_point	ARRAY[1..10]		
*4.0		REAL		
=40.0		END_STRUCT		

## Example of a STRUCTURE

### Motor\_data

Speed, data type Integer
Rated_current, data type Real
Starting_current, data type Real
Direction, data type Bool

Structure with the name "Motor\_data"  
(several elements  
with different data types)

### Display in the Program Editor (Data block DB 1):

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Motor_data	STRUCT		
+0.0	speed	INT	0	
+2.0	rated_current	REAL	0.000000e+000	
+6.0	started_current	REAL	0.000000e+000	
+10.0	direction	BOOL	FALSE	
=12.0		END_STRUCT		



## Defining the Address Priority (Symbolic/Absolute)

Properties - Block Folder Offline

General | Blocks | Checksums | Address priority:

	Behavior as in STEP7 < V5.2	Recommended for symbolic programming
Absolute value has priority	<input type="radio"/> Symbols are applied from the symbol table and the DB for all accesses (I,Q,M,T,C and DB)	<input type="radio"/> Exception: symbol accesses remain on the DB as they were programmed in the code block
Symbol has priority	<input type="radio"/> Exception: for accesses in structurally unchanged data types, the current symbols will be applied	<input checked="" type="radio"/> For all accesses (I,Q,M,T,C and DB)

OK Cancel Help