

# Functions and Function Blocks



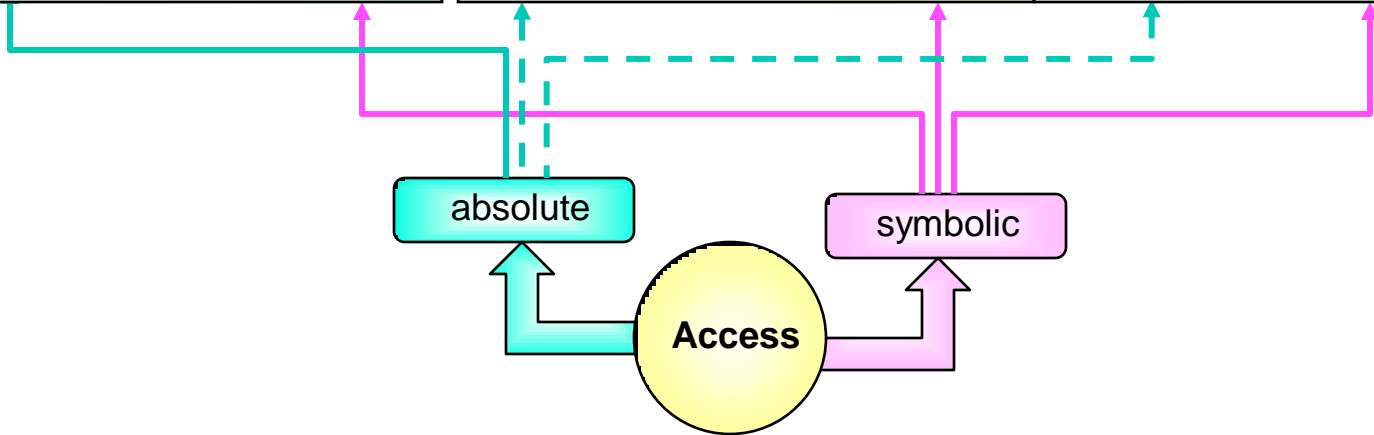
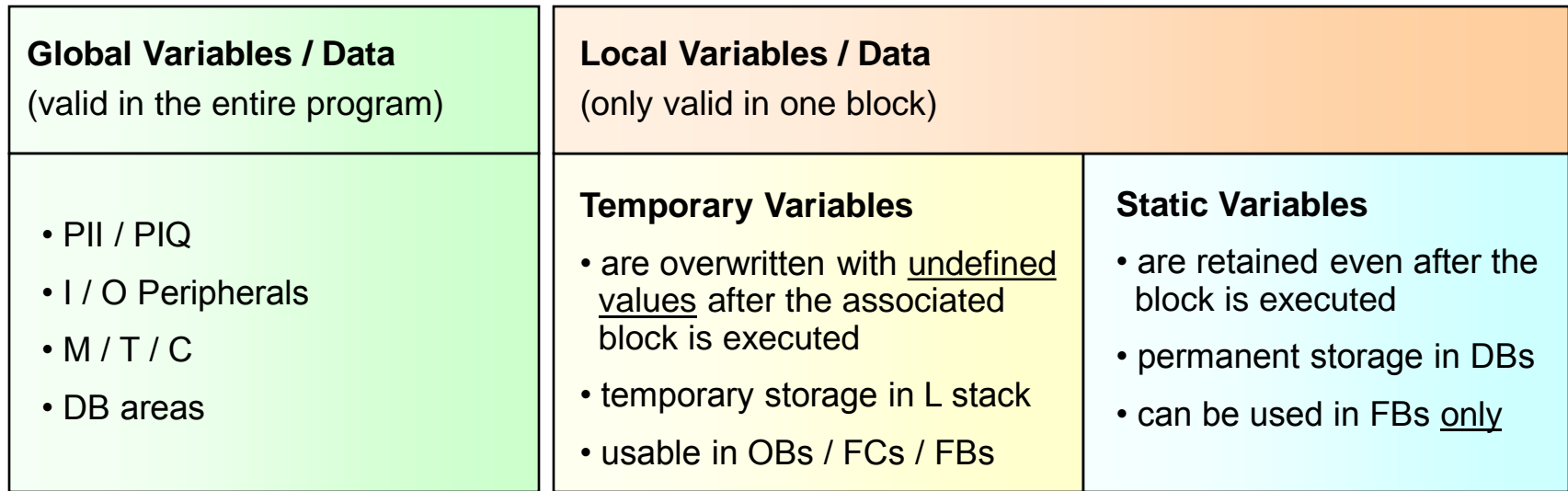
## Objectives

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

- ... be familiar with the purpose of temporary variables
- ... be able to program parameter-assignable functions and their calls
- ... know the difference between functions (FCs) and function blocks (FBs)
- ... be familiar with the purpose of static variables
- ... be able to declare static variables and apply them in the program
- ... be able to program parameter-assignable function blocks and call them



# Variables Overview



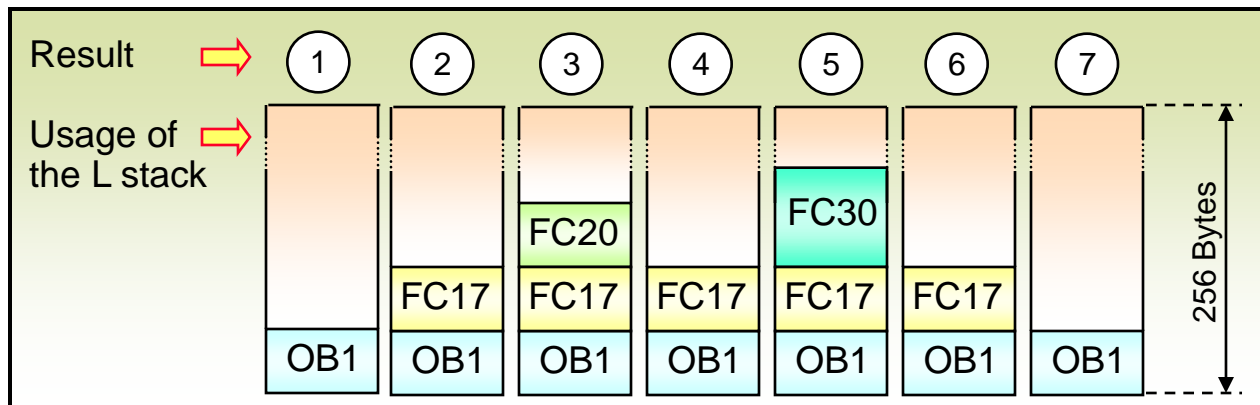
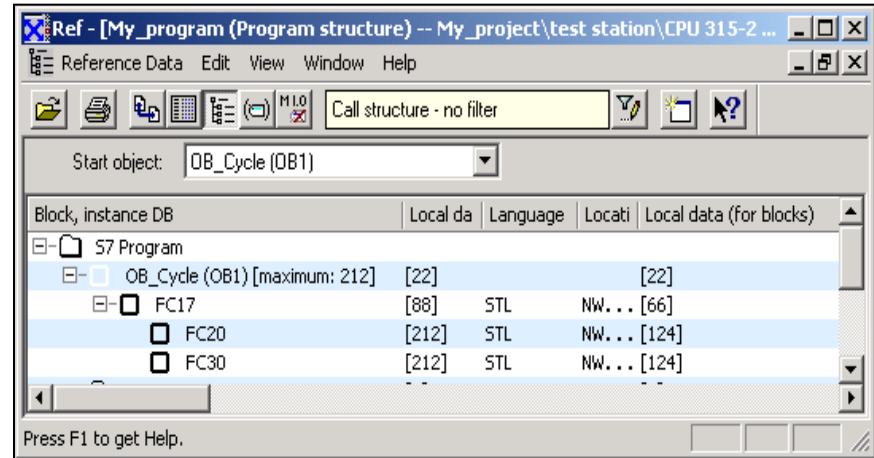
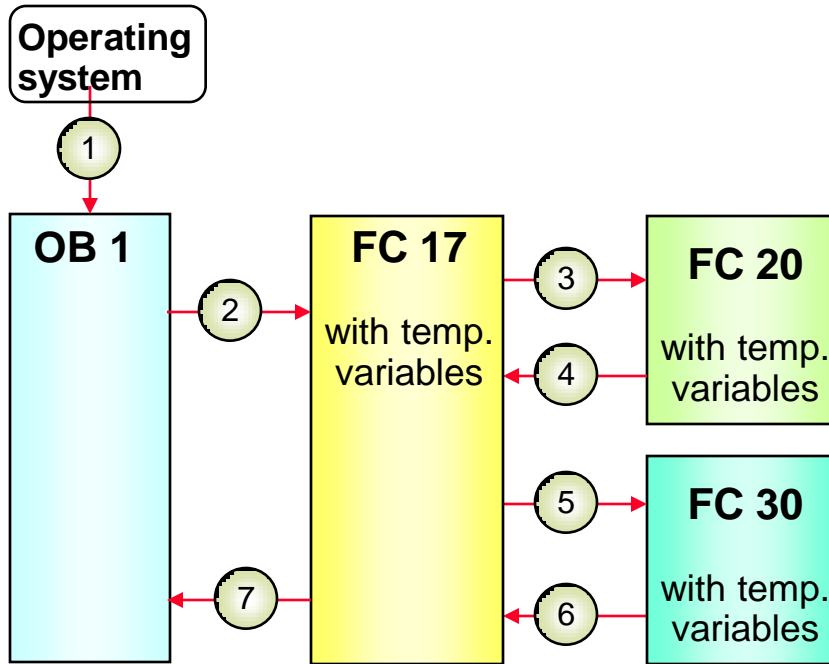
# Temporary Variables

The screenshot displays the SIMATIC Manager interface for a project named 'FC100 -- "FC\_LocalData\_Example" -- My\_Project\SIMATIC 300(1)\CPU 315-2 DP\...MFC100'. The main window shows the 'Contents Of: 'Environment\Interface\TEMP'' table, which lists the variable 'aux\_result' with data type 'Int' and address '0.0'. A red circle highlights the 'Name' column, and a yellow box labeled 'Declaration' points to the 'aux\_result' entry.

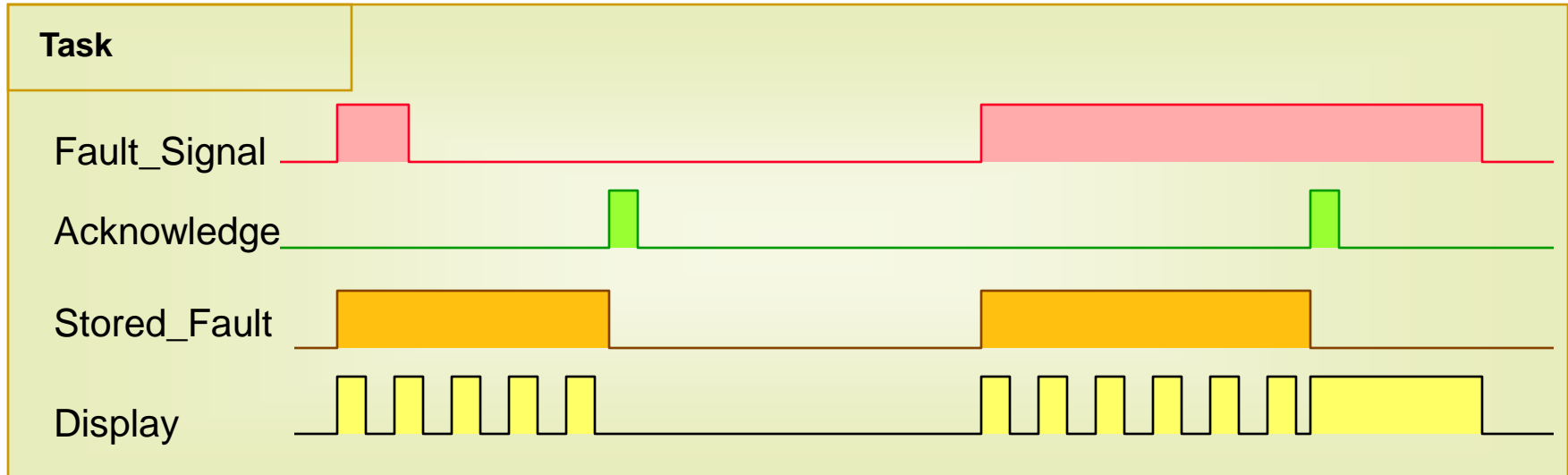
Name	Data Type	Address	Comment
aux_result	Int	0.0	

Below the table, the ladder logic network 'Network 1: Calculation' is shown. It consists of two function blocks: 'SUB\_I' and 'ADD\_I'. The 'SUB\_I' block has inputs 'MW100' (IN1) and 'MW102' (IN2), and its output 'OUT' is connected to the input 'IN1' of the 'ADD\_I' block. The 'ADD\_I' block also has an input 'MW104' (IN2), and its output 'OUT' is connected to 'MW106'. The output of the 'SUB\_I' block is labeled '#aux\_result'.

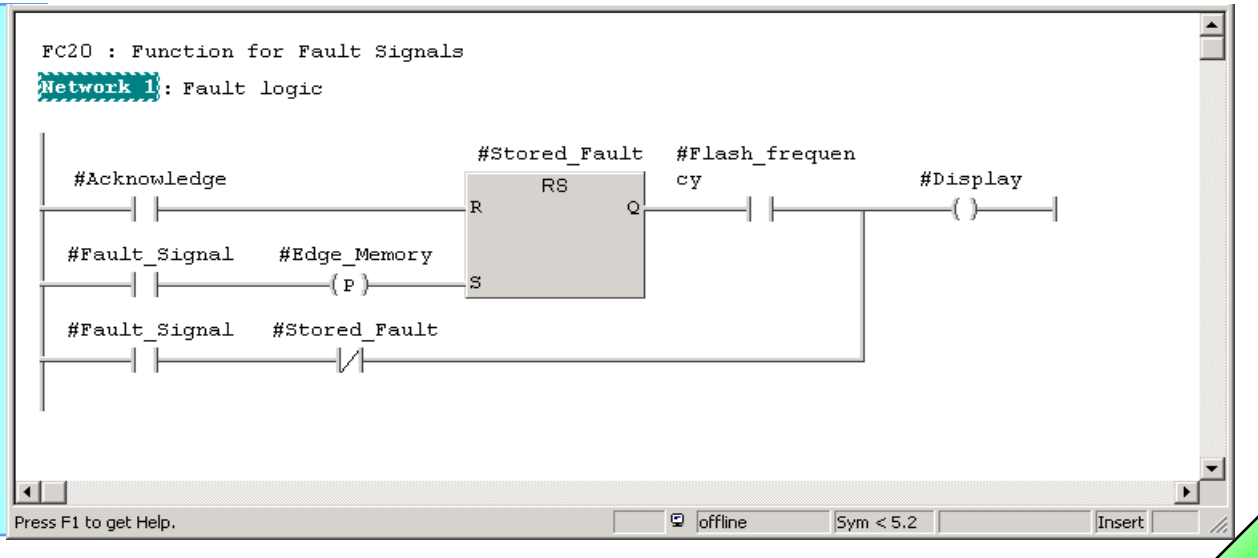
# Total Usage of the Local Data Stack



## Example for a Fault Display

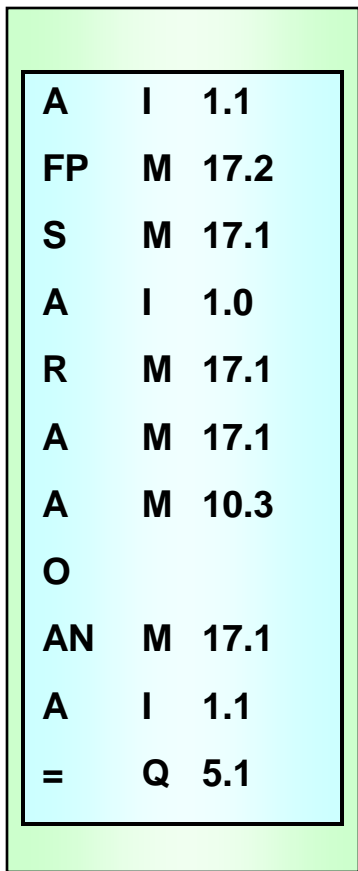


### Solution Suggestion



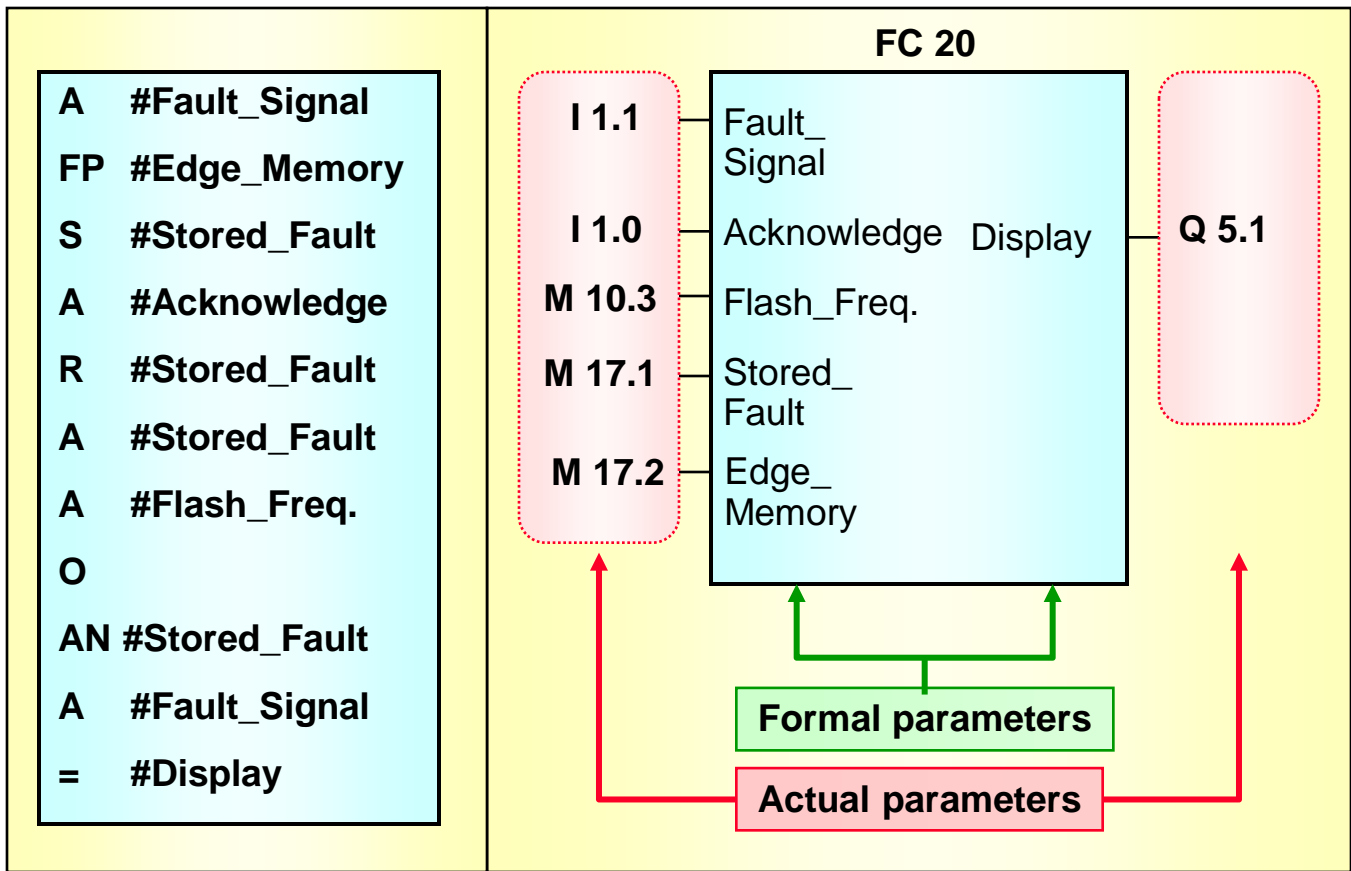
# Parameter-assignable Blocks

**Solution with non-parameter-assignable block**



**Solution with parameter-assignable block**

Program in FC 20	Call of FC 20 (e.g. in OB 1)
------------------	------------------------------



# Declaring Formal Parameters

Type of Parameter	Declaration	Use	Graphic Display
Input parameter	in	Read only	To the left of the block box
Output parameter	out	Write only	To the right of the block box
In/Out parameter	In_out	Read / Write	To the left of the block box

The screenshot shows the SIMATIC Manager interface with a tree view on the left and a parameter declaration table on the right.

**Tree View:**

- Interface
  - IN
    - Fault\_Signal
    - Acknowledge
    - Flash\_frequency
  - OUT
    - Display
  - IN\_OUT**
    - Stored\_Fault
    - Edge\_Memory
  - TEMP
  - RETURN

**Contents Of: 'Environment\Interface\IN\_OUT'**

Name	Data Type	Comment
Stored_Fault	Bool	
Edge_Memory	Bool	



# Editing a Parameter-assignable Block

The screenshot displays the SIMATIC Manager interface for editing a function block (FC20). The window title is "LAD/STL/FBD - [FC20 -- 'FC\_FaultEvaluation' -- SERV2\_S\Ch04\_FC-FB\...\FC20]".

**Interface Tree (Left):**

- Interface
  - IN
    - Fault\_Input
    - Acknowledge
    - Clock\_Memory
  - OUT
    - Indicator
  - IN\_OUT
    - M\_Bit\_fault\_ev
    - Aux\_bit\_edge\_detect
  - TEMP
  - RETURN

**Contents Of: 'Environment\Interface' (Right):**

Name
IN
OUT
IN_OUT
TEMP
RETURN

**Network 1: Fault Evaluation (Main Workspace):**

FC20 : Title:  
**Network 1:** Fault Evaluation

The ladder logic network shows a Set-Reset (SR) coil for the #M\_Bit\_fault\_ev bit. The Set (S) input is connected to a normally open contact for #Aux\_bit\_edge\_detect (with parameter p) and a normally open contact for #Fault\_Input. The Reset (R) input is connected to a normally open contact for #M\_Bit\_fault\_ev and a normally open contact for #Fault\_Input. The output (Q) of the SR coil is connected to a coil for #Indicator.

**Bottom Panel:**

Navigation buttons: 1: Error, 2: Info, 3: Cross-references, 4: Address info, 5: Modify, 6: Diagnostics, 7: Comparison

Status: Press F1 to get Help. | offline | Abs < 5.2 | Insert

# Calling a Parameter-assignable Block

The screenshot shows the SIMATIC Manager interface for a project named "My\_ProjectSIMATIC 300(1)CPU 315-2 DP...FC17". The main window displays "Network 2 : Evaluate Fault 1".

The left-hand pane shows a library of function blocks. Under "FC blocks", the block "FC20 FC\_FaultEvaluation" is circled in red. A yellow arrow points from this block to the main workspace with the text "Drag & Drop".

The main workspace shows the following function block call:

```

Network 2 : Evaluate Fault 1
-----
FC20
EN ----- ENO -----
" S_Fault1 "
Simulate Fault 1,
0=OFF/1=ON
I1.1 - Fault_Signal ----- Display --- ???
" T_Ackn_Fault1 "
Momentary contact
Fault
acknowledgement
I1.0 - Acknowledge
" M_Flash_2Hz "
Flash -
bit -
frequency 2 Hz
M10.3 - Flash_Freq
" M_Fault1 "
Memory bit for
storing Fault 1
M17.1 - Stored_Fault
" M_Aux_Fault1 "
Edge auxiliary
memory bit Fault 1
M17.2 - Edge_Memory
    
```

The status bar at the bottom indicates "offline", "Abs < 5.2", "Nw 2", and "Insert Chg".

# Exercise 1: Editing the Parameter-assignable Function FC 20

The screenshot displays the SIMATIC Manager LAD editor for function FC 20. The title bar indicates the project path: "LAD/STL/FBD - [FC20 -- 'FC\_FaultEvaluation' -- My\_Project\SIMATIC 300(1)\CPU 315-2 DP\...FC20]".

**Interface Parameters:**

Contents Of: 'Environment\Interface'	
	Name
IN	IN
OUT	OUT
IN_OUT	IN_OUT
TEMP	TEMP
RETURN	RETURN

**Network 1: Fault Evaluation**

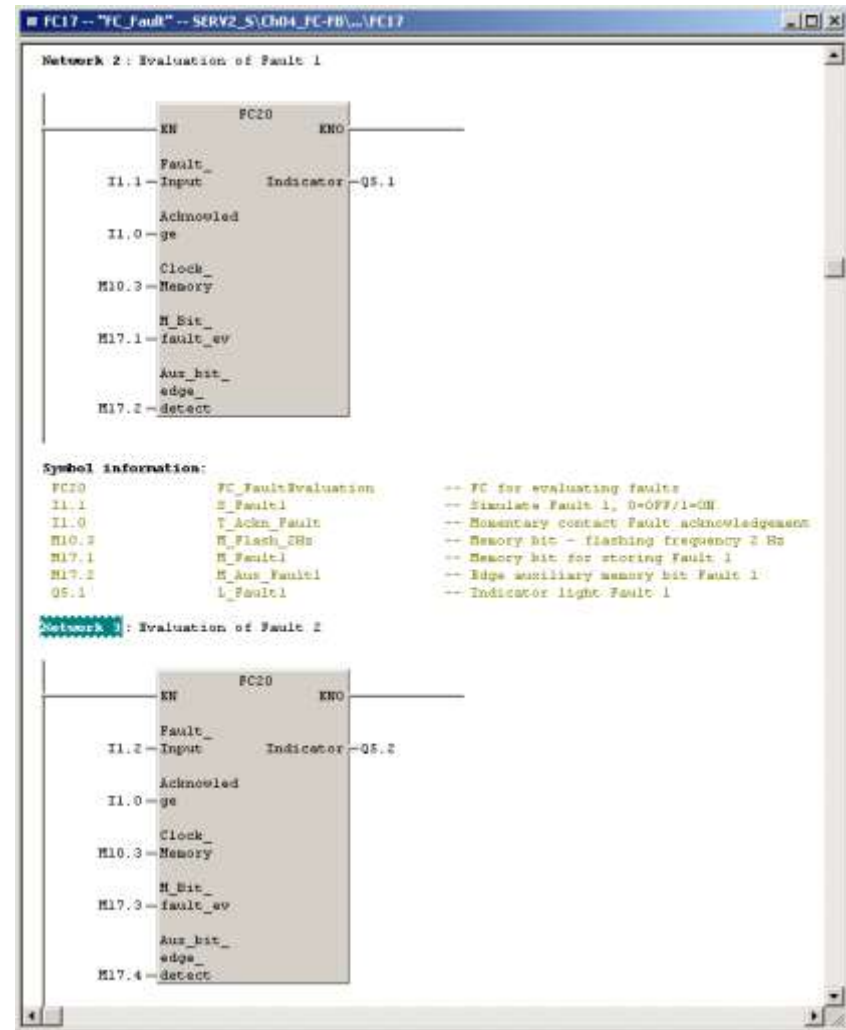
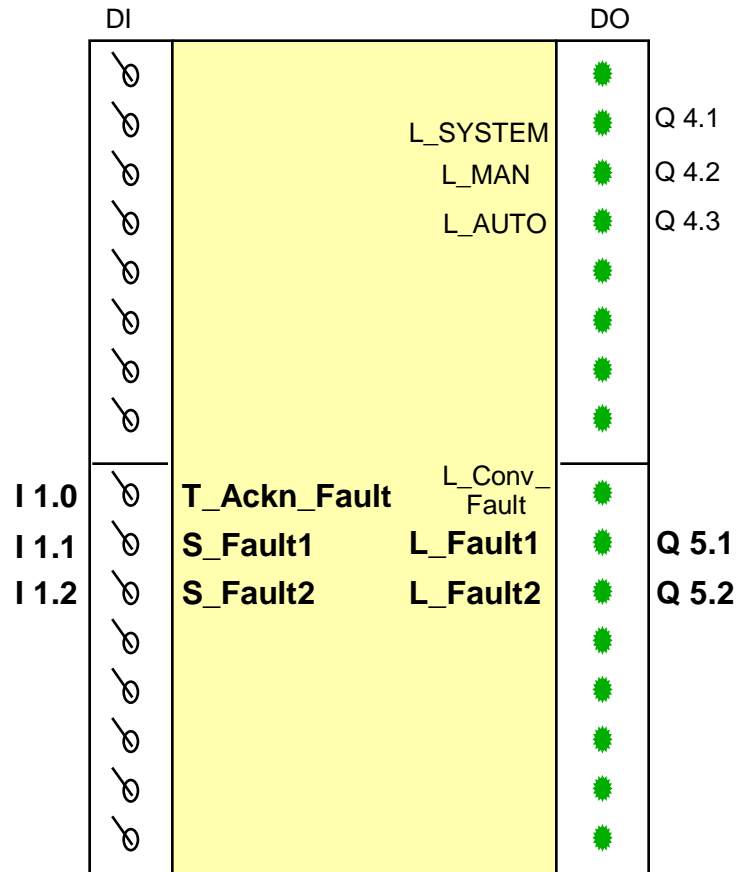
```

#Fault_Input (NO) --- #Aux_bit_edge_detect (P) --- S --- #M_Bit_fault_ev SR --- Q --- #Clock_Memory (NO) --- #Indicator (CO)
#Acknowledge (NO) --- R --- #M_Bit_fault_ev SR
#M_Bit_fault_ev (NC) --- #Fault_Input (NO) --- #Clock_Memory (NO) --- #Indicator (CO)
    
```

The ladder logic network shows a set-reset (SR) coil for the bit variable #M\_Bit\_fault\_ev. The set (S) input is triggered by #Fault\_Input (normally open contact) and #Aux\_bit\_edge\_detect (pulsed normally open contact). The reset (R) input is triggered by #Acknowledge (normally open contact). The output (Q) of the SR coil is connected to #Clock\_Memory (normally open contact) and #Indicator (coil output). A feedback loop is formed by #M\_Bit\_fault\_ev (normally closed contact) and #Fault\_Input (normally open contact) connected to the #Clock\_Memory contact.

The status bar at the bottom shows navigation options (1: Error, 2: Info, 3: Cross-references, 4: Address info, 5: Modify, 6: Diagnostics, 7: Comparison) and system information (offline, Abs < 5.2, Insert Chg).

# Exercise 2: Calling the Parameter-assignable Function FC 20

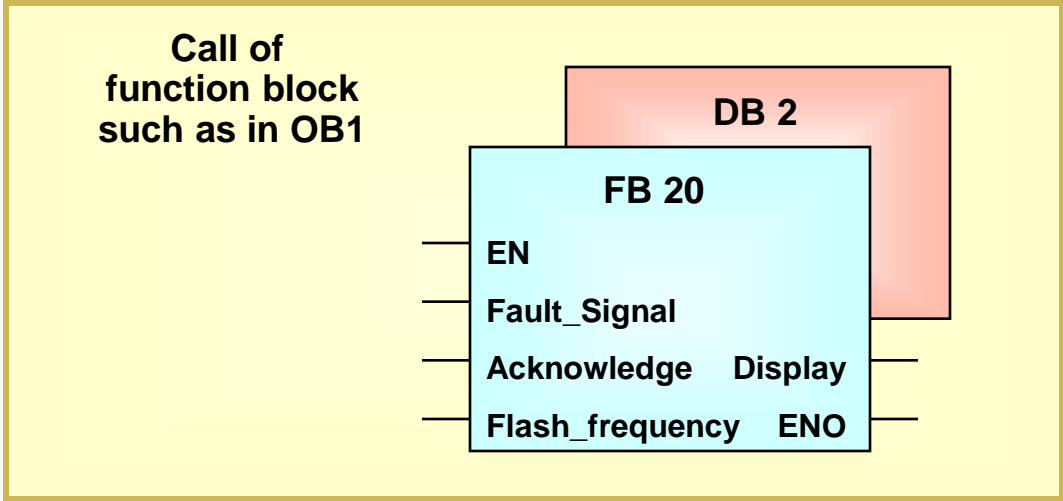


# Function Blocks (FB)

Contents Of: 'Environment\Interface\IN'

Name	Data Type	Address	Initial Value	Exclusion address	Termination address	Comment
Fault_Input	Bool	0.0	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
Acknowledge	Bool	0.1	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
Clock_Memory	Bool	0.2	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	

**Declaration Table  
of the  
Function Block**



# Function Block for Fault Display

Contents Of: 'Environment\Interface\IN'

Name	Data Type	Address	Initial Value	Exclusion address	Termination address	Comment
Fault_Input	Bool	0.0	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
Acknowledge	Bool	0.1	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
Clock_Memory	Bool	0.2	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	

**Declaration Table of the Function Block**

**Open Data Block**

By default, instance data blocks are opened with "Parameter assignment for data blocks" as of STEP 7 V5.2 (see Help). There is no functional difference for this DB to the editor in LAD/STL/FBD.

The DB could be opened with the DB editor in LAD/STL/FBD (see Help for STEP 7 V5.1).

Do you want to open the DB with "Parameter assignment for data blocks"?

Always display this message for Open Instance DB

Yes No

**Instance Data Block**

DB Param - [DB2 -- SERV2\_5\Ch04\_FC-FB]

Data block Edit PLC Debug View Window Help

Address	Declaration	Name	Type	Initial value	Actual value	Comment
1	0.0 in	Fault_Input	BOOL	FALSE	FALSE	
2	0.1 in	Acknowledge	BOOL	FALSE	FALSE	
3	0.2 in	Clock_Memory	BOOL	FALSE	FALSE	
4	2.0 out	Indicator	BOOL	FALSE	FALSE	
5	4.0 stat	M_Bit_fault_ev	BOOL	FALSE	FALSE	Memory bit for fault evaluation
6	4.1 stat	Aux_bit_edge_detect	BOOL	FALSE	FALSE	boolean auxiliary variable for edge detection

Messages  
Press F1 for help. offline

# Generating Instance Data Blocks

1. Generate instance DB with FB call

2. Create new instance DB

In the LAD / FBD / STL Editor

OB1 : "Main Program Sweep (Cycle)"

**Network 1** : Fault Indicator

```
call fb20, db20
```

In the SIMATIC Manager

**Properties - Data Block**

General - Part 1 | General - Part 2 | Calls | Attributes

Name and type: DB20 Instance DB FB20

Symbolic Name: Instance\_DB\_FB20

Symbol Comment: error evaluation

Created in Language: DB

Project path:

Storage location of project: D:\S7-Courses\My\_proje

	Code	Interface
Date created:	27/04/2004 10:12:08	
Last modified:	27/04/2004 10:12:08	27/04/2004 10:12:08

Comment:

OK Cancel Help

LAD/STL/FBD (30:150)



The instance data block DB 20 does not exist. Do you want to generate it?

Yes

No

Details...

Help

# Exercise 3: Editing the Parameter-assignable Function Block FB 20

The screenshot displays the SIMATIC Manager LAD editor for Function Block FB20. The window title is "LAD/STL/FBD - [FB20 -- 'FB\_FaultEvaluation' -- SERV2\_5\Ch04\_FC-FB\...\FB20]".

**Interface Parameters:**

Contents Of: 'Environment\Interface'	
Name	
IN	
OUT	
IN_OUT	
STAT	
TEMP	

**Network 1: Fault Evaluation**

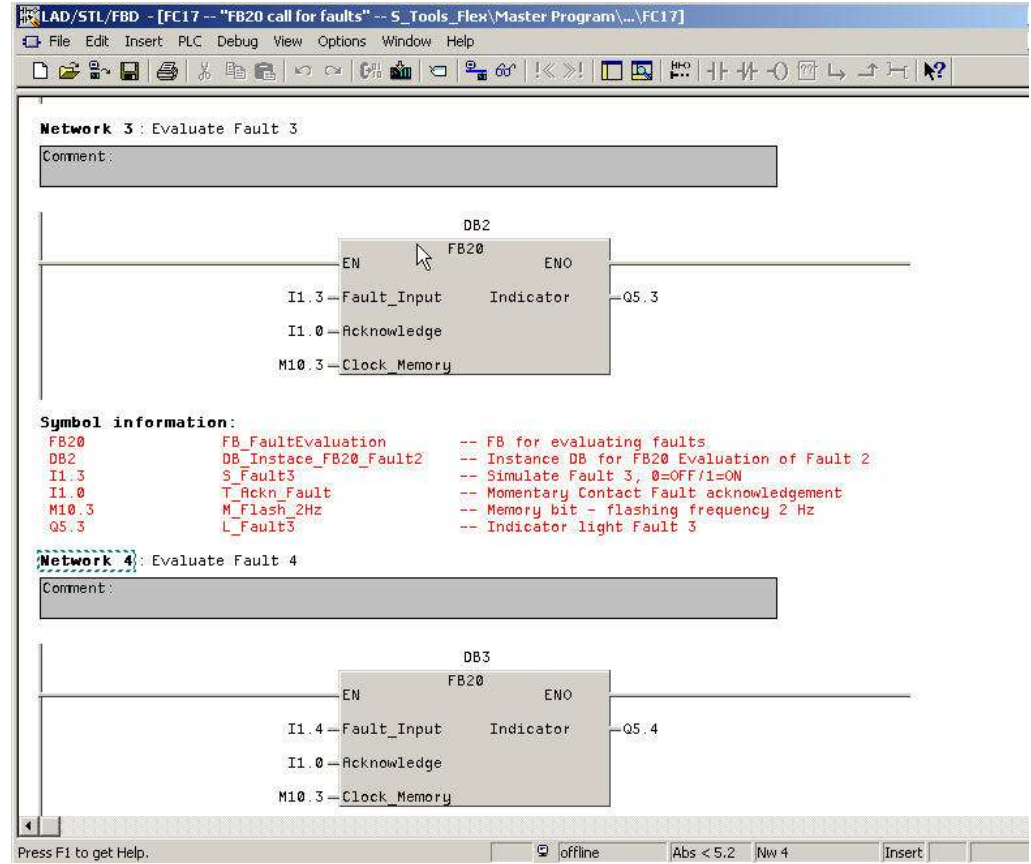
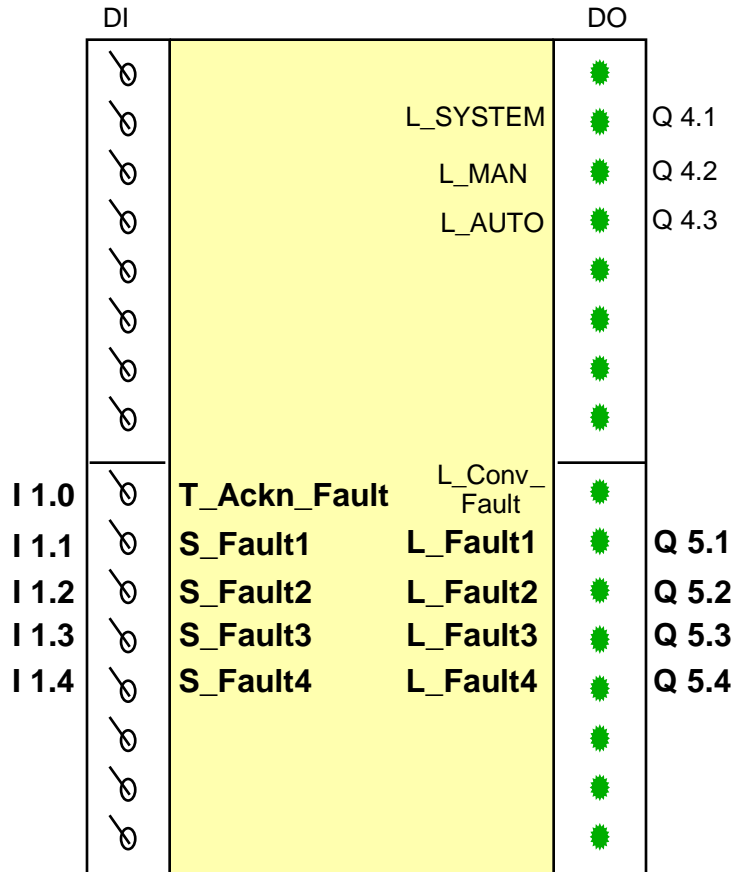
The ladder logic network shows the following logic:

- Inputs:** #Fault\_Input, #Aux\_bit\_edge\_detect, #M\_Bit\_fault\_ev, #Clock\_Memory, #Indicator.
- Logic:** A set coil (S) is triggered by #Fault\_Input AND #Aux\_bit\_edge\_detect (P). A reset coil (R) is triggered by #M\_Bit\_fault\_ev AND #Fault\_Input.
- Output:** The set coil (Q) drives the #Indicator output.

The status bar at the bottom shows "Press F1 to get Help.", "offline", and "Abs < 5.2".



# Exercise 4: Calling the Parameter-assignable Function Block FB 20



## Inserting / Deleting Block Parameters Later On

Name	Data Type	Address	Initial Value	Exclusion address	Termination address	Comment
Fault_Signal	Bool	0.0	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
Acknowledge	Bool	0.1	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
Flash_frequ...	Bool	0.2	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
light_check	bool	0.3	FALSE	<input type="checkbox"/>	<input type="checkbox"/>	
				<input type="checkbox"/>	<input type="checkbox"/>	

**Save (30:22)**

The interface of the block was changed. After Save/Load, this results in an interface conflict with the blocks that reference it. When this block is loaded, the CPU can go into the STOP mode. Continue Save/Load?

Do not display this message again.

Yes No Help

# Checking the Block Consistency

The screenshot shows the SIMATIC Manager interface. On the left, a project tree for 'My\_project' is visible, including 'test station', 'CPU 315-2 DP', 'My\_program', and 'Chapter12'. A context menu is open over the 'Blocks' folder, with 'Check Block Consistency...' selected. A yellow arrow points from this menu item to the 'Check block consistency' dialog box.

The 'Check block consistency' dialog shows the selected object 'OB1' with symbolic name 'OB\_Cycle'. Below it, the 'Object Hierarchy' window displays the 'S7 program Call Tree (References)'. In this tree, several blocks are marked with red symbols indicating inconsistencies: FC17 (a red 'X'), FC20 (a red '!' with a yellow circle), and FC18 (a red '!' with a yellow circle). A yellow box labeled 'Block inconsistency' is positioned to the right of the dialog. Another yellow arrow points from the 'Check Block Consistency...' menu item to the 'S7 program Call Tree' window.

Object name	Symbolic name	Created in la
OB1	OB_Cycle	FBD
OB100	OB_Startup	FBD
OB121		FBD
FB20	FB_Faults	LAD
FC15	FC_Operating_Modes	FBD
FC16	FC_Conveyor	FBD
FC17		
FC18		
FC20		

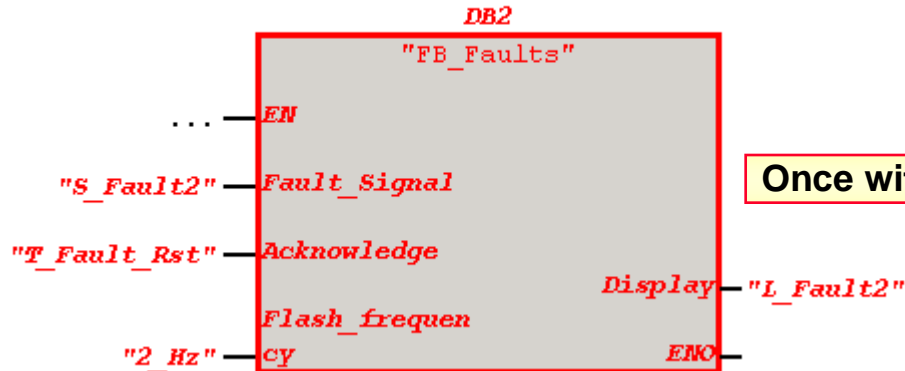
**Block inconsistency**

Object Hierarchy:

- S7 program Call Tree (References)
  - OB1 "OB\_Cycle" [FBD]
  - FC15 "FC\_Operating\_Modes" [FBD]
  - FC16 "FC\_Conveyor" [FBD]
  - FC17 "FC\_Op/Flt\_Mess" [FBD]
  - DB2 -- Instance DB - FB20 "DB\_Instance\_Fault2" [DB Editor]
  - DB3 -- Instance DB - FB20 "DB\_Instance\_Fault3" [DB Editor]
  - FC20 "FC\_Fault" [LAD]
  - FB20 "FB\_Faults" [LAD]
  - FC18 "FC\_Count" [LAD]
  - DB18 -- Global DB "DB\_Parts" [DB Editor]

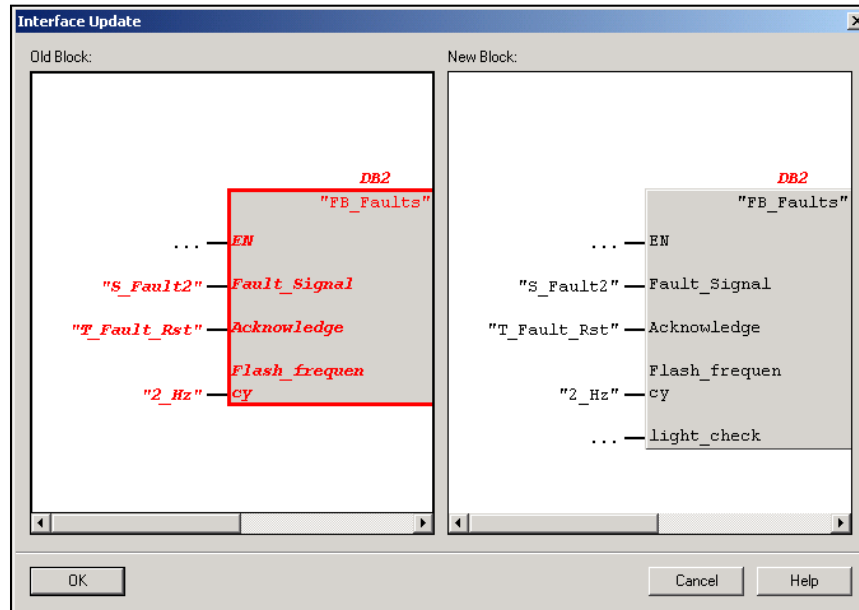
# Corrections when Calling Modified Blocks

Network 3 : Evaluation of disturbance 2



Once with the right

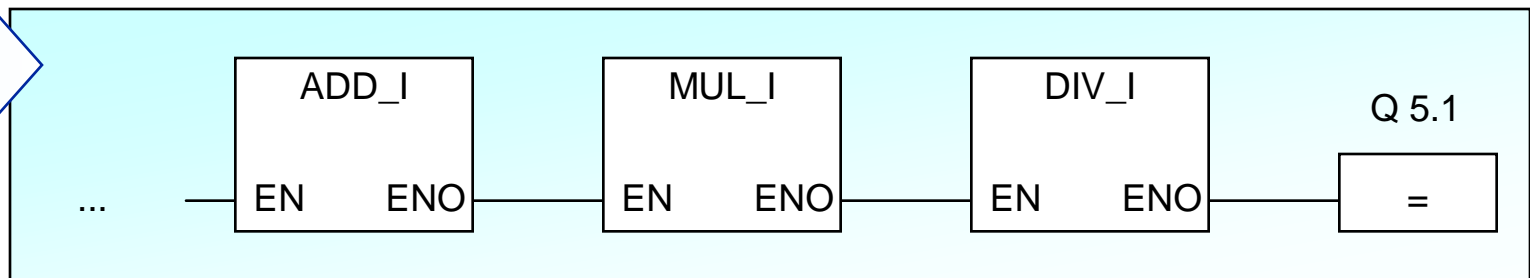
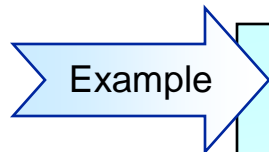
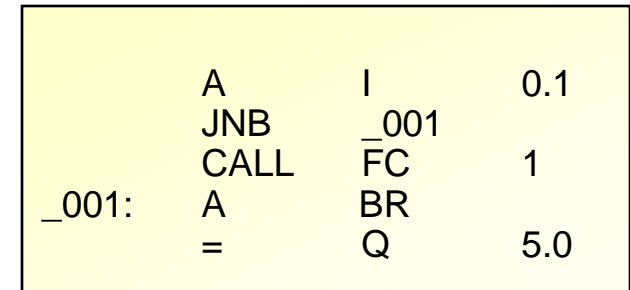
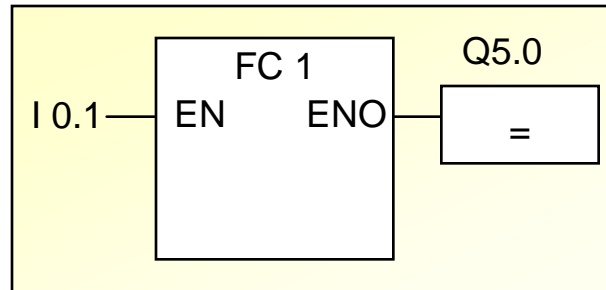
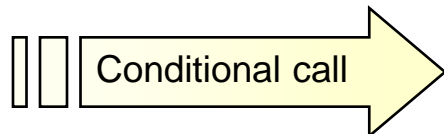
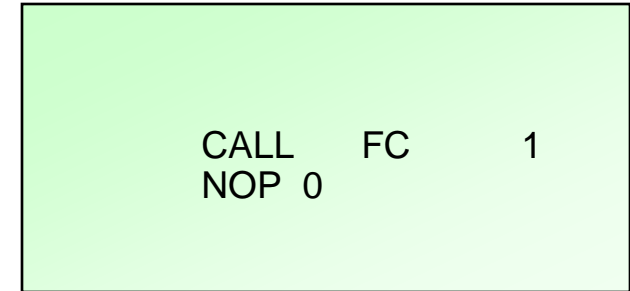
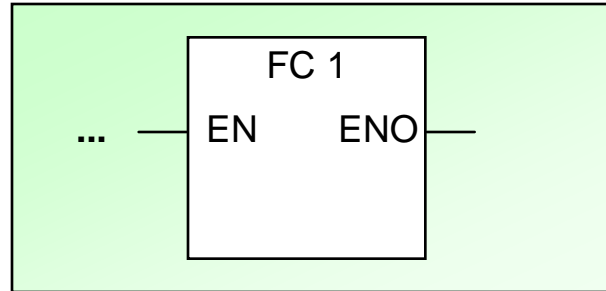
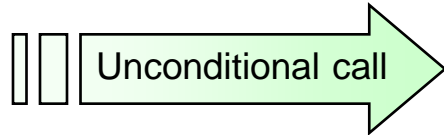
Called Block	
Cut	Ctrl+X
Copy	Ctrl+C
Delete	Del
Insert Network	Ctrl+R
Insert Empty Box	Alt+F9
Go To	
Edit Symbols...	Alt+Return
<b>Update Block Call...</b>	



# Using the EN and ENO Output

LAD/FBD

STL



## Summary: Block Calls

Language	FC		FB	
	without parameters	with parameters	w/o parm., w/o stat var.	with parm., and/or stat var
<b>STL</b>	<ul style="list-style-type: none"> <li>CALL FC1</li> <li>UC FC1</li> <li>CC FC1</li> </ul>	<ul style="list-style-type: none"> <li>CALL FC2 Par1: ... Par2: ... Par3: ...</li> </ul>	<ul style="list-style-type: none"> <li>UC FB1</li> <li>CC FB1</li> </ul>	<ul style="list-style-type: none"> <li>CALL FB2, DB3 Par1: ... Par2: ... Par3: ...</li> </ul>
<b>LAD</b>			not available	
<b>FBD</b>			not available	

## Exercise 5: Recognizing Operand Types

Name	Data Type	Address	Initial Value
Number_1	int	0.0	0
Number_2	int	2.0	0

Statement	Operand Type					
	Global	Local	Absolute	Symbolic	Static	Parameter
L #Number_1						
L #Number_2						
T #Maximum_value						
L „Number_1“						
T MW 40						
T #Number_2						

## If You Want to Know More





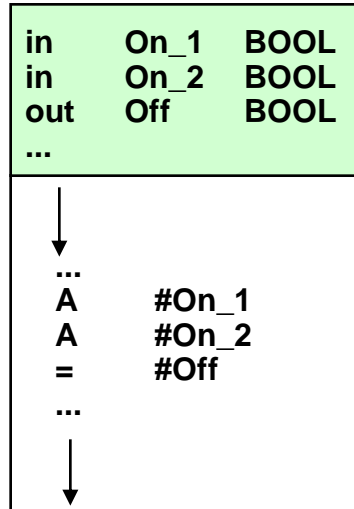
# Comparison of Functions and Function Blocks

Program execution

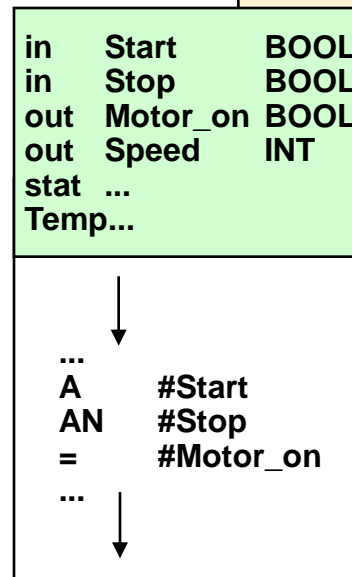
```
CALL FC 10
On_1 := I 0.1
On_2 := I 0.2
Off := Q 8.0
```

```
CALL FB5, DB5
Start := I 0.0
Stop := I 0.1
Motor_on := Q8.0
Speed := QW12
```

FC10



FB 5



0.0	Start	BOOL
0.1	Stop	BOOL
2.0	Motor_on	BOOL
4.0	Speed	INT

DB 5

