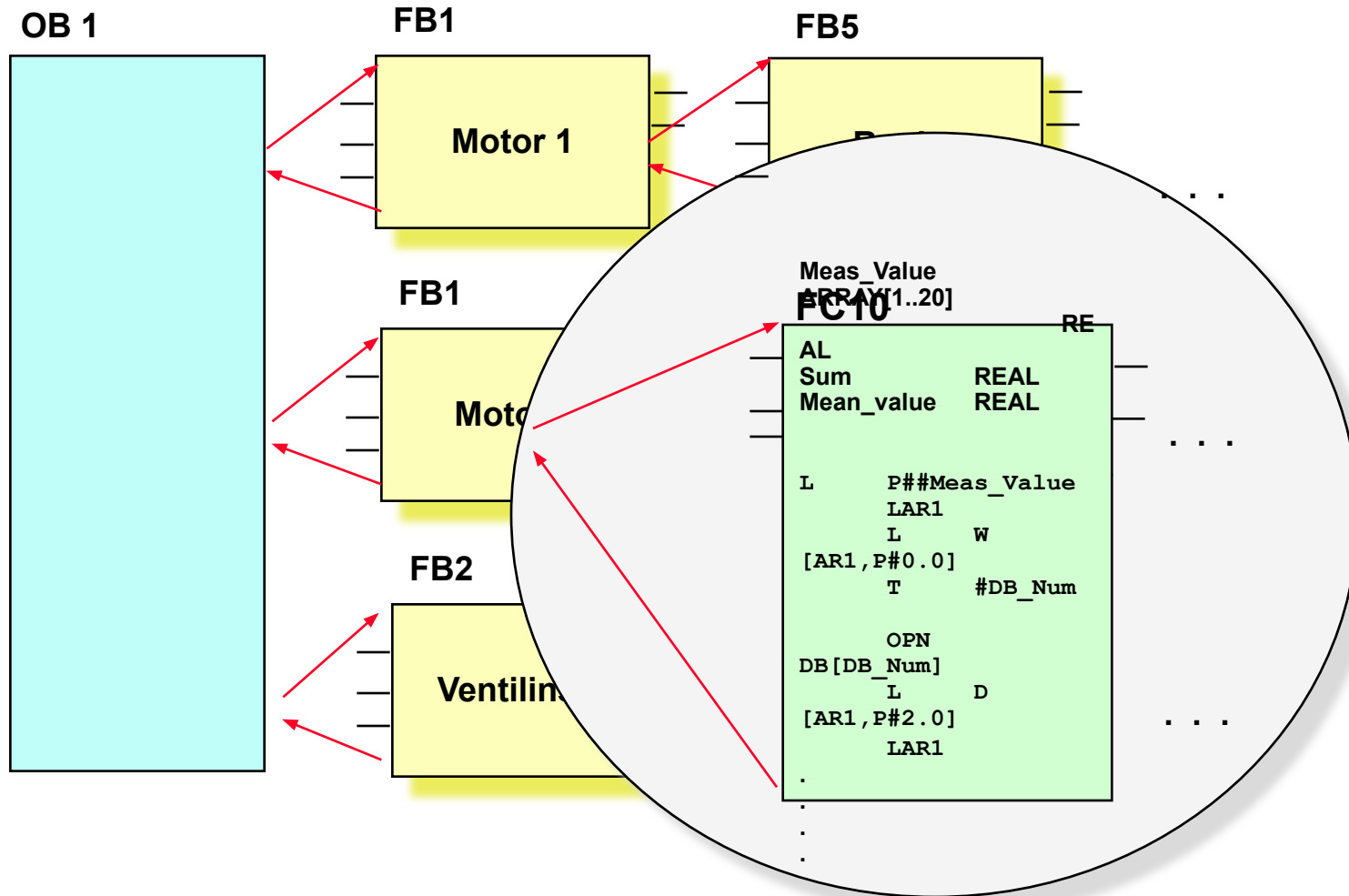


## Приложение: косвенный доступ к параметрам FC и FB



## Вызов функции с параметрами комплексного типа

### Пример: Передача массива (ARRAY) в функцию

**FC21**

Address	Decl.	Name	Type	Initial
0.0	in	Mes_Val	ARRAY[1..10]	
*4.0	in		REAL	

Press F1 for help.

**DB5 "Temperature"**

Address	Name	Type	Initial Value
0.0		STRUCT	
+0.0	sequence	ARRAY[1..10]	5 (2.730000e+002)
*4.0		REAL	

Press F1 for help. Stat. Data:90 Dyn. Data:0 Insert

### Назначение параметров сложного типа возможно только символически

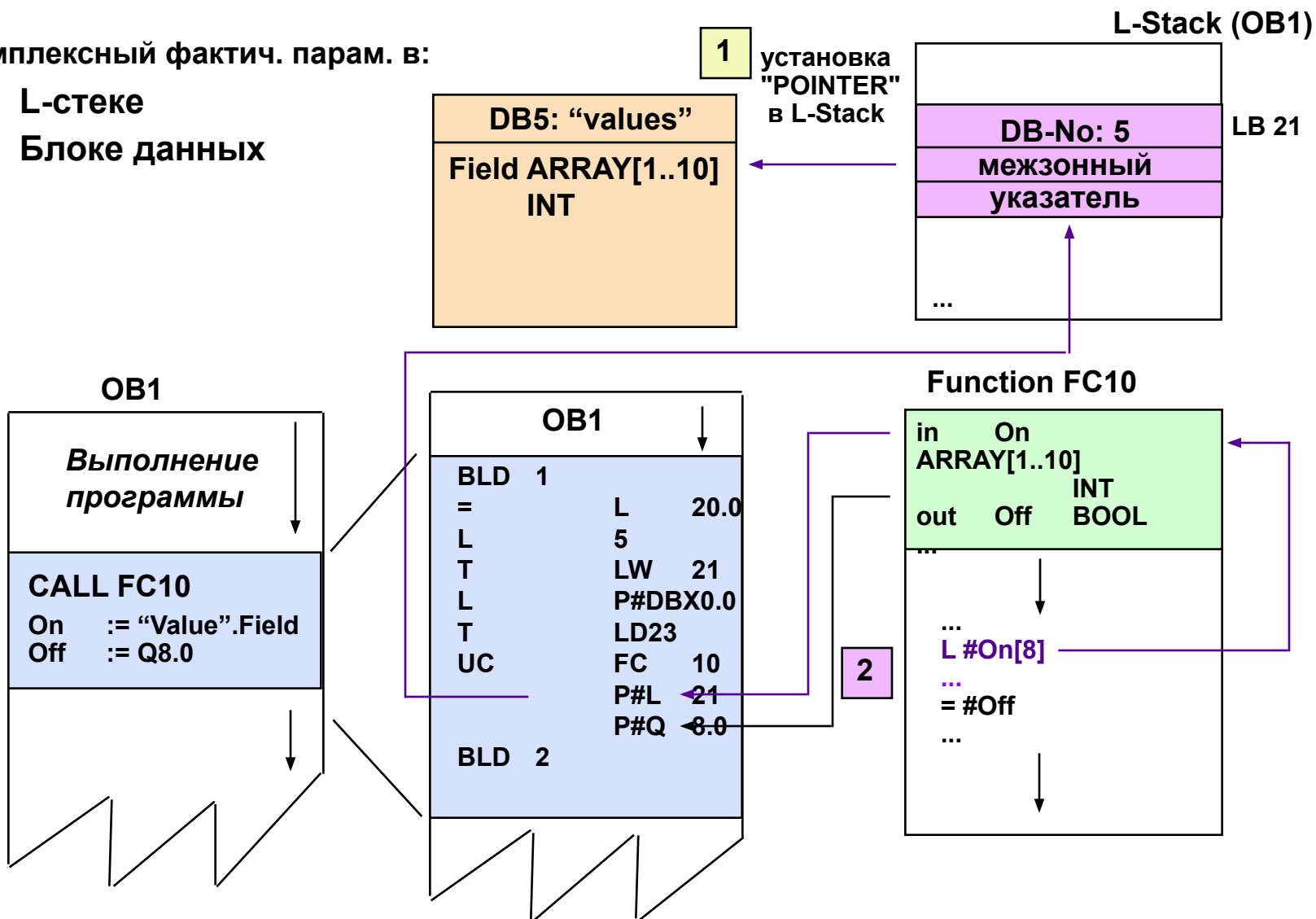
Network 1: Meas\_Val объявлен как array в FC21

```
CALL FC    21
  Meas_Val:="Temperature".sequence
```

## Передача параметров комплексных типов

Комплексный фактич. парам. в:

- L-стеке
- Блоке данных



## Косвенный доступ к сложным типам данных

Address	Declaration	Name Type	Start value	Comment	
0.0	in Meas_Val	ARRAY[1..8]			
*4.0	REAL				
32.0	out Sum	REAL			
36.0	out Mean_Val	REAL			
	in_out				
0.0	temp DB_Num	WORD			

Network 1: Определение номера DB и начального адреса

```

L   P## Meas_Val    // Загрузите адрес POINTER в ACCU1,
LAR1      // и оттуда загрузите в AR1;
L   W [AR1,P#0.0]   // Определите номер DB
T   #DB_Num        // и загрузите его во временную переменную;
OPN DB[DB_Num]      // Открыть DB
L   D [AR1,P#2.0]   // Определите указатель на область
LAR1      // и загрузите его в AR1;

```

Network 2: Вычисление суммы

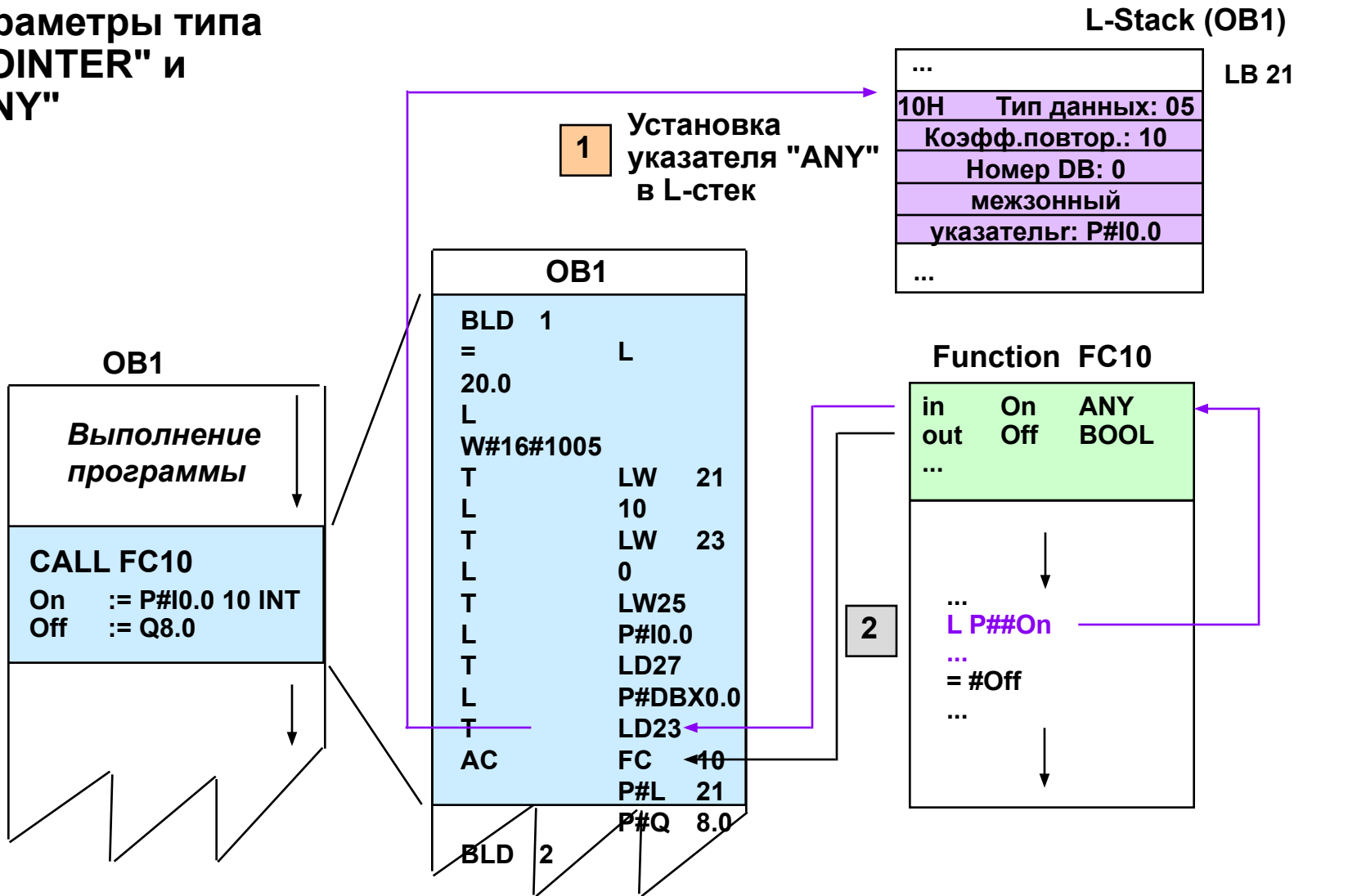
```

L   0.000000e+000   // 0 в ACCU1 (sum =0.0)
L   8                // Счетчик в ACCU1; Sum=0 в ACCU2
BEGN: TAK           // сумма в ACCU1, счетчик в ACCU2
ENT                // счетчик в ACCU3
L   D[AR1,P#0.0]     // элемент массива в ACCU1
+R                // сумма в ACCU1, счетчик в ACCU2
+AR1 P#4.0;         // Увеличение AR1 на 4 байта
TAK                // Счетчик цикла в ACCU1, сумма в ACCU2
LOOP BEGN;          // Уменьшить на 1 счетчик цикла и переход, если
                    // необходимо
T   #Sum             // Запись суммы в #Sum

```

## Передача параметра типа POINTER

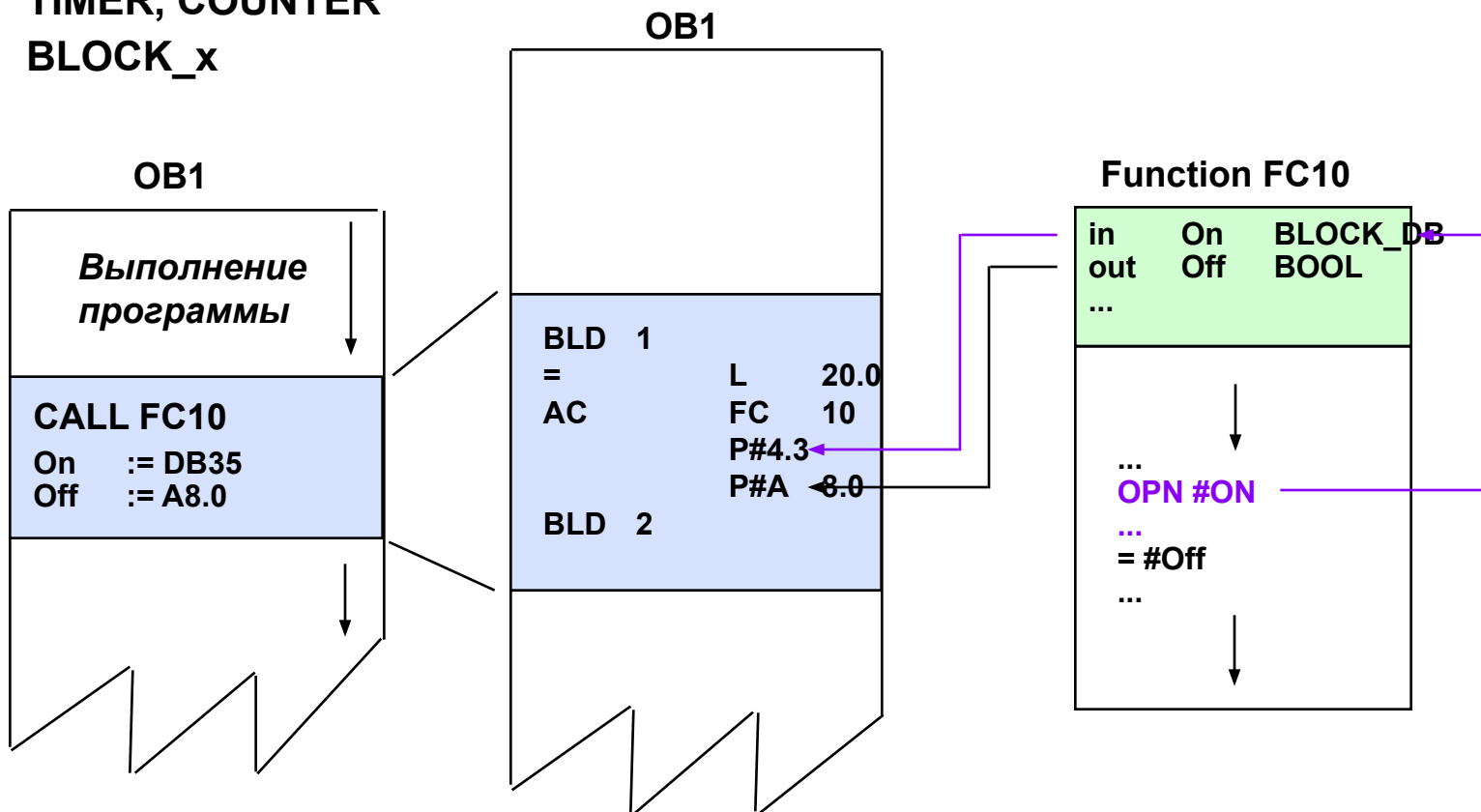
Параметры типа  
"POINTER" и  
"ANY"



## Передача параметров параметрического типа

### Блочные параметры:

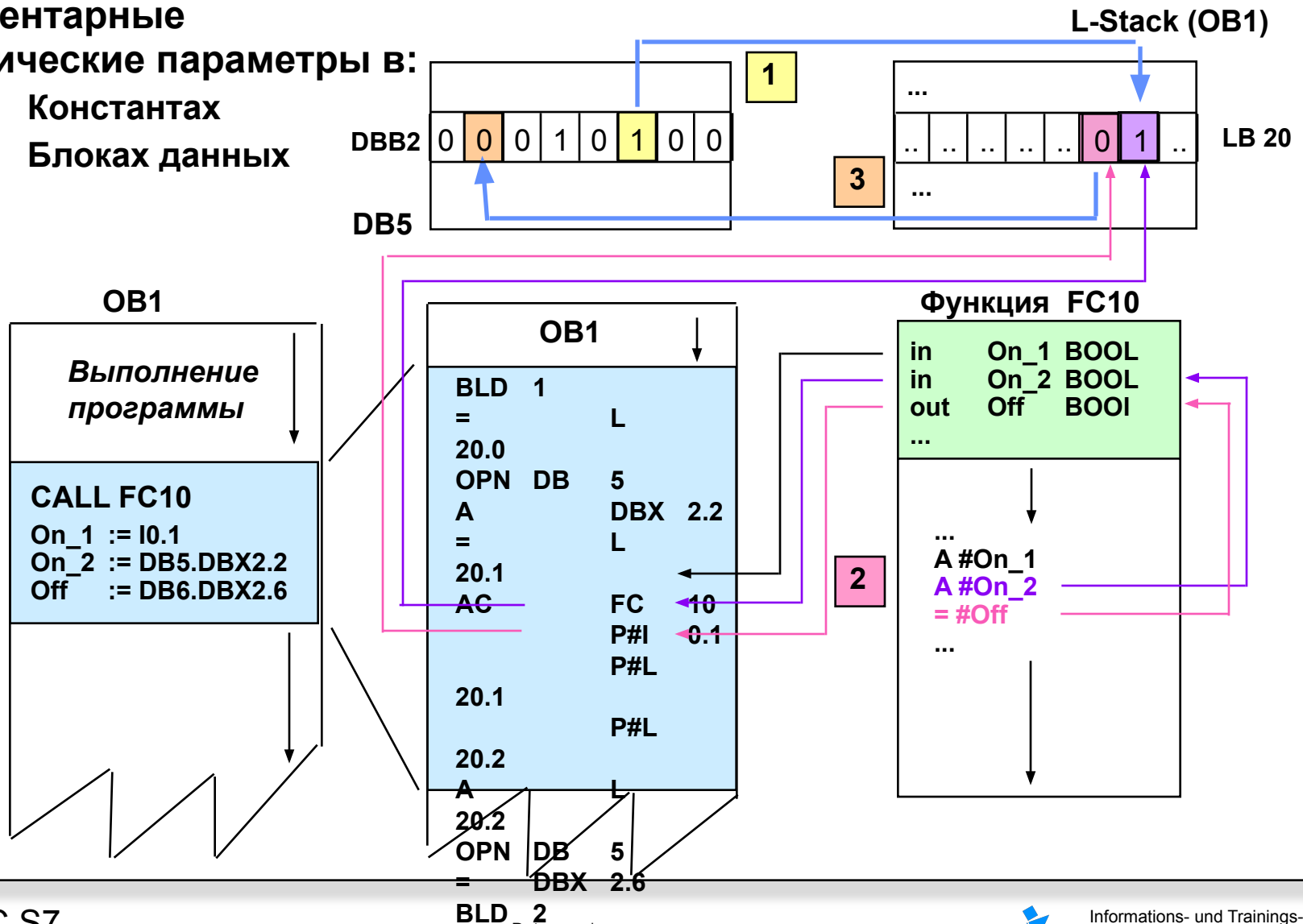
- TIMER, COUNTER
- BLOCK\_x



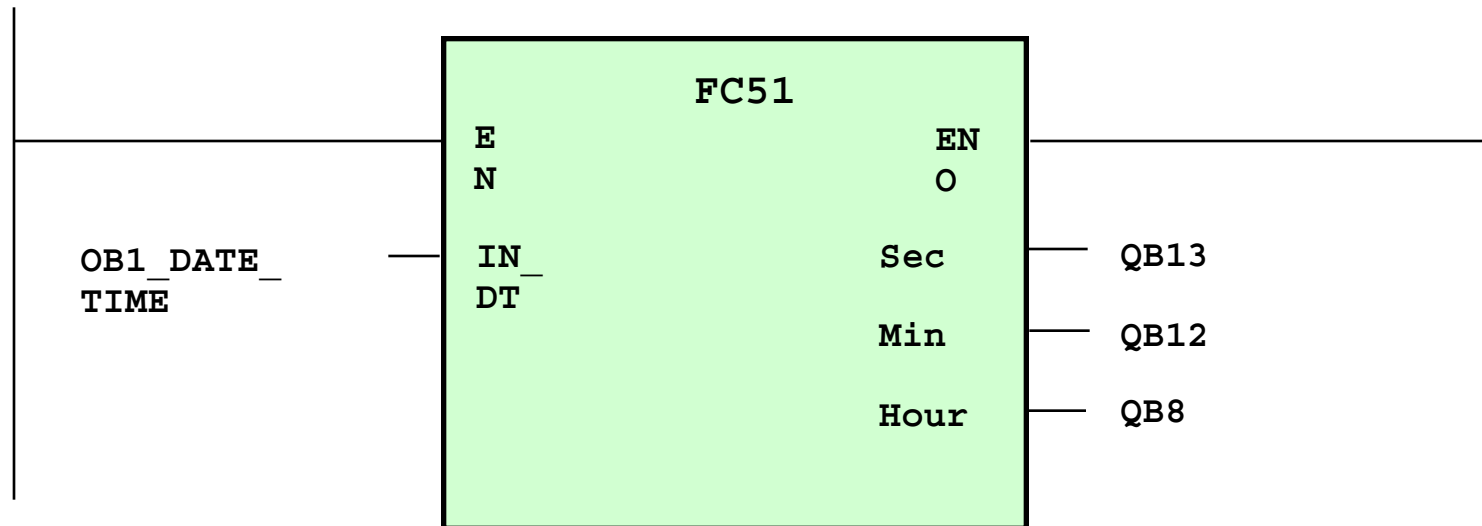
## Специальные возможности для элементарных фактических параметров в DB и константах

Элементарные фактические параметры в:

- Константах
- Блоках данных



## Упражнение А.1: Оценка параметров типа DATE\_AND\_TIME в FC





## Вызов FB с параметрами сложных типов

Пример: Передача массивов в функциональный блок

**FB17**

Address	Decl.	Name	Type	Initial Value	Comment
0.0	in	Meas_1	ARRAY[1..10]		
*4.0	in		REAL		
40.0	out	Sum_1	REAL	0.000000e+000	
44.0	out	Sum_2	REAL	0.000000e+000	
48.0	in_out	Meas_2	ARRAY[1..15]		
*4.0	in_out		REAL		
54.0	stat	DB_Num	INT		
		temp			

**DB2 "Temperature"**

Address	Name	Type	Initial
0.0		STRUCT	
+0.0	Cylinder	ARRAY[1..10]	
*4.0		REAL	
+40.0	Shaft	ARRAY[1..15]	
*4.0		REAL	
=100.0		END_STRUCT	

**Назначение сложных параметров  
ВОЗМОЖНО ТОЛЬКО СИМВОЛИЧЕСКИ**

Network 1:

```

CALL FB 17, DB 30
    Meas_1
:= "Temperature".Cylinder
Sum_1 := MD20
Sum_2 := MD30
Meas_2 := "Temperature".Shaft
  
```

## Косвенная адресация ВХОДНЫХ и ВЫХОДНЫХ параметров

Address	Declaration	Name	Type	Start value	Comment	
0.0	in Meas_1	ARRAY	[1..10]			
*4.0	REAL					
40.0	out Sum_1	REAL	0.000000e+000			
44.0	out Sum_2	REAL	0.000000e+000			
48.0	in_out Meas_2	ARRAY	[1..15]			
*4.0	in_out					
54.0	stat DB_Num	INT	0			

Network 1: Определение стартового адреса Meas\_1

```

LAR1 P##Meas_1 // Загрузите межзонный указатель на параметр без
                // смещения адреса (мультиэкземпляр) в AR1
TAR2           // Загрузить смещение адреса в ACCU1
+AR1           // Добавить смещение адреса к AR1;
                // AR1 теперь указывает на параметр в экземпляре DB
                // экземпляр DB уже открыт

```

Network 2: Доступ к Meas\_1

```

L      0.000000e+000 // 0 в ACCU1 (сумма =0.0)
L      10           // Счетчик в ACCU1; сумма =0 в ACCU2
BEGN:  TAK          // Сумма в ACCU1, счетчик в ACCU2
ENT     // Счетчик в ACCU3
L      D[AR1,P#0.0] // Компонент массива в ACCU1
+R      // Сумма в ACCU1, счетчик в ACCU2
+AR1 P#4.0;        // Увеличение AR1 на 4 байта
TAK     // Счетчик цикла в ACCU1, сумма в ACCU2
LOOP BEGN;         // Уменьшить счетчик цикла на 1 и переход, если это
                    // необходимо
T      #Sum_1       // Запись суммы в #Sum_1

```

## Косвенная адресация in\_out - параметров

Address	Declaration	Name	Type	Start value	Comment
0.0	in Meas_1	ARRAY[1..10]			
*4.0	REAL				
40.0	out Sum 1			+000	
44.0	out Sum 2	REAL 0.000000e+000			
48.0	in_out Meas_2	ARRAY[1..15]			
*4.0	in_out	REAL			
54.0	stat DB_Num				

Network 3: Определение начального адреса Meas\_2

```

LAR1 P##Meas_2 // Загрузка межзонного указателя на POINTER
TAR2           // Загрузка смещения адреса в ACCU1, прибавление к AR1;
+AR1           // AR1-новый указатель на POINTER в экземпляр DB
L      W [AR1,P#0.0] // Загрузка номера DB из POINTER в ACCU1
T      #DB_Num // Запись номера DB (или 0) в статическую переменную
OPN    DB [#DB_Num] // Открытие DB
L      D [AR1,P#2.0] // Загрузка межзонного указателя на параметр
LAR1           // Загрузка указателя в AR1, AR1 указатель на параметр

```

Network 4: Доступ к Meas\_2

```

L      0.000000e+000 // 0 в ACCU1 (сумма =0.0)
L      15           // Счетчик в ACCU1; сумма=0 в ACCU2
BEGN:  TAK           // Сумма в ACCU1, счетчик в ACCU2
ENT           // Счетчик в ACCU3
L      D[AR1,P#0.0] // Элемент массива в ACCU1
+R           // Сумма в ACCU1, счетчик в ACCU2
...          // ...

```

## Передача параметров

### Глубина вложения:

- S7-300: max. 8 S7-400: max. 16

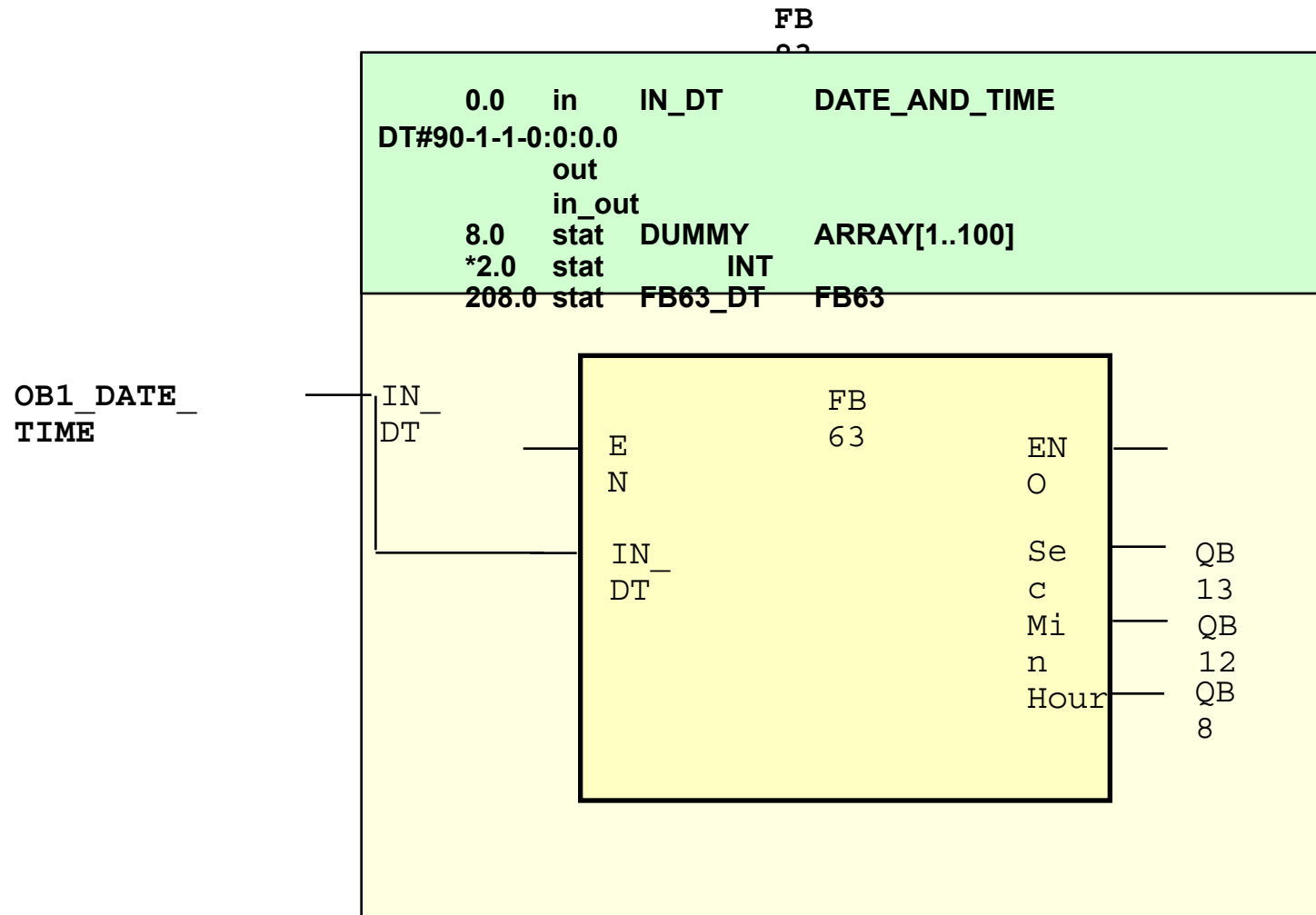


### Передача зависит от блока, типа данных и вида параметра:

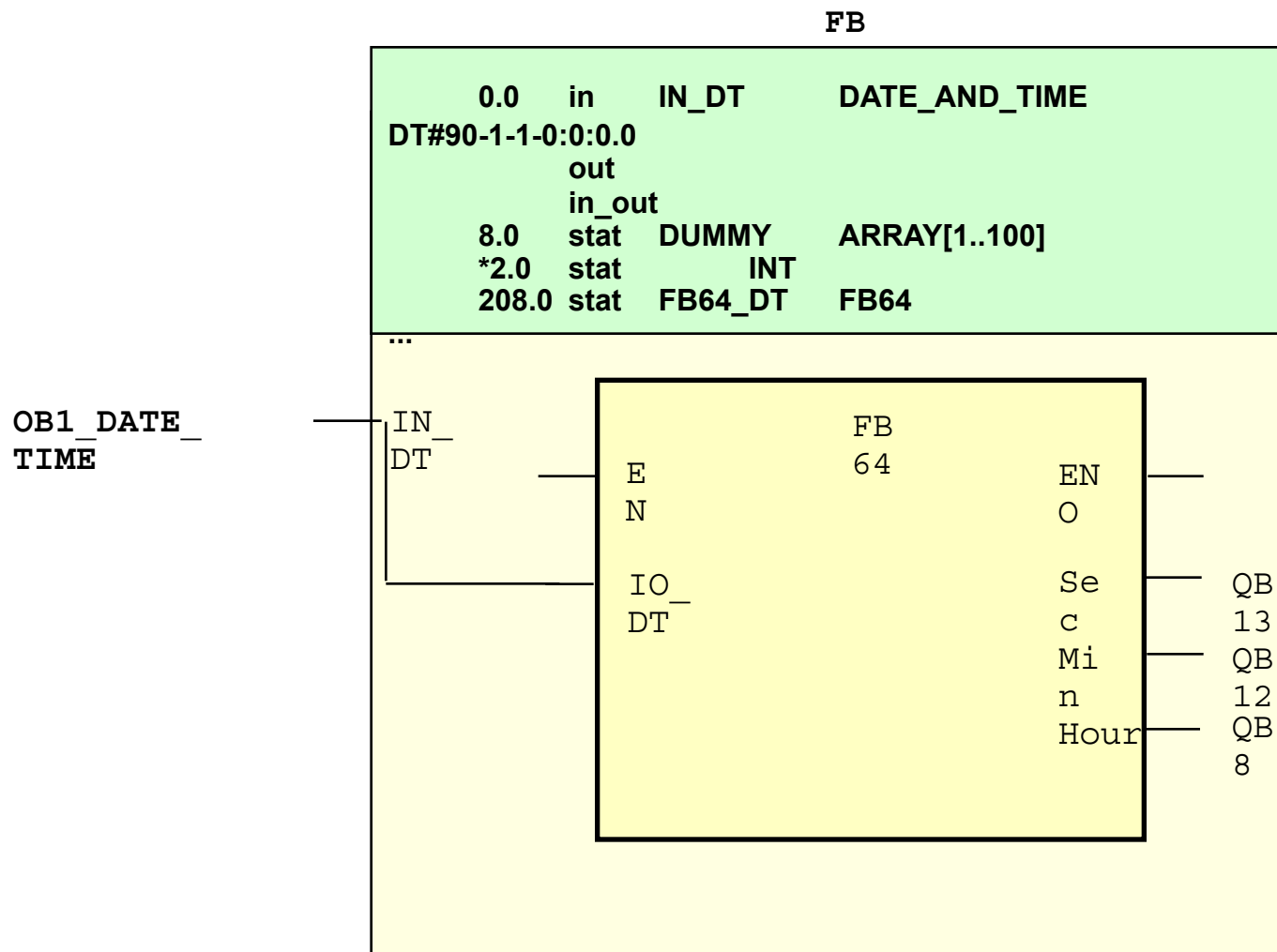
Вызов	FC выз. FC		FB выз. FC			FC выз. FB			FB выз. FB			
Тип данных	E	C	P	E	C	P	E	C	P	E	C	P
Input -> Input	x	-	-	x	x	-	x	-	x	x	x	x
Output -> Output	x	-	-	x	x	-	x	-	-	x	x	-
in/out -> Input	x	-	-	x	-	-	x	-	-	x	-	-
in/out -> Output	x	-	-	x	-	-	x	-	-	x	-	-
in/out -> in/out	x	-	-	x	-	-	x	-	-	x	-	-

E: Элементарный тип данных  
 C: Комплексный тип данных  
 P: Параметрический тип (Timer, Counter, Block\_x)

## Упражнение A.2: Оценка параметра типа DATE\_AND\_TIME в FB



## Упражнение A.3: Оценка In\_Out параметров в FB



## Решение к упражнению А.1: Доступ к DT параметрам в FC

## Решение к упражнению A.2: Доступ к DT параметрам в FB



## **Решение к упражнению А.3: Доступ к I/O параметрам в FB (Part 1)**

## **Решение к упражнению A.3: Доступ к I/O параметрам в FB (Part 2)**