

Конфигурационная посылка и посылка назначения параметров для ET 200S

Конфигурируя DP –мастер, Вы можете определить, какой формат должен использоваться в структуре конфигурации:

- Конфигурация в STEP 7 с помощью HWConfig: Интеграция IM 151/CPU как слэйва, при работе с S7 мастером: Специальный ID формат: **SKF**
- Конфигурация с другим инструментом конфигурации: Интеграция IM 151/CPU как слэйва через DDB: Нормальный ID формат: **AKF**

Конфигурация со STEP 7 V5.1

Когда Вы используете S7- мастер и конфигурируете и параметризуете его с помощью **STEP 7, STEP 7** обеспечивает Вам поддержку во время ввода или Вы можете воспользоваться встроенной системой помощи.

В этом случае, Вы не нуждаетесь в информации этого приложения.

Конфигурация с любым другим программным обеспечением

Если Вы вводите адресные области промежуточной памяти CPU с использованием конфигурационной посылки и посылки назначения параметров, Вы найдете информацию, которая Вам потребуется для IM 151/CPU в этом приложении.

Для вывода конфигурации используется ID формат структуры конфигурационной посылки.

Раздел	Содержание	Страница
A.1	Конфигурационная кодовая посылка (SKF)	A-2
A.2	Конфигурационная кодовая посылка (AKF)	A-4
A.3	Структура кодовой посылки назначения параметров	A-6

A1 Конфигурационная кодовая посылка (SKF)

Длина конфигурационной посылки зависит от числа адресных областей в промежуточной памяти CPU модуля IM 151. Первые 15 байт в структуре посылки уже назначены, потому что первые 3 идентификатора являются 5-байтовыми ID-константами. Максимальная область, которая может быть задана: 0-64 байт/слов входов, 0-64 байт/слов выходов, при возможной асимметричной конфигурации.

Таблица A-1 Структура конфигурационной посылки в специальном формате SKF

Сконфигурированные адресные области	Байт				
	n	n + 1	n + 2	n + 3	n + 4
Фиксированный диапазон (байты 0 ... 14)	04	00	00	QD	C4
	04	00	00	8B	41
	04	00	00	8F	C0
1-я сконфигурированная адресная область (байты 15 ... 19)	См.таблицу A-2				
1-я сконфигурированная адресная область (байты 20 ... 24)					
...					
32-я сконфигурированная адресная область (байты 170 ... 174)					

Идентификаторы для адресных областей

Идентификаторы для конфигурации зависят от типа адресных областей.

Следующая таблица содержит список всех идентификаторов для адресных областей.

Таблица А-2 Идентификаторы адресных областей промежуточной памяти

Адресные области	Идентификаторы (16-ричный код)				
	SFF	Длина (байт)	Изготовитель Длина комментария = 3		
	Байт 0	Байт 1	Байт 2	Байт 3	Байт 4
Входы	См. Рис. А–1	См. Рис. А–2	00 _н	83 _н	40 _н
Выходы			00 _н	93 _н	40 _н

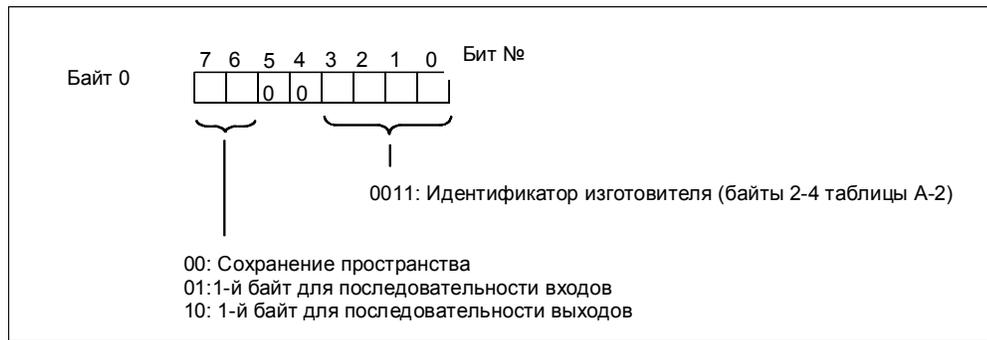


Рис А-1 Описание байта 0 идентификатора адресных областей



Рис А-2 Описание байта 1 идентификатора адресных областей

A2 Конфигурационная кодовая посылка (AKF)

DDB (База данных Устройства) Файл

Если ваш DP - мастер не поддерживает кодовую посылку конфигурации в специальном формате ID (например DP- мастер прочих производителей), Вы можете получить DDB файл с нормальным форматом ID в Интернете в <http://www.ad.siemens.de/csi/gsd> (см. также Раздел 5.1).

Структура конфигурационной кодовой посылки

Длина конфигурационной кодовой посылки зависит от количества адресных областей промежуточной памяти в CPU компоненте. Первые три байта конфигурационной кодовой посылки - всегда "0". Это представляет 3 установленных 1-байтовых идентификатора. Максимальная область, которая может быть задана: 0-16 байт/слов входов, 0-16 байт/слов выходов, возможна асимметричная конфигурация.

Table A-3 Структура конфигурационной посылки в нормальном формате (AKF)

Сконфигурированная адресная область	Байт																		
1.	0 0 0 0 0 0 0 0 0																		
2.	0 0 0 0 0 0 0 0 0																		
3.	0 0 0 0 0 0 0 0 0																		
4.	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">7</td> <td style="text-align: center;">6</td> <td style="text-align: center;">5</td> <td style="text-align: center;">4</td> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">□</td> <td style="text-align: right;">Бит №</td> </tr> </table> <div style="margin-left: 100px;"> <p>Длина входных – выходных данных в байтах или словах+ 1</p> <p>01: входные данные 10: выходные данные</p> <p>0: длина в байтах 1: длина в словах</p> <p>Консистентность: 0: в байтах или словах 1: в общей длине</p> </div>	7	6	5	4	3	2	1	0		□	□	□	□	□	□	□	□	Бит №
7	6	5	4	3	2	1	0												
□	□	□	□	□	□	□	□	Бит №											
:																			
:																			
32nd																			

Конфигурационная кодовая посылка с установленными по умолчанию адресными областями

Если Вы не делаете параметризации IM 151/CPU и не определяете адресных областей для обмена данными с DP мастером, IM 151/CPU после запуска в работу получает установленные по умолчанию адресные области для PROFIBUS-DP.

В таблице ниже показана кодовая посылка с установленными по умолчанию адресными областями :

- 16 слов входных данных; единица консистентности - слово
- 16 слов выходных данных , единица консистентности - слово

Таблица А–4 Структура кодовой посылки с установленными по умолчанию адресными областями (Нормальный формат идентификаторов в соответствии с AKF)

Сконфигурированные адресные области	Байт																		
1.	0 0 0 0 0 0 0 0																		
2.	0 0 0 0 0 0 0 0																		
3.	0 0 0 0 0 0 0 0																		
4.	<table style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">7</td><td style="text-align: center;">6</td><td style="text-align: center;">5</td><td style="text-align: center;">4</td><td style="text-align: center;">3</td><td style="text-align: center;">2</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td><td style="text-align: right;">Бит №</td> </tr> <tr> <td style="text-align: center;">0</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td></td> </tr> </table> <div style="margin-left: 100px;"> <p>Длина входных данных : 16 слов</p> <p>Входные данные</p> <p>Длина в словах</p> <p>Консистентность: слово</p> </div>	7	6	5	4	3	2	1	0	Бит №	0	1	0	1	1	1	1	1	
7	6	5	4	3	2	1	0	Бит №											
0	1	0	1	1	1	1	1												
5.	<table style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">7</td><td style="text-align: center;">6</td><td style="text-align: center;">5</td><td style="text-align: center;">4</td><td style="text-align: center;">3</td><td style="text-align: center;">2</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td><td style="text-align: right;">Бит №</td> </tr> <tr> <td style="text-align: center;">0</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td></td> </tr> </table> <div style="margin-left: 100px;"> <p>Длина входных данных : 16 слов</p> <p>Выходные данные :</p> <p>Длина в словах</p> <p>Консистентность: слово</p> </div>	7	6	5	4	3	2	1	0	Бит №	0	1	1	0	1	1	1	1	
7	6	5	4	3	2	1	0	Бит №											
0	1	1	0	1	1	1	1												

А.3 Структура посылки назначения параметров

Все параметры для DP-слэйва сохраняются в кодовой посылке назначения параметров.

Структура кодовой посылки назначения параметров

Длина кодовой посылки назначения параметров для IM 151/CPU - 16 байтов:

- Стандартная секция; байты от 0 до 6
- Параметры IM 151/CPU; байты от 7 до 15

Структура стандартной секции

Первые 7 байт посылки назначения параметров стандартизированы в соответствии с EN 50170, Том 2, PROFIBUS и имеют следующее содержание:

Байт 0	08 _h	Состояние станции
Байт 1	01 _h	Фактор контроля времени 1
Байт 2	06 _h	Фактор контроля времени 2
Байт 3	0B _h	Время ответа T_{RDY}
Байт 4	80 _h	ID изготовителя, старший байт
Байт 5	6C _h и 6D _h	ID изготовителя, младший байт
Байт 6	00 _h	Группа ID

Рис.А-3 Стандартная секция кодовой посылки назначения параметров

Структура Общих Параметров для IM 151/CPU

Длина общих параметров для IM 151/CPU - 3 байта.

Следующие параметры могут быть установлены:

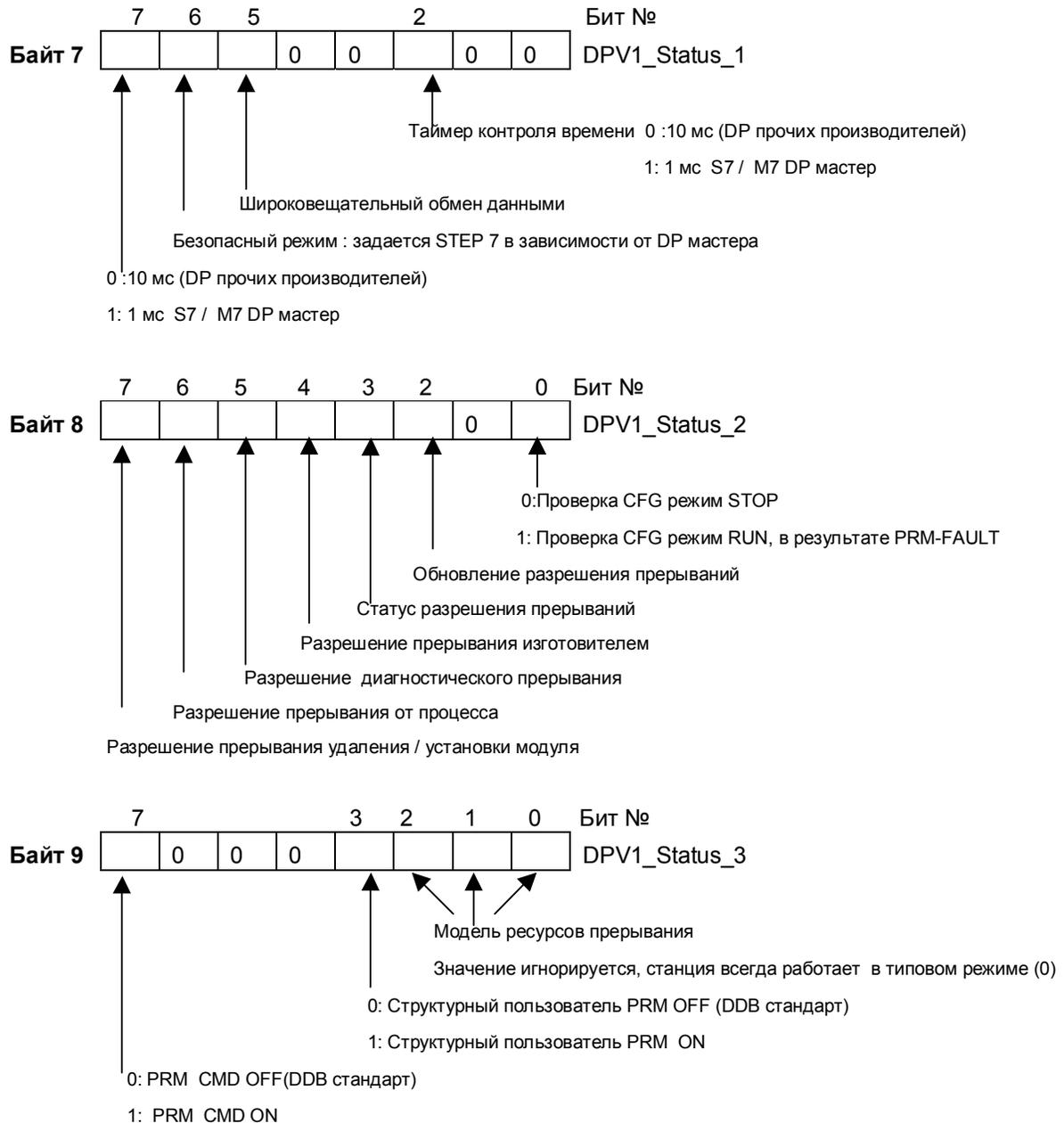


Рис. А-4 Структура общих параметров для IM 151/CPU

Diese Seite ist eine **Vakat**-Seite, die an das Ende eines Kapitels mit ungerader Seitennummer angehängt wird.

Список команд

В

Это приложение содержит полный набор команд для программирования CPU в IM 151/CPU с использованием STEP 7. Для каждой команды указано время выполнения.

Вы найдете детальные описания всех команд, вместе с примерами, в руководствах по программированию на STEP 7.

Примечание:

В случае команд косвенной адресации(пример в Приложении В.4), вы должны прибавить время загрузки адреса к времени выполнения команды(пример в Приложении В.5)

Краткий обзор

Приложение	Содержание	Страница
В.1	Обозначение адресных областей и диапазоны параметров	В-2
В.2	Сокращения	В-3
В.3	Вспомогательные регистры	В-3
В.4	Примеры адресации	В-5
В.5	Время выполнения команд с косвенной адресацией	В-7
В.6	Список команд	В-13

В.1 Обозначение адресных областей и диапазоны параметров

Обозначение	Диапазон параметров	Описание
Q	От 0.0 до 127.7	Выходы (в PIQ)
QB	От 0 до 127	Выходной байт (в PIQ)
QW	От 0 до 126	Выходное слово (в PIQ)
QD	От 0 до 124	Выходное двойное слово (в PIQ)
B	-	Байт с основным регистром косвенной адресации
W	-	Слово с основным регистром косвенной адресации
D	-	Двойное слово с основным регистром косвенной адресации
DBX	От 0.0 до 8191.7	Бит данных в блоке данных
DB	1 до 127	Блок данных
DBB	От 0 до 8191	Байт данных в блоке данных
DBW	От 0 до 8190	Слово данных в блоке данных
DBD	От 0 до 8188	Двойное слово данных в блоке данных
DIX	От 0.0 до 8191.7	Бит данных в блоке - экземпляре
DI	1 до 127	Блок-экземпляр
DIB	От 0 до 8191	Байт данных в блоке - экземпляре
DIW	От 0 до 8190	Слово данных в блоке - экземпляре
DID	От 0 до 8188	Двойное слово данных в блоке - экземпляре
I	От 0.0 до 127.7	Вход (в PII)
IB	От 0 до 127	Входной байт (in PII)
IW	От 0 до 126	Входное слово (in PII)
ID	От 0 до 124	Входное двойное слово (in PII)
L	От 0.0 до 255.7	Локальные данные
LB	От 0 до 255	Байт локальных данных
LW	От 0 до 254	Слово локальных данных
LD	От 0 до 252	Двойное слово локальных данных
M	От 0.0 до 255.7	Меркерный бит
MB	От 0 до 255	Меркерный байт
MW	От 0 до 254	Меркерное слово
MD	От 0 до 252	Меркерное двойное слово
PQB	От 0 до 1535	Периферийный выходной байт
PQW	От 0 до 1534	Периферийное выходное слово
PQD	От 0 до 1535	Периферийное выходное двойное слово
PIB	От 0 до 1535	Периферийный входной байт
PIW	От 0 до 1534	Периферийное входное слово
PID	От 0 до 1532	Периферийное входное двойное слово
T	От 0 до 127	Таймер
C	От 0 до 63	Счетчик
Параметр	-	Использование
B#	-	Константа, 2 или 4 байта
D#	-	константа данных IEC
L#	-	32-разрядная целая константа
P#	-	Константа типа Pointer
S5T#	-	Константа S5 таймера (16 bits)*
T#	**	Константа времени (16/32 bits)
TOD#	-	Константа времени IEC (32 bits)
C#	-	Константа счетчика (16/32 bits)
2#	-	Двоичная константа (16/32 bits)
16#	-	Шестнадцатиричная константа (16/32 bits)

* для загрузки в S5 таймеры

** T # 1D_5M_3M_1S_2MS

В.2 Аббревиатура

Следующие сокращения и мнемоника используются в списке команд:

Сокраще- ние	Используется для	Пример
k8	8–битная константа	32
k16	16– битная константа	62 531
k32	32– битная константа	127 624
i8	8– битное целое число	-155
i16	16– битное целое число	+6523
i32	32– битное целое число	-2 222 222
m	R#x.y (pointer)	R#240.3
n	битовая константа	1001 1100
p	Шестнадцатиричная константа	EA12
LABEL	Символьный адрес перехода (максимально - 4 символа)	DEST

В.3 Вспомогательные регистры

ACCU1 и ACCU2 (32 разряда)

Аккумуляторы - регистры для обработки байтов, слов или двойных слов. ,
Адресные области загружаются в аккумуляторы для дальнейшей обработки.
Результат логической операции(RLO) - всегда в ACCU1.

Аккумуляторы - 32 разрядные регистры.

Обозначения Аккумулятора:

ACCU		Разряды
ACCU1	ACCU2	с 0 до 31
ACCU1–L	ACCU2–L	с 0 до 15
ACCU1–H	ACCU2–H	с 16 до 31
ACCU1–LL	ACCU2–LL	с 0 до 7
ACCU1–LH	ACCU2–LH	с 8 до 15
ACCU1–HL	ACCU2–HL	с 16 до 23
ACCU1–HH	ACCU2–HH	с 24 до 31

Адресные регистры AR1 и AR2 (32 разряда)

Адресные регистры содержат идентификаторы областей памяти и адреса для инструкций, которые используют косвенную адресацию. Адресные регистры - 32 разрядные регистры. Внутризонная и межзонная косвенная адресация имеет следующую структуру:

- Внутризонная косвенная адресация:

00000000 00000bbb bbbbbbbb bbbbbxxx

- Межзонная косвенная адресация:

10000ууу 00000bbb bbbbbbbb bbbbbxxx

Обозначение: b Адрес байта
 x Номер бита
 y Идентификатор области (см. Приложение В.4)

Слово состояния (16 разрядов)

Биты слова состояния оцениваются или устанавливаются инструкциями.

Слово состояния имеет длину-16 бит.

Bit	Assignment	Description
0	/FC	Бит первичного опроса *
1	RLO	Результат логической операции (предыдущей)
2	STA	Статус *
3	OR	Или *
4	OS	Сохраняемое переполнение
5	OV	Переполнение
6	A0	Код состояния 0
7	A1	Код состояния 1
8	BR	Двоичный результат
9 ... 15	Неиспользуются	-
* Бит не может быть оценен в пользовательской программе с L STW инструкцией		

В.4 Примеры Адресации

Примеры адресации	Описание
Прямая адресация	
L +27	Загрузить 16– разрядное целое число "27" в ACCU1
L L#-1	Загрузить 32–разрядное целое число "-1" в ACCU1
L 2#1010101010101010	Загрузить двоичную константу в ACCU1
L DW#16#A0F0 BCFD	Загрузить шестнадцатиричную константу в ACCU1
L 'END'	Загрузить ASCII символы в ACCU1
L T#500 ms	Загрузить константу времени в ACCU1
L P#10.0	Загрузить внутризонную константу в ACCU1
L P#E20.6	Загрузить межзонную константу в ACCU1
L -2.5	Загрузить число вещественного типа в ACCU1
L D#1997-01-20	Загрузить дату
L DOD#13:20:33.125	Загрузить время дня
Прямая адресация	
A I 0.0	Опрос по логическому «И» входного бита 0.0
L IB 1	Загрузить входной байт 1 в ACCU1
L IW 0	Загрузить входное слово 0 в ACCU1
L ID 0	Загрузить входное двойное слово 0 в ACCU1
Косвенная адресация таймеров/счетчиков	
SP T [LW 8]	Запуск таймера; номер таймера находится в локальном слове 8
CU C [LW 10]	Запуск счетчика; номер счетчика находится в локальном слове 10
Внутризонная косвенная адресация памяти	
A I [LD 12] Example: L P#22.2 T LD 12 A I [LD 12]	Операция «И»: адрес входа находится в двойном локальном слове 12 как указатель (формат pointer)
A I [DBD 1]	Операция «И»: адрес входа находится в двойном слове данных 1 как указатель (формат pointer)
A Q [DID 12]	Операция «И»: адрес выхода находится в двойном слове данных 12 экземплярного блока данных как указатель (формат pointer) .
A Q [MD 12]	Операция «И»: адрес выхода находится в двойном меркерном слове 12 как указатель (формат pointer)
Внутризонная косвенная адресация памяти	
A I [AR1,P#12.2]	Операция «И»: адрес входа рассчитывается как сумма содержимого регистра AR1(формат pointer) и константы P#12.2

Межзонная косвенная адресация памяти		
При межзонной косвенной адресации памяти биты адресного регистра с 24 до 26 должны также содержать идентификатор области .		
Идентификатор Области	Двоичный код	16-ричное / Область значение
P	1000 0000	80 периферийная область
I	1000 0001	81 область входов
Q	1000 0010	82 область выходов
M	1000 0011	83 меркерная область
DB	1000 0100	84 область данных
DI	1000 0101	85 область экземплярных данных
L	1000 0110	86 область локальных данных
VL	1000 0111	87 область локальных данных предыдущего блока (вызывающего см. страницу В-5)
L B [AR1,P#8.0]		Загрузить байт в ACCU1: адрес рассчитывается как сумма содержимого регистра AR1(формат pointer) и константы P#8.0
A [AR1,P#32.3]		Операция «И»: адрес идентификатора адресной области рассчитывается как сумма содержимого регистра AR1(формат pointer) и константы P#32.3”
Адресация через параметры		
A Parameter		Адресация через параметры

Вычисление Указателей

Вот - 2 примера того, как вычисляются указатели:

Пример для суммы битовых адресов ≤ 7 :

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

Результат: адресован вход 18.4 (путем сложения адресов битов и байтов)

Пример для суммы битовых адресов > 7 :

```
L MD 0    Любой указатель, например P#10.5
LAR1
A I [AR1,P#10.7]
```

Результат: адресован вход 21.4 (путем сложения адресов битов и байтов с переносом)

V.5 Время выполнения команд с косвенной адресацией

Составная команда

Команды с косвенной адресацией состоят из двух частей:

Часть 1: Загрузка адреса идентификатора адресной области

Часть 2: Выполнение команды

Другими словами, Вы можете вычислить время выполнения команды с косвенной адресацией сложением времени исполнения этих двух частей.

Вычисление Времени Выполнения

Полное время выполнения рассчитывается следующим образом:

$$\begin{array}{l} \text{Время загрузки адреса} \\ + \\ \text{.Выполнение самой команды} \\ \hline = \text{Общее время выполнения команды} \end{array}$$

Время выполнения в Приложении В.6 и в последующих главах приводится для второй части команды с косвенной адресацией (то есть для фактически выполняемой команды).

После чего Вы должны прибавить время, требуемое для загрузки адреса команды к времени выполнения этой команды.

Загрузка Адреса

Время выполнения команды загрузки адреса инструкции в зависимости от различных областей показывается в следующей таблице.

Адрес находится в...	Время выполнения в мксек
Меркерная область памяти M	
Слово	0.8
Двойное слово	2.1
Блок данных DB/DX	
Слово	3.0
Двойное слово	4.1
Локальные данные L	
Слово	0.9
Двойное слово	2.2
AR1/AR2 (Внутризонная косвенная адресация памяти)	1.7
AR1/AR2 (межзонная косвенная адресация памяти)	3.2

Параметре (слово) ... для: - таймеров - счетчиков - вызовов блоков	2.1
Параметре (двойное слово) ... для: биты, байты, слова и двойные слова	4.3

В.5.1 Пример внутризонной косвенной адресации памяти

Пример: A I [DBD 12]

Шаг 1

Загрузка содержимого DBD 12

Адрес из ...	Время выполнения в мксек
Меркерная область памяти M	
Слово	0.8
Двойное слово	2.1
Блок данных DB/DX	
Слово	3.0
Двойное слово	4.1

Шаг 2

Опрос по «И» адресованного входа (см. время выполнения команды в Приложении В.6 и следующих главах).

Время выполнения команды в мкс	
Прямая адресация	Косвенная адресация
0.3 :	1.6 + Время выполнения команды A I

Полное время выполнения команды:

$$\begin{array}{r}
 1,6 \text{ мкс} \\
 + 4,1 \text{ мкс} \\
 \hline
 = 5,7 \text{ мкс}
 \end{array}$$

В.5.2 Пример внутризонной косвенной адресации через адресный регистр

Пример

A I [AR1, P#34.3]

Шаг 1

Загрузка содержимого AR1, и увеличение его на 34.3

Адрес в ...	Время выполнения в мксек
:	:
AR1/AR2 (внутризонный)	1.7
:	:

Шаг 2

Опрос по «И» адресованного входа (время выполнения находится в Приложении В.6 и следующих главах).

Время выполнения команды в мкс	
Прямая адресация	Косвенная адресация
0.3	1.6 +
:	Время выполнения команды A I

Полное время выполнения команды:

$$\begin{array}{r}
 1,7 \text{ мкс} \\
 + 1,6 \text{ мкс} \\
 \hline
 = 3,3 \text{ мкс}
 \end{array}$$

В 5.3 Пример межзонной косвенной адресации памяти

Пример:

U [AR1, P#23.1] ... с P#E 1.0 в AR1

Шаг 1

Загрузка содержимого AR1, и увеличение его на 23.1

Адрес в ...	Время выполнения в мксек
:	:
AR1/AR2 (межзонный)	3,2
:	:

Шаг 2

Опрос по «И» адресованного входа (время выполнения находится в Приложении В.6 и следующих главах).

Время выполнения команды в мкс	
Прямая адресация	Косвенная адресация
0.3 :	1.6 + Время выполнения команды A I

Полное время выполнения команды:

$$\begin{array}{r}
 3,2 \text{ мкс} \\
 + 1,6 \text{ мкс} \\
 \hline
 = 4,8 \text{ мкс}
 \end{array}$$

$$\begin{array}{r}
 \text{Total execution time} \\
 3.2 \text{ ms} \\
 + 1.6 \text{ ms} \\
 \hline
 \underline{\underline{= 4.8 \text{ ms}}}
 \end{array}$$

В.5.4 Пример адресации через параметры

Пример:

Параметр ... I 0.5 находится в списке параметров

Шаг 1

Загрузка адреса I 0.5 через параметр

Адрес в ...	Время выполнения в мксек
:	:
:	:
Параметр(двойное слово)	4,3

Шаг 2

Опрос по «И» адресованного входа (время выполнения находится в Приложении В.6 и следующих главах).

Время выполнения команды в мксек	
Прямая адресация	Косвенная адресация
0.3 :	1.6 + Время выполнения команды A I

Полное время выполнения команды:

$$\begin{array}{r}
 4,3 \text{ мкс} \\
 + 1,6 \text{ мкс} \\
 \hline
 = 5,9 \text{ мкс}
 \end{array}$$

В.6 Двоичные логические команды

Опрос статуса сигнала операнда соответствующей командой и сопряжение результата опроса с результатом логической операции по указанной логической функции.

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек							
				Прямая адресация			Косвенная адресация				
A	I/O	Операция И									
		Вход/ выход	1**/2			0.3				1.6+	
	M	Меркер	1**/2			0.6				1.7+	
	L	Бит локальных данных	2			0.9				1.8+	
	DBX/DIX	Бит данных	2			2.8				2.5+	
	[AR1,m]	I/Q/M/L/DBX/DIX (межзонная	2			-				+	
	[AR2,m]	адресация через AR1/AR2 или				-				+	
	Параметры	через параметр)				-				+	
AN	I/O	Операция И -НЕ									
		Вход/ выход	2			0.5				1.9+	
	M	Меркер				0.8				2.1+	
	L	Бит локальных данных				1.0				2.2+	
	DBX/DIX	Бит данных				3.1				2.8+	
	[AR1,m]	I/Q/M/L/DBX/DIX (межзонная	2			-				-	
	[AR2,m]	адресация через AR1/AR2 или				-				-	
	Параметры	через параметр)				-				-	
Слово состояния для: A, AN			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	Да	-	Да	Да
Команда изменяет:			-	-	-	-	-	Да	Да	Да	1
O	I/O	Операция ИЛИ									
		Вход/ выход	1**/2			0.3				1.6+	
	M	Меркер	1**/2			0.7				1.7+	
	L	Бит локальных данных	2			0.9				1.8+	
	DBX/DIX	Бит данных	2			2.9				2.5+	
	[AR1,m]	I/Q/M/L/DBX/DIX (межзонная	2			-				+	
	[AR2,m]	адресация через AR1/AR2 или				-				+	
	Параметры	через параметр)				-				+	
ON	I/O	Операция ИЛИ-НЕ									
		Вход/ выход	1**/2			0.5				1.6+	
	M	Меркер	1**/2			0.8				2.0+	
	L	Бит локальных данных	2			2.0				2.2+	
	DBX/DIX	Бит данных	2			3.1				2.8+	
	[AR1,m]	I/Q/M/L/DBX/DIX (межзонная	2			-				+	
	[AR2,m]	адресация через AR1/AR2 или				-				+	
	Параметры	через параметр)				-				+	
Слово состояния для: O, ON			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	Да	Да
Команда изменяет:			-	-	-	-	-	0	Да	Да	1

* + время для загрузки адреса команды

** Для команд непосредственной адресации

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек							
				Прямая адресация	Косвенная адресация*						
X	I/O	Исключающее ИЛИ	2	0.3	1.6+						
	M	Вход/ выход									
	L	Меркер									
	DBX/DIX	Бит локальных данных Бит данных									
	[AR1,m] [AR2,m] Параметры	I/Q/M/L/DBX/DIX (межзонная адресация через AR1/AR2 или через параметр)	2	- - -	+ + +						
XN	I/O	Исключающее ИЛИ-НЕ	2	0.5	1.9+						
	M	Вход/ выход									
	L	Меркер									
	DBX/DIX	Бит локальных данных Бит данных									
	[AR1,m] [AR2,m] Параметры	I/Q/M/L/DBX/DIX (межзонная адресация через AR1/AR2 или через параметр)	2	- - -	+ + +						
Слово состояния для: X, XN			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	Да	Да
Команда изменяет:			-	-	-	-	-	0	Да	Да	1

* + время для загрузки адреса команды

В.7 Логические команды со скобками

Сохранение битов слова состояния BR, RLO и OR и идентификаторов команд (A, AN, ...) в стеке скобок. Допускается глубина вложения – до 7 скобок.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
A(И скобка открывается	1	1.7						
AN(И-НЕ скобка открывается	1	1.7						
O(ИЛИ скобка открывается	1	1.7						
ON(ИЛИ-НЕ скобка открывается	1	1.7						
X(ИСКЛЮЧАЮЩЕЕ ИЛИ скобка открывается	1	1.7						
XN(ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ скобка открывается	1	1.7						
Слово состояния для: A(, AN(, O(, ON(, X(, XN(BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		Да	-	-	-	-	Да	-	Да	Да
Команда изменяет:		-	-	-	-	-	0	1	-	0
)		Правая скобка,извлекающая запись из стека скобок, производя сопряжение РЛО процессора и стека скобок	1	1.9						
Слово состояния для:)		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	Да	-
Команда изменяет:		Да	-	-	-	-	Да	1	Да	1

В.8 Команды И перед ИЛИ

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек							
О		Выполнение функций И перед ИЛИ.	1	0.5							
Слово состояния для: О			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	Да	-	Да	Да
Команда изменяет:			-	-	-	-	-	Да	1	-	Да

В.9 Логические команды для счетчиков и таймеров

Опрос двоичного выхода таймеров и счетчиков и сопряжение результата опроса с результатом логической операции(РЛО) в соответствии с логической функцией команды.

Команда	Адрес-ная область	Описание	Длина в словах	Время выполнения в мсек							
				Прямая адресация	Косвенная адресация*						
A	T	Опрос таймера по И	1**/2	0.9	2.1+						
	C	Опрос счетчика по И		0.6	1.8+						
	Таимер/ счетчик параметр	Опрос таймера/ счетчика по И (адресация через параметр)	2	-	+						
AN	T	Опрос таймера по И - НЕ	1**/2	1.1	2.3+						
	C	Опрос счетчика по И - НЕ		0.9	2.1+						
	Таимер/ счетчик параметр	Опрос таймера/ счетчика по И-НЕ (адресация через параметр)	2	-	+						
Слово состояния для: A, AN			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	Да	-	Да	Да
Команда изменяет:			-	-	-	-	-	Да	Да	Да	1
O	T	Опрос таймера по ИЛИ	1**/2	0.9	2.1+						
	C	Опрос счетчика по ИЛИ		0.6	1.8+						
	Таимер/ счетчик параметр	Опрос таймера/ счетчика по ИЛИ (адресация через параметр)	2	-	+						
ON	T		1**/2	1.1	2.3+						
	C			0.9	2.1+						
	Таимер/ счетчик параметр	Опрос таймера/ счетчика по ИЛИ-НЕ (адресация через параметр)	2	-	+						
X	T	Опрос таймера по ИСКЛ. ИЛИ	2	0.9	2.1+						
	C	Опрос счетчика по ИСКЛ. ИЛИ		0.6	1.8+						
	Таимер/ счетчик параметр	Опрос таймера/ счетчика по ИСКЛЮЧАЮЩЕМУ ИЛИ (адресация через параметр)	2	-	+						
XN	T	Опрос таймера по ИСКЛ. ИЛИ-НЕ	2	1.1	2.3+						
	C	Опрос счетчика по ИСКЛ. ИЛИ-НЕ		0.9	2.1+						
	Таимер/ счетчик параметр	Опрос таймера/ счетчика по ИСКЛЮЧАЮЩЕМУ ИЛИ -НЕ (адресация через параметр)	2	-	+						
Слово состояния для: O, ON, X, XN			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	Да	Да
Команда изменяет:			-	-	-	-	-	0	Да	Да	1

* + время для загрузки адреса команды

** При использовании команд непосредственной адресации

В.10 Логические команды с использованием Аккумулятора 1

Команды выполняются с содержанием ACCU1 или ACCU1-L со словом или двойным словом в соответствии с применяемой функцией. Второй операнд –это слово или двойное слово , взятое (в зависимости от команды) как константа или из ACCU2. Результат помещается в ACCU1 или ACCU1-L.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
AW		Операция И с ACCU2-L	1	0.6						
	k16	Операция И с 16-битной константой	2	0.9						
OW		Операция ИЛИ с ACCU2-L	1	0.6						
	k16	Операция ИЛИ с 16-битной константой	2	0.9						
XOW		ИСКЛЮЧАЮЩЕЕ-ИЛИ с ACCU2-L	1	0.6						
	k16	ИСКЛЮЧАЮЩЕЕ-ИЛИ с 16-битной константой	2	0.9						
AD		Операция И с ACCU2	1	2.0						
	k16	Операция И с 16-битной константой	3	2.3						
OD		Операция ИЛИ с ACCU2	1	2.0						
	k16	Операция ИЛИ с 16-битной константой	3	2.3						
XOD		ИСКЛЮЧАЮЩЕЕ-ИЛИ с ACCU2	1	2.0						
	k16	ИСКЛЮЧАЮЩЕЕ-ИЛИ с 16-битной константой	3	2.3						
Слово состояния для: AW, OW, XOW, AD, OD, XOD		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	Да	0	0	-	-	-	-	-

В.11 Логические инструкции с условными битами слова состояния

Оценка статуса определенных условий и их сопряжение с РЛО в соответствии с логической функцией.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек							
A	==0	Функция И с результатом=0 (CC 1=0) и (CC 0=0)	1	0.6							
	>0	Функция И с результатом >0 (CC 1=1) и (CC 0=0)	1	0.9							
	<0	Функция И с результатом <0 (CC 1=0) и (CC 0=1)	1	0.9							
	<>0	Функция И с результатом не равным 0 ((CC 1=0) и (CC 0=1) или (CC 1=1) и (CC 0=0))	1	0.6							
	<=0	Функция И с результатом <=0 ((CC 1=0) и (CC 0=1) или (CC 1=0) и (CC 0=0))	1	0.6							
	>=0	Функция И с результатом >=0 ((CC 1=1) и (CC 0=0) или (CC 1=0) и (CC 0=0))	1	0.6							
	AO	Функция И с результатом неопределенности (CC 1=1) и (CC 0=1)	1	0.6							
	OS	Функция И с битом OS=1	1	0.3							
	BR	Функция И с битом BR=1	1	0.3							
	OV	Функция И с битом OV=1	1	0.3							
Слово состояния для:И с условиями			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			Да	Да	Да	Да	Да	Да	-	Да	Да
Команда изменяет:			-	-	-	-	-	Да	Да	Да	1

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
AN	==0	Функция И-НЕ с результатом=0 (CC 1=0) и (CC 0=0)	1	0.6						
	>0	Функция И-НЕ с результатом >0 (CC 1=1) и (CC 0=0)	1	0.9						
	<0	Функция И-НЕ с результатом <0 (CC 1=0) и (CC 0=1)	1	0.9						
	<>0	Функция И-НЕ с результатом не равным 0 ((CC 1=0) и (CC 0=1) или (CC 1=1) и (CC 0=0))	1	0.9						
	<=0	Функция И-НЕ с результатом <=0 ((CC 1=0) и (CC 0=1) или (CC 1=0) и (CC 0=0))	1	0.3						
	>=0	Функция И-НЕ с результатом >=0 ((CC 1=1) и (CC 0=0) или (CC 1=0) и (CC 0=0))	1	0.3						
	AO	Функция И-НЕ с результатом неопределенности (CC 1=1) и (CC 0=1)	1	0.9						
	OS	Функция И-НЕ с битом OS=1	1	0.6						
	BR	Функция И-НЕ с битом BR=1	1	0.6						
	OV	Функция И-НЕ с битом OV=1	1	0.6						
Слово состояния для:И-НЕ с условиями		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		Да	Да	Да	Да	Да	Да	-	Да	Да
Команда изменяет:		-	-	-	-	-	Да	Да	Да	1

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
O	==0	Функция ИЛИ с результатом=0 (CC 1=0) и (CC 0=0)	1	0.6						
	>0	Функция ИЛИ с результатом >0 (CC 1=1) и (CC 0=0)	1	0.9						
	<0	Функция ИЛИ с результатом <0 (CC 1=0) и (CC 0=1)	1	0.9						
	<>0	Функция ИЛИ с результатом<>0 ((CC 1=0) и (CC 0=1) или (CC 1=1) и (CC 0=0))	1	0.6						
	<=0	Функция ИЛИ с результатом <=0 ((CC 1=0) и (CC 0=1) или (CC 1=0) и (CC 0=0))	1	0.6						
	>=0	Функция ИЛИ с результатом >=0 ((CC 1=1) и (CC 0=0) или (CC 1=0) и (CC 0=0))	1	0.6						
	AO	Функция ИЛИ с результатом неопределенности (CC 1=1) и (CC 0=1)	1	0.6						
	OS	Функция ИЛИ с битом OS=1	1	0.3						
	OV	Функция ИЛИ с битом OV=1	1	0.3						
	BR	Функция ИЛИ с битом BR=1	1	0.3						
Слово состояния для: ИЛИ с условиями		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		Да	Да	Да	Да	Да	-	-	Да	Да
Команда изменяет:		-	-	-	-	-	0	Да	Да	1

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек							
ON	==0	Функция ИЛИ-НЕ с результатом =0 (CC 1=0) и (CC 0=0)	1	0.6							
	>0	Функция ИЛИ-НЕ с результатом >0 (CC 1=1) и (CC 0=0)	1	0.9							
	<0	Функция ИЛИ-НЕ с результатом <0 (CC 1=0) и (CC 0=1)	1	0.9							
	<>0	Функция ИЛИ-НЕ с результатом <>0 ((CC 1=0) и (CC 0=1) или (CC 1=1) и (CC 0=0))	1	0.9							
	<=0	Функция ИЛИ-НЕ с результатом <=0 ((CC 1=0) и (CC 0=1) или (CC 1=0) и (CC 0=0))	1	0.3							
	>=0	Функция ИЛИ-НЕ с результатом >=0 ((CC 1=1) и (CC 0=0) или (CC 1=0) и (CC 0=0))	1	0.3							
	AO	Функция ИЛИ-НЕ с неопределенным результатом(CC 1=1) и (CC 0=1)	1	0.9							
	OS	Функция ИЛИ-НЕ с битом OS=1	1	0.6							
	OV	Функция ИЛИ-НЕ с битом OV=1	1	0.6							
	BR	Функция ИЛИ-НЕ с битом BR=1	1	0.6							
Слово состояния для:ON с условием			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			Да	Да	Да	Да	Да	-	-	Да	Да
Команда изменяет:			-	-	-	-	-	0	Да	Да	1

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
X	==0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с результатом =0 (CC 1=0) и (CC 0=0)	1	0.6						
	>0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с результатом >0 (CC 1=1) и (CC 0=0)	1	0.9						
	<0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с результатом <0 (CC 1=0) и (CC 0=1)	1	0.9						
	<>0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с результатом <>0 ((CC 1=0) и (CC 0=1) или (CC 1=1) и (CC 0=0))	1	0.6						
	<=0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с результатом <=0 ((CC 1=0) и (CC 0=1) или (CC 1=0) и (CC 0=0))	1	0.6						
	>=0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с результатом >=0 ((CC 1=1) и (CC 0=0) или (CC 1=0) и (CC 0=0))	1	0.6						
	AO	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с неопределенным результатом (CC 1=1) и (CC 0=1)	1	0.6						
	OS	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с битом OS=1	1	0.3						
	OV	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с битом OV=1	1	0.3						
	BR	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ с битом BR=1	1	0.3						
Слово состояния для: X с условием		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		Да	Да	Да	Да	Да	-	-	Да	Да
Команда изменяет:		-	-	-	-	-	0	Да	Да	1

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
XN	==0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с результатом =0(СС 1=0) и (СС 0=0)	1	0.6						
	>0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с результатом >0 (СС 1=1) и (СС 0=0)	1	0.9						
	<0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с результатом <0 (СС 1=0) и (СС 0=1)	1	0.9						
	<>0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с результатом<>0((СС 1=0) и (СС 0=1) или (СС 1=1) и (СС 0=0))	1	0.9						
	<=0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с результатом <=0 ((СС 1=0) и (СС 0=1) или (СС 1=0) и (СС 0=0))	1	0.3						
	>=0	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с результатом >=0 ((СС 1=1) и (СС 0=0) или (СС 1=0) и (СС 0=0))	1	0.3						
	АО	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с неопределенным результатом (СС 1=1) и (СС 0=1)	1	0.9						
	OS	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с битом OS=1	1	0.6						
	OV	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с битом OV=1	1	0.6						
	BR	Функция ИСКЛЮЧАЮЩЕЕ- ИЛИ-НЕ с битом BR=1	1	0.6						
Слово состояния для: XN с условием		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		Да	Да	Да	Да	Да	-	-	Да	Да
Команда изменяет:		-	-	-	-	-	0	Да	Да	1

В.12 Команды обработки фронта

Обнаружение фронта. Текущее состояние РЛО сравнивается с состоянием РЛО, сохраненном в " меркере памяти фронта ". Команда FP обнаруживает изменение РЛО с "0" на "1". Команда FN обнаруживает изменение РЛО с "1" на "0".

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек						
				Прямая адресация	Косвенная адресация*					
FP	I/O	Определение переднего фронта РЛО.	2	0.8	2.4+					
	M	Бит указанный в команде является вспомогательным меркером памяти фронта.		1.5	2.7+					
	L			1.6	2.7					
	DBX/DIX			4.0	3.6+					
	[AR1,m]			-	+					
	[AR2,m]			-	+					
	Параметры		-	+						
FN	I/O	Определение заднего фронта РЛО.	2	1.0	2.6+					
	M	Бит указанный в команде является вспомогательным меркером памяти фронта.		1.6	2.8+					
	L			1.7	2.8+					
	DBX/DIX			4.1	3.7+					
	[AR1,m]			-	+					
	[AR2,m]			-	+					
	Параметры		-	+						
Слово состояния для: FP, FN		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	Да	-
Команда изменяет:		-	-	-	-	-	0	Да	Да	1

* + время для загрузки адреса команды

В.13 Установка / сброс битовых адресов

Установка значения "1" или "0" битовых адресов, указанных в командах, в зависимости от РЛО. Выполнение команды может зависеть от выполнения функции MCR (Мастер Контроль Реле).

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек						
				Прямая адресация	Косвенная адресация*					
S	I/O	Установка входа/ выхода в "1" (MCR–зависит)	1**/2	0.3	2.2+					
	M	Установка меркера в "1" (MCR– зависит)	1**/2	0.5 0.8	2.9+ 2.5+					
	L	Установка локального бита "1" (MCR– зависит)	2	2.3 1.3	3.0+ 2.5+					
	DBX/DIX	Установка бита данных в "1" (MCR– зависит)	2	2.9 3.7 4.3	2.5+ 3.5+ 4.1+					
	[AR1,m] [AR2,m] Параметры	Установка I/Q/M/L/DBX/DIX (MCR–зависит) в "1". (межзонная косвенная адресация через AR1/AR2 или через параметр)	2	- - -	+ + +					
R	I/O	Сброс входа/ выхода в "0" (MCR– зависит)	1**/2	0.4	2.3+					
	M	Сброс меркера в "0" (MCR– зависит)	1**/2	0.5 0.9	3.0+ 2.6+					
	L	Сброс локального бита в "0" (MCR– зависит)	2	2.4 1.3	3.2+ 2.6+					
	DBX/DIX	Сброс бита данных в "0" (MCR– зависит)	2	3.0 3.8 4.3	2.7+ 3.6+ 4.3+					
	[AR1,m] [AR2,m] Параметры	Установка I/Q/M/L/DBX/DIX (MCR–зависит) в "0". (межзонная косвенная адресация через AR1/AR2 или через параметр)	2	- - -	+ + +					
=	I/O	Присвоение РЛО входу / выходу (MCR– зависит)	1**/2	0.3	2.2+					
	M	Assign RLO до memory marker (MCR– зависит)	1**/2	0.5 0.9	2.9+ 2.5+					
	L	Assign RLO до local data bit (MCR– зависит)	2	2.3 1.1	3.0+ 2.5+					
	DBX/DIX	Assign RLO до data bit (MCR– зависит)	2	2.6 3.8 4.4	2.3+ 3.6+ 4.3+					
	[AR1,m] [AR2,m] Параметры	Присвоение РЛО I/Q/M/L/DBX/DIX (MCR–зависит) в "0". (межзонная косвенная адресация через AR1/AR2 или через параметр)	2	-	+					
Слово состояния		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	Да	-
Команда изменяет:		-	-	-	-	-	0	Да	-	0

* время для загрузки адреса команды

** Для команд непосредственной адресации

В.14 Команды непосредственно влияющие на РЛО

Непосредственно влияют на РЛО следующие команды:

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
CLR		Установка РЛО в "0"	1	0.3						
Слово состояния для : CLR		BR A1 A0	OV OS	OR	STA	RLO	/FC			
Команда зависит от:		- - -	- -	- -	- -	- -	- -	- -	- -	
Команда изменяет:		- - -	- -	0	0	0	0			
SET		Установка РЛО в "1"	1	0.3						
Слово состояния для:SET		BR A1 A0	OV OS	OR	STA	RLO	/FC			
Команда зависит от:		- - -	- -	- -	- -	- -	- -	- -	- -	
Команда изменяет:		- - -	- -	0	1	1	0			
NOT		Инверсия РЛО	1	0.3						
Слово состояния для:NOT		BR A1 A0	OV OS	OR	STA	RLO	/FC			
Команда зависит от:		- - -	- -	Да	-	Да	-	-	-	
Команда изменяет:		- - -	- -	-	1	Да	-	-	-	
SAVE		Сохраняет РЛО в бите BR	1	0.3						
Слово состояния для:SAVE		BR A1 A0	OV OS	OR	STA	RLO	/FC			
Команда зависит от:		- - -	- -	- -	- -	Да	-	-	-	
Команда изменяет:		Да - -	- -	- -	- -	- -	- -	- -	- -	

В.15 Команды таймера

Старт или сброс таймера (адресация непосредственно или через параметр).
 Время должно быть загружено в ACCU1-L.

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек							
				Прямая адресация	Косвенная адресация*						
SP	T	Запуск таймера как «импульс» при смене РЛО с "0" на "1"	1**/2	9.2	9.7+						
	параметр Timer		2	-	-						
SE	T	Запуск таймера как «удлиненный импульс» при смене РЛО с "0" на "1"	1**/2	9.2	9.7+						
	параметр Timer		2	-	-						
SD	T	Запуск таймера как «задержка включения» при смене РЛО с "0" на "1"	1**/2	9.7	10.2+						
	параметр Timer		2	-	-						
SS	T	Запуск таймера как «задержка включения с памятью» при смене РЛО с "0" на "1"	1**/2	9.7	10.2+						
	параметр Timer		2	-	-						
SF	T	Запуск таймера как «задержка выключения» при смене РЛО с "1" на "0".	1**/2	10.0	10.5+						
	параметр Timer		2	-	-						
FR	T	Разрешение перезапуска таймера при смене РЛО с "0" на "1" (сброс меркера памяти фронта для запуска таймера)	1**/2	2.1	2.7+						
	параметр Timer		2	-	-						
R	T	Сброс таймера	1**/2	1.8	2.4+						
	параметр Timer		2	-	-						
Слово состояния для: SP, SE, SD, SS, SF, FR, R			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	Да	-
Команда изменяет:			-	-	-	-	-	0	-	-	0

* время для загрузки адреса команды

** Для команд непосредственной адресации

В.16 Команды счетчика

Установка счетчика производится через ACCU1-L или через параметр.

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек						
				Прямая адресация	Косвенная адресация*					
S	C	Предустановка счетчика при смене РЛО с "0" на "1"	1**/2	6.6	7.1+					
	Counter парам.		2	-	-					
R	C	Сброс счетчика при смене РЛО с "0" на "1"	1**/2	1.8	2.3+					
	Counter парам.		2	-	-					
CU	C	Увеличение содержимого счетчика на 1 при смене РЛО с "0" на "1"	1**/2	2.8	3.4+					
	Counter парам.		2	-	-					
CD	C	Уменьшение содержимого счетчика на 1 при смене РЛО с "0" на "1"	1**/2	3.0	3.5+					
	Counter парам.		2	-	-					
FR	C	Деблокировка счетчика при смене РЛО с "0" на "1" (сброс меркера памяти счетчика для счета вверх и вниз)	1**/2	2.2	2.7+					
	Counter парам.		2	-	-					
Слово состояния для: S, R, CU, CD, FR		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	Да	-
Команда изменяет:		-	-	-	-	-	0	-	-	0

* время для загрузки адреса команды

** Для команд непосредственной адресации

В.17 Команды загрузки

Команда загрузки заносит содержимое указанного операнда в аккумулятор ACCU1. Старое содержимое ACCU1 сохраняется в ACCU2. Слово состояния не меняется..

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек	
				Прямая адресация	Косвенная адресация*
L	IB	Загрузка ... Входного байта	1**/2	0.6	1.7+
	QB	Выходного байта	1**/2	0.6	1.7+
	PIB	Периферийного входного байта	1**/2	< 125	< 127
	MB	Меркерного байта	1**/2	0.8	1.8+
	LB	Байта локальных данных	2	1.1	2.0+
	DBB	Байта данных	2	2.8	2.6+
	DIB	Байта данных блока - экземпляра ... в ACCU1	2	2.8	2.6+
	Параметры	Загрузка IW/QW/PIW/MW/LW/DBW/DIW вACCU1 (адресация через параметр)	2	-	+
	IW	Загрузка ... Входного слова	1**/2	0.9	1.9+
	QW	Выходного слова	1**/2	0.9	1.9+
	PIW	Периферийного входного слова	1**/2	< 135	< 137
	MW	Меркерного слова	1**/2	1.1	2.1+
	LW	Слова локальных данных	2	1.3	2.3+
	DBW	Слова данных	2	3.3	3.2+
	DIW	Слова данных блока - экземпляра ... в ACCU1-L	2	3.3	3.2+
	Параметры	Загрузка IW/QW/PIW/MW/LW/DBW/DIW вACCU1 (адресация через параметр)	2	-	+
	ID	Загрузка ... Входного двойного слова	1**/2	1.1	2.1+
	QD	Выходного двойного слова	1**/2	1.1	2.1+
	PID	Периферийного входного двойного слова	1**/2	< 145	< 147
	MD	Меркерного двойного слова	1**/2	1.5	2.5+
	LD	Двойного слова локальных данных	2	1.6	2.7+
	DBD	Двойного слова данных	2	4.3	4.2+
	DID	Двойного слова данных блока - экземпляра ... в ACCU1	2	4.3	4.2+
	Параметры	Загрузка IW/QW/PIW/MW/LW/DBW/DIW вACCU1 (адресация через параметр)	2	-	+

* время для загрузки адреса команды

** Для команд непосредственной адресации

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек	
				Прямая адресация	Косвенная адресация*
L	B[AR1,m]	Загрузка (межзонная адресация) Байт	2	-	40.1+
	B[AR2,m]				40.1+
	W[AR1,m]	Слово	2	-	45.6+
	W[AR2,m]				45.6+
	D[AR1,m]	Double слово	2	-	57.4+
D[AR2,m]	... in ACCU1	57.4+			
	k8	Загрузка ... 8–битную константу в ACCU1–LL	1	0.6	-
	k16	16– битную константу в ACCU1–L	2	0.6	-
	L#k32	32– битную константу в ACCU1	3	0.8	-
	Параметры	Загрузить константу в ACCU1 (адресация через параметр)	2	-	+
	2#n	Загрузка 16– битной двоичной константы в ACCU1–L	2	0.6	-
		Загрузка 32– битной двоичной константы в ACCU1	3	0.7	-
	16#p	Загрузка 16– битной 16-тиричной константы в ACCU1–L	2	0.6	-
		Загрузка 32– битной 16-тиричной константы в ACCU1	3	0.7	-
	'xx'	Загрузка 2 символов	2	0.7	-
	'xxxx'	Загрузка 4 символов	3	0.88	-
	D# Дата	Загрузка IEC даты (в BCD-коде)	2	0.8	-
	S5T# значение времени	Загрузка S5 –таймера константы (16 бит)	2	0.8	-
	TOD# значение времени	Загрузка 32– битной константы времени(IEC время дня)	3	0.88	-
	T# значение времени	Загрузка 16– битной константы времени	2	0.88	-
	S# значение счетчика	Загрузка 16– битной константы счетчика	2	0.88	-
		Загрузка 32– битной константы счетчика	3	0.88	-
	P# bit pointer	Загрузка бита в формате указателя	3	0.88	-
	L# integer	Загрузка 32– битной целой константы	3	0.88	-
	q	Загрузка действительного числа	3	0.88	-

+ Время на загрузку адреса

В.18 Команды загрузки для таймеров и счетчиков

Загрузка значения времени или счета в ACCU1. Содержимое ACCU1 сохраняется в ACCU2. Биты слова состояния не меняются и не влияют.

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек	
				Прямая адресация	Косвенная адресация *
L	T	Загрузка значения времени	1**/2	1.7	2.1+
	Timer параметр	Загрузка значения времени Загрузка time value (адресация через параметр)	2	-	+
	C	Загрузка значения счетчика	1**/2	1.5	2.1+
	Counter параметр	Загрузка значения счетчика (адресация через параметр)	2	-	+
LD	T	Загрузка значения времени (в двоично-десятичном коде)	1**/2	5.4	5.9+
	Timer параметр	Загрузка значения времени (в двоично-десятичном коде, адресация через параметр)	2	-	+
	C	Загрузка значения счетчика (в двоично-десятичном коде)	1**/2	4.9	5.4+
	Counter параметр	Загрузка значения счетчика (адресация через параметр)	2	-	+

* +время для загрузки адреса команды

** Для команд непосредственной адресации

В.19 Команды передачи

Команда производит передачу содержимого ACCU1 по указанному адресу. Слово состояния не изменяется и не влияет. Помните, что некоторые команды передачи могут зависеть от МКР (мастер контроль реле).

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек	
				Прямая адресация	Косвенная адресация*
Т	IB	Передача содержимого ACCU1-LL в... Входной байт (MCR-зависит)	1**/2	0.3	1.2+
				1.4	1.6+
	QB	Выходной байт (MCR-зависит)	1**/2	0.3	1.2+
				1.4	1.6+
PQB	Периферийный выходной байт (MCR-зависит)	1***/2	< 125 <126	< 127 < 128	
	MB	Меркерный байт (MCR-зависит)	1**/2	0.4	1.3+
				1.5	1.7+
	LB	Байт локальных данных (MCR-зависит)	2	0.6	1.5+
				1.8	2.0+
	DBB	Байт данных (MCR-зависит)	2	2.5	2.3+
DIB	Байт данных блока экземпляра (MCR-зависит)	2	3.0	2.8+	
				2.5	2.3+
				3.0	2.8+
	V[AR1,m] V[AR2,m] Параметры	Передача содержимого ACCU1-LL в IB/QB/PQB/MB/LB/DBB/DIB (адресация через AR1, AR2 (межзонная адресация) или через параметр)	2	-	+

* +время для загрузки адреса команды

** Для команд непосредственной адресации

*** Прямая адресация к PQB от 0 до 255

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек	
				Прямая адресация	Косвенная адресация*
T	IW	Передача содержимого ACCU1-L в ... Входное слово (MCR–зависит)	1**/2	0.5 1.5	1.5+ 1.8+
	QW	Выходное слово (MCR–зависит)	1**/2	0.5 1.5	1.5+ 1.8+
	PQW	Периферийное выходное слово (MCR–зависит)	1***/2	< 135 < 136	< 137 < 138
	MW	Меркерное слово (MCR–зависит)	1**/2	0.8 1.8	1.7+ 2.1+
	LW	Слово локальных данных (MCR–зависит)	2	0.9 2.0	1.8+ 2.3+
	DBW	Слово данных (MCR–зависит)	2	3.0 3.5	2.9+ 3.4+
	DIW	Слово данных блока-экземпляра (MCR–зависит)	2	3.0 3.5	2.9+ 3.4+
	D[AR1,m] D[AR2,m] Параметры	Передача содержимого ACCU1-LL в IW/QW/PQW/MW/LW/DBW/DIW (адресация через AR1, AR2 (межзонная адресация) или через параметр).	2	-	+
	ID	Передача содержимого ACCU1 в ... Входное двойное слово (MCR–зависит)	1**/2	0.8 1.8	1.7+ 2.1+
	QD	Выходноедвойное слово (MCR–зависит)	1**/2	0.8 1.8	1.7+ 2.1+
	PQD	Периферийное выходное двойное слово(MCR–зависит)	2	< 150 < 151	< 152 < 153
	MD	Меркерное двойное слово (MCR–зависит)	1**/2	1.3 2.3	2.2+ 2.6+
	LD	Двойное слово локальных данных (MCR–зависит)	2	1.4 2.6	3.0+ 3.5+
	DBD	Двойное слово данных (MCR–зависит)	2	4.1 4.6	4.0+ 4.5+
	DID	Двойное слово данных блока-экземпляра (MCR–зависит)	2	4.1 4.6	4.0+ 4.5+
	D[AR1,m] D[AR2,m] Параметры	Передача содержимого ACCU1-LL в ID/QD/PQD/MD/LD/DBD/DID (адресация через AR1, AR2 (межзонная адресация) или через параметр).	2	-	+

* +время для загрузки адреса команды

** Для команд непосредственной адресации

*** Прямая адресация к PQW от 0 до 254

В.20 Команды загрузки и передачи для адресного регистра

Загрузка двойного слова из памяти или регистра в AR1 или AR2 или передача двойного слова из AR1 или AR2 в область памяти или регистр. Слово состояния не изменяется и не влияет.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек
LAR1	-	Загрузка содержимого из ... ACCU1	1	0.3
	AR2	Адресного регистра 2	1	0.3
	DBD	Двойного слова данных	2	3.8
	DID	Двойного слова данных блока экземпляра	2	3.8
	m	32-битной константы как указателя	3	0.5
	LD	Двойного слова локальных данных	2	1.5
	MD	Двойного слова меркеров ... в AR1	2	1.4
LAR2	-	Загрузка содержимого из ... ACCU1	1	0.3
	DBD	Двойного слова данных	2	3.8
	DID	Двойного слова данных блока экземпляра	2	3.8
	m	32-битной константы как указателя	3	0.5
	LD	Двойного слова локальных данных	2	1.5
	MD	Двойного слова меркеров ... в AR2	2	1.4
TAR1	-	Передача содержимого AR1 в ... ACCU1	1	0.7
	AR2	Адресный регистр 2	1	0.3
	DBD	Двойное слово данных	2	4.3
	DID	Двойное слово данных блока экземпляра	2	4.3
	LD	Двойное слово локальных данных	2	1.6
	MD	Двойное слово меркеров	2	1.5
TAR2	-	Передача содержимого AR2 в ... ACCU1	1	0.7
	DBD	Двойное слово данных	2	4.3
	DID	Двойное слово данных блока экземпляра	2	4.3
	LD	Двойное слово локальных данных	2	1.6
	MD	Двойное слово меркеров	2	1.5
CAR		Обмен содержимым AR1 и AR2	1	0.5

В.21 Команды загрузки и передачи для слова состояния

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
L	STW	Загрузить слово состояния* в ACCU1		1.5						
Слово состояния для: L STW		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		Да	Да	Да	Да	Да	0	0	Да	0
Команда изменяет:		-	-	-	-	-	-	-	-	-
T	STW	Передать содержимое ACCU1 (биты с 0 до 8) в слово состояния*		1.4						
Слово состояния для: T STW		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		Да	Да	Да	Да	Да	0	0	Да	0

* Структура слова состояния

В.22 Команды загрузки для номеров и длин блоков данных

Загрузка номеров / длин блоков данных в ACCU1. Содержимое ACCU1 предварительно сохраняется в ACCU2 . Слово состояния не меняется.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек
L	DBNO	Загрузка номера блока данных	1	3.3
L	DINO	Загрузка номера экземплярного блока данных	1	3.3
L	DBLG	Загрузка длины блока данных в байт	1	0.6
L	DILG	Загрузка длины экземплярного блока данных в байт	1	0.6

В.23 Математические команды для целых чисел (16 бит)

Результат математических операций помещается в ACCU1-L.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек							
+I	-	Сумма 2 целых чисел (16 бит) (ACCU1-L)=(ACCU1-L)+ (ACCU2-L)	1	1.5							
-I	-	Разность 2 целых чисел (16 бит) (ACCU1-L)=(ACCU2-L)- (ACCU1-L)	1	1.6							
·I	-	Умножение 2 целых чисел (16 бит) (ACCU1-L)=(ACCU2-L)· (ACCU1-L)	1	2.4							
/I	-	Деление 2 целых чисел (16 бит) (ACCU1-L)=(ACCU2-L): (ACCU1-L) Остаток от деления ACCU1-H.	1	3.4							
Слово состояния для: +I -I ·I /I											
			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	-	-
Команда изменяет:			-	Да	Да	Да	Да	-	-	-	-

В.24 Математические команды для целых чисел (32-бита)

Результат математических операций помещается в ACCU1.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек							
+D	-	Сумма 2 целых чисел (32 бита) (ACCU1)=(ACCU2)+(ACCU1)	1	2.0							
-D	-	Разность 2 целых чисел (32 бита) (ACCU1)=(ACCU2)-(ACCU1)	1	2.7							
D	-	Умножение 2 целых чисел (32 бита) (ACCU1)=(ACCU2)(ACCU1)	1	9.9							
/D	-	Деление 2 целых чисел (32 бита) (ACCU1)=(ACCU2):(ACCU1)	1	10.8							
MOD	-	Деление 2 целых чисел (32 бита) и загрузка остатка от деления в ACCU1: (ACCU1)=остаток от деления [(ACCU2):(ACCU1)]	1	11.3							
Слово состояния для: +D, -D,*D, /D, MOD			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	-	-
Команда изменяет:			-	Да	Да	Да	Да	-	-	-	-

В.25 Математические команды с действительными числами (32 Бита)

Результат математических операций помещается в ACCU1.

Время выполнения команд зависит от величины значений чисел

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
+R	-	Сложение двух действительных чисел (32 бита) $(ACCU1)=(ACCU2)+(ACCU1)$	1	< 35						
-R	-	Вычитание двух действительных чисел (32 бита) $(ACCU1)=(ACCU2)-(ACCU1)$	1	< 35						
·R	-	Умножение двух действительных чисел (32 бита) $(ACCU1)=(ACCU2)·(ACCU1)$	1	< 35						
/R	-	Деление двух действительных чисел (32 бита) $(ACCU1)=(ACCU2):(ACCU1)$	1	< 40						
Слово состояния для: +R, -R,·R, /R										
		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	Да	Да	Да	Да	-	-	-	-
NEGR	-	Получение противоположного числа из ACCU1	1	0.3						
ABS	-	Получение модуля числа из ACCU1	1	0.3						
Слово состояния для: NEGR, ABS										
		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	-	-	-	-	-	-	-	-

В.26 Сложение с константой

Сложение целой константы с ACCU1. Слово состояния не изменяется и не влияет.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек
+	i8	Сложение с 8–битной целой константой	1	0.3
+	i16	Сложение с 16–битной целой константой	2	0.3
+	i32	Сложение с 32–битной целой константой	3	0.6

V.27 Сложение с адресным регистром

Сложение целого числа (16 бит) с содержимым адресных регистров. Слагаемое указано в команде или находится в ACCU 1–L. Слово состояния не изменяется и не влияет.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек
+AR1		Сложение ACCU1–L и AR1	1	0.3
+AR1	m	Сложение константы типа указатель и AR1	2	0.3
+AR2		Сложение ACCU1–L и AR2	1	0.3
+AR2	m	Сложение константы типа указатель и AR2	2	0.3

V.28 Команды сравнения целых чисел (16- бит)

Сравнение целых чисел (16 бит) ACCU1–L и ACCU2–L. РЛО=1 если условие сравнения выполнено.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
==I		ACCU2–L=ACCU1–L	1	1.4						
<>I		ACCU2–L \neq ACCU1–L	1	1.5						
<I		ACCU2–L<ACCU1–L	1	1.5						
<=I		ACCU2–L<=ACCU1–L	1	1.4						
>I		ACCU2–L>ACCU1–L	1	1.5						
>=I		ACCU2–L>=ACCU1–L	1	1.4						
Слово состояния для: ==I, <>I, <I, <=I, >I, >=I		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	Да	Да	0	-	0	Да	Да	1

В.29 Команды сравнения целых чисел (32- бит)

Сравнение целых чисел (16 бит) ACCU1 и ACCU2. РЛО=1 если условие сравнения выполнено.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
==D		ACCU2=ACCU1	1	2.0						
<>D		ACCU2<>ACCU1	1	2.0						
<D		ACCU2<ACCU1	1	2.0						
<=D		ACCU2<=ACCU1	1	2.0						
>D		ACCU2>ACCU1	1	2.0						
>=D		ACCU2>=ACCU1	1	2.0						
Слово состояния для: ==D,< >D, <D, <=D, >D, >=D		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	Да	Да	0	-	0	Да	Да	1

В.30 Команды сравнения 32- битных чисел с плавающей точкой

Сравнение 32- битных с плавающей точкой в ACCU1 и ACCU2. РЛО=1 если условие сравнения выполнено . Время выполнения зависит от сравниваемых величин.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
==R		ACCU2=ACCU1	1	< 45						
<>R		ACCU2<>ACCU1	1	< 45						
<R		ACCU2<ACCU1	1	< 45						
<=R		ACCU2<=ACCU1	1	< 45						
>R		ACCU2>ACCU1	1	< 45						
>=R		ACCU2>=ACCU1	1	< 45						
Слово состояния для: ==R,< >R, <R, <=R, >R, >=R		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	Да	Да	Да	Да	0	Да	Да	1

В.31 Команды сдвига

Сдвиг содержимого ACCU1 или ACCU1-L на указанное число позиций влево или вправо. Если идентификатор адресной области не определен, сдвиг производится в соответствии с содержимым ACCU2-LL. Освобождающиеся позиции заполняются нулями или знаком. Последний сдвигаемый бит сохраняется в бите условного кода CC 1.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
SLW		Сдвиг содержимого ACCU1-L влево	1	2.0						
SLW	0 до 15			0.7						
SLD		Сдвиг содержимого ACCU1 влево	1	3.1						
SLD	0 до 32			3.1						
SRW		Сдвиг содержимого ACCU1-L вправо	1	2.0						
SRW	0 до 15			0.7						
SRD		Сдвиг содержимого ACCU1 вправо	1	3.1						
SRD	0 до 32			3.1						
SSI		Сдвиг содержимого ACCU1-L вправо со знаком	1	1.8						
SSI	0 до 15			0.7						
SSD		Сдвиг содержимого ACCU1 вправо со знаком	1	3.1						
SSD	0 до 32			3.2						
Слово состояния для SLW, SLD, SRW, SRD, SSI, SSD		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	Да	Да	Да	-	-	-	-	-

В.32 Команды кругового сдвига

Круговой сдвиг содержимого ACCU1 влево или вправо на указанное число позиций. Если адресная область не определена, то сдвиг будет произведен на число позиций, указанное в ACCU2-LL.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек							
RLD		Сдвиг содержимого ACCU1 влево	1	3.3							
RLD	0 до 32			3.4							
RRD		Сдвиг содержимого ACCU1 вправо	1	3.5							
RRD	0 до 32			3.5							
Слово состояния для RLD, RRD			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	-	-
Команда изменяет:			-	Да	Да	Да	-	-	-	-	-
RLDA		Сдвиг содержимого ACCU1 на одну позицию влево через условный бит CC 1		1.9							
RRDA		Сдвиг содержимого ACCU1 на одну позицию вправо через условный бит CC 1		1.9							
Слово состояния для RLDA, RRDA			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	-	-
Команда изменяет:			-	Да	0	0	-	-	-	-	-

В.33 Команды перемещения, увеличения и уменьшения содержимого аккумуляторов

Содержимое слова состояния не влияет и не изменяется.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек
CAW		Изменение порядка байтов в ACCU1-LL, LH становятся LH, LL.	1	0.3
CAD		Изменение порядка байтов в ACCU1 LL, LH, HL, HH становятся HH, HL, LH, LL.	1	0.6
TAK		Обмен содержимым ACCU1 и ACCU2	1	0.8
PUSH		Содержимое ACCU1 передается в ACCU2	1	0.3
POP		Содержимое ACCU2 передается в ACCU1	1	0.3
INC	0 до 255	Увеличение ACCU1-LL	1	0.3
DEC	0 до 255	Уменьшение ACCU1-LL	1	0.3

В.34 Нулевые команды и программирования изображений

Слово состояния не изменяется и не влияет.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек
BLD	0 до 255	Команда вызова изображения; Трактуются контроллером как нулевая команда.	1	0.3
NOP	0 1	Нулевые операции	1	0.3 0.3

В.35 Команды преобразования форматов

Результаты помещаются в ACCU1.

Время исполнения команд преобразования действительных чисел зависит от преобразуемых значений.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
BTI	-	Преобразует содержимое ACCU1 из BCD в целое (16 битовое) (BCD в Int)	1	4.7						
BTD	-	Преобразует содержимое ACCU1 из BCD в двойное целое (32 битовое) (BCD в Dint)	1	11.5						
DTR	-	Преобразует содержимое ACCU1 из двойное integer (32 бита) в число с плавающей точкой (32 бита) (Dint в Real)	1	< 15						
ITD	-	Преобразует содержимое ACCU1 из целого (16 бит) в двойное целое (32 бита) (Int в Dint)	1	0.1						
Слово состояния для BTI, BTD,		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	-	-	-	-	-	-	-	-
ITB	-	Преобразует содержимое ACCU1 из целого (16 бит) в BCD (Int в BCD)	1	5.1						
DTB	-	Преобразует содержимое ACCU1 из двойного целого (32 бит) в BCD (DInt в BCD)	1	11.8						
Слово состояния для ITB, DTB		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	-	-	Да	Да	-	-	-	-
RND	-	Преобразует число с плавающей точкой в двойное целое (32 бита).	1	< 20						
RND-	-	Преобразует число с плавающей точкой в двойное целое (32 бита). Округление до ближайшего меньшего.	1	< 20						
RND+	-	Преобразует число с плавающей точкой в двойное целое (32 бита). Округление до ближайшего большего.	1	< 20						
TRUNC	-	Преобразует число с плавающей точкой в двойное целое (32 бита). Дробная часть числа отбрасывается.	1	< 20						
Слово состояния для RND, RND-, RND+, TRUNC		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	-	-	Да	Да	-	-	-	-

В.36 Образование дополнений

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
INVI		Формирует первое дополнение ACCU1-L	1	0.3						
INVD		Формирует первое дополнение ACCU1	1	0.3						
Слово состояния для INVI, INVD		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	-	-	-	-	-	-	-	-
NEGI		Формирует дополнение до двух ACCU1-L (16-битовое целое)	1	1.5						
NEGD		Формирует дополнение до двух ACCU1 (32-битовое целое)	1	2.0						
Слово состояния для NEGI, NEGD, NEGR		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	Да	Да	Да	Да	-	-	-	-

В.37 Команды вызова блоков

Команда	Адресная область	Описание	Длина в Словах	Время выполнения в мксек							
				Прямая адресация	Косвенная адресация*						
CALL	FC	Безусловный вызов функции	1**/2	7.7	-						
	SFC	Безусловный вызов системной функции	2	См. Приложение С для времени выполнения							
UC	FC	Безусловный вызов блока без параметров	1**/2	5.3	6.1+						
Слово состояния для CALL, UC			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	-	-
Команда изменяет:			-	-	-	-	0	0	1	-	0
CC	FC	Условный вызов блока без параметров	1**/2	5.3	6.1+						
Слово состояния для CC			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	Да	-
Команда изменяет:			-	-	-	-	0	0	1	-	0
OPN	DB	Открытие блока данных	2	1.5	2.6+						
	DB парам.	Открытие блока данных (адресованный через параметр)	2	-	-						
	DI	Открытие экземплярного блока данных	2	1.5	2.6+						
Слово состояния для OPN			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	-	-
Команда изменяет:			-	-	-	-	-	-	-	-	-

* +время для загрузки адреса команды

** Для команд непосредственной адресации

В.38 Команды окончания блока

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек						
BE		Конец блока	1	2.8						
BEU		Безусловный конец блока	1	-						
Слово состояния для BE, BEU		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	-	-	0	0	1	-	0	
ВЕС		Условный конец блока при РЛО="1"				3.2				
Слово состояния для ВЕС		BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:		-	-	-	-	-	-	-	-	-
Команда изменяет:		-	-	-	-	Да	0	1	1	0

В.39 Обмен текущих регистров блоков данных

Обмен двух текущих регистров блоков данных. Открытый блок данных становится открытым экземплярным блоком данных и наоборот(обмен содержимым регистров DB и DI. Слово состояния не изменяется и не влияет.

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек
CDB		Обмен двух текущих регистров блоков данных	1	0.4

В.40 Команды перехода

Команды условного перехода. Дистанция перехода при 8–битовой адресации находится между (-128 и +127). Дистанция перехода при 16–битовой адресации находится между (-32768 до -129) или (+128 до +32767).

Примечание

В случае CPU 614 программирование метки перехода всегда связано с началом формирования логической цепочки. Метки перехода не должны вставляться внутрь формирования логической цепочки.

Команда	Адресная область	Описание	Длина в словах				Время выполнения в мксек				
JU	LABEL	Безусловный переход на метку	1*/2				1.8				
Слово состояния для JU			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	-	-
Команда изменяет:			-	-	-	-	-	-	-	-	-
JC	LABEL	Условный переход на метку если РЛО="1"	1/2				1.5				
JCN	LABEL	Условный переход на метку если РЛО="0"	2				1.6				
Слово состояния для JC, JCN			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	Да	-
Команда изменяет:			-	-	-	-	-	0	1	1	0
JCB	LABEL	Условный переход на метку если РЛО="1". Сохранение РЛО в бите BR .	2				1.8				
JNB	LABEL	Условный переход на метку если РЛО="0". Сохранение РЛО в бите BR .	2				1.8				
Слово состояния для JCB, JNB			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	Да	-
Команда изменяет:			Да	-	-	-	-	0	1	1	0
JBI	LABEL	Условный переход на метку если BR="1"	2				1.5				
JNBI	LABEL	Условный переход на метку если BR="0"	2				1.5				
Слово состояния для JBI, JNBI			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			Да	-	-	-	-	-	-	-	-
Команда изменяет:			-	-	-	-	-	0	1	-	0

- 1 слово для перехода на -128 and +127 команд

Команда	Адресная область	Описание	Длина в словах	Время выполнения в мксек							
JO	LABEL	Условный переход на метку если (OV="1")	1*/2	1.5							
Слово состояния для JO			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	Да	-	-	-	-	-
Команда изменяет:			-	-	-	-	-	-	-	-	-
JOS	LABEL	Условный переход на метку если (OS="1")	2	1.6							
Слово состояния для JOS			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	Да	-	-	-	-
Команда изменяет:			-	-	-	-	0	-	-	-	-
JUO	LABEL	Условный переход на метку если "недопустимая команда" (CC 1=1 and CC 0=1)	2	1.8							
JZ	LABEL	Условный переход на метку если результат=0 (CC 1=0 and CC 0=0)	1*/2	1.7							
JP	LABEL	Условный переход на метку если результат >0 (CC 1=1 and CC 0=0)	1*/2	1.8							
JM	LABEL	Условный переход на метку если результат <0 (CC 1=0 and CC 0=1)	1*/2	1.8							
JN	LABEL	Условный переход на метку если результат отличен от 0 (CC 1=1 and CC 0=0) or (CC 1=0) and (CC 0=1)	1*/2	1.8							
JMZ	LABEL	Условный переход на метку если результат =0 (CC 1=0 and CC 0=1) or (CC 1=0 and CC 0=0)	2	1.5							
JPZ	LABEL	Условный переход на метку если результат > или= 0 (CC 1=1 and CC 0=0) or (CC 1=0) and (CC 0=0)	2	1.6							
Слово состояния для JUO, JZ, JP, JM, JN, JMZ, JPZ			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	Да	Да	-	-	-	-	-	-
Команда изменяет:			-	-	-	-	-	-	-	-	-
JL	LABEL	Распределенный переход За этой командой следуют список команд переходов. Адресный идентификатор дает метку перехода для команд из списка. ACCU1-L содержит строки, команда перехода которой должна выполняться.	2	2.7							
LOOP	LABEL	Уменьшение ACCU1-L на 1 и переходна метку пока ACCU1-L не равен 0 (программирование цикла)	2	1.6							
Слово состояния для JL, LOOP			BR	A1	A0	OV	OS	OR	STA	RLO	/FC
Команда зависит от:			-	-	-	-	-	-	-	-	-
Команда изменяет:			-	-	-	-	-	-	-	-	-

* 1 слово для перехода на -128 and +127 команд

В.41 Команды для функции мастер контроля реле (MCR)

MCR заносит значение "0" или оставляет содержимое ячеек памяти неизменным.

MCR=0 соответствует деактивации MCR функции; команда "Т" передает "0" по указанным адресам; команды "S"/"R" не изменяют содержимого ячеек памяти.

MCR=1 соответствует активации MCR функции.

Команда	Адресная область	Описание	Длина в словах					Время выполнения в мксек			
MCR(Открыть MCR зону. Сохранение РЛО в MCR стэке.	1					1.7			
Слово состояния для MCR(BR	A1	A0	OV	OS	OR	STA	RLO	/FC	
Команда зависит от:		-	-	-	-	-	-	-	Да	-	
Команда изменяет:		-	-	-	-	-	0	1	-	0	
)MCR		Закреть MCR зону. Извлечение РЛО из MCR- стэка.	1					1.6			
Слово состояния для)MCR		BR	A1	A0	OV	OS	OR	STA	RLO	/FC	
Команда зависит от:		-	-	-	-	-	-	-	-	-	
Команда изменяет:		-	-	-	-	-	0	1	-	0	
MCR A		Активировать MCR-функцию	1					0.3			
MCR D		Деактивировать MCR-функцию	1					0.3			
Слово состояния для MCR A, MCR D		BR	A1	A0	OV	OS	OR	STA	RLO	/FC	
Команда зависит от:		-	-	-	-	-	-	-	-	-	
Команда изменяет:		-	-	-	-	-	-	-	-	-	

Время выполнения SFC и SFB

C

CPU в IM 151/CPU предоставляет пользователю различные системные функции и системные функциональные блоки для выполнения программы и диагностики. Вы можете вызывать эти системные блоки в вашей пользовательской программе, используя номер SFC(SFB).

Для детального изучения всех системных функций и системных функциональных блоков обратитесь к руководству **Системные и Стандартные Функции STEP7**. В нем подробно описано - как вызвать системные блоки и задать параметры.

C.1 Системные Функции (SFC)

SFC No.	Название	Описание	Время выполнения в ms
0	SET_CLK	Установка встроенных часов	295
1	READ_CLK	Чтение встроенных часов	193
2	SET_RTM	Установка счетчика рабочего времени	62
3	CTRL_RTM	Запуск счетчика рабочего времени	57
4	READ_RTM	Чтение счетчика рабочего времени	86
5	GADR_LGC	Чтение логического адреса канала x на модуле слота y .	148
6	RD_SINFO	Чтение стартовой информации текущего OB.	116 +3 на байт
7	DP_PRAL	Запуск аппаратного прерывания из программы CPU - DP слэйва на DP мастере.	228
14	DPRD_DAT	Чтение непротиворечивых пользовательских данных	141 +3 на байт
15	DPWR_DAT	Запись непротиворечивых пользовательских данных	143
17	ALARM_SQ	Генерирование квитуемых сообщений, связанных с блоками	252
18	ALARM_S	Генерирование неквитуемых сообщений, связанных с блоками	248
19	ALARM_SC	Определение состояния квитирования последнего пришедшего сообщения ALARM_SQ	115
20	BLKMOV	Копирование переменных любого типа Замечание: Вы можете использовать SFC20 для процессно-зависимых DBs, т.е "Unlinked" окно свойств объекта не активировано. В противном случае Вы получите код ошибки W#16#8092 (блок не связан с процессом) в параметре RET_VAL.	84 +1.6 на байт

SFC No.	Название	Описание	Время выполнения в ms
21	FILL	Инициализация области памяти	84 +2.5 на байт
22	CREATE_DB	Создание блока данных	126 +5 на байт
23	DEL_DB	Удаление блока данных	102
24	TEST_DB	Тест блока данных	110
28	SET_TINT	Установка прерывания по времени	212
29	CAN_TINT	Отмена прерывания по времени	64
30	ACT_TINT	Активация прерывания по времени	68
31	QRY_TINT	Опрос прерывания по времени	79
32	SRT_DINT	Запуск прерывания с задержкой	89
33	CAN_DINT	Отмена прерывания с задержкой	78
34	QRY_DINT	Опрос прерывания с задержкой	85
36	MSK_FLT	Маскирование синхронных ошибок	118
37	DMSK_FLT	Демаскирование синхронных ошибок	131
38	READ_ERR	Чтение и удаление ошибок программирования и доступа.	128
39	DIS_IRT	Блокирование новых прерываний	284
40	EN_IRT	Деблокирование новых прерываний	702
41	DIS_AIRT	Задержка обработки прерываний	48
42	EN_AIRT	Деблокирование обработки прерываний	76
43	RE_TRIGR	Повторный запуск контроля времени цикла	95
44	REPL_VAL	Передача заменяющего значения в аккумулятор 1 на уровне, вызвавшем ошибку.	51
46	STP	Перевод CPU в состояние STOP	Нет данных
47	WAIT	Задержка выполнения программы пользователя	861
49	LGC_GADR	Выявление слота модуля, соответствующего логическому адресу	281
50	RD_LGADR	Выявление всех логических адресов модуля .	252
51	RDSYSST	Считывание информации списка состояний системы SFC 51 не может быть прерван.	280 +8 на байт
52	WR_USMSG	Запись специальной диагностической информации в диагностический буфер	964
54	RD_DPARM	Чтение предустановленных динамических параметров для модуля .	2390
55	WR_PARM	Запись динамических параметров для модуля (пока не используется, т.к. все параметры модулей –статические)	2500
56	WR_DPARM	Запись предустановленных динамических параметров для модуля (пока не используется, т.к. параметры модулей –статические)	2520
57	PARM_MOD	Назначение параметров модуля	2520

SFC No.	Название	Описание	Время выполнения в ms
58	WR_REC	Запись данных в спец. модуль (сейчас не используется, т.к. нет модулей, в которые могут записываться пользовательские записи)	2200
59	RD_REC	Чтение данных из модуля (сейчас считываются только диагностические записи 0 и 1)	352
64	TIME_TICK	Вывод системного времени с точностью до 10 ms	50
81	UBLKMOV	Копирование переменных без прерывания	140

C.2 Системные функциональные блоки (SFB)

SFB No.	Название	Описание	Время выполнения в ms
IEC Счетчики			
0	CTU	Счет вверх. Счетчик увеличивается на 1 после каждого положительного фронта импульса.	70
1	CTD	Счет вниз. Счетчик уменьшается на 1 после каждого положительного фронта импульса.	69
2	CTUD	Счет вверх и счет вниз.	85
IEC Таймеры			
3	TP	Импульс длительностью PT.	90
4	TON	Задержка включения длительностью PT.	91
5	TOF	Задержка выключения длительностью PT.	91
Реализация шагового процессора			
32	DRUM	Реализация шагового процессора с максимальным числом шагов - 16.	302

Diese Seite ist eine **Vakat**-Seite, die an das Ende eines Kapitels mit ungerader Seitennummer angehängt wird.

Особенности IM 151/CPU

D

В этой главе, Вы найдете наиболее важные отличия отдельных CPUs семейства SIMATIC S7-300.

Мы также покажем Вам, как переписать программы, которые Вы написали для S7 - 300 - CPUs для IM 151/CPU.

Краткий обзор Главы

Секция	Содержание	Страница
D.1	Различия отдельных S7-300 CPUs	D-2
D.2	Перенос пользовательской программы	D-3

Дополнительная информация

Вы можете найти дополнительную информацию относительно того, как создавать и структурировать программы в руководстве по *STEP 7* и встроенной системе помощи.

D.1 Различия отдельных S7-300 CPU

Следующая таблица показывает наиболее важные различия программирования между двумя CPU семейства SIMATIC S7-300 и IM 151/CPU.

Таблица D-1 Различия отдельных S7-300 CPU

Характеристика	CPU 314	CPU-315-2DP	IM 151/CPU
Часы реального времени	Аппаратные	Аппаратные	Программные
Сохранение памяти	Да, с батареейкой	Да, с батареейкой	Нет
Модуль расширения памяти	Модуль расширения памяти	Модуль расширения памяти	MMC
Установка адреса PROFIBUS	Аппаратная конфигурация	Аппаратная конфигурация	Аппаратная конфигурация, должна совпадать с установкой адресов
Скорость обмена с программатором и панелью оператора	187.5 кбод(MPI)	187.5 кбод(MPI) 12 Мбод (DP)	12 Мбод (DP)
Интерфейс	Активный	Активный	Пассивный
Коммуникации:			
- С программатором и панелью оператора	Да	Да	Да
-Глобальные данные	Да	Да	Нет
-S7основные коммуникации	Да	Да	Да(сервер)
-S7 коммуникации	Да (сервер)	Да (сервер)	Да (сервер)
-Прямая связь	Нет	Да	Нет
Возможность работы с DP	Автономно	Как DP- мастер Как DP- слэйв Автономно	Как DP- слэйв Автономно
Адресация	По установочному месту	Гибкая	Гибкая
Время реакции	1-1.5 мс	1-1.5 мс	Менее 20 мс
Вставка / удаление блоков во время работы	Нет	Нет	Да

D.2 Перенос пользовательской программы

Введение

Под переносом подразумевается возможность использования на удаленном устройстве программы, которая предварительно использовалась централизованно на мастере. Некоторая доработка может быть необходима для перемещения существующей программы частично или полностью с мастера на интеллектуальный слэйв. Ресурсы, требуемые для переноса частей пользовательской программы на интеллектуальный слэйв, зависят от того, как назначение адресов входов и выходов сохраняется в функциональных блоках в исходной программе.

Входы и выходы могут использоваться в функциях в исходной программе различными способами. Используемые адреса в текущем ET200S, могут быть невозможны в IM 151/CPU.

См. описание адресации IM 151/CPU в Разделе 2.1.

Перенос с адресами без уплотнения

Если Вы используете функциональные блоки с адресами ввода – вывода без уплотнения, требуемые части программы могут быть перенесены в IM 151/CPU легко, без потребности в преобразованиях.

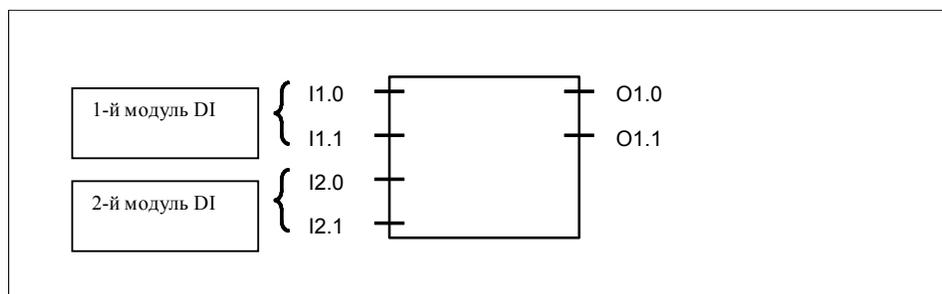


Рис D-1 Пример: FB с адресами без уплотнения

Перенос с уплотненными адресами

Если функциональные блоки с уплотненными адресами ввода - вывода скопирован для IM 151/CPU, уплотненные адреса больше не могут быть назначены для локальных модулей ввода - вывода, потому что CPU IM 151/CPU не может работать с такими адресами. Это требует переназначения в соответствующем функциональном блоке. Переназначение адресов соответствует "разуплотнению" адресов.

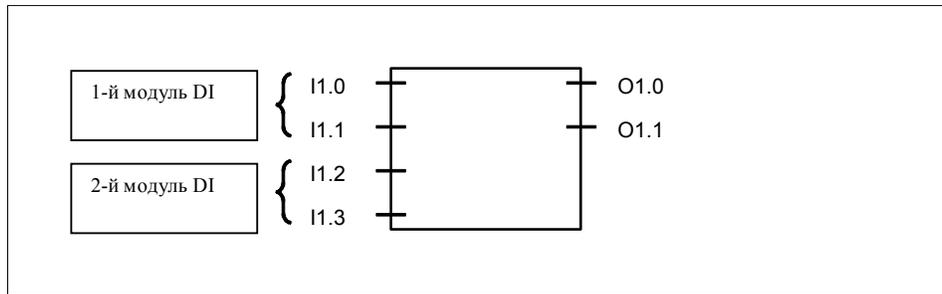


Рис. D-2 Пример: FB с уплотненными адресами

Переименование

Следующие блоки и адресные области могут быть переназначены:

- Входы, выходы,
- Меркеры, таймеры, счетчики
- Функции, функциональные блоки

Процедура переназначения состоит в следующем:

1. В **SIMATIC Manager**, выберите папку "**Blocks**", которая содержит блоки с уплотненными адресами, которые Вы хотите использовать в IM 151/CPU.
2. Выберите **Options ->Rewire**.
3. Введите необходимые замены в таблицу открытого диалогового окна "Переназначение" (старый адрес /новый адрес).

Таблица D-2 Пример: Замена с помощью **Options -> Rewire**

	Старый адрес	Новый адрес
1	I 1.2	I 2.0
2	I 1.3	I 2.1

4. Щелкните на ОК.

Начинается переименование. После переименования, Вы можете выбрать в диалоговом окне, хотите ли сохранить информацию о произведенном переименовании в специальном информационном файле. Файл содержит список старого и нового адресов. Различные блоки также внесены в список вместе с выполненными переименованиями в блоке.

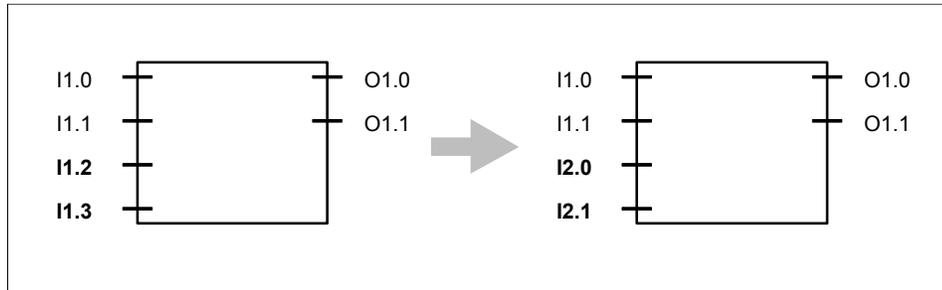


Рис. D-3 Пример: Переименование операндов

Если Вы назначаете символы для входов и выходов, с использованием таблицы символов в *STEP 7*, Вы должны изменить таблицу символов, чтобы адаптировать ее для использования в IM 151/CPU.

См. также встроенную систему помощи *STEP 7*.

Перенос функциональных блоков со входами - выводами в формате слова

Если Вы назначаете адреса входов / выходов в качестве фактического параметра ввода / вывода функционального блока, Вам придется значительно больше перепрограммировать для переноса программы.

Это возможно сделать с помощью адаптации оболочки использования FB, которая делает возможным использование FB с IM 151/CPU. Другой способ - перепрограммирование самого FB. Мы рекомендуем Вам второй способ, потому что это более легко чем перепрограммирование оболочки

См. также встроенную систему помощи *STEP 7*.

Diese Seite ist eine **Vakat**-Seite, die an das Ende eines Kapitels mit ungerader Seitennummer angehängt wird.

Глоссарий

Автономный режим работы

Устройство используется в автономном режиме работы, без обмена данными с мастером и без перекрестной связи с другими DP - слэями. Все модули запускаются, используя параметры по умолчанию и с максимальной конфигурацией (32 слота, 64 консистентных байта).

Адрес

Адрес - обозначение некоторого адресного идентификатора или адресной области (например вход I 12.1; слово меркеров MW 25; блок данных DB 3).

Адрес PROFIBUS

Каждый узел шины должен получить адрес PROFIBUS, чтобы распознавать его как уникальный объект на PROFIBUS.

PC/устройство программирования имеет адрес PROFIBUS, равный "0".

Для устройства децентрализованной периферии ET 200S допустимы адреса PROFIBUS от 1 до 125.

Аккумуляторы

Регистры в CPU - используются как промежуточная память для команд загрузки, передачи, сравнения, вычисления и преобразования.

Блок Данных

Блоки Данных (DB) - области данных в пользовательской программе, которые содержат пользовательские данные. К глобальным блокам данных можно обращаться из всех программных блоков, в то же время, что экземплярные блоки данных могут быть поставлены в соответствие определенным функциональным блокам.

Время Цикла

Время цикла - время, требуемое CPU, для одного выполнения пользовательской программы.

Глубина Вложения

Один блок может вызываться из другого посредством вызова блока. Глубина вложения – число блоков, вызванных одновременно.

Диагностика

Диагностика - это распознавание, локализация, классификация, отображение и дальнейший анализ ошибок, неисправностей и сообщений.

Диагностика включает в себя контрольные функции, которые исполняются автоматически во время работы системы. Это увеличивает возможность использования системы путем сокращения времен запуска в эксплуатацию и простоя.

Диагностический Буфер

Диагностический буфер - область памяти CPU, в которой диагностические события сохраняются в порядке их возникновения.

Диагностические прерывания

Модули, имеющие возможности диагностики, используют диагностические прерывания, чтобы сообщить ошибки системы, которые они обнаружили, центральному CPU.

В SIMATIC S7/M7: Когда ошибка обнаружена или исчезает (например: обрыв провода), ET200S вызывает диагностическое прерывание, если прерывания разрешены. CPU DP –мастера прерывает обработку пользовательской программы более низкого приоритета и обрабатывает блок диагностических прерываний (OB 82).

В SIMATIC S5: диагностические прерывания появляются в диагностике станции. При использовании циклического доступа к диагностическим битам в диагностике станции, Вы можете обнаруживать ошибки типа обрыва провода.

Загрузочная память

Загрузочная память - часть CPU. Она содержит объекты, созданные в устройстве программирования. Она представляет собой интегрированную память или вставную карточку памяти.

Запуск

Режим ЗАПУСК активизируется в течение перехода из состояния STOP в RUN.

Может быть вызван переключателем режимов или после подачи напряжения питания на контроллер. В случае ET200S выполняется перезапуск.

Индикация ошибки (Error display)

Индикация ошибки - одна из возможных реакций операционной системы на ошибку во время выполнения программы. Другие возможные реакции: в ответ на ошибки в пользовательской программе, CPU переходит в режим STOP.

Интеллектуальный DP - слэйв

Особенность определения интеллектуального DP слэйва – в том, что данные входа/выхода, не доступны DP -мастеру непосредственно, без предварительной обработки CPU (в этом случае интерфейсным модулем IM 151/CPU).

Класс приоритета

Операционная система S7-CPU имеет до 26 классов приоритета которым назначены различные организационные блоки. Классы приоритета определяют, какой ОВ прерывает другой ОВ. Если класс приоритета включает несколько ОВ, они не прерывают друг друга, но выполняются последовательно.

Консистентные Данные

Данные, которые обрабатываются совместно и не должны разделяться определяют консистентные (последовательные) данные.

Например, данные аналоговых модулей должны всегда обрабатываться консистентно. Другими словами, значение аналогового модуля не должно быть искажено, считыванием его в два приема в различное время.

Контрольное время цикла

Время в течение выполнения программы CPU, при котором область отображения процесса обновляется.

Масса

Масса образуется всеми соединенными друг с другом неактивными частями оборудования, на которых даже в случае неисправности отсутствует напряжение, опасное для прикосновения.

Мастер

Когда это устройство владеет маркером, оно может посылать данные к ведомым устройствам и запрашивать данные от них. Пример DP –мастера - CPU 315-2 DP.

Мастер-система

Все DP -слэйвы, которые назначены DP –мастеру для чтения и записи, составляют Мастер-систему с DP -мастером.

Маркер памяти

Маркеры памяти - часть системной области памяти CPU для хранения промежуточных результатов. К ним можно обращаться как к битам, байтам, словам или двойным словам.

Область отображения процесса

Это часть системной области памяти CPU. Состояния входных сигналов записываются в область отображения входов в начале циклической программы. В конце циклической программы состояния сигналов в области отображения выходов процесса передаются на выходную периферию.

Обработка ошибок

Если операционная система обнаруживает определенную ошибку (например ошибка доступа), это вызывает организационный блок (ОВ ошибок) предусмотренный на этот случай, который определяет последующее поведение CPU.

Операционная система CPU

Операционная система CPU организует все функции и процессы CPU, которые не связаны со специальной задачей управления.

Организационный блок

Организационный блок (ОВ), представляют собой интерфейс между операционной системой CPU и пользовательской программой. Последовательность обработки пользовательской программы определена в организационных блоках.

Параметр

1) переменная **STEP 7** блоков

2) переменная, для задания поведения модуля (один или больше на модуль). Каждый модуль имеет соответствующие предварительные установки, которые могут быть изменены, с помощью конфигурирования инструментами STEP 7.

Перезапуск

Когда запускается CPU (например, переводом выключателя из START в STOP или, включая напряжение питания), организационный блок ОВ 100 (полный перезапуск) выполняется прежде, чем начинается выполнение циклической программы (ОВ 1). При полном перезапуске, образ отображения процесса входов считывается и пользовательская программа обрабатывается, начиная с первой команды в ОВ 1.

Подписчик

Получатель в прямой связи- *Видит* Прямую Связь

Пользовательская память

Пользовательская память содержит исполняемые блоки и блоки данных пользовательской программы. Пользовательская память может быть интегрирована в CPU (IM 151/CPU) или может быть представлена на вставных картах памяти или модулях памяти. Однако пользовательская программа всегда выполняется в рабочей памяти CPU.

Пользовательская Программа

Программы SIMATIC подразделяются на операционную систему CPU и пользовательскую программу. Последняя создана с помощью программного обеспечения STEP7 на возможных языках программирования (LAD или STL) и сохраняется в кодовых блоках. Данные сохраняются в блоках данных.

Прямая Связь

Прямая связь - специальное взаимодействие между PROFIBUS-DP узлами. Прямая связь характерна тем, что PROFIBUS-DP узлы "слушают" чтобы определить, какие данные DP-слэйв посылает назад его DP - мастеру.

Прерывание от процесса

Процесс прерывания, вызван модулями, способными прерывать процесс при наступлении определенного события в процессе. Процесс прерывается и сообщает об этом CPU. Назначенный операционной системой организационный блок тогда будет обработан в соответствии с приоритетом прерывания.

Прерывания

Операционная система CPU имеет 10 различных классов приоритета, которые управляют обработкой пользовательской программы. Эти уровни во время выполнения включают прерывания, например диагностические прерывания. Когда прерывание вызвано, операционная система автоматически вызывает соответствующий организационный блок, в котором пользователь может запрограммировать желаемую реакцию (например в FB).

Программатор

Программатор - по существу персональный компьютер, который является компактным, портативным и подходящим для промышленного применения. Они оборудованы специальными аппаратными средствами и программным обеспечением для программируемых контроллеров SIMATIC.

Программируемый контроллер

Программируемые контроллеры (PLCs) - электронные контроллеры, чья функция сохраняется как программа в блоке управления. Конфигурация и монтаж узлов независимы от функций системы управления. Программируемый логический контроллер имеет структуру компьютера: он состоит из CPU (центрального процессорного устройства) с памятью, модулями входов и выходов и внутренней шинной системой. Входы - выходы и язык программирования разработаны с целью решения задач управления.

Рабочая память

Рабочая память – RAM память в CPU, к которому обращается процессор в течение выполнения программы.

Рабочий режим

SIMATIC S7 программируемые контроллеры имеют следующие рабочие режимы: STOPO, ³STURT-UP, RUN.

Реакция на ошибки

Ответ на ошибку во время выполнения. Операционная система может реагировать следующими способами: перевод программируемого контроллера, в режим STOP, вызов организационного блока, в котором пользователь может запрограммировать реакцию или индикацию ошибки.

Резервная Память

Резервная память - для сохранения областей памяти CPU без буферной батареи. Конфигурируемое количество таймеров, счетчиков, меркеров памяти и байтов данных (сохраняемые таймеры, счетчики, меркеры памяти, байты данных) может быть сохранено после отключения питания CPU.

Сжатие памяти

Устройство программирования с помощью функции "Сжатие" располагает все действительные блоки в непрерывной последовательности в RAM-памяти CPU с начального адреса пользовательской памяти. Это устраняет все промежутки, которые возникли, когда блоки были удалены или изменены.

Система Автоматизации

Система автоматизации –представляет собой программируемый логический контроллер (PLC), состоящий из по крайней мере одного CPU, различных входных и выходных модулей, а также интерфейсов с оператором.

Системная диагностика

Системная диагностика - термин, описывающий обнаружение, оценку и признак ошибок, которые происходят в программируемом контроллере. Примеры таких ошибок - ошибки программы или отказы модуля. Системные ошибки могут быть индицированы посредством светодиодов или с помощью STEP 7.

Системная память

Системная память встроена в центральный процессор и представляет собой RAM-память. Системная память включает адресные области (например таймеры, счетчики, меркеры памяти, и т.д.) также как области данных (например буферы связи) требуемые внутренней операционной системой.

Системная функция

Системная функция (SFC) - функция, интегрированная в операционную систему CPU, которая может быть вызвана в пользовательских программах, если это необходимо.

Системы распределенного ввода - вывода

Это модули ввода/вывода, которые не находятся в основном устройстве, они удалены на некоторое расстояние от CPU. Примеры таких устройств:

- ET200S, ET 200M, ET 200B, ET 200C, ET 200U, ET 200X, ET 200L
- DP/AS-I Соединения
- S5-95U с PROFIBUS-DP слэив интерфейсом
- Другие DP - слэйвы производства Siemens или других производителей

Распределенные системы ввода - вывода связаны с DP –мастером через PROFIBUS-DP.

Скорость передачи

Скорость передачи данных измеряется в битах, переданных в секунду (скорость передачи = числу бит в секунду).

В случае ET 200S, возможны скорости передачи 9.6 kbps к 12 Mbps.

Слэйв

Слэйв может только обмениваться данными с мастером по его запросу. Под слэйвами мы подразумеваем, например, все DP - слэйвы типа ET-200S, ET- 200B, ET- 200X, ET- 200M, и т.д.

Стартовые события

Стартовые события - определенные события типа ошибок, временных или других прерываний. Они заставляют операционную систему запускать соответствующий организационный блок (если пользователем запрограммировано это). Стартовые события показаны в стартовой информации соответствующего ОВ. Пользователь может разрешить начальные события в пользовательской программе.

Суммарный ток

Сумма токов всех каналов модуля цифровых выходов .

Счетчик

Счетчики - часть i памяти системы CPU. Содержание счетчиков может изменяться командами **STEP 7** (например, счет вверх / вниз).

Таймеры

Таймеры - часть системной области памяти системы CPU. Содержание " ячейки таймера " обновляется автоматически операционной системой асинхронно к пользовательской программе. **STEP 7** инструкции используются, чтобы определить точную функцию таймера (например -задержка) и начинать их выполнение (например -запуск).

Узел

Устройство, которое может посылать, получать или усиливать данные через шину, например : DP – мастер , DP - слэйв, RS 485 - повторитель или активный соединитель звездой.

Файл базы данных устройства

Все специфические свойства DP -слэйва сохраняются в файле базы данных устройства (DDB файл). Формат файла базы данных устройства определен в EN 50170, часть 2, PROFIBUS.

Функция

Функция (FC) в соответствии с IEC 1131-3 – кодовый блок без статических данных. Функция позволяет использовать формальные параметры в пользовательской программе. Функции поэтому подходят для программирования часто повторяемых сложных функций (например вычисления).

ЦПУ (CPU)

Центральный процессорный узел S7 программируемого контроллера с блоком управления и арифметическим логическим устройством, памятью, операционной системой и интерфейсом для устройства программирования.

Шина

Общая линия передачи, соединяющая все узлы шины и имеющая два определенных оконечных подключения. В случае ET 200S, шина - двухпроводный или оптоволоконный кабель.

ET 200

Система децентрализованной периферии ET 200 с протоколом PROFIBUS–DP - это шина для подключения децентрализованной периферии к CPU или адекватному Master-устройству DP. ET 200 отличается малым временем реакции , так как передается небольшое количество данных (.....). ET 200 основывается на стандарте EN 50 70, том 2, PROFIBUS. ET 200 работает по принципу Master–Slave. Master-устройством DP может быть , напр ., интерфейсный модуль IM 308–C или CPU 3 5–2 DP. Slave-устройствами DP могут быть устройства децентрализованной периферии ET 200B, ET 200C, ET 200M, ET 200U или Slave-устройства DP фирмы Siemens или других изготовителей .

DP - мастер

Мастер, который соответствует EN 50170, Том 2, PROFIBUS, называется DP - мастером.

DP - слэйв

Слэйв в шинной системе PROFIBUS с PROFIBUS-DP протоколом, который соответствует EN 50170, Том 2, PROFIBUS стандарт, называется DP- слэйвом.

DP Стандарт

DP стандарт - шинный протокол распределенных систем ввода - вывода ET 200, основанный на EN50170, том 2, PROFIBUS.

FORCE

В течение ввода в действие и пусконаладочных работ , например, эта функция позволяет некоторые выходы установить в "ON" на некоторое время, даже если логические команды пользовательской программы не включают их (например, потому что соответствующие их включению входы - не подключены).

FREEZE

Это команда управления Master-устройства DP группе Slave-устройств DP. После получения команды управления FREEZE Slave-устройство DP замораживает текущее состояние **входов** и передает их циклически на DP–Master. После каждой новой команды управления FREEZE Slave-устройство DP снова замораживает состояние **входов** .Входные данные только тогда будут снова циклически передаваться от Slave-устройства DP на DP–Master, когда DP–Master пошлет команду управления UNFREEZE.

MMC

Микро- карта памяти для SIMATIC систем. Может использоваться как загрузочная память и как портативный носитель данных.

MPI

Многоточечный интерфейс (MPI) - SIMATIC S7 интерфейс для программирования устройств.

ОВ-приоритет

Операционная система CPU различает классы приоритета, типа циклической обработки программы и обработки программы управляемой процессом. Каждому классу приоритета назначены, соответствующие номера организационных блоков (ОВ), в которой S7- пользователь может запрограммировать соответствующую реакцию. ОВ имеют различные стандартные приоритеты, которые определяют порядок, в котором они выполняются или прерываются, когда они активизированы одновременно.

PG -Программирующее устройство

PLC -Программируемый логический контроллер

PROFIBUS

Process Field Bus - Немецкий стандарт полевой шины, определенный в IEC 61158/EN 50170, Том 2, PROFIBUS стандарт. Это определяет функциональные, электрические и механические особенности последовательной полевой шинной системы.

PROFIBUS доступен с протоколами DP (Немецкое сокращение для распределенного ввода - вывода), FMS (спецификация шинного полевого сообщения), PA (= автоматизация процесса) или TF (= технология функции, связанной с процессом).

PROFIBUS Адрес

Каждый узел шины должен получить PROFIBUS адрес, чтобы однозначно идентифицировать его на PROFIBUS шине.

Устройство PC/PG имеет PROFIBUS адрес - "0".

PROFIBUS адрес от 1 до 125, допустим для ET200S распределенной системы ввода - вывода.

Runtime Error

Ошибка, которая происходит во время обработки пользовательской программы на программируемом контроллере (не во внешнем процессе).

SFC -> Системная функция**STEP 7**

Язык программирования для создания пользовательских программ для контроллеров SIMATIC S7.

SYNC

Это - команда управления DP – мастера к группе DP слэймов.

Посредством команды управления SYNC, DP - мастер заставляет DP слэив замораживать текущие статусы выходов. В последующих циклах, DP - слэив хранит данные выходов, но статусы выходной периферии остаются неизменными.

После каждой новой команды управления SYNC, DP - слэив устанавливает выходы, которые это были сохранены как данные выходов. Выхода не обновляются циклически снова, пока DP - мастер не посылает команду управления UNSYNC.

Token

Право доступа к шине

Предметный указатель

Символы

), В-15
)MCR, В-52
+, В-39
+AR1, В-40
+AR2, В-40
+D, В-38
+I, В-37
+R, В-39
-D, В-38
-I, В-37
-R, В-39
*D, В-38
*I, В-37
*R, В-39
/D, В-38
/I, В-37
/R, В-39
=, В-26
==D, В-41
==I, В-40
==R, В-42
<=D, В-41
<=I, В-40
<=R, В-42
<> R, В-42
<>D, В-41
<>I, В-40
<D, В-41
<I, В-40
<R, В-42
>=D, В-41
>=R, В-42
>>=I, В-40
>>D, В-41
>>I, В-40
>>R, В-42

А

Автономная работа, Глоссарий-10
ET 200S, 3-4
ET 200S, 3-9
Адрес, Глоссарий-1
Адрес PROFIBUS , его изменение, 3-6

Адресация, ориентированная на слот, 2-2
Адресная область
Последовательность данных, 2-7
Установки по умолчанию, 2-10
Для передачи пользовательских данных,
2-7
Расширенных модулей, 2-3
Назначение адреса, для аналоговых и
цифровых модулей, 2-3
Идентификаторы адреса, В-2
Адресные регистры, В-4
Адреса
Базовый адрес, 2-5
Для диагностики, 2-7
Для передачи данных пользователя, 2-7
Адресация, 2-1
Расположение адреса, 2-5
косвенная, В-7
интерфейс в STEP 7, 2-9
правила, 2-9
ориентированный на слот, 2-2
определённый пользователем, 2-5
Адресация по умолчанию, 2-2
Аккумулятор, В-3, Глоссарий-1

Б

Блок данных, Глоссарий-2
Блоки, ВМ 147/CPU, 5-13

В

Ввод в действие, 4-1, 4-7, 8-1, 8-2, 8-6, 8-11, 8-17
Взаимообмен данными, образец программы, 2-10
ВМ 147/CPU
Переключатель режимов, 5-4
параметры, 5-16
Время выполнения, пользовательская программа, 7-2
Время обработки
Операционная система, 7-3
Обновление изображения процесса, 7-3
Пользовательская программа, 7-4

Время реакции, 7–6
 Диагностическое прерывание, 7–9
 Самое длинное, 7–8
 Прерывание процесса, 7–9
 Самое короткое, 7–7
 Время реакции диагностического прерывания, 7–9
 Время цикла, 7–2, Глоссарий–2
 увеличение, 7–3
 структура, 7–2
 прерывания цикла, 5–17
 Встроенные часы, 5–12
 Входы, время задержки, 7–5
 Выполнение цикла, 5–17
 Выходы, время задержки, 7–5

Г
 Глубина вложения, Глоссарий–7

Д
 Данные конфигурации, ввод, 5–17
 Диагностика
 модуль, 4–20
 станция, 4–14, 4–22
 Диагностика, 4–1, Глоссарий–3
 система, Глоссарий–11
 использование светодиодов, 4–9
 Диагностика модуля, 4–20
 Диагностика системы, 5–16, Глоссарий–11
 Диагностика станции, 4–14, 4–22
 Диагностический адрес, 2–7, 4–11, 4–13
 Диагностический буфер, Глоссарий–3
 вход, 4–11
 считывание, 4–5
 Диагностическое прерывание, 4–23,
 Глоссарий–3
 ОВ 82, 5–14
 Доступ к ET 200S из PG/PC, 3–3

З
 Загрузочная память, Глоссарий–6
 Задержка, входов/выходов, 7–5
 Заземление, Глоссарий–1
 Заказной номер
 IM 151/CPU, 6–2
 Модуль памяти, 5–8
 Компоненты сети, 3–7
 Запуск, Глоссарий–11
 Запуск, характеристики, 5–16
 ET 200S интерфейсный модуль IM 151/CPU
 A5E00058783-01

И
 Идентификация события в DP- мастере/DP-
 слэйве, 4–12
 Идентификатор, посылка конфигурации,
 специальный формат идентификатора,
 А–3
 Изменение рабочего режима, 4–12
 Индикация, светодиоды, 4–9
 Индикация ошибки, Глоссарий–4
 Инструкции загрузки, 2–8
 Интеллектуальный DP- слэйв, 4–2,
 Глоссарий–6

К
 Кабели, 3–7
 Команда передачи, 2–8
 Компоненты, ET 200S, 1–5
 Контрольное время цикла, Глоссарий–10
 Консистентность, 2–7, 2–8
 Консистентность данных, Глоссарий–2
 Консистентные данные, Глоссарий–2
 Конфигурирование, ET 200S автономно, 3–4
 Конфигурирование, программное
 обеспечение, 1–4
 Конфигурирование, 8–1, 8–7, 8–12
 IM 151/CPU, 4–2

М
 Мастер, Глоссарий–6
 Мастер-система, Глоссарий–6
 Межузловая связь, смотри прямую связь, 3–
 10
 Микро карта памяти, Глоссарий–7
 Микро карта памяти, 5–8
 Модернизация, 8–1, 8–11
 Модуль памяти, 5–8
 Монтаж, 8–1, 8–5

Н
 Назначение параметра с
 DP- мастером, А–1
 Назначение слота, ET–200S, 2–2

О

Обзор продукта, 1-1
Области параметров, STEP 7 инструкции, В-2
Обмен данными, прямой, 3-10
Обновление таймеров S7, 7-4
Обработка прерывания процесса, 7-9
Общие технические спецификации, 6-2
Операционная система
CPU, Глоссарий-8
Время обработки, 7-3
Организационный блок, Глоссарий-8
Отображение процесса, Глоссарий-8
Отображение процесса, время обновления отображения процесса, 7-3
Ошибка во время прямого доступа к входу/выходу, ОВ 122, 5-15
Ошибка/дефект, прямого доступа к входу/выходу, ОВ 122, 5-15
Ошибка времени исполнения, Глоссарий-10
Ошибка, обрабатываемая через ОВ, Glossary-4
Ошибка станции, ОВ 86, 5-15

П

Память
сохраняемая, Глоссарий-1
загрузочная, Глоссарий-6
системная, Глоссарий-11
пользовательская, Глоссарий-12
рабочая, Глоссарий-12
Параметр, Глоссарий-8
Параметры, ВМ 147/CPU, 5-16
Передача данных
принцип, 2-1
с DP-мастером, 2-6
Передача пользовательских данных, DP-мастеру, 2-6
Перезапуск, Глоссарий-10
ОВ 100, 5-14
Переключатель режимов, 5-4
Сброс памяти, 4-4
MRES, 5-4
RUN-P, 5-4
STOP, 5-4
Перемонтаж, D-4
Пользовательская память, Глоссарий-12
Пользовательская программа, Глоссарий-12
Время обработки, 7-4
П-3

Последовательность данных, 2-7, 2-8
Посылка конфигурации, А-1
конфигурирование, нормальный формат ID, А-4
установка по умолчанию, А-5
структура, А-4
специальный формат идентификатора, А-2
Посылка назначения параметра, А-1
конфигурация, 5-18
структура, А-6
Прерывание, 4-14, Глоссарий-6
процесс, Глоссарий-9
Прерывания, расширение цикла, 7-5
Прерывание от процесса, 4-14, 4-23, Глоссарий-9
ОВ 40, 5-14
Прерывание от процесса, время реакции, 7-9
Правила адресации, 2-9
Прерывания по времени, 5-16
Примеры адресации, STEP 7 инструкции, В-5
Приоритет, ОВ, Глоссарий-7
Приоритетный класс, Глоссарий-8
Программирование 8-15, 8-1, 8-2, 8-9
Программирующее устройство, функции 3-9
Программирование программирующего устройства, 1-4
Программирующее устройство
Соединительный кабель, 3-3, 3-7
Предварительные условия, 3-2
Промежуточная память
Доступ в программе потребителя, 2-8
в IM 151/CPU, 2-6
Прямая связь, Глоссарий-3, 3-10,
Прямая связь данных, Глоссарий-3

Р

Рабочая память, Глоссарий-12
Рабочий режим, Глоссарий-7
Распределённое устройство ввода/вывода, Глоссарий-4
Расширение цикла, через прерывания, 7-5
Реакция на ошибки, Глоссарий -5
Реакция на ошибки/дефекты, 5-15
Режим
RUN, 5-5
STOP, 5-5
Резервная память, Глоссарий-1
Руководства, справочник, 1-5
ET 200S интерфейсный модуль IM 151/CPU A5E00058783-01

С

Сброс памяти
 Внутренние события CPU, 4–5
 Сброс памяти IM 151/CPU, 4–4
 С помощью переключателя режимов, 5–4
Сеть, структура, 3–1
Сеть, компоненты, 3–7
Система автоматизации, Глоссарий–1
Системная память, Глоссарий–11

Системные функции, SFC, Глоссарий–11,
 С–1
Скорость передачи, Глоссарий–12
Слэйв, Глоссарий –10
Слэйв, диагностика 4–14
Сжатие, Глоссарий–2
Слово состояния, В–4
Содержание руководства, краткий обзор, 1–
 5
Состояние модуля, 4–21
Состояние станции 1 to 3, 4–16
Сохраняемость, 5–16
Стартовое событие, ОВ, Глоссарий–10
Стартовые события, для ОВs, 5–14
Счётчик, Глоссарий–2

Т

Таймеры (таймерные ячейки), Глоссарий–
 12
Тестовые функции, 3–9
Тест, прогон, 8–1, 8–2, 8–10, 8–17
Технические спецификации
 общие, 6–2
 IM 151/CPU, 6–2
 PROFIBUS–DP, 5–2
Токен, Глоссарий–12

У

Узел, Глоссарий–7
Указатели, расчёт В–6
Установка по умолчанию, А–5
 для адресных областей, 2–10
Установка, 8–1, 8–4

Ф

Файл базы данных устройства, Глоссарий–2
Функция, FC, Глоссарий–6
Функции, через PG, 3–9

ET 200S интерфейсный модуль IM 151/CPU
A5E00058783-01

Ч

Часы, 5–12, 5–16
Часы программного обеспечения, 5–12, 5–
 16

Ц

Цикл, ОВ 1, 5–14

Ш

Шина, Glossary–1
Шинный соединитель, 3–7
Шинная ошибка, 3–9

A

A, B-13, B-17, B-19
A(, B-15
AD, B-18
AN, B-13, B-15, B-17, B-20
AW, B-18

B

BE, B-49
BEC, B-49
BEU, B-49
BLD, B-45
BTD, B-46
BTI, B-46

C

CAD, B-45
CALL, B-48
CAR, B-35
CAW, B-45
CC, B-48
CD, B-29
CDB, B-49
Clock memory, 5-16
CLR, B-27
CPU, Глоссарий-2
 Операционная система, Глоссарий-8
CU, B-29

D

DBs, 5-13
DDB (база данных устройства) файл, 5-2
DEC, B-45
DP, диагностический адрес, 2-7
DP- мастер, Глоссарий-4
DP- слэив, Глоссарий-4
 интеллектуальный, 4-2, Глоссарий-6
 диагностика DP-слэйва, структура, 4-15
DP стандарт, Глоссарий-4
DTB, B-46
DTR, B-46

П-5

E

ET 200, Глоссарий-5
ET 200S
 компоненты, 1-5
 руководства, 1-5
ET 200S руководства, справочник, 1-5

F

FBs, 5-13
FCs, 5-13
Характеристики, 6-2
 IM 151/CPU, 1-4, 5-2
FN, B-25
Force, 5-5, Глоссарий-5
FP, B-25
FR, B-28, B-29
FRCE, LED, 5-5
FREEZE, Глоссарий-5

I

ID изготовителя, CPU 31x-2 DP- слэйв, 4-19
IM 147/CPU, блоки, 5-13
IM 151/CPU
 конфигурирование, 4-2
 характеристики, 1-4
 важные характеристики, 5-2
 сброс памяти, 4-4
IM 151/CPU FO интерфейсный модуль, 6-3,
6-4
 Основная диаграмма цепи, 6-4
 Назначение клемм, 6-3
IM 151/CPU интерфейсный модуль, 6-4
 Основной план подключения, 6-4
 Технические спецификации, 6-5
INC, B-45
INVD, B-47
INVI, B-47
ITB, B-46
ITD, B-46

J

JBI, B-50
JC, B-50
JCB, B-50
JCN, B-50
JL, B-51

ET 200S интерфейсный модуль IM 151/CPU
A5E00058783-01

JM, В-51
 JMZ, В-51
 JN, В-51
 JNB, В-50
 JNBI, В-50
 JO, В-51
 JOS, В-51
 JP, В-51
 JPZ, В-51
 JU, В-50
 JUO, В-51
 JZ, В-51

L

L, В-30, В-31, В-32, В-36
 LAR1, В-35
 LAR2, В-35
 LD, В-32
 LED(светодиод), 4-5
 индикация, 4-9
 FRCE, 5-5
 индикатор, 1-4
 ON, 5-5
 RUN, 5-5
 SF, 5-5
 STOP, 5-5
 LOOP, В-51

M

MCR(, В-52
 MCRA, В-52
 MCRD, В-52
 MMC, Глоссарий-7
 MMC модуль, 5-8
 MOD, В-38
 MPI, 3-2, Глоссарий-7
 MPI узловой адрес, 5-17
 MRES, переключатель режимов, 5-4

N

NEGD, В-47
 NEGI, В-47
 NOP, В-45
 NOT, В-27

O

O, В-13, В-16, В-21
 O(, В-15
 OB, Глоссарий-8
 Стартовое событие, Глоссарий-10
 OB 1, 5-14
 OB 100, 5-14
 OB 122, 4-12, 5-15
 OB 40, 4-14, 5-14
 OB 82, 4-8, 4-12, 5-14
 OB 86, 4-8, 4-12, 5-15
 OB приоритет, Глоссарий-7
 OB10, 5-14
 OB121, 5-15
 OB20, 5-14
 OB35, 5-14
 OB80, 5-15
 OB83, 5-14
 OB85, 5-15
 OBs
 CPU, 5-13
 Стартовое событие, 5-14
 OD, В-18
 ON, В-13, В-17, В-22
 LED, 5-5
 ON(, В-15
 Online -функции, для ET 200S, 3-9
 OPN, В-48
 OW, В-18

P

PC
 соединение с ET 200S, 3-4
 предварительные условия, 3-2
 PG, Глоссарий-9
 Соединение с ET 200X, 3-4
 PLC, Глоссарий-9
 POP, В-45
 PROFIBUS, Глоссарий-9
 PROFIBUS адрес, Глоссарий-9,3-3
 PROFIBUS, Адрес мастера 4-18
 PROFIBUS сеть
 Компоненты сети,3-7
 Структурные принципы,3-2
 PROFIBUS-DP, спецификации,5-2
 PUSH, В-45

R

R, B-26, B-28, B-29
RLD, B-44
RLDA, B-44
RND, B-46
RND+, B-46
RND-, B-46
RRD, B-44
RRDA, B-44
RUN
 LED, 5-5
 режим, 5-5
RUN-P, переключатель режимов, 5-4

S

S, B-26, B-29
S7 таймеры, обновление, 7-4
SAVE, B-27
SD, B-28
SE, B-28
SET, B-27
SF, B-28
 Светодиод, 5-5
SFBS, 5-13
SFC, C-1
SFC DP_PRAL, 4-14
SFC DPRD_DAT, 2-8
SFC DPWR_DAT, 2-8
SFCs, 5-13
 Времена выполнения, C-1
SLD, B-43
SLW, B-43
SP, B-28
SRD, B-43
SRW, B-43
SS, B-28
SSD, B-43
SSI, B-43
STEP 7, Глоссарий-11
 Адресный интерфейс, 2-9
 конфигурирование IM 151/CPU, 4-2
 Установочные параметры, 3-9
STOP
 Светодиод, 5-5
 режим, 5-5
 Переключатель режимов, 5-4
SYNC, Глоссарий-11

T

T, B-33, B-34, B-36
TAK, B-45
TAR1, B-35
TAR2, B-35
TRUNC, B-46

U

UC, B-48

X

X, B-14, B-17, B-23
X(), B-15
XN, B-14, B-17, B-24
XN(), B-15
XOD, B-18
XOW, B-18

