

## Обзор главы

В разделе	Вы найдете	на стр.
14.1	Обзор	14–2
14.2	Загрузка и передача	14–3
14.3	Чтение или передача слова состояния	14–6
14.4	Загрузка значений времени и счетчиков как целых чисел	14–7
14.5	Загрузка значений времени и счетчиков в формате BCD	14–9
14.6	Загрузка и передача между адресными регистрами	14–11
14.7	Загрузка данных из блока данных	14–12

## 14.1. Обзор

### Определение

Операции загрузки (L) и передачи (T) позволяют программировать обмен информацией между модулями ввода или вывода и областями памяти или между областями памяти. CPU выполняет эти операции в каждом цикле как безусловные операции, т.е. результат логической операции на них не влияет.

### Обмен информацией

Операции загрузки и передачи позволяют производить обмен информацией между следующими периферийными областями и областями памяти:

- Модули ввода и вывода и следующие области памяти:
  - отображения процесса на входах и выходах
  - меркеры
  - таймеры и счетчики
  - области данных
- Отображения процесса на входах и выходах и следующие области памяти:
  - меркеры
  - таймеры и счетчики
  - области данных
- Таймеры и счетчики и следующие области памяти:
  - отображения процесса на входах и выходах
  - меркеры
  - области памяти

### Процесс обмена

Операции загрузки и передачи производят обмен информацией через аккумулятор. Операция загрузки L записывает (загружает) содержимое соответствующего исходного адреса в АККУ 1, перемещая все содержащиеся в нем данные в АККУ 2. Старое содержимое АККУ 2 заменяется. Операция передачи T копирует содержимое АККУ 1 и записывает его в соответствующую целевую память. Так как операция передачи только копирует данные, находящиеся в АККУ 1, последние остаются доступными также и для других операций.

Операции загрузки и передачи могут обрабатывать информацию байтами (8 бит), словами (16 бит) и двойными словами (32 бита).

Аккумулятор содержит 32 бита. Данные длиной менее 32 бит располагаются в аккумуляторе справа. Остальные биты аккумулятора заполняются нулями.



## 14.2. Загрузка и передача

### Описание

С помощью операций загрузки и передачи можно передавать информацию из АККУ 1 или в него. Данные при этом могут иметь следующие размеры:

- байт (B, 8 бит)
- слово (W, 16 бит)
- двойное слово (D, 32 бита)

Байт загружается в младший байт младшего слова АККУ 1. Слово загружается в младшее слово АККУ 1. Неиспользуемые байты при загрузке в АККУ 1 сбрасываются в ноль.

### Непосредственная

Операция загрузки может обращаться к константам в 8, 16 и 32 бита, а **адресация** также к символам ASCII. Этот вид адресации называется непосредственной адресацией (см. гл. 9.1 и таблицу 14–1.

Таблица 14–1. Операнды операций загрузки: непосредственная адресация

Операнд	Пример	Пояснение
±..	L +5	Загрузить константу (целое число, 16 бит) в АККУ 1.
B#(...,.)	L B#(1,10)  L B#(1,10,5,4)	Загрузить константу как 2 байта в АККУ 1. (В этом примере 10 находится в младшем байте младшего слова АККУ 1; 1 находится в старшем байте младшего слова АККУ 1, см. рис. 2–1.  Загрузить константу как 4 байта в АККУ 1. (В этом примере 4 и 5 находятся соответственно в младшем и старшем байтах младшего слова АККУ 1; 10 и 1 находятся в младшем и старшем байтах старшего слова АККУ 1, см. рис. 2–1.)
L#	L L#+5	Загрузить 32–битную целую константу в АККУ 1.
16#..	L B#16#EF L W#16#FAFB L DW#16#1FFE_1ABC	Загрузить 16–ричную константу (8 бит) в АККУ 1. Загрузить 16–ричную константу (16 бит) в АККУ 1. Загрузить 16–ричную константу (32 бита) в АККУ 1.
2#..	L 2#1111_0000_1111_0000 L 2#1111_0000_1111_0000_1111_0000_1111_0000	Загрузить двоичную константу (16 бит) в АККУ 1. Загрузить двоичную константу (32 бита) в АККУ 1.
'..'	L 'AB' L 'ABCD'	Загрузить 2 символа в АККУ 1. Загрузить 4 символа в АККУ 1.
C#..	L C#1000	Загрузить константу счетчика (16 бит) в АККУ 1.
S5TIME#..	L S5TIME#2S	Загрузить константу S5TIME (16 бит) в АККУ 1.
..	L 1.0E+5	Загрузить вещественное число (32 бита, IEEE–FP) в АККУ 1.
P#..	L P#E1.0 L P##Start	Загрузить указатель (32 бита) в АККУ 1. Загрузить указатель (32 бита) на локальную переменную (Start) в АККУ 1.
D#	L D#1994–3–15	Загрузить дату (16 бит) в АККУ 1.
T#	L T#0D_1H_1M_0S_0MS	Загрузить значение времени (32 бита) в АККУ 1.
TOD#	L TOD#1:10:3.3	Загрузить время суток (32 бита) в АККУ 1.

**Прямая и  
косвенная  
адресация**

Операции загрузки и передачи могут обращаться к байту (B), слову (W) или двойному слову (D) посредством прямой или косвенной адресации в следующих областях памяти (см. также главы 9.2, 9.3, 9.5):

- Отображение процесса на входах и выходах (обозначения операндов EB, EW, ED, AB, AW, AD).
- Внешние входы и выходы (обозначения операндов PEB, PEW, PED, PAB, PAW, PAD). Внешние входы могут использоваться только в качестве операндов операций загрузки, внешние выходы только как операнды операций передачи.
- Битовая память (обозначения операндов MB, MW, MD).
- Блок данных (обозначения операндов DBB, DBW, DBD, DIB, DIW, DID).
- Локальные данные (временные локальные данные, обозначения операндов LB, LW, LD)

В таблице 14–2 перечислены операнды операций загрузки и передачи, применяющие прямую и косвенную адресацию.

Таблица 14–2. Операнды операций загрузки и передачи: прямая и косвенная адресация

Обозначение операнда	Максимальная адресная область в зависимости от вида адресации		
	прямая	косвенная через память	косвенная через регистр, внутризонная
EB EW ED  AB AW AD	0 до 65 535 0 до 65 534 0 до 65 532  0 до 65 535 0 до 65 534 0 до 65 532	[DBD] 0 до 65 532 [DID] [LD] [MD]	[AR 1, 0 до 8 191 P#byte.bit]  [AR 2, P#byte.bit]
PEB PEW PED (только L)  PAB PAW PAD (только T)	0 до 65 535 0 до 65 534 0 до 65 532  0 до 65 535 0 до 65 534 0 до 65 532	[DBD] 0 до 65 532 [DID] [LD] [MD]	[AR 1, 0 до 8 191 P#byte.bit]  [AR 2, P#byte.bit]
MB MW MD	0 до 65535 0 до 65534 0 до 65532	[DBD] 0 до 65 532 [DID] [LD] [MD]	[AR 1, 0 до 8 191 P#byte.bit]  [AR 2, P#byte.bit]
DBB DBW DBD  DIB DIW DID  LB LW LD	0 до 65 535 0 до 65 534 0 до 65 532  0 до 65 535 0 до 65 534 0 до 65 532  0 до 65 535 0 до 65 534 0 до 65 532	[DBD] 0 до 65 532 [DID] [LD] [MD]	[AR 1, 0 до 8 191 P#byte.bit]  [AR 2, P#byte.bit]

**Косвенная  
адресация с  
указанием области  
памяти в регистре**

Операции загрузки и передачи могут обращаться к байту (B), слову (W) или двойному слову (D) посредством косвенной адресации с указанием области памяти в регистре (см. гл. 9.6).

Таблица 14–3. Операнды операций загрузки и передачи: косвенная адресация через регистр с указанием области памяти

Обозначение операнда <sup>1)</sup>	Адресная область
B (байт), W (слово), D (двойное слово)	[AR 1, P#byte.bit]      0 до 8 191 [AR 2, P#byte.bit]

- 1) Эта область памяти закодирована в битах от 24 до 31 адресного регистра AR 1 или AR 2 (см. гл. 9.6).

**Байт, слово или  
двойное слово  
как параметры**

Операции загрузки и передачи могут использовать в качестве операндов также байт, слово или двойное слово, передаваемые как параметры

Таблица 14–4. Операнды операций загрузки и передачи: байт, слово или двойное слово, передаваемые как параметры

Операнд	Формат адресного параметра
Символическое имя	Байт, слово или двойное слово, передаваемые как параметры

14.3. Чтение или передача слова состояния

Загрузка слова состояния

С помощью операции загрузки L можно загрузить в АККУ 1 биты с 0 по 8 слова состояния (см. рис. 14–1). биты с 9 по 31 АККУ 1 сбрасываются на “0”. Эту команду Вы можете взять из примера, следующего за рис. 14–1.

**Указание**  
В CPU семейства S7–300 биты слова состояния /ER, STA и OR не загружаются командой L STW. Только биты 1, 4, 5, 6, 7 и 8 загружаются на позиции соответствующих битов младшего слова АККУ 1.

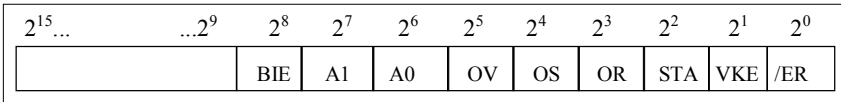


Рис. 14-1. Структура слова состояния

AWL	Объяснение
L STW	Загрузить биты с 0 по 8 слова состояния в младшее слово АККУ 1.

Передача в слово состояния

С помощью операции передачи T можно передать содержимое АККУ 1 в слово состояния (см. рис. 14–1). Команду Вы можете взять из следующего примера.

AWL	Объяснение
T STW	Передать содержимое АККУ 1 в слово состояния.

14.4. Загрузка значений времени и счетчиков как целых чисел

**Загрузка значения времени**      Значение времени хранится в слове таймера в двоичном коде. С помощью следующей операции загрузки L двоичное значение времени можно считать из слова таймера и в том же формате загрузить в младшее слово      АККУ 1:

L <слово таймера>

Этот вид загрузки называют прямой загрузкой значения времени.

Значение времени в слове таймера при обработке программы в CPU уменьшается с начального значения до “0”. Применяя операцию загрузки L со словом таймера в качестве операнда, Вы получаете значение между стартовым временем слова таймера и “0”. Время, прошедшее с момента старта, вычисляется как разность между стартовым временем и временем, считанным в данный момент.

AWL	Объяснение
L T1	Загрузить значение времени таймера T1 в двоичном коде прямо в АККУ1-L.

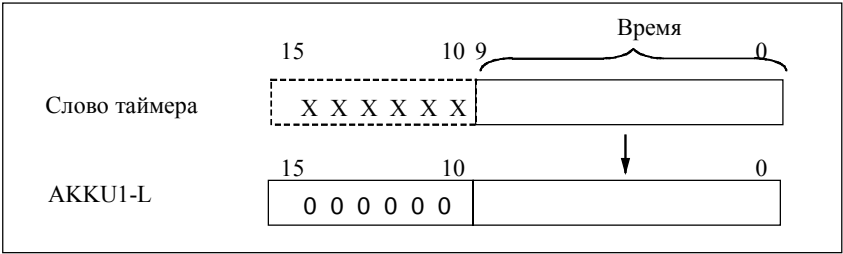


Рис. 14-2. Загрузка значения времени в АККУ 1 операцией L

Вы можете использовать значение, содержащееся в аккумуляторе как результат операции загрузки, для дальнейшей обработки. Однако Вы не можете передать значение из аккумулятора в слово таймера.

**Указание**

Когда Вы считываете слово таймера с помощью операции загрузки L, Вы получаете значение между 0 и 999. Однако Вы не получаете базиса времени, который было загружен вместе со значением времени.

**Загрузка значения счетчика**  
 из слова счетчика и загрузить в том же формате в младшее слово AKKU 1:

Значение счетчика хранится в слове счетчика в двоичном коде. С помощью следующей операции загрузки L Вы можете это двоичное значение прочитать  
  
 L <слово счетчика>  
  
 Этот вид загрузки называют прямой загрузкой значения счетчика.

AWL	Объяснение
L Z1	Загрузить в двоичном коде значение счетчика Z1 прямо в AKKU1-L.

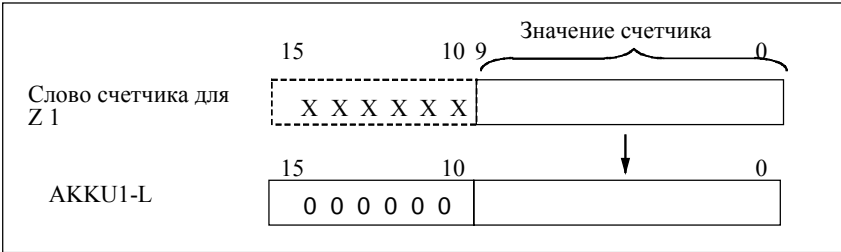


Рис. 14-3. Загрузка значения счетчика в AKKU 1 с помощью операции L

Вы можете использовать значение, содержащееся в аккумуляторе как результат операции загрузки L, для дальнейшей обработки. Однако Вы не можете передать значение из аккумулятора в слово счетчика. Если Вы хотите запустить счетчик с определенным значением, то для этого Вам нужна соответствующая операция установки счетчика (см. гл. 13.2).



14.5. Загрузка значений времени и счетчика в формате BCD

Загрузка значения  
времени в  
формате BCD  
АККУ 1:

Значение времени хранится в слове таймера в двоичном коде. С помощью следующей операции загрузки можно считать значение времени из слова таймера в двоично-десятичном (BCD) коде и загрузить его в младшее слово

LC <слово таймера>

Кроме значения времени загружается также и базис времени. Значение, содержащееся в младшем слове АККУ 1 как результат операции LC, имеет как раз тот формат, который необходим для запуска таймера. Этот вид загрузки называют прямой загрузкой значения времени в формате BCD.

Значение времени в слове таймера уменьшается с начального значения до “0”. Применяя операцию загрузки LC со словом таймера в качестве операнда, Вы получаете значение между стартовым временем слова таймера и “0”. Время, прошедшее с момента старта, вычисляется как разность между стартовым временем и временем, считанным в данный момент.

AWL	Объяснение
LC T1	Загрузить время и базис времени из T1 в формате BCD в АККУ1–L.

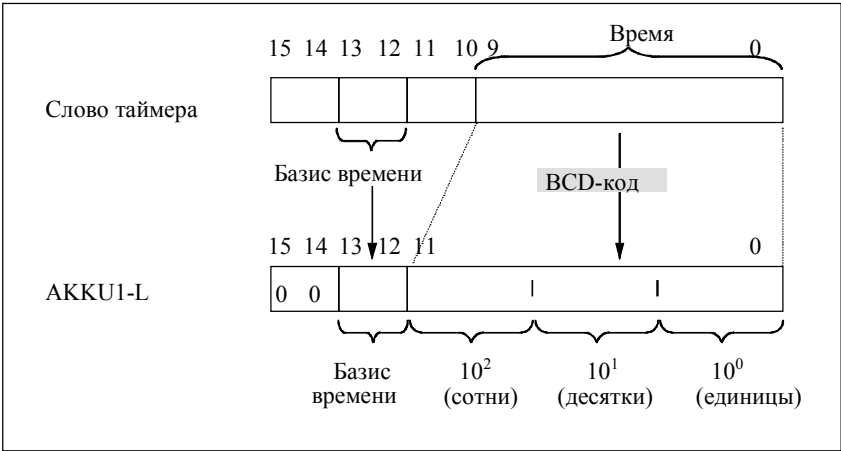


Рис. 14-4. Загрузка значения времени в АККУ 1 с помощью операции LC

Значение, содержащееся в аккумуляторе как результат операции LC, может быть использовано для дальнейшей обработки, например, для передачи значения на выходы для управления дисплеем. Однако Вы не можете значение из аккумулятора передать в слово таймера.

**Загрузка значения счетчика в формате BCD**  
**АККУ 1:**

Значение счетчика хранится в слове счетчика в двоичном коде. С помощью следующей операции загрузки можно считать значение счетчика из слова счетчика в двоично-десятичном (BCD) коде и загрузить его в младшее слово

LC <слово счетчика>

Этот вид загрузки называют прямой загрузкой значения счетчика в формате BCD. Значение, содержащееся в младшем слове АККУ 1 как результат операции LC, имеет как раз тот формат, который необходим для запуска счетчика.

Значение счетчика хранится в слове счетчика в двоичном коде. Вы можете загрузить значение счетчика в младшее слово АККУ 1 в двоично-десятичном (BCD) коде (в формате BCD, см. рис. 14–5). С помощью операции LC Вы можете считать значение счетчика в формате BCD.

AWL	Объяснение
LC Z1	Загрузить значение счетчика Z1 в формате BCD прямо в АККУ1–L.

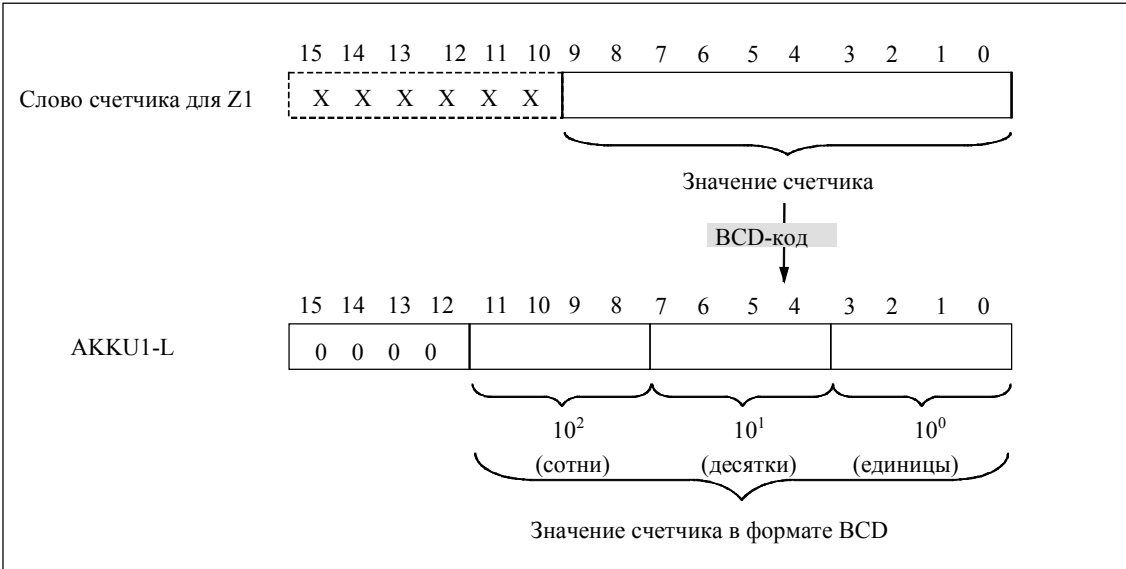


Рис. 14-5. Загрузка значения счетчика в АККУ 1 с помощью операции LC

Значение, содержащееся в аккумуляторе как результат операции LC, может быть использовано для дальнейшей обработки, например, для передачи значения на выходы для управления дисплеем. Однако Вы не можете значение из аккумулятора передать в слово счетчика.

## 14.6. Загрузка и передача между адресными регистрами

### Описание

С помощью следующих операций Ваша программа дает возможность CPU обменивать данные между адресными регистрами или обменивать содержимое двух регистров:

Операция	Значение
LAR1	Загружает содержимое адресуемой области в адресный регистр 1. Если операнд не указан, то LAR1 загружает содержимое АККУ 1 в адресный регистр 1. LAR1 может также использовать в качестве операнда AR 2, т.е. LAR1 может загрузить содержимое AR 2 в AR 1.
LAR2	Загружает содержимое адресуемой области в адресный регистр 2. Если операнд не указан, то LAR2 загружает содержимое АККУ 1 в адресный регистр 2.
TAR1	Передает содержимое адресного регистра 1 в приемник, к которому обращается операция. Если операнд не указан, то TAR1 передает содержимое адресного регистра 1 в АККУ 1. TAR1, кроме того, использовать в качестве операнда AR 2, т.е. TAR1 может передать содержимое AR 1 в AR 2.
TAR2	Передает содержимое адресного регистра 2 в приемник, к которому обращается операция. Если операнд не указан, то TAR2 передает содержимое адресного регистра 2 в АККУ 1.
TAR	Обменивает содержимое AR 1 с содержимым AR 2.

### Непосредственная адресация

Операции LAR1 и LAR2 могут обращаться к 32-битным константам. Этот вид адресации называют непосредственной адресацией (см. гл. 9.1). Непосредственный адрес применяется для непосредственной загрузки 32- битного указателя в адресный регистр (см. табл. 14–5).

LAR1 P#{область}, байт{.бит}  
с {область} = {E, A, M, D, DX, L}  
байт = 0 до 65 535  
{.бит} = 0 до 7

Таблица 14–5. LAR1 и LAR2: непосредственная адресация	
Пример	Описание
LAR1 P#E 0.0	Загружает указатель с информацией об области памяти P#E 0.0 в адресный регистр 1.
LAR2 P#0.0	Загружает внутризонный указатель с адресом 0 в адресный регистр 2.
LAR1 P##Start	Загружает указатель на локальную переменную (Start), содержащую информацию об области памяти в адресный регистр 1.

### Прямая адресация

С операциями LAR1, LAR2, TAR1 и TAR2 можно использовать прямую адресацию.

Таблица 14–6. LAR1, LAR2, TAR1, TAR2: прямая адресация

Операция	Прямая адресация	Обозначение операнда и диапазон
LAR1	{ПУСТО} <sup>1)</sup> или AR 2	DBD DID LD от 0 до 65 532 MD
LAR2	{ПУСТО} <sup>1)</sup>	
TAR1	{ПУСТО} <sup>2)</sup> или AR 2	DBD DID LD от 0 до 65 532 MD
TAR2	{ПУСТО} <sup>2)</sup>	

- 1) {ПУСТО} Если операнд не указан, то LAR1/LAR2 загружает содержимое АККУ 1 в адресный регистр.  
2) {ПУСТО} Если операнд не указан, то TAR1/TAR2 передает содержимое адресного регистра в АККУ 1.

## 14.7. Загрузка данных из блока данных

С помощью операции загрузки можно загрузить в АККУ 1 длину или номер блока данных. В таблице 14–7 приведен обзор для загрузки этого вида. Более подробную информацию о загрузке длины или номера блока данных в АККУ 1 Вы найдете в главе 21.3.

Таблица 14–7. Загрузка длины или номера блока данных в АККУ 1	
Операнд	Объяснение
DBLG	Загружает длину (в байтах) глобального блока данных в АККУ 1.
DILG	Загружает длину (в байтах) экземпляра блока данных в АККУ 1.
DBNO	Загружает номер глобального блока данных АККУ 1.
DINO	Загружает номер экземпляра блока данных в АККУ 1.