

## Обзор главы

В разделе	Вы найдете	на стр.
16.1	Основные арифметические операции	16–2
16.2	Образование абсолютной величины числа с плавающей точкой	16–6
16.3	Расширенные арифметические операции	16–7
16.4	Образование квадрата или квадратного корня числа с плавающей точкой	16–9
16.5	Образование натурального логарифма числа с плавающей точкой	16–11
16.6	Образование экспоненциального значения числа с плавающей точкой	16–12
16.7	Образование тригонометрических функций углов как чисел с плавающей точкой	16–13

## 16.1. Основные арифметические операции

### Описание

В таблице 16–1 перечислены операции AWL, с помощью которых Вы можете складывать, вычитать, умножать и делить числа с плавающей точкой (32 бита, IEEE-FP). Так как числа с плавающей точкой (32 бита, IEEE-FP) принадлежат типу данных REAL (вещественные), то в качестве мнемонического сокращения для этой операции используется “R”.

Таблица 16–1. Арифметические операции: основные арифметические операции для чисел с плавающей точкой (32 бита, IEEE-FP)	
Операция	Функция
+R	Складывает числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и 2 и сохраняет 32-битный результат в АККУ 1.
–R	Вычитает число с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 из числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 2 и сохраняет 32-битный результат в АККУ 1.
*R	Умножает число с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 на число с плавающей точкой (32 бита, IEEE-FP) в АККУ 2 и сохраняет 32-битный результат АККУ 1.
/R	Делит число с плавающей точкой (32 бита, IEEE-FP) в АККУ 2 на число с плавающей точкой (32 бита, IEEE-FP) в АККУ 1. 32-битный результат сохраняется в АККУ 1.

### Связь между арифметическими операциями и аккумуляторами

Описание функций в таблице 16–1 показывает, что арифметические операции соединяют между собой содержимое АККУ 1 и 2. Результат сохраняется в АККУ 1. Старое содержимое АККУ 1 сдвигается в АККУ 2.

В CPU с четырьмя АККУ затем содержимое АККУ 3 копируется в АККУ 2, а содержимое АККУ 4 в АККУ 3. Старое содержимое АККУ 4 не меняется.

**Представление  
результатов при  
соединении двух  
чисел с плавающей  
CPU**

Операция сложения АККУ 1 и 2 как чисел с плавающей точкой (32 бита, IEEE-FP) (+R) указывает CPU сложить содержимое АККУ 1 и АККУ 2 и сохранить результат в АККУ 1. Эта операция заменяет старое содержимое АККУ 1. Старое содержимое АККУ 2 не меняется (см. рис. **точкой в с 2 АККУ** 16–1). Пример программы следует за рис. 16–2.

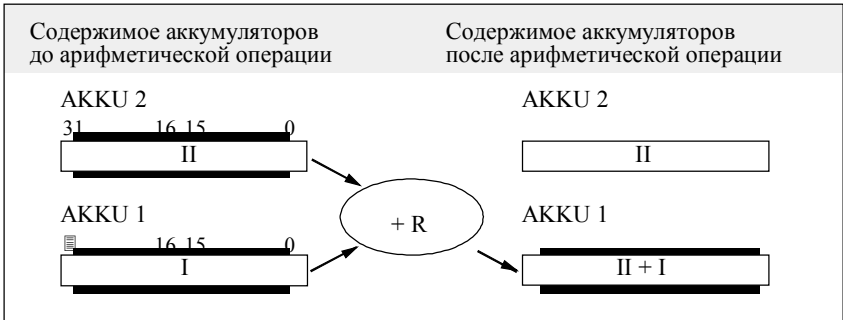


Рис. 16-1. Сложение двух чисел с плавающей точкой (IEEE-FP)

Такая же схема действует и для остальных операций арифметики с плавающей точкой.

Таблица 16–2. Результат операций для ненормализованных чисел в CPU с 2 АККУ				
Операция	Входные значения		Результат	
	АККУ 1	АККУ 2	АККУ 1	АККУ 2
+R	a	b	a	b
-R	a	b	-a	b
*R	a	b	0	b
+R	a	b	FFFF	b

Представление результатов при соединении двух плавающей АККУ 1. Затем содержимое АККУ 3 копируется в АККУ 2, а содержимое с 4 АККУ

Операция сложения АККУ 1 и АККУ 2 как чисел с плавающей точкой (32 бита, IEEE-FP) (+R) указывает CPU сложить содержимое АККУ 1 и 2 и сохранить результат в АККУ 1. Эта операция заменяет старое содержимое чисел с АККУ 4 в АККУ 3 (см. рис. 16–2).

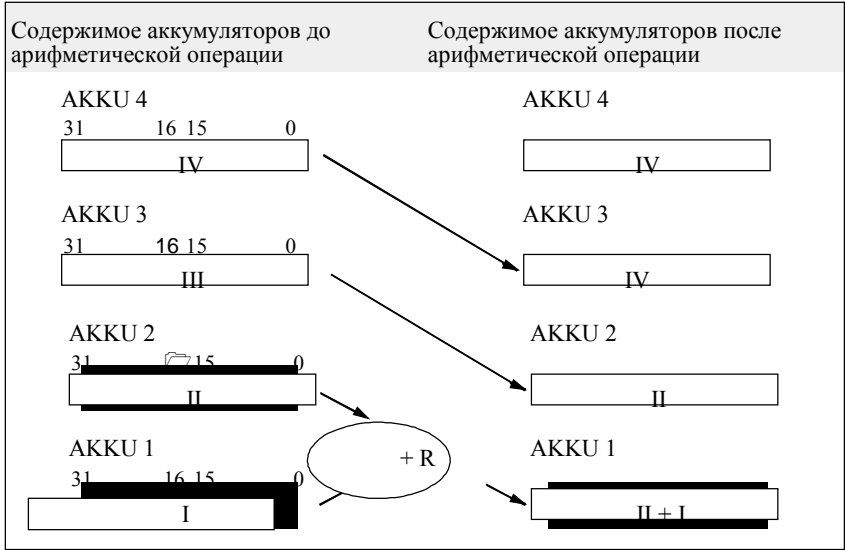


Рис. 16-2. Сложение двух чисел с плавающей точкой (IEEE-FP) в CPU с 4 АККУ

Такая же схема действует для остальных операций арифметики с плавающей точкой.

Пример

AWL	Объяснение
L MD100 L DBD4	Загрузить значение двойного меркерного слова MD100 в АККУ 1. Загрузить значение двойного слова данных DBD4 в АККУ 1. Старое содержимое АККУ 1 сдвигается в АККУ 2. (Значения в этих двойных словах должны быть в формате чисел с плавающей точкой).
+R	CPU расценивает содержимое АККУ 1 и 2 как числа с плавающей точкой (32 бита, IEEE–FP), складывает их и сохраняет результат в АККУ 1.
T DBD16	Передать содержимое АККУ 1 (результат) в двойное слово данных DBD16 (<DBD16> = <MD100> + <DBD4>).

Анализ битов в слове состояния

Арифметические операции влияют на следующие биты слова состояния:

- A1 и A0
- OV
- OS

С помощью операций из таблицы 16–5 Вы можете анализировать эти биты слова состояния. Таблица 16–3 показывает сигнальное состояние битов слова состояния для результатов арифметики с плавающей точкой внутри области допустимых значений. Прочерк (–) у одного из приведенных в таблице битов означает, что результат арифметической операции на этот бит не влияет.

Таблица 16–3. Сигнальное состояние битов в слове состояния: результат арифметической операции с числами с плавающей точкой внутри области допустимых значений

Область допустимых значений для результата операций с числами с плавающей точкой (32 бита)	Биты слова состояния			
	A1	A0	OV	OS
+0, –0 (нуль)	0	0	0	–
–3,402823E+38 < результат < –1,175494E–38 (отрицательное число)	0	1	0	–
+1,175494E–38 < результат < 3,402823E+38 (положительное число)	1	0	0	–

Таблица 16–4. Сигнальное состояние битов слова состояния: результат арифметической операции с числами с плавающей точкой вне области допустимых значений

Недопустимая область для результата операций с числами с плавающей точкой (32 бита)	Биты слова состояния			
	A1	A0	OV	OS
–1,175494E–38 < результат < –1,401298E–45 (отрицательное число) потеря значимости	0	0	1	1
+1,401298E–45 < результат < +1,175494E–38 (положительное число) потеря значимости	0	0	1	1
результат < –3,402823E+38 (отрицательное число) переполнение	0	1	1	1
результат > 3,402823E+38 (положительное число) переполнение	1	0	1	1

Таблица 16–5. Операции, анализирующие биты A1, A0, OV и OS слова состояния

Операция	Ссылка на бит слова состояния или метка перехода	Анализируемые биты в слове состояния (помечены X)	Глава в этом руководстве
U,O,X,UN,ON,XN	>0, <0, <=0, >=0, <=0, ==0, UO, OV, OS	A1, A0, OV, OS	11.3
SPO	<метка перехода>	OV	22.4
SPS	< метка перехода >	OS	22.4
SPU	< метка перехода >	A1 и A0	22.4
SPZ	< метка перехода >	A1 и A0	22.5
SPN	< метка перехода >	A1 и A0	22.5
SPP	< метка перехода >	A1 и A0	22.5
SPM	< метка перехода >	A1 и A0	22.5
SPMZ	< метка перехода >	A1 и A0	22.5
SPPZ	< метка перехода >	A1 и A0	22.5

## 16.2. Образование абсолютной величины числа с плавающей точкой

### Описание

С помощью операции ABS (Образовать абсолютное значение числа с плавающей точкой (32 бита, IEEE-FP)) Вы можете получить абсолютную величину числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1. Абсолютная величина - это неотрицательное число, числовое значение которого равно данному числу с плавающей точкой. Для абсолютной величины знак числа (+ или -) не имеет значения, так, например, 5 - абсолютная величина +5 и -5

### Пример

Следующая программа дает пример операции ABS:

AWL	Объяснение
L DBD0	Загрузить значение из двойного слова данных DBD0 в АККУ 1.
L +12.3E+00	Загрузить значение +12.3E+00 в АККУ 1.
/R	Старое содержимое АККУ 1 сдвигается в АККУ 2. CPU делит содержимое АККУ 2 на содержимое АККУ 1 и сохраняет результат в АККУ 1. передать содержимое АККУ 1 (результат) в двойное меркерное слово MD20 (<MD20> = <DBD0> / 12.3).
T MD20	Инвертировать IEEE-число с плавающей точкой в АККУ 1 (см. гл. 18.4).
NEGR	
T MD24	Передать результат из АККУ 1 в двойное меркерное слово MD24 (<MD24> = [-1] * <MD20>).
ABS	CPU образует абсолютную величину IEEE-числа с плавающей точкой в АККУ 1.
T MD28	Передать абсолютную величину из АККУ 1 в двойное меркерное слово MD28 (<MD28> = ABS[<MD20>]).

## 16.3. Расширенные арифметические операции

### Описание

Таблица 16–6 перечисляет операции AWL, с помощью которых Вы можете применить расширенные арифметические операции к числам с плавающей точкой (32 бита, IEEE-FP).

Таблица 16–6. Расширенные арифметические операции для чисел с плавающей точкой (32 бита, IEEE-FP)

Операция	Функция
SQRT	Вычисляет квадратный корень числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.
SQR	Вычисляет квадрат числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.
LN	Вычисляет натуральный логарифм числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.
EXP	Вычисляет экспоненциальное значение числа с плавающей точкой (32 бита, IEEE-FP) для основание $e$ и сохраняет 32-битный результат в АККУ 1.
SIN	Вычисляет синус числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.
COS	Вычисляет косинус числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.
TAN	Вычисляет тангенс числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.
ASIN	Вычисляет арксинус числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.
ACOS	Вычисляет арккосинус числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.
ATAN	Вычисляет арктангенс числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.

### Связь между расширенными арифметическими операциями и аккумуляторами

Расширенные арифметические операции работают только с АККУ 1. Значение, к которому обращается операция, ожидает в АККУ 1. Результат сохраняется в АККУ 1; предыдущее содержимое АККУ 1 заменяется. Содержимое АККУ 2, АККУ 3 и АККУ 4 не меняется.

**Влияние  
расширенных  
на биты  
слова состояния**

CPU выполняет расширенные математические операции, приведенные в таблице 16–1, не принимая во внимание и не изменяя результат **арифметических** логической операции. Расширенные арифметические операции влияют на **операций** следующие биты:

- A1 и A0
- OV
- OS

В таблице 16–7 перечислены операции AWL, с помощью которых Вы можете анализировать эти биты (см. гл. 11.3 и “Анализ битов состояния“ в данной главе).

Таблица 16–7. Расширенные арифметические операции для чисел с плавающей точкой (32 бита, IEEE–FP)

Операция	Ссылка на биты в слове состояния или метка перехода	Анализируемые биты слова состояния	Глава в этом руководстве
U, O, X, UN, ON, XN	>0, <0, <>0, >=0, <=0, ==0, UO, OV, OS	A1, A0, OV, OS	11.3
SPO	<метка перехода>	OV	22.4
SPS	< метка перехода >	OS	22.4
SPU	< метка перехода >	A1 и A0	22.4
SPZ	< метка перехода >	A1 и A0	22.5
SPN	< метка перехода >	A1 и A0	22.5
SPP	< метка перехода >	A1 и A0	22.5
SPM	< метка перехода >	A1 и A0	22.5
SPMZ	< метка перехода >	A1 и A0	22.5
SPPZ	< метка перехода >	A1 и A0	22.5



## 16.4. Образование квадрата или квадратного корня числа с плавающей точкой

### Описание

Операция SQR (квадрат) вычисляет квадрат числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1.

Операция SQR заменяет старое содержимое АККУ 1; содержимое АККУ 2, АККУ 3 и АККУ 4 не меняется.

Операция SQRT (квадратный корень) вычисляет квадратный корень числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1. Входное значение должно быть больше или равно нулю. Операция SQRT заменяет старое содержимое АККУ 1; содержимое АККУ 2, АККУ 3 и АККУ 4 не меняется.

Эти операции выдают положительный результат, если все операнды больше "0". Единственное исключение: квадратный корень из -0 равен -0.

### Влияние на биты A1, A0, OV и OS слова состояния

Таблица 16–8. Влияние операции SQR на биты A1, A0, OV и OS				
Результат в АККУ 1	A1	A0	OV	OS
+ qNaN	1	1	1	1
+ бесконечность (переполнение)	1	0	1	1
+ нормализован	1	0	0	-
+ ненормализован (потеря значимости)	0	0	1	1
+ нуль	0	0	0	-
- qNaN	1	1	1	1

### Пример

Следующий отрезок программы показывает на примере применение операции SQR.

AWL	Объяснение
AUF DB17	Открыть блок данных DB17. (Пусть он содержит входное значение и служит для сохранения результата)
L DBD0	Загрузить значение из двойного слова данных DBD0 в АККУ 1. (Это значение должно быть в формате числа с плавающей точкой.)
SQR	Вычислить квадрат числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1. Сохранить результат в АККУ 1.
UN OV	Опросить бит состояния OV на 0.
SPB OK	Если при выполнении операции SQR не произошло ошибки, перейти на метку OK.
...	(Здесь реализуется реакция на возникновение ошибки.)
OK: T DBD4	Передать результат из АККУ 1 в двойное слово данных DBD4.

Таблица 16–9. Влияние операции SQRT на биты A0, A1, OS и OV

Результат в АККУ 1	A1	A0	OV	OS
+ qNaN	1	1	1	1
+ бесконечность (переполнение)	1	0	1	1
+ нормализован	1	0	0	-
+ ненормализован (потеря значимости)	0	0	1	1
+ нуль	0	0	0	-
- нуль	0	0	0	-
- qNaN	1	1	1	1

## Пример

Следующий отрезок программы показывает на примере применение операции SQRT.

AWL	Объяснение
L MD10	Загрузить значение из двойного меркерного слова MD10 в АККУ 1. (Это значение должно иметь формат числа с плавающей точкой).
SQRT	Образовать квадратный корень из числа с плавающей точкой (32 бита, IEEE–FP) в АККУ 1. Сохранить результат в АККУ 1.
UN OV	Опросить бит состояния OV на 0.
SPB OK	Если при выполнении операции SQRT не произошло ошибки, то перейти на метку OK. (Здесь реализуется реакция на возникновение ошибки.)
...	Передать результат (АККУ 1) в двойное меркерное слово MD20.
OK: T MD20	

## 16.5. Образовать натуральный логарифм числа с плавающей точкой

### Описание

Операция LN (натуральный логарифм) вычисляет натуральный логарифм числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 и сохраняет 32-битный результат в АККУ 1. Входное значение должно быть больше нуля. Операция LN заменяет старое содержимое АККУ 1; содержимое АККУ 2, АККУ 3 и АККУ 4 не меняется.

### Влияние на биты и OS состояния

Таблица 16–10 описывает влияние, которое операция LN оказывает на **A1, A0, OV** сигнальное состояние битов A1, A0, OV и OS слова состояния. “-” в таблице **слова** указывает на то, что на соответствующий бит влияние не оказывается.

Таблица 16–10. Влияние операции LN на биты состояния A1, A0, OV и OS				
Результат в АККУ 1	A1	A0	OV	OS
+ qNaN	1	1	1	1
+ бесконечность (переполнение)	1	0	1	1
+ нормализован	1	0	0	-
+ ненормализован (потеря значимости)	0	0	1	1
+ нуль	0	0	0	-
- нуль	0	0	0	-
- ненормализован (потеря значимости)	0	0	1	1
- нормализован	0	1	0	-
- бесконечность (переполнение)	0	1	1	1
- qNaN	1	1	1	1

### Пример

Следующий отрезок программ показывает пример применения операции LN.

AWL	Объяснение
L MD10	Загрузить значение из двойного меркерного слова MD10 в АККУ 1. (Это значение должно иметь формат числа с плавающей точкой).
LN	Образовать натуральный логарифм числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1. Сохранить результат в АККУ 1.
UN OV	Опросить бит состояния OV на 0.
SPB OK	Если при выполнении операции LN не произошла ошибка, то перейти на метку OK.
...	(Здесь реализуется реакция на возникновение ошибки.)
OK: T MD20	Передать результат из АККУ 1 в двойное меркерное слово MD20.

## 16.6. Образование экспоненциального значения числа с плавающей точкой

**Описание** Операция EXP (экспоненциальное значение для основание  $e$ ) вычисляет экспоненциальное значение числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1 для основания  $e$  ( $= 2,71828...$ ) и сохраняет 32-битный результат в АККУ 1. Операция EXP заменяет старое содержимое АККУ 1; содержимое АККУ 2, АККУ 3 и АККУ 4 не меняется.

**Влияние на биты и OS состояния** Таблица 16–11 описывает влияние, которое операция EXP оказывает на **A1, A0, OV** сигнальное состояние битов A1, A0, OV и OS слова состояния. “-” в таблице **слова** указывает на то, что на соответствующий бит влияние не оказывается.

Таблица 16–11. Влияние операции EXP на биты A1, A0, OV и OS слова состояния

Результат в АККУ 1	A1	A0	OV	OS
+ qNaN	1	1	1	1
+ бесконечность (переполнение)	1	0	1	1
+ нормализован	1	0	0	-
+ ненормализован (потеря значимости)	0	0	1	1
+ нуль	0	0	0	-
- qNaN	1	1	1	1

**Пример** Следующий отрезок программы показывает пример применения операции EXP.

AWL	Объяснение
L MD10	Загрузить значение из двойного меркерного слова MD10 в АККУ 1. (Это значение должно иметь формат числа с плавающей точкой.)
EXP	Образовать экспоненциальное значение для основания $e$ числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1. Сохранить результат в АККУ 1.
UN OV	Опросить бит состояния OV на 0.
SPB OK	Если при выполнении операции EXP не произошла ошибка, то перейти на метку OK.
...	(Здесь реализуется реакция на возникновение ошибки.)
OK: T MD20	Передать результат из АККУ 1 в двойное меркерное слово MD20.

## 16.7. Образование тригонометрических функций углов как чисел с плавающей точкой

### Описание

С помощью следующих операций Вы можете образовать тригонометрические функции углов, представленных в виде чисел с плавающей точкой (32 бита, IEEE-FP). 32-битный результат сохраняется в АККУ 1, содержимое АККУ 2, АККУ 3 и АККУ 4 не меняется.

Операция	Значение
SIN	Образование синуса угла, заданного в дуговых единицах. Угол хранится в АККУ 1 как число с плавающей точкой.
ASIN	Образование арксинуса числа с плавающей точкой в АККУ 1. Результатом является угол, заданный в дуговых единицах. Значение лежит в следующей области: $-\pi / 2 \leq \text{арксинус (АККУ 1)} \leq +\pi / 2$ , где $\pi = 3.14...$
COS	Образование косинуса угла, заданного в дуговых единицах. Угол хранится в АККУ 1 как число с плавающей точкой.
ACOS	Образование арккосинуса числа с плавающей точкой в АККУ 1. Результатом является угол, заданный в дуговых единицах. Значение лежит в следующей области: $0 \leq \text{арккосинус (АККУ 1)} \leq \pi$ , где $\pi = 3.14...$
TAN	Образование тангенса угла, заданного в дуговых единицах. Угол хранится в АККУ 1 как число с плавающей точкой.
ATAN	Образование арктангенса числа с плавающей точкой в АККУ 1. Результатом является угол, заданный в дуговых единицах. Значение лежит в следующей области: $-\pi / 2 \leq \text{арктангенс (АККУ 1)} \leq +\pi / 2$ , где $\pi = 3.14...$

### Влияние на биты OS

#### слова состояния

В таблице 16–12 показано влияние операций SIN, ASIN, COS, ACOS и A1, A0, OV и ATAN на сигнальное состояние битов A1, A0, OV и OS в слове состояния.

В таблице 16–13 показано, как на эти биты влияет операция TAN. “-” указывает, что на соответствующие биты влияние не оказывается.

Таблица 16–12. Влияние операций SIN, ASIN, COS, ACOS и ATAN

Результат в АККУ 1	A1	A0	OV	OS
+ qNaN	1	1	1	1
+ нормализован	1	0	0	-
+ ненормализован (переполнение)	0	0	1	1
+ нуль	0	0	0	-
- нуль	0	0	0	-
- ненормализован (потеря значимости)	0	0	1	1
- нормализован	0	1	0	-
- qNaN	1	1	1	1

Таблица 16–13. Влияние операции TAN на биты A1, A0, OV и OS слова состояния

Результат в АККУ 1	A1	A0	OV	OS
+ qNaN	1	1	1	1
+ бесконечность (переполнение)	1	0	1	1
+ нормализован	1	0	0	-
+ ненормализован (потеря значимости)	0	0	1	1
+ нуль	0	0	0	-
- нуль	0	0	0	-
- ненормализован (потеря значимости)	0	0	1	1
- нормализован	0	1	0	-
- бесконечность (переполнение)	0	1	1	1
- qNaN	1	1	1	1

**Пример** Следующий отрезок программы показывает пример применения операции SIN.

AWL	Объяснение
L MD10	Загрузить значение из двойного меркерного слова MD10 в АККУ 1. (Это значение должно иметь формат числа с плавающей точкой.)
SIN	Образовать синус числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1. Сохранить результат в АККУ 1.
T MD20	Передать результат из АККУ 1 в двойное меркерное слово MD20.

**Пример** Следующий отрезок программы показывает пример применения операции ASIN.

AWL	Объяснение
L MD10	Загрузить значение из двойного меркерного слова MD10 в АККУ 1. (Это значение должно иметь формат числа с плавающей точкой.)
ASIN	Образовать арксинус числа с плавающей точкой (32 бита, IEEE-FP) в АККУ 1. Сохранить результат в АККУ 1.
UN OV	Опросить бит состояния OV на 0.
SPB OK	Если при выполнении операции ASIN не произошла ошибка, то перейти на метку OK.
...	(Здесь реализуется реакция на возникновение ошибки.)
OK: T MD20	Передать результат из АККУ 1 в двойное меркерное слово MD20.

## Анализ битов в слове состояния

Расширенные арифметические операции влияют на следующие биты в слове состояния:

- A1 и A0
- OV
- OS

## Результат действителен

Прочерк (-) у одного из битов, представленных в таблице, означает, что на этот бит арифметические операции не оказывают влияния.

Таблица 16–14. Аркфункции для чисел с плавающей точкой (32 бита, IEEE–FP) и допустимые области значений для входной величины

Допустимая область результата для операций над числами с плавающей точкой (32 бита)	Бит в слове состояния			
	A1	A0	OV	OS
+0, -0 (нуль)	0	0	0	-
-3,402823E+38 < результат < -1,175494E-38 (отрицательное число)	0	1	0	-
+1,175494E-38 < результат < 3,402823E+38 (положительное число)	1	0	0	-

## Результат недействителен

Таблица 16–15. Сигнальные состояния битов в слове состояния: результат арифметической операции над числами с плавающей точкой вне области допустимых значений

Недопустимая область для результата операций над числами с плавающей точкой (32 бита)	A1	A0	OV	OS
-1,175494E-38 < результат < -1,401298E-45 (отрицательное число) потеря значимости	0	0	1	1
+1,401298E-45 < результат < +1,175494E-38 (положительное число) потеря значимости	0	0	1	1
результат < -3,402823E+38 (отрицательное число) переполнение	0	1	1	1
результат > 3,402823E+38 (положительное число) переполнение	1	0	1	1

## Входное значение недопустимо

Таблица 16–16. Сигнальные состояния битов в слове состояния: входное значение не число с плавающей точкой или вне допустимой области значений

Недопустимая область входного значения для чисел с плавающей точкой (32 бита)	A1	A0	OV	OS
В АККУ 1 не 32-битное число с плавающей точкой (формат IEEE–FP)	1	1	1	1
Недопустимая операция: входная величина в АККУ 1 вне области допустимых значений	1	1	1	1