

**ИНТЕГРИРОВАННАЯ СРЕДА РАЗРАБОТКИ  
И ОТЛАДКИ ПРОГРАММ MC STUDIO**

**Инструменты ядра ELcore**

**(ПРИЛОЖЕНИЕ 1)  
СИСТЕМА ИНСТРУКЦИЙ**

**14.09.2012**

## Порядок использования настоящей документации

Настоящая документация охраняется действующим законодательством Российской Федерации об авторском праве и смежных правах, в частности, законом Российской Федерации «Об авторском праве и смежных правах». ОАО НПЦ «ЭЛВИС» является единственным правообладателем исключительных авторских прав на настоящую документацию.

Настоящую документацию без предварительного согласия ОАО НПЦ «ЭЛВИС» запрещается:

- воспроизводить, т.е. изготавливать один или более экземпляров настоящей документации, ее части, в любой форме, любым способом;
- сдавать в прокат;
- публично показывать, исполнять или сообщать для всеобщего сведения;
- переводить;
- переделывать или другим образом перерабатывать (дорабатывать).

ОАО НПЦ «ЭЛВИС» оставляет за собой право в любой момент вносить изменения (дополнения) в настоящую документацию без предварительного уведомления о таком изменении (дополнении).

ОАО НПЦ «ЭЛВИС» не несет ответственности за вред, причиненный при использовании настоящей документации.

Передача настоящей документации не означает передачи каких-либо авторских прав ОАО НПЦ «ЭЛВИС» на нее.

Возникновение каких-либо прав на материальный носитель, на котором передается настоящая документация, не влечет передачи каких-либо авторских прав на данную документацию.

Все указанные в настоящей документации товарные знаки принадлежат их владельцам.

ОАО НПЦ «ЭЛВИС» ©, 2012

## **Аннотация**

В настоящем документе приведено описание архитектуры DSP-кластера DElcore-26, в состав которого входят два DSP-ядра Elcore-26, и системы инструкций DSP-ядра Elcore-26. Для задач сигнальной обработки в микропроцессоре 1892BM8Я используется DSP-ядро ELcore-26, а в микропроцессорах 1892BM4Я и 1892BM5Я – DSP-кластер DELcore-26.

## СОДЕРЖАНИЕ

<b>1. АРХИТЕКТУРА DSP-КЛАСТЕРА DELCORE-26</b>	<b>6</b>
1.1 Основные технические характеристики DSP-кластера DELcore-26	6
1.2 Структурная схема	6
1.3 Организация памяти	8
1.4 Буфер обмена	9
<b>2. ПРОГРАММНАЯ МОДЕЛЬ DSP-ЯДРА ELCORE-26</b>	<b>10</b>
2.1 Программно-доступные регистры	10
2.1.1 Регистры ALU	12
2.1.2 Регистры AGU, AGU-Y	15
2.1.3 Регистры PCU	15
2.2 Организация памяти	20
2.2.1 Организация программной памяти	20
2.2.2 Организация памяти данных	20
2.2.3 Организация обменов с памятью данных	21
2.2.4 Особенности адресации памяти данных в режимах SCALAR и SIMD	22
<b>3. ФОРМАТЫ И ТИПЫ ДАННЫХ</b>	<b>23</b>
3.1 Общие положения	23
3.2 Форматы данных для чисел с фиксированной точкой	24
3.2.1 Короткий 16-разрядный формат (short)	24
3.2.2 Длинный 32-разрядный формат (long)	25
3.2.3 Длинный двойной 64-разрядный формат ( <code>_int64</code> )	26
3.2.4 Комплексный 16-разрядный формат (X16)	27
3.2.5 Комплексный 8-разрядный формат (X8)	28
3.2.6 Граничные значения для типов данных с фиксированной точкой	30
3.2.7 Особенности выполнения арифметических операций для целых и дробных чисел	32
3.3 Форматы данных для чисел с плавающей точкой	33
3.3.1 Формат 24E8 (min float)	33
3.3.2 Формат 32E16 (min double)	35
3.4 Запись различных типов констант в операндах и памяти	37
<b>4. ПРИЗНАКИ РЕЗУЛЬТАТА ОПЕРАЦИИ</b>	<b>38</b>
<b>5. ОБЩАЯ ХАРАКТЕРИСТИКА СИСТЕМЫ ИНСТРУКЦИЙ</b>	<b>39</b>
5.1 Вычислительные команды	40
5.1.1 Команды сложения/вычитания в форматах с фиксированной точкой	41
5.1.2 Команды умножения/накопления в форматах с фиксированной точкой	42
5.1.3 Команды сдвига в форматах с фиксированной точкой	42
5.1.4 Другие арифметические команды в форматах с фиксированной точкой	43
5.1.5 Округление, преобразования форматов, упаковка/распаковка	44
5.1.6 Логические команды, операции с битами и битовыми полями	44
5.1.7 Команды для обработки данных в формате с плавающей точкой 24E8	45
5.1.8 Команды для обработки данных в формате с плавающей точкой 32E16	45
5.2 Команды пересылок	46
5.3 Команды программного управления	46
<b>6. ФОРМАТЫ ИНСТРУКЦИЙ</b>	<b>47</b>
6.1 Описание форматов инструкций	48
6.2 Два типа операций (OP1 и OP2)	54
6.3 Коды операций	59
6.3.1 Операции с несколькими вариантами кодировки	60
6.3.2 Расширение кода операции за счет формата инструкции и бита #	60
6.4 Таблица полей форматов инструкций	61
6.5 Описание полей форматов инструкций	64
6.5.1 Основное и дополнительное поле кода формата инструкции	65
6.5.2 Основное и дополнительное поле кода операции	65
6.5.3 Поле кода условия (cc)	66
6.5.4 Поля адресов операндов в вычислительных операциях (S/s/D/d)	66
6.5.5 Поля адресов операндов в операциях пересылок (R/RC/RS/Rs/RD/Rd). Поля выбора типа регистра (sr, sc)	67

6.5.6	Поле формата пересылки ( <i>L</i> ).....	69
6.5.7	Поле направления пересылки ( <i>de</i> ).....	69
6.5.8	Поле режима адресации AGU ( <i>mode</i> ) и поле кода блокировки обмена ( <i>u</i> ).....	69
6.5.9	Поле регистра адреса AGU ( <i>A</i> ).....	70
6.5.10	Поле режима адресации AGU-Y ( <i>AT</i> ).....	70
6.5.11	Поля непосредственных операндов ( <i>#5</i> , <i>#16</i> , <i>#32</i> ) и поле выбора непосредственного 5-разрядного операнда ( <i>#</i> ).....	71
6.5.12	Поле выбора режима <i>Scaling</i> ( <i>M</i> ).....	71
<b>7.</b>	<b>СИНТАКСИС И ОПИСАНИЕ ОПЕРАЦИЙ.....</b>	<b>72</b>
7.1	Общие положения.....	72
7.2	Соглашения по обозначениям.....	72
7.3	Автокод.....	74
<b>8.</b>	<b>РАБОТА ПРОГРАММНОГО КОНВЕЙЕРА И ВРЕМЯ ИСПОЛНЕНИЯ ИНСТРУКЦИЙ.....</b>	<b>610</b>
8.1	Работа программного конвейера при последовательной выборке команд.....	610
8.2	Работа программного конвейера при программных переходах.....	612
8.3	Работа программного конвейера при исполнении циклов.....	613
8.4	Работа программного конвейера при исполнении команды STOP.....	614
<b>9.</b>	<b>ОГРАНИЧЕНИЯ ПРИ ИСПОЛНЕНИИ ИНСТРУКЦИЙ.....</b>	<b>615</b>
9.1	Ограничение на адреса результатов одновременно исполняемых операций.....	615
9.2	Ограничения при исполнении инструкций программного управления.....	615
9.3	Ограничения при исполнении инструкций пересылок.....	616
<b>10.</b>	<b>ОСОБЕННОСТИ ИСПОЛНЕНИЯ ИНСТРУКЦИЙ В РЕЖИМАХ SIMD И SCALAR.....</b>	<b>617</b>
10.1	Управление режимом функционирования SIMD/SCALAR.....	617
10.2	Особенности выполнения обменов между регистрами управления и регистрами данных в режиме SIMD. 617	
10.3	Особенности исполнения условных инструкций.....	618

## 1. АРХИТЕКТУРА DSP-КЛАСТЕРА DELCORE-26

DSP-кластер DELcore-26 представляет собой симметричный мультипроцессор, в состав которого входят два DSP-ядра ELcore-26, работающих на общем поле памяти данных, а также буфер обмена XBUF.

Каждое из двух DSP-ядер ELcore-30 представляет собой ядро сопроцессора-акселератора сигнальной обработки. Оно имеет гарвардскую архитектуру с внутренним параллелизмом по потокам обрабатываемых данных и предназначено для обработки информации в форматах с фиксированной и с плавающей точкой. Система инструкций, реализующих параллельно несколько вычислительных операций и пересылок, 7-фазный программный конвейер и гибкие адресные режимы позволяют реализовать алгоритмы сигнальной обработки с высокой производительностью. Каждое DSP-ядро функционирует под управлением RISC-ядра (CPU) и расширяет его возможности по обработке сигналов.

### 1.1 Основные технические характеристики DSP-кластера DELcore-26

- 2 вычислительных ядра DSP ELcore-26;
- объем общей памяти данных 256 Кбайт (128 Кбайт на ядро);
- объем памяти программ 16 Кбайт на ядро;
- максимальная пропускная способность коммутатора ядер с памятью – 256 бит за такт;
- максимальная скорость обмена внешних устройств с памятью кластера – 64 бит за такт;
- суммарная пиковая производительность кластера:
  - 12 операций с плавающей точкой (IEEE 754) за 1 такт;
  - 16 32-разрядных операций с фиксированной точкой за 1 такт;
  - 48 16-разрядных операций с фиксированной точкой за 1 такт;
  - 72 8-разрядных операций с фиксированной точкой за 1 такт.

### 1.2 Структурная схема

Структурная схема 2-ядерного DSP-кластера DELcore-26 приведена на Рисунок 1.1.

На схеме приняты следующие обозначения:

DSP0 – DSP1 – два DSP-ядра ELcore-26;

PMEM – память программ;

XMEM – X-память данных;

YMEM – Y-память данных;

AHB – контроллер шины AMBA AHB (slave);

MEM\_EXT\_PORT, MEM\_MUX\_OUT – распределенный контроллер шины AMBA AXI (slave);

XBUF\_02 – буфер обмена (регистровый файл 32 слова по 32 разряда);

ArbBuf, MA\_LocalArb – распределенный арбитр;

DSP\_logic – вычислительное ядро;

AGU, AGU-Y – адресные генераторы памяти данных;

PAG – адресный генератор памяти программ;

PDC – программный декодер;  
 ALU – арифметико-логическое устройство;  
 RF9 – регистровый файл 32 слова по 128 разрядов, 9 портов;  
 COMM5 – коммутатор входных данных операционных устройств;  
 OP1\_unit, OP2\_unit – операционные (вычислительные) устройства;  
 CCR\_REG, PDN – регистры признаков результата операции и параметра денормализации;

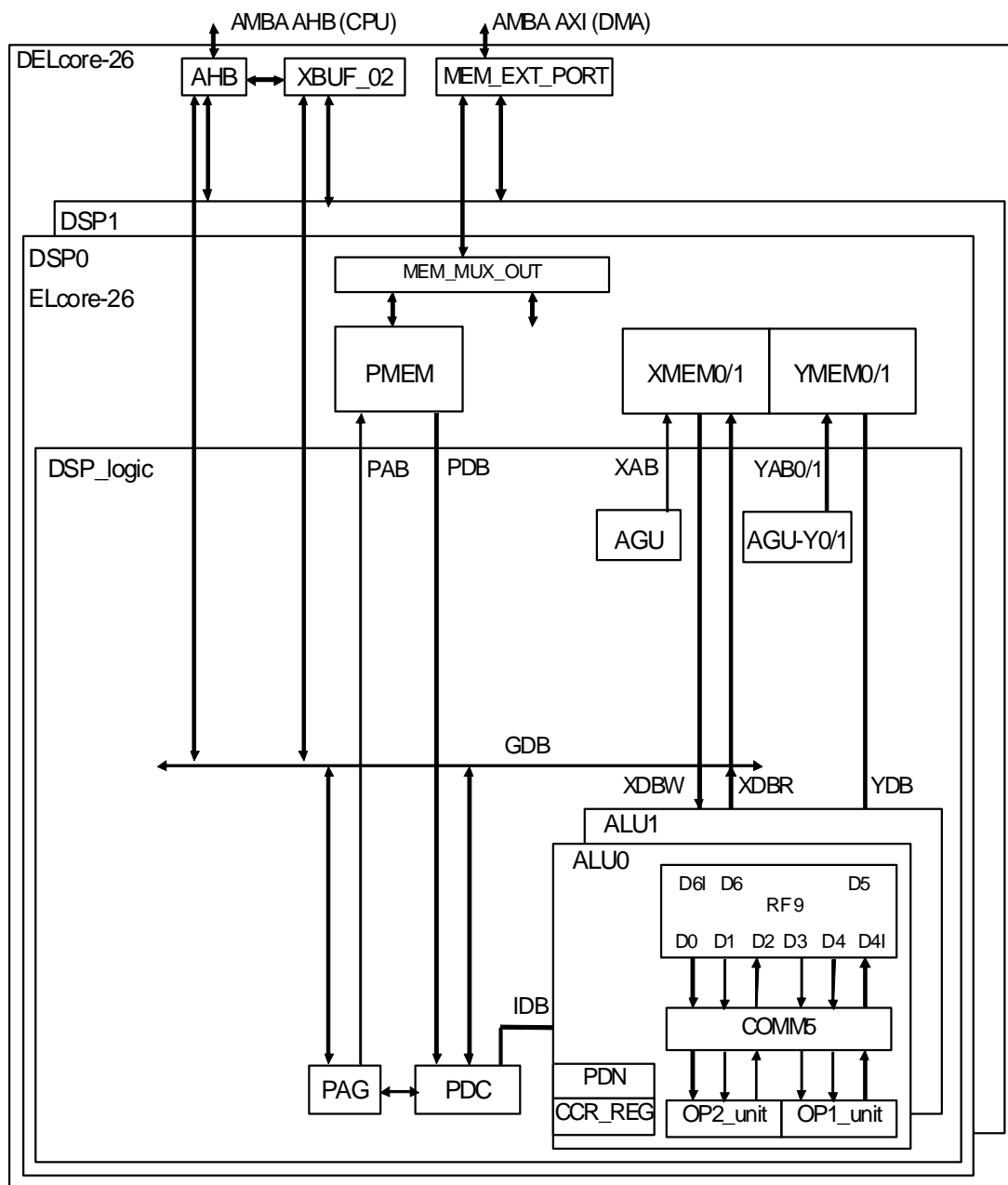


Рисунок 1.1. Структурная схема 2-ядерного DSP-кластера DELcore-26

### 1.3 Организация памяти

Карта памяти DSP0, DSP1 приведена на рисунке.

Адреса в пространстве CPU			Внутренние адреса DSP	
DSP0	DSP1			
0x187F_FFFC 0x187F_FF80	0x18BF_FFFC 0x18BF_FF80	Буфер обмена (XBUF)	0xFFFF 0xFFE0	A0-A7
0x187F_FF7C 0x1848_0180	0x18BF_FF7C 0x1888_0180	Резерв		
0x1848_017C 0x1848_0000	0x1888_017C 0x1888_0000	Регистры данных и управления		
0x1847_FFFC 0x1844_4000	0x1887_FFFC 0x1884_4000	Резерв	0xFFFF 0x1000	
0x1844_3FFC 0x1844_0000	0x1884_3FFC 0x1884_0000	Память программ (PRAM)	0x0FFF = PC_max 0x0000 = PC_min	PC
0x1843_FFFC 0x1842_0000	0x1883_FFFC 0x1882_0000	Резерв	0xFFDF 0x8000	
0x1841_FFFC 0x1841_0000	0x1881_FFFC 0x1881_0000	Память данных (YRAM)	0x7FFF = pY_max 0x4000 = pY_min	A0-A7, AT
0x1840_FFFC 0x1840_0000	0x1880_FFFC 0x1880_0000	Память данных (XRAM)	0x3FFF = pX_max 0x0000 = pX_min	A0-A7

Рисунок 1.2. Карта памяти DSP0-DSP1

Каждое из DSP-ядер имеет свою программную память (PRAM) объемом 4К 32-разрядных слов (16 Кбайт), память данных XRAM объемом 16К 32-разрядных слов YRAM объемом 16К 32-разрядных слов (всего память данных обоих ядер составляет 256 Кбайт).

Объем PRAM (DSP0) – 4К 32-разрядных слов (16 Кбайт).

Объем PRAM (DSP1) – 4К 32-разрядных слов (16 Кбайт).



Объем XRAM и YRAM – 64К 32-разрядных слов (256 Кбайт).

Для обеспечения возможности одновременного доступа к памяти программ и данных DSP как со стороны CPU (DMA), так и со стороны DSP блоки памяти XRAM, YRAM и PRAM аппаратно реализованы как 2-портовые. С внешней стороны возможны как 32-разрядные (CPU), так и 64-разрядные обращения (DMA). Со стороны DSP0–DSP1 возможны 32-разрядные или 64 обращения (чтение и запись) к памяти данных XRAM, YRAM. Программная память PRAM со стороны DSP доступна только для чтения 32/64-разрядных слов инструкций.

## 1.4 Буфер обмена

Для оперативных обменов данными между CPU, DSP0 и DSP1 вводится буфер обмена XBUF объемом 32 32-разрядных слова, доступный по записи и чтению для всех трех ядер.

Буфер обмена XBUF представляет собой многопортовую память и допускает одновременное чтение одной и той же ячейки со стороны нескольких абонентов - CPU, DSP0 и DSP1. При одновременном запросе на запись в одну и ту же ячейку приоритет отдается CPU, затем - DSP0, затем - DSP1.

Буфер обмена XBUF представляет собой векторную память. Это означает, что в режиме SIMD каждое обращение (запись или чтение) со стороны DSP происходит одновременно к двум ячейкам с соседними адресами. Организация обменов DSP с XBUF полностью аналогична организации обменов DSP с памятью данных XRAM (включая назначение бит SI, SW).

Каждое из DSP-ядер (DSP0 и DSP1) выбирает способ обмена с XBUF независимо друг от друга.

Для отображения состояния обменов через XBUF вводится регистр флагов обмена (EFR) – 32 бит, только чтение, начальное состояние EFR=0x0. Каждый разряд этого регистра формируется аппаратно и отображает тип последней транзакции, выполненной с соответствующей ячейкой XBUF (0 – чтение из XBUF, 1 – запись). Регистр EFR относится к пространству регистров управления DSP и имеет в этом пространстве адрес 00011 (sr=0, sc=1). В пространстве CPU этот регистр имеет адрес 0x1848\_010C.

Вводятся два режима обменов с XBUF – обычный и синхронный (семафорный).

В обычном режиме любой из абонентов - CPU, DSP0 и DSP1 - в любое время может обращаться к любой ячейке XBUF, и это обращение немедленно исполняется (с учетом приоритета по записи).

В синхронном режиме (устанавливается битом SNW=1 регистра CSR):

- CPU обращается к XBUF так же, как и в обычном режиме;
- обращения со стороны DSP0 и DSP1 могут выполняться с задержкой в зависимости от состояния регистра EFR и типа обращения. Если тип обращения не совпадает с типом последней транзакции, выполненной с данной ячейкой XBUF (то есть если за записью следует чтение, а за чтением - запись) то исполнение такого обращения происходит без задержки. Если же за записью вновь следует запрос на запись в ту же ячейку (либо за чтением – вновь запрос на чтение), то такое обращение выполняется с задержкой. Выдавшее запрос DSP-ядро переводится в состояние ожидания, продолжающееся до тех пор, пока соответствующий бит EFR не сменит свое значение на противоположное. В регистре DCSR введен бит WT=DCSR[4], указывающий на то, что DSP находится в состоянии ожидания при обращении к XBUF. Одновременная установка в DSP0 и DSP1 бит WT=1 (то есть зависание программы) вызывает прерывание WSE в CPU.

## 2. ПРОГРАММНАЯ МОДЕЛЬ DSP-ЯДРА ELCORE-26

Программная модель DSP-ядра включает в себя программно-доступные регистры исполнительных устройств и память. К исполнительным устройствам относятся:

- вычислительные секции (одна или две) ALU0-ALU1;
- адресные генераторы для X- и Y-памяти данных (AGU и AGU-Y);
- устройство программного управления PCU.

Память DSP-ядра Elcore-26 включает в себя два независимых пространства - память программ и память данных.

### 2.1 Программно-доступные регистры

Программно-доступные регистры Elcore-26 (включая стеки и регистровый файл) приведены на рисунке 2.1.

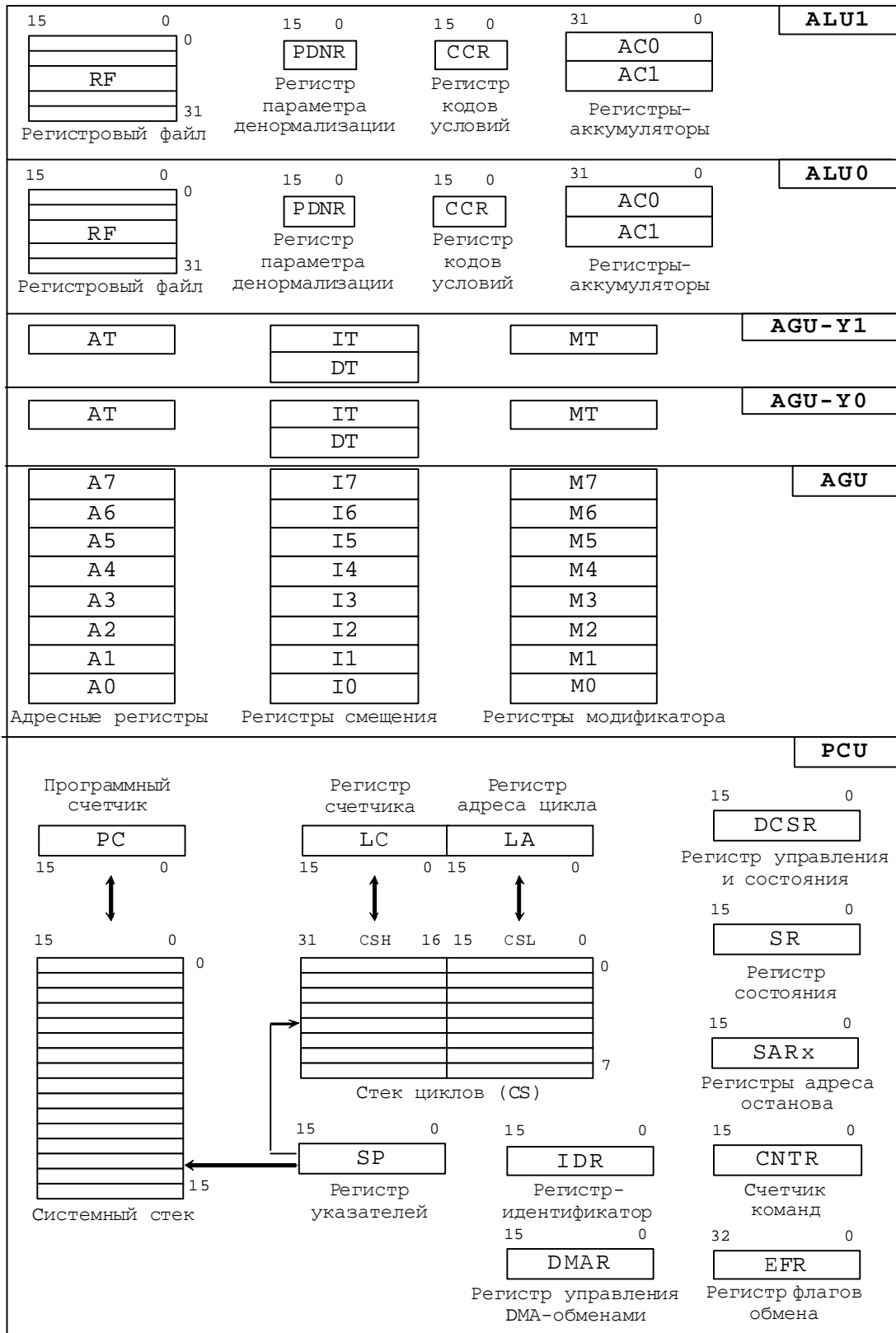


Рис.2.1 Программно-доступные регистры DSP-ядра Elcore-26

По своему назначению все регистры делятся на регистры данных, объединенные в регистровые файлы, и регистры управления (все остальные).

Кроме RF и 32-разрядных регистров-аккумуляторов EFR, AC0 и AC1, все остальные регистры – 16-разрядные. Все регистры доступны как по чтению, так и по записи, за исключением регистра IDR и младшего байта регистра SR, доступных только по чтению.

### **2.1.1 Регистры ALU**

Каждая вычислительная секция ALU содержит регистровый файл RF – реконфигурируемый массив (16x32 или 32x16) регистров данных, регистр параметра денормализации PDNR, регистр кодов условий (регистр признаков) CCR, два 32-разрядных регистра-аккумулятора AC0, AC1.

#### ***Регистровый файл***

Исходные данные и результаты всех операций ALU хранятся в регистровом файле (RF), который представляет собой реконфигурируемый массив регистров данных (32 регистра по 16-разрядов либо 16 регистров по 32 разряда).

16-разрядные регистры данных могут иметь номера с R0 по R31, а 32-разрядные регистры – только четные номера с R0 по R30. При 32-разрядных операциях четные и нечетные регистры объединяются попарно и образуют 16 32-разрядных регистров, причем младшие 16 бит представлены в регистрах с четными номерами, старшие 16 бит - в регистрах с нечетными номерами. Мнемонически для отличия четных 16-разрядных регистров от 32-разрядных к наименованию последних добавляется через точку суффикс L (long), например: R0.L.

#### ***Регистры-аккумуляторы AC0, AC1***

Регистры-аккумуляторы AC0, AC1 являются специализированными 32-разрядными регистрами данных (адресно регистры AC0, AC1 относятся к регистрам управления), предназначенными для накопления результата в операциях умножения с накоплением (MAC, MAC2, MACL, MACX, SAC2). В операциях MAC, MACL регистры AC0, AC1 объединяются в один 64-разрядный регистр для получения 64-разрядного результата.

Начальное состояние AC0 = AC1 = 0x00000000.

### Регистр PDNR

Регистр PDNR - секционный регистр управления, предназначенный для измерения параметра денормализации (PDN) и управления режимом блочной экспоненты и режимом масштабирования (Scaling).

Назначение разрядов регистра PDNR приведено ниже.

#### PDNR:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Esc	-	-	-	-	-	SC		Epdn	-	F	Cpdn				

- **Cpdn** – текущий код PDN;
- **F (X/L)** – формат анализируемой информации (0 – Long, 1 – X16);
- **Epdn** – программный признак разрешения детектирования и изменения PDN (0 – нет разрешения, 1 – разрешение);
- **SC** – величина масштабирования результата (00 – нет сдвига, 01 - сдвиг на 1 разряд, 10 - сдвиг на 2 разряда);
- **Esc** – признак разрешения масштабирования результата (0 – нет разрешения, 1 – разрешение).

Начальное состояние регистра PDNR = 0x0000.

### Регистр CCR

Регистр CCR CCR - секционный регистр, предназначенный для хранения признаков результатов вычислительных операций. Регистр CCR содержит два поля признаков: основное {Ev,U,N,Z,V,C} (разряды [5:0]) и дополнительное {Evm,Um,Nm,Zm,Vm,Cm} (разряды [15:10]). Поле признаков в младшем байте регистра CCR является основным, т.к. на его основе формируются условия исполнения команд.

Поля признаков формируются по следующим правилам:

- 1) При исполнении одной операции типа OP1 (AU/LU/FASU) ее признаки помещаются только в основное поле.
- 2) При исполнении одной операции типа OP2 (MS/SH/FMU) ее признаки помещаются в оба поля.
- 3) При одновременном выполнении двух вычислительных операций признаки, формируемые операцией типа OP1 поступают в основное поле, признаки операции типа OP2 - в дополнительное поле.
- 4) В тех случаях, когда операция типа OP1 заполняет только часть признаков в основном поле, оставшиеся формируются операцией OP2.

Регистр CCR содержит также специальные признаки E, t и два управляющих разряда RND и S. Назначение разрядов регистра CCR приведено ниже.

**CCR:**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Evm	Um	Nm	Zm	Vm	Cm	RND	S	t	E	Ev	U	N	Z	V	C

- **C** – признак переноса, сформированного в результате выполнения операции (0 – нет переноса, 1 – есть перенос);
- **V** – признак переполнения результата (0 – нет переполнения, 1 – есть переполнение);
- **Z** – признак нулевого результата (0 – результат не нулевой, 1 – результат нулевой);
- **N** – знак результата (0 – знак положительный, 1 – знак отрицательный);
- **U** – признак ненормализованного результата (0 – нормализованный результат, 1 – ненормализованный результат);
- **Ev** – запомненный ранее возникший признак переполнения результата (0 – не было переполнения, 1 – было переполнение);
- **E** – экспоненциальный признак (формируется командой CMPE);
- **t** – признак истинности условия после исполнения условной команды (t=0 – безусловная команда либо условие ложно; t=1 – условие истинно);
- **S** – бит включения режима насыщения результата (0 – отключение режима насыщения, 1 – включение режима насыщения);
- **RND** – бит управления режимом округления результата (0 – CR (Convergent Rounding), 1 – TCR (Two’s-Complement Rounding));
- **Cm** – признак переноса сформированного в результате выполнения операции OP2 (0 – нет переноса, 1 – есть перенос);
- **Vm** – признак переполнения результата операции OP2 (0 – нет переполнения, 1 – есть переполнение);
- **Zm** – наличие нулевого результата операции OP2 (0 – результат не нулевой, 1 – результат нулевой);
- **Nm** – значение знака результата операции OP2 (0 – знак положительный, 1 – знак отрицательный);
- **Um** – признак ненормализованного результата операции OP2 (0 – нормализованный результат, 1 – ненормализованный результат);
- **Evm** – запомненный ранее возникший признак переполнения результата операции OP2 (0 – не было переполнения, 1 – было переполнение);

Начальное состояние регистра CCR = 0x0000.

### 2.1.2 Регистры AGU, AGU-Y

Генератор адреса AGU содержит восемь наборов по три регистра: регистр адреса An, регистр смещения In, регистр модификатора Mn (n=0-7).

Каждый генератор адреса AGU-Y0, AGU-Y1 содержит набор из четырех регистров: регистра адреса AT, регистров смещения IT и DT, регистра модификатора MT.

### 2.1.3 Регистры PCU

Устройство программного управления PCU включает в себя набор управляющих регистров и стеков:

- Регистр управления и состояния DCSR;
- Программный счетчик PC;
- Регистр состояния SR;
- Регистр-идентификатор IDR;
- Регистр флагов обмена EFR;
- Регистр управления DMA-обменами DMAR;
- Регистр адреса окончания цикла LA;
- Регистр счетчика циклов LC;
- Системный стек SS;
- Стеки циклов CSL, CSH;
- Регистр указателей стека SP;
- Регистры адреса останова SAR, SAR1, SAR2, SAR3, SAR4, SAR5, SAR6, SAR7;
- Счетчик команд CNTR.

#### **Регистр управления и состояния DCSR**

Регистр управления и состояния (DCSR) содержит разряды управления, определяющие состояние и режим работы DSP-ядра, а также прерывания, формируемые DSP-ядром для обработки в CPU-ядре.

Назначение разрядов регистра DCSR указано ниже.

#### **DCSR:**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RST	RUN	-	DBG	-	-	-	-	-	-	-	WT	STP	BRK	SE	PI

- **RST** – программный RESET;
- **RUN** – состояние исполнения программы;

- **DBG** – режим отладки;
- **WT** – флаг состояния ожидания при обращении к XBUF;
- **STP** – прерывание по останову STOP;
- **BRK** – прерывание по останову BREAK;
- **SE** – прерывание по ошибке стека SE;
- **PI** – программное прерывание PI.

Начальное состояние DCSR = 0x0.

### **Программный счетчик PC**

Регистр программного счетчика PC предназначен для хранения 16-разрядного адреса инструкции в программной памяти. Инкрементированное значение PC заносится в системный стек при инициализации нового программного цикла DO, DOFOR и при входе в подпрограмму.

Начальное состояние PC = 0x0.

### **Регистр состояния SR**

Регистр состояния SR содержит параметры управления и состояния DSP-ядра. Разряды [7:0] регистра SR доступны только по чтению, остальные - по записи/чтению.

Назначение разрядов регистра SR указано ниже.

#### **SR:**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SI	SRSI		BC	YM	-	-	SW	t	E	Ev	U	N	Z	V	C

- **SI** – признак режима SIMD;
- **SRSI** – способ формирования интегральных признаков в режиме SIMD;
- **BC** – признак режима “BroadCasting”, т.е. одновременной загрузки памяти данных всех секций DSP-ядра;
- **YM** – режим адресации памяти YRAM;
- **SW** – режим перекрестного обращения к памяти данных в режиме SIMD;
- **C** – перенос;
- **V** – признак переполнения;
- **Z** – признак нулевого результата;
- **N** – признак отрицательного результата;
- **U** – признак ненормализованного результата;
- **Ev** – флаг переполнения (с сохранением);
- **E** – экспоненциальный признак;
- **t** – признак истинности последнего условия.



Разряды [7:0] регистра SR содержат интегральные признаки предыдущей арифметической операции.

Эти интегральные признаки формируются на основе соответствующих кодов, вырабатываемых в вычислительных секциях ALU0, ALU1 и хранящихся в секционных регистрах кодов условий CCR0, CCR1 в зависимости от управляющего кода SRSI (разряды 14-13 регистра SR) согласно приводимой ниже таблице.

SRSI[14:13]	Алгоритм определения CCR
00	Использование CCR0 нулевой секции
01	Объединение секционных CCR0,1 по "И"
10	Объединение секционных CCR0,1 по "ИЛИ"
11	Резерв

Начальное состояние SR = 0x0.

### **Регистр-идентификатор IDR**

Регистр-идентификатор IDR содержит код версии DSP-ядра согласно приводимой ниже таблице. Доступен только по чтению

IDR[15:0]	Модификация DSP-ядра
0x0115	DSP-ядро Elcore-26
Другие коды	Другие модификации DSP-ядра

### **Регистр флагов обмена EFR**

Регистр флагов обмена EFR (32 разряда) доступен только по чтению.

Каждый разряд этого регистра формируется аппаратно и отображает тип последней транзакции, выполненной с соответствующей ячейкой XBUF (0 – чтение из XBUF, 1 – запись).

Начальное состояние EFR=0x0.

### **Регистр управления DMA-обменами DMAR**

Регистр управления DMA-обменами DMAR является 16-разрядным регистром, доступным по записи и чтению. Регистр DMAR предназначен для управления DMA-обменами и имеет следующую структуру:

#### **DMAR:**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DE7	DE6	DE5	DE4	DE3	DE2	DE1	DE0	M7	M6	M5	M4	M3	M2	M1	M0

DE7- запуск DMA (7 канал);	M7- маска запуска DSP со стороны DMA (7 канал);
DE6- запуск DMA (6 канал);	M6- маска запуска DSP со стороны DMA (6 канал);
DE5- запуск DMA (5 канал);	M5- маска запуска DSP со стороны DMA (5 канал);
DE4- запуск DMA (4 канал);	M4- маска запуска DSP со стороны DMA (4 канал);
DE3- запуск DMA (3 канал);	M3- маска запуска DSP со стороны DMA (3 канал);
DE2- запуск DMA (2 канал);	M2- маска запуска DSP со стороны DMA (2 канал);
DE1- запуск DMA (1 канал);	M1- маска запуска DSP со стороны DMA (1 канал);
DE0- запуск DMA (0 канал);	M0- маска запуска DSP со стороны DMA (0 канал);

Начальное состояние DMAR = 0x0000.

### **Регистр адреса окончания цикла LA**

Регистр адреса цикла LA содержит адрес последней инструкции в программном цикле DO, DOFOR. Этот регистр заносится в стек SS по команде DO, DOFOR и извлекается обратно по окончании вложенного цикла либо по команде ENDDO.

Начальное состояние LA = 0x0

### **Регистр счетчика циклов LC**

Регистр счетчика циклов содержит:

- 1) Текущее значение 14-разрядного счетчика программных циклов Nc – разряды 0-13 регистра LC;
- 2) **LF** – Флаг цикла DO – разряд 14 регистра LC ;
- 3) **FV** - Флаг цикла DOFOR – разряд 15 регистра LC.

Формат регистра LC приведен ниже.

**LC:**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FV	LF	Nc													

Значение счетчика программных циклов Nc определяет количество повторений программного цикла DO, в пределах от 1 до  $(2^{14} - 1)$ . Этот регистр заносится в верхнюю (старшую) половину стека циклов CSL по команде DO (образуется вложенный программный цикл) и извлекается обратно по окончании вложенного цикла либо по команде ENDDO.

Начальное состояние LC = 0x0.

### **Стеки SS, CSL, CSH**

Устройство программного управления содержит системный стек SS и стеки циклов CSL, CSH. Системный стек SS имеет объем 15 16-разрядных слов и используется для автоматического сохранения содержимого регистра программного счетчика PC при входе в подпрограмму или в цикл DO, DOFOR. Стеки циклов имеют объем по 7\*16 бит и предназначены для хранения соответственно длины цикла и адреса последней инструкции цикла (LC и LA). Стеки участвуют в обменах как 16-разрядные регистры управления – SS, CSL и CSH.

### Регистр указателей стека SP

Регистр указателей стека SP содержит указатели на последнее записанное в стеки SS, CSH слово.

Назначение разрядов регистра SP указано ниже.

**SP:**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	UFC	CSE	CP[2:0]		-	-	UFS	SSE	SP[3:0]				

- **CP[2:0]** – указатель стека циклов;
- **CSE** – флаг ошибки стека циклов;
- **UFC** – флаг переполнения стека циклов
- **SP[2:0]** – указатель системного стека;
- **SSE** – флаг ошибки системного стека;
- **UFS** – флаг переполнения системного стека.

Младший байт регистра SP содержит указатель и флаги системного стека; старший байт - указатель и флаги стека циклов.

Начальное состояние SP = 0x0.

### Регистры адреса останова SAR, SAR1, SAR2, SAR3, SAR4, SAR5, SAR6, SAR7

Регистры адреса останова SAR, SAR1, SAR2, SAR3, SAR4, SAR5, SAR6, SAR7 являются специализированными 16-разрядными регистрами, используемыми при отладке DSP-ядра. Регистры определяют точку останова (Breakpoint) - адрес инструкции, непосредственно перед исполнением которой должен произойти останов DSP-ядра. Перед исполнением инструкции с указанным адресом DSP-ядро переходит в состояние останова (RUN=0) и флаг прерывания BRK устанавливается в «1».

Начальное состояние регистров SAR, SAR1, SAR2, SAR3, SAR4, SAR5, SAR6, SAR7 = 0xFFFF.

### Счетчик команд CNTR

Счетчик команд CNTR - специализированный 16-разрядный регистр, предназначенный для отладки DSP-ядра. Регистр CNTR задает пошаговый режим исполнения программ в соответствии с приводимой ниже таблицей.

CNTR	Режим исполнения программ
0x0000	Нормальный режим исполнения программ. Число исполняемых команд не ограничено.
N > 0	Пошаговый режим исполнения программ. После исполнения N инструкций DSP-ядро переходит в состояние останова (RUN=0) и флаг прерывания BRK устанавливается в «1».

Начальное состояние CNTR = 0x0.

## 2.2 Организация памяти

Память DSP-ядра Elcore-26 включает в себя два независимых пространства - память программ и память данных.

32-разрядная память данных состоит из двух областей – X-памяти и Y-памяти. Чтение обеих областей памяти данных может происходить одновременно при помощи адресных генераторов - соответственно AGU и AGU-Y. Запись производится при помощи генератора AGU только в одну из областей. Весь объем памяти данных XRAM, YRAM распределяется поровну между секционными модулями памяти.

### 2.2.1 Организация программной памяти

Память программ PRAM имеет 64-разрядную организацию, позволяющую осуществлять хранение и выборку в течение одного такта как 32-разрядных, так и 64-разрядных инструкций. Память PRAM имеет объем 4К 32-разрядных (или 2К 64-разрядных) слов.

Память PRAM адресуется программным адресным генератором, входящим в состав устройства программного управления.

При последовательном ходе программы адрес программной памяти определяется состоянием программного счетчика PC, при программных переходах адрес определяется инструкцией перехода.

### 2.2.2 Организация памяти данных

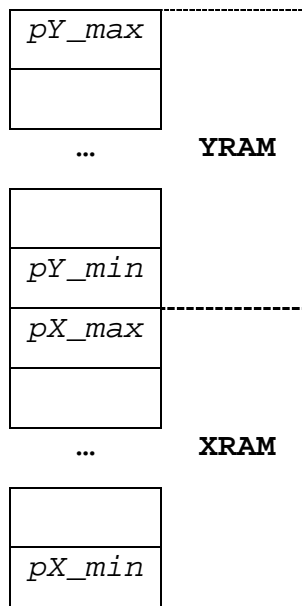
Общее пространство памяти данных DSP-ядра состоит из двух областей: X- и Y-памяти (XRAM, YRAM), имеющих 32-разрядную организацию.

Память XRAM и память YRAM имеют следующий объем:

XRAM – 16К 32-разрядных слов;

YRAM – 16К 32-разрядных слов;

При этом внутренняя адресация памяти XRAM начинается с нулевого адреса, а памяти YRAM - с адреса, следующего за последним адресом XRAM в соответствии с приводимой ниже диаграммой, где  $pX_{min}$ ,  $pX_{max}$  – соответственно минимальный и максимальный адрес X-памяти  $pY_{min}$ ,  $pY_{max}$  – соответственно минимальный и максимальный адрес Y-памяти.



Ниже приводятся граничные адреса X- и Y-памяти (адреса даются в шестнадцатеричной системе счисления).

	<b>pX_min</b>	<b>pX_max</b>	<b>pY_min</b>	<b>pY_max</b>
<b>Elcore-26</b>	0x0000	0x3FFF	0x4000	0x7FFF

### 2.2.3 Организация обменов с памятью данных

Генерация адресов для X- и Y-памяти данных при внутренних обменах DSP-ядра осуществляется 16-разрядными адресными генераторами - AGU и AGU-Y0 (AGU-Y1).

**Адресный генератор AGU является общим для всего DSP-ядра** и производит адресацию всех сегментов X- и Y-памяти данных DSP-ядра как по чтению, так и по записи.

**Адресные генераторы AGU-Y0, AGU-Y1 являются секционными.** В каждой секции DSP-ядра имеется отдельное устройство для генерации адресов Y-памяти - AGU-Y0 и AGU-Y1. Устройства AGU-0, AGU-Y1 адресуют только Y-память и **только по чтению**.

Одновременное обращение к Y-памяти со стороны двух адресных генераторов, AGU и AGU-Y0 (AGU-Y1), является ошибкой. В этих случаях правильно обрабатывается только обращение AGU.

**При чтении Y-памяти при помощи AGU-Y0 (AGU-Y1) полученные данные помещаются всегда в один и тот же predeterminedный регистр R0.L.**

## 2.2.4 Особенности адресации памяти данных в режимах SCALAR и SIMD

Адреса, вырабатываемые генераторами AGU, AGU-Y0 и AGU-Y1, будем обозначать соответственно XAB, YAB0 и YAB1 (так же, как и соответствующие им адресные шины).

В режиме SCALAR указатели памяти, то есть адреса ячеек X- и Y-памяти, к которым происходят обращения, совпадают с вырабатываемыми адресами:

$$pX = XAB, \quad pX_{\min} \leq XAB \leq pY_{\max};$$

$$pY = YAB0, \quad pY_{\min} \leq YAB0 \leq pY_{\max};$$

Примечание. Одновременное обращение к Y-памяти со стороны обоих генераторов, AGU и AGU-Y, запрещено. При таком одновременном обращении к Y-памяти приоритет имеет генератор AGU. Данные, считанные в этом случае генератором AGU-Y, будут неправильными.

В режиме SIMD весь объем памяти данных XRAM, YRAM распределяется поровну между секциями. При этом все ячейки с четными адресами принадлежат к одной секции, все ячейки с нечетными адресами - к другой.

В режиме SIMD указатели памяти для каждой из секций определяются формулами:

$$pX0 = 2 * XAB + (SW), \quad pX_{\min} \leq XAB \leq pY_{\max}/2;$$

$$pX1 = 2 * XAB + (!SW), \quad pX_{\min} \leq XAB \leq pY_{\max}/2;$$

$$pY0 = 2 * YAB0, \quad pY_{\min}/2 \leq YAB0 \leq pY_{\max}/2;$$

$$pY1 = 2 * YAB1 + 1, \quad pY_{\min}/2 \leq YAB1 \leq pY_{\max}/2;$$

Управляющий бит SW (8-й разряд регистра SR) позволяет производить перекрестный обмен между секциями.

Примечание. При  $pX_{\min} \leq XAB \leq pX_{\max}/2$  со стороны генератора AGU происходит обращение к X-памяти, при  $pY_{\min}/2 \leq XAB \leq pY_{\max}/2$  - к Y-памяти.

## 3. ФОРМАТЫ И ТИПЫ ДАННЫХ

### 3.1 Общие положения

DSP-ядро обладает широкими возможностями по работе с различными типами числовых данных: числа с *фиксированной* точкой могут быть представлены с точностью 8/16/32/64 бит; числа с *плавающей* точкой могут быть представлены с точностью 24E8 (соответствует IEEE-754) или 32E16 (числа с плавающей точкой повышенной точности).

В дальнейшем будут использоваться следующие определения форматов и типов данных.

**Формат данных** – характеристика *размера и структуры представления числа* в памяти (регистрах) данных DSP-ядра.

**Тип данных** – правило, устанавливающее *соответствие между математическим значением числа и его представлением в памяти*.

Иными словами, *формат* – это способ размещения, а *тип* – это способ интерпретации данных.

DSP-ядро поддерживает обработку различных типов чисел *с фиксированной точкой* - для целочисленных/дробных, знаковых/беззнаковых, действительных/комплексных чисел.

Для чисел с фиксированной точкой используются следующие форматы данных:

- короткий 16-разрядный формат (*short*);
- длинный 32-разрядный формат (*long*);
- длинный двойной 64-разрядный формат (*\_\_int64*);
- комплексный 16-разрядный формат (*X16*);
- комплексный 8-разрядный формат (*X8*).

Типы данных *с плавающей точкой* предназначены для высокоточной обработки данных, изменяющихся в больших динамических диапазонах. DSP-ядро поддерживает два формата для чисел с плавающей точкой:

- 32-разрядный формат (*24E8*), соответствующий типу **float** (спецификации IEEE754);
- расширенный 48-разрядный формат (*32E16*) для представления чисел с плавающей точкой повышенной точности (тип **double**).

Ниже приводится определение указанных форматов и соответствующих им типов данных, поддерживаемых DSP-ядром.

## 3.2 Форматы данных для чисел с фиксированной точкой

### 3.2.1 Короткий 16-разрядный формат (short)

Короткий (16-разрядный) формат размещается в 16-разрядном регистре RF, либо занимает половину слова памяти данных.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s															

Формат предназначен для хранения следующих типов данных:

- 16-разрядное целое со знаком (**short**) в дополнительном коде;
- 16-разрядное целое без знака (**unsigned short**);
- 16-разрядное дробное (**fractional short**) в дополнительном коде.

Примечание. Здесь и далее

1) Тип **short** определяет 16-разрядные целые числа со знаком в дополнительном коде. Значение числа определяется формулой:

$$s = -s[15]*2^{15} + s[14]*2^{14} + s[13]*2^{13} + \dots + s[1]*2^1 + s[0]*2^0$$

Некоторые возможные значения для чисел данного типа приведены в таблице.

Шестнадцатиричный код	Значение (десятичное)
0x8000	-32768 (минимальное значение)
0xFFFF	-1
0x0000	0
0x0001	1
0x7FFF	32767 (максимальное значение)

Данный тип является основным для 16-разрядного формата, то есть, если не оговаривается иное, операции в формате short используют числа данного типа.

2) Тип **unsigned short** определяет 16-разрядные целые числа без знака. Значение числа определяется формулой:

$$s = s[15]*2^{15} + s[14]*2^{14} + s[13]*2^{13} + \dots + s[1]*2^1 + s[0]*2^0$$

Некоторые возможные значения для чисел данного типа приведены в таблице.

Шестнадцатиричный код	Значение (десятичное)
0xFFFF	65535 (максимальное значение)
0x8000	32768
0x7FFF 0	32767
x0001	1
0x0000	0 (минимальное значение)

Данный тип используется как входной в операции MPUU. Кроме того, 16-разрядные целые числа без знака могут использоваться для подготовки адресных указателей.



3) Тип **fractional short** определяет 16-разрядные дробные числа со знаком в дополнительном коде. Значение числа определяется формулой:

$$s = -s[15]*2^0 + s[14]*2^{-1} + s[13]*2^{-2} + \dots + s[1]*2^{-14} + s[0]*2^{-15}$$

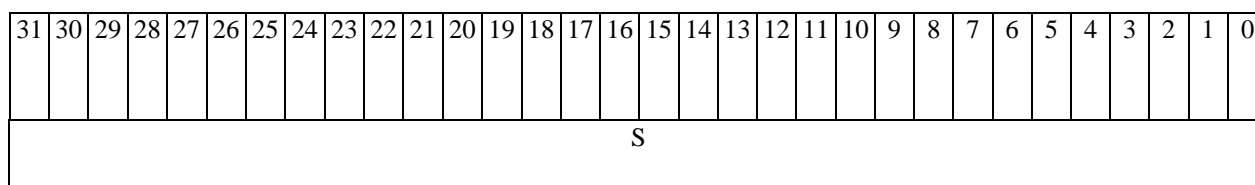
Некоторые возможные значения для чисел данного типа указаны в таблице.

16-ричн.код	Значение (десятичное)
0x8000	-1 (минимальное значение)
0xFFFF	$-2^{-15}$
0x0000	0
0x0001	$2^{-15}$
0x7FFF	$2^{-1} + 2^{-2} + \dots + 2^{-14} + 2^{-15} = 1 - 2^{-15}$ (максимальное значение)

Данный тип используется как входной в операциях MPF, MPF2, MPF2S.

### 3.2.2 Длинный 32-разрядный формат (long)

Длинный 32-разрядный формат размещается в 32-разрядном регистре RF, либо в одном из регистров-аккумуляторов (AC1, AC0), либо занимает одно слово памяти данных.



Формат предназначен для хранения следующих типов данных:

- 32-разрядное целое со знаком (**long**) в дополнительном коде;
- 32-разрядное целое без знака (**unsigned long**);
- 32-разрядное дробное (**fractional long**) в дополнительном коде.

1) Тип **long** определяет 32-разрядные целые числа со знаком в дополнительном коде. Значение числа определяется формулой:

$$s = -s[31]*2^{31} + s[30]*2^{30} + s[29]*2^{29} + \dots + s[1]*2^1 + s[0]*2^0$$

Некоторые возможные значения для чисел данного типа приведены в таблице.

Шестнадцатиричный код	Значение (десятичное)
0x80000000	-2147483648 (минимальное значение)
0xFFFFFFFF	-1
0x00000000	0
0x00000001	1
0x7FFFFFFF	2147483647 (максимальное значение)

Данный тип является основным для 32-разрядного формата, то есть, если не оговаривается иное, операции в формате long используют числа данного типа.

2) Тип **unsigned long** определяет 32-разрядные целые числа без знака. Значение числа определяется формулой:

$$S = S[31]*2^{31} + S[30]*2^{30} + S[29]*2^{29} + \dots + S[1]*2^1 + S[0]*2^0$$

Некоторые возможные значения для чисел данного типа приведены в таблице.

Шестнадцатиричный код	Значение (десятичное)
0xFFFFFFFF	4294967295 (максимальное значение)
0x80000000	2147483648
0x7FFFFFFF 0x00000001	2147483647
0x00000000	1
	0 (минимальное значение)

Данный тип является выходным для операции MPUU.

3) Тип **fractional long** определяет 32-разрядные дробные числа со знаком в дополнительном коде. Значение числа определяется формулой:

$$S = -S[31]*2^0 + S[30]*2^{-1} + S[29]*2^{-2} + \dots + S[1]*2^{-30} + S[0]*2^{-31}$$

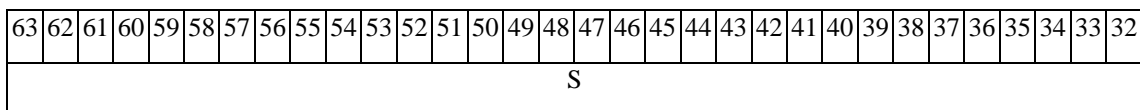
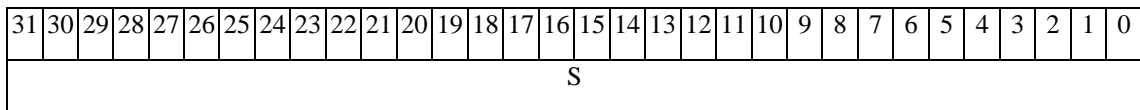
Некоторые возможные значения для чисел данного типа приведены в таблице.

Шестнадцатиричный код	Значение (десятичное)
0x80000000	-1 (минимальное значение)
0xFFFFFFFF	$-2^{-31}$
0x00000000	0
0x00000001	$2^{-31}$
0x7FFFFFFF	$2^{-1} + 2^{-2} + \dots + 2^{-30} + 2^{-31} = 1 - 2^{-31}$ (максимальное значение)

Данный тип является выходным в операциях MPF, MPF2, MPF2S.

### 3.2.3 Длинный двойной 64-разрядный формат (**\_\_int64**)

Длинный 64-разрядный размещается в двух 32-разрядных регистрах RF, либо в 2-х регистрах-аккумуляторах AC1, AC0 (AC1 – старшее слово, AC0 – младшее), либо занимает два слова памяти данных.



Формат предназначен для хранения следующего типа данных:

- 64-разрядное целое со знаком (**\_\_int64**) в дополнительном коде;

1) Тип **\_\_int64** определяет 64-разрядные целые числа со знаком в дополнительном коде. Значение числа определяется формулой:

$$S = -S[63]*2^{63} + S[62]*2^{62} + S[61]*2^{61} + \dots + S[1]*2^1 + S[0]*2^0$$

Некоторые возможные значения для чисел данного типа приведены в таблице.

Шестнадцатиричный код	Значение (десятичное)
0x80000000	-2147483648 (минимальное значение)
0xFFFFFFFF	-1
0x00000000	0
0x00000001	1
0x7FFFFFFF	2147483647 (максимальное значение)

Данный тип является выходным в операциях умножения с накоплением MAC, MACL и умножения MPYL.

Примечание: для выполнения арифметических действий над данными, представленными в разрядной сетке свыше 32 разрядов, к примеру, над 48- или 64-разрядными данными, предусмотрены также операции сложения/вычитания с учетом запоминаемого бита переноса – ADC, ADCL, SBC, SBCL.

### 3.2.4 Комплексный 16-разрядный формат (X16)

Комплексный формат **X16** используется для представления пары 16-разрядных дробных или целых чисел вида (Re,Im). Размещается в 32-разрядном регистре RF, либо занимает одно слово памяти данных.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Re																Im															

Формат предназначен для хранения следующих типов данных:

- пара 16-разрядных целых чисел (Re,Im) со знаком в дополнительном коде (**X16**);
- пара 16-разрядных дробных чисел (Re,Im) со знаком в дополнительном коде (**fractional X16**).

Пара чисел (Re,Im) представляет собой две компоненты – действительную и мнимую – комплексного числа. При этом действительная часть размещается в старшем полуслове, а мнимая часть – в младшем.

1) Тип **X16** определяет пару 16-разрядных целых чисел (Re,Im) со знаком в дополнительном коде. Значение компонент (Re,Im) определяется формулами:

$$Re = -S[31] \cdot 2^{15} + S[30] \cdot 2^{14} + S[29] \cdot 2^{13} + \dots + S[17] \cdot 2^1 + S[16] \cdot 2^0$$

$$Im = -S[15] \cdot 2^{15} + S[14] \cdot 2^{14} + S[13] \cdot 2^{13} + \dots + S[1] \cdot 2^1 + S[0] \cdot 2^0$$

Некоторые возможные значения для чисел данного типа приведены в таблице.

Шестнадцатиричный код	Значение (десятичное)
0x0000FFFF	(0,-1)
0x80000001	(-32768,1)
0x7FFF0000	(32767,0)
0x7FFE8001	(32766,-32767)

2) Тип **fractional X16** определяет пару 16-разрядных дробных чисел (Re,Im) со знаком в дополнительном коде. Значение компонент (Re,Im) определяется формулами:

$$\text{Re} = -S[31]*2^0 + S[30]*2^{-1} + S[29]*2^{-2} + \dots + S[17]*2^{-14} + S[16]*2^{-15}$$

$$\text{Im} = -S[15]*2^0 + S[14]*2^{-1} + S[13]*2^{-2} + \dots + S[1]*2^{-14} + S[0]*2^{-15}$$

Некоторые возможные значения для чисел данного типа указаны в таблице.

Шестнадцатиричный код	Значение
0x0000FFFF	$(0, -2^{-15})$
0x80000001	$(-1, 2^{-15})$
0x7FFF0000	$(1 - 2^{-15}, 0)$
0x7FFE8001	$(1 - 2^{-14}, -1 + 2^{-15})$

Данный тип является выходным в операциях MPX, MACX.

### 3.2.5 Комплексный 8-разрядный формат (X8)

Комплексный 8-разрядный (байтный) формат **X8** используется для представления двух пар 8-разрядных дробных или целых чисел вида (Re1,Im1) и (Re0,Im0). Формат X8 размещается в 32-разрядном регистре RF, либо занимает одно слово памяти данных, и упакован следующим образом:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Re1								Re0								Im1								Im0							

Формат предназначен для хранения следующих типов данных:

- две пары 8-разрядных целых чисел (Re1,Im1) и (Re0,Im0) со знаком в дополнительном коде (**X8**);
- две пары 8-разрядных дробных чисел (Re1,Im1) и (Re0,Im0) со знаком в дополнительном коде (**fractional X8**).

1) Тип **X8** определяет две пары 8-разрядных целых чисел (Re1,Im1) и (Re0,Im0) со знаком в дополнительном коде. Каждая пара составляет комплексное число. Значения компонент чисел определяются формулами:

$$\text{Re1} = -S[31]*2^7 + S[30]*2^6 + S[29]*2^5 + \dots + S[25]*2^1 + S[24]*2^0$$

$$\text{Im1} = -S[15]*2^7 + S[14]*2^6 + S[13]*2^5 + \dots + S[9]*2^1 + S[8]*2^0$$

$$\text{Re0} = -S[23]*2^7 + S[22]*2^6 + S[21]*2^5 + \dots + S[17]*2^1 + S[16]*2^0$$

$$\text{Im0} = -S[7]*2^7 + S[6]*2^6 + S[5]*2^5 + \dots + S[1]*2^1 + S[0]*2^0$$

Некоторые возможные значения для чисел данного типа приведены в таблице.

Шестнадцатиричный код	Значение	
	(Re1,Im1)	(Re0,Im0)
0x0001FFFE	(0,-1)	(1,-2)
0x80000201	(-128,2) (127,-128)	(0,1)
0x7FFF8000	(127,-128)	(-1,0)
0x7FFE8001		(-2,1)

Данный тип используется как входной в операции MACX.

2) Тип **fractional X8** определяет две пары 8-разрядных дробных чисел (Re,Im) со знаком в дополнительном коде. Каждая пара составляет комплексное число. Значения компонент определяются формулами:

$$\text{Re1} = -S[31] \cdot 2^0 + S[30] \cdot 2^{-1} + S[29] \cdot 2^{-2} + \dots + S[25] \cdot 2^{-6} + S[24] \cdot 2^{-7}$$

$$\text{Im1} = -S[15] \cdot 2^0 + S[14] \cdot 2^{-1} + S[13] \cdot 2^{-2} + \dots + S[9] \cdot 2^{-6} + S[8] \cdot 2^{-7}$$

$$\text{Re0} = -S[23] \cdot 2^0 + S[22] \cdot 2^{-1} + S[21] \cdot 2^{-2} + \dots + S[17] \cdot 2^{-6} + S[16] \cdot 2^{-7}$$

$$\text{Im0} = -S[7] \cdot 2^0 + S[6] \cdot 2^{-1} + S[5] \cdot 2^{-2} + \dots + S[1] \cdot 2^{-6} + S[0] \cdot 2^{-7}$$

Некоторые возможные значения для чисел данного типа указаны в таблице.

16-ричн.код	Значение	
	(Re1,Im1)	(Re0,Im0)
0x0001FFFE	(0,2 <sup>-7</sup> )	(2 <sup>-7</sup> ,-2 <sup>-6</sup> )
0x80000201	(-1,2 <sup>-6</sup> )	(0,2 <sup>-7</sup> )
0x7FFF8000	(1-2 <sup>-7</sup> ,-1)	1-2 <sup>-6</sup> ,0)
0x7FFE8001	(1-2 <sup>-7</sup> ,-1)	(1-2 <sup>-6</sup> ,2 <sup>-7</sup> )

Данный тип используется как входной в операциях MPX, MACX.

### 3.2.6 Граничные значения для типов данных с фиксированной точкой

Во всех знаковых типах с фиксированной точкой данные представлены в *дополнительном коде*. Ниже приводятся граничные значения для указанных типов чисел. В случае комплексных чисел приводимые граничные значения относятся к каждой из компонент.

Граничные значения		Форматы			
		8 разрядов	16 разрядов	32 разряда	64 разряда
Наименьшее значение	16-ричное представление	0x80	0x8000	0x80000000	0x8000000000000000
	дробное	-1.0	-1.0	-1.0	-1.0
	целое	-128	-32768	-2147483648	-9223372036854775808
Наибольшее значение	16-ричное представление	0x7F	0x7FFF	0x7FFFFFFF	0x7FFFFFFFFFFFFFFF
	дробное	$1 - 2^{-7}$	$1 - 2^{-15}$	$1 - 2^{-31}$	$1 - 2^{-63}$
	целое	127	32767	2147483647	9223372036854775807

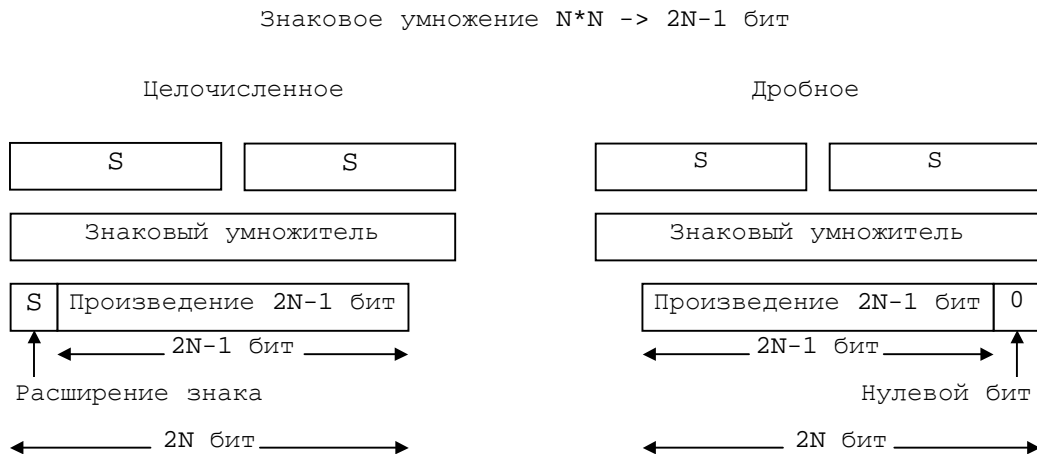
Граничные значения используются при выполнении арифметических операций в режиме насыщения (Saturation). Включение данного режима подразумевает присвоение результату операции граничного значения в случае выхода результата за пределы разрешенного диапазона.

Ниже приводится перечень операций, в которых может быть использован режим насыщения.

Long	Short	Complex
<u>Блок MS</u>		
	MPF	
	MPF2	
	MPF2S	
		MPX
ASLL	ASL	ASLX
<u>Блок AU</u>		
ABSL	ABS	
NEGL	NEG	
ADDL	ADD	ADDX
SUBL	SUB	SUBX
ADCL	ADC	
ADC16L	AD1	
SBCL	SBC	
ADDSUBL	ADDSUB	ADDSUBX
RNDL		
ADDLR	ASH	
SUBLR	SAH	
ADDLRTR		
SUBLRTR		
FTRL		

### 3.2.7 Особенности выполнения арифметических операций для целых и дробных чисел

Представление результата операции над целыми (integer) числами или дробями (fractional) – одинаково для операций сложения или вычитания, но различается при умножении. Различие в формировании умножения двух чисел в целом и дробном формате проиллюстрировано на рисунке 3.1.



**Рисунок 3.1** Различие целочисленных/дробных умножений

Ключевое различие между целочисленным и дробным умножением находится в представлении  $(2N - 1)$  - разрядного результата.

*В дробном умножении*  $(2N - 1)$  значащие биты результата должны быть выровнены по левой границе, а младший бит должен быть заполнен нулем, чтобы обеспечить дробное представление.

*В целочисленном умножении*  $(2N - 1)$  значащие биты результата должны быть выровнены по правому краю и дублируется знаковый разряд.



### 3.3 Форматы данных для чисел с плавающей точкой

#### 3.3.1 Формат 24E8 (min float)

32-разрядный формат (24E8) предназначен для хранения чисел с плавающей точкой типа **float**, соответствующих спецификации IEEE-754 с некоторыми ограничениями, указанными ниже. Формат размещается в 32-разрядном регистре или занимает одно слово памяти данных.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s	e								f																						

Формат имеет структуру, состоящую из трех полей:

$s$  =  $S[31]$  – знак числа;

$e[7:0]$  =  $S[30:23]$  – экспонента (порядок) числа со смещением +127;

$f[22:0]$  =  $S[22:0]$  – дробная часть мантииссы числа.

#### Значение числа

Значение числа X типа float вычисляется по формуле:

$$X = (-1)^s * 2^E * F,$$

где **E** – экспонента без смещения:

$$E = e[7] * 2^7 + e[6] * 2^6 + e[5] * 2^5 + e[4] * 2^4 + e[3] * 2^3 + e[2] * 2^2 + e[1] * 2^1 + e[0] * 2^0 - 127;$$

**F** – полная мантиисса:

$$F = 1 + f[22] * 2^{-1} + f[21] * 2^{-2} + \dots + f[1] * 2^{-22} + f[0] * 2^{-23};$$

$F \geq 1.0$ , т.е. мантиисса может быть только нормализованной и только положительной. Бит, соответствующий 1.0, «умалчивается».

Некоторые возможные значения для чисел данного типа приведены в таблице.

	e	f	Значение числа
/1	$1 \leq e \leq 254$	x	числа с плавающей точкой
	255	0	$+\infty$
	255	0	$-\infty$
	255	xx..x, f≠0	NaN: не - число общего вида, только входное
	255	11..1	QNaN: выходное не – число
			Примеры чисел:
	254	11..1	Минимальное число $-2^{127}(1.0+1/2^1+..+1/2^{23})$
	127	10..0	-1.5
	127	00..0	-1.0
	1	00..0	Максимальное отрицательное число $-2^{-126}$
	0	0	Нуль, входной или выходной
/1	0	0	Нуль, только входной
	1	00..0	Минимальное положительное число $2^{-126}$
	127	00..0	1.0
	127	10..0	1.5
	254	11..1	Максимальное число $2^{127}(1.0+1/2^1+..+1/2^{23})$

**Случаи формирования результата QNaN:**

- 1) Входной операнд NaN.
- 2) Входной операнд QNaN.
- 3)  $(\pm \infty)*0$ .
- 4)  $(+\infty) + (-\infty)$ ,  $(-\infty) + (+\infty)$ ,  $(+\infty) - (+\infty)$ ,  $(-\infty) - (-\infty)$ .

### Способы округления

Во всех специально не оговоренных случаях используется один вариант округления: к ближайшему числу (при равноудаленности – к четному, с нулевым младшим битом мантиссы).

Ф до округления	Ф после округления	Пояснение
100x 00	100x	Округление не требуется
100x 01	100x	Округление к меньшему
1000 10	1000	Равноудаленность, округление к четному
1001 10	1010	Равноудаленность, округление к четному
1000 11	1001	Округление к большему
1001 11	1010	Округление к большему

### Ограничения

Тип **float** соответствует спецификации IEEE-754 со следующими ограничениями:

- Денормализованные числа не обрабатываются, на выходе не бывает денормализованного результата, он заменяется нулем.
- Не используется в качестве результата знаковый ноль.
- Не используется подраздел из спецификации IEEE-754 работы с не числами, а лишь фиксируется не число в виде результата и оно всегда имеет вид QNaN и не имеет знака.
- Используется одна мода округления в отличие от четырех, предложенных в стандарте.
- Результат команды FINR (обратной величины квадратного корня) из нуля со знаком минус не равен минус бесконечности, как должно быть по спецификации IEEE-754, а равен QNaN.
- Не реализованы предусмотренные стандартом исключения “UNDERFLOW”, “INEXACT”.

### 3.3.2 Формат 32E16 (mun double)

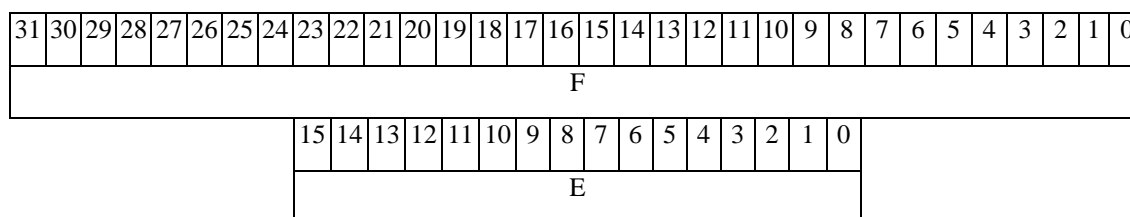
Расширенный 48-разрядный формат (**32E16**) предназначен представления чисел с плавающей точкой повышенной точности (тип **double**).

Формат имеет структуру, состоящую из двух полей:

E[15:0] – экспонента (порядок) числа – 16-разрядное целое со знаком в дополнительном коде;

F[31:0] – мантисса числа – 32-разрядное дробное число в дополнительном коде.

Мантисса F размещается в 32-разрядном регистре RF или одной ячейке памяти; экспонента - в 16-разрядном регистре RF или младшем полуслове ячейки памяти.



### Значение числа

Значение числа  $X$  типа **double**, за исключением нескольких специальных случаев, рассмотренных ниже, вычисляется по формуле:

$$X = 2^E * F,$$

где

$$E = - E[15] * 2^{15} + E[14] * 2^{14} + E[13] * 2^{13} + \dots + E[1] * 2^1 + E[0] * 2^0;$$

$$F = - F[31] * 2^0 + F[30] * 2^{-1} + F[29] * 2^{-2} + \dots + F[1] * 2^{-30} + F[0] * 2^{-31};$$

мантисса может быть только нормализованной.

В таблице указаны специальные случаи для чисел типа **double**.

E[15:0]	F[31:0]	Назначение
-32768=0x8000	0x00000000 (допустимо х..х)	число 0
32767=0x7FFF	0x00000000	$+\infty$
32767=0x7FFF	0x80000000	$-\infty$
32767=0x7FFF	0x7FFFFFFF	NaN
$0x8001 \leq E \leq 0x7FFE$	$0x40000000 \leq E \leq 0x7FFFFFFF$ $0x80000000 \leq E \leq 0xFFFFFFFF$	число >0 число <0

Некоторые примеры записи чисел данного типа приведены ниже.

E	F	Значение
0x8000	0x00000000	0
0x0000	0x40000000	+0.5
0xFFFF	0x80000000	-0.5
0x0001	0x40000000	+1.
0x0000	0x80000000	-1.
0x0004	0x54000000	+10.5
0x0004	0xAC000000	-10.5
0x0007	0x7F800000	+127.5
0x0008	0x40000000	+128.
0x0008	0x40400000	+128.5
0x0007	0x80800000	-127.5
0x0007	0x80000000	-128.
0x0008	0xBFC00000	-128.5
0x0002	0x6487ED50	$\pi$
0x0002	0x56FC2A2C	e

Для преобразования чисел различных типов предусмотрены операции CVFEF, CVFE, CVFI, CVIF.

### 3.4 Запись различных типов констант в операндах и памяти

Средства программирования и отладки DSP-ядра Elcore-26 обеспечивают поддержку записи различных типов констант в память и использования их в качестве непосредственных операндов. Ниже в таблице приведены синтаксические правила и примеры задания различных типов констант в ассемблере DSP-ядра Elcore-26. Более подробную информацию об этом можно получить в описаниях средств программирования DSP-ядра Elcore-26, входящих в состав среды MCStudio.

Тип данных	Непосредственный операнд	Константа в памяти
Целый 16-разрядный (short)	#s <i>Пример: -32767</i>	.dw #s <i>Пример: .dw -32767</i>
Целый 32-разрядный (long)	#S <i>Пример: -32767*32767</i>	.dl #S <i>Пример: .dl 0x80000000</i>
Целый комплексный (X16) (16+16)	#[Re,Im] <i>Пример: [-32767,0x8000]</i>	.dl #[Re,Im] <i>Пример: .dl [-32767,0x8000]</i>
Целый комплексный байтный (X8) (8+8+8+8)	#[@Re1,Re0],[@Im1,Im0] <i>Пример: [[-1,0],[@17,0x80]]</i>	.dl #[@Re1,Re0],[@Im1,Im0] <i>Пример: .dl [[@1,0],[@17,0x80]]</i>
Дробный 16-разрядный (fractional short)	#s <i>Пример: -0.875</i>	.fr #s <i>Пример: .fr 0.99999</i>
Дробный 32-разрядный (fractional long)	#S <i>Пример: 0.875</i>	.frl #S <i>Пример: .frl -0.999999999</i>
Дробный комплексный (fractional X16)	#[Re,Im] <i>Пример: [0.875,-0.375]</i>	.frl #[Re,Im] <i>Пример: .frl [-0.375,0.875]</i>
Дробный комплексный байтный (fractional X8) (8+8+8+8)	#[@Re1,Re0],[@Im1,Im0] <i>Пример: [[@0.5,-0.5],[@0.25,-0.5]]</i>	.frl #[@Re1,Re0],[@Im1,Im0] <i>Пример: .frl [[@-0.5,0.2],[@0.17,0.8]]</i>
Плавающая точка 24E8 (float)	#S <i>Пример: #-2.75</i>	.real #S <i>Пример: .real -3.7e6</i>
Плавающая точка 32E16 (double)	-	.double #S <i>Пример: .double -31.25e-1</i>

## 4. ПРИЗНАКИ РЕЗУЛЬТАТА ОПЕРАЦИИ

Ниже приводятся стандартные правила формирования признаков результата вычислительной операции: **U**(unnormalized), **N**(negative), **Z**(zero), **V**(overflow), **C**(carry). Для отдельных операций некоторые признаки могут формироваться по иным специально оговоренным правилам. В дальнейшем при описании правил формирования признаков используются следующие обозначения: *msb* – номер старшего (знакового) разряда результата *D*, т.е. *msb*=31 для 32-разрядных чисел и *msb*=15 для 16-разрядных.

Кроме указанных выше основных признаков, при выполнении операций могут формироваться и некоторые дополнительные признаки, определение которых дано ранее при описании регистра ССР.

Признак	Стандартные правила формирования признаков	
	Все вычислительные операции (кроме сдвига)	Операции сдвига: ASL, ASLL, ASLX, ASR, ASRL, ASRX, ASRLE, LSL, LSL, LSLX, LSR, LSRL, LSRX, ROL, ROLL, ROR, RORL
<b>U</b>	U = 0, если D[msb]≠D[msb-1]; U = 1, если D[msb]=D[msb-1];	
<b>N</b>	N=D[msb];	
<b>Z</b>	Z = 1, если D=0; Z = 0, если D≠0;	
<b>V</b>	V = 1, если D[msb+1]≠D[msb]; V = 0, если D[msb+1]=D[msb];	Операции ASL, ASLL, ASLX: V = 0, если хотя бы один разряд, выдвигаемый за пределы разрядной сетки или на место знака, не равен знаку; V = 1, иначе;
<b>C</b>	C = Cout[msb], если режим Scaling выключен; C = Cout[msb+1], если режим Scaling включен.	C принимает значения последнего из битов, выдвинутых за разрядную сетку результата D[msb:0] вправо или влево, в зависимости от направления сдвига. При нулевом сдвиге C = 0.

### Пояснения.

Арифметическое устройство выполнено как полный 33-разрядный сумматор/вычитатель с дополнительным старшим разрядом под номером *msb*+1, используемым только для формирования признаков. На выход поступают 32 младших разряда результата *D*[*msb*:0]. Каждый из 33-х каскадов сумматора формирует как соответствующий бит результата *D*[*i*], так и перенос в следующий разряд *Cout*[*i*].

## 5. ОБЩАЯ ХАРАКТЕРИСТИКА СИСТЕМЫ ИНСТРУКЦИЙ

Система инструкций DSP-ядра ориентирована на высокопроизводительную параллельную обработку данных. В рамках одной инструкции может выполняться несколько команд (операций). Таким образом, здесь и далее в рамках настоящего документа приняты следующие терминологические определения:

**Инструкция** – набор команд (операций), выполняющихся одновременно.

**Команда (операция)** – часть инструкции, определяющая действие того или иного исполнительного устройства DSP-ядра.

Инструкции размещаются в программной памяти DSP-ядра последовательно в порядке нарастания адреса. Каждая инструкция занимает одно (короткие форматы) или два (длинные форматы) 32-разрядных слова. Адрес чтения программной памяти формируется с помощью регистра программного счетчика PC, автоматически инкрементирующегося (на 1 или 2) при последовательном ходе программы.

Команды (операции) DSP-ядра по своему действию делятся на три больших группы:

### 1) Вычислительные команды

Каждая из команд данного типа производит некоторое действие над данными, хранящимися в регистровом файле (RF) DSP-ядра, и полученные результаты помещаются также в регистры RF. Кроме того, формируется набор признаков результата, который помещается в регистр CCR.

### 2) Команды пересылок

При помощи команд пересылок производится обмен данными между регистрами, регистрами и памятью данных, либо загрузка непосредственных данных в регистры.

### 3) Команды программного управления

Команды программного управления производят изменения в последовательности исполнения инструкций DSP-ядра. С их помощью организуются программные переходы, циклы и др.

Большинство команд, включая все вычислительные команды и команды пересылок, исполняется за один командный цикл. Исключение составляют команды программных переходов B, BD, BS, J, JD, JS, RTS выполняющиеся за два командных цикла и команда ENDDO, выполняющаяся за три командных цикла.

## 5.1 Вычислительные команды

Вычислительные команды, в свою очередь, делятся по характеру исполняемой операции и по форматам обрабатываемых данных на более мелкие группы команд, приведенные ниже в таблицах.

При определении мнемоники команд приняты следующие соглашения:

- команды, работающие в длинном формате, имеют на конце суффикс “L”;
- команды, работающие в комплексных форматах, имеют на конце суффикс “X”;
- команды, работающие в формате с плавающей точкой, имеют префикс “F”.

Имеются некоторые исключения из приведенных выше правил, в частности, для команд, работающих одновременно с различными форматами данных. Детальное описание команд дается в следующем разделе.



### 5.1.1 Команды сложения/вычитания в форматах с фиксированной точкой

Мнемоника	Содержание команды (формат данных)
	<b>Сложение</b>
ADC	Сложение с переносом (short)
ADCL	Сложение с переносом (long)
ADC16L	Сложение смешанное
ADD	Сложение (short)
ADDL	Сложение (long)
ADDLR	Сложение (long) с округлением
ADDLRTR	Сложение (long) с округлением и преобразованием формата (в short)
ADDX	Сложение комплексное (X16)
ADI	Сложение и инкремент (short)
	<b>Вычитание</b>
SBC	Вычитание с переносом (short)
SBCL	Вычитание с переносом (long)
SUB	Вычитание (short)
SUBL	Вычитание (long)
SUBLR	Вычитание (long) с округлением
SUBLRTR	Вычитание (long) с округлением и преобразованием формата (в short)
SUBX	Вычитание комплексное (X16)
	<b>Сложение-вычитание</b>
ADDSUB	Сложение-вычитание (short)
ADDSUBL	Сложение-вычитание (long)
ADDSUBX	Сложение-вычитание (X16)
ASH	Сложение и вычитание двух пар чисел (short)
SAH	Сложение и вычитание двух пар чисел (short)
	<b>Инкремент/декремент</b>
INC	Инкремент (short)
INCL	Инкремент (long)
DEC	Декремент (short)
DECL	Декремент (long)

### 5.1.2 Команды умножения/накопления в форматах с фиксированной точкой

Мнемоника	Содержание команды (формат данных)
<b>Умножение</b>	
MPF	Умножение дробное со знаком (short)
MPF2	Парное умножение дробное со знаком (short)
MPF2S	Парное умножение дробное со знаком (short), с перестановкой сомножителей
MPSS	Умножение целое со знаком (short)
MPUU	Умножение целое без знака (short)
MPX	Умножение дробное комплексное (X8), второй операнд - комплексно-сопряженный
MPYL	Умножение целое со знаком (long)
<b>Умножение с накоплением (MAC)</b>	
MAC	Умножение целое со знаком (short) и накопление (в формате __Int64)
MACL	Умножение целое со знаком (long) и накопление (в формате __Int64)
MACX	Умножение дробное комплексно-сопряженное (X8) и целочисленное (X16)
MAC2	Парное умножение (short) и накопление 2-х результатов (в формате long)
SAC2	Парное накопление (в формате long) со знаком

### 5.1.3 Команды сдвига в форматах с фиксированной точкой

Мнемоника	Содержание команды (формат данных)
<b>Арифметический сдвиг</b>	
ASL	Арифметический сдвиг влево (short)
ASLL	Арифметический сдвиг влево (long)
ASLX	Арифметический сдвиг влево (X16)
ASR	Арифметический сдвиг вправо (short)
ASRL	Арифметический сдвиг вправо (long)
ASRX	Сдвиг арифметический вправо (X16)
<b>Логический сдвиг</b>	
LSL	Логический сдвиг влево (short)
LSLL	Логический сдвиг влево (long)
LSLX	Логический сдвиг влево (X16)
LSR	Логический сдвиг вправо (short)
LSRL	Логический сдвиг вправо (long)
LSRX	Логический сдвиг вправо (X16)
<b>Циклический сдвиг на один разряд</b>	
ROL	Циклический сдвиг на один разряд влево (short)
ROLL	Циклический сдвиг на один разряд влево (long)
ROR	Циклический сдвиг на один разряд вправо (short)
RORL	Циклический сдвиг на один разряд вправо (long)

### 5.1.4 Другие арифметические команды в форматах с фиксированной точкой

Мнемоника	Содержание команды (формат данных)
	<b>Абсолютное значение</b>
ABS	Абсолютное значение (short)
ABSL	Абсолютное значение (long)
	<b>Обнуление регистра</b>
CLR	Обнуление (очистка) регистра (short)
CLRL	Обнуление (очистка) регистра (long)
	<b>Изменение знака</b>
NEG	Изменение знака (short)
NEGL	Изменение знака (long)
	<b>Транзит</b>
TR	Транзит (short)
TRL	Транзит (long)
	<b>Сравнение</b>
CMP	Сравнение (short)
CMPL	Сравнение (long)
CMPM	Сравнение модулей (short)
CMPML	Сравнение модулей (long)
CS2	Парная операция выбора большего из двух чисел (short) с фиксацией бита выбора
	<b>Максимум/минимум</b>
MAX	Выбор большего числа (short)
MAXL	Выбор большего числа (long)
MAXM	Выбор числа с большим модулем (short)
MAXML	Выбор числа с большим модулем (long)
MIN	Выбор меньшего числа (short)
MINL	Выбор меньшего числа (long)
MINM	Выбор числа с меньшим модулем (short)
MINML	Выбор числа с меньшим модулем (long)
	<b>Определение признаков операнда</b>
TST	Определение признаков операнда (short)
TSTL	Определение признаков операнда (long)
TSTX	Определение признаков операнда (X16)

### 5.1.5 Округление, преобразования форматов, упаковка/распаковка

Мнемоника	Содержание команды (формат данных)
	<b>Округление</b>
RNDL	Округление
	<b>Преобразование формата</b>
FTR	Преобразование формата
FTRFL	Преобразование формата
FTRL	Преобразование формата
	<b>Упаковка/распаковка</b>
PACK	Упаковка (short)
PACKL	Упаковка (long)
DISPFX	Распаковка (дробная) X8 в X16
DISPX	Распаковка (целочисленная) X8 в X16

### 5.1.6 Логические команды, операции с битами и битовыми полями

Мнемоника	Содержание команды (формат данных)
	<b>Логические команды</b>
AND	Логическое И (short)
ANDC	Логическое И с инверсией (short)
ANDCL	Логическое И с инверсией (long)
ANDI	Инверсия логического И (short)
ANDL	Логическое И (long)
EOR	Логическое исключаящее ИЛИ (short)
EORL	Логическое исключаящее ИЛИ (long)
NOT	Логическое отрицание (short)
NOTL	Логическое отрицание (long)
OR	Логическое ИЛИ (short)
ORC	Логическое ИЛИ с инверсией (short)
ORCL	Логическое ИЛИ с инверсией (long)
ORI	Инверсия логического ИЛИ (short)
ORL	Логическое ИЛИ (long)
	<b>Определение параметра денормализации</b>
PDN	Определение параметра денормализации (short)
PDNL	Определение параметра денормализации (long)
PDNX	Определение параметра денормализации (X16)
	<b>Операции с битами и битовыми полями</b>
BTST	Проверка разряда (short)
BTSTL	Проверка разряда (long)
MSKG	Формирование маски (short)
MSKGL	Формирование маски (long)
INSL	Побитное мультиплексирование (long)
SWL	Перестановка (long)
	<b>Сложение бит</b>
SMB	Сложение бит (short)
SMBL	Сложение бит (long)

### 5.1.7 Команды для обработки данных в формате с плавающей точкой 24E8

Мнемоника	Содержание команды (формат данных)
FADD	Сложение (24E8)
FSUB	Вычитание (24E8)
FAS	Сложение-вычитание (24E8)
FINT	Округление к ближайшему целому (24E8)
FLOOR	Округление к ближайшему целому (24E8)
FMPY	Умножение (24E8)
FTST	Определение признаков операнда (24E8)
FIN	1-я итерация обратной величины
FINR	1-я итерация обратной величины квадратного корня
CVFI	Преобразование формата: формат 24E8 в 32-разрядное целое в дополнительном коде
CVIF	Преобразование формата: 32-разрядное целое в дополнительном коде со знаком в формат 24E8

### 5.1.8 Команды для обработки данных в формате с плавающей точкой 32E16

Мнемоника	Содержание команды (формат данных)
CMPE	Сравнение экспонент
ASRLE	Условный арифметический сдвиг вправо (long)
PDNE	Измерение параметра денормализации 16-разрядной мантииссы
PDNLE	Измерение параметра денормализации 32-разрядной мантииссы
CVEF	Преобразование формата: формат 32E16 в 24E8
CVFE	Преобразование формата: формат 24E8 в 32E16

## 5.2 Команды пересылок

При помощи команд пересылок производится обмен данными между регистрами, регистрами и памятью данных, либо загрузка непосредственных данных в регистры.

Для всех видов команд пересылок используется одна и та же мнемоническая запись – MOVE, однако форматы и коды инструкций зависят от типа пересылки и ее параметров. Подробно форматы и коды инструкций рассматриваются в следующем разделе.

Мнемоника	Содержание команды (формат данных)
MOVE	Пересылка данных

## 5.3 Команды программного управления

Команды программного управления производят изменения в последовательности исполнения инструкций DSP-ядра. С их помощью организуются программные переходы, циклы, вход в подпрограмму и выход из нее, останов DSP-ядра.

Команды программных переходов B, BD, BS, J, JD, JS являются условными, остальные команды - безусловные.

Мнемоника	Содержание команды (формат данных)
DO	Оператор цикла
DOFOR	Оператор бесконечного цикла
ENDDO	Окончание цикла
B	Ветвление программы
BD	Ветвление программы (отложенное)
BS	Вызов подпрограммы
J	Программный переход
JD	Программный переход (отложенный)
JS	Вызов подпрограммы
RTS	Возврат из подпрограммы
NOP	Пустая операция
STOP	Останов

## 6. ФОРМАТЫ ИНСТРУКЦИЙ

В рамках одной инструкции в каждой из SIMD-секций может быть выполнено до двух вычислительных операций и до двух операций пересылок.

Количество одновременно исполняемых операций и их тип определяют *формат инструкции* – т.е. размер и структуру кода.

Участки кода инструкции, определяющие те или иные параметры входящих в нее операций, называются *полями* инструкции.

В настоящем разделе рассматриваются форматы и назначение полей инструкций DSP-ядра Elcore-26.

В приводимой ниже таблице дан перечень форматов инструкций.

Формат	Условие	Операция 1	Операция 2	Пересылка 1	Пересылка 2	Длина кода, 32-р. слов
1	[cc]	OP #5/S1,S2,D				1
2	[cc]	OP #16/32,S2,D				2
2d			DO #16, #16			2
2t	[cc]			R/R.L/RC $\beta$ $\rightarrow$ R/R.L/RC		1
3		OP #16,d				1
3m	[cc]		B/J #16			1
3mb	[cc]		B/J Ai			1
4		OP #5/S,D		XRAM $\beta$ $\rightarrow$ R.L		1
4t				XRAM $\beta$ $\rightarrow$ BUF		1
5		OP #5/S,D		R/R.L $\beta$ $\rightarrow$ R/R.L		1
6		OP #5/S,D		R $\beta$ $\rightarrow$ RC		1
6t	[cc]			XRAM $\beta$ $\rightarrow$ R.L		1
7	[cc]	OP S,D		#16/32 $\rightarrow$ RC/R/R.L		2
7t	[cc]			XRAM(Ai+#16) $\beta$ $\rightarrow$ R.L		2
8a		OP2 #5/S1,S2,D	OP1[s] S1,S2,D	XRAM $\beta$ $\rightarrow$ R.L	YRAM $\rightarrow$ R0	2
8b		OP2 #5/S1,S2,D	OP1[s] S1,S2,D	R/R.L $\beta$ $\rightarrow$ R/R.L	YRAM $\rightarrow$ R0	2
8c	[cc]	OP2 #5/S1,S2,D	OP1[s] S1,S2,D	R.L $\beta$ $\rightarrow$ R.L		2
8d		OP2 #5/S1,S2,D	OP1[s] S1,S2,D	R $\beta$ $\rightarrow$ RC		2

Обозначения:

$S, S1, S2, D$	– 16- или 32-разрядный регистр данных;
$s, d, R$	– 16-разрядный регистр данных;
$R.L$	– 32-разрядный регистр данных;
$\#x$	– $x$ -разрядные непосредственные данные;
$RC$	– управляющий регистр;
$cc$	– код условия;
$Ai$	– адресный регистр, $i=0,1,\dots,7$ .

## 6.1 Описание форматов инструкций

Форматы инструкций, таким образом, отличаются друг от друга по следующим признакам:

1) По возможности использования условия при выполнении инструкции.

Все инструкции DSP Elcore-26 делятся на условно исполняемые (при истинности специфицированного условия) и безусловные. Заметим, что специфицированное условие имеет действие на все операции, входящие в инструкцию.

2) По количеству одновременно выполняемых вычислительных операций. Одновременно в рамках одной инструкции может быть выполнено не более 2-х вычислительных операций. Некоторые вычислительные операции (например, ADDSUB) могут включать в себя несколько арифметических действий. Тем не менее при рассмотрении системы инструкций каждая из них считается одной операцией.

3) По типу и количеству операндов, используемых вычислительной операцией.

В качестве операндов-источников могут использоваться регистры данных ( $S, S1, S2$ ) либо непосредственные данные  $\#5, \#16$  или  $\#32$ . Операндами-получателями (приемниками) результата являются регистры данных  $D$  (для некоторых команд получателями результата являются также регистры  $CCR, PDNR, AC0, AC1$ ).

Формат инструкции зависит от числа адресуемых операндов в вычислительных операциях. В форматах 1, 2, 8 это число – не более трех ( $S1, S2, D$ ). Такие операции называются 3-адресными. В форматах 4 – 7 разрешены 2-адресные вычислительные операции, в которых число адресуемых операндов не более двух ( $S, D$ ). Поле каждого адресуемого операнда состоит из 5 бит, при помощи которых адресуется один из регистров  $RF$ . Для операций сдвига в поле операнда  $S1 (S)$  может помещаться непосредственное значение сдвига  $\#5$ .



4) По количеству и типу одновременно выполняемых операций пересылок.

Максимальное количество одновременно выполняемых операций пересылок – две (в форматах 8a, 8b). Источником в операции пересылки могут быть регистры данных (R - 16-разрядные или R.L - 32-разрядные), регистры управления RC, память XRAM или YRAM, а данные #5, #16 или #32. Получателем могут являться регистры данных, регистры управления RC, память XRAM или YRAM.

5) Некоторые операции программного управления и пересылок требуют для своей кодировки специальных форматов – 2d, 3m, 3mb, 2t, 6t, 7t.

### **Формат 1**

OP[.cc] #5/S1,S2,D

Содержит одну вычислительную операцию (3-, 2- или 1-адресную) с условным исполнением, длина кода - одно слово (32 бита).

В рамках данного формата может быть выполнена любая вычислительная операция. Операнды-источники (S1, S2) и приемник (D) – регистры регистрового файла (16- или 32-разрядные). Вместо 1-го источника может использоваться непосредственный 5-разрядный операнд #5 (параметр сдвига, номер бита).

### **Формат 2**

OP[.cc] #16/#32,S2, D

Содержит одну 3-адресную вычислительную операцию с условным исполнением, длина кода - два слова (64 бита).

Первый операнд-источник – непосредственное 16- или 32-разрядное значение, содержащееся во втором слове инструкции. Второй операнд-источник (S2) и приемник (D) – регистры регистрового файла (16- или 32-разрядные).

### **Формат 2d**

DO #16, #16

Формат 2d применяется для кодирования команды DO с непосредственно заданным числом повторения циклов, содержащимся во втором слове инструкции. Длина кода – 64 разряда.

### **Формат 2t**

MOVE.cc	R, R
	R.L, R.L
	R, RC
	RC, R

Формат 2t является производным от формата б и применяется для кодирования условной пересылки между регистром данных и регистром управления.

Длина кода - 32 разряда.

### **Формат 3**

ОР #16, d

Содержит одну 2-адресную вычислительную операцию с безусловным исполнением. Длина кода - одно слово.

Обрабатываются только 16-разрядные операнды.

1-й операнд (источник) #16 – непосредственное значение.

2-й операнд – источник и приемник (d) – 16-разрядный регистр регистрового файла.

### **Формат 3m**

B	[.cc] #16
BD	
BS	
J	
JD	
JS	

Формат 3m применяется для кодирования команд программных переходов B, BD, BS, J, JD, JS с непосредственно заданным адресом перехода (или с переходом по метке). Длина кода – 32 разряда.

### **Формат 3mb**

B	[.cc] Ai
BD	
BS	
J	
JD	
JS	

Формат 3mb применяется для кодирования команд программных переходов B, BD, BS, J, JD, JS с переходом по адресу, содержащемуся в адресном регистре Ai. Длина кода – 32 разряда.

#### Формат 4

OP #5/S,D [M]	<XRAM>, R.L
	R.L, <XRAM>

В этом формате 2-адресная (или одноадресная) вычислительная операция сочетается с параллельной пересылкой между ячейкой памяти данных <XRAM> и 32-разрядным регистром R.L регистрового файла. Ячейка памяти данных <XRAM> адресуется при помощи одного из адресных регистров в соответствии с режимами адресации, поддерживаемыми адресным генератором AGU.

Длина кода - 32 разряда.

#### Формат 5

OP #5/S,D [M]	R, R
	R.L, R.L

В формате 5 двухадресная (или одноадресная) вычислительная операция сочетается с параллельной пересылкой между двумя регистрами данных (оба регистра, источник и приемник, имеют одинаковую разрядность).

Длина кода - 32 разряда.

#### Формат 6

OP #5/S,D [M]	R, RC
	RC, R

В формате 6 двухадресная (или одноадресная) вычислительная операция сочетается с параллельной пересылкой между регистром данных и регистром управления. При обменах с регистрами-аккумуляторами AC0, AC1 используются 32-разрядные регистры данных, в остальных случаях - 16-разрядные.

Длина кода - 32 разряда.

#### Формат 6t

MOVE.cc	<XRAM>, R.L
	R.L, <XRAM>

Формат 6t является производным от формата 4 и применяется для кодирования условных пересылок между ячейкой памяти данных <XRAM> и 32-разрядным регистром регистрового файла.

Длина кода - 32 разряда.

### Формат 7

OP.cc S,D[M]	#16,R
	#16,RC
	#32,R.L

В формате 7 двухадресная (или одноадресная) вычислительная операция сочетается с параллельной пересылкой непосредственных данных, содержащихся во втором слове инструкции, в регистр данных или управления.

Длина кода - 64 разряда.

### Формат 7t

MOVE.cc	<XRAM(Ai+#16)>, R.L
	R.L, <XRAM(Ai+#16)>

Формат 7t является производным от формата 6t и применяется для кодирования условных пересылок между ячейкой памяти данных <XRAM>, адресуемой в режиме непосредственного смещения адреса (величина смещения #16 содержится во втором слове инструкции) и 32-разрядным регистром регистрового файла.

Длина кода - 64 разряда.

### Формат 8a

OP2 #5/S3,S4,D2	OP1[s] S1,S2,D [M]	<XRAM>, R.L	<YRAM>, R0
		R.L, <XRAM>	

В формате 8a выполняются две вычислительных операции (3-, 2- или 1-адресных) в сочетании с двумя параллельными пересылками, одна из которых - между ячейкой памяти данных <XRAM> и 32-разрядным регистром R.L регистрового файла, другая - из ячейки памяти данных <YRAM> в 32-разрядный регистр R0 регистрового файла. Память данных <XRAM>, <YRAM> адресуется в соответствии с режимами адресации, поддерживаемыми соответствующими адресными генераторами AGU, AGU-Y.

Длина кода - 64 разряда.

**Формат 8b**

OP2 #5/S3,S4,D2	OP1[s] S1,S2,D [M]	R, R	<YRAM>, R0
		R.L, R.L	

В формате 8b выполняются две вычислительных операции (3-, 2- или 1-адресных) в сочетании с двумя параллельными пересылками, одна из которых - между двумя регистрами данных (16- или 32-разрядными), другая - из ячейки памяти данных <YRAM> в 32-разрядный регистр R0 регистрового файла. Память данных <YRAM> адресуется в соответствии с режимами адресации, поддерживаемыми AGU-Y.

Длина кода - 64 разряда.

**Формат 8c**

OP2[.cc] #5/S3,S4,D2	OP1[s] S1,S2,D [M]	R.L, R.L
----------------------	--------------------	----------

В формате 8c по специфицированному условию выполняются две вычислительных операции (3-, 2- или 1-адресных) в сочетании с одной пересылкой между двумя 32-разрядными регистрами R.L регистрового файла.

Длина кода - 64 разряда.

**Формат 8d**

OP2 #5/S3,S4,D2	OP1[s] S1,S2,D [M]	R, RC
		RC, R

В формате 8d выполняются две вычислительных операции (3-, 2- или 1-адресных) в сочетании с одной пересылкой между регистром данных и регистром управления.

Длина кода - 64 разряда.

## 6.2 Два типа операций (OP1 и OP2)

Параллельное выполнение двух вычислительных операций в одной SIMD-секции DSP Elcore-26 возможно только при условии, что они выполняются на разных операционных устройствах (ОУ). Например, не могут одновременно исполняться операции AND и OR, так как обе они должны исполняться при помощи одного и того же логического устройства LU.

Вычислительные команды в зависимости от исполняющего их операционного устройства можно разделить на две большие группы (два типа) - OP1 и OP2. Только принадлежность к различным группам дает возможность двум операциям исполняться одновременно. К первому типу (OP1) относятся операции, исполняемые при помощи арифметического или логического устройства (AU, LU), ко второму типу (OP2) - операции, исполняемые при помощи умножителя-сдвигателя MS.

Ниже в таблице в алфавитном порядке дан перечень вычислительных операций DSP Elcore-26 с указанием их типа и форматов инструкций, в которых данные операции могут применяться, а также кода операции (КОП).

Мнемоника	Тип	Содержание команды (формат данных)	Форматы	КОП
ABS	OP1	Абсолютное значение (short)	1,4,5,6,7,8	001 0000
ABSL	OP1	Абсолютное значение (long)	1,4,5,6,7,8	011 0000
ADC	OP1	Сложение с переносом (short)	1,2,3,4,5,6,7,8	000 0101
ADCL	OP1	Сложение с переносом (long)	1,2,4,5,6,7,8	010 0101
ADC16L	OP1	Сложение смешанное	1,2,4,5,6,7,8	000 1000
ADD	OP1	Сложение (short)	1,2,3,4,5,6,7,8	000 0100
ADDL	OP1 OP2	Сложение (long)	1,2,4,5,6,7,8	010 0100 111 1110
ADDLR	OP1	Сложение (long) с округлением	1,2,4,5,6,7,8	010 1001
ADDLRT R	OP1	Сложение (long) с округлением и преобразованием формата (в short)	1,4,5,6,7,8	010 1010
ADDSUB	OP1	Сложение-вычитание (short)	1,4,5,6,8c,8d	000 0111
ADDSUB L	OP1	Сложение-вычитание (long)	1,4,5,6,8c,8d	001 1011
ADDSUB X	OP1	Сложение-вычитание (X16)	1,4,5,6,8c,8d	010 0000
ADDX	OP1	Сложение комплексное (X16)	1,2,4,5,6,7,8	010 0111
ADI	OP1	Сложение и инкремент (short)	1,2,3,4,5,6,7,8	010 1111
AND	OP1	Логическое И (short)	1,2,3,4,5,6,7,8	100 0001
ANDC	OP1	Логическое И с инверсией (short)	1,2,3,4,5,6,7,8	100 0010
ANDCL	OP1	Логическое И с инверсией (long)	1,2,4,5,6,7,8	101 0010
ANDI	OP1	Инверсия логического И (short)	1,2,3,4,5,6,7,8	100 0011
ANDL	OP1	Логическое И (long)	1,2,4,5,6,7,8	101 0001
ASH	OP1	Сложение и вычитание двух пар чисел (short)	1,4,5,6,7,8	011 1110
ASL	OP2	Арифметический сдвиг влево (short)	1,4,5,6,7,8	110 0100
ASLL	OP2	Арифметический сдвиг влево (long)	1,4,5,6,7,8	110 1100
ASLX	OP2	Арифметический сдвиг влево (X16)	1,4,5,6,7,8	110 0101
ASR	OP2	Арифметический сдвиг вправо (short)	1,4,5,6,7,8	111 0100
ASRL	OP2	Арифметический сдвиг вправо (long)	1,4,5,6,7,8	111 1100
ASRLE	OP2	Условный арифметический сдвиг вправо (long)	1,4,5,6,8	110 1101
ASRX	OP2	Сдвиг арифметический вправо (X16)	1,4,5,6,7,8	111 0101
B	OP1	Ветвление программы	3m,3mb	001 1100
BD	OP1	Ветвление программы (отложенное)	3m,3mb	001 1110
BS	OP1	Вызов подпрограммы	3m,3mb	010 1100
BTST	OP2	Проверка разряда (short)	4,5,6,7,8	111 0000
BTSTL	OP2	Проверка разряда (long)	4,5,6,7,8	111 1010
CLR	OP1	Обнуление (очистка) регистра (short)	1,4,5,6,7,8	000 0001
CLRL	OP1	Обнуление (очистка) регистра (long)	1,4,5,6,7,8	010 0001
CMP	OP1	Сравнение (short)	1,2,3,4,5,6,7,8	001 0101
CMPE	OP1	Сравнение экспонент	1,2,4,5,6,7,8	010 1110
CMPL	OP1	Сравнение (long)	1,2,4,5,6,7,8	011 0101
CMPM	OP1	Сравнение модулей (short)	1,2,3,4,5,6,7,8	001 0110
CMPML	OP1	Сравнение модулей (long)	1,2,4,5,6,7,8	011 0110
CS2	OP2	Парная операция выбора большего из двух чисел (short) с фиксацией бита выбора	1,8	110 0110
CVEF	OP1	Преобразование формата: формат 32E16 в 24E8	1,4,5,6,7,8	001 1100
CVFE	OP1	Преобразование формата: формат 24E8 в 32E16	1,4,5,6,8c,8d	001 1101
CVFI	OP1	Преобразование формата: формат 24E8 в	1,4,5,6,7,8	000 1110

Мнемоника	Тип	Содержание команды (формат данных)	Форматы	КОП
		32-разрядное целое в дополнительном коде		
CVIF	OP1	Преобразование формата: 32-разрядное целое в дополнительном коде со знаком в формат 24E8	1,4,5,6,7,8	000 1111
DEC	OP1	Декремент (short)	1,4,5,6,7,8	001 0010
DECL	OP1	Декремент (long)	1,4,5,6,7,8	011 0010
DISPFX	OP1	Распаковка (дробная) X8 в X16	1,4,5,6,7,8	100 1110
DISPX	OP1	Распаковка (целочисленная) X8 в X16	1,4,5,6,7,8	100 1101
DO DOR DO_R DOR_R	OP1	Оператор цикла	2d,3	000 1100 000 1101 000 1110 000 1111
DOFOR DOFORR	OP1	Оператор бесконечного цикла	3	000 1010 000 1011
ENDDO	OP1	Окончание цикла	3	001 1011
EOR	OP1	Логическое исключающее ИЛИ (short)	1,2,3,4,5,6,7,8	100 1000
EORL	OP1	Логическое исключающее ИЛИ (long)	1,2,4,5,6,7,8	101 1000
FADD	OP1	Сложение (24E8)	1,2,4,5,6,7,8	000 1010
FAS	OP1	Сложение-вычитание (24E8)	1,4,5,6,8c,8d	000 1011
FIN	OP2	1-я итерация обратной величины	1,4,5,6	001 1110
FINR	OP2	1-я итерация обратной величины квадратного корня	1,4,5,6	001 1111
FINT	OP1	Округление к ближайшему целому (24E8)	1,4,5,6,7,8	000 1101
FLOOR	OP1	Округление к ближайшему целому (24E8)	1,4,5,6,7,8	000 1100
FMPY	OP2	Умножение (24E8)	1,2,4,5,6,7,8	110 1111
FSUB	OP1	Вычитание (24E8)	1,2,4,5,6,7,8	010 1101
FTR	OP1	Преобразование формата	1,4,5,6,7,8	000 0110
FTRFL	OP1	Преобразование формата	1,4,5,6,7,8	000 1001
FTRL	OP1	Преобразование формата	1,4,5,6,7,8	010 0110
FTST	OP1	Определение признаков операнда (24E8)	1,4,5,6,7,8	010 1100
INC	OP1	Инкремент (short)	1,4,5,6,7,8	000 0010
INCL	OP1	Инкремент (long)	1,4,5,6,7,8	010 0010
INSL	OP1	Побитное мультиплексирование (long)	1,8	101 0100
J	OP1	Программный переход	3m,3mb	001 1101
JD	OP1	Программный переход (отложенный)	3m,3mb	001 1110
JS	OP1	Вызов подпрограммы	3m,3mb	010 1101
LSL	OP2	Логический сдвиг влево (short)	1,4,5,6,7,8	110 0001
LSLL	OP2	Логический сдвиг влево (long)	1,4,5,6,7,8	110 1000
LSLX	OP2	Логический сдвиг влево (X16)	1,4,5,6,7,8	110 0010
LSR	OP2	Логический сдвиг вправо (short)	1,4,5,6,7,8	111 0001
LSRL	OP2	Логический сдвиг вправо (long)	1,4,5,6,7,8	111 1000
LSRX	OP2	Логический сдвиг вправо (X16)	1,4,5,6,7,8	111 0010
MAC	OP2	Умножение целое со знаком (short) и накопление (в формате __Int64)	1,4,5,6,8	110 1001
MACL	OP2	Умножение целое со знаком (long) и накопление (в формате __Int64)	1,4,5,6,8	111 1011



Мнемоника	Тип	Содержание команды (формат данных)	Форматы	КОП
MACX	OP2	Умножение дробное комплексно-сопряженное (X8) и целочисленное (X16)	1,4,5,6,8	111 1001
MAC2	OP2	Парное умножение (short) и накопление 2-х результатов (в формате long)	1,4,5,6,8	110 1110
MAX	OP1	Выбор большего числа (short)	1,2,3,4,5,6,7,8	001 0111
MAXL	OP1	Выбор большего числа (long)	1,2,4,5,6,7,8	011 0111
MAXM	OP1	Выбор числа с большим модулем (short)	1,2,3,4,5,6,7,8	001 1001
MAXML	OP1	Выбор числа с большим модулем (long)	1,2,4,5,6,7,8	011 1001
MIN	OP1	Выбор меньшего числа (short)	1,2,3,4,5,6,7,8	001 1000
MINL	OP1	Выбор меньшего числа (long)	1,2,4,5,6,7,8	011 1000
MINM	OP1	Выбор числа с меньшим модулем (short)	1,2,3,4,5,6,7,8	001 1010
MINML	OP1	Выбор числа с меньшим модулем (long)	1,2,4,5,6,7,8	011 1010
MOVE	OP3	Пересылка данных	3,4,5,6,7,8,2t,6t,7t	110 1101 110 1111 110 0111
MPF	OP2	Умножение дробное со знаком (short)	1,2,3,4,5,6,7,8	111 1101
MPF2	OP2	Парное умножение дробное со знаком (short)	1,4,5,6,8	111 1101
MPF2S	OP2	Парное умножение дробное со знаком (short), с перестановкой сомножителей	1,4,5,6,8	110 0011
MPSS	OP2	Умножение целое со знаком (short)	1,2,3,4,5,6,7,8	111 1110
MPUU	OP2	Умножение целое без знака (short)	1,2,3,4,5,6,7,8	111 1011
MPX	OP2	Умножение дробное комплексное (X8), второй операнд - комплексно-сопряженный	1,2,4,5,6,7,8	111 0110
MPYL	OP2	Умножение целое со знаком (long)	1,4,5,6,8	111 0011
MSKG	OP2	Формирование маски (short)	1,4,5,6,7,8	110 1010
MSKGL	OP2	Формирование маски (long)	1,4,5,6,7,8	110 1011
NEG	OP1	Изменение знака (short)	1,4,5,6,7,8	001 0001
NEGL	OP1	Изменение знака (long)	1,4,5,6,7,8	011 0001
NOP	OP1 OP2	Пустая операция	3,8	100 0000 110 0000
NOT	OP1	Логическое отрицание (short)	1,4,5,6,7,8	100 1001
NOTL	OP1	Логическое отрицание (long)	1,4,5,6,7,8	101 1001
OR	OP1	Логическое ИЛИ (short)	1,4,5,6,7,8	100 0101
ORC	OP1	Логическое ИЛИ с инверсией (short)	1,2,3,4,5,6,7,8	100 0110
ORCL	OP1	Логическое ИЛИ с инверсией (long)	1,2, 4,5,6,7,8	101 0101
ORI	OP1	Инверсия логического ИЛИ (short)	1,2,3,4,5,6,7,8	100 0111
ORL	OP1	Логическое ИЛИ (long)	1,2,4,5,6,7,8	101 0101
PACK	OP1	Упаковка (short)	1,4,5,6,7,8	100 1100
PACKL	OP1	Упаковка (long)	1,4,5,6,7,8	101 1100
PDN	OP1	Определение параметра денормализации (short)	1,4,5,6,7,8	100 1111
PDNE	OP1	Измерение параметра денормализации 16-разрядной мантиссы	1,4,5,6,7,8	100 0100
PDNL	OP1	Определение параметра денормализации (long)	1,4,5,6,7,8	101 1111
PDNLE	OP1	Измерение параметра денормализации 32-	1,4,5,6,7,8	101 1110

Мнемоника	Тип	Содержание команды (формат данных)	Форматы	КОП
		разрядной мантиссы		
PDNX	OP1	Определение параметра денормализации (X16)	1,4,5,6,7,8	101 1101
RNDL	OP1	Округление	1,4,5,6,7,8	010 1000
ROL	OP2	Циклический сдвиг на один разряд влево (short)	1,4,5,6,7,8	110 0011
ROLL	OP2	Циклический сдвиг на один разряд влево (long)	1,4,5,6,7,8	110 1001
ROR	OP2	Циклический сдвиг на один разряд вправо (short)	1,4,5,6,7,8	111 0011
RORL	OP2	Циклический сдвиг на один разряд вправо (long)	1,4,5,6,7,8	111 1001
RTS	OP1	Возврат из подпрограммы	3	010 0000
SAC2	OP2	Парное накопление (в формате long) со знаком	1,8	111 0110
SAH	OP1	Сложение и вычитание двух пар чисел (short)	1,4,5,6,7,8	011 1111
SBC	OP1	Вычитание с переносом (short)	1,2,3,4,5,6,7,8	001 0100
SBCL	OP1	Вычитание с переносом (long)	1,2,4,5,6,7,8	011 0100
SMB	OP2	Сложение бит (short)	4,5,6,7,8	111 0111
SMBL	OP2	Сложение бит (long)	4,5,6,7,8	111 0111
STOP	OP1	Останов	3	011 1110
SUB	OP1	Вычитание (short)	1,2,3,4,5,6,7,8	001 0011
SUBL	OP1 OP2	Вычитание (long)	1,2,4,5,6,7,8	011 0011 111 1111
SUBLR	OP1	Вычитание (long) с округлением	1,2,4,5,6,7,8	011 1100
SUBLRT R	OP1	Вычитание (long) с округлением и преобразованием формата (в short)	1,2,4,5,6,7,8	011 1101
SUBX	OP1	Вычитание комплексное (X16)	1,2,4,5,6,7,8	011 1011
SWL	OP1	Перестановка (long)	1,2,4,5,6,7,8	101 1011
TR	OP1 OP2	Транзит (short)	4,5,6,7,8	100 1010 110 0110
TRL	OP1 OP2	Транзит (long)	4,5,6,7,8	101 1010 110 1110
TST	OP1	Определение признаков операнда (short)	1,4,5,6,7,8	000 0011
TSTL	OP1	Определение признаков операнда (long)	1,4,5,6,7,8	010 0011
TSTX	OP1	Определение признаков операнда (X16)	1,4,5,6,7,8	010 1011

### 6.3 Коды операций

Ниже в таблице приведены коды всех операций (КОП) DSP Elcore-26, включая операции управления и пересылок (выделенные жирным шрифтом).

AU				LU + BFU		MS	
КОП	Операция	КОП	Операция	КОП	Операция	КОП	Операция
0000000	NOP	0100000	RTS/ADDSUB X	1000000	NOP	1100000	NOP
0000001	CLR	0100001	CLRL	1000001	AND	1100001	LSL
0000010	INC	0100010	INCL	1000010	ANDC	1100010	LSLX
0000011	TST	0100011	TSTL	1000011	ANDI	1100011	ROL/MPF2S
0000100	ADD	0100100	ADDL	1000100	PDNE	1100100	ASL
0000101	ADC	0100101	ADCL	1000101	OR	1100101	ASLX
0000110	FTR	0100110	FTRL	1000110	ORC	1100110	TR/CS2
0000111	ADDSUB	0100111	ADDX	1000111	ORI	1100111	MOVE.cc
0001000	ADC16L	0101000	RNDL	1001000	EOR	1101000	LSLL
0001001	FTRFL	0101001	ADDLR	1001001	NOT	1101001	ROLL/MAC
0001010	DOFOR/FADD	0101010	ADDLRTR	1001010	TR	1101010	MSKG
0001011	DOFORR/FAS	0101011	TSTX	1001011		1101011	MSKGL
0001100	DO/FLOOR	0101100	BS/FTST	1001100	PACK	1101100	ASLL
0001101	DOR/FINT	0101101	JS/FSUB	1001101	DISPX	1101101	MIA/ASRLE
0001110	DO_R/CVFI	0101110	CMPE	1001110	DISPFX	1101110	TRL/MAC2
0001111	DOR_R/CVIF	0101111	ADI	1001111	PDN	1101111	MIP/FMPY
0010000	ABS	0110000	ABSL	1010000		1110000	BTST
0010001	NEG	0110001	NEGL	1010001	ANDL	1110001	LSR
0010010	DEC	0110010	DECL	1010010	ANDCL	1110010	LSRX
0010011	SUB	0110011	SUBL	1010011		1110011	ROR/MPYL
0010100	SBC	0110100	SBCL	1010100	INSL	1110100	ASR
0010101	CMP	0110101	CMPL	1010101	ORL	1110101	ASRX
0010110	CMPM	0110110	CMPML	1010110	ORCL	1110110	MPX/SAC2
0010111	MAX	0110111	MAXL	1010111		1110111	SMB/SMBL
0011000	MIN	0111000	MINL	1011000	EORL	1111000	LSRL
0011001	MAXM	0111001	MAXML	1011001	NOTL	1111001	RORL/MACX
0011010	MINM	0111010	MINML	1011010	TRL	1111010	BTSTL
0011011	ENDDO /ADDSUBL	0111011	SUBX	1011011	SWL	1111011	MPUU/MACL
0011100	B/CVEF	0111100	SUBLR	1011100	PACKL	1111100	ASRL
0011101	J/CVFE	0111101	SUBLRTR	1011101	PDNX	1111101	MPF/MPF2
0011110	BD/FIN	0111110	STOP/ASH	1011110	PDNLE	1111110	MPSS/ADDL
0011111	JD/FINR	0111111	SAH	1011111	PDNL	1111111	SUBL

### 6.3.1 Операции с несколькими вариантами кодировки

Операции MOVE, DO, DOFOR могут иметь различные варианты кодировки в зависимости от параметров данных операций.

#### Варианты кодировки операции MOVE

Операция пересылки может иметь следующие варианты кодировки:

- 1) Операции MOVE *не имеет кода операции* в тех случаях, когда наличие и тип пересылки определяется кодом формата;
- 2) Условные пересылки имеют код *MOVE.cc* (1100111);
- 3) Пересылки непосредственных данных в регистры адресных генераторов AGU, AGU-Y имеют код *MIA* (1101101);
- 4) Пересылки непосредственных данных в регистры адресных генераторов AGU, AGU-Y имеют код *MIP* (1101111).

#### Варианты кодировки операции DO

Операция цикла DO может иметь два варианта кодировки:

- 1) Операция DO с абсолютной адресацией последней команды цикла и непосредственно заданным числом повторений имеет код *DO* (0001100);
- 2) Операция DO с адресацией последней команды цикла относительно программного счетчика PC и непосредственно заданным числом повторений имеет код *DOR* (0001101);
- 3) Операция DO с абсолютной адресацией последней команды цикла и косвенно заданным числом повторений имеет код *DO\_R* (0001110);
- 4) Операция DO с адресацией последней команды цикла относительно программного счетчика PC и косвенно заданным числом повторений имеет код *DOR\_R* (0001111).

#### Варианты кодировки операции DOFOR

Операция бесконечного цикла DOFOR может иметь два варианта кодировки:

- 1) Операция DOFOR с абсолютной адресацией последней команды цикла имеет код *DOFOR* (0001010);
- 2) Операция DOFOR с адресацией последней команды цикла относительно программного счетчика PC имеет код *DOFORR* (0001011).

### 6.3.2 Расширение кода операции за счет формата инструкции и бита #

В некоторых случаях двум различным операциям соответствует одинаковый код. Выбор исполняемой операции в этих случаях производится с учетом либо кода формата, либо бита # (бит непосредственного операнда #5 в командах сдвига).

В тех случаях, когда код вычислительной операции совпадает с кодом операции управления или пересылки, выбор исполняемой операции определяется форматом инструкции согласно таблице:

КОП	Формат	
	2d, 3	другие
0001010	DOFOR	FADD
0001011	DOFORR	FAS
0001100	DO	FLOOR
0001101	DOR	FINT
0001110	DO_R	CVFI
0001111	DOR_R	CVIF
0011011	ENDDO	ADDSUBL
0011100	B	CVEF
0011101	J	CVFE
0011110	BD	FIN
0011111	JD	FINR
0100000	RTS	ADDSUBX
0101100	BS	FTST
0101101	JS	FSUB
0111110	STOP	ASH
0111111	-	SAH
1101101	MOVE(MIA)	ASRLE
1101111	MOVE(MIP)	FMPY

Для кодирования команд, реализуемых на базе устройства умножения-сдвига (MS) и относящихся к группе операций OP2, производится расширение кода операции за счет бита # (бит непосредственного операнда #5 в командах сдвига) согласно таблице:

КОП	Операция	
	# = 0	# = 1
1100011	ROL	MPF2S
1100110	TR	CS2
1101001	ROLL	MAC
1101110	TRL	MAC2
1110011	ROR	MPYL
1110110	MPX	SAC2
1110111	SMB	SMBL
1111001	RORL	MACX
1111011	MPUU	MACL
1111101	MPF	MPF2
1111110	MPSS	ADDL(OP2)
1111111	-	SUBL(OP2)

#### 6.4 Таблица полей форматов инструкций.

Участки кода инструкции, определяющие те или иные параметры входящих в нее операций, называются полями инструкции. Ниже приводится таблица полей форматов инструкций, в которой определяются структура форматов инструкций и назначение входящих в них полей.

Таблица полей форматов инструкций

Разряды слова инструкции																												Формат			
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1 / s1 / #5					D / d				S2 / s2				#	0	cc				0	0	0	1	OP					1			
xxxxx					D / d				S2 / s2				0	0	cc				0	0	1	0	OP					2			
#32 / #I16																															
xxxxx					#16																0	0	1	0	OP					2d	
#16																															
d					#16																0	0	1	1	OP					3	
cc				1	#16																0	0	1	1	OP					3 m	
cc				0	xxxxx								A								0	0	1	1	OP					3mb	
S / s / #5					D / d				R				A				mode		u	0	1	de	#	OP					4		
xxxxx					cc				0	R				A				mode		u	0	1	de	0	OP					6t	
xxxxx					cc				1	R				A				010		0	0	1	de	0	OP					7t	
#16																															
S / s / #5					D / d				R				sc	RC				1	0	d	#	OP					6				
xxx			sr	L	cc				s/d/s/S				sc	RC/RC/d/D				1	0	d	0	OP					2t				

Разряды слова инструкции																											Формат					
3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	14	1	1	1	1	9	8	7	6	5	4	3	2	1	0	
S / s / #5					D / d					RS /Rs					L	RD /Rd					1	1	0	#	OP					5		
S / s					D / d					RD /Rd / RC					L/sc	sr	cc					1	1	1	0	OP					7	
#32 / #16																																
S1/s1					D/d					S2/s2					M	R					u	0	0	0	0	OP1					8a	
S3/s3/ #5					D2/d2					S4/s4					AT		mode			A			d	OP3	#	OP2			OP3=00			
S1/s1					D/d					S2/s2					M	RS/Rs					0	0	0	0	OP1					8b		
S3/s3/ #5					D2/d2					S4/s4					AT		RD/Rd					L	x	OP3	#	OP2			OP3=01			
S1/s1					D/d					S2/s2					M	RS			x	0	0	0	0	OP1					8c			
S3/s3/ #5					D2/d2					S4/s4					cc		RD			x	cc			OP3	#	OP2			OP3=10			
S1/s1					D/d					S2/s2					M	Rs/Rd					0	0	0	0	OP1					8d		
S3/s3/ #5					D2/d2					S4/s4					xx		RC			sc	d	OP3	#	OP2			OP3=11					

Обозначения:

- **S, D, RS, RD, R** – 32-разрядные регистры регистрового файла
- **s, d, Rs, Rd** – 16-разрядные регистры регистрового файла
- **#n** – n-разрядные непосредственные данные
- **RC** – управляющие регистры
- **cc** – код условия
- **A** – номер генератора адреса
- **mode** – код режима адресации AGU
- **u** – код блокировки обмена (0 – обмен, 1 – блокировка)
- **x** – произвольное значение разряда
- **de** – направление обмена (de=1 при засылке в регистры регистрового файла)
- **L** – разрядность регистров (0 – 16 разрядов, 1 – 32 разряда)
- **M** – признак масштабирования результата
- **#** – признак выбора непосредственного операнда, например, короткого #5 в операциях сдвига
- **sr** – выбор группы регистров (0 – управление, 1 – RF)
- **sc** – выбор группы регистров управления (0 – AGU, 1 – PCU)

## 6.5 Описание полей форматов инструкций

В настоящем разделе содержится детальное описание кодировки полей инструкции.

Здесь и в дальнейшем, при описании форматов инструкций, приняты следующие *соглашения по обозначениям* (соответствующие правилам языка Verilog):

1) Разряды формата данных (операнда) указываются в квадратных скобках, через двоеточие.

Пример:  $S[4:0]$ .

2) Слева указывается старший разряд (в примере - 4-й), справа - младший (0-й).



### 6.5.1 Основное и дополнительное поле кода формата инструкции

Разряды [10:7] первого слова каждой инструкции составляют **основное поле кода формата**, в котором содержится код формата данной инструкции. Производные форматы 2d, 2t, 3m, 3mb, 4t, 6t, 7t определяются не только кодом формата, но и кодом инструкции.

Код формата (основное поле)	Формат
0 0 0 1	1
0 0 1 0	2
0 0 1 1	3
0 1 x x	4
1 0 x x	5
1 1 0 x	6
1 1 1 0	7
0 0 0 0	8

Выбор среди форматов параллельных инструкций 8a, 8b, 8c, 8d осуществляется при помощи **дополнительного поля кода формата** (OP3), размещенного в разрядах [7:6] второго слова инструкции.

Код формата (дополнительное поле)	Формат
0 0	8a
0 1	8b
1 0	8c
1 1	8d

### 6.5.2 Основное и дополнительное поле кода операции

Разряды [6:0] первого слова каждой инструкции составляют **основное поле кода операции**, в котором содержится код исполняемой операции.

В параллельных инструкциях (формат 8) данное поле отводится для кодирования операции OP1, исполняемое на арифметико-логическом устройстве, а для кода операции OP2, исполняемой на умножителе-сдвигателе, отводится **дополнительное поле кода операции** (разряды [4:0] второго слова инструкции). Поле OP2 содержит только 5 младших разрядов кода операции, так как оба старших разряда кода операции для всех операций умножителя-сдвигателя равны единице.

### 6.5.3 Поле кода условия (cc)

Коды условия (cc) для условно исполняемых инструкций приведены в таблице.

№ кода	Код условия	Условие	Мнемоника
0	0000	$C = 0$	cc (Carry Clear) / hs (Higher or Same)
1	0001	$C = 1$	cs (Carry Set) / lo (LOwer)
2	0010	$Z = 0$	ne (Not Equal to zero)
3	0011	$Z = 1$	eq (EQual to zero)
4	0100	$N = 0$	pl (Plus)
5	0101	$N = 1$	mi (MInus)
6	0110	$N \wedge V = 0$	ge (GrEater than or equal)
7	0111	$N \wedge V = 1$	lt (Less Than)
8	1000	$Z   (N \wedge V) = 0$	gt (Greater Than)
9	1001	$Z   (N \wedge V) = 1$	le (Less than or Equal)
10	1010	$U = 0$	nr (NoRmalized)
11	1011	$U \& (\sim V) = 1$	un (UnNormalized)
12	1100	$V = 1$	vs (oVerflow Set)
13	1101	$V = 0$	vc (oVerflow clear)
14	1110	$t = 1$	t( признак истинности условия после исполнения условной команды)
15	1111	–	al (Always)

Поле кода условия размещается:

- Для форматов 1, 2, 7 - в разрядах [14:11] первого слова инструкции;
- Для форматов 3m, 3mb - в разрядах [31:28] первого слова инструкции;
- Для формата 8c - в разрядах {[16:15],[9:8]} второго слова инструкции.

### 6.5.4 Поля адресов операндов в вычислительных операциях (S/s/D/d)

Исходные операнды и результаты вычислительных операций помещаются в регистрах реконфигурируемого регистрового файла (RF), который представляет собой набор из 32-х 16-разрядных регистров R0–R31, которые могут быть реконфигурированы в 16 32-разрядных регистров.

Регистровый файл состоит из двух банков 16-разрядных регистров: нулевого банка с четными адресами (R0, R2, ..., R30) и первого банка - с нечетными адресами (R1, R3, ..., R31). При 32-разрядных операциях соответствующие регистры двух банков объединяются попарно и образуют 16 32-разрядных регистров, причем младшие 16 бит хранятся в регистрах с четными номерами, старшие 16 бит - в регистрах с нечетными номерами.

Адрес регистра RF определяется 5-разрядным кодом согласно приводимой ниже таблице (регистры данных).

Адреса 32-разрядных операндов могут быть только четными, адреса 16-разрядных операндов - любыми.

В таблице полей форматов поля адресов операндов обозначаются следующим образом:

S, S1, S2, S3, S4, D, D1, D2 – 32-разрядные регистры регистрового файла;

s, s1, s2, s3, s4, d, d1, d2 – 16-разрядные регистры регистрового файла.

### 6.5.5 Поля адресов операндов в операциях пересылок (R/RC/RS/Rs/RD/Rd). Поля выбора типа регистра (sr, sc).

Адрес регистра, являющегося источником либо получателем в операциях пересылок, определяется 5-разрядным (при 32-разрядных пересылках - 4-разрядным) кодом (R/RC/RS/Rs/RD/Rd) и полями выбора типа регистра (sr, sc) согласно таблице:

sr	sc	Регистр	Код адреса регистра (R/RC/RS/Rs/RD/Rd)
<u>Регистры данных (R/RS/Rs/RD/Rd)</u>			
1	X	R0	00000
1	X	R1	00001
1	X	R2	00010
1	X	R3	00011
1	X	R4	00100
1	X	R5	00101
1	X	R6	00110
1	X	R7	00111
1	X	R8	01000
1	X	R9	01001
1	X	R10	01010
1	X	R11	01011
1	X	R12	01100
1	X	R13	01101
1	X	R14	01110
1	X	R15	01111
1	X	R16	10000
1	X	R17	10001
1	X	R18	10010
1	X	R19	10011
1	X	R20	10100
1	X	R21	10101
1	X	R22	10110
1	X	R23	10111
1	X	R24	11000
1	X	R25	11001
1	X	R26	11010
1	X	R27	11011
1	X	R28	11100
1	X	R29	11101
1	X	R30	11110
1	X	R31	11111

**RS, RD, R** – 32-разрядные регистры регистрового файла;

**Rs, Rd** – 16-разрядные регистры регистрового файла;

**RC** – 16-разрядный регистр управления (AC0, AC1 - 32-разрядные).

sr	sc	Регистр	Код адреса регистра
<u>Регистры управления (RC) – адресные генераторы</u>			
AGU			
0	0	A0	00000
0	0	A1	00001
0	0	A2	00010
0	0	A3	00011
0	0	A4	00100
0	0	A5	00101
0	0	A6	00110
0	0	A7	00111
0	0	I0	01000
0	0	I1	01001
0	0	I2	01010
0	0	I3	01011
0	0	I4	01100
0	0	I5	01101
0	0	I6	01110
0	0	I7	01111
0	0	M0	10000
0	0	M1	10001
0	0	M2	10010
0	0	M3	10011
0	0	M4	10100
0	0	M5	10101
0	0	M6	10110
0	0	M7	10111
<u>AGU-Y – секционные регистры</u>			
0	0	AT	11000
0	0	IT	11001
0	0	MT	11010
0	0	DT	11011
<u>Регистры управления (RC) – PCU</u> <u>и секционные регистры CCR, PDNR, AC0, AC1</u>			
PCU			
0	1	DCSR	00000
0	1	SR	00001
0	1	IDR	00010
0	1	PC	01000
0	1	SS	01001
0	1	LA	01010
0	1	CSL	01011
0	1	LC	01100
0	1	CSH	01101
0	1	SP	01110
0	1	SAR	01111
0	1	CNTR	10000

sr	sc	Регистр	Код адреса регистра
CCR, PDNR, AC0, AC1 – секционные регистры			
0	1	PDNR	11000
0	1	CCR	11001
0	1	AC0	11010
0	1	AC1	11011

Так как адреса 32-разрядных операндов могут быть только четными, для их кодирования достаточно 4-х разрядов (поле R в форматах 4, 6t, 7t, 8a; поля RS, RD в формате 8c). В этих случаях код адреса регистра соответствует 4-м старшим разрядам приведенного в таблице 5-разрядного кода.

Для операции чтения из Y-памяти по адресу, сформированному генератором AGU-Y (<Пересылка 2>) приемником всегда является регистр R0.L, таким образом, кодировка регистра-получателя для такой операции не требуется.

### 6.5.6 Поле формата пересылки (L)

Поле L определяет формат данных в операциях пересылок между регистрами RF (форматы 5, 2t, 8b) либо непосредственных данных в регистр RF (формат 7). При L=0 выполняется 16-разрядная пересылка, при L=1 - 32-разрядная.

Все обмены с памятью данных - 32-разрядные. Разрядность пересылок между регистрами управления и данных определяется разрядностью регистра управления, участвующего в пересылке: при обменах между регистрами RF и регистрами-аккумуляторами AC0, AC1 - пересылки 32-разрядные, в остальных случаях - 16-разрядные.

### 6.5.7 Поле направления пересылки (de)

Поле **de** определяет направление перемещения данных в операциях пересылок между регистрами RF и памятью, либо между регистрами RF и регистрами управления. При de=1 производится запись в регистр RF, при de=0 – чтение из регистра RF.

### 6.5.8 Поле режима адресации AGU (mode) и поле кода блокировки обмена (u)

Поле **mode** определяет режим адресации памяти в операциях пересылок между регистрами RF и памятью данных под управлением адресного генератора AGU. Режимы адресации памяти приведены в таблице.

Код блокировки обмена **u** позволяет управлять выполнением пересылки независимо от режима адресации памяти. При u=0 пересылка выполняется, при u=1 модификация адреса An происходит в соответствии с режимом адресации без выполнения самой пересылки.

mode	u	Мнемоника	Режим адресации, наличие обмена	Автокод (ea)
000	X	-	Отсутствие пересылки	nm
001	X	(An)	Косвенная без модификации	k
100	0	(An)+	Пост - автоинкремент	i
100	1	(An)+	Пост – автоинкремент, отмена пересылки	ui
101	0	(An)-	Пост - автодекремент	d
101	1	(An)-	Пост – автодекремент, отмена пересылки	ud
110	0	(An)+In	Пост - автоувеличение	ai
110	1	(An)+In	Пост – автоувеличение, отмена пересылки	uai
111	0	(An)-In	Пост - автоуменьшение	ad
111	1	(An)-In	Пост – автоуменьшение, отмена пересылки	uad
011	X	(An+In)	Индексирование (An не меняется)	In
010	X	(An+dspl)	Адресация с непосредственным смещением (An не меняется)	Ds

### 6.5.9 Поле регистра адреса AGU (A)

Поле **A** определяет адресный регистр AGU, используемый для адресации памяти в операциях пересылок между регистрами RF и памятью данных. Кроме того, в формате 3mb адресный регистр определяет адрес программного перехода.

Поле A	Адресный регистр
000	A0
001	A1
010	A2
011	A3
100	A4
101	A5
110	A6
111	A7

### 6.5.10 Поле режима адресации AGU-Y (AT)

Поле **AT** совместно с управляющим параметром YM (11-й разряд регистра SR) определяет режим адресации Y-памяти в операциях пересылок между Y-памятью и регистрами RF, выполняемыми под управлением адресного генератора AGU-Y. Режимы адресации приведены в таблице.

Поле AT	YM	Мнемоника	Режим адресации	Автокод (ea)
00	X	-	Отмена пересылки	uT
01	X	(AT)	Косвенная	AT
10	X	(AT)+IT	Пост - автоувеличение	ATr
11	0	(AT)+IT	Индексирование (AT не меняется)	ATIT
11	1	(AT)+DT	Пост - автоувеличение	ATr

### 6.5.11 Поля непосредственных операндов (#5, #16, #32) и поле выбора непосредственного 5-разрядного операнда (#)

В вычислительных операциях, также как в операциях пересылок и управления, могут использоваться непосредственные операнды. Короткие 5-разрядные непосредственные операнды #5 могут использоваться в операциях ASL, ASLL, ASLX, ASR, ASRL, ASRX, BTST, BTSTL, LSL, LSLX, LSR, LSRL, LSRX, MSK, MSKL вместо операнда S1 (S). Выбор непосредственного операнда #5 для этих операций контролируется полем # (при #=0 в операции участвует регистровый операнд, при #=1 - непосредственный 5-разрядный операнд). Поле #, кроме того, используется для расширения поля кода операции (см. п.6.3.2).

Для непосредственных операндов #16, #32 используется либо специальное поле (в форматах 2d, 3, 3m разряды [26:11] используются для 16-разрядного непосредственного операнда #16) либо дополнительное (второе) слово инструкции.

### 6.5.12 Поле выбора режима *Scaling* (M)

Поле M в параллельных инструкциях (форматы 8a, 8b, 8c, 8d) позволяет установить (M=1) или отключить (M=0) режим *Scaling* в арифметических операциях, использующих этот режим.

Масштабирование выполняется путем арифметического сдвига результата операции вправо на 0/1/2 бита, при этом величина сдвига определяется полем SC (разряды 9, 8) регистра PDNR.

Включение режима масштабирования может быть выполнено двумя способами. Первый способ - установка в «1» бита 15 (ESC) регистра PDNR. Другой способ включения этого режима состоит в установке в «1» поля M непосредственно в командном слове (форматы 8a-8d). Синтаксически это выражается в добавлении к мнемоническому имени команды суффикса “s”, например, ADDLs, SUBXs и т.п. Подробно синтаксис и кодирование инструкций рассматриваются в следующем разделе.

Перечень операций, в которых может быть использован режим масштабирования, приведен ниже.

Long	Short	X16
<b>Блок AU</b>		
ABSL	ABS	
NEGL	NEG	
ADDL	ADD	ADDX
SUBL	SUB	SUBX
ADCL	ADC	
ADC16L	AD1	
SBCL	SBC	
ADDSUBL	ADDSUB	ADDSUBX
RNDL		
ADDLR	ASH	
SUBLR	SAH	
ADDLRTR		
SUBLRTR		
FTRL		

## 7. СИНТАКСИС И ОПИСАНИЕ ОПЕРАЦИЙ

### 7.1 Общие положения

Каждая инструкция может содержать до двух вычислительных операций и до двух операций пересылок.

Синтаксически инструкция записывается в одну строку, в которой поля вычислительных операций и пересылок отделены друг от друга некоторым количеством пробелов или табуляций. Каждая новая инструкция начинается с новой строки.

Операции, составляющие инструкцию, записываются в следующем порядке:

<Операция ОР2> < Операция ОР1> <Пересылка 1> <Пересылка 2>

Система инструкций DSP-ядра Elcore-26 является *ортогональной* в том смысле, что каждая операция, входящая в состав инструкции, кодируется независимо от других в соответствующих полях инструкции. Тем самым, имея описание способов кодировки для каждой из операций, можно однозначно определить код всей инструкции при любом сочетании составляющих ее операций.

В настоящем разделе приводится описание операций DSP-ядра Elcore-26. В описание каждой операции (команды) включено:

- Мнемоническое имя;
- Содержание исполняемой операции;
- Словесное описание операции с указанием специальных случаев и вырабатываемых признаков;
- Тип операции и разрешенные форматы;
- Синтаксис написания команды на ассемблере и кодировка соответствующих полей инструкции;
- Автокод (структура инструкции) – мнемоническая запись, позволяющая однозначно установить формат и код инструкции.

### 7.2 Соглашения по обозначениям

1) Для описания содержания исполняемой операции применяется обобщенный язык, базирующийся, в основном, на средствах языка С. Используются также математические символы  $| |$  и  $\sqrt{\quad}$  для обозначения абсолютного значения и квадратного корня числа.

Оператор присваивания выражается стрелкой  $\rightarrow$ . Присваивание выполняется слева направо. Например, содержание операции ADC (сложение с переносом) выражается следующей формулой:

$$s1 + s2 + C \rightarrow d;$$



которая означает, что вычисляется сумма операндов-источников  $s_1, s_2$  и входного переноса  $C$  и результат помещается в операнд-приемник  $d$ .

Кроме того, для описания манипуляций с отдельными битами и битовыми полями применяются конструкции языка Verilog, в частности:

- рассматриваемые разряды операнда указываются в квадратных скобках, через двоеточие, причем слева указывается старший разряд. Пример:  $D[30:23]$ ;
- фигурные скобки  $\{ \}$  означают конкатенацию битовых полей, старшие разряды стоят слева.

2) При описании синтаксиса инструкций используются следующие условные обозначения:

$\langle \rangle$  - в фигурные скобки заключаются операции (вычислительные или пересылки, для которых указывается лишь тип, без раскрытия операндов и параметров.

$[ ]$  - в квадратные скобки заключаются части инструкции, которые могут быть опущены. Это касается, например, команды **MOVE**, выполняющейся параллельно с вычислительной операцией –  $[M]$ ; модификаторов кода условия  $[.cc]$ ; включения режима **Scaling**  $[s]$  в арифметических командах 8-го формата.

**XRAM, YRAM** – ссылки на X- или Y- память данных;

**S, S1, S2, D** – 16- или 32-разрядный регистр данных;

**s, d, R** – 16-разрядный регистр данных;

**R.L** – 32-разрядный регистр данных;

**#x** – x-разрядные непосредственные данные;

**RC** – управляющий регистр;

**cc** – код условия;

**An** – адресный регистр,  $n=0,1,\dots,7$ .

3) В таблице признаков результата способ их формирования отмечается следующим образом:

$\surd$  - стандартный способ формирования;

\* - нестандартный способ формирования, оговаривается особо в самой таблице или примечании;

– - признак не формируется.

### 7.3 Автокод

В описание кодировки инструкции включается автокод.

Автокод (структура инструкции) – это мнемоническая запись, полностью и однозначно определяющая код инструкции. Автокод используется при дизассемблировании программ для DSP-ядра Elcore-26.

Генерация автокода производится по следующим правилам:

- 1) В форматах 2, 3, 4, 5, 6, 7 к имени вычислительной команды добавляется в качестве суффикса номер формата. Пример: ADD2 cc, 17, R2, R3.
- 2) В формате 8 в начале инструкции ставится ключевое слово PROL.
- 3) Некоторые операции в зависимости от своих параметров и типа входных операндов могут иметь различные варианты автокода:

Операция	Варианты автокода
DO	DO, DOR, DO_R, DOR_R
DOFOR	DOFOR, DOFORR
MOVE	MR, MRL, MRC, MRCd, MLM, MLR, MIA, MIP, MI, MIL, MIC
B	Bcc, BccI, Bccm
BD	BDcc, BDccI, BDccm
BS	BSc, BSccI, BSccm
J	Jcc, JccI, Jccm
JD	JDcc, JDccI, JDccm
JS	JSc, JSccI, JSccm

### 7.4 Описание инструкций

**ABS**
**Абсолютное значение (short)**
**ABS**
**Операция:** short s,d; |s| à d

**Описание:** В формате short вычисляется абсолютное значение операнда источника s и помещается в операнд-приемник d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	–	√	√	–

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** ABS[.cc] s,d

**Примеры:** ABS.ne R1,R5

**ABS** R1,R5

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	0	0	cc				0	0	0	1	0	0	1	0	0	0	0	

**Автокод:** ABS cc, s1, d

**Формат 4:** ABS s,d <XRAMBàR.L>

**Пример:** ABS R1,R2 R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode	u	0	1	de	0	0	0	1	0	0	0	0			

**Автокод:** ABS4 s, d, MLR,ea,A,R

**ABS4** s, d, MLM,ea,A,R

Формат 5: **ABS** s,d <R/R.L**β**àR/R.L>

Пример: **ABS** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	0	0	0	0				

Автокод: ABS5 s, d, MR,rs,rd

ABS5 s, d, MRL,RS,RD

Формат 6: **ABS** s,d <R**β**àRC>

Пример: **ABS** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	0	0	0	0				

Автокод: ABS6 s, d ,MRC,rs,RC

ABS6 s, d ,MRCd,RC,rd

Формат 7: **ABS[.cc]** s,d #16/32**à**RC/R/R.L>

Пример: **ABS** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	0	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ABS7 s, d, MIL,cc,#32,RD

ABS7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **ABS[s]** s,d <XRAM**β**àR.L> <YRAM**à**R0>

Примеры: LSRL R5,R0,R8 **ABS** R1,R2 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **ABSs** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				0	0	0	0	0	0	M	R				u	0	0	0	0	0	0	0	0	1	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode		A		de	0	0	#	OP2									

Автокод: PROL <OP2> ABS[s] s,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> ABS[s] s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 ABS R1,R2 R8,R15 (AT),R0

LSRL R5,R0,R8 ABSs R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	M	RS/Rs				0	0	0	0	0	0	0	1	0	0	0	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd		L	0	0	1	#	OP2										

Автокод: PROL <OP2> ABS[s] s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> ABS[s] s,d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 ABS R1,R2 R8.L,R16.L

LSRL.eq R5,R0,R8 ABSs R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				0	0	0	0	0	0	M	RS				0	0	0	0	0	0	0	0	0	1	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD		0	cc	1	0	#	OP2										

Автокод: PROL <OP2 cc> ABS[s] s,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> ABS[s] s,d <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 ABS R1,R2 CCR,R17

LSRL R5, R0, R8 **ABSs** R1, R2 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d						0	0	0	0	0	M	Rs/Rd					0	0	0	0	0	0	1	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

АВТОКОД: PROL <OP2> ABS[s] s,d <R3àRC>

**ABSL**      Абсолютное значение (*long*)      **ABSL**

**Операция:**      `long S,D;      |S|    à    D`

**Описание:** В формате `long` вычисляется абсолютное значение операнда источника `S` и помещается в операнд-приемник `D`.

**Тип:**      `OP1`

**Форматы:**      `1, 4, 5, 6, 7, 8`

**Признаки результата:**

U	N	Z	V	C
√	-	√	√	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ABSL**[.cc] S,D

Примеры: **ABSL.ne** R0,R4

**ABSL**      R0,R4

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	0	1	0	1	1	0	0	0	0		

Автокод:      `ABSL cc, S1, D`

Формат 4: **ABSL** S,D      <XRAM~~3~~àR.L>

Пример:      **ABSL** R0,R2      R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	1	0	0	0	0			

Автокод:      `ABSL4 S, D, MLR,ea,A,R`

`ABSL4 S, D, MLM,ea,A,R`

Формат 5: **ABSL** S,D <R/R.L**а**R/R.L>

Пример: **ABSL** R4,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	0	0	0	0				

Автокод: **ABSL5** S, D, MR,rs,rd

**ABSL5** S, D, MRL,RS,RD

Формат 6: **ABSL** S,D <R**а**RC>

Пример: **ABSL** R8,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	0	0	0	0				

Автокод: **ABSL6** S, D ,MRC,rs,RC

**ABSL6** S, D ,MRCd,RC,rd

Формат 7: **ABSL[.cc]** S,D <#16/32**а**RC/R/R.L>

Пример: **ABSL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	0	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ABSL7** S, D, MIL,cc,#32,RD

**ABSL7** S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **ABSL[s]** S,D <XRAM**а**R.L> <YRAM**а**R0>

Примеры: LSRL R5,R0,R8 **ABSL** R0,R2 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **ABSLs** R0,R2 R8,(A0)+ (AT),R0

Код инструкции:



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	0	0	1	1	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode		A		de	0	0	#	OP2										

Автокод: PROL <OP2> ABSL[s] S,D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> ABSL[s] S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **ABSL** R0,R2 R8,R15 (AT),R0

LSRL R5,R0,R8 **ABSLs** R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS/Rs				0	0	0	0	0	0	1	1	0	0	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd		L	0	0	1	#	OP2											

Автокод: PROL <OP2> ABSL[s] S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> ABSL[s] S,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 **ABSL** R4,R2 R8.L,R16.L

LSRL.eq R5,R0,R8 **ABSLs** R4,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS				0	0	0	0	0	0	0	1	1	0	0	0	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				Cc	RD		0	cc	1	0	#	OP2											

Автокод: PROL <OP2 cc> ABSL[s] S,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> ABSL[s] S,D <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 **ABSL** R0,R2 CCR,R17

LSRL R5,R0,R8 **ABSLs** R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D						0	0	0	0	0	0	Rs/Rd						0	0	0	0	0	1	1	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ABSL[s] S,D <R3àRC>

**ADC**
**Сложение с переносом (short)**
**ADC**
**Операция:**
3-адресный вариант: `short s1,s2,d;`      $s1 + s2 + C \rightarrow d$ 
2-адресный вариант: `short s,d;`      $s + d + C \rightarrow d$ 

**Описание:** В формате `short` вычисляется сумма операндов-источников ( $s1,s2$  - в 3-адресном варианте;  $s,d$  - в 2-адресном варианте) и переноса  $C$ ; результат помещается в операнд-приемник  $d$ . В качестве первого операнда-источника ( $s1$  - в 3-адресном варианте;  $s$  - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:**                                    **OP1**
**Форматы:**                             1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**
Формат 1: **ADC [.cc] s1,s2,d**
Примеры:     **ADC.ne R1,R5,R12**

        **ADC**     R1,R5,R12
 
Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc			0	0	0	1	0	0	0	0	1	0	1				

 Автокод:     **ADC cc, s1, s2, d**
Формат 2: **ADC [.cc] #16,s2,d**
Пример:     **ADC.ne 15,R5,R12**
Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	1	0	0	0	0	0	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: ADC2 cc, #16, s2, d

Формат 3: **ADC** #16, d

Пример: **ADC** 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16												0	0	1	1	0	0	0	0	1	0	1					

Автокод: ADC3 #16, d

Формат 4: **ADC** s, d <XRAM $\beta$ aR.L>

Пример: **ADC** R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode				u	0	1	0	0	0	0	0	0	1	0	1

Автокод: ADC4 s, d, MLR, ea, A, R

ADC4 s, d, MLM, ea, A, R

Формат 5: **ADC** s, d <R/R.L $\beta$ aR/R.L>

Пример: **ADC** R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	0	0	1	0	1		

Автокод: ADC5 s, d, MR, rs, rd

ADC5 s, d, MRL, RS, RD

Формат 6: **ADC** s, d <RβàRC>

Пример: **ADC** R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	0	0	0	1	0	1			

Автокод: **ADC6** s, d, MRC, rs, RC

**ADC6** s, d, MRCd, RC, rd

Формат 7: **ADC[.cc]** s, d <#16/32àRC/R/R.L>

Пример: **ADC** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ADC7** s, d, MIL, cc, #32, RD

**ADC7** s, d, MIC, cc, #16, RC

Формат 8a: <OP2> **ADC[s]** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Примеры: **LSRL** R5, R0, R8 **ADC** R1, R2, R5 R8, (A0)+ (AT), R0

**LSRL** R5, R0, R8 **ADCs** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	0	0	0	0	0	0	1	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: **PROL** <OP2> **ADC[s]** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **ADC[s]** s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **LSRL** R5, R0, R8 **ADC** R1, R2, R5 R8, R15 (AT), R0

LSRL R5,R0,R8 **ADCs** R1,R2,R5 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs				0	0	0	0	0	0	0	0	0	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> ADC[s] s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> **ADC[s]** s1,s2,d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 **ADC** R1,R2,R5 R8.L,R16.L

LSRL.eq R5,R0,R8 **ADCs** R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS				0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2								

Автокод: PROL <OP2 сс> ADC[s] s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2>**ADC[s]** s1,s2,d <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 **ADC** R1,R2,R5 CCR,R17

LSRL R5,R0,R8 **ADCs** R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	Rs/Rd				0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: PROL <OP2> ADC[s] s1,s2,d <R $\beta$ àRC>

**ADCL**
**Сложение с переносом (long)**
**ADCL**
**Операция:**

3-адресный вариант: long S1, S2, D;      S1 + S2 + C → D

2-адресный вариант: long S, D;      S + D + C → D

**Описание:** В формате long вычисляется сумма операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте) и переноса C; результат помещается в операнд-приемник D. В качестве первого операнда-источника S1 в 3-адресном варианте может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ADCL**[.cc] S1, S2, D

Примеры: **ADCL**.ne R2, R6, R12

**ADCL** R2, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	0	0	1	0	1				

*Автокод:* ADCL cc, S1, S2, D

Формат 2: **ADCL**[.cc] #32, S2, D

Пример: **ADCL**.ne 15, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1	D	S2	0	0	cc	0	0	1	0	0	1	0	0	1	0	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: ADCL2 cc, #32, S2, D

Формат 4: **ADCL** S,D <XRAM $\beta$ àR.L>

Пример: **ADCL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	0	0	1	0	1		

Автокод: ADCL4 S, D, MLR,ea,A,R

ADCL4 S, D, MLM,ea,A,R

Формат 5: **ADCL** S,D <R/R.L $\beta$ àR/R.L>

Пример: **ADCL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	0	1	0	1				

Автокод: ADCL5 S, D, MR,rs,rd

ADCL5 S, D, MRL,RS,RD

Формат 6: **ADCL** S,D <R $\beta$ àRC>

Пример: **ADCL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S	D	R	sc	RC	1	0	de	0	0	1	0	0	1	0	0	1	0	1	0	1											

Автокод: ADCL6 S, D, MRC,rs,RC

ADCL6 S, D, MRCd,RC,rd

Формат 7: **ADCL[.cc]** S,D <#16/32 $\beta$ àRC/R/R.L>



Пример: **ADCL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ADCL**7 S, D, MIL,cc,#32,RD

**ADCL**7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2>**ADCL**[s] S1,S2,D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **ADCL** R2,R4,R6 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **ADCLs** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	1	0	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2								

Автокод: PROL <OP2> **ADCL**[s] S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2>**ADCL**[s] S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **ADCL** R2,R4,R6 R8,R15 (AT),R0

LSRL R5,R0,R8 **ADCLs** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	1	0	0	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> **ADCL**[s] S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc>                    **ADCL** S1,S2,D    <R.L**B**àR.L>

*Примеры:*    LSRL.eq R5,R0,R8            **ADCL**    R2,R4,R6            R8.L,R16.L

                  LSRL.eq R5,R0,R8            **ADCLs** R2,R4,R6            R8.L,R16.L

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	0	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD		0	cc		1	0	#	OP2									

*Автокод:*    PROL            <OP2 cc>            ADCL[s] S1,S2,D            <R.L**B**àR.L>

Формат 8d: <OP2>                    **ADCL** S1,S2,D    <R**B**àRC>

*Примеры:*    LSRL R5,R0,R8            **ADCL**    R2,R4,R6            CCR,R17

                  LSRL R5,R0,R8            **ADCLs** R2,R4,R6            CCR,R17

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	0	0	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC		sc	de	1	1	#	OP2										

*Автокод:*    PROL            <OP2>            ADCL[s] S1,S2,D            <R**B**àRC>

**ADC16L**
**Сложение смешанное**
**ADC16L**
**Операция:**

3-адресный вариант: short s1; long S2,D;

if (V=0) {16{s1[15]},s1[15:0]} + S2 **à** D

else {16{C}, s1[15:0]} + S2 **à** D

2-адресный вариант: short s; long D;

if (V=0) {16{s[15]},s[15:0]} + D **à** D

else {16{C}, s[15:0]} + D **à** D

**Описание:** Вычисляется сумма 16-разрядного операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) и 32-разрядного операнда-источника (S2 - в 3-адресном варианте; D - в 2-адресном варианте) и переноса C в зависимости от значения признака V; результат помещается в операнд-приемник D. В качестве первого операнда-источника s1 в 3-адресном варианте может использоваться непосредственный операнд #16.

Если V=0, то старшие разряды 16-разрядного операнда дополняются до 32-х разрядов значением 15-го разряда этого операнда,

Если V=1, то дополнение до 32-х разрядов производится значением бита переноса C.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ADC16L[.cc] s1,S2,D**

Примеры: **ADC16L.ne** R2,R6,R12

**ADC16L** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				S2				0	0	cc				0	0	0	1	0	0	0	1	0	0	0			

Автокод: ADC16L cc, s1, S2, D

Формат 2: **ADC16L[.cc]** #32,S2,D

Пример: **ADC16L.ne** 15,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				S2				0	0	cc				0	0	1	0	0	0	0	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: ADC16L2 cc, #32, S2, D

Формат 4: **ADC16L** s,D <XRAMβàR.L>

Пример: **ADC16L** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				A				mode		u	0	1	de	0	0	0	0	0	1	0	0	0	

Автокод: ADC16L4 s, D, MLR,ea,A,R

ADC16L4 s, D, MLM,ea,A,R

Формат 5: **ADC16L** s,D <R/R.LβàR/R.L>

Пример: **ADC16L** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	1	0	0	0			

Автокод: ADC16L5 s, D, MR,rs,rd

ADC16L5 s, D, MRL,RS,RD

Формат 6: **ADC16L** s,D <R~~В~~RC>

Пример: **ADC16L** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	0	0	0	0	0	1	0	0	0			

Автокод: ADC16L6 s, D, MRC,rs,RC

ADC16L6 s, D, MRCd,RC,rd

Формат 7: **ADC16L[.cc]** s,D<#16/32~~а~~RC/R/R.L>

Пример: **ADC16L** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	1	0	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ADC16L7 s, D, MIL,cc,#32,RD

ADC16L7 s, D, MIC,cc,#16,RC

Формат 8а: <OP2>**ADC16L[s]** s1,S2,D <XRAM~~В~~R.L> <YRAM~~а~~R0>

Примеры: LSRL R5,R0,R8 **ADC16L** R2,R4,R6 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **ADC16Ls** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				S2				M	R				u	0	0	0	0	0	0	0	0	0	0	1	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3/s3/#5	D2/d2	S4/s4	AT	mode	A	de	0	0	#	OP2
----------	-------	-------	----	------	---	----	---	---	---	-----

Автокод: PROL <OP2> ADC16L[s] s1,S2,D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **ADC16L**[s] s1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **ADC16L** R2,R4,R6 R8,R15 (AT),R0  
 LSRL R5,R0,R8 **ADC16Ls** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				S2				M	RS/Rs				0	0	0	0	0	0	0	0	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> ADC16L[s] s1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **ADC16L** s1,S2,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 **ADC16L** R2,R4,R6 R8.L,R16.L  
 LSRL.eq R5,R0,R8 **ADC16Ls** R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				S2				M	RS				0	0	0	0	0	0	0	0	0	0	1	0	0	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> ADC16L[s] s1,S2,D <R.L $\beta$ àR.L>

Формат 8d: <OP2>**ADC16L** s1,S2,D <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 **ADC16L** R2,R4,R6 CCR,R17  
 LSRL R5,R0,R8 **ADC16Ls** R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				S2				M	Rs/Rd				0	0	0	0	0	0	0	0	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ADC16L[s] s1,S2,D <R**β**RC>

**ADD**
*Сложение (short)*
**ADD**
**Операция:**

3-адресный вариант: short s1,s2,d;     s1 + s2 → d

2-адресный вариант: short s,d;     s + d → d

**Описание:** В формате short вычисляется сумма операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ADD[.cc] s1,s2,d**

Примеры: **ADD.ne R1,R5,R12**

**ADD R1,R5,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc			0	0	0	1	0	0	0	0	1	0	0				

Автокод: **ADD cc, s1, s2, d**

Формат 2: **ADD[.cc] #16,s2,d**

Пример: **ADD.ne 15,R5,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---



s1	d	s2	0	0	cc	0	0	1	0	0	0	0	0	0	1	0	0
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: ADD2 cc, #16, s2, d

Формат 3: **ADD** #16, d

Пример: **ADD** 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	0	0	0	0	1	0	0						

Автокод: ADD3 #16, d

Формат 4: **ADD** s, d <XRAM $\beta$ aR.L>

Пример: **ADD** R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	0	0	0	0	1	0	0	

Автокод: ADD4 s, d, MLR, ea, A, R

ADD4 s, d, MLM, ea, A, R

Формат 5: **ADD** s, d <R/R.L $\beta$ aR/R.L>

Пример: **ADD** R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	0	0	1	0	0		

Автокод: ADD5 s, d, MR, rs, rd

ADD5 s, d, MRL, RS, RD

Формат 6: **ADD** s, d <R $\beta$ aRC>

Пример: **ADD** R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	0	0	0	1	0	0			

Автокод: ADD6 s, d, MRC, rs, RC

ADD6 s, d, MRCd, RC, rd

Формат 7: **ADD[.cc]** s, d <#16/32àRC/R/R.L>

Пример: **ADD** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	0	1	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ADD7 s, d, MIL, cc, #32, RD

ADD7 s, d, MIC, cc, #16, RC

Формат 8a: <OP2>**ADD[s]** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **ADD** R1, R2, R5 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 **ADDs** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	0	0	0	0	0	0	1	0	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> ADD[s] s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2>**ADD[s]** s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **ADD** R1, R2, R5 R8, R15 (AT), R0

LSRL R6, R0, R8 **ADDs** R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs				0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> ADD[s] s1,s2,d <R/R.L $\bar{\beta}$ R/R.L> <YRAM $\bar{\alpha}$ R0>

Формат 8с: <OP2.сс>                    **ADD[s]** s1,s2,d <R.L $\bar{\beta}$ R.L>

Примеры: LSRL.eq R6,R0,R8            **ADD** R1,R2,R5                    R8.L,R16.L

LSRL.eq R6,R0,R8            **ADDs** R1,R2,R5                    R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1				d				s2				M	RS				0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL            <OP2 сс>            **ADD[s]** s1,s2,d                    <R.L $\bar{\beta}$ R.L>

Формат 8d: <OP2>                    **ADD[s]** s1,s2,d <R $\bar{\beta}$ RC>

Примеры: LSRL R6,R0,R8            **ADD** R1,R2,R5                    CCR,R17

LSRL R6,R0,R8            **ADDs** R1,R2,R5                    CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1				d				s2				M	Rs/Rd				0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL            <OP2>                    **ADD[s]** s1,s2,d                    <R $\bar{\beta}$ RC>

**ADDL**

 Сложение (*long*)

**ADDL**
**Операция:**

3-адресный вариант:  $\text{long } S1, S2, D; \quad S1 + S2 \rightarrow D$

2-адресный вариант:  $\text{long } S, D; \quad S + D \rightarrow D$

**Описание:** В формате *long* вычисляется сумма операндов-источников ( $S1, S2$  - в 3-адресном варианте;  $S, D$  - в 2-адресном варианте); результат помещается в операнд-приемник  $D$ . В качестве первого операнда-источника  $S1$  в 3-адресном варианте может использоваться непосредственный операнд #32 с расширением знака до 32-х разрядов.

Особенностью данной операции является то, что в параллельной инструкции (формат 8) она может использоваться как операция типа  $OP1$  и/или  $OP2$ , в остальных форматах – только как  $OP1$ . Мнемоническое включение режима “Scaling” (**ADDLs**) возможно только при исполнении операции типа  $OP1$ .

**Тип:**  $OP1, OP2$

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1 (OP1): **ADDL**[.cc]  $S1, S2, D$

Примеры: **ADDL.ne**  $R2, R6, R12$

**ADDL**  $R2, R6, R12$

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	0	0	1	0	0				

*Автокод:* **ADDL** cc,  $S1, S2, D$

Формат 2(ОП1): **ADDL[.cc] #32,S2,D**

Пример: **ADDL.ne 15,R6,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	0	1	0	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: **ADDL2 cc, #32, S2, D**

Формат 4(ОП1): **ADDL S,D <XRAMβàR.L>**

Пример: **ADDL R2,R4 R8,(A0)+**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	0	0	1	0	0		

Автокод: **ADDL4 S, D, MLR,ea,A,R**

**ADDL4 S, D, MLM,ea,A,R**

Формат 5(ОП1): **ADDL S,D <R/R.LβàR/R.L>**

Пример: **ADDL R2,R4 R8.L,R12.L**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	0	1	0	0				

Автокод: **ADDL5 S, D, MR,rs,rd**

**ADDL5 S, D, MRL,RS,RD**

Формат 6(ОП1): **ADDL S,D <RβàRC>**

Пример: **ADDL R2,R6 R7,CCR**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	0	0	1	0	0				

Автокод: **ADDL6 S, D, MRC,rs,RC**

ADDL6 S, D, MRCd, RC, rd

Формат 7 (OP1): **ADDL[.cc]** S, D <#16/32àRC/R/R.L>

Пример: **ADDL** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ADDL7** S, D, MLC, cc, #32, RD

**ADDL7** S, D, MIC, cc, #16, RC

Формат 8a (OP1): <OP2> **ADDL[s]** S1, S2, D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **ADDL** R2, R4, R6 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 **ADDLs** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	0	0	1	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode			A	de	0	0	#	OP2							

Автокод: PROL <OP2> **ADDL[s]** S1, S2, D <XRAMβàR.L> <YRAMàR0>

Формат 8a (OP2): **ADDL** S3, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Пример: **ADDL** R2, R4, R6 OR R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode			A	de	0	0	1	1	1	1	1	1	1	0				

Автокод: PROL ADDL S3,S4,D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b(OP1): <OP2> ADDL[s] S1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 ADDL R2,R4,R6 R8,R15 (AT),R0

LSRL R6,R0,R8 ADDLs R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	0	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> ADDL[s] S1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8b(OP2): ADDL S3,S4,D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: ADDL R2,R4,R6 ORL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	RD/Rd				L	0	0	1	1	1	1	1	1	1	0				

Автокод: PROL ADDL S3,S4,D2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c(OP1): <OP2.cc> ADDL S1,S2,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6,R0,R8 ADDL R2,R4,R6 R8.L,R16.L

LSRL.eq R6,R0,R8 ADDLs R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	0	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3/s3/#5	D2/d2	S4/s4	cc	RD	0	cc	1	0	#	OP2
----------	-------	-------	----	----	---	----	---	---	---	-----

Автокод: PROL <OP2 cc> ADDL[s] S1,S2,D <R.LβàR.L>

Формат 8c(OP2): **ADDL**[.cc] S3,S4,D2 <OP1> <R.LβàR.L>

Пример: **ADDL**.eq R6,R0,R8 ORL R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc	RD				0	cc	1	0	1	1	1	1	1	1	0				

Автокод: PROL ADDL cc S3,S4,D2 <OP1> <R.LβàR.L>

Формат 8d(OP1): <OP2> **ADDL** S1,S2,D <RβàRC>

Примеры: L SRL R6,R0,R8 **ADDL** R2,R4,R6 CCR,R17

L SRL R6,R0,R8 **ADDLs** R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	0	0	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ADDL[s] S1,S2,D <RβàRC>

Формат 8d(OP2): **ADDL** S3,S4,D2 <OP1> <RβàRC>

Пример: **ADDL** R2,R4,R6 ORL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	Rs/Rd				0	0	0	0	OP1										



63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	1	1	1	1	1	1	1	0		

Автокод: PROL

ADDL S3,S4,D2

<OP1>

<R**B**àRC>

### ADDLR Сложение (long) с округлением ADDLR

#### Операция:

3-адресный вариант: long S1, S2, D;  $(S1 + S2)_{\text{Round}} \rightarrow D$

2-адресный вариант: long S, D;  $(S + D)_{\text{Round}} \rightarrow D$

**Описание:** В формате long вычисляется сумма операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); 32-х разрядное значение результата округляется и помещается в операнд-приемник D. В качестве первого операнда-источника S1 в 3-адресном варианте может использоваться непосредственный операнд #32.

Округление производится в соответствии с режимом округления, который устанавливается в 9-м разряде (бит RND) регистра CCR.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

#### Признаки результата:

U	N	Z	V	C
√	√	√	√	√

#### Синтаксис ассемблера и код инструкции:

Формат 1: **ADDLR[.cc]** S1, S2, D

Примеры: **ADDLR.ne** R2, R6, R12

**ADDLR** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	0	1	0	0	0	1			

Автокод: ADDLR cc, S1, S2, D

Формат 2: **ADDLR[.cc]** #32, S2, D

Пример: **ADDLR.ne** 15, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	0	1	0	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: ADDLR2 cc, #32, S2, D

Формат 4: **ADDLR** S,D <XRAM $\beta$ aR.L>

Пример: **ADDLR** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	0	1	0	0	1		

Автокод: ADDLR4 S, D, MLR,ea,A,R

ADDLR4 S, D, MLM,ea,A,R

Формат 5: **ADDLR** S,D <R/R.L $\beta$ aR/R.L>

Пример: **ADDLR** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	1	0	0	1				

Автокод: ADDLR5 S, D, MR,rs,rd

ADDLR5 S, D, MRL,RS,RD

Формат 6: **ADDLR** S,D <R $\beta$ aRC>

Пример: **ADDLR** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	0	1	0	0	1				

Автокод: ADDLR6 S, D, MRC,rs,RC

ADDLR6 S, D, MRCd,RC,rd

Формат 7: **ADDLR**[.cc] S,D <#16/32àRC/R/R.L>

Пример: **ADDLR** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc			1	1	0	0	0	1	0	1	0	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ADDLR7** S, D, MIL,cc,#32,RD

**ADDLR7** S, D, MIC,cc,#16,RC

Формат 8а: <OP2>**ADDLR**[s] S1,S2,D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 **ADDLR** R2,R4,R6 R8,(A0)+ (AT),R0

LSRL R6,R0,R8 **ADDLRs** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R			u	0	0	0	0	0	0	1	0	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2										

Автокод: PROL <OP2> **ADDLR**[s] S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **ADDLR**[s] S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 **ADDLR** R2,R4,R6 R8,R15 (AT),R0

LSRL R6,R0,R8 **ADDLRs** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs			0	0	0	0	0	1	0	1	0	0	1					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd					L	0	0	1	#	OP2				

Автокод: PROL <OP2> ADDLR[s] S1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс>                    **ADDLR** S1,S2,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6,R0,R8            **ADDLR** R2,R4,R6 R8.L,R16.L

LSRL.eq R6,R0,R8            **ADDLRs** R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S1					D					S2					M	RS					0	0	0	0	0	0	1	0	1	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc		RD					0	cc	1	0	#	OP2				

Автокод: PROL <OP2 сс> ADDLR[s] S1,S2,D <R.L $\beta$ àR.L>

Формат 8d: <OP2>                    **ADDLR** S1,S2,D <R $\beta$ àRC>

Примеры: LSRL R6,R0,R8            **ADDLR** R2,R4,R6 CCR,R17

LSRL R6,R0,R8            **ADDLRs** R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1					D					S2					M	Rs/Rd					0	0	0	0	0	1	0	1	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> ADDLR[s] S1,S2,D <R $\beta$ àRC>

## **ADDLRTR** Сложение (*long*) с округлением и преобразованием формата (*в short*) **ADDLRTR**

### Операция:

3-адресный вариант:  $\text{long } S1, S2, d; (S1 + S2)_{\text{Round}} \rightarrow d[15:0]$

2-адресный вариант:  $\text{long } S, D; (S + D)_{\text{Round}} \rightarrow D[15:0]$

**Описание:** В формате *long* вычисляется сумма операндов-источников (*S1, S2* - в 3-адресном варианте; *S, D* - в 2-адресном варианте); 32-разрядное значение результата округляется до 16-разрядного значения и помещается в операнд-приемник *D/d*.

Округление производится в соответствии с режимом округления, установленным битом  $\text{RND}=\text{CCR}[9]$ .

В форматах 1, 8 в качестве регистра приемника используется 16-разрядный регистр.

В форматах 4, 5, 6, 7 – 16-разрядный результат помещается в младшие 16 разрядов регистра приемника, старшие 16 разрядов регистра приемника не изменяются.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

### Признаки результата:

U	N	Z	V	C
√	√	√	√	√

### Синтаксис ассемблера и код инструкции:

Формат 1: **ADDLRTR**[.cc] *S1, S2, d*

Примеры: **ADDLRTR.ne** *R2, R6, R12*

**ADDLRTR** *R2, R6, R12*

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				0	0	cc			0	0	0	1	0	1	0	1	0	1	0				

*Автокод:* **ADDLRTR** *cc, S1, S2, d*

Формат 4: **ADDLRTR** S,D<XRAM $\beta$ àR.L>

Пример: **ADDLRTR** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	0	1	0	1	0	1	0	

Автокод: **ADDLRTR4** S, D, MLR,ea,A,R

**ADDLRTR4** S, D, MLM,ea,A,R

Формат 5: **ADDLRTR** S,D<R/R.L $\beta$ àR/R.L>

Пример: **ADDLRTR** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	1	0	1	0				

Автокод: **ADDLRTR5** S, D, MR,rs,rd

**ADDLRTR5** S, D, MRL,RS,RD

Формат 6: **ADDLRTR** S,D<R $\beta$ àRC>

Пример: **ADDLRTR** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	0	1	0	1	0				

Автокод: **ADDLRTR6** S, D, MRC,rs,RC

**ADDLRTR6** S, D, MRCd,RC,rd

Формат 7: **ADDLRTR**[.cc] S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **ADDLRTR** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ADDLRTR7 S, D, MIL, cc, #32, RD

ADDLRTR7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **ADDLRTR[s]** S1, S2, d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **ADDLRTR** R2, R4, R6 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 **ADDLRTRs** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				M	R				u	0	0	0	0	0	0	1	0	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> ADDLRTR[s] S1, S2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **ADDLRTR[s]** S1, S2, d <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **ADDLRTR** R2, R4, R6 R8, R15 (AT), R0

LSRL R6, R0, R8 **ADDLRTRs** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				M	RS/Rs				0	0	0	0	0	0	1	0	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> ADDLRTR[s] S1, S2, d <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **ADDLRTR** S1, S2, d <R.LβàR.L>

Примеры: LSRL.eq R6, R0, R8 **ADDLRTR** R2, R4, R6 R8.L, R16.L

LSRL.eq R6, R0, R8 **ADDLRTRs** R2, R4, R6 R8.L, R16.L

Код инструкции:



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				M	RS				0	0	0	0	0	0	1	0	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> ADDLRTR[s] S1,S2,d <R.L~~β~~R.L>

Формат 8d: <OP2> **ADDLRTR** S1,S2,d <R~~β~~RC>

Примеры: LSRL R6,R0,R8 **ADDLRTR** R2,R4,R6 CCR,R17

LSRL R6,R0,R8 **ADDLRTRs** R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				M	Rs/Rd				0	0	0	0	0	1	0	1	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ADDLRTR[s] S1,S2,d <R~~β~~RC>

**ADDSUB      Сложение-вычитание (short)      ADDSUB**
**Операция:**

3-адресный вариант: short s1, s2, d;

s1 + s2 → d

s2 - s1 → s2

2-адресный вариант: short s, d;

s + d → s

d - s → d

**Описание:** В формате short вычисляется сумма и разность операндов-источников (s2, s1 – в 3-адресном варианте; d, s – в 2-адресном варианте). Результат сложения помещается по адресу d в 3-адресном варианте, по адресу s в 2-адресном варианте. Результат вычитания помещается по адресу s2 в 3-адресном варианте, по адресу d в 2-адресном варианте.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 8c, 8d

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
U+&U-	N+	Z+	V+ V-	C+

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ADDSUB** [ .cc ] s1, s2, d

Примеры: **ADDSUB**.ne R1, R5, R12

**ADDSUB** R1, R5, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc			0	0	0	1	0	0	0	0	1	1	1				

Автокод: **ADDSUB**cc, s1, s2, d

Формат 4: **ADDSUB** s,d <XRAM $\beta$ àR.L>

Пример: **ADDSUB** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				R				A				mode				u	0	1	de	0	0	0	0	0	0	1	1	1

Автокод: **ADDSUB4** s, d, MLR,ea,A,R

**ADDSUB4** s, d, MLM,ea,A,R

Формат 5: **ADDSUB** s,d <R/R.L $\beta$ àR/R.L>

Пример: **ADDSUB** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	0	1	1	1			

Автокод: **ADDSUB5** s, d, MR,rs,rd

**ADDSUB5** s, d, MRL,RS,RD

Формат 6: **ADDSUB** s,d <R $\beta$ àRC>

Пример: **ADDSUB** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	0	0	0	0	1	1	1		

Автокод: **ADDSUB6** s, d ,MRC,rs,RC

**ADDSUB6** s, d ,MRCd,RC,rd

Формат 8с: <OP2.cc> **ADDSUB**[s] s1,s2,d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6,R0,R8 **ADDSUB** R1,R2,R5 R8.L,R16.L

LSRL.eq R6,R0,R8 **ADDSUB** R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS				0	0	0	0	0	0	0	0	0	0	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5					D2/d2					S4/s4					cc		RD					0	cc		1	0	#	OP2				

Автокод: PROL <OP2 cc> ADDSUB[s] s1,s2,d <R.LβàR.L>

Формат 8d: <OP2> ADDSUB[s] s1,s2,d <RβàRC>

Примеры: LSRL R6,R0,R8 ADDSUB R1,R2,R5 CCR,R17

LSRL R6,R0,R8 ADDSUB R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1					d					s2					M	Rs/Rd					0	0	0	0	0	0	0	0	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> ADDSUB [s] s1,s2,d <RβàRC>

**ADDSUBL      Сложение-вычитание (long)      ADDSUBL**

**Операция:**

3-адресный вариант: long S1, S2, D;

S1 + S2 → D

S2 - S1 → S2

2-адресный вариант: long S, D;

S + D → S

D - S → D

**Описание:** В формате long вычисляется сумма и разность операндов-источников (S2, S1 – в 3-адресном варианте; D, S – в 2-адресном варианте). Результат сложения помещается по адресу D в 3-адресном варианте, по адресу S в 2-адресном варианте. Результат вычитания помещается по адресу S2 в 3-адресном варианте, по адресу D в 2-адресном варианте.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 8с, 8d

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
U+&U-	N+	Z+	V+ V-	C+

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ADDSUBL**[.cc] S1, S2, D

Примеры: **ADDSUBL.ne** R2, R6, R12

**ADDSUBL** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	0	1	1	0	1	1				

Автокод: **ADDSUBL** cc, S1, S2, D

Формат 4: **ADDSUBL** S,D<XRAM $\beta$ àR.L>

Пример: **ADDSUBL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	0	1	1	0	1	1			

Автокод: **ADDSUBL4** S, D, MLR,ea,A,R

**ADDSUBL4** S, D, MLM,ea,A,R

Формат 5: **ADDSUBL** S,D<R/R.L $\beta$ àR/R.L>

Пример: **ADDSUBL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	0	1	1				

Автокод: **ADDSUBL5** S, D, MR,rs,rd

**ADDSUBL5** S, D, MRL,RS,RD

Формат 6: **ADDSUBL** S,D<R $\beta$ àRC>

Пример: **ADDSUBL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	1	1	0	1	1				

Автокод: **ADDSUBL6** S, D, MRC,rs,RC

**ADDSUBL6** S, D, MRCd,RC,rd

Формат 8с: <OP2.сс> **ADDSUBL[s]** S1,S2,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6,R0,R8 **ADDSUBL** R2,R4,R6 R8.L,R16.L

LSRL.eq R6,R0,R8 **ADDSUBLs** R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	0	0	1	1	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0	cc		1	0	#	OP2							

Автокод: PROL <OP2 cc> ADDSUBL[s] S1,S2,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> ADDSUBL[s] S1,S2,D <R $\beta$ àRC>

Примеры: LSRL R6,R0,R8 ADDSUBL R2,R4,R6 CCR,R17

LSRL R6,R0,R8 ADDSUBLs R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	0	1	1	0	1	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ADDSUBL[s] S1,S2,D <R $\beta$ àRC>

**ADDSUBX      Сложение-вычитание (X16)      ADDSUBX**
**Операция:**

3-адресный вариант: X16 S1, S2, D;

$$S1[31:16] + S2[31:16] \rightarrow D[31:16]$$

$$S1[15:0] + S2[15:0] \rightarrow D[15:0]$$

$$S2[31:16] - S1[31:16] \rightarrow S2[31:16]$$

$$S2[15:0] - S1[15:0] \rightarrow S2[15:0]$$

2-адресный вариант: X16 S, D;

$$S[31:16] + D[31:16] \rightarrow S[31:16]$$

$$S[15:0] + D[15:0] \rightarrow S[15:0]$$

$$D[31:16] - S[31:16] \rightarrow D[31:16]$$

$$D[15:0] - S[15:0] \rightarrow D[15:0]$$

**Описание:** В формате X16 вычисляется сумма и разность операндов-источников (S2, S1 - в 3-адресном варианте; D, S - в 2-адресном варианте), при этом независимо складываются и вычитаются старшие 16 разрядов операндов-источников и младшие 16 разрядов операндов. Результат сложения помещается по адресу D в 3-адресном варианте, по адресу S в 2-адресном варианте. Результат вычитания помещается по адресу S2 в 3-адресном варианте, по адресу D в 2-адресном варианте.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 8c, 8d

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
U+&U-	Nr+	Zr+	V+ V-	Nr-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ADDSUBX**[.cc] S1, S2, D

Примеры: **ADDSUBX**.ne R2, R6, R12



**ADDSUBX** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	0	0	0	0	0	0	0	0	0

Автокод: **ADDSUBX** cc, S1, S2, D

Формат 4: **ADDSUBX** S,D<XRAM $\beta$ àR.L>

Пример: **ADDSUBX** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	0	0	0	0	0	0	0

Автокод: **ADDSUBX4** S, D, MLR,ea,A,R

**ADDSUBX4** S, D, MLM,ea,A,R

Формат 5: **ADDSUBX** S,D<R/R.L $\beta$ àR/R.L>

Пример: **ADDSUBX** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd			1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0

Автокод: **ADDSUBX5** S, D, MR,rs,rd

**ADDSUBX5** S, D, MRL,RS,RD

Формат 6: **ADDSUBX** S,D<R $\beta$ àRC>

Пример: **ADDSUBX** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC			1	0	de	0	0	1	0	0	0	0	0	0	0	0	0	0

Автокод: **ADDSUBX6** S, D, MRC,rs,RC

**ADDSUBX6** S, D, MRCd,RC,rd

Формат 8с: <OP2.cc> **ADDSUBX[s]** S1,S2,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6,R0,R8 **ADDSUBX** R2,R4,R6 R8.L,R16.L

LSRL.eq R6,R0,R8      **ADDSUBXs** R2,R4,R6      R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL      <OP2 cc>      ADDSUBX[s] S1,S2,D      <R.L $\beta$ àR.L>

Формат 8d: <OP2>      **ADDSUBX[s]** S1,S2,D      <R $\beta$ àRC>

Примеры: LSRL R6,R0,R8      **ADDSUBX** R2,R4,R6      CCR,R17

LSRL R6,R0,R8      **ADDSUBXs** R2,R4,R6      CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL      <OP2>      ADDSUBX[s] S1,S2,D      <R $\beta$ àRC>

**ADDX**
*Сложение комплексное (X16)*
**ADDX**
**Операция:**
3-адресный вариант: X16 S1, S2, D;

 $S1[31:16] + S2[31:16] \rightarrow D[31:16]$ 
 $S1[15:0] + S2[15:0] \rightarrow D[15:0]$ 
2-адресный вариант: X16 S, D;

 $S[31:16] + D[31:16] \rightarrow D[31:16]$ 
 $S[15:0] + D[15:0] \rightarrow D[15:0]$ 

**Описание:** В формате X16 вычисляется сумма операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте), при этом независимо складываются старшие 16 разрядов и младшие 16 разрядов операндов-источников; результат сложения старших 16 разрядов операндов помещается в старшие 16 разрядов операнда-приемника D, а результат сложения младших 16 разрядов помещается в младшие 16 разрядов операнда-приемника D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
Ur&Ui	Nr	Zr	Vr Vi	Cr

**Синтаксис ассемблера и код инструкции:**
Формат 1: **ADDX[.cc]** S1, S2, D

 Пример: **ADDX.ne** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	0	0	1	1	1				

 Автокод: **ADDX cc, S1, S2, D**

Формат 4: **ADDX** S, D <XRAM $\beta$ àR.L>

Пример: **ADDX** R2, R4 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	0	0	1	0	0	1	1	1

Автокод: **ADDX4** S, D, MLR, ea, A, R

**ADDX4** S, D, MLM, ea, A, R

Формат 5: **ADDX** S, D <R/R.L $\beta$ àR/R.L>

Пример: **ADDX** R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	0	1	1	1				

Автокод: **ADDX5** S, D, MR, rs, rd

**ADDX5** S, D, MRL, RS, RD

Формат 6: **ADDX** S, D <R $\beta$ àRC>

Пример: **ADDX** R2, R6 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	0	0	1	0	0	1	1	1	

Автокод: **ADDX6** S, D, MRC, rs, RC

**ADDX6** S, D, MRCd, RC, rd

Формат 7: **ADDX[.cc]** S, D <#16/32 $\beta$ àRC/R/R.L>

Пример: **ADDX** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	0	1	0	0	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ADDX7 S, D, MIL, cc, #32, RD

ADDX7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2>ADDX[s] S1, S2, D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 ADDX R2, R4, R6 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 ADDXs R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	0	0	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> ADDX[s] S1, S2, D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> ADDX[s] S1, S2, D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 ADDX R2, R4, R6 R8, R15 (AT), R0

LSRL R6, R0, R8 ADDXs R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	0	0	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> ADDX[s] S1, S2, D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> ADDX S1, S2, D <R.LβàR.L>

Примеры: LSRL.eq R6, R0, R8 ADDX R2, R4, R6 R8.L, R16.L

LSRL.eq R6, R0, R8 ADDXs R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	0	1	0	0	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0	cc		1	0	#	OP2							

Автокод: PROL <OP2 cc> ADDX[s] S1, S2, D <R.L~~В~~ÀR.L>

Формат 8d: <OP2> **ADDX** S1, S2, D <R~~В~~ÀRC>

Примеры: LSRL R6, R0, R8 **ADDX** R2, R4, R6 CCR, R17

LSRL R6, R0, R8 **ADDXs** R2, R4, R6 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	0	1	0	0	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ADDX[s] S1, S2, D <R~~В~~ÀRC>

**AD1**      **Сложение и инкремент (short)**      **AD1****Операция:**

3-адресный вариант: short s1,s2,d;      s1 + s2 + 1 → d

2-адресный вариант: short s,d;      s + d + 1 → d

**Описание:** В формате short вычисляется сумма операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте) и инкрементируется на 1; результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:**      OP1

**Форматы:**      1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **AD1[.cc]** s1,s2,d

Примеры: **AD1.ne** R1,R5,R12

**AD1**      R1,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	0	1	0	1	0	1	1	1	1			

Автокод:      AD1 cc, s1, s2, d

Формат 2: **AD1[.cc]** #16,s2,d

Пример:      **AD1.ne** 15,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s1	d	s2	0	0	cc	0	0	1	0	0	1	0	1	1	1	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: AD1 2 cc, #16, s2, d

Формат 3: **AD1** #16, d

Пример: **AD1** 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
d											#16											0	0	1	1	0	1	0	1	1	1	1

Автокод: AD1 3 #16, d

Формат 4: **AD1** s, d <XRAMβàR.L>

Пример: **AD1** R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	1	0	1	1	1	1		

Автокод: AD1 4 s, d, MLR, ea, A, R

AD1 4 s, d, MLM, ea, A, R

Формат 5: **AD1** s, d <R/R.LβàR/R.L>

Пример: **AD1** R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	1	1	1	1				

Автокод: AD1 5 s, d, MR, rs, rd

AD1 5 s, d, MRL, RS, RD

Формат 6: **AD1** s, d <RβàRC>

Пример: **AD1** R1, R5 R7, CCR



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	1	0	1	1	1	1	1	1	1	1

Автокод: AD1 6 s, d, MRC, rs, RC

AD1 6 s, d, MRCd, RC, rd

Формат 7: **AD1[.cc]** s, d <#16/32àRC/R/R.L>

Пример: **AD1** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: AD1 7 s, d, MIL, cc, #32, RD

AD1 7 s, d, MIC, cc, #16, RC

Формат 8а: <OP2> **AD1[s]** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **AD1** R1, R2, R5 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 **AD1s** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				s2				M	R				u	0	0	0	0	0	0	1	0	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> AD1[s] s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **AD1[s]** s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **AD1** R1, R2, R5 R8, R15 (AT), R0

LSRL R6, R0, R8 **AD1s** R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1				d				s2				M	RS/Rs				0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> AD1[s] s1,s2,d <R/R.L $\bar{\beta}$ R/R.L> <YRAM $\bar{\alpha}$ R0>

Формат 8с: <OP2.сс> AD1[s] s1,s2,d <R.L $\bar{\beta}$ R.L>

Примеры: LSRL.eq R6,R0,R8 AD1 R1,R2,R5 R8.L,R16.L

LSRL.eq R6,R0,R8 AD1s R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1				d				s2				M	RS				0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 сс> AD1[s] s1,s2,d <R.L $\bar{\beta}$ R.L>

Формат 8d: <OP2> AD1[s] s1,s2,d <R $\bar{\beta}$ RC>

Примеры: LSRL R6,R0,R8 AD1 R1,R2,R5 CCR,R17

LSRL R6,R0,R8 AD1s R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
s1				d				s2				M	Rs/Rd				0	0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> AD1[s] s1,s2,d <R $\bar{\beta}$ RC>

**AND**
*Логическое И (short)*
**AND**
**Операция:**

3-адресный вариант: short s1,s2,d;     s1 & s2 → d

2-адресный вариант: short s,d;     s & d → d

**Описание:** В формате short выполняется побитное логическое умножение операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **AND[.cc]** s1,s2,d

Примеры: **AND.ne** R1,R5,R12

**AND** R1,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	0	1	1	0	0	0	0	0	0	1		

*Автокод:*     **AND cc, s1, s2, d**

Формат 2: **AND[.cc]** #16,s2,d

Пример: **AND.ne** 15,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1				d				s2				0	0	cc				0	0	1	0	1	0	0	0	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: AND2 cc, #16, s2, d

Формат 3: **AND** #16, d

Пример: **AND** 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16												0	0	1	1	1	0	0	0	0	0	0	1				

Автокод: AND3 #16, d

Формат 4: **AND** s, d <XRAM $\beta$ àR.L>

Пример: **AND** R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	0	0	0	0	0	0	0	1

Автокод: AND4 s, d, MLR, ea, A, R

AND4 s, d, MLM, ea, A, R

Формат 5: **AND** s, d <R/R.L $\beta$ àR/R.L>

Пример: **AND** R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	0	0	0	0	0	0	0	1

Автокод: AND5 s, d, MR, rs, rd

AND5 s, d, MRL, RS, RD

Формат 6: **AND** s, d <R $\beta$ àRC>

Пример: **AND** R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	0	0	0	0	0	1		

Автокод: AND6 s, d, MRC, rs, RC

AND6 s, d, MRCd, RC, rd

Формат 7: **AND[.cc]** s, d <#16/32àRC/R/R.L>

Пример: **AND** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	0	0	0	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: AND7 s, d, MIL, cc, #32, RD

AND7 s, d, MIC, cc, #16, RC

Формат 8a: <OP2>**AND** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **AND** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	1	0	0	0	0	0	0	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **AND** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2>**AND** s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **AND** R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				s2				M	RS/Rs				0	0	0	0	1	0	0	0	0	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd					L	0	0	1	#	OP2				

Автокод: PROL <OP2> AND s1,s2,d <R/R.L $\bar{\mathbf{B}}$ R/R.L> <YRAM $\bar{\mathbf{A}}$ R0>

Формат 8с: <OP2.cc> AND s1,s2,d <R.L $\bar{\mathbf{B}}$ R.L>

Пример: LSRL.eq R6,R0,R8 AND R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
s1					d					s2					M	RS					0	0	0	0	0	1	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc		RD					0	cc	1	0	#	OP2				

Автокод: PROL <OP2 cc> AND s1,s2,d <R.L $\bar{\mathbf{B}}$ R.L>

Формат 8d: <OP2> AND s1,s2,d <R $\bar{\mathbf{B}}$ RC>

Пример: LSRL R6,R0,R8 AND R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1					d					s2					M	Rs/Rd					0	0	0	0	1	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> AND s1,s2,d <R $\bar{\mathbf{B}}$ RC>

**ANDC***Логическое И с инверсией (short)***ANDC****Операция:**3-адресный вариант: short s1,s2,d; s1 & ~s2 à d2-адресный вариант: short s,d; s & ~d à d

**Описание:** В формате short выполняется побитное логическое умножение инверсии операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) с вторым операндом (s2 - в 3-адресном варианте; d - в 2-адресном варианте). Результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**Формат 1: **ANDC[.cc]** s1,s2,dПримеры: **ANDC.ne** R1,R5,R12**ANDC** R1,R5,R12Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1					d				s2			0	0	cc			0	0	0	1	1	0	0	0	0	1	0				

Автокод: **ANDC** cc, s1, s2, dФормат 2: **ANDC[.cc]** #16,s2,dПример: **ANDC.ne** 15,R5,R12Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				s2				0	0	cc				0	0	1	0	1	0	0	0	0	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: ANDC2 cc, #16, s2, d

Формат 3: ANDC #16, d

Пример: ANDC 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	1	0	0	0	0	1	0						

Автокод: ANDC3 #16, d

Формат 4: ANDC s, d <XRAM $\beta$ àR.L>

Пример: ANDC R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	0	0	0	0	0	1	0	

Автокод: ANDC4 s, d, MLR, ea, A, R

ANDC4 s, d, MLM, ea, A, R

Формат 5: ANDC s, d <R/R.L $\beta$ àR/R.L>

Пример: ANDC R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	0	0	0	1	0			

Автокод: ANDC5 s, d, MR, rs, rd

ANDC5 s, d, MRL, RS, RD

Формат 6: ANDC s, d <R $\beta$ àRC>

Пример: ANDC R1, R5 R7, CCR



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	0	0	0	0	1	0		

Автокод:  $ANDC6\ s, d, MRC, rs, RC$

$ANDC6\ s, d, MRCd, RC, rd$

Формат 7:  $ANDC[.cc]\ s, d\ \langle\#16/32\ \beta\ \text{RC/R/R.L}\rangle$

Пример:  $ANDC\ R2, R0\ 0x12345678, R16.L$

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	0	0	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод:  $ANDC7\ s, d, MIL, cc, \#32, RD$

$ANDC7\ s, d, MIC, cc, \#16, RC$

Формат 8a:  $\langle OP2 \rangle\ ANDC\ s1, s2, d\ \langle XRAM\ \beta\ \text{R.L} \rangle\ \langle YRAM\ \alpha\ R0 \rangle$

Примеры:  $LSRL\ R6, R0, R8\ ANDC\ R1, R2, R5\ R8, (A0)+\ (AT), R0$

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	1	0	0	0	0	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode					A	de	0	0	#	OP2					

Автокод:  $PROL\ \langle OP2 \rangle\ ANDC\ s1, s2, d\ \langle XRAM\ \beta\ \text{R.L} \rangle\ \langle YRAM\ \alpha\ R0 \rangle$

Формат 8b:  $\langle OP2 \rangle\ ANDC\ s1, s2, d\ \langle R/R.L\ \beta\ \text{R/R.L} \rangle\ \langle YRAM\ \alpha\ R0 \rangle$

Пример:  $LSRL\ R6, R0, R8\ ANDC\ R1, R2, R5\ R8, R15\ (AT), R0$

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs				0	0	0	0	1	0	0	0	0	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd					L	0	0	1	#	OP2				

Автокод: PROL <OP2> ANDC s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> ANDC s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 ANDC R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1					d					s2					M	RS					0	0	0	0	0	1	0	0	0	0	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5					D2/d2					S4/s4					cc		RD					0	cc		1	0	#	OP2				

Автокод: PROL <OP2 cc> ANDC s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> ANDC s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 ANDC R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1					d					s2					M	Rs/Rd					0	0	0	0	1	0	0	0	0	0	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> ANDC s1,s2,d <R $\beta$ àRC>

## ANDCL Логическое И с инверсией (long) ANDCL

### Операция:

3-адресный вариант: long S1, S2, D;      S1 & ~S2 → D

2-адресный вариант: long S, D;      S & ~D → D

**Описание:** В формате long выполняется побитное логическое умножение инверсии операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) с вторым операндом (S2 - в 3-адресном варианте; D - в 2-адресном варианте). Результат помещается в операнд-приемник D. В качестве первого операнда-источника (S1 - в 3-адресном; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

### Признаки результата:

U	N	Z	V	C
√	√	√	-	-

### Синтаксис ассемблера и код инструкции:

Формат 1: **ANDCL**[.cc] S1, S2, D

Примеры: **ANDCL.ne** R2, R6, R12

**ANDCL** R2, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	1	0	1	0	0	1	0				

*Автокод:*      **ANDCL** cc, S1, S2, D

Формат 2: **ANDCL**[.cc] #32, S2, D

Пример: **ANDCL.ne** 15, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	1	0	1	0	1	0	0	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: ANDCL2 cc, #32, S2, D

Формат 4: **ANDCL** S,D <XRAM $\beta$ àR.L>

Пример: **ANDCL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	1	0	1	0	1	0	0	1	0

Автокод: ANDCL4 S, D, MLR,ea,A,R

ANDCL4 S, D, MLM,ea,A,R

Формат 5: **ANDCL** S,D <R/R.L $\beta$ àR/R.L>

Пример: **ANDCL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	0	0	1	0	0	1	0	

Автокод: ANDCL5 S, D, MR,rs,rd

ANDCL5 S, D, MRL,RS,RD

Формат 6: **ANDCL** S,D <R $\beta$ àRC>

Пример: **ANDCL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	0	1	0	0	1	0		

Автокод: ANDCL6 S, D, MRC,rs,RC

ANDCL6 S, D, MRCd,RC,rd

Формат 7: **ANDCL[.cc]** S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **ANDCL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	0	0	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ANDCL7** S, D, MIL,cc,#32,RD

**ANDCL7** S, D, MIC,cc,#16,RC

Формат 8а: <OP2>**ANDCL** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **ANDCL** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	1	0	1	0	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **ANDCL** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **ANDCL** S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Примеч: LSRL R6,R0,R8 **ANDCL** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	1	0	1	0	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> **ANDCL** S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **ANDCL** S1,S2,D <R.LβàR.L>

Пример: LSRL.eq R6,R0,R8 **ANDCL** R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	1	0	1	0	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> ANDCL S1,S2,D <R.L~~B~~R.L>

Формат 8d: <OP2> **ANDCL** S1,S2,D <R~~B~~RC>

Пример: LSRL R6,R0,R8 **ANDCL** R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	1	0	1	0	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ANDCL S1,S2,D <R~~B~~RC>

**ANDI    Инверсия логического И (short)            ANDI**
**Операция:**

3-адресный вариант: short s1,s2,d;       $\sim (s1 \ \& \ s2) \ \grave{\text{a}} \ d$

2-адресный вариант: short s,d;             $\sim (s \ \& \ d) \ \grave{\text{a}} \ d$

**Описание:** В формате short выполняется побитное логическое умножение операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); инверсия результата помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:**    OP1

**Форматы:**    1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ANDI[.cc]** s1,s2,d

*Примеры:*    **ANDI.ne** R1,R5,R12

**ANDI**            R1,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	0	1	1	0	0	0	0	1	1			

*Автокод:*            **ANDI** cc, s1, s2, d

Формат 2: **ANDI[.cc]** #16,s2,d

*Пример:*        **ANDI.ne** 15,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s1	d	s2	0	0	cc	0	0	1	0	1	0	0	0	0	1	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: ANDI2 cc, #16, s2, d

Формат 3: **ANDI** #16, d

Пример: **ANDI** 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
d											#16											0	0	1	1	1	0	0	0	0	1	1

Автокод: ANDI3 #16, d

Формат 4: **ANDI** s, d <XRAM $\beta$ aR.L>

Пример: **ANDI** R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	0	0	0	0	0	1	1	

Автокод: ANDI4 s, d, MLR, ea, A, R

ANDI4 s, d, MLM, ea, A, R

Формат 5: **ANDI** s, d <R/R.L $\beta$ aR/R.L>

Пример: **ANDI** R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	0	0	0	1	1			

Автокод: ANDI5 s, d, MR, rs, rd

ANDI5 s, d, MRL, RS, RD

Формат 6: **ANDI** s, d <R $\beta$ aRC>

Пример: **ANDI** R1, R5 R7, CCR



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	0	0	0	1	1			

Автокод: ANDI6 s, d, MRC, rs, RC

ANDI6 s, d, MRCd, RC, rd

Формат 7: **ANDI**[.cc] s, d <#16/32àRC/R/R.L>

Пример: **ANDI** R2, R0 0x12345678, R16.L

Код инструкции:

3	3	2	2	2	2	2	2	2	2	2	1	1	1	16	15	1	1	1	1	10	9	8	7	6	5	4	3	2	1	0
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7		4	3	2	1											
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	0	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ANDI7 s, d, MIL, cc, #32, RD

ANDI7 s, d, MIC, cc, #16, RC

Формат 8a: <OP2> **ANDI** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **ANDI** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	R				u	0	0	0	0	1	0	0	0	0	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **ANDI** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **ANDI** s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **ANDI** R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s1	d	s2	0	RS/Rs	0	0	0	0	0	1	0	0	0	0	1	1
----	---	----	---	-------	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	RD/Rd					L	0	0	1	#	OP2					

Автокод: PROL <OP2> ANDI s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> **ANDI** s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 **ANDI** R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1					d					s2					0	RS					0	0	0	0	0	1	0	0	0	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc	RD					0	cc	1	0	#	OP2					

Автокод: PROL <OP2 сс> ANDI s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> **ANDI** s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 **ANDI** R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1					d					s2					0	Rs/Rd					0	0	0	0	1	0	0	0	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> ANDI s1,s2,d <R $\beta$ àRC>

**ANDL**
**Логическое И (long)**
**ANDL**
**Операция:**

3-адресный вариант: long S1, S2, D;      S1 & S2 → D

2-адресный вариант: long S, D;      S & D → D

**Описание:** В формате long выполняется побитное логическое умножение операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); результат помещается в операнд-приемник D. В качестве первого операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ANDL[.cc]** S1, S2, D

Примеры: **ANDL.ne** R2, R6, R12

**ANDL** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	1	0	1	0	0	0	1				

Автокод: **ANDL cc, S1, S2, D**

Формат 2: **ANDL[.cc]** #32, S2, D

Пример: **ANDL.ne** 15, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1	D	S2	0	0	cc	0	0	1	0	1	0	1	0	0	0	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: ANDL2 cc, #32, S2, D

Формат 4: **ANDL** S,D <XRAM $\beta$ aR.L>

Пример: **ANDL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	1	0	1	0	0	0	1		

Автокод: ANDL4 S, D, MLR,ea,A,R

ANDL4 S, D, MLM,ea,A,R

Формат 5: **ANDL** S,D <R/R.L $\beta$ aR/R.L>

Пример: **ANDL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	0	0	0	1				

Автокод: ANDL5 S, D, MR,rs,rd

ANDL5 S, D, MRL,RS,RD

Формат 6: **ANDL** S,D <R $\beta$ aRC>

Пример: **ANDL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	0	0	0	1				

Автокод: ANDL6 S, D, MRC,rs,RC

ANDL6 S, D, MRCd,RC,rd

Формат 7: **ANDL**[.cc] S,D <#16/32àRC/R/R.L>

Пример: **ANDL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc			1	1	1	0	1	0	1	0	0	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ANDL7** S, D, MIL,cc,#32,RD

**ANDL7** S, D, MIC,cc,#16,RC

Формат 8а: <OP2>**ANDL** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **ANDL** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R			u	0	0	0	0	1	0	1	0	0	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2										

Автокод: PROL <OP2> **ANDL** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **ANDL** S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **ANDL** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs			0	0	0	0	1	0	1	0	0	0	1					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd			L	0	0	1	#	OP2										

Автокод: PROL <OP2> **ANDL** S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **ANDL** S1,S2,D <R.LβàR.L>

Пример: LSRL.eq R6,R0,R8 **ANDL** R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	1	0	1	0	0	0	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> ANDL S1,S2,D <R. L **β** R.L>

Формат 8d: <OP2> ANDL S1,S2,D <R **β** RC>

Пример: LSRL R6,R0,R8 ANDL R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	1	0	1	0	0	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ANDL S1,S2,D <R **β** RC>

**ASH Сложение и вычитание двух пар чисел (short) ASH**
**Операция:**

3-адресный вариант: long S1, S2, D;

$$S1[31:16] + S1[15:0] \rightarrow D[31:16]$$

$$S2[15:0] - S2[31:16] \rightarrow D[15:0]$$

2-адресный вариант: long S, D;

$$S[31:16] + S[15:0] \rightarrow D[31:16]$$

$$D[15:0] - D[31:16] \rightarrow D[15:0]$$

**Описание:** В формате short вычисляется сумма старшей и младшей части первого операнда-источника (S1– в 3-адресном варианте; S – в 2-адресном варианте), и разность младшей и старшей части второго операнда-источника (S2– в 3-адресном варианте; D – в 2-адресном варианте). Результат сложения помещается в старшую часть операнда-приемника D, результат вычитания помещается в младшую часть операнда-приемника D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
U+&U-	N+	Z+	V+ V-	N-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ASH[.cc] S1, S2, D**

Примеры: **ASH.ne R2, R6, R12**

**ASH R2, R6, R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	0	1	0	1	1	1	1	1	1	0		

Автокод: ASH cc, S1, S2, D

Формат 4: **ASH** S,D <XRAM**В**àR.L>

Пример: **ASH** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	1	1	1	1	1	1	0	

Автокод: ASH4 S, D, MLR,ea,A,R

ASH4 S, D, MLM,ea,A,R

Формат 5: **ASH** S,D <R/R.L**В**àR/R.L>

Пример: **ASH** R2,R4 R8.L,R12.L

Код инструкции:

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0
S			D				RS/Rs				RD/Rd																					

Автокод: ASH5 S, D, MR,rs,rd

ASH5 S, D, MRL,RS,RD

Формат 6: **ASH** S,D <R**В**àRC>

Пример: **ASH** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	1	1	1	1	1	0		

Автокод: ASH6 S, D, MRC,rs,RC

ASH6 S, D, MRCd,RC,rd

Формат 7: **ASH[.cc]** S,D <#16/32**В**àRC/R/R.L>

Пример: **ASH** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---



S	D	RD/Rd/RC	L/sc	sr	cc	1	1	1	0	0	1	1	1	1	1	0
---	---	----------	------	----	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ASH7 S, D, MIL, cc, #32, RD

ASH7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **ASH[s]** S1, S2, D <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **ASH** R2, R4, R6 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 **ASHs** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	1	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2									

Автокод: PROL <OP2> ASH[s] S1, S2, D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **ASH[s]** S1, S2, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **ASH** R2, R4, R6 R8, R15 (AT), R0

LSRL R6, R0, R8 **ASHs** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	1	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> ASH[s] S1, S2, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **ASH[s]** S1, S2, D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6, R0, R8 **ASH** R2, R4, R6 R8.L, R16.L

LSRL.eq R6,R0,R8    **ASHs** R2,R4,R6    R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	1	1	1	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc	RD					0	cc	1	0	#	OP2					

Автокод: PROL    <OP2 cc>    ASH[s] S1,S2,D    <R.L**β**àR.L>

Формат 8d: <OP2>    **ASH[s]** S1,S2,D    <R**β**àRC>

Примеры: LSRL R6,R0,R8    **ASH** R2,R4,R6    CCR,R17

LSRL R6,R0,R8    **ASHs** R2,R4,R6    CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	1	1	1	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL    <OP2>    ASH[s] S1,S2,D    <R**β**àRC>

**ASL                      Арифметический сдвиг влево (short)                      ASL**

**Операция:**

3-адресный вариант: short s1,s2,d; (s2 << s1[4:0]) **à** d

2-адресный вариант: short s,d; (d << s[4:0]) **à** d

**Описание:** Операнд-источник s2 (в 3-адресном варианте) или d (в 2-адресном варианте) сдвигается влево на количество разрядов, заданное младшими пятью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте) и результат помещается в операнд-приемник d. Старшие одиннадцать разрядов s1 (или s) при этом должны быть установлены в «0». Вытолкнутые за пределы разрядной сетки разряды теряются, кроме последнего вытолкнутого, который запоминается в признаке C. Младшие освободившиеся в результате сдвига разряды заполняются нулями. Если при сдвиге влево происходит переполнение, вырабатывается признак переполнения V.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	–	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ASL[.cc]** s1/#5,s2,d

Примеры: **ASL.ne** R1,R5,R12

**ASL** 17,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s1/#5	d	s2	#	0	cc	0	0	0	1	1	1	0	0	1	0	0
-------	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

Автокод: ASL cc, s1/#5, s2, d

Формат 4: **ASL** s/#5, d <XRAM $\beta$ àR.L>

Примеры: **ASL** R1, R2 R8, (A0)+

**ASL** 17, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				R				A				mode		u	0	1	de	#	1	1	0	0	1	0	0		

Автокод: ASL4 s, d, MLR, ea, A, R

ASL4 s, d, MLM, ea, A, R

Формат 5: **ASL** s, d <R/R.L $\beta$ àR/R.L>

Примеры: **ASL** R1, R2 R8.L, R12.L

**ASL** 15, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				RS/Rs				L	RD/Rd				1	1	0	#	1	1	0	0	1	0	0				

Автокод: ASL5 s, d, MR, rs, rd

ASL5 s, d, MRL, RS, RD

Формат 6: **ASL** s, d <R $\beta$ àRC>

Примеры: **ASL** R1, R5 R7, CCR

**ASL** 1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				R				sc	RC				1	0	de	#	1	1	0	0	1	0	0				

Автокод: ASL6 s, d, MRC, rs, RC

ASL6 s, d, MRCd, RC, rd

Формат 7: **ASL**[.cc] s, d <#16/32 $\beta$ àRC/R/R.L>

Пример: **ASL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ASL7** s, d, MIL,cc,#32,RD

**ASL7** s, d, MIC,cc,#16,RC

Формат 8a: **ASL** s3/#5,s4,d2<OP1> <XRAM~~β~~R.L> <YRAM~~α~~R0>

Примеры: **ASL** R1,R2,R5 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

**ASL** 1,R2,R5 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	mode				A	de	0	0	#	0	0	1	0	0					

Автокод: **PROL** **ASL** s3/#5,s4,d2 <OP1> <XRAM~~β~~R.L> <YRAM~~α~~R0>

Формат 8b: **ASL** s3/#5,s4,d2 <OP1> <R/R.L~~β~~R/R.L> <YRAM~~α~~R0>

Примеры: **ASL** R1,R2,R5 ANDL R6,R0,R8 R8,R15 (AT),R0

**ASL** 1,R2,R5 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	RD/Rd				L	0	0	1	#	0	0	1	0	0					

Автокод: PROL ASL s3/#5,s4,d2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: **ASL**[.cc] s3/#5,s4,d2 <OP1> <R.L $\beta$ àR.L>

Примеры: **ASL**.eq R1,R2,R5 ANDL R6,R0,R8 R8.L,R16.L

**ASL**.eq 1,R2,R5 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				cc	RD				0	cc	1	0	#	0	0	1	0	0					

Автокод: PROL ASL cc s3,s4,d2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: **ASL** s3/#5,s4,d2<OP1> <R $\beta$ àRC>

Примеры: **ASL** R1,R2,R5 ANDL R6,R0,R8 CCR,R17

**ASL** 1,R2,R5 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				0	0	RC				sc	de	1	1	#	0	0	1	0	0				

Автокод: PROL ASL s3/#5,s4,d2 <OP1> <R $\beta$ àRC>



*ASLL*      *Арифметический сдвиг влево (long)*      *ASLL*

**Операция:**

3-адресный вариант: short s1; long S2, D;

$$(S2 \ll s1[5:0]) \rightarrow D$$

2-адресный вариант: short s; long D;

$$(D \ll s[5:0]) \rightarrow D$$

**Описание:** Операнд-источник S2 (в 3-адресном варианте) или D (в 2-адресном варианте) сдвигается влево на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте) и результат помещается в операнд-приемник D. Старшие десять разрядов s1 (или s) при этом должны быть установлены в «0». Вытолкнутые за пределы разрядной сетки разряды теряются, кроме последнего вытолкнутого, который запоминается в признаке C. Младшие освободившиеся в результате сдвига разряды заполняются нулями. Если при сдвиге влево происходит переполнение, вырабатывается признак переполнения V.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	–	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ASLL[.cc]** s1/#5, S2, D

Примеры: **ASLL.ne** R2, R6, R12

**ASLL** 17, R6, R12



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					D					S2					#	0	cc					0	0	0	1	1	1	0	1	1	0	0

Автокод: ASLL cc, s1/#5, S2, D

Формат 4: **ASLL** s/#5,D <XRAM $\beta$ aR.L>

Примеры: **ASLL** R2,R4 R8,(A0)+

**ASLL** 17,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s/#5					D					R					A					mode	u	0	1	de	#	1	1	0	1	1	0	0

Автокод: ASLL4 s, D, MLR,ea,A,R

ASLL4 s, D, MLM,ea,A,R

Формат 5: **ASLL** s,D <R/R.L $\beta$ aR/R.L>

Примеры: **ASLL** R2,R4 R8.L,R12.L

**ASLL** 15,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					RS/Rs					L	RD/Rd					1	1	0	#	1	1	0	1	1	0	0

Автокод: ASLL5 s, D, MR,rs,rd

ASLL5 s, D, MRL,RS,RD

Формат 6: **ASLL** S,D <R $\beta$ aRC>

Примеры: **ASLL** R2,R6 R7,CCR

**ASLL** 1,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					R					sc	RC					1	0	de	#	1	1	0	1	1	0	0

Автокод: ASLL6 S, D ,MRC,rs,RC

ASLL6 S, D ,MRCd,RC,rd

Формат 7: **ASLL**[.cc] s,D <#16/32àRC/R/R.L>

Пример: **ASLL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ASLL7** s, D, MIL,cc,#32,RD

**ASLL7** s, D, MIC,cc,#16,RC

Формат 8a: **ASLL** s3/#5,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **ASLL** R1,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

**ASLL** 1,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT	mode				A	de	0	0	#	0	1	1	0	0					

Автокод: PROL **ASLL** s3/#5,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **ASLL** s3/#5,S4,D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **ASLL** R1,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

**ASLL** 1,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT		RD/Rd				L	0	0	1	#	0	1	1	0	0				

Автокод: PROL ASLL s3/#5,S4,D2 <OP2> <R/R.L $\bar{B}$ aR/R.L> <YRAMaR0>

Формат 8с: **ASLL**[.cc] s3/#5,S4,D2 <OP1> <R.L $\bar{B}$ aR.L>

Примеры: **ASLL**.eq R1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

**ASLL**.eq 1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				cc		RD				0	cc	1	0	#	0	1	1	0	0				

Автокод: PROL ASLL cc s3,S4,D2 <OP1> <R.L $\bar{B}$ aR.L>

Формат 8d: **ASLL** s3/#5,S4,D2 <OP1> <R $\bar{B}$ aRC>

Примеры: **ASLL** R1,R2,R6 ANDL R6,R0,R8 CCR,R17

**ASLL** 1,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				0	0	RC				sc	de	1	1	#	0	1	1	0	0				

Автокод: PROL ASLL s3/#5,S4,D2 <OP1> <R $\bar{B}$ aRC>



**Синтаксис ассемблера и код инструкции:**

Формат 1: **ASLX**[.cc] s1/#5,S2,D

Примеры: **ASLX.ne** R1,R6,R12

**ASLX** 17,R6,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					D					S2					#	0	cc					0	0	0	1	1	1	0	0	1	0	1

*Автокод:* **ASLX** cc, s1/#5, S2, D

Формат 4: **ASLX** s/#5,D <XRAM $\beta$ R.L>

Примеры: **ASLX** R1,R2 R8,(A0)+

**ASLX** 17,R2 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s/#5					D					R					A					mode	u	0	1	de	#	1	1	0	0	1	0	1

*Автокод:* **ASLX4** s, D, MLR,ea,A,R

**ASLX4** s, D, MLM,ea,A,R

Формат 5: **ASLX** s,D <R/R.L $\beta$ R/R.L>

Примеры: **ASLX** R1,R2 R8.L,R12.L

**ASLX** 15,R2 R8.L,R12.L

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					RS/Rs					L	RD/Rd					1	1	0	#	1	1	0	0	1	0	1

*Автокод:* **ASLX5** s, D, MR,rs,rd

**ASLX5** s, D, MRL,RS,RD

Формат 6: **ASLX** s,D <R $\beta$ RC>

Примеры: **ASLX** R1,R6 R7,CCR

**ASLX** 1,R6 R7,CCR

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s/#5	D	R	sc	RC	1	0	de	#	1	1	0	0	1	0	1
------	---	---	----	----	---	---	----	---	---	---	---	---	---	---	---

Автокод: ASLX6 s, D, MRC, rs, RC  
ASLX6 s, D, MRCd, RC, rd

Формат 7: **ASLX[.cc]** s, D <#16/32àRC/R/R.L>

Пример: **ASLX** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ASLX7 s, D, MIL, cc, #32, RD  
ASLX7 s, D, MIC, cc, #16, RC

Формат 8а: **ASLX** s3/#5, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **ASLX** R1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0  
**ASLX** 1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT	mode				A	de	0	0	#	0	0	1	0	1					

Автокод: PROL ASLX s3/#5, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **ASLX** s3/#5, S4, D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **ASLX** R1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0  
**ASLX** 1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1/s1	D/d	S2/s2	M	RS/Rs	0	0	0	0	OP1
-------	-----	-------	---	-------	---	---	---	---	-----

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5					D2				S4				AT		RD/Rd				L	0	0	1	#	0	0	1	0	1			

Автокод: PROL ASLX s3/#5, S4, D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: **ASLX**[.cc] s3/#5, S4, D2 <OP1> <R.L $\beta$ àR.L>

Примеры: **ASLX**.eq R1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

**ASLX**.eq 1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1					D/d				S2/s2				M	RS				0	0	0	0	0	OP1								

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5					D2				S4				cc		RD				0	cc	1	0	#	0	0	1	0	1			

Автокод: PROL ASLX cc s3, S4, D2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: **ASLX** s3/#5, S4, D2 <OP1> <R $\beta$ àRC>

Примеры: **ASLX** R1, R2, R6 ANDL R6, R0, R8 CCR, R17

**ASLX** 1, R2, R6 ANDL R6, R0, R8 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1					D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5					D2				S4				0	0	RC				sc	de	1	1	#	0	0	1	0	1			

Автокод: PROL ASLX s3/#5, S4, D2 <OP1> <R $\beta$ àRC>

**ASR**                  *Арифметический сдвиг вправо (short)*      **ASR**

**Операция:**

3-адресный вариант: short s1,s2,d; (s2 >> s1[5:0])    **à**    d

2-адресный вариант: short s,d;                    (d >> s[5:0])    **à**    d

**Описание:** Операнд-источник s2 (в 3-адресном варианте) или d (в 2-адресном варианте) сдвигается вправо на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте). Старшие десять разрядов s1 (или s) при этом должны быть установлены в «0». Результат помещается в операнд-приемник d. Вытолкнутые за пределы разрядной сетки разряды теряются, кроме последнего вытолкнутого, который запоминается в признаке C. Старшие освободившиеся в результате сдвига разряды заполняются знаком.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:**    OP2

**Форматы:**                                    1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	–	√

**Синтаксис ассемблера и код инструкции:**

Формат 1:    **ASR[.cc] s1/#5, s2, d**

Примеры:    **ASR.ne R1, R5, R12**

**ASR**        17, R5, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					d					s2					#	0	cc					0	0	0	1	1	1	1	0	1	0	0

*Автокод:*                    **ASR cc, s1/#5, s2, d**



Формат 4: **ASR** s/#5,d<XRAM $\beta$ àR.L>

Примеры: **ASR** R1,R2 R8,(A0)+

**ASR** 17,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				R				A				mode		u	0	1	de	#	1	1	1	0	1	0	0		

Автокод: ASR4 s, d, MLR,ea,A,R

ASR4 s, d, MLM,ea,A,R

Формат 5: **ASR** s,d <R/R.L $\beta$ àR/R.L>

Примеры: **ASR** R1,R2 R8.L,R12.L

**ASR** 15,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				RS/Rs				L	RD/Rd				1	1	0	#	1	1	1	0	1	0	0				

Автокод: ASR5 s, d, MR,rs,rd

ASR5 s, d, MRL,RS,RD

Формат 6: **ASR** s,d <R $\beta$ àRC>

Примеры: **ASR** R1,R5 R7,CCR

**ASR** 1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				R				sc	RC				1	0	de	#	1	1	1	0	1	0	0				

Автокод: ASR6 s, d ,MRC,rs,RC

ASR6 s, d ,MRCd,RC,rd

Формат 7: **ASR[.cc]** s,d <#16/32 $\beta$ àRC/R/R.L>

Пример: **ASR** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	0	1	1	0	1	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ASR7 s, d, MIL, cc, #32, RD

ASR7 s, d, MIC, cc, #16, RC

Формат 8a: **ASR** s3/#5, s4, d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: **ASR** R1, R2, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

**ASR** 1, R2, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	mode				A	de	0	0	#	1	0	1	0	0					

Автокод: PROL ASR s3/#5, s4, d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: **ASR** s3/#5, s4, d2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: **ASR** R1, R2, R5 ANDL R6, R0, R8 R8, R15 (AT), R0

**ASR** 1, R2, R5 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

6	6	6	6	5	5	5	5	5	5	5	5	5	5	5	4	4	4	4	4	4	4	4	4	4	4	3	3	3	3	3	3	3	3
s3/#5				d2				s4				T	RD/Rd																				

Автокод: PROL ASR s3/#5, s4, d2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: **ASR**[.cc] s3/#5,s4,d2 <OP1> <R.L**β**àR.L>

Примеры: **ASR**.eq R1,R2,R5 ANDL R6,R0,R8 R8.L,R16.L

**ASR**.eq 1,R2,R5 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				cc	RD				0	cc	1	0	#	1	0	1	0	0	0				

Автокод: PROL ASR cc s3,s4,d2 <OP1> <R.L**β**àR.L>

Формат 8d: **ASR** s3/#5,s4,d2<OP1> <R**β**àRC>

Примеры: **ASR** R1,R2,R5 ANDL R6,R0,R8 CCR,R17

**ASR** 1,R2,R5 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				0	0	RC				sc	de	1	1	#	1	0	1	0	0				

Автокод: PROL ASR s3/#5,s4,d2 <OP1> <R**β**àRC>



s1/#5	D	S2	#	0	cc	0	0	0	1	1	1	1	1	1	0	0
-------	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

Автокод: ASRL cc, s1/#5, S2, D

Формат 4: **ASRL** s/#5, D <XRAM $\beta$ àR.L>

Примеры: **ASRL** R2, R4 R8, (A0)+

**ASRL** 17, R4 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				D				R				A				mode		u	0	1	de	#	1	1	1	1	1	1	0	0	

Автокод: ASRL4 s, D, MLR, ea, A, R

ASRL4 s, D, MLM, ea, A, R

Формат 5: **ASRL** s, D <R/R.L $\beta$ àR/R.L>

Примеры: **ASRL** R2, R4 R8.L, R12.L

**ASRL** 15, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				D				RS/Rs				L	RD/Rd				1	1	0	#	1	1	1	1	1	1	0	0			

Автокод: ASRL5 s, D, MR, rs, rd

ASRL5 s, D, MRL, RS, RD

Формат 6: **ASRL** s, D <R $\beta$ àRC>

Примеры: **ASRL** R2, R6 R7, CCR

**ASRL** 1, R6 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				D				R				sc	RC				1	0	de	#	1	1	1	1	1	1	0	0			

Автокод: ASRL6 s, D, MRC, rs, RC

ASRL6 s, D, MRCd, RC, rd

Формат 7: **ASRL[.cc]** s, D <#16/32 $\beta$ àRC/R/R.L>

Пример: **ASRL** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	1	1	1	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ASRL7 s, D, MIL, cc, #32, RD

ASRL7 s, D, MIC, cc, #16, RC

Формат 8a: **ASRL** s3/#5, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **ASRL** R1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

**ASRL** 1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT	mode				A	de	0	0	#	1	1	1	0	0					

Автокод: PROL ASRL s3/#5, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **ASRL** s3/#5, S4, D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **ASRL** R1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

**ASRL** 1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT	RD/Rd				L	0	0	1	#	1	1	1	0	0					

Автокод: PROL ASRL s3/#5, S4, D2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: **ASRL**[.cc] s3/#5, S4, D2 <OP1> <R.LβàR.L>

Примеры: **ASRL.eq** R1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

**ASRL.eq** 1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				cc	RD				0	cc	1	0	#	1	1	1	0	0					

Автокод: PROL ASRL s3/#5, S4, D2 <OP1> <R.LβàR.L>

Формат 8d: **ASRL** s3/#5, S4, D2 <OP1> <RβàRC>

Примеры: **ASRL** R1, R2, R6 ANDL R6, R0, R8 CCR, R17

**ASRL** 1, R2, R6 ANDL R6, R0, R8 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				0	0	RC				sc	de	1	1	#	1	1	1	0	0				

Автокод: PROL ASRL s3/#5, S4, D2 <OP1> <RβàRC>

**ASRLE** *Условный арифметический сдвиг вправо* **ASRLE**

**Операция:** short s1; long S2,D;

```
if(E=1) (D >> s1) à D
if(E=0) (S2 >> s1) à S2
```

**Описание:** В зависимости от значения экспоненциального бита E (6-й разряд CCR), производится арифметический сдвиг вправо операнда S2 или операнда D на количество разрядов, заданное s1. Результат помещается на место исходного операнда. Вытолкнутые за пределы разрядной сетки разряды теряются. Старшие освободившиеся в результате сдвига разряды заполняются знаком. Команда ASRLE используется при выполнении сложения/вычитания в расширенном формате с плавающей точкой 32E16.

**Ограничение:** Команда ASRLE не может сочетаться с пересылкой, в которой источником является какой-либо регистр данных (RF).

**Тип:** OP2

**Форматы:** 8a, 8d

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	√

**Синтаксис ассемблера и код инструкции:**

*Формат 8a:* **ASRLE** s3,S4,D2 <OP1> <XRAMàR.L> <YRAMàR0>

*Пример:* **ASRLE** R1,R2,R6 ANDL R6,R0,R8 (A0)+,R4 (AT),R0

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R			u	0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				AT	mode			A	de	0	0	0	0	1	1	0	1						

*Автокод:* PROL ASRLE s3,S4,D2 <OP1> <XRAMàR.L> <YRAMàR0>



Формат 8d: **ASRLE** s3, S4, D2 <OP1> <RCàR>

Пример: **ASRLE** R1, R2, R6 ANDL R6, R0, R8 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				0	0	RC				sc	de	1	1	0	0	1	1	0	1				

Автокод: PROL ASRLE s3, S4, D2 <OP1> <RCàR>

**ASRX                  Сдвиг арифметический вправо (X16)                  ASRX****Операция:**

3-адресный вариант:    short s1; X16 S2,D;  
 $(S2[31:16] \gg s1[5:0]) \text{ à } D[31:16]$   
 $(S2[15:0] \gg s1[5:0]) \text{ à } D[15:0]$

2-адресный вариант:    short s1; X16 D;  
 $(D[31:16] \gg s[5:0]) \text{ à } D[31:16]$   
 $(D[15:0] \gg s[5:0]) \text{ à } D[15:0]$

**Описание:** В формате X16 операнд-источник S2 (в 3-адресном варианте) или D (в 2-адресном варианте) сдвигается вправо на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте), при этом старшие десять разрядов s1 (или s) должны быть установлены в «0». Результат сдвига старших 16 разрядов операнда S2 (D) помещается в старшие 16 разрядов операнда-приемника D, а результат сдвига младших 16 разрядов операнда помещается в младшие 16 разрядов операнда-приемника D. Вытолкнутые за пределы разрядной сетки разряды теряются. Старшие освободившиеся в результате сдвига разряды заполняются соответствующими знаковыми разрядами.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
*	-	*	-	-
Ur&Ui		Zr&Zi		

**Синтаксис ассемблера и код инструкции:**

Формат 1:    **ASRX[.cc] s1/#5,S2,D**

Примеры: **ASRX**.ne R1,R6,R12

**ASRX** 17,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					D					S2					#	0	cc					0	0	0	1	1	1	1	0	1	0	1

Автокод: ASRX cc, s1/#5, S2, D

Формат 4: **ASRX** s/#5,D <XRAM $\beta$ àR.L>

Примеры: **ASRX** R1,R2 R8,(A0)+

**ASRX** 17,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s/#5					D					R					A					mode	u	0	1	de	#	1	1	1	0	1	0	1

Автокод: ASRX4 s, D, MLR,ea,A,R

ASRX4 s, D, MLM,ea,A,R

Формат 5: **ASRX** s,D <R/R.L $\beta$ àR/R.L>

Примеры: **ASRX** R1,R2 R8.L,R12.L

**ASRX** 15,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					RS/Rs					L	RD/Rd					1	1	0	#	1	1	1	0	1	0	1

Автокод: ASRX5 s, D, MR,rs,rd

ASRX5 s, D, MRL,RS,RD

Формат 6: **ASRX** s,D <R $\beta$ àRC>

Примеры: **ASRX** R1,R6 R7,CCR

**ASRX** 1,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s/#5	D	R	sc	RC	1	0	de	#	1	1	1	0	1	0	1
------	---	---	----	----	---	---	----	---	---	---	---	---	---	---	---

Автокод: ASRX6 s, D ,MRC,rs,RC

ASRX6 s, D ,MRCd,RC,rd

Формат 7: **ASRX**[.cc] s,D <#16/32àRC/R/R.L>

Пример: **ASRX** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ASRX7 s, D, MIL,cc,#32,RD

ASRX7 s, D, MIC,cc,#16,RC

Формат 8а: **ASRX** s3/#5,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **ASRX** R1,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

**ASRX** 1,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT	mode				A	de	0	0	#	1	0	1	0	1					

Автокод: PROL ASRX s3/#5,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **ASRX** s3/#5,S4,D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **ASRX** R1,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

**ASRX** 1,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1/s1	D/d	S2/s2	M	RS/Rs	0	0	0	0	OP1
-------	-----	-------	---	-------	---	---	---	---	-----

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5					D2				S4				AT		RD/Rd				L	0	0	1	#	1	0	1	0	1	0	1	

Автокод: PROL ASRX s3/#5,S4,D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: **ASRX**[.cc] s3/#5,S4,D2 <OP1> <R.L $\beta$ àR.L>

Примеры: **ASRX**.eq R1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

**ASRX**.eq 1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1					D/d				S2/s2				M	RS				0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5					D2				S4				cc	RD				0	cc	1	0	#	1	0	1	0	1	0	1		

Автокод: PROL ASRX cc s3,S4,D2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: **ASRX** s3/#5,S4,D2 <OP1> <R $\beta$ àRC>

Примеры: **ASRX** R1,R2,R6 ANDL R6,R0,R8 CCR,R17

**ASRX** 1,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1					D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5					D2				S4				0	0	RC				sc	de	1	1	#	1	0	1	0	1	0	1	

Автокод: PROL ASRX s3/#5,S4,D2 <OP1> <R $\beta$ àRC>

**B****Ветвление программы****B**

**Операция:** Ветвление программы (**Branch**) одним из способов:

- 1) по метке или непосредственному значению:

$$\text{if(cc)} \quad \text{PC} + \#16 \rightarrow \text{PC};$$

- 2) по адресному регистру:

$$\text{if(cc)} \quad \text{PC} + \text{An} \rightarrow \text{PC}.$$

**Описание:** Если специфицированное условие истинно, происходит программный переход по адресу PC + смещение. Смещение является разностью между текущим адресом и адресом назначения. Если условие ложно, продолжается последовательное выполнение программы.

Смещение может быть задано одним из следующих способов:

- 1) при помощи метки или непосредственного значения #16 (прямой способ),
- 2) путем указания адресного регистра An, хранящего величину смещения (косвенный способ).

Значение смещения является целым числом со знаком, представленным в дополнительном коде.

**Тип:** OP1

**Форматы:** 3m, 3mb

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**

*Формат 3m* (Ветвление по метке или непосредственному значению):  
**B[.cc] #16**

*Примеры:* **B.ne label\_1**

**B 0x17**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
сс				1	#16																0	0	1	1	0	0	1	1	1	0	0

Автокод: ВссI сс, #16

Вссm сс, label

Формат 3mb (Ветвление по адресному регистру): **В[.сс]** An

Пример: **В.еg** A7

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
сс				0	0	0	0	0	0	0	0	0	0	A			0	0	0	0	0	0	1	1	0	0	1	1	1	0	0

Автокод: Всс сс, An

**BD Ветвление программы (отложенное) BD**

**Операция:** Ветвление программы отложенное (**Branch Delayed**) одним из способов:

- 1) по метке или непосредственному значению:

$$\text{if}(cc) \text{ PC} + \#16 \rightarrow \text{PC};$$

- 2) по адресному регистру:

$$\text{if}(cc) \text{ PC} + \text{An} \rightarrow \text{PC}.$$

**Описание:** Если специфицированное условие истинно, происходит программный переход по адресу  $\text{PC} + \text{смещение}$ . Смещение является разностью между текущим адресом и адресом назначения. Если условие ложно, продолжается последовательное выполнение программы.

Смещение может быть задано одним из следующих способов:

- 1) при помощи метки или непосредственного значения #16 (прямой способ),
- 2) путем указания адресного регистра An, хранящего величину смещения (косвенный способ).

Значение смещения является целым числом со знаком, представленным в дополнительном коде.

Отличием команды отложенного ветвления **BD** от обычной команды программного ветвления **B** является то, что следующая за командой отложенного ветвления инструкция выполняется всегда.

**Тип:** OP1

**Форматы:** 3m, 3mb

**Признаки результата:**

U	N	Z	V	C
–	–	–	–	–



**Синтаксис ассемблера и код инструкции:**

Формат 3m (Ветвление по метке или непосредственному значению):

**BD[.cc] #16**

Примеры: **BD.ne label\_1**

**BD 0x17**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
cc				1	#16																0	0	1	1	0	0	1	1	1	1	0

Автокод: **BDccI cc, #16**

**BDccm cc, label**

Формат 3mb (Ветвление по адресному регистру): **BD[.cc] An**

Пример: **BD.eq A7**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
cc				0	0	0	0	0	0	0	0	0	A				0	0	0	0	0	0	1	1	0	0	1	1	1	1	0

Автокод: **BDcc cc, An**

**BS**
**Вызов подпрограммы**
**BS**
**Операция:**

 Вызов подпрограммы (**Branch to Subroutine**) одним из способов:

- 1) по метке или непосредственному значению:

```
if(cc) {PC + 1 à SS; SP[3:0] + 1 à SP[3:0]; PC + #16 à PC;}
```

- 2) по адресному регистру:

```
if(cc) {PC + 1 à SS; SP[3:0] + 1 à SP[3:0]; PC + An à PC;}
```

**Описание:** Если специфицированное условие истинно, происходит программный переход на подпрограмму по адресу PC + смещение. Адрес следующей за командой перехода инструкции заносится в системный стек SS, указатель системного стека SP[3:0] инкрементируется. Смещение является разностью между текущим адресом и адресом назначения. Если условие ложно, продолжается последовательное выполнение программы.

Смещение может быть задано одним из следующих способов:

- 1) при помощи метки или непосредственного значения #16 (прямой способ),
- 2) путем указания адресного регистра An, хранящего величину смещения (косвенный способ).

Значение смещения является целым числом со знаком, представленным в дополнительном коде.

**Тип:** OP1

**Форматы:** 3m, 3mb

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
–	–	–	–	–

**Синтаксис ассемблера и код инструкции:**

Формат 3m (Переход по метке или непосредственному значению):

**BS[.cc] #16**

Примеры: **BS.ne label\_1**

**BS 0x17**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
cc				1	#16																0	0	1	1	0	1	0	1	1	0	0

Автокод: **BSccI cc, #16**

**BSccm cc, label**

Формат 3mb (Переход по адресному регистру): **BS[.cc] An**

Пример: **BS.eq A7**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
cc				0	0	0	0	0	0	0	0	0	0	A			0	0	0	0	0	0	1	1	0	1	0	1	1	0	0

Автокод: **BScc cc, An**

**BTST**
*Проверка разряда (short)*
**BTST**
**Операция:** short s,d; d[s] à c

**Описание:** Значение бита операнда d запоминается в признаке C, номер бита задается операндом-источником s. В качестве операнда-источника s может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	√

**Синтаксис ассемблера и код инструкции:**
Формат 4: **BTST** s/#5,d<XRAMβàR.L>

 Примеры: **BTST** R1,R2 R8,(A0)+

**BTST** 1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				R				A				mode	u	0	1	de	#	1	1	1	0	0	0	0			

 Автокод: **BTST4** s/#5, d, MLR,ea,A,R

**BTST4** s/#5, d, MLM,ea,A,R

Формат 5: **BTST** s/#5,d<R/R.LβàR/R.L>

 Примеры: **BTST** R1,R2 R8.L,R12.L

**BTST** 1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				RS/Rs				L	RD/Rd				1	1	0	#	1	1	1	0	0	0	0				

 Автокод: **BTST5** s/#5, d, MR,rs,rd

**BTST5** s/#5, d, MRL,RS,RD

Формат 6: **BTST** s/#5,d<R $\beta$ àRC>

Примеры: **BTST** R1,R2 R7,CCR

**BTST** 1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					d			R			sc	RC				1	0	de	#	1	1	1	0	0	0	0					

Автокод: **BTST6** s/#5, d ,MRC,rs,RC

**BTST6** s/#5, d ,MRCd,RC,rd

Формат 7: **BTST[.cc]** s,d #16/32àRC/R/R.L>

Пример: **BTST** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d			RD/Rd/RC			L/sc	sr	cc			1	1	1	0	1	1	1	0	0	0	0						

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **BTST7** s, d, MIL,cc,#32,RD

**BTST7** s, d, MIC,cc,#16,RC

Формат 8а: **BTST** s3/#5,d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: **BTST** R1,R2 ANDL R5,R0,R8 R8,(A0)+ (AT),R0

**BTST** 1,R2 ANDL R5,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d			S2/s2			M	R			u	0	0	0	0	OP1												

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2			d2			AT	mode			A	de	0	0	#	1	0	0	0	0								

Автокод: **PROL** **BTST** s3/#5,d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: **BTST** s3/#5,d2 <OP1> <R/R.L $\bar{\mathbf{B}}$ R/R.L> <YRAM $\bar{\mathbf{A}}$ R0>

Примеры: **BTST** R1,R2 LSRL R5,R0,R8 R8,R15 (AT),R0

**BTST** 1,R2 LSRL R5,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				d2				AT	RD/Rd				L	0	0	1	#	1	0	0	0	0					

Автокод: PROL BTST s3/#5,d2 <OP2> <R/R.L $\bar{\mathbf{B}}$ R/R.L> <YRAM $\bar{\mathbf{A}}$ R0>

Формат 8c: **BTST**[.cc] s3/#5,d2 <OP1> <R.L $\bar{\mathbf{B}}$ R.L>

Примеры: **BTST.eq** R1,R2 ANDL R5,R0,R8 R8.L,R16.L

**BTST.eq** 1,R2 ANDL R5,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				d2				cc	RD				0	cc	1	0	#	1	0	0	0	0					

Автокод: PROL BTST cc s3/#5,d2 <OP1 > <R.L $\bar{\mathbf{B}}$ R.L>

Формат 8d: **BTST** s3/#5,d2 <OP1> <R $\bar{\mathbf{B}}$ RC>

Примеры: **BTST** R1,R2 ANDL R5,R0,R8 CCR,R17

**BTST** 1,R2 ANDL R5,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				d2				0	0	RC				sc	de	1	1	#	1	0	0	0	0				

Автокод: PROL      BTST      s3/#5,d2      <OP1>      <R**B**àRC>

**BTSTL**
**Проверка разряда (long)**
**BTSTL**

**Операция:** long s,D; D[s] à C

**Описание:** Значение бита операнда D запоминается в признаке C, номер бита задается операндом-источником s. В качестве операнда-источника s может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	√

**Синтаксис ассемблера и код инструкции:**

Формат 4: **BTSTL** s/#5,D <XRAM $\beta$ àR.L>

Примеры: **BTSTL** R1,R2 R8,(A0)+

**BTSTL** 1,R2 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				A				mode		u	0	1	de	#	1	1	1	1	0	1	0		

*Автокод:* BTSTL4 s/#5, D, MLR,ea,A,R

BTSTL4 s/#5, D, MLM,ea,A,R

Формат 5: **BTSTL** s/#5,D <R/R.L $\beta$ àR/R.L>

Примеры: **BTSTL** R1,R2 R8.L,R12.L

**BTSTL** 1,R2 R8.L,R12.L

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RS/Rs				L	RD/Rd				1	1	0	#	1	1	1	1	0	1	0				

*Автокод:* BTSTL5 s/#5, D, MR,rs,rd



BTSTL5 s/#5, D, MRL,RS,RD

Формат 6: **BTSTL** s/#5,D <R $\beta$ RC>

Примеры: **BTSTL** R1,R2 R7,CCR

**BTSTL** 1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	#	1	1	1	1	0	1	0				

Автокод: BTSTL6 s/#5, D ,MRC,rs,RC

BTSTL6 s/#5, D ,MRCd,RC,rd

Формат 7: **BTSTL**[.cc] s,D #16/32 $\beta$ RC/R/R.L>

Пример: **BTSTL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: BTSTL7 s, D, MIL,cc,#32,RD

BTSTL7 s, D, MIC,cc,#16,RC

Формат 8а: **BTSTL** s3/#5,D2 <OP1> <XRAM $\beta$ R.L> <YRAM $\beta$ R0>

Примеры: **BTSTL** R1,R2 LSRL R5,R0,R8 R8,(A0)+ (AT),R0

**BTSTL** 1,R2 LSRL R5,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				0	0	0	0	0	M	R				u	0	0	0	0	OP1								

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

s3/#5	D2	D2	AT	mode	A	de	0	0	#	1	1	0	1	0
-------	----	----	----	------	---	----	---	---	---	---	---	---	---	---

Автокод: PROL BTSTL s3/#5,D2 <OP1> <XRAM~~β~~R.L> <YRAM~~α~~R0>

Формат 8b: **BTSTL** s3/#5,D2 <OP1> <R/R.L~~β~~R/R.L> <YRAM~~α~~R0>

Примеры: **BTSTL** R1,R2 LSRL R5,R0,R8 R8,R15 (AT),R0

**BTSTL** 1,R2 LSRL R5,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				D2				AT	RD/Rd				L	0	0	1	#	1	1	0	1	0					

Автокод: PROL BTSTL s3/#5,D2 <OP1> <R/R.L~~β~~R/R.L> <YRAM~~α~~R0>

Формат 8c: **BTSTL**[.cc] s3/#5,D2 <OP1> <R.L~~β~~R.L>

Примеры: **BTSTL**.eq R1,R2 LSRL R5,R0,R8 R8.L,R16.L

**BTSTL**.eq 1,R2 LSRL R5,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				D2				cc	RD				0	cc	1	0	#	1	1	0	1	0					

Автокод: PROL BTSTL cc s3/#5,D2 <OP1> <R.L~~β~~R.L>

Формат 8d: **BTSTL** s3/#5,D2 <OP1> <R~~β~~RC>

Примеры: **BTSTL** R1,R2 LSRL R5,R0,R8 CCR,R17

**BTSTL** 1,R2 LSRL R5,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				D2				0	0	RC				sc	de	1	1	#	1	1	0	1	0				

Автокод: PROL      BTSTL      s3/#5,D2      <OP1>      <R**В**àRC>

**CLR**
**Обнуление регистра (short)**
**CLR**
**Операция:** short d; 0 à d

**Описание:** В 16-разрядный операнд-приемник d записывается нуль.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** CLR[.cc] d

**Пример:** CLR.ne R12

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	0	0	0	0	d				d				0	0	cc			0	0	0	1	0	0	0	0	0	0	0	0	0	0	1

**Автокод:** CLR cc d

**Формат 4:** CLR d <XRAMβàR.L>

**Пример:** CLR R2R8, (A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	0	0	0	0	d				R				A			mode			u	0	1	de	0	0	0	0	0	0	0	0	0	1

**Автокод:** CLR4 d, MLR, ea, A, R

CLR4 d, MLM, ea, A, R

**Формат 5:** CLR d <R/R.LβàR/R.L>

**Пример:** CLR R2 R8.L, R12.L

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Автокод: CLR5 d, MR,rs,rd  
CLR5 d, MRL,RS,RD

Формат 6: CLR d <RβàRC>

Пример: CLR R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	d				R				sc	RC				1	0	de	0	0	0	0	0	0	0	0	0	0	0	0	1

Автокод: CLR6 d ,MRC,rs,RC  
CLR6 d ,MRCd,RC,rd

Формат 7: CLR[.cc] d #16/32àRC/R/R.L>

Пример: CLR R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
					6	5	4	3	2	1	0	9	8	7			4	3	2	1													
0	0	0	0	0	d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: CLR7 d, MIL,cc,#32,RD  
CLR7 d, MIC,cc,#16,RC

Формат 8а: <OP2> CLR d <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 CLR R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	d				d				0	R				u	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5					D2/d2					S4/s4					AT		mode			A		de		0		0		#		OP2		

Автокод: PROL <OP2> CLR d <XRAM $\beta$ àR.L>  
<YRAMàR0>

Формат 8b: <OP2> CLR d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 CLR R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	d					d					0	RS/Rs				0	0	0	0	0	0	0	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32		
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd				L		0		0		1		#		OP2		

Автокод: PROL <OP2> CLR d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> CLR d <R.L $\beta$ àR.L>

Пример: LSRL.eq R5,R0,R8 CLR R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
0	0	0	0	0	d					d					0	RS				0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5					D2/d2					S4/s4					cc		RD			0		cc		1		0		#		OP2		

Автокод: PROL <OP2 cc> CLR d <R.L $\beta$ àR.L>

Формат 8d: <OP2> CLR d <R $\beta$ àRC>

Пример: LSRL R5,R0,R8 CLR R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
0	0	0	0	0	d					d					0	Rs/Rd				0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> CLR d <R3àRC>

**CLRL Обнуление регистра (long) CLRL**

**Операция:** long D; 0 à D

**Описание:** В 32-разрядный операнд-приемник D записывается нуль.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **CLRL**[.cc] D

Пример: **CLRL**.ne R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	D				D				0	0	cc			0	0	0	1	0	1	0	0	0	0	0	0	0	1

Автокод: CLRL cc D

Формат 4: **CLRL** D<XRAMβàR.L>

Пример: **CLRL** R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	D				R				A			mode		u	0	1	de	0	0	1	0	0	0	0	0	0	1

Автокод: CLRL4 D, MLR, ea, A, R

CLRL4 D, MLM, ea, A, R

Формат 5: **CLRL** D<R/R.LβàR/R.L>

Пример: **CLRL** R2 R8.L, R12.L

Код инструкции:



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	D					RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	0	0	0	0	0	0	0	1

Автокод: CLRL5 D, MR,rs,rd  
 CLRL5 D, MRL,RS,RD

Формат 6: **CLRL** D<RβàRC>

Пример: **CLRL** R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	D					R				sc	RC				1	0	de	0	0	1	0	0	0	0	0	0	0	0	1

Автокод: CLRL6 D ,MRC,rs,RC  
 CLRL6 D ,MRCd,RC,rd

Формат 7: **CLRL**[.cc] D #16/32àRC/R/R.L>

Пример: **CLRL** R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
					6	5	4	3	2	1	0	9	8	7			4	3	2	1												
0	0	0	0	0	D					RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: CLRL7 D, MIL,cc,#32,RD  
 CLRL7 D, MIC,cc,#16,RC

Формат 8а: <OP2> **CLRL** D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 **CLRL** R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	D					D				0	R				u	0	0	0	0	0	0	0	0	1	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de		0		0		#		OP2				

Автокод: PROL <OP2> CLRL D <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> CLRL D <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Пример: LSRL R5,R0,R8 CLRL R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	0	0	0	0	D				D				0	RS/Rs				0	0	0	0	0	0	0	0	1	0	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L		0		0		1		#		OP2				

Автокод: PROL <OP2> CLRL D <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> CLRL D <R.L $\beta$ R.L>

Пример: LSRL.eq R5,R0,R8 CLRL R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	D				D				0	RS				0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0		cc		1		0		#		OP2				

Автокод: PROL <OP2 cc> CLRL D <R.L $\beta$ R.L>

Формат 8d: <OP2> CLRL D <R $\beta$ RC>

Пример: LSRL R5,R0,R8 CLRL R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
0	0	0	0	0	D				D				0	Rs/Rd				0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3/s3/#5	D2/d2	S4/s4	0	0	RC	sc	de	1	1	#	OP2
----------	-------	-------	---	---	----	----	----	---	---	---	-----

Автокод: PROL <OP2> CLRL D <R3àRC>

**CMP**
**Сравнение (short)**
**CMP**

**Операция:** short  $s, d;$

$\{U, N, Z, V, C\} (d - s) \rightarrow CCR[4:0]$

**Описание:** В формате short вычисляется разность операнда-приемника  $d$  и операнда-источника  $s$ , результат не запоминается. Вырабатываются признаки результата. В качестве операнда-источника  $s$  может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **CMP[.cc] s, d**

*Пример:* **CMP.ne R1, R5**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				d				0	0	cc			0	0	0	1	0	0	1	0	1	0	1				

*Автокод:* **CMP cc, s, d**

Формат 2: **CMP[.cc] #16, d**

*Пример:* **CMP.ne 15, R12**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	d				d				0	0	cc			0	0	1	0	0	0	1	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

*Автокод:* **CMP2 cc, #16, d**

Формат 3: **CMP** #16, d

Пример: **CMP** 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16												0	0	1	1	0	0	1	0	1	0	1					

Автокод: **CMP3** #16, d

Формат 4: **CMP** s, d <XRAM $\beta$ àR.L>

Пример: **CMP** R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	0	1	0	1	0	1		

Автокод: **CMP4** s, d, MLR, ea, A, R

**CMP4** s, d, MLM, ea, A, R

Формат 5: **CMP** s, d <R/R.L $\beta$ àR/R.L>

Пример: **CMP** R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	0	1	0	1				

Автокод: **CMP5** s, d, MR, rs, rd

**CMP5** s, d, MRL, RS, RD

Формат 6: **CMP** s, d <R $\beta$ àRC>

Пример: **CMP** R1, R2 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	0	1	0	1				

Автокод: **CMP6** s, d, MRC, rs, RC

**CMP6** s, d, MRCd, RC, rd

Формат 7: **CMP** [.cc] s,d #16/32àRC/R/R.L>

Пример: **CMP** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **CMP7** s, d, MIL,cc,#32,RD

**CMP7** s, d, MIC,cc,#16,RC

Формат 8a: <OP2> **CMP** s,d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **CMP** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				d				0	R				u	0	0	0	0	0	0	0	1	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **CMP** s,d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **CMP** s,d <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **CMP** R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				d				0	RS/Rs				0	0	0	0	0	0	0	1	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> **CMP** s,d <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **CMP** s,d <R.LβàR.L>

Примеры: LSRL.eq R5,R0,R8      **CMP** R1,R2      R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				d				0	RS				0	0	0	0	0	0	0	0	1	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2							

Автокод: PROL      <OP2 cc>      CMP      s,d      <R.L~~β~~R.L>

Формат 8d: <OP2>      **CMP** s,d      <R~~β~~RC>

Пример: LSRL R5,R0,R8      **CMP** R1,R2      CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				d				0	Rs/Rd				0	0	0	0	0	0	1	0	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC			sc	de	1	1	#	OP2								

Автокод: PROL      <OP2>      CMP      s,d      <R~~β~~RC>

## СМРЕ      Сравнение экспонент формата 32E16    СМРЕ

### Операция:

3-адресный вариант: short s1,s2; long D;

```
if(s1>s2) { s1 à D[15:0]; s1 - s2 à D[31:16]; 1 à CCR[6]; }
else      { s2 à D[15:0]; s2 - s1 à D[31:16]; 0 à CCR[6]; }
```

2-адресный вариант: short s; long D;

```
if(s1>d[15:0]) { s à D[15:0]; s-d à D[31:16]; 1 à CCR[6]; }
else          { d à D[15:0]; d-s à D[31:16]; 0 à CCR[6]; }
```

**Описание:** Совместная обработка экспонент (s1,s2) двух чисел плавающей точки расширенного формата 32E16 с целью приведения к единой экспоненте. Включает три следующих действия:

- 1) формирование экспоненциального признака (бит E): если  $s1 > s2$ , то  $E=1$ , иначе  $E=0$ .
- 2) поиск максимальной экспоненты:  $\max(s1,s2) \rightarrow D[15:0]$
- 3) расчет модуля разности экспонент:  $|S1-S2| \rightarrow D[31:16]$

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

### Признаки результата:

U	N	Z	V	C
-	-	√	-	-

### Синтаксис ассемблера и код инструкции:

Формат 1: СМРЕ[.cc] s1,s2,D

Пример: СМРЕ.ne R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	0	cc				0	0	0	1	0	1	0	1	1	1	0			

**Автокод:** СМРЕ cc, s1, s2, D



Формат 2: **CMPE**[.cc] #16,s2,D

Пример: **CMPE**.ne 15,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	0	cc				0	0	1	0	0	1	0	1	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: **CMPE2** cc, #16, s2, D

Формат 4: **CMPE** s,D <XRAM $\beta$ àR.L>

Пример: **CMPE** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				A				mode		u	0	1	de	0	0	1	0	1	1	1	0		

Автокод: **CMPE4** s, D, MLR,ea,A,R

**CMPE4** s, D, MLM,ea,A,R

Формат 5: **CMPE** s,D <R/R.L $\beta$ àR/R.L>

Пример: **CMPE** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	1	1	1	0				

Автокод: **CMPE5** s, D, MR,rs,rd

**CMPE5** s, D, MRL,RS,RD

Формат 6: **CMPE** s,D <R $\beta$ àRC>

Пример: **CMPE** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	0	0	1	0	1	1	1	0				

Автокод: **CMPE6** s, D, MRC,rs,RC

CMPE6 s, D, MRCd, RC, rd

Формат 7: **CMPE**[.cc] s, D <#16/32àRC/R/R.L>

Пример: **CMPE** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc		sr	cc				1	1	1	0	0	1	0	1	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: CMPE7 s, D, MIL, cc, #32, RD

CMPE7 s, D, MIC, cc, #16, RC

Формат 8a: <OP2>**CMPE** s1, s2, D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **CMPE** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	R				u	0	0	0	0	0	0	1	0	1	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2								

Автокод: PROL <OP2> CMPE s1, s2, D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **CMPE** s1, s2, D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **CMPE** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	RS/Rs				0	0	0	0	0	0	1	0	1	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> CMPE s1, s2, D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс>      **CMPE** s1,s2,D      <R.L $\beta$ aR.L>

Пример: LSRL.eq R6,R0,R8      **CMPE** R2,R4,R6      R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	RS				0	0	0	0	0	0	1	0	1	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc	RD			0	cc	1	0	#	OP2							

Автокод: PROL      <OP2 сс>      **CMPE** s1,s2,D      <R.L $\beta$ aR.L>

Формат 8d: <OP2>      **CMPE** s1,s2,D      <R $\beta$ aRC>

Пример: LSRL R6,R0,R8      **CMPE** R2,R4,R6      CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	Rs/Rd				0	0	0	0	0	1	0	1	1	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC			sc	de	1	1	#	OP2						

Автокод: PROL      <OP2>      **CMPE** s1,s2,D      <R $\beta$ aRC>

**CMPL**
**Сравнение (long)**
**CMPL**
**Операция:** long S,D; {U,N,Z,V,C}( D - S) à CCR[4:0]

**Описание:** В формате long вычисляется разность операнда-приемника D и операнда-источника S, результат не запоминается. Вырабатываются признаки результата. В качестве операнда-источника S может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**
Формат 1: **CMPL[.cc] S,D**

 Пример: **CMPL.ne R2,R6**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				D				0	0	cc			0	0	0	1	0	1	1	0	1	0	1				

Автокод: CMPL cc, S, D

Формат 2: **CMPL[.cc] #32,D**

 Пример: **CMPL.ne 15,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	D				D				0	0	cc			0	0	1	0	0	1	1	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: CMPL2 cc, #32, D

Формат 4: **CMPL** S,D <XRAM**à**R.L>

Пример: **CMPL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	1	0	1	0	1	0	1	

Автокод: **CMPL4** S, D, MLR,ea,A,R

**CMPL4** S, D, MLM,ea,A,R

Формат 5: **CMPL** S,D <R/R.L**à**R/R.L>

Пример: **CMPL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	0	1	0	1				

Автокод: **CMPL5** S, D, MR,rs,rd

**CMPL5** S, D, MRL,RS,RD

Формат 6: **CMPL** S,D <R**à**RC>

Пример: **CMPL** R2,R4 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	0	1	0	1				

Автокод: **CMPL6** S, D ,MRC,rs,RC

**CMPL6** S, D ,MRCd,RC,rd

Формат 7: **CMPL**[.cc] S,D #16/32**à**RC/R/R.L>

Пример: **CMPL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

#16/32

Автокод: CMPL7 S, D, MIL, cc, #32, RD

CMPL7 S, D, MIC, cc, #16, RC

Формат 8а: <OP2> **CMPL** S, D <XRAM $\beta$ àR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **CMPL** R2, R4 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				D				0	R				u	0	0	0	0	0	0	1	1	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2									

Автокод: PROL <OP2> CMPL S, D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **CMPL** S, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **CMPL** R2, R4 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				D				0	RS/Rs				0	0	0	0	0	0	1	1	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> CMPL S, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **CMPL** S, D <R.L $\beta$ àR.L>

Пример: LSRL.eq R6, R0, R8 **CMPL** R2, R4 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				D				0	RS				0	0	0	0	0	0	1	1	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32		
S3/s3/#5					D2/d2					S4/s4					cc		RD					0		cc		1	0	#	OP2				

Автокод: PROL <OP2 cc> CMPL S,D <R.ЛВàR.L>

Формат 8d: <OP2> CMPL S,D <RВàRC>

Пример: LSRL R6,R0,R8 CMPL R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
S					D					D					0		Rs/Rd					0		0		0		0		1	1	0	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32				
S3/s3/#5					D2/d2					S4/s4					0		0		RC					sc		de		1	1	#	OP2				

Автокод: PROL <OP2> CMPL S,D <RВàRC> CMPM Сравнение модулей (short) CMPM

Операция: short s,d;

$$\{U,N,Z,C\}(|d|-|s|) \text{ à } \{CCR[4:2],CCR[0]\}$$

**Описание:** В формате short вычисляется разность модулей операнда-приемника d и операнда-источника s, результат не запоминается. Вырабатываются признаки результата. В качестве операнда-источника s может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	-	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: CMPM[.cc] s,d

Пример: CMPM.ne R1,R5

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				d				0	0	cc				0	0	0	1	0	0	1	0	1	1	0			

Автокод: CМРМ cc, s, d

Формат 2: CМРМ[.cc] #16,d

Пример: CМРМ.ne 15,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	d				d				0	0	cc				0	0	1	0	0	0	1	0	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: CМРМ2 cc, #16, d

Формат 3: CМРМ #16,d

Пример: CМРМ 0x11,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16												0	0	1	1	0	0	1	0	1	1	0					

Автокод: CМРМ3 #16, d

Формат 4: CМРМ s,d <XRAM $\beta$ àR.L>

Пример: CМРМ R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	0	1	0	1	1	0		

Автокод: CМРМ4 s, d, MLR,ea,A,R

CМРМ4 s, d, MLM,ea,A,R

Формат 5: CМРМ s,d <R/R.L $\beta$ àR/R.L>

Пример: CМРМ R1,R2 R8.L,R12.L



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	1	0	1	1	0			

Автокод: CMPM5 s, d, MR,rs,rd  
 CMPM5 s, d, MRL,RS,RD

Формат 6: **CMPM** s,d <RβàRC>

Пример: **CMPM** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	0	1	1	0				

Автокод: CMPM6 s, d ,MRC,rs,RC  
 CMPM6 s, d ,MRCd,RC,rd

Формат 7: **CMPM[.cc]** s,d #16/32àRC/R/R.L>

Пример: **CMPM** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: CMPM7 s, d, MIL,cc,#32,RD  
 CMPM7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **CMPM** s,d <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 **CMPM** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				d				0	R				u	0	0	0	0	0	0	1	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de		0		0		#		OP2				

Автокод: PROL <OP2> CMPM s,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> CMPM s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 CMPM R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
s				d				d				0		RS/Rs			0		0		0		0		0		1		0		1		1		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L		0		0		1		#		OP2				

Автокод: PROL <OP2> CMPM s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> CMPM s,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R5,R0,R8 CMPM R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
s				d				d				0		RS			0		0		0		0		0		0		1		0		1		1		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0		cc		1		0		#		OP2				

Автокод: PROL <OP2 cc> CMPM s,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> CMPM s,d <R $\beta$ àRC>

Пример: LSRL R5,R0,R8 CMPM R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
s				d				d				0		Rs/Rd			0		0		0		0		0		0		1		0		1		1		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5				D2/d2				S4/s4				0		0		RC			sc		de		1		1		#		OP2			

Автокод: PROL <OP2> CMPM s,d <R $\beta$ àRC>



**CMPML**
**Сравнение модулей (long)**
**CMPML**

**Операция:** long S,D;

$$\{U,N,Z,C\}(|D| - |S|) \text{ à } \{CCR[4:2],CCR[0]\}$$

**Описание:** В формате long вычисляется разность модулей операнда-приемника D и операнда-источника S, результат не запоминается. Вырабатываются признаки результата. В качестве операнда-источника S может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **CMPML[.cc] S,D**

Пример: **CMPML.ne R2,R6**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	0	1	0	1	1	0	1	1	0		

Автокод: **CMPML cc, S, D**

Формат 2: **CMPML[.cc] #32,D**

Пример: **CMPML.ne 15,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	1	0	0	1	1	0	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: **CMPML2 cc, #32, D**

Формат 4: **СМРМЛ** S,D <XRAM $\beta$ àR.L>

Пример: **СМРМЛ** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	0	1	1	0	0	

Автокод: **СМРМЛ4** S, D, MLR,ea,A,R

**СМРМЛ4** S, D, MLM,ea,A,R

Формат 5: **СМРМЛ** S,D <R/R.L $\beta$ àR/R.L>

Пример: **СМРМЛ** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	0	1	1	0	0		

Автокод: **СМРМЛ5** S, D, MR,rs,rd

**СМРМЛ5** S, D, MRL,RS,RD

Формат 6: **СМРМЛ** S,D <R $\beta$ àRC>

Пример: **СМРМЛ** R2,R4 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	1	0	0	0	0	1	1	0	1	1	0	0		

Автокод: **СМРМЛ6** S, D ,MRC,rs,RC

**СМРМЛ6** S, D ,MRCd,RC,rd

Формат 7: **СМРМЛ[.cc]** S,D #16/32 $\beta$ àRC/R/R.L>

Пример: **СМРМЛ** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	RD/Rd/RC	L/sc	sr	cc	1	1	0	0	0	1	1	0	1	1	0
---	---	----------	------	----	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: CMPML7 S, D, MIL, cc, #32, RD

CMPML7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **CMPML** S, D <XRAM $\beta$ àR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **CMPML** R2, R4 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				D				0	R				u	0	0	0	0	0	0	1	1	0	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode			A	de	0	0	#	OP2							

Автокод: PROL <OP2> CMPML S, D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **CMPML** S, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **CMPML** R2, R4 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				D				0	RS/Rs				0	0	0	0	0	0	1	1	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	RD/Rd			L	0	0	1	#	OP2							

Автокод: PROL <OP2> CMPML S, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **CMPML** S, D <R.L $\beta$ àR.L>

Пример: LSRL.eq R6, R0, R8 **CMPML** R2, R4 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				D				0	RS				0	0	0	0	0	0	0	1	1	0	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5					D2/d2					S4/s4					cc		RD					0	cc		1	0	#	OP2				

Автокод: PROL <OP2 cc> CMPML S,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> CMPML S,D <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 CMPML R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S					D					D					0		Rs/Rd					0	0	0	0	0	1	1	0	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32		
S3/s3/#5					D2/d2					S4/s4					0		0		RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> CMPML S,D <R $\beta$ àRC>

**CS2**                      **Парная операция выбора большего из двух чисел (short) с фиксацией бита выбора**                      **CS2**

**Операция:** long S1, S2, D;

```

if(S1[15:0] ≤ S1[31:16])          S1[31:16] à D[15:0];
                                (AC0<<1) + 0 à AC0;

else                               S1[15:0] à D[15:0];
                                (AC0<<1) + 1 à AC0;

if(S2[15:0] ≤ S2[31:16])          S2[31:16] à S2[15:0];
                                (AC1<<1) + 0 à AC1;

else                               S1[15:0] à S2[15:0];
                                (AC1<<1) + 1 à AC1;
    
```

**Описание:** Сравниваются между собой старшая и младшая часть первого входного операнда S1 (S1[31:16] и S1[15:0]) и второго входного операнда S2 (S2[31:16] и S2[15:0]). Большее из сравниваемых чисел помещается соответственно в D[15:0] и S2[15:0], а бит, указывающий на результат сравнения, – в младший разряд соответствующего регистра-аккумулятора (AC0[0] и AC1[0]) с одновременным сдвигом всех разрядов AC0 и AC1 на один бит влево.

**Тип:** OP2

**Форматы:** 1, 8

**Признаки результата:**

U	N	Z	V	C
–	*	–	–	*

\*N= $\sim$ AC1[0] – бит результата сравнения второй пары.

\*C= $\sim$ AC0[0] – бит результата сравнения первой пары.



**Синтаксис ассемблера и код инструкции:**

Формат 1: **CS2**[.cc] S1,S2,D

Пример: **CS2.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				1	0	cc				0	0	0	1	1	1	0	0	1	1	0			

Автокод: CS2 cc, S1,S2,D

Формат 8a: **CS2** S3,S4,D2 <OP1> <XRAMβR.L> <YRAMαR0>

Пример: **CS2** R2,R4,R12 ORL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	1	0	0	1	1	0					

Автокод: PROL CS2 S3,S4,D2 <OP1> <XRAMβR.L> <YRAMαR0>

Формат 8b: **CS2** S3,S4,D2 <OP1> <R/R.LβR/R.L> <YRAMαR0>

Пример: **CS2** R2,R4,R12 ORL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	RD/Rd				L	0	0	1	1	0	0	1	1	0					

Автокод: PROL CS2 S3,S4,D2 <OP1> <R/R.LβR/R.L> <YRAMαR0>

Формат 8c: **CS2**[.cc] S3,S4,D2 <OP1> <R.LβR.L>

Пример: **CS2.eq** R2,R4,R12 ORL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc	RD				0	cc	1	0	1	0	0	0	1	1	0				

Автокод: PROL CS2 cc S3,S4,D2 <OP1> <R.L $\bar{B}$ aR.L>

Формат 8d: CS2 S3,S4,D2 <OP1> <R $\bar{B}$ aRC>

Пример: CS2 R2,R4,R18 ORL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	1	0	0	1	1	0				

Автокод: PROL CS2 S3,S4,D2 <OP1> <R $\bar{B}$ aRC>

**CVEF Преобразование формата: 32E16 в 24E8**
**CVEF**
**Операция:**

3-адресный вариант: long S1; short s2; float D;

```

if((s2[15:0] == 0x7fff)&&(S1[30:0]==0))
    {S1[31],0x7f800000} à D[31:0];
else if((s2[15:0] == 0x7fff)&&(S1[30:0]!=0))
    0x7fffffff à D[31:0];
else if((s2[15:0] != 0x7fff)&&(S1[31:0]==0))    0 à D[31:0];
else if((s2[15:0] ≤ -126)&&(S1[31:0]!=0x8000))    0 à D[31:0];
else if((s2[15:0] ≤ -127)&&(S1[31:0]==0x8000))    0 à D[31:0];
else if((s2[15:0] ≥ 129)&&(S1[31:0]!=0x8000))
    {S1[31],0x7f800000} à D[31:0];
else if((s2[15:0] ≥ 128)&&(S1[31:0]==0x8000))
    0xff800000 à D[31:0];
else {S1[31],{{s2[15:0] +127}[7:0]},{|S1[30:8]|}} à D[31:0];
    
```

2-адресный вариант: long S; float D;

```

if((D[15:0] == 0x7fff)&&(S[30:0]==0))
    {S[31],0x7f800000} à D[31:0];
else if((D[15:0] == 0x7fff)&&(S[30:0]!=0))
    0x7fffffff à D[31:0];
else if((D[15:0] != 0x7fff)&&(S[31:0]==0))    0 à D[31:0];
else if((D[15:0] ≤ -126)&&(S[31:0]!=0x8000))    0 à D[31:0];
else if((D[15:0] ≤ -127)&&(S[31:0]==0x8000))    0 à D[31:0];
else if((D[15:0] ≥ 129)&&(S[31:0]!=0x8000))
    {S[31],0x7f800000} à D[31:0];
else if((D[15:0] ≥ 128)&&(S[31:0]==0x8000))
    0xff800000 à D[31:0];
else {S[31],{{D[15:0] +127}[7:0]},{|S[30:8]|}} à D[31:0];
    
```

**Описание:** Выполняется преобразование E-формата плавающей точки повышенной точности 32E16 в формат 24E8 (стандарт IEEE-754). Первый входной операнд – 32-разрядная мантисса S1 (в 3-адресном варианте) или S (в 2-адресном варианте), второй входной операнд - 16-разрядная экспонента s2 (в 3-адресном варианте) или D (в 2-адресном варианте), результат помещается в операнд-приемник D. Сокращение разрядности мантиссы выполняется без округления.

Варианты преобразования

№	Вход		Выход (D)	Пояснение
	S2[15:0]	S1[31:0]		
1	S2≠0x7FFF	0x00000000	0x00000000	число 0
2	0x7FFF	0x00000000	0x7F800000	+∞
3	0x7FFF	0x80000000	0xFF800000	-∞
4	0x7FFF	S1[30:0]≠0	0x7FFFFFFF	QNaN
5	E ≤ -126	S1≠0x80000000	0x00000000	число->0
6	E ≤ -127	S1=0x80000000	0x00000000	число->0
7	E ≥ 129	S1[31]=0	0x7F800000	число->+∞
8	E ≥ 129	S1[31]=1, S1≠0x80000000	0xFF800000	число->-∞
9	E ≥ 128	S1=0x80000000	0xFF800000	число->-∞

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
-	√	√	√	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **CVEF[.cc] S,D**

Пример: **CVEF.ne R2,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	0	cc				0	0	0	1	0	0	1	1	1	0	1			

Автокод: CVEF cc, S1, s2, D

Формат 4: **CVEF** S, D <XRAMβàR.L>

Пример: **CVEF** R2, R4 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	0	1	1	1	0	1		

Автокод: CVEF4 S, D, MLR, ea, A, R

CVEF4 S, D, MLM, ea, A, R

Формат 5: **CVEF** S, D <R/R.LβàR/R.L>

Пример: **CVEF** R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	1	0	1				

Автокод: CVEF5 S, D, MR, rs, rd

CVEF5 S, D, MRL, RS, RD

Формат 6: **CVEF** S, D <RβàRC>

Пример: **CVEF** R2, R4 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				R				sc	RC				1	0	de	0	0	0	1	1	1	0	1				

Автокод: CVEF6 S, D, MRC, rs, RC

CVEF6 S, D, MRCd, RC, rd

Формат 8а: <OP2> **CVEF** S, D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5, R0, R8 **CVEF** R2, R4 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	R				u	0	0	0	0	0	0	0	0	1	1	1	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> CVEF S,D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> CVEF S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 CVEF R2,R4 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	RS/Rs				0	0	0	0	0	0	0	1	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> CVEF S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> CVEF S,D <R.L $\beta$ àR.L>

Пример: LSRL.eq R5,R0,R8 CVEF R2,R4 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	RS				0	0	0	0	0	0	0	1	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> CVEF S,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> CVEF S,D <R $\beta$ àRC>

Пример: LSRL R5,R0,R8 CVEF R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				s2				0	Rs/Rd				0	0	0	0	0	0	1	1	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> CVEF S,D <R**B**àRC>

**CVFE Преобразование формата: 24E8 в 32E16 CVFE****Операция:**

3-адресный вариант: float S1; short s2; long D;

```

if(S1[30:23] == 0)    0 à s2[15:0];          0 à D[31:0];
else if((S1[30:23]==0xff)&&(S1[22:0]==0))
    0x7fff à s2[15:0], {S1[31],{31{0}}}àD[31:0];
else if((S1[30:23]==0xff)&&(S1[22:0]!=0))
    0x7fff à s2[15:0],    0x7fffffff à D[31:0];
else
    {S1[30:23]-126}[15:0] à s2[15:0],
    {{{0,1,S1[22:0]}*(-1)S}[24:0],{7{0}}}[31:0]àD[31:0].

```

2-адресный вариант: float S; short d;

```

if(S[30:23] == 0)    0 à S[31:0];          0 à d[15:0];
else if((S[30:23]==0xff)&&(S[22:0]==0))    {S[31],{31{0}}}à
S[31:0];    0x7fff à d[15:0];
else if((S[30:23]==0xff)&&(S[22:0]!=0))
0x7fffffff àS[31:0];    0x7fff à d[15:0];
else {{{0,1,S[22:0]}*(-1)S}[24:0],{7{0}}}[31:0] à
S[31:0];
    {S[30:23]-126}[15:0] à d[15:0].

```

**Описание:** Выполняется преобразование 32-разрядного формата плавающей точки 24E8 (стандарт IEEE-754) в E-формат повышенной точности 32E16. Входной операнд – S1 (в 3-адресном варианте) или S (в 2-адресном варианте); 16-разрядная экспонента результата помещается по адресу s2 (в 3-адресном варианте) или d (в 2-адресном варианте), 32-разрядная мантисса результата – по адресу D (в 3-адресном варианте) или S (в 2-адресном варианте).



## Варианты преобразования

№	Вход (S1)	Выход		Пояснение
		s2[15:0]	D[31:0]	
1	0x00000000 0x80000000	0x0000	0x00000000	число 0
2	0x7F800000	0x7FFF	0x00000000	+∞
3	0xFF800000	0x7FFF	0x80000000	-∞
4	0x7FFFFFFF	0x7FFF	0X7FFFFFFF F	QNaN
5	[0xx...x] [1xx...x]	$-32766 \leq E \leq +32766$	[01x...x] [10x...x]	число>0 число<0

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
-	√	√	√	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **CVFE[.cc]** S1,s2,D

*Пример:* **CVFE.ne** R2,R4,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	0	cc				0	0	0	1	0	0	1	1	1	0	1			

*Автокод:* CVFE cc, S1,s2, D

Формат 4: **CVFE** S,D <XRAMBàR.L>

*Пример:* **CVFE** R2,R4 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	0	0	1	1	1	0	1		

Автокод: CVFE4 S, D, MLR, ea, A, R

CVFE4 S, D, MLM, ea, A, R

Формат 5: **CVFE** S, D <R/R.L $\beta$ àR/R.L>

Пример: **CVFE** R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	1	1	1	0	1			

Автокод: CVFE5 S, D, MR, rs, rd

CVFE5 S, D, MRL, RS, RD

Формат 6: **CVFE** S, D <R $\beta$ àRC>

Пример: **CVFE** R2, R4 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				R				sc	RC				1	0	de	0	0	0	0	1	1	1	0	1			

Автокод: CVFE6 S, D, MRC, rs, RC

CVFE6 S, D, MRCd, RC, rd

Формат 8а: <OP2> **CVFE** S1, s2, D <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: LSRL R5, R0, R8 **CVFE** R2, R4, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	R				u	0	0	0	0	0	0	0	1	1	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode					A	de	0	0	#	OP2					

Автокод: PROL <OP2> CVFE S1, s2, D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **CVFE** S1,s2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 **CVFE** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	RS/Rs				0	0	0	0	0	0	1	1	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> **CVFE** S1,s2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **CVFE** S1,s2,D <R.L $\beta$ àR.L>

Пример: LSRL.eq R5,R0,R8 **CVFE** R2,R5,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	RS				0	0	0	0	0	0	0	1	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2								

Автокод: PROL <OP2 cc> **CVFE** S1,s2,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> **CVFE** S1,s2,D <R $\beta$ àRC>

Пример: LSRL R5,R0,R8 **CVFE** R2,R4,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				s2				0	Rs/Rd				0	0	0	0	0	0	1	1	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: PROL <OP2> **CVFE** S1,s2,D <R $\beta$ àRC>

**CVFI Преобразование формата: формат 24Е8 в 32-разрядное целое в дополнительном коде CVFI**

**Операция:** float S; long D; (long) S → D

**Описание:** Выполняется преобразование формата: операнд-источник S, который является числом в формате с плавающей точкой 24Е8, преобразуется в 32-разрядное целое со знаком (в дополнительном коде). Результат помещается в операнд-приемник D. Если точное преобразование невозможно, то выполняется округление; способ округления – к ближайшему целому числу, а при равноудаленности – к четному. При выходе результата за пределы целочисленного диапазона, а также в особых случаях выполняется ограничение сверху и снизу.

Особые случаи ограничения результата:

если  $S > 2^{31} - 1$ ,

$S = +\infty$ ,

$S = \text{QNaN}, \text{NaN}$ , то  $D = 0x7FFFFFFF$ ;

если  $S < -2^{31}$ ,

$S = -\infty$ , то  $D = 0x80000000$ .

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	–

**Синтаксис ассемблера и код инструкции:**

Формат 1: CVFI[.cc] S,D

Пример: CVFI.ne R2,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	0	cc				0	0	0	1	0	0	0	1	1	1	0	

Автокод: CVFI cc, S, D

Формат 4: CVFI S,D <XRAM $\beta$ aR.L>

Пример: CVFI R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	0	0	0	1	1	1	0	

Автокод: CVFI4 S, D, MLR,ea,A,R

CVFI4 S, D, MLM,ea,A,R

Формат 5: CVFI S,D <R/R.L $\beta$ aR/R.L>

Пример: CVFI R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	1	1	1	0			

Автокод: CVFI5 S, D, MR,rs,rd

CVFI5 S, D, MRL,RS,RD

Формат 6: CVFI S,D <R $\beta$ aRC>

Пример: CVFI R2,R4 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	0	0	1	1	1	0			

Автокод: CVFI6 S, D ,MRC,rs,RC

CVFI6 S, D ,MRCd,RC,rd

Формат 7: CVFI[.cc] S,D #16/32 $\beta$ aRC/R/R.L>

Пример: CVFI R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	RD/Rd/RC	L/sc	sr	cc	1	1	1	0	0	0	0	1	1	1	0
---	---	----------	------	----	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: CVFI7 S, D, MIL, cc, #32, RD

CVFI7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **CVFI** S, D <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: LSRL R5, R0, R8 **CVFI** R2, R4 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	0	0	0	0	1	1	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode		A	de	0	0	#	OP2											

Автокод: PROL <OP2> CVFI S, D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **CVFI** S, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R5, R0, R8 **CVFI** R2, R4 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS/Rs				0	0	0	0	0	0	0	0	1	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd		L	0	0	1	#	OP2											

Автокод: PROL <OP2> CVFI S, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **CVFI** S, D <R.L $\beta$ àR.L>

Пример: LSRL.eq R5, R0, R8 **CVFI** R2, R4 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS				0	0	0	0	0	0	0	0	0	1	1	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5					D2/d2					S4/s4					cc		RD					0	cc		1	0	#	OP2				

Автокод: PROL <OP2 cc> CVFI S,D <R.LβàR.L>

Формат 8d: <OP2> CVFI S,D <RβàRC>

Пример: LSRL R5,R0,R8 CVFI R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					D					0	0	0	0	0	0	Rs/Rd					0	0	0	0	0	0	0	1	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> CVFI S,D <RβàRC>

***CVIF Преобразование формата: 32-разрядное целое в дополнительном коде в формат 24E8CVIF***

**Операция:** long S; float D; (float) S → D

**Описание:** Выполняется преобразование формата: операнд-источник S, который является 32-разрядным целым числом со знаком (в дополнительном коде), преобразуется в число в формате с плавающей точкой 24E8. Предварительно операнд-источник S преобразуется в прямой код и, при необходимости, округляется к 24-разрядной нормализованной мантиссе. Способ округления – к ближайшему числу, а при равноудаленности – к четному. Результат помещается в операнд-приемник D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
–	√	√	–	–

**Синтаксис ассемблера и код инструкции:**

Формат 1: **CVIF**[.cc] S,D

*Пример:* **CVIF.ne** R2,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	0	1	0	0	0	1	1	1	1		

*Автокод:* CVIF cc, S, D

Формат 4: **CVIF** S,D <XRAMB→R.L>

*Пример:* **CVIF** R2,R4 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	0	0	1	1	1	1			

*Автокод:* CVIF4 S, D, MLR,ea,A,R

CVIF4 S, D, MLM,ea,A,R



Формат 5: **CVIF** S,D <R/R.L $\beta$ àR/R.L>

Пример: **CVIF** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	1	1	1	1			

Автокод: CVIF5 S, D, MR,rs,rd

CVIF5 S, D, MRL,RS,RD

Формат 6: **CVIF** S,D <R $\beta$ àRC>

Пример: **CVIF** R2,R4 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	0	1	1	1	1				

Автокод: CVIF6 S, D ,MRC,rs,RC

CVIF6 S, D ,MRCd,RC,rd

Формат 7: **CVIF[.cc]** S,D #16/32 $\beta$ àRC/R/R.L>

Пример: **CVIF** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	1	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: CVIF7 S, D, MIL,cc,#32,RD

CVIF7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **CVIF** S,D <XRAM $\beta$ àR.L> <YRAM $\beta$ àR0>

Пример: LSRL R5,R0,R8 **CVIF** R2,R4 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	0	0	0	0	0	0	R	u	0	0	0	0	0	0	0	0	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2									

Автокод: PROL <OP2> CVIF S,D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> CVIF S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 CVIF R2,R4 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS/Rs				0	0	0	0	0	0	0	0	1	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> CVIF S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> CVIF S,D <R.L $\beta$ àR.L>

Пример: LSRL.eq R5,R0,R8 CVIF R2,R4 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS				0	0	0	0	0	0	0	0	0	1	1	1	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				cc	RD			0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> CVIF S,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> CVIF S,D <R $\beta$ àRC>

Пример: LSRL R5,R0,R8 CVIF R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	Rs/Rd				0	0	0	0	0	0	0	0	1	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> CVIF S,D <R**B**àRC>

**DEC**
**Декремент (short)**
**DEC**
**Операция:** short s,d; s - 1 à d

**Описание:** В формате short из операнда-источника s вычитается единица, результат помещается в операнд-приемник d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** DEC[.cc] s,d

**Пример:** DEC.ne R1,R5

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s					d					0	0	0	0	0	0	cc					0	0	0	1	0	0	1	0	0	1	0

**Автокод:** DEC cc, s1, d

**Формат 4:** DEC s,d <XRAMBàR.L>

**Пример:** DEC R1,R2 R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s					d					R			A			mode		u	0	1	de	0	0	0	1	0	0	1	0		

**Автокод:** DEC4 s, d, MLR,ea,A,R

DEC4 s, d, MLM,ea,A,R

**Формат 5:** DEC s,d <R/R.LBàR/R.L>

**Пример:** DEC R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	1	0	0	1	0			

Автокод: DEC5 s, d, MR,rs,rd  
 DEC5 s, d, MRL,RS,RD

Формат 6: **DEC** s,d <R**B**àRC>

Пример: **DEC** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	0	0	1	0				

Автокод: DEC6 s, d ,MRC,rs,RC  
 DEC6 s, d ,MRCd,RC,rd

Формат 7: **DEC**[.cc] s,d #16/32àRC/R/R.L>

Пример: **DEC** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	0	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: DEC7 s, d, MIL,cc,#32,RD  
 DEC7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **DEC**[s] s,d <XRAM**B**àR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **DEC** R1,R2 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **DECs** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	M	R				u	0	0	0	0	0	0	0	1	0	0	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de		0		0		#		OP2				

Автокод: PROL <OP2> DEC[s] s,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> DEC[s] s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 DEC R1,R2 R8,R15 (AT),R0

LSRL R5,R0,R8 DEC $\mathbf{s}$  R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
s				d				0		0		0		0		M		RS/Rs				0		0		0		0		1		0		0		1		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd				L		0		0		1		#		OP2			

Автокод: PROL <OP2> DEC[s] s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> DEC[s] s,d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 DEC R1,R2 R8.L,R16.L

LSRL.eq R5,R0,R8 DEC $\mathbf{s}$  R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0										
s				d				0		0		0		0		M		RS				0		0		0		0		0		1		0		0		1		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0		cc		1		0		#		OP2			

Автокод: PROL <OP2 cc> DEC[s] s,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> DEC[s] s,d <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 DEC R1,R2 CCR,R17

LSRL R5,R0,R8 DEC $\mathbf{s}$  R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s	d	0	0	0	0	0	0	M	Rs/Rd	0	0	0	0	0	0	1	0	0	1	0
---	---	---	---	---	---	---	---	---	-------	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: PROL      <OP2>      DEC[s]      s, d      <R**B**aRC>

**DECL**
*Декремент (long)*
**DECL**

**Операция:** long S,D; S - 1 à D

**Описание:** В формате long из операнда-источника S вычитается единица, результат помещается в операнд-приемник D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **DECL**[.cc] S,D

*Пример:* **DECL.ne** R2,R6

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	0	cc				0	0	0	1	0	1	1	0	0	1	0	

*Автокод:* DECL cc, S, D

Формат 4: **DECL** S,D <XRAM**В**àR.L>

*Пример:* **DECL** R2,R4 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	1	0	0	1	0			

*Автокод:* DECL4 S, D, MLR,ea,A,R

DECL4 S, D, MLM,ea,A,R

Формат 5: **DECL** S,D <R/R.L**В**àR/R.L>

*Пример:* **DECL** R2,R4 R8.L,R12.L



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	0	0	1	0			

Автокод: DECL5 S, D, MR,rs,rd  
DECL5 S, D, MRL,RS,RD

Формат 6: **DECL** S,D <RβàRC>

Пример: **DECL** R2,R4 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	0	0	1	0				

Автокод: DECL6 S, D ,MRC,rs,RC  
DECL6 S, D ,MRCd,RC,rd

Формат 7: **DECL**[.cc] S,D #16/32àRC/R/R.L>

Пример: **DECL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	0	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: DECL7 S, D, MIL,cc,#32,RD  
DECL7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **DECL**[s] S,D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 **DECL** R2,R4 R8,(A0)+ (AT),R0

LSRL R6,R0,R8 **DECLs** R2,R4 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	M	R				u	0	0	0	0	0	0	1	1	0	0	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de	0	0	#	OP2								

Автокод: PROL <OP2> DECL[s] S,D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> DECL[s] S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 DECL R2,R4 R8,R15 (AT),R0

LSRL R6,R0,R8 DECLs R2,R4 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	M	RS/Rs				0	0	0	0	0	1	1	0	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> DECL[s] S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> DECL[s] S,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6,R0,R8 DECL R2,R4 R8.L,R16.L

LSRL.eq R6,R0,R8 DECLs R2,R4 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	M	RS				0	0	0	0	0	0	1	1	0	0	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> DECL[s] S,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> DECL[s] S,D <R $\beta$ àRC>

Примеры: LSRL R6,R0,R8 DECL R2,R4 CCR,R17

LSRL R6,R0,R8 DECLs R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	0	0	0	0	0	M	Rs/Rd	0	0	0	0	0	1	1	0	0	1	0
---	---	---	---	---	---	---	---	-------	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: *PROL*      <OP2>      *DECL[s]*      *S,D*      <**R**à*RC*>

**DISPFX    Распаковка (дробная) X8 в X16    DISPFX**

**Операция:** (X8) s; (X16) D;

$$\{s[15:8], 8\{0\}\} \rightarrow D[15:0]$$

$$\{s[7:0], 8\{0\}\} \rightarrow D[31:16]$$

**Описание:** Выполняется распаковка содержимого операнда-источника s в операнд-приемник D. Источник s рассматривается как комплексное число дробное X8 (старший байт – действительная часть, младший байт – мнимая часть). Приемник D - дробное комплексное число в формате X16 (старшее полуслово – действительная часть, младшее полуслово – мнимая часть).

При этом старшие 8 разрядов регистра-источника s дополненные справа нулями помещаются в старшие 16 разрядов регистра-приемника D, младшие 8 разрядов регистра-источника s дополненные справа нулями помещаются в младшие 16 разрядов регистра-приемника D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
Ur&Ui	Nr	Zr	Zi	Ni

**Синтаксис ассемблера и код инструкции:**

Формат 1: **DISPFX[.cc] s, D**

Примеры: **DISPFX.ne R1, R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				x x x x x				0	0	cc				0	0	0	1	1	0	0	1	1	1	0			

**Автокод:** DISPFX cc, s, D

Формат 4: **DISPFX** s,D <XRAM**β**àR.L>

Пример: **DISPFX** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				A				mode		u	0	1	de	0	1	0	0	1	1	1	0		

Автокод: **DISPFX4** s, D, MLR,ea,A,R

**DISPFX4** s, D, MLM,ea,A,R

Формат 5: **DISPFX** s,D <R/R.L**β**àR/R.L>

Пример: **DISPFX** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	1	1	1	0				

Автокод: **DISPFX5** s, D, MR,rs,rd

**DISPFX5** s, D, MRL,RS,RD

Формат 6: **DISPFX** s,D <R**β**àRC>

Пример: **DISPFX** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	0	1	0	0	1	1	1	0				

Автокод: **DISPFX6** s, D ,MRC,rs,RC

**DISPFX6** s, D ,MRCd,RC,rd

Формат 7: **DISPFX[.cc]** s,D #16/32**β**àRC/R/R.L>

Пример: **DISPFX** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	1	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: DISPFX7 s, D, MIL, cc, #32, RD

DISPFX7 s, D, MIC, cc, #16, RC

Формат 8a: <OP2>            **DISPFX** s, D            <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R5, R0, R8    **DISPFX** R1, R2    R8, (A0)+            (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																				
s				D				0				0				0				0				R				u				0				0				0				0				1				0				0				1				0				1				1				1				0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32			
S3/s3/#5					D2/d2					S4/s4					AT		mode			A		de		0		0		#		OP2				

Автокод: PROL <OP2>            **DISPFX** s, D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2>            **DISPFX** s, D <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R5, R0, R8    **DISPFX** R1, R2    R8, R15            (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																												
s				D				0				0				0				0				0				RS/Rs				0				0				0				0				1				0				0				1				1				1				0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32			
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd			L		0		0		1		#		OP2				

Автокод: PROL <OP2>            **DISPFX** s, D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc>            **DISPFX** s, D            <R.LβàR.L>

Примеры: LSRL.eq R5, R0, R8    **DISPFX** R1, R2    R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																												
s				D				0				0				0				0				0				RS				0				0				0				0				1				0				0				1				1				1				0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3/s3/#5	D2/d2	S4/s4	cc	RD	0	cc	1	0	#	OP2
----------	-------	-------	----	----	---	----	---	---	---	-----

Автокод: PROL <OP2 cc> DISPFX s,D <R.L $\bar{\alpha}$ R.L>

Формат 8d: <OP2> DISPFX s,D <R $\bar{\alpha}$ RC>

Пример: LSRL R5,R0,R8 DISPFX R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				0	0	0	0	0	0	Rs/Rd				0	0	0	0	1	0	0	1	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> DISPFX s,D <R $\bar{\alpha}$ RC>

### *DISPX* *Распаковка (целочисленная) X8 в X16* *DISPX*

**Операция:** (X8) *s*; (X16) *D*;

{8{s[15]}, s[15:8]} **à** D[15:0]

{8{s[7]}, s[7:0]} **à** D[31:16]

**Описание:** Выполняется распаковка содержимого операнда-источника *s* в операнд-приемник *D*. Источник *s* рассматривается как целое комплексное число X8 (старший байт – действительная часть, младший байт – мнимая часть). Приемник *D* - целое комплексное число в формате X16 (старшее полуслово – действительная часть, младшее полуслово – мнимая часть).

При этом старшие 8 разрядов регистра-источника *s* расширенные знаком (значением 15-го разряда) помещаются в старшие 16 разрядов регистра-приемника *D*, младшие 8 разрядов регистра-источника *s* расширенные знаком (значением 7-го разряда) помещаются в младшие 16 разрядов регистра-приемника *D*.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
Ur&Ui	Nr	Zr	Zi	Ni

**Синтаксис ассемблера и код инструкции:**

Формат 1: **DISPX** [.cc] *s*/#5,*D*

Примеры: **DISPX.ne** R1,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				x x x x x				0	0	cc				0	0	0	1	1	0	0	1	1	0	1			

Автокод: **DISPX** cc, *s*, *D*

Формат 4: **DISPX** *s*,*D* <XRAM**В**àR.L>



Пример: **DISPX** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				A				mode		u	0	1	de	0	1	0	0	1	1	0	1		

Автокод: **DISPX4** s, D, MLR,ea,A,R

**DISPX4** s, D, MLM,ea,A,R

Формат 5: **DISPX** s,D <R/R.L**β**àR/R.L>

Пример: **DISPX** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	1	1	0	1				

Автокод: **DISPX5** s, D, MR,rs,rd

**DISPX5** s, D, MRL,RS,RD

Формат 6: **DISPX** s,D <R**β**àRC>

Пример: **DISPX** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	0	1	0	0	1	1	0	1				

Автокод: **DISPX6** s, D ,MRC,rs,RC

**DISPX6** s, D ,MRCd,RC,rd

Формат 7: **DISPX[.cc]** s,D #16/32**à**RC/R/R.L>

Пример: **DISPX** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

#16/32

Автокод: DISPX7 s, D, MIL, cc, #32, RD

DISPX7 s, D, MIC, cc, #16, RC

Формат 8a: &lt;OP2&gt; DISPX[s] s, D &lt;XRAMβàR.L&gt; &lt;YRAMàR0&gt;

Примеры: LSRL R5, R0, R8 DISPX R1, R2 R8, (A0)+ (AT), R0

LSRL R5, R0, R8 DISPXs R1, R2 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				0	0	0	0	0	0	R				u	0	0	0	0	1	0	0	1	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2									

Автокод: PROL &lt;OP2&gt; DISPX[s] s, D &lt;XRAMβàR.L&gt; &lt;YRAMàR0&gt;

Формат 8b: &lt;OP2&gt; DISPX[s] s, D &lt;R/R.LβàR/R.L&gt; &lt;YRAMàR0&gt;

Примеры: LSRL R5, R0, R8 DISPX R1, R2 R8, R15 (AT), R0

LSRL R5, R0, R8 DISPXs R1, R2 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				0	0	0	0	0	0	RS/Rs				0	0	0	0	1	0	0	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL &lt;OP2&gt; DISPX[s] s, D &lt;R/R.LβàR/R.L&gt; &lt;YRAMàR0&gt;

Формат 8c: &lt;OP2.cc&gt; DISPX[s] s, D &lt;R.LβàR.L&gt;

Примеры: LSRL.eq R5, R0, R8 DISPX R1, R2 R8.L, R16.L

LSRL.eq R5, R0, R8 DISPXs R1, R2 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s	D	0	0	0	0	0	0	RS	0	0	0	0	0	1	0	0	1	1	0	1
---	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2								

Автокод: PROL <OP2 cc> DISPX[s] s,D <R.LβàR.L>

Формат 8d: <OP2> DISPX s,D <RβàRC>

Примеры: LSRL R5,R0,R8 DISPX R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				0	0	0	0	0	0	Rs/Rd				0	0	0	0	1	0	0	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC			sc	de	1	1	#	OP2									

Автокод: PROL <OP2> DISPX s,D <RβàRC>

*DO*
*Оператор цикла*
*DO*

**Операция:** Инициализация программного цикла одним из способов:

1) число повторений цикла  $N$  - по непосредственному значению  $\#N$  ( $0 < N < 16384$ ); адрес (абсолютный) последней команды цикла - по метке или непосредственному значению  $\#16$ :

```
PC + 2 à SS;      SP[3:0] + 1 à SP[3:0];
LA      à CSL;    #16      à LA;
LC      à CSH;    {2'b01, #N[13:0]} à LC;
                      CP[2:0] + 1 à CP[2:0];
```

2) число повторений цикла  $N$  - по непосредственному значению  $\#N$  ( $0 < N < 16384$ ); адрес (относительно PC) последней команды цикла - по метке или непосредственному значению  $\#16$ :

```
PC + 2 à SS;      SP[3:0] + 1 à SP[3:0];
LA      à CSL;    PC + #16   à LA;
LC      à CSH;    {2'b01, #N[13:0]} à LC;
                      CP[2:0] + 1 à CP[2:0];
```

3) число повторений цикла  $N$  - по регистру  $d$ ; адрес (абсолютный) последней команды цикла - по метке или непосредственному значению  $\#16$ :

```
PC + 1 à SS;      SP[3:0] + 1 à SP[3:0];
LA      à CSL;    #16      à LA;
LC      à CSH;    {2'b01, d[13:0]} à LC;
                      CP[2:0] + 1 à CP[2:0];
```

4) число повторений цикла  $N$  - по регистру  $d$ ; адрес (относительно PC) последней команды цикла - по метке или непосредственному значению  $\#16$ :

```
PC + 1 à SS;      SP[3:0] + 1 à SP[3:0];
LA      à CSL;    PC + #16   à LA;
LC      à CSH;    {2'b01, d[13:0]} à LC;
                      CP[2:0] + 1 à CP[2:0];
```

**Описание:** Выполняется инициализация программного цикла, включающая определение адреса первой и последней команды цикла, а также число повторений.

Адрес первой команды цикла (адрес следующей за командой *DO* инструкции) вычисляется как  $PC + 2$  в случае, если команда *DO* кодируется двумя словами (формат 2d), или

PC + 1, если команда DO кодируется одним словом (формат 3). Этот адрес заносится в системный стек SS, указатель системного стека SP[3:0] инкрементируется.

Адрес последней команды цикла определяется при помощи непосредственного значения #16, которое рассматривается либо как абсолютный адрес (1-й и 3-й способ), либо как смещение относительно текущего значения PC (2-й и 4-й способ). Значение смещения является целым числом со знаком, представленным в дополнительном коде. Адрес последней команды цикла загружается в регистр LA, а предыдущее значение LA – в стек CSL.

Число повторений цикла N может быть задано при помощи либо непосредственного значения #N (1-й и 2-й способ), либо 16-разрядного регистра данных d (3-й и 4-й способ). Число повторений цикла загружается в разряды [13:0] регистра LC (разряды [15:14] отведены под флаги), а предыдущее значение LC – в стек CSH. Указатель стека циклов CP[2:0] инкрементируется.

При завершении цикла DO стеки возвращают последние загруженные в них значения в соответствующие регистры, указатели SP[3:0], CP[2:0] декрементируются, и продолжается последовательное исполнение программы.

При использовании цикла DO следует учитывать ограничения, приведенные в п. 9.2.

**Тип:** OP1

**Форматы:** 2d, 3

**Признаки результата:**

U	N	Z	V	C
–	–	–	–	–

**Синтаксис ассемблера и код инструкции:**

*Формат 2d (Число повторений цикла определяется по непосредственному значению #N, абсолютный адрес последней команды цикла – по непосредственному значению #16):*

**DO** #N, ##16

*Пример:* **DO** 5, #0xfe

Формат 2d (Число повторений цикла определяется по непосредственному значению #N, абсолютный адрес последней команды цикла - по метке):

**DO** #N, #label

Пример: **DO** 5, #label\_1

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	#16																0	0	1	0	0	0	0	1	1	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#N																		

Автокод: **DO** #N, #16

**DOM** #N, label

Формат 2d (Число повторений цикла определяется по непосредственному значению #N, адрес последней команды цикла относительно PC - по непосредственному значению #16):

**DO** #N, #16

Пример: **DO** 5, 0xfe

Формат 2d (Число повторений цикла определяется по непосредственному значению #N, адрес последней команды цикла относительно PC - по метке):

**DO** #N, label

Пример: **DO** 5, label\_1

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	#16																0	0	1	0	0	0	0	1	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#N																		

Автокод: **DOR** #N, #16

**DORM** #N, label

Формат 3 (Число повторений цикла определяется по регистру *d*, абсолютный адрес последней команды цикла - по непосредственному значению #16):

**DO** *d*,##16

Пример: **DO** R15,#0xfe

Формат 3 (Число повторений цикла определяется по регистру *d*, абсолютный адрес последней команды цикла - по метке):

**DO** *d*,#*label*

Пример: **DO** R15,#*label\_1*

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16												0	0	1	1	0	0	0	1	1	1	0					

Автокод: DO\_R *d*,#16

DO\_RM *d*,*label*

Формат 3 (Число повторений цикла определяется по регистру *d*, адрес последней команды цикла относительно PC - по непосредственному значению #16):

**DO** *d*,#16

Пример: **DO** R15,0xfe

Формат 3 (Число повторений цикла определяется по регистру *d*, адрес последней команды цикла относительно PC - по метке):

**DO** *d*,*label*

Пример: **DO** R15,*label\_1*

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16												0	0	1	1	0	0	0	1	1	1	1					

Автокод: DOR\_R *d*,#16

DOR\_RM *d*,*label*

**DOFOR Оператор бесконечного цикла DOFOR**

**Операция:** Инициализация бесконечного программного цикла одним из способов:

1) адрес (абсолютный) последней команды цикла определяется по метке или непосредственному значению #16:

```
PC + 1 à SS;                SP[3:0] + 1 à SP[3:0];
LA    à CSL;    #16 à LA;
LC    à CSH;    0x8000 à LC;  CP[2:0] + 1 à CP[2:0];
```

2) адрес (относительно PC) последней команды цикла определяется по метке или непосредственному значению #16:

```
PC + 1 à SS;                SP[3:0] + 1 à SP[3:0];
LA    à CSL;    PC + #16 à LA;
LC    à CSH;    0x8000 à LC;  CP[2:0] + 1 à CP[2:0];
```

**Описание:** Выполняется инициализация бесконечного программного цикла, включающая определение адреса первой и последней команды цикла.

Адрес первой команды цикла (адрес следующей за командой DOFOR инструкции) вычисляется как PC + 1. Этот адрес заносится в системный стек SS, указатель системного стека SP[3:0] инкрементируется.

Адрес последней команды цикла определяется при помощи непосредственного значения #16, которое рассматривается либо как абсолютный адрес (1-й способ), либо как смещение относительно текущего значения PC (2-й способ). Значение смещения является целым числом со знаком, представленным в дополнительном коде. Адрес последней команды цикла загружается в регистр LA, а предыдущее значение LA – в стек CSL.

В разряд 15 регистра LC (флаг бесконечного цикла FV) заносится «1», а предыдущее значение регистра LC загружается в стек CSH. Указатель стека циклов CP[2:0] инкрементируется.

При использовании цикла DOFOR следует учитывать ограничения, приведенные в п. 9.2.



**Тип:** OP1

**Форматы:** 3

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 3 (Абсолютный адрес последней команды цикла определяется по непосредственному значению #16):

**DOFOR ##16**

Пример: **DOFOR #0xfe**

Формат 3 (Абсолютный адрес последней команды цикла определяется по метке):

**DOFOR #label**

Пример: **DOFOR #label\_1**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	#16																0	0	1	1	0	0	0	1	0	1	0

Автокод: **DOFOR #16**

**DOFORM #label**

Формат 3 (Адрес последней команды цикла относительно PC определяется по непосредственному значению #16):

**DOFOR #16**

Пример: **DOFOR 0xfe**

Формат 3 (Адрес последней команды цикла относительно PC определяется по метке):

**DOFOR label**

Пример: **DOFOR label\_1**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	#16																0	0	1	1	0	0	0	1	0	1	1

Автокод: DOFORR #16

DOFORRM label

**ENDDO**
**Окончание цикла**
**ENDDO**

**Операция:** Окончание программного цикла.

$LA + 1(2) \rightarrow PC;$                        $SP[3:0] - 1 \rightarrow SP[3:0];$   
 $CSL \rightarrow LA;$   
 $CSH \rightarrow LC;$                                $CP[2:0] - 1 \rightarrow CP[2:0];$

**Описание:** Выполняется выход из текущего программного цикла DO, DOFOR. Если в стадии исполнения находится несколько вложенных друг в друга циклов, то по команде ENDDO производится выход из самого внутреннего.

При выходе из цикла стеки возвращают последние загруженные в них значения в соответствующие регистры: стек CSL – в регистр LA, стек CSH – в регистр LC. В регистр PC загружается адрес команды, следующей за последней командой цикла. Указатели SP[3:0], CP[2:0] декрементируются, после чего продолжается последовательное исполнение программы.

**Тип:** OP1

**Форматы:** 3

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 3 : **ENDDO**

*Пример:* **ENDDO**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	1	1

*Автокод:* ENDDO

***EOR                    Логическое исключающее ИЛИ (short)   EOR*****Операция:*****3-адресный вариант:*** short *s1,s2,d*;       $s1 \oplus s2 \rightarrow d$ ***2-адресный вариант:*** short *s,d*;       $s \oplus d \rightarrow d$ 

**Описание:** В формате short выполняется сложение по модулю 2 операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:*****Формат 1:*** ***EOR[.cc]*** *s1,s2,d****Пример:*** ***EOR.ne*** R1,R5,R12***Код инструкции:***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc			0	0	0	1	1	0	0	1	0	0	0				

***Автокод:***      ***EOR cc, s1, s2, d******Формат 2:*** ***EOR[.cc]*** #16,*s2,d****Пример:*** ***EOR.ne*** 15,R5,R12***Код инструкции:***

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc			0	0	1	0	1	0	0	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: EOR2 cc, #16, s2, d

Формат 3: EOR #16, d

Пример: EOR 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	1	0	0	1	0	0	0						

Автокод: EOR3 #16, d

Формат 4: EOR s, d <XRAM $\beta$ àR.L>

Пример: EOR R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	0	0	1	0	0	0		

Автокод: EOR4 s, d, MLR, ea, A, R

EOR4 s, d, MLM, ea, A, R

Формат 5: EOR s, d <R/R.L $\beta$ àR/R.L>

Пример: EOR R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	1	0	0	0				

Автокод: EOR5 s, d, MR, rs, rd

EOR5 s, d, MRL, RS, RD

Формат 6: EOR s, d <R $\beta$ àRC>

Пример: EOR R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	1	0	0	0	0	0	0	0

Автокод: EOR6 s, d, MRC, rs, RC

EOR6 s, d, MRCd, RC, rd

Формат 7: **EOR**[.cc] s, d <#16/32àRC/R/R.L>

Пример: **EOR** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	1	0	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: EOR7 s, d, MIL, cc, #32, RD

EOR7 s, d, MIC, cc, #16, RC

Формат 8a: <OP2> **EOR** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **EOR** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	R				u	0	0	0	0	1	0	0	1	0	0	0	0	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **EOR** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **EOR** s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **EOR** R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	RS/Rs				0	0	0	0	1	0	0	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> EOR s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> EOR s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 EOR R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	RS				0	0	0	0	0	1	0	0	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0	cc	1	0	#	OP2								

Автокод: PROL <OP2 cc> EOR s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> EOR s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 EOR R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	Rs/Rd				0	0	0	0	1	0	0	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> EOR s1,s2,d <R $\beta$ àRC>

**EORL**      *Логическое исключающее ИЛИ (long)*      **EORL**

**Операция:**

3-адресный вариант: long S1, S2, D;       $S1 \oplus S2 \rightarrow D$

2-адресный вариант: long S, D;       $S \oplus D \rightarrow D$

**Описание:** В формате long выполняется сложение по модулю 2 операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте), результат помещается в операнд-приемник D. В качестве первого операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:**      OP1

**Форматы:**      1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **EORL[.cc]** S1, S2, D

*Пример:*      **EORL.ne** R2, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	1	0	1	1	0	0	0				

*Автокод:*      **EORL cc, S1, S2, D**

Формат 2: **EORL[.cc]** #32, S2, D

*Пример:*      **EORL.ne** 15, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	1	0	1	0	1	1	0	0	0				



63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: EORL2 cc, #32, S2, D

Формат 4: **EORL** S,D <XRAM $\beta$ aR.L>

Пример: **EORL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	1	0	1	1	0	0	0			

Автокод: EORL4 S, D, MLR,ea,A,R

EORL4 S, D, MLM,ea,A,R

Формат 5: **EORL** S,D <R/R.L $\beta$ aR/R.L>

Пример: **EORL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	1	0	0	0				

Автокод: EORL5 S, D, MR,rs,rd

EORL5 S, D, MRL,RS,RD

Формат 6: **EORL** S,D <R $\beta$ aRC>

Пример: **EORL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	1	0	0	0				

Автокод: EORL6 S, D, MRC,rs,RC

EORL6 S, D, MRCd,RC,rd

Формат 7: **EORL**[.cc] S,D <#16/32 $\beta$ aRC/R/R.L>

Пример: **EORL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: EORL7 S, D, MIL, cc, #32, RD

EORL7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **EORL** S1, S2, D <XRAM~~β~~àR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **EORL** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	R				u	0	0	0	0	1	0	1	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> EORL S1, S2, D <XRAM~~β~~àR.L> <YRAMàR0>

Формат 8b: <OP2> **EORL** S1, S2, D <R/R.L~~β~~àR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **EORL** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	RS/Rs				0	0	0	0	1	0	1	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> EORL S1, S2, D <R/R.L~~β~~àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **EORL** S1, S2, D <R.L~~β~~àR.L>

Пример: LSRL.eq R6, R0, R8 **EORL** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	RS				0	0	0	0	0	1	0	1	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> EORL S1,S2,D <R $\beta$ RC>

Формат 8d: <OP2> **EORL** S1,S2,D <R $\beta$ RC>

Пример: LSRL R6,R0,R8 **EORL** R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	Rs/Rd				0	0	0	0	1	0	1	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> EORL S1,S2,D <R $\beta$ RC>

**FADD**
**Сложение (24E8)**
**FADD**
**Операция:**

3-адресный вариант: float S1,S2,D;     S1 + S2 → D

2-адресный вариант: float S,D;     S + D     → D

**Описание:** В формате 24E8 выполняется сложение операндов-источников (S1,S2 - в 3-адресном варианте; S,D - в 2-адресном варианте), результат помещается в операнд-приемник D. В качестве первого операнда-источника S1 в 3-адресном варианте может использоваться непосредственный операнд #32 (формат 2).

**Тип:**     OP1

**Форматы:**     1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	√	√	√	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FADD[.cc]** S1,S2,D

Пример:     **FADD.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	0	0	1	0	1	0				

Автокод:     FADD cc, S1, S2, D

Формат 2: **FADD[.cc]** #32,S2,D

Пример:     **FADD.ne** 15,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	1	0	0	0	0	1	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: FADD2 cc, #32, S2, D

Формат 4: **FADD** S,D <XRAM $\beta$ àR.L>

Пример: **FADD** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	0	0	0	1	0	1	0	

Автокод: FADD4 S, D, MLR,ea,A,R

FADD4 S, D, MLM,ea,A,R

Формат 5: **FADD** S,D <R/R.L $\beta$ àR/R.L>

Пример: **FADD** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	1	0	1	0			

Автокод: FADD5 S, D, MR,rs,rd

FADD5 S, D, MRL,RS,RD

Формат 6: **FADD** S,D <R $\beta$ àRC>

Пример: **FADD** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	0	1	0	1	0				

Автокод: FADD6 S, D, MRC,rs,RC

FADD6 S, D, MRCd,RC,rd

Формат 7: **FADD**[.cc] S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **FADD** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: FADD7 S, D, MIL, cc, #32, RD

FADD7 S, D, MIC, cc, #16, RC

Формат 8а: <OP2> **FADD** S1, S2, D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **FADD** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	R				u	0	0	0	0	0	0	0	0	0	1	0	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> FADD S1, S2, D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **FADD** S1, S2, D <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **FADD** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	RS/Rs				0	0	0	0	0	0	0	0	0	0	1	0	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> FADD S1, S2, D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **FADD** S1, S2, D <R.LβàR.L>

Пример: LSRL.eq R6, R0, R8 **FADD** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1	D	S2	0	RS	0	0	0	0	0	0	0	0	0	1	0	1	0
----	---	----	---	----	---	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0	cc		1	0	#	OP2							

Автокод: PROL <OP2 cc> FADD S1, S2, D <R.LβàR.L>

Формат 8d: <OP2> FADD S1, S2, D <RβàRC>

Пример: LSRL R6, R0, R8 FADD R2, R4, R6 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	Rs/Rd				0	0	0	0	0	0	0	0	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> FADD S1, S2, D <RβàRC>

**FAS**
**Сложение-вычитание (24E8)**
**FAS**
**Операция:**
3-адресный вариант: float S1, S2, D;

 $S1 + S2 \rightarrow D$ 
 $S2 - S1 \rightarrow S2$ 
2-адресный вариант: float S, D;

 $S + D \rightarrow D$ 
 $D - S \rightarrow S$ 

**Описание:** В формате 24E8 вычисляется сумма и разность операндов-источников (S2, S1 – в 3-адресном варианте; D, S – в 2-адресном варианте). Результат сложения помещается по адресу D в 3-адресном варианте, по адресу S в 2-адресном варианте. Результат вычитания помещается по адресу S2 в 3-адресном варианте по адресу D в 2-адресном варианте.

**Тип:**

OP1

**Форматы:**

1, 4, 5, 6, 8c, 8d

**Признаки результата:**

U	N	Z	V	C
–	√	√	√	–

**Синтаксис ассемблера и код инструкции:**
Формат 1: **FAS**[.cc] S1, S2, D

Пример: **FAS**.ne R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	0	0	1	0	1	1				

Автокод: FAS cc, S1, S2, D

Формат 4: **FAS** S, D <XRAM $\beta$ àR.L>



Пример: **FAS** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	0	0	0	1	0	1	1	

Автокод: FAS4 S, D, MLR,ea,A,R  
 FAS4 S, D, MLM,ea,A,R

Формат 5: **FAS** S,D <R/R.L**В**àR/R.L>

Пример: **FAS** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	1	0	1	1			

Автокод: FAS5 S, D, MR,rs,rd  
 FAS5 S, D, MRL,RS,RD

Формат 6: **FAS** S,D <R**В**àRC>

Пример: **FAS** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	0	1	0	1	1				

Автокод: FAS6 S, D, MRC,rs,RC  
 FAS6 S, D, MRCd,RC,rd

Формат 8с: <OP2.сс> **FAS** S1,S2,D <R.L**В**àR.L>

Пример: LSRL.eq R6,R0,R8 **FAS** R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	RS				0	0	0	0	0	0	0	0	0	0	1	0	1	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3/s3/#5	D2/d2	S4/s4	cc	RD	0	cc	1	0	#	OP2
----------	-------	-------	----	----	---	----	---	---	---	-----

Автокод: PROL <OP2 cc> FAS S1, S2, D <R.LβàR.L>

Формат 8d: <OP2> FAS S1, S2, D <RβàRC>

Пример: LSRL R6, R0, R8 FAS R2, R4, R6 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	Rs/Rd				0	0	0	0	0	0	0	1	0	1	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> FAS S1, S2, D <RβàRC>

**FIN** Первое приближение к обратной величине **FIN**

**Операция:** float S,D; (1.0/S) à D

**Описание:** Первое приближение к обратной величине:

$$D = 2^{-E}, \text{ если } f = 0;$$

$$D = 2^{-(E+1)} \cdot (2.0 / (1.0 + f)), \text{ если } f \neq 0;$$

$$D = 0, \text{ если } S = +\infty / -\infty.$$

Варианты обработки экспоненты

№	e	F	e результата
1	x	f=0	-e+254 = ~e + 254 + 1
2	x	f ≠ 0	-e+253 = ~e + 254

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7

**Признаки результата:**

U	N	Z	V	C
-	√	√	*	-

\*V=1, если S=0/QNaN.

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FIN[.cc] S,D**

*Пример:* **FIN.ne R6,R12**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D								0	0	0	0	0	0	cc				0	0	0	1	0	0	1	1	1	1	0

*Автокод:* **FIN cc, S, D**

Формат 4: **FIN S,D <XRAMBàR.L>**

*Пример:* **FIN R1,R2 R8,(A0)+**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D								R				A	mode	u	0	1	de	0	0	0	1	1	1	1	0		

Автокод: FIN4 S, D, MLR,ea,A,R  
 FIN4 S, D, MLM,ea,A,R

Формат 5: **FIN** S,D <R/R.L~~В~~R/R.L>

Пример: **FIN** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	1	1	0				

Автокод: FIN5 S, D, MR,rs,rd  
 FIN5 S, D, MRL,RS,RD

Формат 6: **FIN** S,D <R~~В~~RC>

Пример: **FIN** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	1	1	1	1	0				

Автокод: FIN6 S, D ,MRC,rs,RC  
 FIN6 S, D ,MRCd,RC,rd

**FINR** Первое приближение к обратной величине квадратного корня  
**FINR**

**Операция:** float S,D; (1.0/√S) → D

**Описание:** Первое приближение к обратной величине квадратного корня:

$D = 2^{-E/2}$ , если  $f=0$  и  $E$  - четное;

$D = 2^{-(E+1)/2} * (\sqrt{2.0})$ , если  $f = 0$  и  $E$  - нечетное;

$D = 2^{-(E/2+1)} * (2.0 / \sqrt{(1.0 + f)})$ , если  $f \neq 0$  и  $E$ - четное;

$D = 2^{-(E+1)/2} * \sqrt{(2.0 / (1.0 + f))}$ , если  $f \neq 0$  и  $E$  - нечетное;

$D = 0$ , если  $S = +\infty$ .

Варианты обработки экспоненты

№	e	f	e результата
1	нечетное	$f=0$	$(-e-1) \gg 1 + 190 = \sim e \gg 1 + 191$
2	четное	$f=0$	$(-e) \gg 1 + 191 = \sim e \gg 1 + 191$
3	нечетное	$f \neq 0$	$(-e-1) \gg 1 + 192 = \sim e \gg 1 + 191$
4	четное	$f \neq 0$	$(-e) \gg 1 + 191 = \sim e \gg 1 + 191$

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7

**Признаки результата:**

U	N	Z	V	C
-	√	√	*	-

\*V=1, если  $S=0/S<0/QNaN$ .

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FINR**[.cc] S,D

Пример: **FINR.ne** R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	0	0	0	0	0	0	0	cc	0	0	0	1	0	0	1	1	1	1	1
---	---	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---	---	---	---	---

Автокод: FINR cc, S, D

Формат 4: **FINR** S,D <XRAM↔R.L>

Пример: **FINR** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	0	1	1	1	1	1	1		

Автокод: FINR4 S, D, MLR,ea,A,R

FINR4 S, D, MLM,ea,A,R

Формат 5: **FINR** S,D <R/R.L↔R/R.L>

Пример: **FINR** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	1	1	1				

Автокод: FINR5 S, D, MR,rs,rd

FINR5 S, D, MRL,RS,RD

Формат 6: **FINR** S,D <R↔RC>

Пример: **FINR** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	1	1	1	1	1				

Автокод: FINR6 S, D ,MRC,rs,RC

FINR6 S, D ,MRCd,RC,rd

### FINT Округление к ближайшему целому (24E8) FINT

**Операция:** float S,D; (S)<sub>Round</sub> → D

**Описание:** Операнд S в формате с плавающей точкой 24E8 округляется к целому значению, представленному также в формате 24E8; результат помещается по адресу D. Способ округления мантиссы - к ближайшему числу, а при равноудаленности – к четному.

Примеры исполнения операции FINT S,D

№	S	D	№	S	D	№	S	D	№	S	D
1	-2.75	-3.0	5	-1.75	-2.0	9	1.75	2.0	13	2.75	3.0
2	-2.5	-2.0	6	-1.5	-2.0	10	1.5	2.0	14	2.5	2.0
3	-2.25	-2.0	7	-1.25	-1.0	11	1.25	1.0	15	2.25	2.0
4	-2.0	-2.0	8	-1.0	-1.0	12	1.0	1.0	16	2.0	2.0

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	√	√	*	-

\*V = 1, если D = ±∞/QNaN/ NAN.

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FINT**[.cc] S,D

Примеры: **FINT.ne** R2,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	cc				0	0	0	1	0	0	0	1	1	0	1			

Автокод: **FINT** cc, S, D

Формат 4: **FINT** S,D <XRAMB → R.L>

Пример: **FINT** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	0	0	0	1	1	0	1	

Автокод: FINT4 S, D, MLR, ea, A, R

FINT4 S, D, MLM, ea, A, R

Формат 5: **FINT** S, D <R/R.L $\beta$ àR/R.L>

Пример: **FINT** R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	0	1	1	0	1		

Автокод: FINT5 S, D, MR, rs, rd

FINT5 S, D, MRL, RS, RD

Формат 6: **FINT** S, D <R $\beta$ àRC>

Пример: **FINT** R2, R4 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	0	0	0	1	1	0	1		

Автокод: FINT6 S, D, MRC, rs, RC

FINT6 S, D, MRCd, RC, rd

Формат 7: **FINT**[.cc] S, D <#16/32 $\beta$ àRC/R/R.L>

Пример: **FINT** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	1	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: FMPY7 S, D, MIL, cc, #32, RD



FMPY7 S, D, MIC, cc, #16, RC

Формат 8а: <OP2> **FINT** S, D <XRAM $\beta$ àR.L> <YRAMàR0>

Пример: LSRL R5, R0, R8 **FINT** R2, R4 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	0	0	0	0	0	1	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2										

Автокод: PROL <OP2> **FINT** S, D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **FINT** S, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R5, R0, R8 **FINT** R2, R4 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS/Rs				0	0	0	0	0	0	0	0	1	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd			L	0	0	1	#	OP2										

Автокод: PROL <OP2> **FINT** S, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **FINT** S, D <R.L $\beta$ àR.L>

Пример: LSRL.eq R5, R0, R8 **FINT** R2, R4 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS				0	0	0	0	0	0	0	0	0	0	1	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD			0	cc	1	0	#	OP2										

Автокод: PROL <OP2 cc> **FINT** S, D <R.L $\beta$ àR.L>

Формат 8d: <OP2> **FINT** S,D <RβàRC>

Пример: LSRL R5,R0,R8 **FINT** R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	Rs/Rd				0	0	0	0	0	0	0	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> FINT S,D <RβàRC>

FLOOR Округление к ближайшему целому (24E8) FLOOR

**Операция:** float S,D; (S)<sub>Round</sub> → D

**Описание:** Операнд S в формате с плавающей точкой 24E8 округляется к ближайшему целому значению снизу, представленному также в формате 24E8; результат помещается по адресу D.

Примеры исполнения операции FLOOR S,D

N	S	D	N	S	D	N	S	D	N	S	D
1	-2.75	-3.0	5	-1.75	-2.0	9	1.75	1.0	13	2.75	2.0
2	-2.5	-3.0	6	-1.5	-2.0	10	1.5	1.0	14	2.5	2.0
3	-2.25	-3.0	7	-1.25	-2.0	11	1.25	1.0	15	2.25	2.0
4	-2.0	-2.0	8	-1.0	-1.0	12	1.0	1.0	16	2.0	2.0

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
–	√	√	*	–

\*V = 1, если D = ±∞/QNaN/ NAN.

**Синтаксис ассемблера и код инструкции:**

Формат 1: FLOOR[.cc] S,D

*Примеры:* FLOOR.ne R2,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	0	1	0	0	0	1	1	0	0		

*Автокод:* FLOOR cc, S, D

Формат 4: FLOOR S,D <XRAMβàR.L>

*Пример:* FLOOR R2,R4 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	R	A	mode	u	0	1	de	0	0	0	0	1	1	0	0
---	---	---	---	------	---	---	---	----	---	---	---	---	---	---	---	---

Автокод: FLOOR4 S, D, MLR, ea, A, R  
 FLOOR4 S, D, MLM, ea, A, R

Формат 5: **FLOOR** S, D <R/R.L $\bar{a}$ R/R.L>

Пример: **FLOOR** R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	1	1	0	0				

Автокод: FLOOR5 S, D, MR, rs, rd  
 FLOOR5 S, D, MRL, RS, RD

Формат 6: **FLOOR** S, D <R $\bar{a}$ RC>

Пример: **FLOOR** R2, R4 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	0	1	1	0	0				

Автокод: FLOOR6 S, D, MRC, rs, RC  
 FLOOR6 S, D, MRCd, RC, rd

Формат 7: **FLOOR**[.cc] S, D <#16/32 $\bar{a}$ RC/R/R.L>

Пример: **FLOOR** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: FMPY7 S, D, MIL, cc, #32, RD  
 FMPY7 S, D, MIC, cc, #16, RC

Формат 8а: <OP2> **FLOOR** S, D <XRAM $\bar{a}$ R/R.L> <YRAM $\bar{a}$ R0>

Пример: LSRL R5, R0, R8 **FLOOR** R2, R4 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	0	0	0	0	0	1	1	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode		A	de	0	0	#	OP2											

Автокод: PROL <OP2> FLOOR S,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **FLOOR** S,D <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 **FLOOR** R2,R4 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS/Rs				0	0	0	0	0	0	0	0	1	1	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd		L	0	0	1	#	OP2											

Автокод: PROL <OP2> FLOOR S,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **FLOOR** S,D <R.LβàR.L>

Пример: LSRL.eq R5,R0,R8 **FLOOR** R2,R4 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS				0	0	0	0	0	0	0	0	0	0	1	1	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD		0	cc	1	0	#	OP2											

Автокод: PROL <OP2 cc> FLOOR S,D <R.LβàR.L>

Формат 8d: <OP2> **FLOOR** S,D <RβàRC>

Пример: LSRL R5,R0,R8 **FLOOR** R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	0	0	0	0	0	0	Rs/Rd	0	0	0	0	0	0	0	0	1	1	0	0
---	---	---	---	---	---	---	---	-------	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> FLOOR S, D <R3àRC>

**FMPY**
**Умножение (24E8)**
**FMPY**
**Операция:**

3-адресный вариант: float S1,S2,D;      S1 \* S2 → D

2-адресный вариант: float S,D;      S \* D → D

**Описание:** Операнд-источник S1 (в 3-адресном варианте) или S (в 2-адресном варианте) умножается на S2 (в 3-адресном варианте) или D (в 2-адресном варианте) и результат помещается в операнд-приемник D. Исходные операнды и результат представлены в формате 24E8. Способ округления произведения мантисс - к ближайшему числу, а при равно удаленности – к четному.

В качестве первого операнда-источника S1 может использоваться непосредственный операнд #32 (формат 2).

**Тип:** OP2

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	√	√	*	-

\*V = 1, если D = ±∞/QNaN/ NAN.

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FMPY[.cc]** S1,S2,D

Пример: **FMPY.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	1	1	0	1	1	1	1				

Автокод: FMPY cc, S1, S2, D

Формат 4: **FMPY** S,D <XRAM $\beta$ àR.L>

Пример: **FMPY** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	1	1	0	1	1	0	1	1	1	1

Автокод: FMPY4 S, D, MLR,ea,A,R

FMPY4 S, D, MLM,ea,A,R

Формат 5: **FMPY** S,D <R/R.L $\beta$ àR/R.L>

Пример: **FMPY** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	1	0	1	1	1	1	1			

Автокод: FMPY5 s, D, MR,rs,rd

FMPY5 s, D, MRL,RS,RD

Формат 6: **FMPY** S,D <R $\beta$ àRC>

Пример: **FMPY** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	1	0	1	1	1	1	1			

Автокод: FMPY6 S, D ,MRC,rs,RC

FMPY6 S, D ,MRCd,RC,rd

Формат 7: **FMPY**[.cc] S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **FMPY** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	1	1	1	1	1		



63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: FMPY7 S, D, MIL, cc, #32, RD

FMPY7 S, D, MIC, cc, #16, RC

Формат 8a: **FMPY** S3, S4, D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Пример: **FMPY** R4, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	0	0	1	1	1	1					

Автокод: PROL FMPY S3, S4, D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: **FMPY** S3, S4, D2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: **FMPY** R4, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	RD/Rd				L	0	0	1	0	0	1	1	1	1					

Автокод: PROL FMPY s3/#5, S4, D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: **FMPY**[.cc] S3, S4, D2 <OP1> <R.L $\beta$ àR.L>

Пример: **FMPY**.eq R4, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3	D2	S4	cc	RD	0	cc	1	0	0	0	1	1	1	1
----	----	----	----	----	---	----	---	---	---	---	---	---	---	---

Автокод: PROL FMPY cc S3, S4, D2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: **FMPY** S3, S4, D2 <OP1> <R $\beta$ àRC>

Пример: **FMPY** R4, R2, R6 ANDL R6, R0, R8 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	0	0	1	1	1	1				

Автокод: PROL FMPY S3, S4, D2 <OP1> <R $\beta$ àRC>

**FSUB**
**Вычитание (24E8)**
**FSUB**
**Операция:**

3-адресный вариант: float S1,S2,D;      S2 - S1 → D

2-адресный вариант: float S,D;      D - S → D

**Описание:** В формате 24E8 выполняется вычитание первого операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) из второго операнда-источника (S2 - в 3-адресном варианте; D - в 2-адресном варианте), результат помещается в операнд-приемник D. В качестве первого операнда-источника S1 может использоваться непосредственный операнд #32 (формат 2).

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	√	√	√	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FSUB[.cc]** S1,S2,D

Пример: **FSUB.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	0	1	1	0	1				

Автокод:      FSUB cc, S1, S2, D

Формат 2: **FSUB[.cc]** #32,S2,D

Пример: **FSUB.ne** 15,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1	D	S2	0	0	cc	0	0	1	0	0	1	0	1	1	0	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: FSUB2 cc, #32, S2, D

Формат 4: **FSUB** S,D <XRAM $\beta$ àR.L>

Пример: **FSUB** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	0	1	1	0	1		

Автокод: FSUB4 S, D, MLR,ea,A,R

FSUB4 S, D, MLM,ea,A,R

Формат 5: **FSUB** S,D <R/R.L $\beta$ àR/R.L>

Пример: **FSUB** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	1	1	0	1				

Автокод: FSUB5 S, D, MR,rs,rd

FSUB5 S, D, MRL,RS,RD

Формат 6: **FSUB** S,D <R $\beta$ àRC>

Пример: **FSUB** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	0	1	1	0	1				

Автокод: FSUB6 S, D, MRC,rs,RC

FSUB6 S, D, MRCd,RC,rd

Формат 7: **FSUB**[.cc] S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **FSUB** R2,R0 0x12345678,R16.L

Код инструкции:

3	3	2	2	2	2	2	2	2	2	2	1	1	1	16	15	1	1	1	1	10	9	8	7	6	5	4	3	2	1	0
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7			4	3	2	1										
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	1	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: FSUB7 S, D, MIL, cc, #32, RD

FSUB7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **FSUB** S1, S2, D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **FSUB** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	R				u	0	0	0	0	0	1	0	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> FSUB S1, S2, D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **FSUB** S1, S2, D <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **FSUB** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	RS/Rs				0	0	0	0	0	1	0	1	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> FSUB S1, S2, D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **FSUB** S1, S2, D <R.LβàR.L>

Пример: LSRL.eq R6, R0, R8 **FSUB** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	RS				0	0	0	0	0	0	1	0	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD		0	cc		1	0	#	OP2									

Автокод: PROL <OP2 cc> FSUB S1, S2, D <R.L $\beta$ àR.L>

Формат 8d: <OP2> **FSUB** S1, S2, D <R $\beta$ àRC>

Пример: LSRL R6, R0, R8 **FSUB** R2, R4, R6 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	Rs/Rd				0	0	0	0	0	1	0	1	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC		sc	de	1	1	#	OP2										

Автокод: PROL <OP2> FSUB S1, S2, D <R $\beta$ àRC>

**FTR                      Преобразование формата                      FTR**

**Операция:** short s; long D; {16{s[15]},s[15:0]} → D[31:0]

**Описание:** Исходный операнд s преобразуется из формата short в формат long путем расширения знака. Результат помещается в операнд-приемник D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	–	–

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FTR**[.cc] S,D

*Пример:* **FTR.ne** R0,R4

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	0	cc				0	0	0	1	0	0	0	0	1	1	0	

*Автокод:* FTR cc, S1, D

Формат 4: **FTR** S,D <XRAM**β**àR.L>

*Пример:* **FTR** R0,R2 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	0	0	0	0	1	1	0		

*Автокод:* FTR4 S, D, MLR,ea,A,R

FTR4 S, D, MLM,ea,A,R

Формат 5: **FTR** S,D <R/R.L**β**àR/R.L>

*Пример:* **FTR** R4,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	0	0	1	1	0		

Автокод: FTR5 S, D, MR,rs,rd  
FTR5 S, D, MRL,RS,RD

Формат 6: **FTR** S,D <R**β**RC>

Пример: **FTR** R8,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	0	0	0	1	1	0			

Автокод: FTR6 S, D ,MRC,rs,RC  
FTR6 S, D ,MRCd,RC,rd

Формат 7: **FTR**[.cc] S,D <#16/32**β**RC/R/R.L>

Пример: **FTR** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	0	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: FTR7 S, D, MIL,cc,#32,RD  
FTR7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **FTR** S,D <XRAM**β**R.L> <YRAM**β**R0>

Пример: LSRL R5,R0,R8 **FTR** R0,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	0	0	0	0	0	0	1	1	0



63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de	0	0	#	OP2								

Автокод: PROL <OP2> FTR S,D <XRAM $\beta$ àR.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> FTR S,D <R/R.L $\beta$ àR/R.L> <YRAM $\alpha$ R0>

Пример: LSRL R5,R0,R8 FTR R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
S				D				0				0				0				0				RS/Rs				0	0	0	0	0	0	0	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> FTR S,D <R/R.L $\beta$ àR/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> FTR S,D <R.L $\beta$ àR.L>

Пример: LSRL.eq R5,R0,R8 FTR R4,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
S				D				0				0				0				RS				0	0	0	0	0	0	0	0	0	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				Cc		RD			0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> FTR S,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> FTR S,D <R $\beta$ àRC>

Пример: LSRL R5,R0,R8 FTR R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
S				D				0				0				0				Rs/Rd				0	0	0	0	0	0	0	0	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3/s3/#5	D2/d2	S4/s4	0	0	RC	sc	de	1	1	#	OP2
----------	-------	-------	---	---	----	----	----	---	---	---	-----

Автокод: PROL <OP2> FTR S,D <R**В**à RC>

**FTRFL          Преобразование формата          FTRFL**
**Операция:**    short s, long D;    { s[15:0], 16{0} } **à** D

**Описание:** Операнд s преобразуется из формата short в формат long. Исходный 16-разрядный операнд помещается в старшие 16 разрядов приемника D, младшие 16 разрядов приемника D заполняются нулями.

**Тип:**                                  OP1

**Форматы:**                          1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**
Формат 1: **FTRFL**[.cc] S,D

*Пример:*    **FTRFL**.ne R0,R4

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					D					0	0	0	0	0	0	0	cc		0	0	0	1	0	0	0	1	0	0	1		

*Автокод:*        **FTRFL** cc, S1, D

Формат 4: **FTRFL** S,D <XRAM**β**à R.L>

*Пример:*    **FTRFL** R0,R2      R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					D					R		A		mode		u	0	1	de	0	0	0	0	1	0	0	1				

*Автокод:*        **FTRFL4** S, D, MLR, ea, A, R

**FTRFL4** S, D, MLM, ea, A, R

Формат 5: **FTRFL** S,D <R/R.L**β**à R/R.L>

Пример: **FTRFL** R4,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	1	0	0	1			

Автокод: FTRFL5 S, D, MR,rs,rd

FTRFL5 S, D, MRL,RS,RD

Формат 6: **FTRFL** S,D <RβàRC>

Пример: **FTRFL** R8,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	0	0	1	0	0	1				

Автокод: FTRFL6 S, D ,MRC,rs,RC

FTRFL6 S, D ,MRCd,RC,rd

Формат 7: **FTRFL**[.cc] S,D <#16/32àRC/R/R.L>

Пример: **FTRFL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: FTRFL7 S, D, MIL,cc,#32,RD

FTRFL7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **FTRFL** S,D <XRAMβàR.L>

<YRAMàR0>

Примеры: LSRL R5,R0,R8 **FTRFL** R0,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	0	0	0	0	0	0	R	u	0	0	0	0	0	0	0	0	1	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT		mode			A		de		0		0		#		OP2	

Автокод: PROL <OP2> FTRFL S,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> FTRFL S,D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 FTRFL R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																					
S					D					0					0					0					0					RS/Rs					0					0					0					0					0					0					1					0					0					1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd			L		0		0		1		#		OP2	

Автокод: PROL <OP2> FTRFL S,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> FTRFL S,D <R.LβàR.L>

Примеры: LSRL.eq R5,R0,R8 FTRFL R4,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																					
S					D					0					0					0					0					RS					0					0					0					0					0					0					1					0					0					1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					Cc		RD			0		cc		1		0		#		OP2	

Автокод: PROL <OP2 cc> FTRFL S,D <R.LβàR.L>

Формат 8d: <OP2> FTRFL S,D <RβàRC>

Примеры: LSRL R5,R0,R8 FTRFL R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																
S					D					0					0					0					0					Rs/Rd					0					0					0					0					0					1					0					0					1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> FTRFL S, D <R3àRC>

***FTRL*                      Преобразование формата                      *FTRL***

**Операция:**

long S; short d; S[31:16] + round(S[15:0]) → d[15:0]

**Описание:** Исходный операнд S преобразуется из формата long в формат short путем округления младших 16-ти разрядов. Округление выполняется в соответствии с режимом округления, определяемым битом RND (разряд 9 регистра CCR). Старшие 16 разрядов округленного числа запоминаются в регистре приемника d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FTRL[.cc] S,D**

*Пример:* **FTRL.ne R0,R4**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D								0	0	0	0	0	0	cc				0	0	0	1	0	1	0	0	1	1	0

*Автокод:* **FTRL cc, S1, D**

Формат 4: **FTRL S,D <XRAM~~3~~àR.L>**

*Пример:* **FTRL R0,R2 R8,(A0)+**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D								R				A				mode	u	0	1	de	0	0	1	0	0	1	1	0

*Автокод:* **FTRL4 S, D, MLR,ea,A,R**

**FTRL4 S, D, MLM,ea,A,R**

Формат 5: **FTRL** S,D <R/R.L $\beta$ àR/R.L>

Пример: **FTRL** R4,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	0	0	1	1	0			

Автокод: **FTRL5** S, D, MR,rs,rd

**FTRL5** S, D, MRL,RS,RD

Формат 6: **FTRL** S,D <R $\beta$ àRC>

Пример: **FTRL** R8,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	0	0	1	1	0				

Автокод: **FTRL6** S, D ,MRC,rs,RC

**FTRL6** S, D ,MRCd,RC,rd

Формат 7: **FTRL[.cc]** S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **FTRL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **FTRL7** S, D, MIL,cc,#32,RD

**FTRL7** S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **FTRL[s]** S,D <XRAM $\beta$ àR.L> <YRAM $\beta$ àR0>

Примеры: LSRL R5,R0,R8 **FTRL** R0,R2 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **FTRLS** R0,R2 R8,(A0)+ (AT),R0

Код инструкции:



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	0	0	1	0	0	0	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2										

Автокод: PROL <OP2> FTRL[s] S,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> FTRL[s] S,D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 FTRL R0,R2 R8,R15 (AT),R0

LSRL R5,R0,R8 FTRLS R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS/Rs				0	0	0	0	0	0	1	0	0	0	1	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd			L	0	0	1	#	OP2										

Автокод: PROL <OP2> FTRL[s] S,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> FTRL[s] S,D <R.LβàR.L>

Примеры: LSRL.eq R5,R0,R8 FTRL R4,R2 R8.L,R16.L

LSRL.eq R5,R0,R8 FTRLS R4,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS				0	0	0	0	0	0	0	1	0	0	0	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				Cc	RD			0	cc	1	0	#	OP2										

Автокод: PROL <OP2 cc> FTRL[s] S,D <R.LβàR.L>

Формат 8d: <OP2> FTRL[s] S,D <RβàRC>

Примеры: LSRL R5,R0,R8 FTRL R0,R2 CCR,R17

LSRL R5,R0,R8 FTRLS R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	Rs/Rd				0	0	0	0	0	1	0	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL &lt;OP2&gt; FTRL[s] S,D &lt;RβàRC&gt;

**FTST Определение признаков операнда (24E8) FTST**

**Операция:** float S; {N,Z,V} à CCR[3:1]

**Описание:** Определяются признаки операнда S в формате с плавающей точкой (24E8) и записываются в регистр CCR.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
–	√	√	*	–

\*V=1, если S=±∞/QNaN/NaN.

**Синтаксис ассемблера и код инструкции:**

Формат 1: **FTST[.cc] S**

*Пример:* **FTST.ne R0**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	0	cc				0	0	0	1	0	1	0	1	1	0	0			

*Автокод:* FTST cc, S

Формат 4: **FTST S <XRAMβàR.L>**

*Пример:* **FTST R0 R8, (A0)+**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				R				A				mode	u	0	1	de	0	0	1	0	1	1	0	0			

*Автокод:* FTST4 S, MLR, ea, A, R

FTST4 S, MLM, ea, A, R

Формат 5: **FTST S <R/R.LβàR/R.L>**

Пример: **FTST** R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	0	1	1	0	0			

Автокод: FTST5 S, MR,rs,rd  
 FTST5 S, MRL,RS,RD

Формат 6: **FTST** S <R**B**àRC>

Пример: **FTST** R8 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				R				sc	RC				1	0	de	0	0	1	0	1	1	0	0				

Автокод: FTST6 S, MRC,rs,RC  
 FTST6 S, MRCd,RC,rd

Формат 7: **FTST**[.cc] S <#16/32**B**àRC/R/R.L>

Пример: **FTST** R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: FTST7 S, MIL,cc,#32,RD  
 FTST7 S, MIC,cc,#16,RC

Формат 8а: <OP2> **FTST** S <XRAM**B**àR.L> <YRAM**B**àR0>

Пример: LSRL R5,R0,R8 **FTST** R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	R				u	0	0	0	0	0	0	1	0	1	1	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> FTST S <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> FTST S <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Пример: LSRL R5,R0,R8 FTST R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	RS/Rs				0	0	0	0	0	0	1	0	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> FTST S <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> FTST s <R.L $\beta$ R.L>

Пример: LSRL.eq R5,R0,R8 FTST R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	RS				0	0	0	0	0	0	1	0	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> FTST S <R.L $\beta$ R.L>

Формат 8d: <OP2> FTST S <R $\beta$ RC>

Пример: LSRL R5,R0,R8 FTST R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	Rs/Rd				0	0	0	0	0	1	0	1	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

АВТОКОД: PROL <OP2> FTST S <Rβà RC>

**INC**
**Инкремент (short)**
**INC**
**Операция:** short s,d; s + 1 à d

**Описание:** В формате short к операнду-источнику s прибавляется единица, результат помещается в операнд-приемник d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** INC[.cc] s,d

**Пример:** INC.ne R1,R5

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	0	cc				0	0	0	1	0	0	0	0	0	0	1	0	

**Автокод:** INC cc, s1, d

**Формат 4:** INC s,d <XRAM $\beta$ àR.L>

**Пример:** INC R1,R2 R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode	u	0	1	de	0	0	0	0	0	0	0	0	1	0	

**Автокод:** INC4 s, d, MLR,ea,A,R

INC4 s, d, MLM,ea,A,R

**Формат 5:** INC s,d <R/R.L $\beta$ àR/R.L>

**Пример:** INC R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	0	0	0	0	0	1	0

Автокод: INC5 s, d, MR,rs,rd  
INC5 s, d, MRL,RS,RD

Формат 6: **INC** s,d <R**B**àRC>

Пример: **INC** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	0	0	0	0	0	0	1	0	

Автокод: INC6 s, d ,MRC,rs,RC  
INC6 s, d ,MRCd,RC,rd

Формат 7: **INC[.cc]** s,d <#16/32àRC/R/R.L>

Пример: **INC** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	0	0	0	0	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: INC7 s, d, MIL,cc,#32,RD  
INC7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **INC[s]** s,d <XRAM**B**àR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **INC** R1,R2 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **INC**s R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				0	0	0	0	0	M	R				u	0	0	0	0	0	0	0	0	0	0	0	0	1	0



63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de	0	0	#	OP2								

Автокод: PROL <OP2> INC[s] s,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> INC[s] s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 INC R1,R2 R8,R15 (AT),R0

LSRL R5,R0,R8 INCs R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	M	RS/Rs				0	0	0	0	0	0	0	0	0	0	0	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> INC[s] s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> INC[s] s,d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 INC R1,R2 R8.L,R16.L

LSRL.eq R5,R0,R8 INCs R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	M	RS				0	0	0	0	0	0	0	0	0	0	0	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2								

Автокод: PROL <OP2 cc> INC[s] s,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> INC[s] s,d <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 INC R1,R2 CCR,R17

LSRL R5,R0,R8 INCs R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s	d	0	0	0	0	0	0	M	Rs/Rd	0	0	0	0	0	0	0	0	0	0	1	0
---	---	---	---	---	---	---	---	---	-------	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: PROL      <OP2>      INC[s]      s,d      <R**B**aRC>

**INCL**
**Инкремент (long)**
**INCL**
**Операция:** long S,D; S + 1 à D

**Описание:** В формате long к операнду-источнику S прибавляется единица, результат помещается в операнд-приемник D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** **INCL**[.cc] S,D

**Пример:** **INCL**.ne R2,R6

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	0	1	0	1	0	0	0	1	0		

**Автокод:** INCL cc, S, D

**Формат 4:** **INCL** S,D <XRAMβàR.L>

**Пример:** **INCL** R2,R4 R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	0	0	0	0	1	0		

**Автокод:** INCL4 S, D, MLR,ea,A,R

INCL4 S, D, MLM,ea,A,R

**Формат 5:** **INCL** S,D <R/R.LβàR/R.L>

**Пример:** **INCL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	0	0	0	0	1	0		

Автокод: INCL5 S, D, MR,rs,rd  
INCL5 S, D, MRL,RS,RD

Формат 6: **INCL** S,D <RβàRC>

Пример: **INCL** R2,R4 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	0	0	0	0	1	0			

Автокод: INCL6 S, D ,MRC,rs,RC  
INCL6 S, D ,MRCd,RC,rd

Формат 7: **INCL[.cc]** S,D <#16/32àRC/R/R.L>

Пример: **INCL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	0	0	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: INCL7 S, D, MIL,cc,#32,RD  
INCL7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **INCL[s]** S,D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 **INCL** R2,R4 R8,(A0)+ (AT),R0

LSRL R6,R0,R8 **INCLs** R2,R4 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	0	0	0	0	0	0	M	R	u	0	0	0	0	0	0	1	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de		0		0		#		OP2				

Автокод: PROL <OP2> INCL[s] S,D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> INCL[s] S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 INCL R2,R4 R8,R15 (AT),R0

LSRL R6,R0,R8 INCLs R2,R4 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S		D				0	0	0	0	0	M	RS/Rs				0	0	0	0	0	0	1	0	0	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L		0		0		1		#		OP2				

Автокод: PROL <OP2> INCL[s] S,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> INCL[s] S,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6,R0,R8 INCL R2,R4 R8.L,R16.L

LSRL.eq R6,R0,R8 INCLs R2,R4 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S		D				0	0	0	0	0	M	RS				0	0	0	0	0	0	0	1	0	0	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0		cc		1		0		#		OP2				

Автокод: PROL <OP2 cc> INCL[s] S,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> INCL[s] S,D <R $\beta$ àRC>

Примеры: LSRL R6,R0,R8 INCL R2,R4 CCR,R17

LSRL R6,R0,R8 INCLs R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	M	Rs/Rd				0	0	0	0	0	1	0	0	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> INCL[s] S,D <R**B**àRC>

**INSL**
**Побитное мультиплексирование**
**INSL**
**Операция:** `long S1,S2,D; (~S2 & S1) | ( S2 & D) à D`

**Описание:** Содержимое регистра S2 выполняет роль маски. В разряды регистра приемника D, соответствующие единичным значениям маски, заносятся значения разрядов регистра источника S1.

**Тип:** OP1

**Форматы:** 1

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	–

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** `INSL[.cc] S1,S2,D`
**Пример:** `INSL.ne R2,R6,R12`
**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	0	1	1	0	1	0	1	0	0			

**Автокод:** `INSL cc, S1, S2, D`

**J                      Программный переход                      J**

**Операция:** Программный переход (**Jump**) одним из способов:

1) по метке или непосредственному значению: `if(cc) #16 à PC;`

2) по адресному регистру: `if(cc) An à PC;`

**Описание:** Если специфицированное условие истинно, происходит программный переход по адресу перехода. Если условие ложно, продолжается последовательное выполнение программы.

Адрес перехода может быть задан одним из следующих способов:

- 1) при помощи метки или непосредственного значения #16 (прямой способ),
- 2) путем указания адресного регистра An, хранящего адрес перехода (косвенный способ).

Значение адреса перехода является целым числом без знака.

**Тип:** OP1

**Форматы:** 3m, 3mb

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
–	–	–	–	–

**Синтаксис ассемблера и код инструкции:**

Формат 3m (Ветвление по метке или непосредственному значению):  
**J[.cc] #16**

Примеры: **J.ne label\_1**

**J 0x17**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
cc				1	#16												0	0	1	1	0	0	1	1	1	0	1				



Автокод: JccI cc, #16

Jccm cc, label

Формат 3mb (Ветвление по адресному регистру): J[.cc] An

Пример: **J.eq** A7

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
cc				0	0	0	0	0	0	0	0	0	A				0	0	0	0	0	0	1	1	0	0	1	1	1	0	1

Автокод: Jcc cc, An

***JD*** Программный переход (отложенный) ***JD***

**Операция:** Программный переход отложенный (**Jump Delayed**) одним из способов:

- 1) по метке или непосредственному значению: `if(cc) #16 à PC;`
- 2) по адресуному регистру: `if(cc) An à PC;`

**Описание:** Если специфицированное условие истинно, происходит программный переход по адресу перехода. Если условие ложно, продолжается последовательное выполнение программы.

Адрес перехода может быть задан одним из следующих способов:

- 1) при помощи метки или непосредственного значения #16 (прямой способ),
- 2) путем указания адресного регистра An, хранящего адрес перехода (косвенный способ).

Значение адреса перехода является целым числом без знака.

Отличием команды отложенного ветвления **JD** от обычной команды программного перехода **J** является то, что следующая за командой отложенного перехода инструкция выполняется всегда.

**Тип:** OP1  
**Форматы:** 3m, 3mb

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 3m (Ветвление по метке или непосредственному значению):  
**JD[.cc] #16**

**Примеры:** **JD.ne label\_1**  
**JD 0x17**

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

сс	1	#16	0	0	1	1	0	0	1	1	1	1	1
----	---	-----	---	---	---	---	---	---	---	---	---	---	---

Автокод: JDccI сс, #16

JDccm сс, label

Формат 3mb (Ветвление по адресному регистру): **JD[.сс] An**

Пример: **JD.eq A7**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
сс				0	0	0	0	0	0	0	0	0	0	0	A	0	0	0	0	0	0	0	1	1	0	0	1	1	1	1	1

Автокод: JDcc сс, An

*JS*
*Вызов подпрограммы*
*JS*

**Операция:** Вызов подпрограммы (**Jump to Subroutine**) одним из способов:

1) по метке или непосредственному значению:

```
if(cc) { PC + 1 à SS; SP[3:0] + 1 à SP[3:0]; #16 à PC; }
```

2) по адресному регистру:

```
if(cc) { PC + 1 à SS; SP[3:0] + 1 à SP[3:0]; An à PC; }
```

**Описание:** Если специфицированное условие истинно, происходит программный переход на подпрограмму по адресу перехода. Адрес следующей за командой перехода инструкции заносится в системный стек *SS*, указатель системного стека *SP[3:0]* инкрементируется. Если условие ложно, продолжается последовательное выполнение программы.

Адрес перехода может быть задан одним из следующих способов:

- 1) при помощи метки или непосредственного значения #16 (прямой способ),
- 2) путем указания адресного регистра *An*, хранящего адрес перехода (косвенный способ).

Значение адреса перехода является целым числом без знака.

**Тип:** OP1

**Форматы:** 3m, 3mb

**Признаки результата:**

U	N	Z	V	C
–	–	–	–	–

**Синтаксис ассемблера и код инструкции:**

*Формат 3m (Переход по метке или непосредственному значению):*  
**JS[.cc] #16**

*Примеры:* **JS.ne label\_1**

**JS 0x17**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
сс				1	#16																0	0	1	1	0	1	0	1	1	0	1

Автокод: JSccI сс, #16

JSccm сс, label

Формат 3mb (Переход по адресному регистру): **JS[.сс]** An

Пример: **JS.eq** A7

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
сс				0	0	0	0	0	0	0	0	0	0	0	A	0	0	0	0	0	0	0	1	1	0	1	0	1	1	0	1

Автокод: JScc сс, An

**LSL**
**Логический сдвиг влево (short)**
**LSL**
**Операция:**

3-адресный вариант: short s1,s2,d; (s2 << s1[5:0]) à d

2-адресный вариант: short s,d; (d << s[5:0]) à d

**Описание:** Операнд-источник s2 (в 3-адресном варианте) или d (в 2-адресном варианте) сдвигается влево на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте) и результат помещается в операнд-приемник d. Старшие десять разрядов s1 (или s) при этом должны быть установлены в «0». Вытолкнутые за пределы разрядной сетки разряды теряются, кроме последнего вытолкнутого, который запоминается в признаке C. Младшие освободившиеся в результате сдвига разряды заполняются нулями.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:**

OP2

**Форматы:**

1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **LSL[.cc]** s1/#5,s2,d

Примеры: **LSL.ne** R1,R5,R12

**LSL.ne** 17,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					d					s2					#	0	cc					0	0	0	1	1	1	0	0	0	0	1

*Автокод:* LSL cc, s1/#5, s2, d

Формат 4: **LSL** s/#5,d<XRAM $\beta$ àR.L>

Примеры: **LSL** R1,R2 R8,(A0)+

**LSL** 17,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				R				A				mode		u	0	1	de	#	1	1	0	0	0	0	1		

Автокод: LSL4 s, d, MLR,ea,A,R

LSL4 s, d, MLM,ea,A,R

Формат 5: **LSL** s,d <R/R.L $\beta$ àR/R.L>

Примеры: **LSL** R1,R2 R8.L,R12.L

**LSL** 15,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				RS/Rs				L	RD/Rd				1	1	0	#	1	1	0	0	0	0	1				

Автокод: LSL5 s, d, MR,rs,rd

LSL5 s, d, MRL,RS,RD

Формат 6: **LSL** s,d <R $\beta$ àRC>

Примеры: **LSL** R1,R5 R7,CCR

**LSL** 1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				d				R				sc	RC				1	0	de	#	1	1	0	0	0	1					

Автокод: LSL6 s, d ,MRC,rs,RC

LSL6 s, d ,MRCd,RC,rd

Формат 7: **LSL[.cc]** s,d <#16/32 $\beta$ àRC/R/R.L>

Пример: **LSL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	0	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: LSL7 s, d, MIL, cc, #32, RD

LSL7 s, d, MIC, cc, #16, RC

Формат 8a: **LSL** s3/#5, s4, d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: **LSL** R1, R2, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

**LSL** 1, R2, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	mode				A	de	0	0	#	0	0	0	0	1					

Автокод: PROL LSL s3/#5, s4, d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: **LSL** s3/#5, s4, d2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: **LSL** R1, R2, R5 ANDL R6, R0, R8 R8, R15 (AT), R0

**LSL** 1, R2, R5 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	RD/Rd				L	0	0	1	#	0	0	0	0	1					



Автокод: PROL LSL s3/#5,s4,d2 <OP2> <R/R.L**β**àR/R.L> <YRAMàR0>

Формат 8с: **LSL**[.cc] s3/#5,s4,d2 <OP1> <R.L**β**àR.L>

Примеры: **LSL**.eq R1,R2,R5 ANDL R6,R0,R8 R8.L,R16.L

**LSL**.eq 1,R2,R5 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				cc	RD				0	cc	1	0	#	0	0	0	0	0	0	0	0	1	

Автокод: PROL LSL cc s3,s4,d2 <OP1> <R.L**β**àR.L>

Формат 8d: **LSL** s3/#5,s4,d2<OP1> <R**β**àRC>

Примеры: **LSL** R1,R2,R5 ANDL R6,R0,R8 CCR,R17

**LSL** 1,R2,R5 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				0	0	RC				sc	de	1	1	#	0	0	0	0	0	0	0	1	

Автокод: PROL LSL s3/#5,s4,d2 <OP1> <R**β**àRC>

***LSLL***      **Логический сдвиг влево (*long*)**      ***LSLL***

**Операция:**

3-адресный вариант:

short s1;      long S2,D;      (S2 << s1[5:0])      à      D

2-адресный вариант: short s;

long D;      (D << s[5:0])      à      D

**Описание:** Операнд-источник S2 (в 3-адресном варианте) или D (в 2-адресном варианте) сдвигается влево на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте) и результат помещается в операнд-приемник D. Старшие десять разрядов s1 (или s) при этом должны быть установлены в «0». Вытолкнутые за пределы разрядной сетки разряды теряются, кроме последнего вытолкнутого, который запоминается в признаке C. Младшие освободившиеся в результате сдвига разряды заполняются нулями.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:**      OP2

**Форматы:**      1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	√

**Синтаксис ассемблера и код инструкции:**

Формат 1:      **LSLL[.cc]** s1/#5,S2,D

Примеры:      **LSLL.ne** R2,R6,R12

**LSLL.ne** 17,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					D					S2					#	0	cc					0	0	0	1	1	1	0	1	0	0	0

Автокод: LSL L cc, s1/#5, S2, D

Формат 4: LSL L s/#5, D <XRAM  $\beta$ à R.L>

Примеры: LSL L R2, R4 R8, (A0)+

LSL L 17, R4 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s/#5					D					R					A					mode	u	0	1	de	#	1	1	0	1	0	0	0

Автокод: LSL L4 s, D, MLR, ea, A, R

LSL L4 s, D, MLM, ea, A, R

Формат 5: LSL L s, D <R/R.L  $\beta$ à R/R.L>

Примеры: LSL L R2, R4 R8.L, R12.L

LSL L 15, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					RS/Rs					L	RD/Rd					1	1	0	#	1	1	0	1	0	0	0

Автокод: LSL L5 s, D, MR, rs, rd

LSL L5 s, D, MRL, RS, RD

Формат 6: LSL L S, D <R  $\beta$ à RC>

Примеры: LSL L R2, R6 R7, CCR

LSL L 1, R6 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					R					sc	RC					1	0	de	#	1	1	0	1	0	0	0

Автокод: LSL L6 S, D, MRC, rs, RC

LSL L6 S, D, MRCd, RC, rd

Формат 7: LSL L[.cc] s, D <#16/32  $\beta$ à RC/R/R.L>

Пример: LSL L R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: LSL7 s, D, MIL, cc, #32, RD

LSL7 s, D, MIC, cc, #16, RC

Формат 8a: **LSLL** s3/#5, S4, D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: **LSLL** R1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

**LSLL** 1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/#5				D2				S4				AT	mode				A	de	0	0	#	0	1	0	0	0					

Автокод: PROL LSL7 s3/#5, S4, D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: **LSLL** s3/#5, S4, D2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: **LSLL** R1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

**LSLL** 1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT	RD/Rd				L	0	0	1	#	0	1	0	0	0					

Автокод: `PROL LSLL s3/#5,S4,D2 <OP2> <R.R.LBàR.R.L> <YRAMàR0>`

Формат 8с: `LSLL[.cc] s3/#5,S4,D2 <OP1> <R.LBàR.L>`

Примеры: `LSLL.eq R1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L`

`LSLL.eq 1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L`

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				cc	RD				0	cc	1	0	#	0	1	0	0	0					

Автокод: `PROL LSLL cc s3,S4,D2 <OP1> <R.LBàR.L>`

Формат 8d: `LSLL s3/#5,S4,D2 <OP1> <RBàRC>`

Примеры: `LSLL R1,R2,R6 ANDL R6,R0,R8 CCR,R17`

`LSLL 1,R2,R6 ANDL R6,R0,R8 CCR,R17`

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				0	0	RC				sc	de	1	1	#	0	1	0	0	0				

Автокод: `PROL LSLL s3/#5,S4,D2 <OP1> <RBàRC>`

**LSLX**      *Логический сдвиг влево (X16)*      **LSLX**

**Операция:**

3-адресный вариант:

X16 S1,S2,D; (S2[31:16] << S1[21:16]) à D[31:16]  
 (S2[15:0] << S1[5:0])      à D[15:0]

2-адресный вариант:

X16 S,D; (D[31:16] << S[21:16]) à D[31:16]  
 (D[15:0] << S[5:0])      à D[15:0]

**Описание:** В формате X16 операнд-источник S2 (в 3-адресном варианте) или D (в 2-адресном варианте) сдвигается влево на количество разрядов, заданное S1 (в 3-адресном варианте) или S (в 2-адресном варианте), при этом старшие 16 разрядов операнда (действительная часть Re) и младшие 16 разрядов операнда (мнимая часть Im) сдвигаются независимо – соответственно на S1[21:16] и S1[5:0] разрядов в 3-адресном варианте или S[21:16] и S[5:0] разрядов в 2-адресном варианте. Разряды S1[31:22], S1[15:6] (S[31:22], S[15:6]) при этом должны быть установлены в «0». Результат сдвига старших 16 разрядов операнда помещается в старшие 16 разрядов операнда-приемника D, а результат сдвига младших 16 разрядов операнда помещается в младшие 16 разрядов операнда-приемника D. Вытолкнутые за пределы разрядной сетки разряды теряются. Младшие освободившиеся в результате сдвига разряды заполняются нулями.

В качестве первого операнда-источника может использоваться непосредственный операнд #5, который определяет сдвиг и мнимой и действительной части.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
---	---	---	---	---

Ur&Ui	*	Zr&Zi	-	*
-------	---	-------	---	---

\*N – последний вытолкнутый разряд мнимой части

\*C – последний вытолкнутый разряд действительной части

### Синтаксис ассемблера и код инструкции:

Формат 1: **LSLX**[.cc] S1/#5,S2,D

Примеры: **LSLX**.ne R1,R6,R12

**LSLX**.ne 17,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S1/#5					D					S2					#	0	cc					0	0	0	1	1	1	0	0	0	1	0

Автокод: LSLX cc, s1/#5, S2, D

Формат 4: **LSLX** S/#5,D <XRAM $\beta$ aR.L>

Примеры: **LSLX** R1,R2 R8,(A0)+

**LSLX** 17,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S/#5					D					R					A					mode	u	0	1	de	#	1	1	0	0	0	1	0

Автокод: LSLX4 S, D, MLR,ea,A,R

LSLX4 S, D, MLM,ea,A,R

Формат 5: **LSLX** S,D <R/R.L $\beta$ aR/R.L>

Примеры: **LSLX** R1,R2 R8.L,R12.L

**LSLX** 15,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S/#5					D					RS/Rs					L	RD/Rd					1	1	0	#	1	1	0	0	0	1	0

Автокод: LSLX5 S, D, MR,rs,rd

LSLX5 S, D, MRL,RS,RD

Формат 6: **LSLX** S,D <R $\beta$ àRC>

Примеры: **LSLX** R1,R6 R7,CCR

**LSLX** 1,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S/#5					D					R					sc	RC					1	0	de	#	1	1	0	0	0	1	0

Автокод: LSLX6 S, D ,MRC,rs,RC

LSLX6 S, D ,MRCd,RC,rd

Формат 7: **LSLX**[.cc] S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **LSLX** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc			1	1	1	0	1	1	0	0	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: LSLX7 S, D, MIL,cc,#32,RD

LSLX7 S, D, MIC,cc,#16,RC

Формат 8a: **LSLX** S3/#5,S4,D2 <OP1> <XRAM $\beta$ àR.L> <YRAM $\beta$ àR0>

Примеры: **LSLX** R1,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

**LSLX** 1,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R			u	0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/#5				D2				S4				AT	mode	A	de	0	0	#	0	0	0	1	0								

Автокод: PROL LSLX S3/#5,S4,D2 <OP1> <XRAM $\beta$ àR.L> <YRAM $\beta$ àR0>

Формат 8b: **LSLX** S3/#5,S4,D2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAM $\beta$ àR0>

Примеры: **LSLX** R1,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0



**LSLX** 1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/#5				D2				S4				AT	RD/Rd				L	0	0	1	#	0	0	0	1	0					

Автокод: PROL LSLX S3/#5, S4, D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: **LSLX**[.cc] S3/#5, S4, D2 <OP1> <R.L $\beta$ àR.L>

Примеры: **LSLX**.eq R1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

**LSLX**.eq 1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/#5				D2				S4				cc	RD				0	cc	1	0	#	0	0	0	1	0					

Автокод: PROL LSLX cc S3, S4, D2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: **LSLX** S3/#5, S4, D2 <OP1> <R $\beta$ àRC>

Примеры: **LSLX** R1, R2, R6 ANDL R6, R0, R8 CCR, R17

**LSLX** 1, R2, R6 ANDL R6, R0, R8 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/#5				D2				S4				0	0	RC				sc	de	1	1	#	0	0	0	1	0					

Автокод: PROL LSLX S3/#5, S4, D2 <OP1> <R $\beta$ àRC>

**LSR**
**Логический сдвиг вправо (short)**
**LSR**
**Операция:**

3-адресный вариант: short s1,s2,d; (s2 >> s1[5:0]) **à** d

2-адресный вариант: short s,d; (d >> s[5:0]) **à** d

**Описание:** Операнд-источник s2 (в 3-адресном варианте) или d (в 2-адресном варианте) сдвигается вправо на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте) и результат помещается в операнд-приемник d. Старшие десять разрядов s1 (или s) при этом должны быть установлены в «0». Вытолкнутые за пределы разрядной сетки разряды теряются, кроме последнего вытолкнутого, который запоминается в признаке C.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	-	√	-	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **LSR[.cc]** s1/#5,s2,d

Примеры: **LSR.ne** R1,R5,R12

**LSR.ne** 17,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					d					s2					#	0	cc					0	0	0	1	1	1	1	0	0	0	1

*Автокод:* LSR cc, s1/#5, s2, d

Формат 4: **LSR** s/#5,d <XRAM**β**àR.L>

Примеры: **LSR** R1,R2 R8,(A0)+

**LSR** 17,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					d				R			A			mode		u	0	1	de	#	1	1	1	0	0	0	1			

Автокод: LSR4 s, d, MLR,ea,A,R

LSR4 s, d, MLM,ea,A,R

Формат 5: **LSR** s,d <R/R.L $\beta$ àR/R.L>

Примеры: **LSR** R1,R2 R8.L,R12.L

**LSR** 15,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					d				RS/Rs			L	RD/Rd			1	1	0	#	1	1	1	0	0	0	1					

Автокод: LSR5 s, d, MR,rs,rd

LSR5 s, d, MRL,RS,RD

Формат 6: **LSR** s,d <R $\beta$ àRC>

Примеры: **LSR** R1,R5 R7,CCR

**LSR** 1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					d				R			sc	RC			1	0	de	#	1	1	1	0	0	1						

Автокод: LSR6 s, d ,MRC,rs,RC

LSR6 s, d ,MRCd,RC,rd

Формат 7: **LSR[.cc]** s,d <#16/32 $\beta$ àRC/R/R.L>

Пример: **LSR** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	0	0	0	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: LSR7 s, d, MIL, cc, #32, RD

LSR7 s, d, MIC, cc, #16, RC

Формат 8a: **LSR** s3/#5, s4, d2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **LSR** R1, R2, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

**LSR** 1, R2, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	mode				A	de	0	0	#	1	0	0	0	1					

Автокод: PROL LSR s3/#5, s4, d2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **LSR** s3/#5, s4, d2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **LSR** R1, R2, R5 ANDL R6, R0, R8 R8, R15 (AT), R0

**LSR** 1, R2, R5 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	RD/Rd				L	0	0	1	#	1	0	0	0	1					

Автокод: PROL LSR s3/#5, s4, d2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: **LSR**[.cc] s3/#5, s4, d2 <OP1> <R.LβàR.L>

Примеры: **LSR**.eq R1, R2, R5 ANDL R6, R0, R8 R8.L, R16.L

**LSR**.eq 1,R2,R5    **ANDL** R6,R0,R8    R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				cc	RD				0	cc	1	0	#	1	0	0	0	0	1				

Автокод: PROL    LSR cc s3,s4,d2    <OP1>    <R.L**B**àR.L>

Формат 8d: **LSR** s3/#5,s4,d2<OP1>    <R**B**àRC>

Примеры:    **LSR** R1,R2,R5    **ANDL** R6,R0,R8    **CCR**,R17

**LSR** 1,R2,R5    **ANDL** R6,R0,R8    **CCR**,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				0	0	RC				sc	de	1	1	#	1	0	0	0	1				

Автокод: PROL    LSR s3/#5,s4,d2    <OP1>    <R**B**àRC>

**LSRL                      Логический сдвиг вправо (long)                      LSRL**

**Операция:**

3-адресный вариант:

```
short s1;      long S2,D;              (S2 >> s1[5:0]) à D
```

2-адресный вариант:

```
short s1;      long D;              (D >> s[5:0]) à D
```

**Описание:** Операнд-источник S2 (в 3-адресном варианте) или D (в 2-адресном варианте) сдвигается вправо на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте) и результат помещается в операнд-приемник D. Старшие десять разрядов s1 (или s) при этом должны быть установлены в «0». Вытолкнутые за пределы разрядной сетки разряды теряются, кроме последнего вытолкнутого, который запоминается в признаке С.

В качестве первого операнда может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	–	√	–	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **LSRL[.cc] s1/#5,S2,D**

Примеры: **LSRL.ne R2,R6,R12**

**LSRL.ne 17,R6,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					D					S2					#	0	cc					0	0	0	1	1	1	1	1	0	0	0

Автокод: LSRL cc, s1/#5, S2, D

Формат 4: **LSRL** s/#5,D <XRAM $\beta$ àR.L>

Примеры: **LSRL** R2,R4 R8,(A0)+

**LSRL** 17,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
s/#5					D					R					A					mode					u	0	1	de	#	1	1	1	1	0	0	0

Автокод: LSRL4 s, D, MLR,ea,A,R

LSRL4 s, D, MLM,ea,A,R

Формат 5: **LSRL** s,D <R/R.L $\beta$ àR/R.L>

Примеры: **LSRL** R2,R4 R8.L,R12.L

**LSRL** 15,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					RS/Rs					L	RD/Rd					1	1	0	#	1	1	1	1	0	0	0

Автокод: LSRL5 s, D, MR,rs,rd

LSRL5 s, D, MRL,RS,RD

Формат 6: **LSRL** s,D <R $\beta$ àRC>

Примеры: **LSRL** R2,R6 R7,CCR

**LSRL** 1,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					R					sc	RC					1	0	de	#	1	1	1	1	0	0	0

Автокод: LSRL6 s, D ,MRC,rs,RC

LSRL6 s, D ,MRCd,RC,rd

Формат 7: **LSRL[.cc]** s,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **LSRL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s					D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	1	0	0	0	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: LSRL7 s, D, MIL, cc, #32, RD

LSRL7 s, D, MIC, cc, #16, RC

Формат 8a: **LSRL** s3/#5, S4, D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: **LSRL** R1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

**LSRL** 1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1					D/d				S2/s2				M	R				u	0	0	0	0	OP1								

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5					D2				S4				AT	mode				A	de	0	0	#	1	1	0	0	0				

Автокод: PROL LSRL s3/#5, S4, D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: **LSRL** s3/#5, S4, D2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: **LSRL** R1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

**LSRL** 1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1					D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5					D2				S4				AT	RD/Rd				L	0	0	1	#	1	1	0	0	0				

Автокод: PROL LSRL s3/#5, S4, D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: **LSRL**[.cc] s3/#5, S4, D2 <OP1> <R.L $\beta$ àR.L>

Примеры: **LSRL**.eq R1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L



**LSRL**.eq 1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				cc	RD				0	cc	1	0	#	1	1	0	0	0					

Автокод: PROL LSRL cc s3/#5,S4,D2 <OP1> <R.L**β**R.L>

Формат 8d: **LSRL** s3/#5,S4,D2 <OP1> <R**β**RC>

Примеры: **LSRL** R1,R2,R6 ANDL R6,R0,R8 CCR,R17

**LSRL** 1,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				0	0	RC				sc	de	1	1	#	1	1	0	0	0				

Автокод: PROL LSRL s3/#5,S4,D2 <OP1> <R**β**RC>

**LSRX                      Логический сдвиг вправо (X16)                      LSRX**

**Операция:**

3-адресный вариант: X16 S1, S2, D;

$(S2[31:16] \gg S1[21:16]) \text{ à } D[31:16]$

$(S2[15:0] \gg S1[5:0]) \text{ à } D[15:0]$

2-адресный вариант: X16 S, D;

$(D[31:16] \gg S[21:16]) \text{ à } D[31:16]$

$(D[15:0] \gg S[5:0]) \text{ à } D[15:0]$

**Описание:** В формате X16 операнд-источник S2 (в 3-адресном варианте) или D (в 2-адресном варианте) сдвигается вправо на количество разрядов, заданное S1 (в 3-адресном варианте) или S (в 2-адресном варианте), при этом старшие 16 разрядов операнда (действительная часть Re) и младшие 16 разрядов операнда (мнимая часть Im) сдвигаются независимо – соответственно на S1[21:16] и S1[5:0] разрядов в 3-адресном варианте или S[21:16] и S[5:0] разрядов в 2-адресном варианте. Разряды S1[31:22], S1[15:6] (S[31:22], S[15:6]) при этом должны быть установлены в «0». Результат сдвига старших 16 разрядов операнда помещается в старшие 16 разрядов операнда-приемника D, а результат сдвига младших 16 разрядов операнда помещается в младшие 16 разрядов операнда-приемника D. Вытолкнутые за пределы разрядной сетки разряды теряются. Старшие освободившиеся в результате сдвига разряды заполняются нулями.

В качестве первого операнда-источника может использоваться непосредственный операнд #5, который определяет сдвиг и мнимой и действительной части.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
Ur&Ui	*	Zr&Zi	-	*

\*N – последний вытолкнутый разряд действительной части

\*C – последний вытолкнутый разряд мнимой части

### Синтаксис ассемблера и код инструкции:

Формат 1: **LSRX**[.cc] S1/#5, S2, D

Примеры: **LSRX.ne** R1, R6, R12

**LSRX.ne** 1, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S1/#5					D					S2					#	0	cc					0	0	0	1	1	1	1	0	0	1	0

Автокод: **LSRX** cc, S1/#5, S2, D

Формат 4: **LSRX** S/#5, D <XRAM $\beta$ àR.L>

Примеры: **LSRX** R1, R2 R8, (A0)+

**LSRX** 1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
S/#5					D					R					A					mode	u	0	1	de	#	1	1	1	1	0	0	1	0

Автокод: **LSRX4** S, D, MLR, ea, A, R

**LSRX4** S, D, MLM, ea, A, R

Формат 5: **LSRX** S, D <R/R.L $\beta$ àR/R.L>

Примеры: **LSRX** R1, R2 R8.L, R12.L

**LSRX** 15, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					RS/Rs					L	RD/Rd					1	1	0	#	1	1	1	0	0	1	0

Автокод: **LSRX5** S, D, MR, rs, rd

**LSRX5** S, D, MRL, RS, RD

Формат 6: **LSRX** S, D <R $\beta$ àRC>

Примеры: **LSRX** R1,R6 R7,CCR

**LSRX** 1,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					D					R					sc	RC					1	0	de	#	1	1	1	0	0	1	0

Автокод: **LSRX6** S, D ,MRC,rs,RC

**LSRX6** S, D ,MRCd,RC,rd

Формат 7: **LSRX**[.cc] S,D <#16/32àRC/R/R.L>

Пример: **LSRX** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S					D					RD/Rd/RC					L/sc	sr	cc					1	1	1	0	1	1	1	0	0	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **LSRX7** S, D, MIL,cc,#32,RD

**LSRX7** S, D, MIC,cc,#16,RC

Формат 8а: **LSRX** S3/#5,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **LSRX** R1,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

**LSRX** 1,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1					D/d					S2/s2					M	R					u	0	0	0	0	OP1					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32				
S3/#5				D2				S4				AT		mode		A		de		0		0		#		1		0		0		1		0	

Автокод: PROL LSRX s3/#5, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **LSRX** S3/#5, S4, D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **LSRX** R1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

**LSRX** 1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M		RS/Rs				0		0		0		0		OP1					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32						
S3/#5				D2				S4				AT		RD/Rd				L		0		0		1		#		1		0		0		1		0	

Автокод: PROL LSRX S3/#5, S4, D2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с: **LSRX**[.cc] S3/#5, S4, D2 <OP1> <R.LβàR.L>

Примеры: **LSRX**.eq R1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

**LSRX**.eq 1, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M		RS				0		0		0		0		OP1					

6	6	6	6	5	5	5	5	5	5	5	5	5	5	4	4	4	4	4	4	4	4	4	4	4	4	3	3	3	3	3	3	3	3	3	
S3/#5				D2				S4				c		RD				c																	

Автокод: PROL LSRX cc S3, S4, D2 <OP1> <R.LβàR.L>

Формат 8d: **LSRX** S3/#5, S4, D2 <OP1> <RβàRC>

Примеры: **LSRX** R1, R2, R6 ANDL R6, R0, R8 CCR, R17

**LSRX** 1, R2, R6 ANDL R6, R0, R8 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/#5				D2				S4				0	0	RC				sc	de	1	1	#	1	0	0	1	0				

Автокод: PROL

LSRX S3/#5, S4, D2

&lt;OP1&gt;

 <R**В**RC>

### MAC Умножение (short) с накоплением (\_\_Int64) MAC

#### Операция:

3-адресный вариант: short s1; long S2,D;

{AC1,AC0}  $\leftarrow$  {S2,D}

$s1[15:0] * S2[15:0] + \{AC1,AC0\} \leftarrow \{AC1,AC0\}$

2-адресный вариант: long S,D;

{AC1,AC0}  $\leftarrow$  {S,D}

$S2[15:0] * D[15:0] + \{AC1,AC0\} \leftarrow \{AC1,AC0\}$

**Описание:** Вычисляется произведение операндов-источников (s1,S2[15:0] - в 3-адресном варианте; S[15:0],D[15:0] - в 2-адресном варианте), представленных в формате short, и 32-разрядное произведение (long) складывается с содержимым 64-разрядного аккумулятора (\_\_Int64) AC={AC1,AC0}, составленного из двух 32-разрядных регистров AC1 (старшая часть) и AC0 (младшая часть). Все вычисления - целочисленные со знаком. Состояние регистра AC1 на момент начала операции записывается по адресу S2 (в 3-адресном варианте) или S (в 2-адресном варианте), состояние регистра AC0 на момент начала операции – по адресу D.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 8

#### Признаки результата:

U	N	Z	V	C
√	√	√	√	√

Примечания:

Признаки результата формируются по состоянию 64-разрядного аккумулятора AC={AC1,AC0}.

Признак Z формируется по состоянию АС на момент начала операции, остальные – по новому результату АС.

### Синтаксис ассемблера и код инструкции:

Формат 1: **MAC[.cc] s1,S2,D**

Пример: **MAC.ne R2,R6,R12**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				S2				1	0	cc			0	0	0	1	1	1	0	1	0	0	1				

*Автокод:* **MAC cc, s1, S2, D**

Формат 4: **MAC S,D <XRAM $\beta$ àR.L>**

Пример: **MAC R2,R4 R8,(A0)+**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	1	1	1	0	1	0	0	1		

*Автокод:* **MAC4 S, D, MLR,ea,A,R**

**MAC4 S, D, MLM,ea,A,R**

Формат 5: **MAC S,D <R/R.L $\beta$ àR/R.L>**

Пример: **MAC R2,R4 R8.L,R12.L**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd			1	1	0	1	1	1	0	1	0	0	1					

*Автокод:* **MAC5 S, D, MR,rs,rd**

**MAC5 S, D, MRL,RS,RD**

Формат 6: **MAC S,D <R $\beta$ àRC>**

Пример: **MAC R2,R6 R7,CCR**

*Код инструкции:*



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	1	1	1	0	1	0	0	1				

Автокод: MAC6 S, D, MRC, rs, RC

MAC6 S, D, MRCd, RC, rd

Формат 8a: **MAC** s3, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Пример: **MAC** R4, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				AT	mode				A	de	0	0	1	0	1	0	0	1					

Автокод: PROL MAC s3, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **MAC** s3, S4, D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Пример: **MAC** R4, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				AT	RD/Rd				L	0	0	1	1	0	1	0	0	1					

Автокод: PROL MAC s3, S4, D2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: **MAC**[.cc] s3, S4, D2 <OP1> <R.LβàR.L>

Пример: **MAC**.eq R4, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				cc		RD				0	cc		1	0	1	0	1	0	1	0	0	0	1

Автокод: PROL      MAC cc s3,S4,D2      <OP1>      <R.LβàR.L>

Формат 8d: **MAC** s3,S4,D2      <OP1>      <RβàRC>

Пример:      **MAC** R4,R2,R6      ANDL R6,R0,R8      CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				0	0	RC				sc	de	1	1	1	0	1	0	0	0	1			

Автокод: PROL      MAC s3,S4,D2      <OP1>      <RβàRC>

**MACL Умножение (long) с накоплением (\_\_Int64) MACL**
**Операция:**

3-адресный вариант: long S1, S2, D;

{AC1, AC0}  $\leftarrow$  {S2, D}

$S1[31:0] * S2[31:0] + \{AC1, AC0\} \leftarrow \{AC1, AC0\}$

2-адресный вариант: long S, D;

{AC1, AC0}  $\leftarrow$  {S, D}

$S2[31:0] * D[31:0] + \{AC1, AC0\} \leftarrow \{AC1, AC0\}$

**Описание:** Вычисляется произведение операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте), представленных в формате long, и 64-разрядное произведение (\_\_Int64) складывается с содержимым 64-разрядного аккумулятора (\_\_Int64) AC={AC1, AC0}, составленного из двух 32-разрядных регистров AC1 (старшая часть) и AC0 (младшая часть). Все вычисления - целочисленные со знаком. Состояние регистра AC1 на момент начала операции записывается по адресу S2 (в 3-адресном варианте) или S (в 2-адресном варианте), состояние регистра AC0 на момент начала операции – по адресу D.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

Примечания:

Признаки результата формируются по состоянию 64-разрядного аккумулятора AC={AC1, AC0}.

Признак Z формируется по состоянию AC на момент начала операции, остальные – по новому результату AC.

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MACL**[.cc] s1,S2,D

Пример: **MACL.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				S2				1	0	cc				0	0	0	1	1	1	1	1	1	0	1	1		

Автокод: **MACL** cc, s1, S2, D

Формат 4: **MACL** S,D <XRAM $\beta$ àR.L>

Пример: **MACL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	1	1	1	1	1	1	0	1	1	

Автокод: **MACL4** S, D, MLR,ea,A,R

**MACL4** S, D, MLM,ea,A,R

Формат 5: **MACL** S,D <R/R.L $\beta$ àR/R.L>

Пример: **MACL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	1	1	1	1	1	1	0	1	1			

Автокод: **MACL5** S, D, MR,rs,rd

**MACL5** S, D, MRL,RS,RD

Формат 6: **MACL** S,D <R $\beta$ àRC>

Пример: **MACL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	1	1	1	1	1	1	0	1	1			

Автокод: MACL6 S, D, MRC, rs, RC

MACL6 S, D, MRCd, RC, rd

Формат 8a: **MACL** s3, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Пример: **MACL** R4, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				AT	mode				A	de	0	0	1	1	1	0	1	1					

Автокод: PROL MACL s3, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **MACL** s3, S4, D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Пример: **MACL** R4, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				AT	RD/Rd				L	0	0	1	1	1	1	0	1	1					

Автокод: PROL MACL s3, S4, D2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: **MACL**[.cc] s3, S4, D2 <OP1> <R.LβàR.L>

Пример: **MACL**.eq R4, R2, R6 ANDL R6, R0, R8 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

s3	D2	S4	cc	RD	0	cc	1	0	1	1	1	0	1	1
----	----	----	----	----	---	----	---	---	---	---	---	---	---	---

Автокод: PROL      MACL s3, S4, D2      <OP1>      <R.LβàR.L>

Формат 8d: **MACL** s3, S4, D2      <OP1>      <RβàRC>

Пример:      **MACL** R4, R2, R6      ANDL R6, R0, R8      CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				S4				0	0	RC				sc	de	1	1	1	1	1	0	1	1				

Автокод: PROL      MACL s3, S4, D2      <OP1>      <RβàRC>

**MACX Умножение дробное комплексно-сопряженное (X8) и целочисленное с накоплением (X16) MACX**

**Операция:**

3-адресный вариант: long S1, S2, D;

$S1[31:24] * S2[31:24] + S1[15:8] * S2[15:8] + AC1 \rightarrow AC1$

$S1[15:8] * S2[31:24] - S1[31:24] * S2[15:8] + AC0 \rightarrow AC0$

$((S1[23:16]*S2[23:16])\ll 1) + ((S1[7:0]*S2[7:0])\ll 1) \rightarrow D[31:16]$

$((S1[7:0]*S2[23:16])\ll 1) - ((S1[23:16]*S2[7:0])\ll 1) \rightarrow D[15:0]$

2-адресный вариант: long S, D;

$S[31:24] * D[31:24] + S[15:8] * D[15:8] + AC1 \rightarrow AC1$

$S[15:8] * D[31:24] - S[31:24] * D[15:8] + AC0 \rightarrow AC0$

$((S[23:16]*D[23:16])\ll 1) + ((S[7:0]*D[7:0])\ll 1) \rightarrow D[31:16]$

$((S[7:0]*D[23:16])\ll 1) - ((S[23:16]*D[7:0])\ll 1) \rightarrow D[15:0]$

**Описание:** Выполняются одновременно две операции над комплексными числами: дробное комплексно-сопряженное умножение и целочисленное комплексно-сопряженное умножение с накоплением. Входные комплексные числа упакованы в 32-разрядных операндах S1 и S2. Комплексно-сопряженное умножение выполняется над сомножителями в дробном знаковом 8-разрядном формате (X8), выходное произведение имеет дробный знаковый 16-разрядный формат (X16).

Комплексная MAC-операция (комплексное сопряженное умножение и накопление комплексных произведений в аккумуляторах AC0, AC1) выполняется над входными сомножителями в целом знаковом 8-разрядном формате (X8). 17-разрядные действительная и мнимая компоненты произведения добавляются к 32-разрядным аппаратным аккумуляторам AC1 и AC0, соответственно.

Возможен режим насыщения (бит CCR[8] управляющего регистра равен 1). Он относится одновременно к обеим операциям: умножению и MAC. При переполнении любой компоненты

в операции умножения она заменяется на 0x7FFF (переполнение сверху), либо на 0x8000 (переполнение снизу).

При выполнении комплексной операции MAC на переполнение контролируются оба аккумулятора. При переполнении любого из них в него заносится значение 0xFFFFFFFF (переполнение сверху), либо 0x80000000 (переполнение снизу).

Значения аккумуляторов AC1 и AC0 не выводятся.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*

Примечания. 1) Признаки U, N, V, C формируются по новым значениям аккумуляторов: AC1н и AC0н.

2) Признак Z формируется по предшествующим значениям аккумуляторов: AC1 и AC0.

\*U=1, если оба аккумулятора, AC1н и AC0н, не нормализованы, т.е. AC1н[31]=AC1н[30] и AC0н[31]=AC0н[30] (иначе 0).

\*N=AC1н[31] – знак результата AC1н.

\*Z=1 при нулевых результатах AC1 и AC0 (иначе 0).

\*V=1 при переполнении любого из результатов, AC1 или AC0 (иначе 0), независимо от бита управления режимом насыщения CCR[8].

\*C= AC0н[31] – знак результата AC0н.

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MACX[.cc] S1, S2, D**

Пример: **MACX.ne R2, R6, R12**



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				1	0	cc			0	0	0	1	1	1	1	1	1	0	0	1			

Автокод:                    MACX cc, S1, S2, D

Формат 4:    **MACX** S,D <XRAM $\beta$ àR.L>

Пример:        **MACX** R2,R4        R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	1	1	1	1	1	1	0	0	1	

Автокод:        MACX4 S, D, MLR,ea,A,R

MACX4 S, D, MLM,ea,A,R

Формат 5:    **MACX** S,D <R/R.L $\beta$ àR/R.L>

Пример:        **MACX** R2,R4        R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	1	1	1	1	1	1	0	0	1			

Автокод:        MACX5 S, D, MR,rs,rd

MACX5 S, D, MRL,RS,RD

Формат 6:    **MACX** S,D <R $\beta$ àRC>

Пример:        **MACX** R2,R6        R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	1	1	1	1	1	1	0	0	1			

Автокод:        MACX6 S, D ,MRC,rs,RC

MACX6 S, D ,MRCd,RC,rd

Формат 8а:    **MACX** S3,S4,D2 <OP1> <XRAM $\beta$ àR.L> <YRAM $\beta$ àR0>

Пример:        **MACX** R4,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	1	1	1	0	0	1					

Автокод: PROL MACX S3,S4,D2 <OP1> <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: **MACX** S3,S4,D2 <OP1> <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Пример: **MACX** R4,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	RD/Rd				L	0	0	1	1	1	1	0	0	1					

Автокод: PROL MACX S3,S4,D2 <OP2> <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: **MACX**[.cc] S3,S4,D2 <OP1> <R.L $\beta$ R.L>

Пример: **MACX**.eq R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc	RD				0	cc	1	0	1	1	1	0	0	1					

Автокод: PROL MACX cc S3,S4,D2 <OP1> <R.L $\beta$ R.L>

Формат 8d: **MACX** S3,S4,D2 <OP1> <R $\beta$ RC>

Пример: **MACX** R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	1	1	1	0	0	1				

АВТОКОД: PROL      MACX S3, S4, D2      <OP1>      <R**B**àRC>

**MAC2 Парное умножение (short) с накоплением (long) MAC2**
**Операция:**

3-адресный вариант: long S1, S2, D;

AC0 **à** D

S1[15:0] \* S2[15:0] + AC0 **à** AC0

AC1 **à** S2

S1[31:16] \* S2[31:16] + AC1 **à** AC1

2-адресный вариант: long S, D;

AC0 **à** D

S[15:0] \* D[15:0] + AC0 **à** AC0

AC1 **à** S

S[31:16] \* D[31:16] + AC1 **à** AC1

**Описание:** Вычисляются произведения двух пар операндов-источников (S1[15:0]\*S2[15:0] и S1[31:16]\*S2[31:16] - в 3-адресном варианте; S[15:0]\*D[15:0] и S[31:16]\*D[31:16] - в 2-адресном варианте), представленных в формате short, и 32-разрядные произведения (long) складываются с содержимым соответствующего 32-разрядного аккумулятора AC0 и AC1. Все вычисления - целочисленные со знаком. Состояние регистра AC1 на момент начала операции записывается по адресу S2 (в 3-адресном варианте) или S (в 2-адресном варианте), состояние регистра AC0 на момент начала операции – по адресу D.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*

Примечания. 1) Признаки U, N, V, C формируются по новым значениям аккумуляторов: AC1н и AC0н.

2) Признак Z формируется по предшествующим значениям аккумуляторов: AC1 и AC0.

\*U=1, если оба аккумулятора, AC1<sub>n</sub> и AC0<sub>n</sub>, не нормализованы, т.е. AC1<sub>n</sub>[31]=AC1<sub>n</sub>[30] и AC0<sub>n</sub>[31]=AC0<sub>n</sub>[30] (иначе 0).

\*N=AC1<sub>n</sub>[31] – знак результата AC1<sub>n</sub>.

\*Z=1 при нулевых результатах AC1 и AC0 (иначе 0).

\*V=1 при переполнении AC1 (иначе 0), независимо от бита управления режимом насыщения CCR[8].

\*C= AC0<sub>n</sub>[31] – знак результата AC0<sub>n</sub>.

### Синтаксис ассемблера и код инструкции:

Формат 1: **MAC2**[.cc] S1, S2, D

Пример: **MAC2.ne** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				1	0	cc			0	0	0	1	1	1	0	1	1	1	0				

Автокод: MAC2 cc, S1, S2, D

Формат 4: **MAC2** S, D <XRAMβàR.L>

Пример: **MAC2** R2, R4 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	1	1	1	0	1	1	1	0		

Автокод: MAC24 S, D, MLR, ea, A, R

MAC24 S, D, MLM, ea, A, R

Формат 5: **MAC2** S, D <R/R.LβàR/R.L>

Пример: **MAC2** R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	1	1	1	0	1	1	1	0	1	1	1	0

Автокод: MAC25 S, D, MR,rs,rd

MAC25 S, D, MRL,RS,RD

Формат 6: **MAC2** S,D <RβàRC>

Пример: **MAC2** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	1	1	1	0	1	1	1	0	1	1	1	0

Автокод: MAC26 S, D ,MRC,rs,RC

MAC26 S, D ,MRCd,RC,rd

Формат 8a: **MAC2** S3,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Пример: **MAC2** R4,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	1	0	1	1	1	0					

Автокод: PROL MAC2 S3,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **MAC2** S3,S4,D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Пример: **MAC2** R4,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	RD/Rd				L	0	0	1	1	0	1	1	1	0					

Автокод: PROL MAC2 S3,S4,D2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с: **MAC2**[.cc] S3,S4,D2 <OP1> <R.L**β**àR.L>

Пример: **MAC2**.eq R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc	RD				0	cc	1	0	1	0	1	1	1	1	0				

Автокод: PROL MAC2 cc S3,S4,D2 <OP1> <R.L**β**àR.L>

Формат 8d: **MAC2** S3,S4,D2 <OP1> <R**β**àRC>

Пример: **MAC2** R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	1	0	1	1	1	0				

Автокод: PROL MAC2 S3,S4,D2 <OP1> <R**β**àRC>

**MAX**                      **Выбор большего числа (short)**                      **MAX**

**Операция:**

3-адресный вариант: short s1,s2,d;

if (s1 > s2) s1 à d

else s2 à d

2-адресный вариант: short s,d;

if (s > d) s à d

**Описание:** Определяется большее число: в формате short вычисляется разность операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); если результат отрицательный или равен нулю, то в операнд-приемник d помещается первый операнд-источник (s1 - в 3-адресном варианте; s - в 2-адресном варианте), иначе в операнд-приемник d помещается второй операнд-источник s2 в 3-адресном варианте (в 2-адресном варианте содержимое приемника d не изменяется). В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	-	*

\*C=1, если s1 > s2 (иначе 0).

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MAX[.cc]** s1,s2,d

Пример: **MAX.ne** R1,R5,R12

Код инструкции:



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	0	1	0	0	1	0	1	1	1			

Автокод: MAX cc, s1, s2, d

Формат 2: MAX[.cc] #16, s2, d

Пример: MAX.ne 15, R5, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	1	0	0	0	1	0	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: MAX2 cc, #16, s2, d

Формат 3: MAX #16, d

Пример: MAX 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	0	0	1	0	1	1	1						

Автокод: MAX3 #16, d

Формат 4: MAX s, d <XRAM $\beta$ àR.L>

Пример: MAX R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	0	1	0	1	1	1		

Автокод: MAX4 s, d, MLR, ea, A, R

MAX4 s, d, MLM, ea, A, R

Формат 5: MAX s, d <R/R.L $\beta$ àR/R.L>

Пример: MAX R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s	d	RS/Rs	L	RD/Rd	1	1	0	0	0	0	1	0	1	1	1
---	---	-------	---	-------	---	---	---	---	---	---	---	---	---	---	---

Автокод: MAX5 s, d, MR,rs,rd  
 MAX5 s, d, MRL,RS,RD

Формат 6: **MAX** s,d <RβàRC>

Пример: **MAX** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	0	1	1	1				

Автокод: MAX6 s, d ,MRC,rs,RC  
 MAX6 s, d ,MRCd,RC,rd

Формат 7: **MAX[.cc]** s,d <#16/32àRC/R/R.L>

Пример: **MAX** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	0	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MAX7 s, d, MIL,cc,#32,RD  
 MAX7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **MAX** s1,s2,d <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MAX** R1,R2,R5 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	0	0	1	0	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2										

Автокод: PROL <OP2> MAX s1,s2,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **MAX** s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MAX** R1,R2,R5 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs			0	0	0	0	0	0	1	0	1	1	1					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd			L	0	0	1	#	OP2										

Автокод: PROL <OP2> MAX s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **MAX** s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 **MAX** R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS			0	0	0	0	0	0	1	0	1	1	1					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD			0	cc	1	0	#	OP2										

Автокод: PROL <OP2 cc> MAX s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> **MAX** s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 **MAX** R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	Rs/Rd			0	0	0	0	0	0	1	0	1	1	1					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC			sc	de	1	1	#	OP2									

Автокод: PROL <OP2> MAX s1,s2,d <RbàRC>

**MAXL**
**Выбор большего числа (long)**
**MAXL**
**Операция:**
3-адресный вариант: long S1, S2, D;

if (S1 &gt; S2) S1 → D

else S2 → D

2-адресный вариант: long S, D;

if (S &gt; D) S → D

**Описание:** Определяется большее число: в формате long вычисляется разность операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); если результат отрицательный или равен нулю, то в операнд-приемник D помещается первый операнд-источник (S1 - в 3-адресном варианте; S - в 2-адресном варианте), иначе в операнд-приемник D помещается второй операнд-источник S2 в 3-адресном варианте (в 2-адресном варианте содержимое приемника D не изменяется). В качестве первого операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:**

OP1

**Форматы:**

1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	*

\*C=1, если S1 &gt; S2 (иначе 0).

**Синтаксис ассемблера и код инструкции:**
Формат 1: **MAXL[.cc]** S1, S2, D

 Пример: **MAXL.ne** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	0	1	0	1	1	0	1	1	1			

Автокод: MAXL cc, S1, S2, D

Формат 2: **MAXL**[.cc] #32,S2,D

Пример: **MAXL**.ne 15,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	0	1	1	0	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: MAXL2 cc, #32, S2, D

Формат 4: **MAXL** S,D <XRAMβàR.L>

Пример: **MAXL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	0	1	1	1		

Автокод: MAXL4 S, D, MLR,ea,A,R

MAXL4 S, D, MLM,ea,A,R

Формат 5: **MAXL** S,D <R/R.LβàR/R.L>

Пример: **MAXL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	0	1	1	1				

Автокод: MAXL5 S, D, MR,rs,rđ

MAXL5 S, D, MRL,RS,RD

Формат 6: **MAXL** S,D <RβàRC>

Пример: **MAXL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D				R				sc	RC				1	0	de	0	0	1	1	0	1	1	1	1	1	1	1	1

Автокод: MAXL6 S, D, MRC,rs,RC

MAXL6 S, D, MRCd,RC,rd

Формат 7: **MAXL**[.cc] S,D <#16/32àRC/R/R.L>

Пример: **MAXL** R2,R0 0x12345678,R16.L

Код инструкции:

3	3	2	2	2	2	2	2	2	2	2	1	1	1	1	16	15	1	1	1	1	10	9	8	7	6	5	4	3	2	1	0
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7			4	3	2	1											
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	0	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MAXL7 S, D, MIL,cc,#32,RD

MAXL7 S, D, MIC,cc,#16,RC

Формат 8a: <OP2> **MAXL** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MAXL** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	0	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> MAXL S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **MAXL** S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MAXL** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	0	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd					L	0	0	1	#	OP2				

Автокод: PROL <OP2> MAXL S1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> MAXL S1,S2,D <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 MAXL R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S1					D					S2					M	RS					0	0	0	0	0	0	1	1	0	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5					D2/d2					S4/s4					сс		RD					0	сс		1	0	#	OP2				

Автокод: PROL <OP2 сс> MAXL S1,S2,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> MAXL S1,S2,D <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 MAXL R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1					D					S2					M	Rs/Rd					0	0	0	0	0	1	1	0	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> MAXL S1,S2,D <R $\beta$ àRC>



### МАХМ Выбор числа с большим модулем (short) МАХМ

#### Операция:

3-адресный вариант: short s1,s2,d;

if ( $|s1| > |s2|$ ) s1 **à** d

else s2 **à** d

2-адресный вариант: short s,d;

if ( $|s| > |d|$ ) s **à** d

**Описание:** В формате short вычисляется разность модулей операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); если результат отрицательный или равен нулю, то в операнд-приемник d помещается первый операнд-источник (s1 - в 3-адресном варианте; s - в 2-адресном варианте), иначе в операнд-приемник d помещается второй операнд-источник s2 в 3-адресном варианте (в 2-адресном варианте содержимое приемника d не изменяется). В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

#### Признаки результата:

U	N	Z	V	C
√	√	√	-	*

\*C=1, если  $|s1| > |s2|$  (иначе 0).

#### Синтаксис ассемблера и код инструкции:

Формат 1: **МАХМ[.cc]** s1,s2,d

Пример: **МАХМ.ne** R1,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s1	d	s2	0	0	cc	0	0	0	1	0	0	1	1	0	0	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

Автокод: MAXM cc, s1, s2, d

Формат 2: **MAXM[.cc]** #16,s2,d

Пример: **MAXM.ne** 15,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	1	0	0	0	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: MAXM2 cc, #16, s2, d

Формат 3: **MAXM** #16,d

Пример: **MAXM** 0x11,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	0	0	1	1	0	0	1						

Автокод: MAXM3 #16, d

Формат 4: **MAXM** s,d <XRAM $\beta$ àR.L>

Пример: **MAXM** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	0	1	1	0	0	1		

Автокод: MAXM4 s, d, MLR,ea,A,R

MAXM4 s, d, MLM,ea,A,R

Формат 5: **MAXM** s,d <R/R.L $\beta$ àR/R.L>

Пример: **MAXM** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	0	0	1				

Автокод: MAXM5 s, d, MR,rs,rd

MAXM5 s, d, MRL,RS,RD

Формат 6: **MAXM** s,d <R**В**àRC>

Пример: **MAXM** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	1	0	0	1				

Автокод: MAXM6 s, d ,MRC,rs,RC

MAXM6 s, d ,MRCd,RC,rd

Формат 7: **MAXM[.cc]** s,d <#16/32**à**RC/R/R.L>

Пример: **MAXM** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MAXM7 s, d, MIL,cc,#32,RD

MAXM7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **MAXM** s1,s2,d <XRAM**В**àR.L> <YRAM**à**R0>

Пример: LSRL R6,R0,R8 **MAXM** R1,R2,R5 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	0	0	0	1	1	0	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3/s3/#5	D2/d2	S4/s4	AT	mode	A	de	0	0	#	OP2
----------	-------	-------	----	------	---	----	---	---	---	-----

Автокод: PROL <OP2> MAXM s1,s2,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2>MAXM s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 MAXM R1,R2,R5 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs				0	0	0	0	0	0	0	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> MAXM s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> MAXM s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 MAXM R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS				0	0	0	0	0	0	0	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> MAXM s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> MAXM s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 MAXM R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	Rs/Rd				0	0	0	0	0	0	1	1	0	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> MAXM s1,s2,d <R $\beta$ àRC>

**MAXML**      **Выбор числа с большим модулем (long)**      **MAXML**

**Операция:**

3-адресный вариант: long S1, S2, D;

if (|S1| > |S2|) S1 à D

else S2 à D

2-адресный вариант: long S, D;

if (|S| > |D|) S à D

**Описание:** В формате long вычисляется разность модулей операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); если результат отрицательный или равен нулю, то в операнд-приемник D помещается первый операнд-источник (S1 - в 3-адресном варианте; S - в 2-адресном варианте), иначе в операнд-приемник D помещается второй операнд-источник S2 в 3-адресном варианте (в 2-адресном варианте содержимое приемника D не изменяется). В качестве первого операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	*

\*C=1, если |S1| > |S2| (иначе 0).

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MAXML**[.cc] S1, S2, D

Пример: **MAXML**.ne R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	0	1	0	1	1	1	0	0	1			

Автокод: MAXML cc, S1, S2, D

Формат 2: MAXML[.cc] #32, S2, D

Пример: MAXML.ne 15, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	0	1	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: MAXML2 cc, #32, S2, D

Формат 4: MAXML S, D <XRAMβàR.L>

Пример: MAXML R2, R4 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	1	0	0	1		

Автокод: MAXML4 S, D, MLR, ea, A, R

MAXML4 S, D, MLM, ea, A, R

Формат 5: MAXML S, D <R/R.LβàR/R.L>

Пример: MAXML R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	1	0	0	1			

Автокод: MAXML5 S, D, MR,rs,rd

MAXML5 S, D, MRL,RS,RD

Формат 6: **MAXML** S,D <RβàRC>

Пример: **MAXML** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	1	0	0	1				

Автокод: MAXML6 S, D, MRC,rs,RC

MAXML6 S, D, MRCd,RC,rd

Формат 7: **MAXML[.cc]** S,D <#16/32àRC/R/R.L>

Пример: **MAXML** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MAXML7 S, D, MIL,cc,#32,RD

MAXML7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **MAXML** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MAXML** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	1	0	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> MAXML S1,S2,D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> MAXML S1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 MAXML R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	RD/Rd					L	0	0	1	#	OP2					

Автокод: PROL <OP2> MAXML S1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> MAXML S1,S2,D <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 MAXML R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc	RD					0	cc	1	0	#	OP2					

Автокод: PROL <OP2 cc> MAXML S1,S2,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> MAXML S1,S2,D <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 MAXML R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	1	1	0	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> MAXML S1,S2,D <R $\beta$ àRC>



**MIN**
**Выбор меньшего числа (short)**
**MIN**
**Операция:**
3-адресный вариант: short s1,s2,d;

if (s1 &lt; s2) s1 → d

else s2 → d

2-адресный вариант: short s,d;

if (s &lt; d) s → d

**Описание:** Определяется меньшее число: в формате short вычисляется разность операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); если результат положительный или равен нулю, то в операнд-приемник d помещается первый операнд-источник (s1 - в 3-адресном варианте; s - в 2-адресном варианте), иначе в операнд-приемник d помещается второй операнд-источник s2 в 3-адресном варианте (в 2-адресном варианте содержимое приемника d не изменяется). В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:**

OP1

**Форматы:**

1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	*

\*C=1, если s1 &lt; s2 (иначе 0).

**Синтаксис ассемблера и код инструкции:**
Формат 1: **MIN[.cc]** s1,s2,d

 Пример: **MIN.ne** R1,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	0	cc				0	0	0	1	0	0	1	1	0	0	0			

Автокод: MIN cc, s1, s2, d

Формат 2: **MIN**[.cc] #16,s2,d

Пример: **MIN**.ne 15,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	1	0	0	0	1	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: MIN2 cc, #16, s2, d

Формат 3: **MIN** #16,d

Пример: **MIN** 0x11,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	0	0	1	1	0	0	0						

Автокод: MIN3 #16, d

Формат 4: **MIN** s,d <XRAM $\beta$ àR.L>

Пример: **MIN** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	0	1	1	0	0	0		

Автокод: MIN4 s, d, MLR,ea,A,R

MIN4 s, d, MLM,ea,A,R

Формат 5: **MIN** s,d <R/R.L $\beta$ àR/R.L>

Пример: **MIN** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s	d	RS/Rs	L	RD/Rd	1	1	0	0	0	0	1	1	0	0	0
---	---	-------	---	-------	---	---	---	---	---	---	---	---	---	---	---

Автокод: MIN5 s, d, MR,rs,rd  
 MIN5 s, d, MRL,RS,RD

Формат 6: **MIN** s,d <R $\beta$ RC>

Пример: **MIN** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	1	0	0	0				

Автокод: MIN6 s, d ,MRC,rs,RC  
 MIN6 s, d ,MRCd,RC,rd

Формат 7: **MIN**[.cc] s,d <#16/32 $\beta$ RC/R/R.L>

Пример: **MIN** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	1	1	0	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MIN7 s, d, MIL,cc,#32,RD  
 MIN7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **MIN** s1,s2,d <XRAM $\beta$ R.L> <YRAM $\beta$ R0>

Пример: LSRL R6,R0,R8 **MIN** R1,R2,R5 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	0	0	0	1	1	0	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode		A		de		0		0		#		OP2					

Автокод: PROL <OP2> MIN s1,s2,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> MIN s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 MIN R1,R2,R5 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
s1				d				s2				M		RS/Rs				0		0		0		0		0		1		1		0		0		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd				L		0		0		1		#		OP2			

Автокод: PROL <OP2> MIN s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> MIN s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 MIN R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
s1				d				s2				M		RS				0		0		0		0		0		0		1		1		0		0		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0		cc		1		0		#		OP2			

Автокод: PROL <OP2 cc> MIN s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> MIN s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 MIN R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
s1				d				s2				M		Rs/Rd				0		0		0		0		0		1		1		0		0		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32		
S3/s3/#5				D2/d2				S4/s4				0		0		RC				sc		de		1		1		#		OP2			

Автокод: PROL <OP2> MIN s1,s2,d <RbàRC>

**MINL                      Выбор меньшего числа (long)                      MINL**

**Операция:**

3-адресный вариант: long S1, S2, D;

if (S1 < S2) S1 à D

else S2 à D

2-адресный вариант: long S, D;

if (S < D) S à D

**Описание:** Определяется большее число: в формате long вычисляется разность операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); если результат положительный или равен нулю, то в операнд-приемник D помещается первый операнд-источник (S1 - в 3-адресном варианте; S - в 2-адресном варианте), иначе в операнд-приемник D помещается второй операнд-источник S2 в 3-адресном варианте (в 2-адресном варианте содержимое приемника D не изменяется). В качестве первого операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	*

\*C=1, если S1 < S2 (иначе 0).

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MINL[.cc]** S1, S2, D

Пример: **MINL.ne** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	1	1	0	0	0				

Автокод: MINL cc, S1, S2, D

Формат 2: **MINL**[.cc] #32,S2,D

Пример: **MINL**.ne 15,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	1	0	0	1	1	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: MINL2 cc, #32, S2, D

Формат 4: **MINL** S,D <XRAM**β**aR.L>

Пример: **MINL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	1	0	0	0		

Автокод: MINL4 S, D, MLR,ea,A,R

MINL4 S, D, MLM,ea,A,R

Формат 5: **MINL** S,D <R/R.L**β**aR/R.L>

Пример: **MINL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd			1	1	0	0	0	1	1	1	0	0	0					

Автокод: MINL5 S, D, MR,rs,rđ

MINL5 S, D, MRL,RS,RD

Формат 6: **MINL** S,D <R**β**aRC>

Пример: **MINL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	1	0	0	0	0	0	0	0

Автокод: MINL6 S, D, MRC,rs,RC

MINL6 S, D, MRCd,RC,rd

Формат 7: **MINL**[.cc] S,D <#16/32аRC/R/R.L>

Пример: **MINL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	1	0	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MINL7 S, D, MIL,cc,#32,RD

MINL7 S, D, MIC,cc,#16,RC

Формат 8a: <OP2> **MINL** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MINL** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	1	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode					A	de	0	0	#	OP2					

Автокод: PROL <OP2> **MINL** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **MINL** S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MINL** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	1	0	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



S3/s3/#5	D2/d2	S4/s4	AT	RD/Rd	L	0	0	1	#	OP2
----------	-------	-------	----	-------	---	---	---	---	---	-----

Автокод: PROL <OP2> MINL S1, S2, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> **MINL** S1, S2, D <R.L $\beta$ àR.L>

Пример: LSRL.eq R6, R0, R8 **MINL** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	1	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 сс> MINL S1, S2, D <R.L $\beta$ àR.L>

Формат 8d: <OP2> **MINL** S1, S2, D <R $\beta$ àRC>

Пример: LSRL R6, R0, R8 **MINL** R2, R4, R6 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	1	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> MINL S1, S2, D <R $\beta$ àRC>



s1	d	s2	0	0	cc	0	0	0	1	0	0	1	1	0	1	0
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

Автокод: MINM cc, s1, s2, d

Формат 2: **MINM**[.cc] #16,s2,d

Пример: **MINM**.ne 15,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	1	0	0	0	1	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: MINM2 cc, #16, s2, d

Формат 3: **MINM** #16,d

Пример: **MINM** 0x11,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	0	0	1	1	0	1	0						

Автокод: MINM3 #16, d

Формат 4: **MINM** s,d <XRAMβàR.L>

Пример: **MINM** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	0	1	1	0	1	0		

Автокод: MINM4 s, d, MLR,ea,A,R

MINM4 s, d, MLM,ea,A,R

Формат 5: **MINM** s,d <R/R.LβàR/R.L>

Пример: **MINM** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	1	1	0	1	0			

Автокод: MINM5 s, d, MR,rs,rd

MINM5 s, d, MRL,RS,RD

Формат 6: **MINM** s,d <R~~В~~RC>

Пример: **MINM** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	1	0	1	0				

Автокод: MINM6 s, d ,MRC,rs,RC

MINM6 s, d ,MRCd,RC,rd

Формат 7: **MINM[.cc]** s,d <#16/32~~а~~RC/R/R.L>

Пример: **MINM** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	0	0	0	0	0	1	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MINM7 s, d, MIL,cc,#32,RD

MINM7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **MINM** s1,s2,d <XRAM~~В~~R.L> <YRAM~~а~~R0>

Пример: LSRL R6,R0,R8 **MINM** R1,R2,R5 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	0	0	0	1	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2										

Автокод: PROL <OP2> MINM s1,s2,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2>MINM s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 MINM R1,R2,R5 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs			0	0	0	0	0	0	1	1	0	1	0					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd			L	0	0	1	#	OP2										

Автокод: PROL <OP2> MINM s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> MINM s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 MINM R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS			0	0	0	0	0	0	1	1	0	1	0					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD			0	cc	1	0	#	OP2										

Автокод: PROL <OP2 cc> MINM s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> MINM s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 MINM R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	Rs/Rd			0	0	0	0	0	0	1	1	0	1	0					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC			sc	de	1	1	#	OP2									

Автокод: PROL <OP2> MINM s1,s2,d <R**B**ARC>

**MINML Выбор числа с меньшим модулем (long) MINML**
**Операция:**

3-адресный вариант: long S1, S2, D;

if (|S1| < |S2|) S1 à D

else S2 à D

2-адресный вариант: long S, D;

if (|S| < |D|) S à D

**Описание:** В формате long вычисляется разность модулей операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); если результат положительный или равен нулю, то в операнд-приемник D помещается первый операнд-источник (S1 - в 3-адресном варианте; S - в 2-адресном варианте), иначе в операнд-приемник D помещается второй операнд-источник S2 в 3-адресном варианте (в 2-адресном варианте содержимое приемника D не изменяется). В качестве первого операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	*

\*C=1, если |S1| < |S2| (иначе 0).

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MINML[.cc] S1, S2, D**

Пример: **MINML.ne R2, R6, R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	0	1	0	1	1	1	0	1	0			

Автокод: MINML cc, S1, S2, D

Формат 2: **MINML**[.cc] #32,S2,D

Пример: **MINML**.ne 15,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	0	1	1	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: MINML2 cc, #32, S2, D

Формат 4: **MINML** S,D <XRAM $\beta$ AR.L>

Пример: **MINML** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	1	0	1	0		

Автокод: MINML4 S, D, MLR,ea,A,R

MINML4 S, D, MLM,ea,A,R

Формат 5: **MINML** S,D <R/R.L $\beta$ AR/R.L>

Пример: **MINML** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	1	0	1	0				

Автокод: MINML5 S, D, MR,rs,rd

MINML5 S, D, MRL,RS,RD

Формат 6: **MINML** S,D <R $\beta$ ARC>

Пример: **MINML** R2,R6 R7,CCR

Код инструкции:



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	1	0	1	0				

Автокод: MINML6 S, D, MRC,rs,RC

MINML6 S, D, MRCd,RC,rd

Формат 7: **MINML**[.cc] S,D <#16/32аRC/R/R.L>

Пример: **MINML** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MINML7 S, D, MIL,cc,#32,RD

MINML7 S, D, MIC,cc,#16,RC

Формат 8a: <OP2>**MINML** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MINML** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode					A	de	0	0	#	OP2					

Автокод: PROL <OP2> MINML S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **MINML** S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **MINML** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3/s3/#5	D2/d2	S4/s4	AT	RD/Rd	L	0	0	1	#	OP2
----------	-------	-------	----	-------	---	---	---	---	---	-----

Автокод: PROL <OP2> MINML S1, S2, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> **MINML** S1, S2, D <R.L $\beta$ àR.L>

Пример: LSRL.eq R6, R0, R8 **MINML** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	1	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 сс> MINML S1, S2, D <R.L $\beta$ àR.L>

Формат 8d: <OP2> **MINML** S1, S2, D <R $\beta$ àRC>

Пример: LSRL R6, R0, R8 **MINML** R2, R4, R6 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	1	1	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> MINML S1, S2, D <R $\beta$ àRC>

**MOVE**
**Пересылка данных MOVE**

**Операция:** Пересылка данных

<XRAM**B**àR.L>

<YRAM**a**R0>

<R/R.L**B**àR/R.L>

<R**B**àRC>

<#16/#32**a**RC/R/R.L>

**Описание:** Выполняется пересылка данных. В зависимости от выбора источника и приемника возможны следующие типы пересылок:

Тип пересылки	Условное обозначение
Пересылка между X(Y)-памятью и регистром данных	<XRAM <b>B</b> àR.L>
Пересылка из Y-памяти в регистр данных R0	<YRAM <b>a</b> R0>
Пересылка между регистрами данных	<R/R.L <b>B</b> àR/R.L>
Пересылка между регистром данных и управления	<R <b>B</b> àRC>
Запись непосредственного значения в регистр данных или управления	<#16/#32 <b>a</b> RC/R/R.L>

В рамках одной инструкции может быть выполнена одна (форматы 2t, 3, 4, 5, 6, 6t, 7, 7t, 8c, 8d) или две (форматы 8a, 8b) пересылки данных. Формат инструкции определяется типом и количеством одновременно исполняемых пересылок, наличием условия (форматы 2t, 6t, 7, 7t, 8c), а также режимом адресации с непосредственным смещением (формат 7t). Ниже приведено соответствие между форматами инструкций и типами используемых пересылок.

№	Формат	Условие	Операция 1	Операция 2	Пересылка 1	Пересылка 2	Длина кода, 32-р. слов
1	2t	[cc]			R/R.L/RCβàR/R.L/R C		1
2	3				#16àRC		1
3	4		<OP>		XRAMβàR.L		1
4	5		<OP>		R/R.LβàR/R.L		1
5	6		<OP>		RβàRC		1
6	6t	[cc]			XRAMβà R.L		1
7	7	[cc]	<OP>		#16/#32àRC/R/R.L		2
8	7t	[cc]			XRAM(Ai+#16)βàR. L		2
9	8a		<OP2>	<OP1>	XRAMβàR.L	YRAMàR 0	2
10	8b		<OP2>	<OP1>	R/R.Lβà R/R.L	YRAMàR 0	2
11	8c	[cc]	<OP2>	<OP1>	R.LβàR.L		2
12	8d		<OP2>	<OP1>	RβàRC		2

1) Формат 2t

Условная пересылка данных между двумя регистрами регистрового файла или регистром регистрового файла и регистром управления.

2) Формат 3

Безусловная пересылка непосредственного значения #16 в регистр управления.

3) Формат 4

Безусловная пересылка между X(Y)-памятью и регистром данных.

4) Формат 5

Безусловная пересылка между регистрами данных.

5) Формат 6

Безусловная пересылка между регистром данных и управления.

6) Формат 6t

Условная пересылка между X(Y)-памятью и регистром данных (кроме режима адресации с непосредственным смещением).

7) Формат 7

Условная пересылка непосредственного значения #16/#32 в регистр управления или данных.

8) Формат 7t

Условная пересылка между X(Y)-памятью и регистром данных (режим адресации с непосредственным смещением)..

9) Формат 8a

Безусловная пересылка между X-памятью и регистром данных одновременно с пересылкой из Y-памяти в регистр данных R0.L.

10) Формат 8b

Безусловная пересылка между регистрами данных одновременно с пересылкой из Y-памяти в регистр данных R0.L.

11) Формат 8c

Условная пересылка между 32-разрядными регистрами данных.

12) Формат 8d

Безусловная пересылка между регистром данных и управления.

**Тип:** -

**Форматы:** 2t, 3, 4, 5, 6, 6t, 7, 7t, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**

Синтаксис для операции пересылки зависит от того, выполняется она отдельно или на фоне других операций. В первом случае необходимо полное написание команды **MOVE**, во втором достаточно одной буквы **m** либо вовсе обойтись без нее, при этом в командной строке команда пересылки должна стоять после вычислительных команд.

В отличие от других операций, операция **MOVE** в большинстве случаев кодируется не полем кода операции, а кодом формата, а также кодом режима адресации (формат 7t). Направление пересылки определяется полем **de** (direction of exchange): при **de** = 1 производится запись в RF, при **de** = 0 – чтение из него.

Ниже приводятся правила и примеры использования команды **MOVE**.

Формат 2t: **MOVE[.cc]** <R/R.L/RC **↔** R/R.L/RC>

**Примеры:**

**MOVE.eq** R11, R5

**MOVE.ne** R10.L, R4.L

**MOVE.eq** R10.L, AC0

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	sr	L	cc		0	s/S/s/d			sc	d/D/RC/RC		1	0	de	0	1	1	0	0	1	1	1	1	1	1	1	1	1	

**Автокод:** MR cc, s, d

MRL cc, S, D

MRC cc, s, RC

MRCd cc, RC, d

Формат  $3(\#16 \rightarrow RC(AGU, AGU-Y))$  *Безусловная пересылка*  
 непосредственного значения #16 в регистр управления AGU, AGU-Y (MIA):

**MOVE** #16, d

Пример: **MOVE** 0x11, A5

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
d				#16																		0	0	1	1	1	1	0	1	1	0	1

Автокод: MIA #16, d

Формат  $3(\#16 \rightarrow RC(PCU, ALU))$  *Безусловная пересылка*  
 значения #16 в регистр управления PCU, ALU (MIP):

**MOVE** #16, d

Пример: **MOVE** 0x11, A5

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
d				#16																		0	0	1	1	1	1	0	1	1	1	1

Автокод: MIP #16, d

Формат 4: <OP> **[M]** <XRAMB> R.L:

Примеры: <OP> **M** (A0)+, R8

<OP> **M** R10.L, (A0)

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/S				d/D				R				A				mode		u	0	1	de	0	OP								

Автокод: <OP>, MLR, ea, A, R

<OP>, MLM, ea, A, R

Формат 5: <OP> **[M]** R[.L], R[.L]

Пример: <OP> R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s/S	d/D	RS/Rs	L	RD/Rd	1	1	0	0	OP
-----	-----	-------	---	-------	---	---	---	---	----

Автокод: <OP>, MR, rs, rd

<OP>, MRL, RS, RD

Формат 6: <OP> [M] <RβàRC>

Примеры: <OP> R11, M5

<OP> AC1, R2.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	OP										

Автокод: <OP>, MRC, rs, RC

<OP>, MRCd, RC, rd

Формат 6t: MOVE[.cc] <XRAMβàR.L>:

Примеры: MOVE.vs (A0)+, R8

MOVE.ge R10.L, (A0)

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	cc				0	R				A				mode		u	0	1	de	0	1	1	0	0	1	1	1

Автокод: MLR, cc, ea, A, R

MLM, cc, ea, A, R

Формат 7: <OP.cc> <#16/32àRC/R/R.L>

Пример: <OP.cc> 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	OP									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: <OP>, MIL, cc, #32, RD

<OP>, MIC, cc, #16, RC



Формат 7t: **MOVE[.cc]** <(Ai+#16)**Вà**R.L>:

Примеры: **MOVE.ne** (A0+1),R8

**MOVE** R10.L,(A0+0x11)

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	cc				0	R				A				mode		u	0	1	de	0	1	1	0	0	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: MLDR,cc,ea,A,R

MLDM,cc,ea,A,R

Формат 8a: <OP2> <OP1> **[M]** <XRAMВàR.L>  
<YRAMàR0>

Пример: <OP2> <OP1> R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode		A		de	0	0	#	OP2									

Автокод: PROL <OP2> <OP1> <XRAMВàR.L> <YRAMàR0>

Формат 8b: <OP2> <OP1> **[M]** <R/R.LВàR/R.L>  
<YRAMàR0>

Пример: <OP2> <OP1> R8,R15  
(AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	RS/Rs				0	0	0	0	1	0	0	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> <OP1> <R/R.L**В**àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> <OP1> [М] <R.L**В**àR.L>

Пример: <OP2.сс> <OP1> R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	RS				0	0	0	0	0	0	1	0	0	1	0	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				сс		RD				0	сс		1	0	#	OP2							

Автокод: PROL <OP2 сс> <OP1> <R.L**В**àR.L>

Формат 8d: <OP2> <OP1> [М] <R**В**àRC>

Пример: <OP2> <OP1> CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	Rs/Rd				0	0	0	0	1	0	0	1	0	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> <OP1> <R**В**àRC>

**MPF    Умножение дробное со знаком (short)    MPF**

**Операция:**

3-адресный вариант: short s1,s2; long D;

{(s1 [15:0] \* s2 [15:0]), 0} **à** D[31:0]

2-адресный вариант: short s; long D;

{(s [15:0] \* d [15:0]), 0} **à** D[31:0]

**Описание:** В формате short вычисляется произведение операндов-источников (s1,s2 - в 3-адресном варианте; s,D[15:0] - в 2-адресном варианте); 32-разрядный результат сдвигается влево на один разряд и помещается в операнд-приемник D. В освободившийся младший разряд помещается ноль. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP2

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	–

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MPF[.cc]** s1,s2,D

Пример: **MPF.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	0	cc				0	0	0	1	1	1	1	1	1	1	0	1		

Автокод: MPF cc, s1, s2, D

Формат 2: **MPF[.cc]** #16,s2,D

Пример: **MPF.ne** 10,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	0	cc				0	0	1	0	1	1	1	1	1	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: MPF2 cc, #16, s2, D

Формат 3: **MPF** #16, D

Пример: **MPF** 0x10, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D				#16												0	0	1	1	1	1	1	1	1	0	1					

Автокод: MPF3 #16, D

Формат 4: **MPF** s, D <XRAM $\beta$ R.L>

Пример: **MPF** R2, R4 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s		D				R				A				mode		u	0	1	de	0	1	1	1	1	1	1	0	1			

Автокод: MPF4 s, D, MLR, ea, A, R

MPF4 s, D, MLM, ea, A, R

Формат 5: **MPF** s, D <R/R.L $\beta$ R/R.L>

Пример: **MPF** R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s		D				RS/Rs				L	RD/Rd				1	1	0	0	1	1	1	1	1	1	0	1					

Автокод: MPF5 s, D, MR, rs, rd

MPF5 s, D, MRL, RS, RD

Формат 6: **MPF** s, D <R $\beta$ RC>

Пример: **MPF** R2, R6 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	0	1	1	1	1	1	1	0	1			

Автокод: MPF6 s, D, MRC, rs, RC  
MPF6 s, D, MRCd, RC, rd

Формат 7: **MPF** [.cc] s, D <#16/32àRC/R/R.L>

Пример: **MPF** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	1	1	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MPF7 s, D, MIL, cc, #32, RD  
MPF7 s, D, MIC, cc, #16, RC

Формат 8a: **MPF** s3, s4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Пример: **MPF** R4, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				AT	mode				A	de	0	0	0	1	1	1	0	1					

Автокод: PROL MPF s3, s4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **MPF** s3, s4, D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Пример: **MPF** R4, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				AT		RD/Rd				L	0	0	1	0	1	1	1	0	1				

Автокод: PROL MPF s3,s4,D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: **MPF**[.cc] s3,s4,D2 <OP1> <R.L $\beta$ àR.L>

Пример: **MPF**.eq R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				cc		RD				0	cc		1	0	0	1	1	1	0	1			

Автокод: PROL MPF cc s3,s4,D2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: **MPF** s3,s4,D2 <OP1> <R $\beta$ àRC>

Пример: **MPF** R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				s4				0	0	RC				sc	de	1	1	0	1	1	1	0	1				

Автокод: PROL MPF s3,s4,D2 <OP1> <R $\beta$ àRC>

## *MPF2 Парное умножение дробное со знаком (short) MPF2*

### Операция:

3-адресный вариант: long S1, S2, D;

$((S1[15:0] * S2[15:0]) \ll 1)[31:16] \rightarrow D[15:0],$

$((S1[31:16] * S2[31:16]) \ll 1)[31:16] \rightarrow D[31:16]$

2-адресный вариант: long S, D;

$((S[15:0] * D[15:0]) \ll 1)[31:16] \rightarrow D[15:0],$

$(S[31:16] * D[31:16]) \ll 1)[31:16] \rightarrow D[31:16]$

**Описание:** В формате short вычисляется произведение операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте), при этом независимо перемножаются старшие и младшие 16 разрядов операндов-источников. Умножение дробное, знаковое, парное, с сохранением 16-ти старших разрядов результатов в операнде-приемнике D.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 8

### Признаки результата:

U	N	Z	V	C
*	*	*	*	*

\*U=1, если оба произведения, D[31:16] и D[15:0], не нормализованы (иначе 0).

\*N=D[31] – знак произведения D[31:16].

\*Z=1 при нулевых результатах D[31:16] и D[15:0] (иначе 0).

\*V=1 при переполнении D[31:16] или D[15:0] (иначе 0).

\*C=D[15] – знак произведения D[15:0].

### Синтаксис ассемблера и код инструкции:

Формат 1: **MPF2[.cc]** S1, S2, D

Пример: **MPF2.ne** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				1	0	cc			0	0	0	1	1	1	1	1	1	1	1	0	1		

Автокод: MPF2 cc, S1, S2, D

Формат 4: **MPF2** S,D <XRAM $\beta$ àR.L>

Пример: **MPF2** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	1	1	1	1	1	1	1	0	1		

Автокод: MPF24 S, D, MLR,ea,A,R

MPF24 S, D, MLM,ea,A,R

Формат 5: **MPF2** S,D <R/R.L $\beta$ àR/R.L>

Пример: **MPF2** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd			1	1	0	1	1	1	1	1	1	1	1	0	1			

Автокод: MPF25 S, D, MR,rs,rd

MPF25 S, D, MRL,RS,RD

Формат 6: **MPF2** S,D <R $\beta$ àRC>

Пример: **MPF2** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC			1	0	de	1	1	1	1	1	1	1	0	1				

Автокод: MPF26 S, D ,MRC,rs,RC

MPF26 S, D ,MRCd,RC,rd

Формат 8а: **MPF2** S3,S4,D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Пример: **MPF2** R4,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	1	1	1	1	0	1					

Автокод: PROL MPF2 S3,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **MPF2** S3,S4,D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Пример: **MPF2** R4,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	RD/Rd				L	0	0	1	1	1	1	1	0	1					

Автокод: PROL MPF2 S3,S4,D2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: **MPF2**[.cc] S3,S4,D2 <OP1> <R.LβàR.L>

Пример: **MPF2**.eq R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc	RD				0	cc	1	0	1	1	1	1	0	1					

Автокод: PROL MPF2 cc S3,S4,D2 <OP1> <R.LβàR.L>

Формат 8d: **MPF2** S3,S4,D2 <OP1> <RβàRC>

Пример: **MPF2** R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
3/#5		D2				S4				0	0	RC				sc	de	1	1	1	1	1	1	1	0	1			

Автокод: PROL      MPF2 S3, S4, D2      <OP1>      <R**B**àRC>

**MPF2S** Парное умножение дробное со знаком (*short*) *с перестановкой*  
 сомножителей **MPF2S**

**Операция:**

3-адресный вариант: long S1,S2,D;

$((S1[15:0] * S2[31:16]) \ll 1)[31:16] \text{ à } D[15:0],$

$((S1[31:16] * S2[15:0]) \ll 1)[31:16] \text{ à } D[31:16]$

2-адресный вариант: long S,D;

$((S[15:0] * D[31:16]) \ll 1)[31:16] \text{ à } D[15:0],$

$((S[31:16] * D[15:0]) \ll 1)[31:16] \text{ à } D[31:16]$

**Описание:** В формате short вычисляется произведение операндов-источников (S1,S2 - в 3-адресном варианте; S,D - в 2-адресном варианте), при этом независимо перемножаются старшие и младшие 16 разрядов операндов-источников. Умножение дробное, знаковое, парное, с перестановкой сомножителей (по сравнению с операцией MPF2), с сохранением 16-ти старших разрядов результатов в операнде-приемнике D.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*

\*U=1, если оба произведения, D[31:16] и D[15:0], не нормализованы (иначе 0).

\*N=D[31] – знак произведения D[31:16].

\*Z=1 при нулевых результатах D[31:16] и D[15:0] (иначе 0).

\*V=1 при переполнении D[31:16] или D[15:0] (иначе 0).

\*C=D[15] – знак произведения D[15:0].

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MPF2S[.cc] S1,S2,D**

Пример: **MPF2S.ne R2,R6,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				1	0	cc				0	0	0	1	1	1	1	1	1	1	0	1		

Автокод: **MPF2S cc, S1, S2, D**

Формат 4: **MPF2S S,D <XRAMβàR.L>**

Пример: **MPF2S R2,R4 R8,(A0)+**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	1	1	1	1	1	1	1	0	1	

Автокод: **MPF2S4 S, D, MLR,ea,A,R**

**MPF2S4 S, D, MLM,ea,A,R**

Формат 5: **MPF2S S,D <R/R.LβàR/R.L>**

Пример: **MPF2S R2,R4 R8.L,R12.L**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	1	1	1	1	1	1	1	1	0	1		

Автокод: **MPF25 S, D, MR,rs,rd**

**MPF25 S, D, MRL,RS,RD**

Формат 6: **MPF2S S,D <RβàRC>**

Пример: **MPF2S R2,R6 R7,CCR**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	1	1	1	1	1	1	1	1	0	1		

Автокод: **MPF2S6 S, D ,MRC,rs,RC**

**MPF2S6 S, D ,MRCd,RC,rd**

Формат 8а: **MPF2** S3,S4,D2 <OP1> <XRAM**β**àR.L> <YRAMàR0>

Пример: **MPF2** R4,R2,R6 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	1	1	1	1	0	1					

Автокод: PROL MPF2S S3,S4,D2 <OP1> <XRAM**β**àR.L> <YRAMàR0>

Формат 8b: **MPF2S** S3,S4,D2 <OP1> <R/R.L**β**àR/R.L> <YRAMàR0>

Пример: **MPF2S** R4,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	RD/Rd				L	0	0	1	1	1	1	1	0	1					

Автокод: PROL MPF2S S3,S4,D2 <OP2> <R/R.L**β**àR/R.L> <YRAMàR0>

Формат 8с: **MPF2S**[.cc] S3,S4,D2 <OP1> <R.L**β**àR.L>

Пример: **MPF2S**.eq R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc	RD				0	cc	1	0	1	1	1	1	0	1					

Автокод: PROL MPF2S cc S3,S4,D2 <OP1> <R.L**β**àR.L>

Формат 8d: **MPF2S** S3,S4,D2 <OP1> <R**β**àRC>

Пример: **MPF2S** R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	1	1	1	1	1	0	1			

 Автокод: PROL      MPF2S S3,S4,D2      <OP1>      <R**B**àRC>

**MPSS**
**Умножение целое со знаком (short)**
**MPSS**
**Операция:**
3-адресный вариант: short s1,s2,D;

 $s1[15:0] * s[15:0] \rightarrow D[31:0]$ 
2-адресный вариант: short s,D;

 $s[15:0] * d[15:0] \rightarrow D[31:0]$ 

**Описание:** В формате short вычисляется произведение (целочисленное со знаком) операндов-источников (s1,s2 - в 3-адресном варианте; s,D[15:0] - в 2-адресном варианте); 32-разрядный результат помещается в операнд-приемник D. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP2

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**
Формат 1: **MPSS[.cc]** s1,s2,D

 Пример: **MPSS.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	0	cc				0	0	0	1	1	1	1	1	1	1	1	0		

Автокод: MPSS cc, s1, s2, D

Формат 2: **MPSS[.cc]** #16,s2,D

 Пример: **MPSS.ne** 10,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1				D				s2				0	0	cc				0	0	1	0	1	1	1	1	1	1	1	1	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: MPSS2 cc, #16, s2, D

Формат 3: **MPSS** #16,D

Пример: **MPSS** 0x10,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D				#16												0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Автокод: MPSS3 #16, D

Формат 4: **MPSS** s,D <XRAM $\beta$ àR.L>

Пример: **MPSS** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				D				R				A				mode		u	0	1	de	0	1	1	1	1	1	1	1	1	1	0

Автокод: MPSS4 s, D, MLR,ea,A,R

MPSS4 s, D, MLM,ea,A,R

Формат 5: **MPSS** s,D <R/R.L $\beta$ àR/R.L>

Пример: **MPSS** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				D				RS/Rs				L	RD/Rd				1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	0

Автокод: MPSS5 s, D, MR,rs,rđ

MPSS5 s, D, MRL,RS,RD

Формат 6: **MPSS** s,D <R $\beta$ àRC>

Пример: **MPSS** R2,R6 R7,CCR

Код инструкции:



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	0	1	1	1	1	1	1	1	1	1	0	

Автокод: MPSS6 s, D, MRC, rs, RC

MPSS6 s, D, MRCd, RC, rd

Формат 7: **MPSS[.cc]** s, D <#16/32а RC/R/R.L>

Пример: **MPSS** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	1	1	1	1	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MPSS7 s, D, MIL, cc, #32, RD

MPSS7 s, D, MIC, cc, #16, RC

Формат 8а: **MPSS** s3, s4, D2 <OP1> <XRAMβà R.L> <YRAMà R0>

Пример: **MPSS** R4, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				AT	mode				A	de	0	0	0	0	1	1	1	1	0				

Автокод: PROL MPSS s3, s4, D2 <OP1> <XRAMβà R.L> <YRAMà R0>

Формат 8b: **MPSS** s3, s4, D2 <OP1> <R/R.Lβà R/R.L> <YRAMà R0>

Пример: **MPSS** R4, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				AT		RD/Rd				L	0	0	1	0	1	1	1	1	1	0			

Автокод: PROL MPSS s3,s4,D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: **MPSS**[.cc] s3,s4,D2 <OP1> <R.L $\beta$ àR.L>

Пример: **MPSS**.eq R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				cc	RD				0	cc	1	0	0	1	1	1	1	0					

Автокод: PROL MPSS cc s3,s4,D2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: **MPSS** s3,s4,D2 <OP1> <R $\beta$ àRC>

Пример: **MPSS** R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				0	0	RC				sc	de	1	1	0	1	1	1	1	0				

Автокод: PROL MPSS s3,s4,D2 <OP1> <R $\beta$ àRC>

**MPUU**
*Умножение целое без знака (short)*
**MPUU**
**Операция:**
3-адресный вариант: short s1,s2,D;

 $s1 [15:0] * s2 [15:0] \rightarrow D[31:0]$ 
2-адресный вариант: short s,D;

 $s [15:0] * d [15:0] \rightarrow D[31:0]$ 

**Описание:** В формате short вычисляется произведение (целочисленное без знака) операндов-источников (s1,s2 - в 3-адресном варианте; s,D[15:0] - в 2-адресном варианте); 32-разрядный результат помещается в операнд-приемник D. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:**

OP2

**Форматы:**

1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	-	√	-	-

**Синтаксис ассемблера и код инструкции:**
Формат 1: **MPUU[.cc]** s1,s2,D

 Пример: **MPUU.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	0	cc				0	0	0	1	1	1	1	1	0	1	1			

Автокод: MPUU cc, s1, s2, D

Формат 2: **MPUU[.cc]** #16,s2,D

 Пример: **MPUU.ne** 10,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				D				s2				0	0	cc				0	0	1	0	1	1	1	1	1	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: MPUU2 cc, #16, s2, D

Формат 3: **MPUU** #16,D

Пример: **MPUU** 0x10,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D				#16											0	0	1	1	1	1	1	1	0	1	1						

Автокод: MPUU3 #16, D

Формат 4: **MPUU** s,D <XRAM $\beta$ àR.L>

Пример: **MPUU** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s		D				R			A		mode		u	0	1	de	0	1	1	1	1	1	0	1	1						

Автокод: MPUU4 s, D, MLR,ea,A,R

MPUU4 s, D, MLM,ea,A,R

Формат 5: **MPUU** s,D <R/R.L $\beta$ àR/R.L>

Пример: **MPUU** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s		D				RS/Rs			L	RD/Rd			1	1	0	0	1	1	1	1	0	1	1								

Автокод: MPUU5 s, D, MR,rs,rd

MPUU5 s, D, MRL,RS,RD

Формат 6: **MPUU** s,D <R $\beta$ àRC>

Пример: **MPUU** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	0	1	1	1	1	0	1	1				

Автокод: MPUU6 s, D, MRC, rs, RC

MPUU6 s, D, MRCd, RC, rd

Формат 7: MPUU[.cc] s, D <#16/32а RC/R/R.L>

Пример: MPUU R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	1	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MPUU7 s, D, MIL, cc, #32, RD

MPUU7 s, D, MIC, cc, #16, RC

Формат 8а: MPUU s3, s4, D2 <OP1> <XRAMβа R.L> <YRAMа R0>

Пример: MPUU R4, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				AT	mode				A	de	0	0	0	1	1	0	1	1					

Автокод: PROL MPUU s3, s4, D2 <OP1> <XRAMβа R.L> <YRAMа R0>

Формат 8b: MPUU s3, s4, D2 <OP1> <R/R.Lβа R/R.L> <YRAMа R0>

Пример: MPUU R4, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				AT		RD/Rd				L	0	0	1	0	1	1	0	1	1				

Автокод: PROL MPUU s3,s4,D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: MPUU[.cc] s3,s4,D2 <OP1> <R.L $\beta$ àR.L>

Пример: MPUU.eq R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				cc		RD				0	cc		1	0	0	1	1	0	1	1			

Автокод: PROL MPUU cc s3,s4,D2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: MPUU s3,s4,D2 <OP1> <R $\beta$ àRC>

Пример: MPUU R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				0	0	RC				sc	de	1	1	0	1	1	0	1	1				

Автокод: PROL MPUU s3,s4,D2 <OP1> <R $\beta$ àRC>

**MPX** Умножение дробное комплексное, второй операнд – комплексно-сопряженный **MPX**

**Операция:**

3-адресный вариант: (X8) S1, S2; (X16) D;

$$((S1[31:24]*S2[31:24])\ll 1) + ((S1[15:8]*S2[15:8])\ll 1) \rightarrow D[31:16]$$

$$((S1[15:8]*S2[31:24])\ll 1) - ((S1[31:24]*S2[15:8])\ll 1) \rightarrow D[15:0]$$

2-адресный вариант: (X8) S; (X16) D;

$$((S[31:24]*D[31:24])\ll 1) + ((S[15:8]*D[15:8])\ll 1) \rightarrow D[31:16]$$

$$((S[15:8]*D[31:24])\ll 1) - ((S[31:24]*D[15:8])\ll 1) \rightarrow D[15:0]$$

**Описание:** Вычисляется произведение двух комплексных дробных чисел, представленных в формате X8. Первым сомножителем является операнд S1 (в 3-адресном варианте) или S (в 2-адресном варианте); вторым сомножителем является комплексное сопряжение операнда S2 (в 3-адресном варианте) или D (в 2-адресном варианте). Результат в формате X16 помещается в регистр D.

**Тип:** OP2

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*

\*U= Ur&Ui.

\*N= Nr

\*Z= Zr

\*V= Vr|Vi

\*C – бит не контролируется

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MPX[.cc] S1,S2,D**

Пример: **MPX.ne R2,R6,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	0	1	1	1	1	0	1	1	0			

Автокод: **MPX cc, S1, S2, D**

Формат 2: **MPX[.cc] #16,S2,D**

Пример: **MPX.ne 10,R5,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
x				D				S2				0	0	cc				0	0	1	0	1	1	1	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: **MPX2 cc, #16, S2, D**

Формат 4: **MPX S,D <XRAMβàR.L>**

Пример: **MPX R2,R4 R8,(A0)+**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	1	1	1	0	1	1	0		

Автокод: **MPX4 S, D, MLR,ea,A,R**

**MPX4 S, D, MLM,ea,A,R**

Формат 5: **MPX S,D <R/R.LβàR/R.L>**

Пример: **MPX R2,R4 R8.L,R12.L**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	1	1	0	1	1	0				

Автокод: **MPX5 S, D, MR,rs,rd**

**MPX5 S, D, MRL,RS,RD**



Формат 6: **MPX** S, D <R**β**RC>

Пример: **MPX** R2, R6 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	1	1	0	1	1	0				

Автокод: MPX6 S, D, MRC, rs, RC

MPX6 S, D, MRCd, RC, rd

Формат 7: **MPX[.cc]** S, D <#16/32**β**RC/R/R.L>

Пример: **MPX** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MPX7 S, D, MIL, cc, #32, RD

MPX7 S, D, MIC, cc, #16, RC

Формат 8а: **MPX** S3, S4, D2 <OP1> <XRAM**β**R.L> <YRAM**β**R0>

Пример: **MPX** R4, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	0	1	0	1	1	0					

Автокод: PROL MPX S3, S4, D2 <OP1> <XRAM**β**R.L> <YRAM**β**R0>

Формат 8b: **MPX** S3, S4, D2 <OP1> <R/R.L**β**R/R.L> <YRAM**β**R0>

Пример: **MPX** R4, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				AT	RD/Rd				L	0	0	1	0	1	0	1	1	0					

Автокод: PROL MPX s3,s4,D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: **MPX**[.cc] s3,s4,D2 <OP1> <R.L $\beta$ àR.L>

Пример: **MPX**.eq R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				cc	RD				0	cc	1	0	0	1	0	1	1	0					

Автокод: PROL MPX cc s3,s4,D2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: **MPX** s3,s4,D2 <OP1> <R $\beta$ àRC>

Пример: **MPX** R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				D2				s4				0	0	RC				sc	de	1	1	0	1	0	1	1	0				

Автокод: PROL MPX s3,s4,D2 <OP1> <R $\beta$ àRC>

**MPYL**

 Умножение целое со знаком (*long*)

**MPYL**
**Операция:**
3-адресный вариант: `long S1, S2, D;`
 $S1[31:0] * S2[31:0] \rightarrow \{S2[31:0], D[31:0]\}$ 
2-адресный вариант: `long S, D;`
 $S[31:0] * D[31:0] \rightarrow \{S[31:0], D[31:0]\}$ 

**Описание:** В формате `long` вычисляется произведение операндов-источников (`S1, S2` - в 3-адресном варианте; `S, D` - в 2-адресном варианте). Умножение целочисленное со знаком, результат – 64-разрядный (`__Int64`). Старшие 32 разряда результата сохраняются в операнде-приемнике по адресу `S2` (в 3-адресном варианте) или `S` (в 2-адресном варианте), младшие 32 разряда – по адресу `D`.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	√

**Синтаксис ассемблера и код инструкции:**
Формат 1: **MPYL**[.cc] S1, S2, D

 Пример: **MPYL.ne** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S1					D					S2					1	0	cc					0	0	0	1	1	1	1	1	1	0	1

 Автокод: `MPYL cc, S1, S2, D`
Формат 4: **MPYL** S, D <XRAM $\beta$  R.L>

 Пример: **MPYL** R2, R4 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	1	1	1	1	1	1	1	1	0	1	

Автокод: MPYL4 S, D, MLR, ea, A, R

MPYL4 S, D, MLM, ea, A, R

Формат 5: **MPYL** S, D <R/R.L $\beta$ àR/R.L>

Пример: **MPYL** R2, R4 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	1	1	1	1	1	1	1	1	0	1		

Автокод: MPYL5 S, D, MR, rs, rd

MPYL5 S, D, MRL, RS, RD

Формат 6: **MPYL** S, D <R $\beta$ àRC>

Пример: **MPYL** R2, R6 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	1	1	1	1	1	1	1	1	0	1		

Автокод: MPYL6 S, D, MRC, rs, RC

MPYL6 S, D, MRCd, RC, rd

Формат 8a: **MPYL** S3, S4, D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Пример: **MPYL** R4, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	1	1	1	1	1	1	0	1			

Автокод: PROL MPYL S3, S4, D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: **MPYL** S3, S4, D2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: **MPYL** R4,R2,R6 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	RD/Rd				L	0	0	1	1	1	1	1	1	0	1				

Автокод: PROL MPYL S3,S4,D2 <OP2> <R/R.L~~β~~R/R.L> <YRAM~~α~~R0>

Формат 8с: **MPYL**[.cc] S3,S4,D2 <OP1> <R.L~~β~~R.L>

Пример: **MPYL.eq** R4,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc	RD				0	cc	1	0	1	1	1	1	1	0	1				

Автокод: PROL MPYL cc S3,S4,D2 <OP1> <R.L~~β~~R.L>

Формат 8d: **MPYL** S3,S4,D2 <OP1> <R~~β~~RC>

Пример: **MPYL** R4,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	1	1	1	1	0	1				

Автокод: PROL MPYL S3,S4,D2 <OP1> <R~~β~~RC>

**MSKG Формирование маски (short)**
**MSKG**
**Операция:**

3-адресный вариант: short s1,s2,d;

$$\{\{s2,0xffff\} \ll s1[5:0]\}[31:16] \text{ à } d$$

2-адресный вариант: short s,d;

$$\{\{d,0xffff\} \ll s[5:0]\}[31:16] \text{ à } d$$

**Описание:** Операнд-источник s2 (в 3-адресном варианте) или d (в 2-адресном варианте) сдвигается влево на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте) и результат помещается в операнд-приемник d. Вытолкнутые за пределы разрядной сетки разряды теряются. Младшие освободившиеся в результате сдвига разряды заполняются единицами.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **MSKG[.cc] s1/#5,s2,d**

Примеры: **MSKG.ne R1,R5,R12**

**MSKG.ne 11,R5,R12**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					d					s2					#	0	cc					0	0	0	1	1	1	0	1	0	1	0

*Автокод:* **MSKG cc, s1/#5, s2, d**

Формат 4: **MSKG** s/#5,d <XRAM $\beta$ àR.L>

Примеры: **MSKG** R1,R2 R8,(A0)+

**MSKG** 11,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					d					R			A			mode		u	0	1	de	#	1	1	0	1	0	1	0		

Автокод: **MSKG4** s, d, MLR,ea,A,R

**MSKG4** s, d, MLM,ea,A,R

Формат 5: **MSKG** s,d <R/R.L $\beta$ àR/R.L>

Примеры: **MSKG** R1,R2 R8.L,R12.L

**MSKG** 15,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					d					RS/Rs			L	RD/Rd		1	1	0	#	1	1	0	1	0	1	0	1	0			

Автокод: **MSKG5** s, d, MR,rs,rđ

**MSKG5** s, d, MRL,RS,RD

Формат 6: **MSKG** s,d <R $\beta$ àRC>

Примеры: **MSKG** R1,R5 R7,CCR

**MSKG** 1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5					d					R			sc	RC		1	0	de	#	1	1	0	1	0	1	0	1	0			

Автокод: **MSKG6** s, d ,MRC,rs,RC

**MSKG6** s, d ,MRCd,RC,rđ

Формат 7: **MSKG[.cc]** s,d <#16/32 $\beta$ àRC/R/R.L>

Пример: **MSKG** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s	d	RD/Rd/RC	L/sc	sr	cc	1	1	1	0	1	1	0	1	0	1	0
---	---	----------	------	----	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MSKG7 s, d, MIL, cc, #32, RD

MSKG7 s, d, MIC, cc, #16, RC

Формат 8a: **MSKG** s3/#5, s4, d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: **MSKG** R1, R2, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

**MSKG** 1, R2, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	mode				A	de	0	0	#	0	1	0	1	0					

Автокод: PROL MSKG s3/#5, s4, d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: **MSKG** s3/#5, s4, d2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: **MSKG** R1, R2, R5 ANDL R6, R0, R8 R8, R15 (AT), R0

**MSKG** 1, R2, R5 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				AT	RD/Rd				L	0	0	1	#	0	1	0	1	0					

Автокод: PROL MSKG s3/#5, s4, d2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>



Формат 8с: **MSKG**[.cc] s3/#5,s4,d2 <OP1> <R.L**B**àR.L>

Примеры: **MSKG**.eq R1,R2,R5 ANDL R6,R0,R8 R8.L,R16.L

**MSKG**.eq 1,R2,R5 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				cc	RD				0	cc	1	0	#	0	1	0	1	0					

Автокод: PROL MSKG cc s3,s4,d2 <OP1> <R.L**B**àR.L>

Формат 8d: **MSKG** s3/#5,s4,d2 <OP1> <R**B**àRC>

Примеры: **MSKG** R1,R2,R5 ANDL R6,R0,R8 CCR,R17

**MSKG** 1,R2,R5 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				d2				s4				0	0	RC				sc	de	1	1	#	0	1	0	1	0				

Автокод: PROL MSKG s3/#5,s4,d2 <OP1> <R**B**àRC>

**MSKGL**
**Формирование маски (long)**
**MSKGL**
**Операция:**
3-адресный вариант: short s1; long S2,D;

 $\{\{S2,0xffffffff\} \ll s1[5:0]\}[63:32] \text{ à } D$ 
2-адресный вариант: short s; long D;

 $\{\{D,0xffffffff\} \ll s[5:0]\}[63:32] \text{ à } D$ 

**Описание:** Операнд-источник S2 (в 3-адресном варианте) или D (в 2-адресном варианте) сдвигается влево на количество разрядов, заданное младшими шестью разрядами s1 (в 3-адресном варианте) или s (в 2-адресном варианте) и результат помещается в операнд-приемник D. Вытолкнутые за пределы разрядной сетки разряды теряются. Младшие освободившиеся в результате сдвига разряды заполняются единицами.

В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #5.

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**
Формат 1: **MSKGL**[.cc] s1/#5,S2,D

Примеры: **MSKGL.ne** R2,R6,R12

**MSKGL.ne** 17,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1/#5					D					S2					#	0	cc					0	0	0	1	1	1	0	1	0	1	1

Автокод: MSKGL cc, s1/#5, S2, D

Формат 4: **MSKGL** s/#5,D <XRAM $\beta$ aR.L>

Примеры: **MSKGL** R2,R4 R8,(A0)+

**MSKGL** 17,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				D				R				A				mode		u	0	1	de	#	1	1	0	1	0	1	1		

Автокод: MSKGL4 s, D, MLR,ea,A,R

MSKGL4 s, D, MLM,ea,A,R

Формат 5: **MSKGL** s,D <R/R.L $\beta$ aR/R.L>

Примеры: **MSKGL** R2,R4 R8.L,R12.L

**MSKGL** 15,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				D				RS/Rs				L	RD/Rd				1	1	0	#	1	1	0	1	0	1	1				

Автокод: MSKGL5 s, D, MR,rs,rd

MSKGL5 s, D, MRL,RS,RD

Формат 6: **MSKGL** S,D <R $\beta$ aRC>

Примеры: **MSKGL** R2,R6 R7,CCR

**MSKGL** 1,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s/#5				D				R				sc	RC				1	0	de	#	1	1	0	1	0	1	1				

Автокод: MSKGL6 S, D ,MRC,rs,RC

MSKGL6 S, D ,MRCd,RC,rd

Формат 7: **MSKGL**[.cc] s,D <#16/32 $\beta$ aRC/R/R.L>

Пример: **MSKGL** R2,R0 0x12345678,R16.L

Код инструкции:

3	3	2	2	2	2	2	2	2	2	2	1	1	1	16	15	1	1	1	1	10	9	8	7	6	5	4	3	2	1	0
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7			4	3	2	1										
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	1	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: MSKGL7 s, D, MIL, cc, #32, RD

MSKGL7 s, D, MIC, cc, #16, RC

Формат 8a: **MSKGL** s3/#5, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **MSKGL** R1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

**MSKGL** 1, R2, R6 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT	mode				A	de	0	0	#	0	1	0	1	1					

Автокод: PROL MSKGL s3/#5, S4, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **MSKGL** s3/#5, S4, D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **MSKGL** R1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

**MSKGL** 1, R2, R6 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				AT	RD/Rd				L	0	0	1	#	0	1	0	1	1					

Автокод: PROL MSKGL s3/#5, S4, D2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с: **MSKGL**[.cc] s3/#5,S4,D2 <OP1> <R.L**β**R.L>

Примеры: **MSKGL**.eq R1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L  
**MSKGL**.eq 1,R2,R6 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				cc	RD				0	cc	1	0	#	0	1	0	1	1					

Автокод: PROL MSKGL cc s3,S4,D2 <OP1> <R.L**β**R.L>

Формат 8d: **MSKGL** s3/#5,S4,D2 <OP1> <R**β**RC>

Примеры: **MSKGL** R1,R2,R6 ANDL R6,R0,R8 CCR,R17  
**MSKGL** 1,R2,R6 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3/#5				D2				S4				0	0	RC				sc	de	1	1	#	0	1	0	1	1				

Автокод: PROL MSKGL s3/#5,S4,D2 <OP1> <R**β**RC>

**NEG**
*Изменение знака (short)*
**NEG**
**Операция:** short s,d; 0 - s **à** d

**Описание:** В формате short из нуля вычитается операнд-источник s, результат помещается в операнд-приемник d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	-

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** **NEG**[.cc] s,d

**Пример:** **NEG.ne** R1,R5

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	0	cc				0	0	0	1	0	0	1	0	0	0	1		

**Автокод:** **NEG** cc, s, d

**Формат 4:** **NEG** s,d <XRAM**à**R.L>

**Пример:** **NEG** R1,R2 R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode	u	0	1	de	0	0	0	1	0	0	0	1			

**Автокод:** **NEG4** s, d, MLR,ea,A,R

**NEG4** s, d, MLM,ea,A,R

**Формат 5:** **NEG** s,d <R/R.L**à**R/R.L>

**Пример:** **NEG** R1,R2 R8.L,R12.L

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	1	0	0	0	0	1	0	

Автокод: NEG5 s, d, MR,rs,rd

NEG5 s, d, MRL,RS,RD

Формат 6: **NEG** s,d <RβàRC>

Пример: **NEG** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	0	0	0	1	0	0	1	

Автокод: NEG6 s, d ,MRC,rs,RC

NEG6 s, d ,MRCd,RC,rd

Формат 7: **NEG[.cc]** s,d #16/32àRC/R/R.L>

Пример: **NEG** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	0	0	0	1	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: NEG7 s, d, MIL,cc,#32,RD

NEG7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **NEG[s]** s,d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **NEG** R1,R2 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **NEGs** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	M	R				u	0	0	0	0	0	0	0	0	1	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de	0	0	#	OP2								

Автокод: PROL <OP2> NEG[s] s,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **NEG[s]** s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **NEG** R1,R2 R8,R15 (AT),R0

LSRL R5,R0,R8 **NEGs** R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	M	RS/Rs				0	0	0	0	0	0	1	0	0	0	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> NEG[s] s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **NEG[s]** s,d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 **NEG** R1,R2 R8.L,R16.L

LSRL.eq R5,R0,R8 **NEGs** R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	M	RS				0	0	0	0	0	0	0	1	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> NEG[s] s,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> **NEG[s]** s,d <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 **NEG** R1,R2 CCR,R17

LSRL R5,R0,R8 **NEGs** R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---



s	d	0	0	0	0	0	0	M	Rs/Rd	0	0	0	0	0	0	1	0	0	0	1
---	---	---	---	---	---	---	---	---	-------	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: PROL      <OP2>      NEG[s]      s, d      <R**B**aRC>

**NEGL**
**Изменение знака (long)**
**NEGL**
**Операция:** long S,D; 0 - S **à** D

**Описание:** В формате long из нуля вычитается операнд-источник S, результат помещается в операнд-приемник D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	-

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** **NEGL**[.cc] S,D

**Пример:** **NEGL.ne** R2,R6

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	0	1	0	1	1	0	0	0	1		

**Автокод:** **NEGL** cc, S, D

**Формат 4:** **NEGL** S,D <XRAM**à**R.L>

**Пример:** **NEGL** R2,R4 R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	1	0	0	0	1			

**Автокод:** **NEGL4** S, D, MLR,ea,A,R

**NEGL4** S, D, MLM,ea,A,R

**Формат 5:** **NEGL** S,D <R/R.L**à**R/R.L>

**Пример:** **NEGL** R2,R4 R8.L,R12.L

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	0	0	0	1	0	0	0	1

Автокод: NEGL5 S, D, MR,rs,rd

NEGL5 S, D, MRL,RS,RD

Формат 6: **NEGL** S,D <R**β**RC>

Пример: **NEGL** R2,R4 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	0	0	0	1	0	0	1	

Автокод: NEGL6 S, D ,MRC,rs,RC

NEGL6 S, D ,MRCd,RC,rd

Формат 7: **NEGL[.cc]** S,D #16/32**à**RC/R/R.L>

Пример: **NEGL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	0	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: NEGL7 S, D, MIL,cc,#32,RD

NEGL7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **NEGL[s]** S,D <XRAM**β**R.L> <YRAM**à**R0>

Примеры: LSRL R6,R0,R8 **NEGL** R2,R4 R8,(A0)+ (AT),R0

LSRL R6,R0,R8 **NEGLs** R2,R4 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	M	R				u	0	0	0	0	0	0	1	1	0	0	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de	0	0	#	OP2								

Автокод: PROL <OP2> NEGL[s] S,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> NEGL[s] S,D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 NEGL R2,R4 R8,R15 (AT),R0

LSRL R6,R0,R8 NEGLs R2,R4 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	M	RS/Rs				0	0	0	0	0	1	1	0	0	0	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> NEGL[s] S,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> NEGL[s] S,D <R.LβàR.L>

Примеры: LSRL.eq R6,R0,R8 NEGL R2,R4 R8.L,R16.L

LSRL.eq R6,R0,R8 NEGLs R2,R4 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	M	RS				0	0	0	0	0	0	1	1	0	0	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> NEGL[s] S,D <R.LβàR.L>

Формат 8d: <OP2> NEGL[s] S,D <RβàRC>

Примеры: LSRL R6,R0,R8 NEGL R2,R4 CCR,R17

LSRL R6,R0,R8 NEGLs R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	0	0	0	0	0	M	Rs/Rd	0	0	0	0	0	1	1	0	0	0	1
---	---	---	---	---	---	---	---	-------	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> NEGL[s] S,D <R̂aRC>

**NOP**
*Пустая операция*
**NOP**
**Операция:** Пустая операция

**Описание:** Пустая операция. Изменяется только состояние программного счетчика PC. В параллельной инструкции (формат 8) NOP может быть операцией типа OP1 или OP2.

**Тип:** OP1, OP2

**Форматы:** 3, 8

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**
**Формат 3 (OP1):** NOP

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

*Автокод:* NOP

**Формат 8a (OP1):** <OP2> NOP <XRAM $\beta$ àR.L> <YRAMàR0>

*Примеры:* LSRL R5,R0,R8 NOP R8,(A0)+ (AT),R0

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	R				u	0	0	0	0	1	0	0	0	0	0	0	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

*Автокод:* PROL <OP2> NOP <XRAM $\beta$ àR.L> <YRAMàR0>

**Формат 8a (OP2):** NOP <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: **NOP** ORL R5, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
s				d				0	0	0	0	0	0	0	AT	mode				A	de	0	0	0	0	0	0	0	0	0	0	0

Автокод: PROL NOP <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b (OP1): <OP2> **NOP** <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R5, R0, R8 **NOP** R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	RS/Rs				0	0	0	0	1	0	0	0	0	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> NOP <R/R.LβàR/R.L> <YRAMàR0>

Формат 8b (OP2): **NOP** <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **NOP** ORL R5, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
s				d				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	0	0	0	0	0	0	0	0	0	0

Автокод: PROL NOP <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с (OP1): <OP2.cc> **NOP** <R.LβàR.L>

Примеры: LSRL.eq R5, R0, R8 **NOP** R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				0	0	0	0	0	0	RS				0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2								

Автокод: PROL <OP2 cc> NOP <R.L $\beta$ àR.L>

Формат 8с (OP2): NOP [.cc] <OP1> <R.L $\beta$ àR.L>

Примеры: NOP.eq ORL R5,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS			0	0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
s				d				0	0	0	0	0	cc		RD			0	cc		1	0	0	0	0	0	0	0	0	0	0	0

Автокод: PROL NOP cc <OP1> <R.L $\beta$ àR.L>

Формат 8d (OP1): <OP2> NOP <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 NOP CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				0	0	0	0	0	0	Rs/Rd				0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC			sc	de	1	1	#	OP2									

Автокод: PROL <OP2> NOP <R $\beta$ àRC>

Формат 8d (OP2): NOP <OP1> <R $\beta$ àRC>

Примеры: NOP ORL R5,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	Rs/Rd			0	0	0	0	OP1											



63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s				d				0	0	0	0	0	0	0	0	RC				sc	de	1	1	0	0	0	0	0	0	0	

Автокод: PROL      NOP      <OP1>      <R**3**a RC>

**NOT**

Логическое отрицание (short)

**NOT**
**Операция:** short s,d; ~ s à d

**Описание:** В формате short производится побитная логическая инверсия операнда-источника s, результат помещается в операнд-приемник d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** NOT[.cc] s,d

**Пример:** NOT.ne R1,R5

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	0	cc				0	0	0	1	1	0	0	1	0	0	1		

**Автокод:** NOT cc, s, d

**Формат 4:** NOT s,d <XRAMβàR.L>

**Пример:** NOT R1,R2 R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode	u	0	1	de	0	1	0	0	1	0	0	1			

**Автокод:** NOT4 s, d, MLR,ea,A,R

NOT4 s, d, MLM,ea,A,R

**Формат 5:** NOT s,d <R/R.LβàR/R.L>

**Пример:** NOT R1,R2 R8.L,R12.L

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	1	0	0	1	0	0	1	

Автокод: NOT5 s, d, MR,rs,rd

NOT5 s, d, MRL,RS,RD

Формат 6: **NOT** s,d <RβàRC>

Пример: **NOT** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	1	0	0	1	0	0	1	

Автокод: NOT6 s, d ,MRC,rs,RC

NOT6 s, d ,MRCd,RC,rd

Формат 7: **NOT**[.cc] s,d #16/32àRC/R/R.L>

Пример: **NOT** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	1	0	0	1	0	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: NOT7 s, d, MIL,cc,#32,RD

NOT7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **NOT** s,d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **NOT** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	R				u	0	0	0	0	1	0	0	0	1	0	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> NOT s,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> NOT s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 NOT R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	RS/Rs				0	0	0	0	1	0	0	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> NOT s,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> NOT s,d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 NOT R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	RS				0	0	0	0	0	1	0	0	1	0	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0	cc	1	0	#	OP2								

Автокод: PROL <OP2 cc> NOT s,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> NOT s,d <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 NOT R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	Rs/Rd				0	0	0	0	1	0	0	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> NOT s,d <R $\beta$ àRC>

**NOTL**
*Логическое отрицание (long)*
**NOTL**
**Операция:** long S,D; ~ S à D

**Описание:** В формате long производится побитная логическая инверсия операнда-источника S, результат помещается в операнд-приемник D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**
**Синтаксис ассемблера и код инструкции:**
Формат 1: **NOTL**[.cc] S,D

U	N	Z	V	C
√	√	√	-	-

*Пример:* **NOTL**.ne R2,R6

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	0	cc				0	0	0	1	1	0	1	1	0	0	0	1

*Автокод:* NOTL cc, S, D

Формат 4: **NOTL** S,D <XRAM~~В~~àR.L>

*Пример:* **NOTL** R2,R4 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	1	0	1	1	0	0	1			

*Автокод:* NOTL4 S, D, MLR,ea,A,R

NOTL4 S, D, MLM,ea,A,R

Формат 5: **NOTL** S,D <R/R.L~~В~~àR/R.L>

*Пример:* **NOTL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	1	0	0	1				

Автокод: NOTL5 S, D, MR,rs,rd  
 NOTL5 S, D, MRL,RS,RD

Формат 6: **NOTL** S,D <R**β**RC>

Пример: **NOTL** R2,R4 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	1	0	0	1				

Автокод: NOTL6 S, D ,MRC,rs,RC  
 NOTL6 S, D ,MRCd,RC,rd

Формат 7: **NOTL**[.cc] S,D #16/32**β**RC/R/R.L>

Пример: **NOTL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: NOTL7 S, D, MIL,cc,#32,RD  
 NOTL7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **NOTL** S,D <XRAM**β**R.L> <YRAM**β**R0>

Примеры: LSRL R6,R0,R8 **NOTL** R2,R4 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	1	0	1	1	0	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de		0		0		#		OP2				

Автокод: PROL <OP2> NOTL S,D <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> NOTL S,D <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Примеры: LSRL R6,R0,R8 NOTL R2,R4 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
S				D				0				0				0				0				RS/Rs				0		0		0		1		0		1		1		0		0		1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L		0		0		1		#		OP2				

Автокод: PROL <OP2> NOTL S,D <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> NOTL S,D <R.L $\beta$ R.L>

Примеры: LSRL.eq R6,R0,R8 NOTL R2,R4 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																		
S				D				0				0				0				0				RS				0		0		0		0		1		0		1		1		0		0		1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0		cc		1		0		#		OP2				

Автокод: PROL <OP2 cc> NOTL S,D <R.L $\beta$ R.L>

Формат 8d: <OP2> NOTL S,D <R $\beta$ RC>

Примеры: LSRL R6,R0,R8 NOTL R2,R4 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																		
S				D				0				0				0				M				Rs/Rd				0		0		0		0		1		0		1		1		0		0		1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5				D2/d2				S4/s4				0		0		RC			sc		de		1		1		#		OP2			

Автокод: PROL <OP2> NOTL S,D <RВàRC>



**OR**
**Логическое ИЛИ (short)**
**OR**
**Операция:**

3-адресный вариант: short s1,s2,d;     s1 | s2 à d

2-адресный вариант: short s,d;     s | d à d

**Описание:** В формате short выполняется побитное логическое сложение операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **OR[.cc]** s1,s2,d

Примеры: **OR.ne** R1,R5,R12

**OR** R1,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	0	1	1	0	0	0	1	0	1			

Автокод: OR cc, s1, s2, d

Формат 2: **OR[.cc]** #16,s2,d

Пример: **OR.ne** 15,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	1	0	1	0	0	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: OR2 сс, #16, s2, d

Формат 3: OR #16, d

Пример: OR 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16												0	0	1	1	1	0	0	0	1	0	1					

Автокод: OR3 #16, d

Формат 4: OR s, d <XRAM $\beta$ aR.L>

Пример: OR R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				A				mode		u	0	1	de	0	1	0	0	0	0	1	0	1					

Автокод: OR4 s, d, MLR, ea, A, R

OR4 s, d, MLM, ea, A, R

Формат 5: OR s, d <R/R.L $\beta$ aR/R.L>

Пример: OR R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	0	1	0	1				

Автокод: OR5 s, d, MR, rs, rd

OR5 s, d, MRL, RS, RD

Формат 6: OR s, d <R $\beta$ aRC>

Пример: OR R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	0	0	1	0	1			

Автокод: OR6 s, d, MRC, rs, RC

OR6 s, d, MRCd, RC, rd

Формат 7: **OR**[.cc] s, d <#16/32àRC/R/R.L>

Пример: **OR** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	0	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: OR7 s, d, MIL, cc, #32, RD

OR7 s, d, MIC, cc, #16, RC

Формат 8а: <OP2> **OR** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **OR** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	1	0	0	0	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode					A	de	0	0	#	OP2					

Автокод: PROL <OP2> **OR** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **OR** s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **OR** R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs				0	0	0	0	1	0	0	0	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd					L	0	0	1	#	OP2				

Автокод: PROL <OP2> OR s1,s2,d <R/R.L $\bar{\mathbf{B}}$ R/R.L> <YRAM $\bar{\mathbf{a}}$ R0>

Формат 8с: <OP2.сс> OR s1,s2,d <R.L $\bar{\mathbf{B}}$ R.L>

Пример: LSRL.eq R6,R0,R8 OR R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1					d					s2					M	RS					0	0	0	0	0	1	0	0	0	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
S3/s3/#5					D2/d2					S4/s4					cc		RD					0	cc		1	0	#	OP2				

Автокод: PROL <OP2 сс> OR s1,s2,d <R.L $\bar{\mathbf{B}}$ R.L>

Формат 8d: <OP2> OR s1,s2,d <R $\bar{\mathbf{B}}$ RC>

Пример: LSRL R6,R0,R8 OR R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1					d					s2					M	Rs/Rd					0	0	0	0	1	0	0	0	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> OR s1,s2,d <R $\bar{\mathbf{B}}$ RC>

**ORC**                      *Логическое ИЛИ с инверсией (short)*                      **ORC**

**Операция:**

3-адресный вариант: short s1,s2,d;      s1 | ~s2 à d

2-адресный вариант: short s,d;      s | ~d à d

**Описание:** В формате short выполняется побитное логическое сложение операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) с инверсией второго операнда (s2 - в 3-адресном варианте; d - в 2-адресном варианте). Результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:**    OP1  
**Форматы:**                                  1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ORC**[.cc] s1,s2,d

Примеры:    **ORC**.ne R1,R5,R12

**ORC**      R1,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc			0	0	0	1	1	0	0	0	1	1	0				

*Автокод:*      ORC cc, s1, s2, d

Формат 2: **ORC**[.cc] #16,s2,d

Пример:      **ORC**.ne 15,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s1	d	s2	0	0	cc	0	0	1	0	1	0	0	0	1	1	0
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: ORC2 cc, #16, s2, d

Формат 3: **ORC** #16, d

Пример: **ORC** 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	1	0	0	0	1	1	0						

Автокод: ORC3 #16, d

Формат 4: **ORC** s, d <XRAM $\beta$ àR.L>

Пример: **ORC** R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	0	0	0	0	1	1	0	

Автокод: ORC4 s, d, MLR, ea, A, R

ORC4 s, d, MLM, ea, A, R

Формат 5: **ORC** s, d <R/R.L $\beta$ àR/R.L>

Пример: **ORC** R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	0	0	1	1	0			

Автокод: ORC5 s, d, MR, rs, rd

ORC5 s, d, MRL, RS, RD

Формат 6: **ORC** s, d <R $\beta$ àRC>

Пример: **ORC** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	0	1	1	0				

Автокод: ORC6 s, d ,MRC,rs,RC

ORC6 s, d ,MRCd,RC,rd

Формат 7: **ORC**[.cc] s,d <#16/32àRC/R/R.L>

Пример: **ORC** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ORC7 s, d, MIL,cc,#32,RD

ORC7 s, d, MIC,cc,#16,RC

Формат 8a: <OP2> **ORC** s1,s2,d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 **ORC** R1,R2,R5 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	1	0	0	0	0	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **ORC** s1,s2,d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **ORC** s1,s2,d <R/R.LβàR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **ORC** R1,R2,R5 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs				0	0	0	0	1	0	0	0	1	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> ORC s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> ORC s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 ORC R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS				0	0	0	0	0	1	0	0	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> ORC s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> ORC s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 ORC R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	Rs/Rd				0	0	0	0	1	0	0	0	1	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ORC s1,s2,d <R $\beta$ àRC>



**ORCL                      Логическое ИЛИ с инверсией (long)                      ORCL**

**Операция:**

3-адресный вариант: long S1, S2, D;                      S1 | ~S2 à D

2-адресный вариант: long S, D;                      S | ~D à D

**Описание:** В формате long выполняется побитное логическое сложение операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) с инверсией второго операнда (S2 - в 3-адресном варианте; D - в 2-адресном варианте). Результат помещается в операнд-приемник D. В качестве первого операнда-источника (S1 - в 3-адресном; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:**    OP1  
**Форматы:**                                      1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ORCL[.cc]** S1, S2, D

Примеры:    **ORCL.ne** R2, R6, R12

**ORCL**        R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	0	1	1	0	1	0	1	1	0			

Автокод:                      ORCL cc, S1, S2, D

Формат 2: **ORCL[.cc]** #32, S2, D

Пример:        **ORCL.ne** 15, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	1	0	1	0	1	0	1	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: ORCL2 cc, #32, S2, D

Формат 4: **ORCL** S,D <XRAM $\beta$ àR.L>

Пример: **ORCL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	1	0	1	0	1	0	1	1	0	

Автокод: ORCL4 S, D, MLR,ea,A,R

ORCL4 S, D, MLM,ea,A,R

Формат 5: **ORCL** S,D <R/R.L $\beta$ àR/R.L>

Пример: **ORCL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	0	1	0	1	1	0		

Автокод: ORCL5 S, D, MR,rs,rd

ORCL5 S, D, MRL,RS,RD

Формат 6: **ORCL** S,D <R $\beta$ àRC>

Пример: **ORCL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	0	1	0	1	1	0		

Автокод: ORCL6 S, D, MRC,rs,RC

ORCL6 S, D, MRCd,RC,rd

Формат 7: **ORCL**[.cc] S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **ORCL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	0	1	0	1	1	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ORCL7 S, D, MIL, cc, #32, RD

ORCL7 S, D, MIC, cc, #16, RC

Формат 8а: <OP2>**ORCL** S1, S2, D <XRAM $\beta$ àR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **ORCL** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	1	0	1	0	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> ORCL S1, S2, D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **ORCL** S1, S2, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеч.: LSRL R6, R0, R8 **ORCL** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	1	0	1	0	1	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> ORCL S1, S2, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **ORCL** S1, S2, D <R.L $\beta$ àR.L>

Пример: LSRL.eq R6, R0, R8 **ORCL** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	0	1	0	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0	cc		1	0	#	OP2							

Автокод: PROL <OP2 cc> ORCL S1,S2,D <R.L $\beta$ àR.L>

Формат 8d: <OP2>ORCL S1,S2,D <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 ORCL R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	1	0	1	0	1	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ORCL S1,S2,D <R $\beta$ àRC>

**ORI**
**Инверсия логического ИЛИ (short)**
**ORI**
**Операция:**

3-адресный вариант: short s1,s2,d;    ~ (s1 | s2) à d

2-адресный вариант: short s,d;    ~ (s | d) à d

**Описание:** В формате short выполняется побитное логическое сложение операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); инверсия результата помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ORI[.cc]** s1,s2,d

Примеры: **ORI.ne** R1,R5,R12

**ORI** R1,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	0	1	1	0	0	0	1	1	1			

*Автокод:* ORI cc, s1, s2, d

Формат 2: **ORI[.cc]** #16,s2,d

Пример: **ORI.ne** 15,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	1	0	1	0	0	0	0	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: ORI2 cc, #16, s2, d

Формат 3: ORI #16, d

Пример: ORI 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	1	0	0	0	1	1	1						

Автокод: ORI3 #16, d

Формат 4: ORI s, d <XRAM $\beta$ àR.L>

Пример: ORI R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode	u	0	1	de	0	1	0	0	0	0	1	1	1		

Автокод: ORI4 s, d, MLR, ea, A, R

ORI4 s, d, MLM, ea, A, R

Формат 5: ORI s, d <R/R.L $\beta$ àR/R.L>

Пример: ORI R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	0	1	1	1				

Автокод: ORI5 s, d, MR, rs, rd

ORI5 s, d, MRL, RS, RD

Формат 6: ORI s, d <R $\beta$ àRC>

Пример: ORI R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	0	0	1	1	1			

Автокод: ORI6 s, d, MRC, rs, RC

ORI6 s, d, MRCd, RC, rd

Формат 7: **ORI**[.cc] s, d <#16/32~~a~~RC/R/R.L>

Пример: **ORI** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	0	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ORI7 s, d, MIL, cc, #32, RD

ORI7 s, d, MIC, cc, #16, RC

Формат 8a: <OP2> **ORI** s1, s2, d <XRAM~~B~~aR.L> <YRAM~~a~~R0>

Пример: LSRL R6, R0, R8 **ORI** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	R				u	0	0	0	0	0	1	0	0	0	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **ORI** s1, s2, d <XRAM~~B~~aR.L> <YRAM~~a~~R0>

Формат 8b: <OP2> **ORI** s1, s2, d <R/R.L~~B~~aR/R.L> <YRAM~~a~~R0>

Пример: LSRL R6, R0, R8 **ORI** R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	RS/Rs				0	0	0	0	1	0	0	0	1	1	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT		RD/Rd					L	0	0	1	#	OP2				

Автокод: PROL <OP2> ORI s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> ORI s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 ORI R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s1					d					s2					0	RS					0	0	0	0	0	1	0	0	0	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc		RD					0	cc	1	0	#	OP2				

Автокод: PROL <OP2 сс> ORI s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> ORI s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 ORI R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1					d					s2					0	Rs/Rd					0	0	0	0	1	0	0	0	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					0	0	RC					sc	de	1	1	#	OP2				

Автокод: PROL <OP2> ORI s1,s2,d <R $\beta$ àRC>



**ORL**
**Логическое ИЛИ (long)**
**ORL**
**Операция:**

3-адресный вариант: long S1, S2, D;      S1 | S2 **à** D

2-адресный вариант: long S, D;      S | D **à** D

**Описание:** В формате long выполняется побитное логическое сложение операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); результат помещается в операнд-приемник D. В качестве первого операнда-источника (S1 - в 3-адресном варианте; S - в 2-адресном варианте) может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ORL[.cc]** S1, S2, D

Примеры: **ORL.ne** R2, R6, R12

**ORL** R2, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	1	0	1	0	1	0	1				

*Автокод:* ORL cc, S1, S2, D

Формат 2: **ORL[.cc]** #32, S2, D

Пример: **ORL.ne** 15, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1	D	S2	0	0	cc	0	0	1	0	1	0	1	0	1	0	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: ORL2 cc, #32, S2, D

Формат 4: **ORL** S,D <XRAM $\beta$ R.L>

Пример: **ORL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	1	0	1	0	1	0	1	0	1	

Автокод: ORL4 S, D, MLR,ea,A,R

ORL4 S, D, MLM,ea,A,R

Формат 5: **ORL** S,D <R/R.L $\beta$ R/R.L>

Пример: **ORL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	0	1	0	1	0	1		

Автокод: ORL5 S, D, MR,rs,rd

ORL5 S, D, MRL,RS,RD

Формат 6: **ORL** S,D <R $\beta$ RC>

Пример: **ORL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	0	1	0	1	0	1		

Автокод: ORL6 S, D, MRC,rs,RC

ORL6 S, D, MRCd,RC,rd

Формат 7: **ORL**[.cc] S,D <#16/32 $\beta$ RC/R/R.L>

Пример: **ORL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	0	1	0	1	0	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ORL7 S, D, MIL, cc, #32, RD

ORL7 S, D, MIC, cc, #16, RC

Формат 8а: <OP2> **ORL** S1, S2, D <XRAM~~β~~àR.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **ORL** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	1	0	1	0	1	0	1	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **ORL** S1, S2, D <XRAM~~β~~àR.L> <YRAMàR0>

Формат 8b: <OP2> **ORL** S1, S2, D <R/R.L~~β~~àR/R.L> <YRAMàR0>

Пример: LSRL R6, R0, R8 **ORL** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	1	0	1	0	1	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> **ORL** S1, S2, D <R/R.L~~β~~àR/R.L> <YRAMàR0>

Формат 8с: <OP2.cc> **ORL** S1, S2, D <R.L~~β~~àR.L>

Пример: LSRL.eq R6, R0, R8 **ORL** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	1	0	1	0	1	0	1	0	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> ORL S1, S2, D <R.L $\beta$ aR.L>

Формат 8d: <OP2> ORL S1, S2, D <R $\beta$ aRC>

Пример: LSRL R6, R0, R8 ORL R2, R4, R6 CCR, R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	1	0	1	0	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ORL S1, S2, D <R $\beta$ aRC>

**PACK**
*Упаковка (short)*
**PACK**
**Операция:**
3-адресный вариант: short s1,s2,d; {s2[15:8],s1[15:8]} → d

2-адресный вариант: short s,d; {d[15:8],s[15:8]} → d

**Описание:** В формате short выполняется пересылка старших байт операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте) в операнд-приемник d. Старший байт первого операнда-источника s1 (s) пересылается в младший байт приемника d, старший байт второго операнда-источника s2 (d) - в старший байт приемника d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**
Формат 1: **PACK[.cc]** s1,s2,d

 Примеры: **PACK.ne** R1,R5,R12

**PACK** R1,R5,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc			0	0	0	1	1	0	0	1	1	0	0				

 Автокод: **PACK cc, s1, s2, d**
Формат 4: **PACK** s,d <XRAMB → R.L>

 Пример: **PACK** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode			u	0	1	de	0	1	0	0	1	1	0	0	

 Автокод: **PACK4 s, d, MLR,ea,A,R**

PACK4 s, d, MLM,ea,A,R

Формат 5: **PACK** s,d <R/R.L**à**R/R.L>

Пример: **PACK** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	1	1	0	0				

Автокод: **PACK5** s, d, MR,rs,rd

**PACK5** s, d, MRL,RS,RD

Формат 6: **PACK** s,d <R**à**RC>

Пример: **PACK** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	1	1	0	0				

Автокод: **PACK6** s, d ,MRC,rs,RC

**PACK6** s, d ,MRCd,RC,rd

Формат 7: **PACK**[.cc] s,d <#16/32**à**RC/R/R.L>

Пример: **PACK** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **PACK7** s, d, MIL,cc,#32,RD

**PACK7** s, d, MIC,cc,#16,RC

Формат 8a: <OP2> **PACK** s1,s2,d <XRAM**à**R.L> <YRAM**à**R0>

Пример: LSRL R6,R0,R8 **PACK** R1,R2,R5 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	R				u	0	0	0	0	0	1	0	0	1	1	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> PACK s1,s2,d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **PACK** s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R6,R0,R8 **PACK** R1,R2,R5 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	RS/Rs				0	0	0	0	1	0	0	1	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> PACK s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **PACK** s1,s2,d <R.L $\beta$ àR.L>

Пример: LSRL.eq R6,R0,R8 **PACK** R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	RS				0	0	0	0	0	1	0	0	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> PACK s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> **PACK** s1,s2,d <R $\beta$ àRC>

Пример: LSRL R6,R0,R8 **PACK** R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	Rs/Rd				0	0	0	0	1	0	0	1	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> ПАКК s1,s2,d <R**B**àRC>



**PACKL**
*Упаковка (long)*
**PACKL**
**Операция:**
*3-адресный вариант:* long S1, S2, D; {S2[31:16], S1[31:16]} → D

*2-адресный вариант:* long S, D; {D[31:16], S[31:16]} → D

**Описание:** В формате long выполняется пересылка старших 16 разрядов операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте) в операнд-приемник D. Старшие 16 разрядов первого операнда-источника S1 (S) пересылается в младшие 16 разрядов приемника D, старшие 16 разрядов второго операнда-источника S2 (D) - в старшие 16 разрядов приемника D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** **PACKL**[.cc] S1, S2, D

**Примеры:** **PACKL**.ne R2, R6, R12

**PACKL** R2, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	1	0	1	1	1	0	0				

*Автокод:* **PACKL** cc, S1, S2, D

**Формат 4:** **PACKL** S, D <XRAM $\beta$ àR.L>

**Пример:** **PACKL** R2, R4 R8, (A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	1	0	1	1	1	0	0			

*Автокод:* **PACKL4** S, D, MLR, ea, A, R

PACKL4 S, D, MLM,ea,A,R

Формат 5: **PACKL** S,D <R/R.L**В**àR/R.L>

Пример: **PACKL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	1	1	0	0				

Автокод: **PACKL5** S, D, MR,rs,rd

**PACKL5** S, D, MRL,RS,RD

Формат 6: **PACKL** S,D <R**В**àRC>

Пример: **PACKL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	1	1	0	0				

Автокод: **PACKL6** S, D, MRC,rs,RC

**PACKL6** S, D, MRCd,RC,rd

Формат 7: **PACKL**[.cc] S,D <#16/32**В**àRC/R/R.L>

Пример: **PACKL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **PACKL7** S, D, MIL,cc,#32,RD

**PACKL7** S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **PACKL** S1,S2,D <XRAM**В**àR.L> <YRAM**В**àR0>

Пример: LSRL R6,R0,R8 **PACKL** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	1	0	1	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> PACKL S1,S2,D <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> **PACKL** S1,S2,D <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Пример: LSRL R6,R0,R8 **PACKL** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	1	0	1	1	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> PACKL S1,S2,D <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> **PACKL** S1,S2,D <R.L $\beta$ R.L>

Пример: LSRL.eq R6,R0,R8 **PACKL** R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	1	0	1	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> PACKL S1,S2,D <R.L $\beta$ R.L>

Формат 8d: <OP2> **PACKL** S1,S2,D <R $\beta$ RC>

Пример: LSRL R6,R0,R8 **PACKL** R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	1	0	1	1	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

АВТОКОД: PROL <OP2> PACKL S1, S2, D <R<sup>B</sup>RC>

### *PDN Определение параметра денормализации (short) PDN*

**Операция:** short  $s, d;$  pdn( $s$ )  $\rightarrow$   $d$

**Описание:** В формате short определяется параметр денормализации операнда-источника  $s$  (количество разрядов слева до старшей значащей цифры без учета разряда знака), результат помещается в операнд-приемник  $d$ .

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: PDN[.cc]  $s, d$

Пример: PDN.ne R1, R5

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	0	cc				0	0	0	1	1	0	0	1	1	1	1		

Автокод: PDN cc, s, d

Формат 4: PDN  $s, d$  <XRAM $\beta$  $\rightarrow$ R.L>

Пример: PDN R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode				u	0	1	de	0	1	0	0	1	1	1	1

Автокод: PDN4 s, d, MLR, ea, A, R

PDN4 s, d, MLM, ea, A, R

Формат 5: PDN  $s, d$  <R/R.L $\beta$  $\rightarrow$ R/R.L>

Пример: PDN R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	1	1	1	1	1	1	1	1	1

Автокод: PDN5 s, d, MR,rs,rd  
 PDN5 s, d, MRL,RS,RD

Формат 6: **PDN** s,d <RβàRC>

Пример: **PDN** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	1	1	1	1	1	1	1	1

Автокод: PDN6 s, d ,MRC,rs,RC  
 PDN6 s, d ,MRCd,RC,rd

Формат 7: **PDN[.cc]** s,d #16/32àRC/R/R.L>

Пример: **PDN** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	1	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: PDN7 s, d, MIL,cc,#32,RD  
 PDN7 s, d, MIC,cc,#16,RC

Формат 8а: <OP2> **PDN** s,d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **PDN** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				0	0	0	0	0	0	R				u	0	0	0	0	1	0	0	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode		A		de		0		0		#		OP2					

Автокод: PROL <OP2> PDN s,d <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> PDN s,d <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Примеры: LSRL R5,R0,R8 PDN R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
s				d				0				0				0				0				RS/Rs				0		0		0		1		0		0		1		1		1		1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd				L		0		0		1		#		OP2			

Автокод: PROL <OP2> PDN s,d <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> PDN s,d <R.L $\beta$ R.L>

Примеры: LSRL.eq R5,R0,R8 PDN R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
s				d				0				0				0				0				RS				0		0		0		0		1		0		0		1		1		1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0		cc		1		0		#		OP2			

Автокод: PROL <OP2 cc> PDN s,d <R.L $\beta$ R.L>

Формат 8d: <OP2> PDN s,d <R $\beta$ RC>

Примеры: LSRL R5,R0,R8 PDN R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0														
s				d				0				0				0				0				Rs/Rd				0		0		0		1		0		0		1		1		1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32		
S3/s3/#5				D2/d2				S4/s4				0		0		RC				sc		de		1		1		#		OP2			

Автокод: PROL <OP2> PDN s,d <R**B**RC>



**PDNE** *Определение параметра денормализации 16-разрядной мантиссы*
**PDNE**
**Операция:** short s; long D;

 $\{32\{(|s[15:0])\}\}\&\{pdn(s),D[15:0]\} \mathring{a} D$ 

**Описание:** Измеряется параметр денормализации входного операнда (16-разрядной мантиссы) s. Результат измерения параметра денормализации записывается в старшее полуслово приемника D[31:16]. Младшее полуслово приемника D[15:0] не изменяется. Если же входная мантисса равна нулю, то сбрасываются оба полуслова приемника, т.е. D[31:0]=0.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	√	-	-

**Синтаксис ассемблера и код инструкции:**
Формат 1: **PDNE[.cc]** s,D

 Пример: **PDNE.ne** R1,R2

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				D								0	0	0	0	0	0	cc				0	0	0	1	1	0	0	0	1	0	0

Автокод: PDNE cc, s, D

Формат 4: **PDNE** s,D <XRAM $\mathring{a}$ R.L>

 Пример: **PDNE** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
s				D								R				A				mode	u	0	1	de	0	1	0	0	0	0	1	0	0

Автокод: PDNE4 s, D, MLR,ea,A,R

PDNE4 s, D, MLM,ea,A,R

Формат 5: **PDNE** s,D <R/R.L**à**R/R.L>

Пример: **PDNE** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	0	1	0	0				

Автокод: PDNE5 s, D, MR,rs,rd

PDNE5 s, D, MRL,RS,RD

Формат 6: **PDNE** s,D <R**à**RC>

Пример: **PDNE** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				R				sc	RC				1	0	de	0	1	0	0	0	1	0	0				

Автокод: PDNE6 s, D ,MRC,rs,RC

PDNE6 s, D ,MRCd,RC,rd

Формат 7: **PDNE[.cc]** s,D #16/32**à**RC/R/R.L>

Пример: **PDNE** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: PDNE7 s, D, MIL,cc,#32,RD

PDNE7 s, D, MIC,cc,#16,RC

Формат 8а: <OP2> **PDNE** s,D <XRAM**à**R.L> <YRAM**à**R0>

Примеры: LSRL R5,R0,R8 **PDNE** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s	D	0	0	0	0	0	0	R	u	0	0	0	0	1	0	0	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT		mode			A		de		0		0		#		OP2			

Автокод: PROL <OP2> PDNE s,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> PDNE s,D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 PDNE R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
s				D				0				0				0				0				RS/Rs				0		0		0		1		0		0		0		1		0		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT		RD/Rd			L		0		0		1		#		OP2			

Автокод: PROL <OP2> PDNE s,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> PDNE s,D <R.LβàR.L>

Примеры: LSRL.eq R5,R0,R8 PDNE R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																		
s				D				0				0				0				0				RS				0		0		0		0		1		0		0		0		1		0		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				cc		RD			0		cc		1		0		#		OP2			

Автокод: PROL <OP2 cc> PDNE s,D <R.LβàR.L>

Формат 8d: <OP2> PDNE s,D <RβàRC>

Примеры: LSRL R5,R0,R8 PDNE R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
s				D				0				0				0				0				Rs/Rd				0		0		0		1		0		0		0		1		0		0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> PDNE s, D <R**B**àRC>

### PDNL Определение параметра денормализации (long) PDNL

**Операция:** long S; short d; pdn(S) → d

**Описание:** В формате long определяется параметр денормализации операнда-источника S (количество разрядов слева до старшей значащей цифры без учета разряда знака), результат помещается в операнд-приемник d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: PDNL[.cc] S,d

*Пример:* PDNL.ne R2,R5

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				d				0	0	0	0	0	0	0	0	cc				0	0	0	1	1	0	1	1	1	1	1	1

*Автокод:* PDN cc, S, d

Формат 4: PDNL S,d <XRAMB→R.L>

*Пример:* PDNL R2,R1 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode				u	0	1	de	0	1	0	1	1	1	1	1

*Автокод:* PDNL4 s, d, MLR,ea,A,R

PDNL4 s, d, MLM,ea,A,R

Формат 5: PDNL S,d <R/R.LB→R/R.L>

*Пример:* PDNL R2,R1 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	1	1	1	1	1	1	1	1	1

Автокод: PDNL5 S, d, MR,rs,rd

PDNL5 S, d, MRL,RS,RD

Формат 6: PDNL S,d <RβàRC>

Пример: PDNL R0,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				d				R				sc	RC				1	0	de	0	1	0	1	1	1	1	1	1	1	1	1

Автокод: PDNL6 S, d ,MRC,rs,RC

PDNL6 S, d ,MRCd,RC,rd

Формат 7: PDNL[.cc] S,d #16/32àRC/R/R.L>

Пример: PDNL R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	1	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: PDNL7 S, d, MIL,cc,#32,RD

PDNL7 S, d, MIC,cc,#16,RC

Формат 8а: <OP2> PDNL S,d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 PDNL R0,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	d	0	0	0	0	0	0	R	u	0	0	0	0	1	0	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode			A	de	0	0	#	OP2							

Автокод: PROL <OP2> PDNL S,d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> PDNL S,d <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 PDNL R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				d				0	0	0	0	0	0	RS/Rs				0	0	0	0	1	0	1	1	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	RD/Rd			L	0	0	1	#	OP2							

Автокод: PROL <OP2> PDNL s,d <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> PDNL S,d <R.LβàR.L>

Примеры: LSRL.eq R5,R0,R8 PDNL R0,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				d				0	0	0	0	0	0	RS				0	0	0	0	0	1	0	1	1	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc	RD			0	cc	1	0	#	OP2							

Автокод: PROL <OP2 cc> PDNL S,d <R.LβàR.L>

Формат 8d: <OP2> PDNL S,d <RβàRC>

Примеры: LSRL R5,R0,R8 PDNL R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				d				0	0	0	0	0	0	Rs/Rd				0	0	0	0	1	0	1	1	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> PDNL S,d <RβàRC>



**PDNLE** *Определение параметра денормализации  
разрядной мантиссы* **PDNLE**

32-

**Операция:** long S, D; {32{(|S[31:0])}} & {pdn(S), D[15:0]} → D

**Описание:** Измеряется параметр денормализации входного операнда (32-разрядной мантиссы) S. Результат измерения параметра денормализации записывается в старшее полуслово приемника D[31:16]. Младшее полуслово приемника D[15:0] не изменяется. Если же входная мантисса равна нулю, то сбрасываются оба полусллова приемника, т.е. D[31:0]=0.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **PDNLE**[.cc] S, D

*Пример:* **PDNLE**.ne R0, R2

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	0	cc				0	0	0	1	1	0	1	1	1	1	0	

*Автокод:* PDNLE cc, S, D

Формат 4: **PDNLE** S, D <XRAMβàR.L>

*Пример:* **PDNLE** R0, R2 R8, (A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	1	0	1	1	1	1	0		

*Автокод:* PDNLE4 S, D, MLR, ea, A, R

PDNLE4 S, D, MLM, ea, A, R

Формат 5: **PDNLE** S,D <R/R.L $\beta$ àR/R.L>

Пример: **PDNLE** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	1	1	1	0				

Автокод: PDNLE5 s, D, MR,rs,rd

PDNLE5 s, D, MRL,RS,RD

Формат 6: **PDNLE** s,D <R $\beta$ àRC>

Пример: **PDNLE** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	1	1	1	0				

Автокод: PDNLE6 S, D ,MRC,rs,RC

PDNLE6 S, D ,MRCd,RC,rd

Формат 7: **PDNLE[.cc]** S,D #16/32 $\beta$ àRC/R/R.L>

Пример: **PDNLE** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	1	1	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: PDNLE7 S, D, MIL,cc,#32,RD

PDNLE7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **PDNLE** S,D <XRAM $\beta$ àR.L> <YRAM $\beta$ àR0>

Примеры: LSRL R5,R0,R8 **PDNLE** R0,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	1	0	1	1	1	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de	0	0	#	OP2								

Автокод: PROL <OP2> PДNLE S,D <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> PДNLE S,D <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Примеры: LSRL R5,R0,R8 PДNLE R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
S				D				0				0				0				0				RS/Rs				0	0	0	0	1	0	1	1	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> PДNLE S,D <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> PДNLE S,D <R.L $\beta$ R.L>

Примеры: LSRL.eq R5,R0,R8 PДNLE R0,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
S				D				0				0				0				0				RS				0	0	0	0	0	1	0	1	1	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2								

Автокод: PROL <OP2 cc> PДNLE S,D <R.L $\beta$ R.L>

Формат 8d: <OP2> PДNLE S,D <R $\beta$ RC>

Примеры: LSRL R5,R0,R8 PДNLE R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
S				D				0				0				0				0				Rs/Rd				0	0	0	0	1	0	1	1	1	1	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0		0		RC			sc	de	1	1	#	OP2							

Автокод: PROL <OP2> PDNLE S,D <R**B**ARC>

### PDNX Определение параметра денормализации (X16) PDNX

**Операция:** (X16) S; short d;

$\min(\text{pdn}(S[31:16]), \text{pdn}(S[15:0])) \rightarrow d$

**Описание:** Определяется параметр денормализации операнда-источника S, рассматриваемого как комплексное число в формате X16. Для этого определяется количество разрядов слева до старшей значащей цифры (без учета разряда знака) в действительной S[31:16] и мнимой S[15:0] части числа, наименьшее из полученных чисел помещается в операнд-приемник d.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
-	-	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: PDNX[.cc] S,d

*Пример:* PDNX.ne R2,R5

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				d				0	0	0	0	0	0	0	cc				0	0	0	1	1	0	1	1	1	0	1		

*Автокод:* PDN cc, S, d

Формат 4: PDNX S,d <XRAMB $\rightarrow$ R.L>

*Пример:* PDNX R2,R1 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	0	1	1	1	0	1		

*Автокод:* PDNX4 s, d, MLR, ea, A, R

PDNX4 s, d, MLM,ea,A,R

Формат 5: **PDNX** S,d <R/R.L $\beta$ àR/R.L>

Пример: **PDNX** R2,R1 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	1	1	0	1				

Автокод: PDNX5 S, d, MR,rs,rd

PDNX5 S, d, MRL,RS,RD

Формат 6: **PDNX** S,d <R $\beta$ àRC>

Пример: **PDNX** R0,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				d				R				sc	RC				1	0	de	0	1	0	1	1	1	0	1				

Автокод: PDNX6 S, d ,MRC,rs,RC

PDNX6 S, d ,MRCd,RC,rd

Формат 7: **PDNX[.cc]** S,d #16/32àRC/R/R.L>

Пример: **PDNX** R2,R0 0x12345678,R16.L

Код инструкции:

3	3	2	2	2	2	2	2	2	2	2	1	1	1	1	16	15	1	1	1	1	10	9	8	7	6	5	4	3	2	1	0
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7			4	3	2	1											
S				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: PDNX7 S, d, MIL,cc,#32,RD

PDNX7 S, d, MIC,cc,#16,RC

Формат 8а: <OP2> **PDNX** S,d <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **PDNX** R0,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	d	0	0	0	0	0	0	0	R	u	0	0	0	0	1	0	1	1	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode			A	de	0	0	#	OP2							

Автокод: PROL <OP2> PDNX S,d <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> PDNX S,d <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Примеры: LSRL R5,R0,R8 PDNX R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					d					0	0	0	0	0	0	RS/Rs					0	0	0	0	1	0	1	1	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	RD/Rd			L	0	0	1	#	OP2							

Автокод: PROL <OP2> PDNX s,d <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> PDNX S,d <R.L $\beta$ R.L>

Примеры: LSRL.eq R5,R0,R8 PDNX R0,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S					d					0	0	0	0	0	0	RS					0	0	0	0	0	1	0	1	1	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc	RD			0	cc	1	0	#	OP2							

Автокод: PROL <OP2 cc> PDNX S,d <R.L $\beta$ R.L>

Формат 8d: <OP2> PDNX S,d <R $\beta$ RC>

Примеры: LSRL R5,R0,R8 PDNX R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					d					0	0	0	0	0	0	Rs/Rd					0	0	0	0	1	0	1	1	1	0	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> PDNX S,d <R**B**àRC>



**RNDL Округление**
**RNDL**

**Операция:** long S,D;

$\{S[31:16], \{16\{0}\}\} + \text{round}(S[15:0]) \rightarrow D$

**Описание:** Округляется 32-х разрядное значение операнда источника и запоминается в регистре приемника. Режим округления устанавливается битом RND=CCR[9].

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **RNDL**[.cc] S,D

*Пример:* **RNDL**.ne R0,R2

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	0	1	0	1	0	1	0	0	0		

*Автокод:* RNDL cc, S, D

Формат 4: **RNDL** S,D <XRAM $\beta$ àR.L>

*Пример:* **RNDL** R0,R2 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	0	1	0	0	0			

*Автокод:* RNDL4 S, D, MLR,ea,A,R

RNDL4 S, D, MLM,ea,A,R

Формат 5: **RNDL** S,D <R/R.L $\beta$ àR/R.L>

Пример: **RNDL** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	0	1	0	0	0			

Автокод: RNDL5 S, D, MR,rs,rd  
RNDL5 S, D, MRL,RS,RD

Формат 6: **RNDL** S,D <RβàRC>

Пример: **RNDL** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	0	1	0	0	0				

Автокод: RNDL6 S, D ,MRC,rs,RC  
RNDL6 S, D ,MRCd,RC,rd

Формат 7: **RNDL[.cc]** S,D #16/32àRC/R/R.L>

Пример: **RNDL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: RNDL7 S, D, MIL,cc,#32,RD  
RNDL7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **RNDL** S,D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **RNDL** R0,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	0	0	1	0	1	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode		A		de	0	0	#	OP2									

Автокод: PROL <OP2> RNDL S,D <XRAM $\beta$ aR.L> <YRAMaR0>

Формат 8b: <OP2> RNDL S,D <R/R.L $\beta$ aR/R.L> <YRAMaR0>

Примеры: LSRL R5,R0,R8 RNDL R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS/Rs				0	0	0	0	0	1	0	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd		L	0	0	1	#	OP2										

Автокод: PROL <OP2> RNDL S,D <R/R.L $\beta$ aR/R.L> <YRAMaR0>

Формат 8c: <OP2.cc> RNDL S,D <R.L $\beta$ aR.L>

Примеры: LSRL.eq R5,R0,R8 RNDL R0,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	RS				0	0	0	0	0	0	1	0	1	0	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD		0	cc	1	0	#	OP2										

Автокод: PROL <OP2 cc> RNDL S,D <R.L $\beta$ aR.L>

Формат 8d: <OP2> RNDL S,D <R $\beta$ aRC>

Примеры: LSRL R5,R0,R8 RNDL R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	Rs/Rd				0	0	0	0	0	1	0	1	0	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

АВТОКОД: PROL      <OP2>      RNDL      S, D      <R**B**àRC>

**ROL**
*Сдвиг циклический влево (short)*
**ROL**
**Операция:** short  $s, d; \{s[14:0], C\} \text{ à } d$ 

**Описание:** Операнд источника  $s$  сдвигается влево на один разряд и запоминается в приемнике  $d$ . В младший освободившийся в результате сдвига разряд записывается признак  $C$ , установленный в регистре  $CCR$  на момент начала операции. В свою очередь, в бит  $C$  регистра  $CCR$  заносится вытолкнутый влево разряд  $s[15]$ .

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	* s[15]

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** **ROL**[.cc]  $s, d$ 
**Примеры:** **ROL**.ne R1, R12

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	0	0	cc				0	0	0	1	1	1	0	0	0	1	1	

**Автокод:** **ROL** cc, s, d

**Формат 4:** **ROL**  $s, d$  <XRAMB à R.L>

**Примеры:** **ROL** R1, R2 R8, (A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	1	0	0	0	1	1		

**Автокод:** **ROL4**  $s, d, MLR, ea, A, R$ 
**ROL4**  $s, d, MLM, ea, A, R$

Формат 5: **ROL** s,d <R/R.L**В**àR/R.L>

Примеры: **ROL** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	1	0	0	0	0	1	1			

Автокод: **ROL5** s, d, MR,rs,rd

**ROL5** s, d, MRL,RS,RD

Формат 6: **ROL** s,d <R**В**àRC>

Примеры: **ROL** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	1	0	0	0	1	1				

Автокод: **ROL6** s, d ,MRC,rs,RC

**ROL6** s, d ,MRCd,RC,rd

Формат 7: **ROL[.cc]** s,d <#16/32**В**àRC/R/R.L>

Пример: **ROL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	0	0	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ROL7** s, d, MIL,cc,#32,RD

**ROL7** s, d, MIC,cc,#16,RC

Формат 8а: **ROL** s3,d2 <OP1> <XRAM**В**àR.L> <YRAM**В**àR0>

Примеры: **ROL** R1,R5 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1/s1	D/d	S2/s2	M	R	u	0	0	0	0	OP1
-------	-----	-------	---	---	---	---	---	---	---	-----

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	AT	mode				A	de	0	0	0	0	0	0	0	1	1		

Автокод: PROL ROL s3,d2 <OP1> <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: ROL s3,d2 <OP1> <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Примеры: ROL R1,R3 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	0	0	0	0	0	1	1		

Автокод: PROL ROL s3,d2 <OP2> <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: ROL[.cc] s3,d2 <OP1> <R.L $\beta$ R.L>

Примеры: ROL.eq R1,R25 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	cc	RD				0	cc	1	0	0	0	0	0	0	1	1		

Автокод: PROL ROL cc s3,d2 <OP1> <R.L $\beta$ R.L>

Формат 8d: ROL s3,d2 <OP1> <R $\beta$ RC>

Примеры: ROL R1,R25 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	RC				sc	de	1	1	0	0	0	0	1	1				

Автокод: PROL      ROL s3,d2      <OP1>      <R**B**àRC>



**ROLL**
*Сдвиг циклический влево (long)*
**ROLL**
**Операция:**  $\text{long } S, D; a\{S[30:0], C\} \rightarrow D$ 

**Описание:** Операнд источника  $S$  сдвигается влево на один разряд и запоминается в приемнике  $D$ . В младший освободившиеся в результате сдвига разряд записывается признак  $C$ , установленный в регистре  $CCR$  на момент начала операции. В свою очередь, в бит  $C$  регистра  $CCR$  заносится вытолкнутый влево разряд  $S[31]$ .

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	* S[31]

**Синтаксис ассемблера и код инструкции:**
**Формат 1:** **ROLL**[.cc]  $S, D$ 
**Примеры:** **ROLL**.ne  $R0, R12$ 
**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	cc				0	0	0	1	1	1	0	1	0	0	1		

**Автокод:** **ROLL** cc,  $S1, D$ 
**Формат 4:** **ROLL**  $S, D$  <XRAM $\beta$ A $\alpha$ R.L>

**Примеры:** **ROLL**  $R0, R2$   $R8, (A0)+$ 
**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	1	1	0	1	0	0	1			

**Автокод:** **ROLL4**  $S, D, MLR, ea, A, R$ 
**ROLL4**  $S, D, MLM, ea, A, R$

Формат 5: **ROLL** S,D <R/R.L $\bar{a}$ R/R.L>

Примеры: **ROLL** R8,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	1	0	1	0	0	1				

Автокод: ROLL5 S, D, MR,rs,rd

ROLL5 S, D, MRL,RS,RD

Формат 6: **ROLL** S,D <R $\bar{a}$ RC>

Примеры: **ROLL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	1	0	1	0	1	0	1			

Автокод: ROLL6 S, D ,MRC,rs,RC

ROLL6 S, D ,MRCd,RC,rd

Формат 7: **ROLL[.cc]** S,D <#16/32 $\bar{a}$ RC/R/R.L>

Пример: **ROLL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	0	1	0	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: ROLL7 S, D, MIL,cc,#32,RD

ROLL7 S, D, MIC,cc,#16,RC

Формат 8а: **ROLL** S3,D2<OP1> <XRAM $\bar{a}$ R.L> <YRAM $\bar{a}$ R0>

Примеры: **ROLL** R8,R12 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1/s1	D/d	S2/s2	M	R	u	0	0	0	0	OP1
-------	-----	-------	---	---	---	---	---	---	---	-----

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				0	0	0	0	0	0	AT	mode				A	de	0	0	0	0	1	0	0	1			

Автокод: PROL ROLL S3,D2 <OP1> <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>

Формат 8b: ROLL S3,D2 <OP1> <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Примеры: ROLL R0,R4 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	0	0	1	0	0	1			

Автокод: PROL ROLL S3,D2 <OP2> <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>

Формат 8c: ROLL[.cc] S3,D2 <OP1> <R.L $\beta$ R.L>

Примеры: ROLL.eq R0,R2 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	cc	RD				0	cc	1	0	0	0	1	0	0	1			

Автокод: PROL ROLL cc S3,D2 <OP1> <R.L $\beta$ R.L>

Формат 8d: ROLL S3,D2<OP1> <R $\beta$ RC>

Примеры: ROLL R0,R24 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				0	0	0	0	0	0	RC				sc	de	1	1	0	0	1	0	0	1				

Автокод: PROL      ROLL S3,D2      <OP1>      <R**B**àRC>

**ROR**
**Сдвиг циклический вправо (short)**
**ROR**

**Операция:** short s,d; { C,s[15:1]} à d

**Описание:** Операнд источника s сдвигается вправо на один разряд и запоминается в приемнике d. В старший освободившийся в результате сдвига разряд записывается признак C, установленный в регистре CCR на момент начала операции. В свою очередь, в бит C регистра CCR заносится вытолкнутый вправо разряд s[0].

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	* s[0]

**Синтаксис ассемблера и код инструкции:**

Формат 1: **ROR[.cc]** s,d

Примеры: **ROR.ne** R1,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	0	0	cc				0	0	0	1	1	1	1	0	0	1	1	

Автокод: ROR cc, s1, d

Формат 4: **ROR** s,d <XRAMBàR.L>

Примеры: **ROR** R1,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	1	1	0	0	1	1		

Автокод: ROR4 s, d, MLR,ea,A,R

ROR4 s, d, MLM,ea,A,R

Формат 5: **ROR** s,d <R/R.L $\beta$ àR/R.L>

Примеры: **ROR** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	1	1	0	0	1	1				

Автокод: **ROR5** s, d, MR,rs,rd

**ROR5** s, d, MRL,RS,RD

Формат 6: **ROR** s,d <R $\beta$ àRC>

Примеры: **ROR** R1,R5 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	1	1	0	0	1	1				

Автокод: **ROR6** s, d ,MRC,rs,RC

**ROR6** s, d ,MRCd,RC,rd

Формат 7: **ROR[.cc]** s,d <#16/32 $\beta$ àRC/R/R.L>

Пример: **ROR** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **ROR7** s, d, MIL,cc,#32,RD

**ROR7** s, d, MIC,cc,#16,RC

Формат 8а: **ROR** s3,d2 <OP1> <XRAM $\beta$ àR.L> <YRAM $\beta$ àR0>

Примеры: **ROR** R1,R5 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1/s1	D/d	S2/s2	M	R	u	0	0	0	0	OP1
-------	-----	-------	---	---	---	---	---	---	---	-----

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	AT	mode				A	de	0	0	0	1	0	0	1	1			

Автокод: PROL ROR s3,d2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: ROR s3,d2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: ROR R1,R3 ANDL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	0	1	0	0	1	1			

Автокод: PROL ROR s3,d2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: ROR[.cc] s3,d2 <OP1> <R.L $\beta$ àR.L>

Примеры: ROR.eq R1,R25 ANDL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	cc	RD				0	cc	1	0	0	1	0	0	1	0	0	1	1

Автокод: PROL ROR cc s3,d2 <OP1> <R.L $\beta$ àR.L>

Формат 8d: ROR s3,d2 <OP1> <R $\beta$ àRC>

Примеры: ROR R1,R25 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	RC				sc	de	1	1	0	1	0	0	1	1				

Автокод: PROL

ROR S3,D2

&lt;OP1&gt;

 <RВàRC>



**RORL**
*Сдвиг циклический вправо (long)*
**RORL**

**Операция:** long S,D; {C,S[31:1]} à D

**Описание:** Операнд источника S сдвигается вправо на один разряд и запоминается в приемнике D. В старший освободившийся в результате сдвига разряд записывается признак C, установленный в регистре CCR на момент начала операции. В свою очередь, в бит C регистра CCR заносится вытолкнутый вправо разряд S[0].

**Тип:** OP2

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	* S[0]

**Синтаксис ассемблера и код инструкции:**

Формат 1: **RORL[.cc]** S,D

*Примеры:* **RORL.ne** R0,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	0	cc				0	0	0	1	1	1	1	1	1	0	0	1

*Автокод:* RORL cc, S1, D

Формат 4: **RORL** S,D <XRAMβàR.L>

*Примеры:* **RORL** R0,R2 R8,(A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	1	1	1	1	1	0	0	1		

*Автокод:* RORL4 S, D, MLR,ea,A,R

RORL4 S, D, MLM,ea,A,R

Формат 5: **RORL** S,D <R/R.L**à**R/R.L>

Примеры: **RORL** R8,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	1	1	1	0	0	1				

Автокод: RORL5 S, D, MR,rs,rd

RORL5 S, D, MRL,RS,RD

Формат 6: **RORL** S,D <R**à**RC>

Примеры: **RORL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	1	1	1	0	0	1				

Автокод: RORL6 S, D ,MRC,rs,RC

RORL6 S, D ,MRCd,RC,rd

Формат 7: **RORL[.cc]** S,D <#16/32**à**RC/R/R.L>

Пример: **RORL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	1	0	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: RORL7 S, D, MIL,cc,#32,RD

RORL7 S, D, MIC,cc,#16,RC

Формат 8а: **RORL** S3,D2<OP1> <XRAM**à**R.L> <YRAM**à**R0>

Примеры: **RORL** R8,R12 ANDL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1/s1	D/d	S2/s2	M	R	u	0	0	0	0	OP1
-------	-----	-------	---	---	---	---	---	---	---	-----

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				0	0	0	0	0	0	AT	mode				A	de	0	0	0	1	1	0	0	1			

Автокод: **PROL RORL S3,D2 <OP1> <XRAM $\beta$ R.L> <YRAM $\alpha$ R0>**

Формат 8b: **RORL S3,D2 <OP1> <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>**

Примеры: **RORL R0,R4 ANDL R6,R0,R8 R8,R15 (AT),R0**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	0	1	1	0	0	1			

Автокод: **PROL RORL S3,D2 <OP2> <R/R.L $\beta$ R/R.L> <YRAM $\alpha$ R0>**

Формат 8c: **RORL[.cc] S3,D2 <OP1> <R.L $\beta$ R.L>**

Примеры: **RORL.eq R1,R25 ANDL R6,R0,R8 R8.L,R16.L**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				0	0	0	0	0	0	cc	RD				0	cc	1	0	0	1	1	0	0	1			

Автокод: **PROL RORL cc S3,D2 <OP1> <R.L $\beta$ R.L>**

Формат 8d: **RORL S3,D2<OP1> <R $\beta$ RC>**

Примеры: **RORL R0,R24 ANDL R6,R0,R8 CCR,R17**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				0	0	0	0	0	0	RC				sc	de	1	1	0	1	1	0	0	1				

Автокод: PROL RORL S3,D2 <OP1> <R**B**àRC>

**RTS**
**Возврат из подпрограммы**
**RTS**
**Операция:**  $SS \rightarrow PC; \quad SP[3:0] - 1 \rightarrow SP[3:0]$ 

**Описание:** Выполняется возврат из подпрограммы. Адрес следующей команды считывается из системного стека *SS* и записывается в программный счетчик *PC*, указатель системного стека *SP[3:0]* уменьшается на единицу.

**Тип:** OP1

**Форматы:** 3m

**Признаки результата:**

U	N	Z	V	C
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**
Формат 3m: **RTS**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0

Автокод: RTS

**SAC2** *Парная операция накопления со знаком SAC2*

**Операция:** long S1, S2, D;

AC0  $\hat{a}$  D[31:0];

AC0 +  $(-1)^{S2[0]} * S1[15:0]$   $\hat{a}$  AC0;

AC1 +  $(-1)^{S2[16]} * S1[31:16]$   $\hat{a}$  AC1;

{S2[0], S2[31:1]}  $\hat{a}$  S2[31:0];

**Описание:** Парная операция накопления со знаком, выполняется в целом знаковом формате. Целые 16-разрядные числа S1[31:16] и S1[15:0], упакованные в 32-разрядном операнде S1, прибавляются (или вычитаются) к 32-разрядным аккумуляторам AC1 и AC0, соответственно. Знак накопления (сложение или вычитание) определяется, соответственно, битами 16 и 0 входного 32-разрядного операнда S2.

После накопления выполняется также круговой правый сдвиг 32-разрядного операнда S2 на 1 бит. Возможен режим насыщения – Saturation (бит CCR[8] управляющего регистра равен 1), тогда при переполнении любого аккумулятора сверху в него заносится 0x7FFFFFFF, а при переполнении снизу в него заносится 0x80000000. В операнд-приемник D выводится значение аккумулятора AC0, предшествующее выполнению инструкции.

**Тип:** OP2

**Форматы:** 1, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*

Примечания. 1) Признаки U, N, V, C формируются по новым значениям аккумуляторов: AC1н и AC0н.

2) Признак Z формируется по предшествующим значениям аккумуляторов: AC1 и AC0.

\*U=1, если оба аккумулятора, AC1н и AC0н, не нормализованы, т.е. AC1н[31]=AC1н[30] и AC0н[31]=AC0н[30] (иначе 0).

\*N=AC1n[31] – знак результата AC1n.

\*Z=1 при нулевых результатах AC1 и AC0 (иначе 0).

\*V=1 при переполнении любого из результатов, AC1 или AC0 (иначе 0), независимо от бита управления режимом насыщения CCR[8].

\*C= AC0n[31] – знак результата AC0n.

### Синтаксис ассемблера и код инструкции:

Формат 1: **SAC2**[.cc] S1,S2,D

Пример: **SAC2.ne** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				1	0	cc				0	0	0	1	1	1	1	0	1	1	0			

Автокод: SAC2 cc, S1,S2,D

Формат 8a: **SAC2** S3,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Пример: **SAC2** R2,R4,R12 ORL R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT	mode				A	de	0	0	1	1	0	1	1	0					

Автокод: PROL SAC2 S3,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **SAC2** S3,S4,D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Пример: **SAC2** R2,R4,R12 ORL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT		RD/Rd				L	0	0	1	1	1	0	1	1	0				

Автокод: PROL SAC2 S3,S4,D2 <OP1> <R/R.L**В**àR/R.L> <YRAMàR0>

Формат 8с: **SAC2**[.cc] S3,S4,D2 <OP1> <R.L**В**àR.L>

Пример: **SAC2**.eq R2,R4,R12 ORL R6,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	RS				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc		RD				0	cc		1	0	1	1	0	1	1	0			

Автокод: PROL SAC2 cc S3,S4,D2 <OP1> <R.L**В**àR.L>

Формат 8d: **SAC2** S3,S4,D2 <OP1> <R**В**àRC>

Пример: **SAC2** R2,R4,R18 ORL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				0	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	1	1	0	1	1	0				

Автокод: PROL SAC2 S3,S4,D2 <OP1> <R**В**àRC>



**SAH Сложение и вычитание двух пар чисел (short) SAH**
**Операция:**

3-адресный вариант: long S1, S2, D;

$$S1[31:16] - S1[15:0] \rightarrow D[31:16]$$

$$S2[15:0] + S2[31:16] \rightarrow D[15:0]$$

2-адресный вариант: long S, D;

$$S[31:16] - S[15:0] \rightarrow D[31:16]$$

$$D[15:0] + D[31:16] \rightarrow D[15:0]$$

**Описание:** В формате short вычисляется разность старшей и младшей части первого операнда-источника (S1 – в 3-адресном варианте; S – в 2-адресном варианте), и сумма младшей и старшей части второго операнда-источника (S2 – в 3-адресном варианте; D – в 2-адресном варианте). Результат вычитания помещается в старшую часть операнда-приемника D, результат сложения помещается в младшую часть операнда-приемника.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
U+&U-	N+	Z+	V+ V-	N-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **SAH[.cc]** S1, S2, D

Примеры: **SAH.ne** R2, R6, R12

**SAH** R2, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S1				D				S2				0	0	cc				0	0	0	1	0	1	1	1	1	1	1	1	1	1	1

Автокод: SAH cc, S1, S2, D

Формат 4: **SAH** S,D <XRAM $\beta$ àR.L>

Пример: **SAH** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	1	1	1	1	1	1

Автокод: SAH4 S, D, MLR,ea,A,R

SAH4 S, D, MLM,ea,A,R

Формат 5: **SAH** S,D <R/R.L $\beta$ àR/R.L>

Пример: **SAH** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	1	1	1	1	1	1	1	1

Автокод: SAH5 S, D, MR,rs,rd

SAH5 S, D, MRL,RS,RD

Формат 6: **SAH** S,D <R $\beta$ àRC>

Пример: **SAH** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	1	1	1	1	1	1	1	1

Автокод: SAH6 S, D, MRC,rs,RC

SAH6 S, D, MRCd,RC,rd

Формат 7: **SAH**[.cc] S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **SAH** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	RD/Rd/RC	L/sc	sr	cc	1	1	1	0	0	1	1	1	1	1	1
---	---	----------	------	----	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: SAH7 S, D, MIL, cc, #32, RD

SAH7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **SAH[s]** S1, S2, D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **SAH** R2, R4, R6 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 **SAHs** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	1	1	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2									

Автокод: PROL <OP2> SAH[s] S1, S2, D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **SAH[s]** S1, S2, D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **SAH** R2, R4, R6 R8, R15 (AT), R0

LSRL R6, R0, R8 **SAHs** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	1	1	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> SAH[s] S1, S2, D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **SAH[s]** S1, S2, D <R.LβàR.L>

Примеры: LSRL.eq R6, R0, R8 **SAH** R2, R4, R6 R8.L, R16.L

LSRL.eq R6, R0, R8 **SAHs** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	1	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> SAH[s] S1,S2,D <R. L **β** R.L>

Формат 8d: <OP2>SAH[s] S1,S2,D <R **β** RC>

Примеры: LSRL R6,R0,R8 **SAH** R2,R4,R6 CCR,R17

LSRL R6,R0,R8 **SAHs** R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	1	1	1	1	1	1	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> SAH[s] S1,S2,D <R **β** RC>

**SBC**
**Вычитание с переносом (short)**
**SBC**
**Операция:**

3-адресный вариант: short s1,s2,d; s2 - s1 - (~C) à d

2-адресный вариант: short s,d; d - s1 - (~C) à d

**Описание:** В формате short вычисляется разность операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте) и переноса C; результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:** OP1

**Форматы:** 1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **SBC[.cc]** s1,s2,d

Примеры: **SBC.ne** R1,R5,R12

**SBC** R1,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	0	1	0	0	1	0	1	0	0			

*Автокод:* SBC cc, s1, s2, d

Формат 2: **SBC[.cc]** #16,s2,d

Пример: **SBC.ne** 15,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s1	d	s2	0	0	cc	0	0	1	0	0	0	1	0	1	0	0
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32										
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16											0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Автокод: SBC2 cc, #16, s2, d

Формат 3: SBC #16, d

Пример: SBC 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16												0	0	1	1	0	0	1	0	1	0	0					

Автокод: SBC3 #16, d

Формат 4: SBC s, d <XRAM $\beta$ AR.L>

Пример: SBC R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	0	0	0	1	0	1	0	0	

Автокод: SBC4 s, d, MLR, ea, A, R

SBC4 s, d, MLM, ea, A, R

Формат 5: SBC s, d <R/R.L $\beta$ AR/R.L>

Пример: SBC R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	1	0	1	0	0			

Автокод: SBC5 s, d, MR, rs, rd

SBC5 s, d, MRL, RS, RD

Формат 6: SBC s, d <R $\beta$ ARC>

Пример: SBC R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	0	1	0	0	0	0	0	0

Автокод: SBC6 s, d, MRC, rs, RC  
SBC6 s, d, MRCd, RC, rd

Формат 7: **SBC[.cc]** s, d <#16/32àRC/R/R.L>

Пример: **SBC** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	0	1	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: SBC7 s, d, MIL, cc, #32, RD  
SBC7 s, d, MIC, cc, #16, RC

Формат 8a: <OP2>**SBC[s]** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5, R0, R8 **SBC** R1, R2, R5 R8, (A0)+ (AT), R0  
LSRL R5, R0, R8 **SBCs** R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	0	0	1	0	1	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **SBC[s]** s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2>**SBC[s]** s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R5, R0, R8 **SBC** R1, R2, R5 R8, R15 (AT), R0  
LSRL R5, R0, R8 **SBCs** R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs				0	0	0	0	0	0	0	1	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> SBC[s] s1,s2,d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс> SBC[s] s1,s2,d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R5,R0,R8 SBC R1,R2,R5 R8.L,R16.L  
 LSRL.eq R5,R0,R8 SBCs R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS				0	0	0	0	0	0	0	1	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 сс> SBC[s] s1,s2,d <R.L $\beta$ àR.L>

Формат 8d: <OP2> SBC[s] s1,s2,d <R $\beta$ àRC>

Примеры: LSRL R5,R0,R8 SBC R1,R2,R5 CCR,R17  
 LSRL R5,R0,R8 SBCs R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	Rs/Rd				0	0	0	0	0	0	1	0	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> SBC[s] s1,s2,d <R $\beta$ àRC>



**SBCL**
**Вычитание с переносом (long)**
**SBCL**
**Операция:**

3-адресный вариант: long S1,S2,D; S2 - S1 - (~C) à D

2-адресный вариант: long S,D; D - S - (~C) à D

**Описание:** В формате long вычисляется разность операндов-источников (S1,S2 - в 3-адресном варианте; S,D - в 2-адресном варианте) и переноса C; результат помещается в операнд-приемник D. В качестве первого операнда-источника S1 в 3-адресном варианте может использоваться непосредственный операнд #32.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **SBCL[.cc]** S1,S2,D

Примеры: **SBCL.ne** R2,R6,R12

**SBCL** R2,R6,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	1	0	1	0	0				

*Автокод:* SBCL cc, S1, S2, D

Формат 2: **SBCL[.cc]** #32,S2,D

Пример: **SBCL.ne** 15,R6,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	1	0	0	1	1	0	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: SBCL2 cc, #32, S2, D

Формат 4: **SBCL** S,D <XRAM $\beta$ àR.L>

Пример: **SBCL** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	0	1	0	0		

Автокод: SBCL4 S, D, MLR,ea,A,R

SBCL4 S, D, MLM,ea,A,R

Формат 5: **SBCL** S,D <R/R.L $\beta$ àR/R.L>

Пример: **SBCL** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	0	1	0	0				

Автокод: SBCL5 S, D, MR,rs,rd

SBCL5 S, D, MRL,RS,RD

Формат 6: **SBCL** S,D <R $\beta$ àRC>

Пример: **SBCL** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	0	1	0	0				

Автокод: SBCL6 S, D, MRC,rs,RC

SBCL6 S, D, MRCd,RC,rd

Формат 7: **SBCL[.cc]** S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **SBCL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	0	1	0	0	0	0	0

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **SBCL7** S, D, MIL,cc,#32,RD

**SBCL7** S, D, MIC,cc,#16,RC

Формат 8а: <OP2>**SBCL**[s] S1,S2,D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **SBCL** R2,R4,R6 R8,(A0)+ (AT),R0

LSRL R5,R0,R8 **SBCLs** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	0	1	0	0	0	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **SBCL**[s] S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2>**SBCL**[s] S1,S2,D <R/R.LβàR/R.L> YRAMàR0>

Примеры: LSRL R5,R0,R8 **SBCL** R2,R4,R6 R8,R15 (AT),R0

LSRL R5,R0,R8 **SBCLs** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	0	1	0	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> **SBCL**[s] S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Формат 8с: <OP2.сс>                    **SBCL** S1,S2,D            <R.L**β**àR.L>

*Примеры:*    LSRL.eq R5,R0,R8                    **SBCL**    R2,R4,R6            R8.L,R16.L

                  LSRL.eq R5,R0,R8                    **SBCLs** R2,R4,R6            R8.L,R16.L

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	1	0	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

*Автокод:*    PROL            <OP2 сс>                    SBCL[s] S1,S2,D                    <R.L**β**àR.L>

Формат 8d: <OP2>                    **SBCL** S1,S2,D            <R**β**àRC>

*Примеры:*    LSRL R5,R0,R8                    **SBCL**    R2,R4,R6                    CCR,R17

                  LSRL R5,R0,R8                    **SBCLs** R2,R4,R6                    CCR,R17

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	1	0	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

*Автокод:*    PROL            <OP2>                    SBCL[s] S1,S2,D                    <R**β**àRC>

**SMB**
**Сложение бит (short)**
**SMB**

**Операция:** short  $s, d; \quad \Sigma s[n] \text{ à } d$

**Описание:** Подсчитывается количество единиц в коде, хранящемся в регистре источника  $s$  и результат записывается в регистр приемника  $d$ .

**Тип:** OP2

**Форматы:** 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
–	–	√	–	–

**Синтаксис ассемблера и код инструкции:**

Формат 4: **SMB**  $s, d$  <XRAM**β**àR.L>

*Примеры:* **SMB** R1, R2 R8, (A0)+

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	1	1	0	1	1	1		

*Автокод:* SMB4  $s, d, MLR, ea, A, R$

SMB4  $s, d, MLM, ea, A, R$

Формат 5: **SMB**  $s, d$  <R/R.L**β**àR/R.L>

*Примеры:* **SMB** R1, R2 R8.L, R12.L

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	1	1	0	1	1	1				

*Автокод:* SMB5  $s, d, MR, rs, rd$

SMB5  $s, d, MRL, RS, RD$

Формат 6: **SMB**  $s, d$  <R**β**àRC>

*Примеры:* **SMB** R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				R				sc	RC				1	0	de	0	1	1	1	0	1	1	1	1	0	1	1	1

Автокод: SMB6 s, d, MRC, rs, RC

SMB6 s, d, MRCd, RC, rd

Формат 7: **SMB**[.cc] s, d <#16/32àRC/R/R.L>

Пример: **SMB** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	1	1	0	1	1	1	0	1	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: SMB7 s, d, MIL, cc, #32, RD

SMB7 s, d, MIC, cc, #16, RC

Формат 8a: **SMB** s3, d2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **SMB** R1, R5 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32		
s3				d2				0	0	0	0	0	0	AT	mode				A	de	0	0	0	1	0	1	1	1	1	1	1	1	1

Автокод: PROL SMB s3, d2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **SMB** s3, d2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **SMB** R1, R3 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	0	1	0	1	1	1			

Автокод: PROL    SMB s3,d2    <OP2>    <R/R.LβàR/R.L>    <YRAMàR0>

Формат 8с:    **SMB**[.cc] s3,d2    <OP1>    <R.LβàR.L>

Примеры:    **SMB**.eq R1,R25    ANDL R6,R0,R8    R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	cc	RD				0	cc	1	0	0	1	0	1	1	1			

Автокод: PROL    SMB cc s3,d2    <OP1>    <R.LβàR.L>

Формат 8d:    **SMB** s3,d2 <OP1>    <RβàRC>

Примеры:    **SMB** R1,R25    ANDL R6,R0,R8    CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s3				d2				0	0	0	0	0	0	0	RC				sc	de	1	1	0	1	0	1	1	1			

Автокод: PROL    SMB s3,d2    <OP1>    <RβàRC>

**SMBL**

 Сложение бит (*long*)

**SMBL**
**Операция:** long S; short d;  $\Sigma S[n]$  à d

**Описание:** Подсчитывается количество единиц в коде, хранящемся в регистре источника S и результат записывается в регистр приемника d.

**Тип:** OP2

**Форматы:** 4, 5, 6, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
-	-	√	-	-

**Синтаксис ассемблера и код инструкции:**
**Формат 4:** **SMBL** S, D <XRAMβàR.L>

**Примеры:** **SMBL** R0, R2 R8, (A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	1	1	1	1	1	1	0	1	1	1	

**Автокод:** SMBL4 S, D, MLR, ea, A, R

SMBL4 S, D, MLM, ea, A, R

**Формат 5:** **SMBL** S, D <R/R.LβàR/R.L>

**Примеры:** **SMBL** R8, R2 R8.L, R12.L

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	1	1	1	1	1	0	1	1	1			

**Автокод:** SMBL5 S, D, MR, rs, rd

SMBL5 S, D, MRL, RS, RD

**Формат 6:** **SMBL** S, D <RβàRC>

**Примеры:** **SMBL** R2, R6 R7, CCR



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D				R				sc	RC				1	0	de	1	1	1	1	0	1	1	1	1	1	1	1	0

Автокод: SMBL6 S, D, MRC, rs, RC

SMBL6 S, D, MRCd, RC, rd

Формат 8a: **SMBL** S3, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Примеры: **SMBL** R8, R12 ANDL R6, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32		
S3				D2				0	0	0	0	0	0	AT	mode				A	de	0	0	1	1	0	1	1	1	1	1	1	1	1

Автокод: PROL SMBL S3, D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Формат 8b: **SMBL** S3, D2 <OP1> <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **SMBL** R0, R4 ANDL R6, R0, R8 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32		
S3				D2				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	1	1	0	1	1	1	1	1	1	1	1

Автокод: PROL SMBL S3, D2 <OP2> <R/R.LβàR/R.L> <YRAMàR0>

Формат 8c: **SMBL**[.cc] S3, D2 <OP1> <R.LβàR.L>

Примеры: **SMBL**.eq R1, R25 ANDL R6, R0, R8 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	RS				0	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				0	0	0	0	0	0	cc	RD				0	cc	1	0	1	1	0	1	1	1			

Автокод: PROL SMBL cc S3,D2 <OP1> <R.LβàR.L>

Формат 8d: **SMBL** S3,D2<OP1> <RβàRC>

Примеры: **SMBL** R0,R24 ANDL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D/d				S2/s2				M	Rs/Rd				0	0	0	0	OP1										

6	6	6	6	5	5	5	5	5	5	53	52	51	50	49	48	47	4	4	4	4	4	4	41	40	39	38	37	36	35	34	33	32
3	2	1	0	9	8	7	6	5	4								6	5	4	3	2											
S3				D2				0	0	0	0	0	0	0	RC				sc	de	1	1	1	1	0	1	1	1				

Автокод: PROL SMBL S3,D2 <OP1> <RβàRC>

**STOP**
*Останов*
**STOP**
**Операция:** 0 à DCSR[14]; 1 à DCSR[3]

**Описание:** Выполняется переход DSP-ядра в состояние останова. Бит RUN=DCSR[14] устанавливается в «0», бит STP=DCSR[3] устанавливается в «1».

**Тип:** OP1

**Форматы:** 3m

**Признаки результата:**

U	N	Z	V	C
–	–	–	–	–

**Синтаксис ассемблера и код инструкции:**
Формат 3m: **STOP**
*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	1	0

*Автокод:* STOP

**SUB**
*Вычитание (short)*
**SUB**
**Операция:**

3-адресный вариант: short s1,s2,d;      s2 - s1 → d

2-адресный вариант: short s,d;      d - s → d

**Описание:** В формате short вычисляется разность операндов-источников (s1,s2 - в 3-адресном варианте; s,d - в 2-адресном варианте); результат помещается в операнд-приемник d. В качестве первого операнда-источника (s1 - в 3-адресном варианте; s - в 2-адресном варианте) может использоваться непосредственный операнд #16.

**Тип:**

OP1

**Форматы:**

1, 2, 3, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **SUB**[.cc] s1,s2,d

Примеры: **SUB**.ne R1,R5,R12

**SUB** R1,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				0	0	cc				0	0	0	1	0	0	1	0	0	1	1			

*Автокод:* SUB cc, s1, s2, d

Формат 2: **SUB**[.cc] #16,s2,d

Пример: **SUB**.ne 15,R5,R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

s1	d	s2	0	0	cc	0	0	1	0	0	0	1	0	0	1	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	#16															

Автокод: SUB2 cc, #16, s2, d

Формат 3: SUB #16, d

Пример: SUB 0x11, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d				#16											0	0	1	1	0	0	1	0	0	1	1						

Автокод: SUB3 #16, d

Формат 4: SUB s, d <XRAM $\beta$ àR.L>

Пример: SUB R1, R2 R8, (A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode				u	0	1	de	0	0	0	1	0	0	1	1

Автокод: SUB4 s, d, MLR, ea, A, R

SUB4 s, d, MLM, ea, A, R

Формат 5: SUB s, d <R/R.L $\beta$ àR/R.L>

Пример: SUB R1, R2 R8.L, R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	0	0	1	1				

Автокод: SUB5 s, d, MR, rs, rd

SUB5 s, d, MRL, RS, RD

Формат 6: SUB s, d <R $\beta$ àRC>

Пример: SUB R1, R5 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	0	0	1	0	0	1	1				

Автокод: SUB6 s, d, MRC, rs, RC

SUB6 s, d, MRCd, RC, rd

Формат 7: SUB[.cc] s, d <#16/32àRC/R/R.L>

Пример: SUB R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	1	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: SUB7 s, d, MIL, cc, #32, RD

SUB7 s, d, MIC, cc, #16, RC

Формат 8a: <OP2>SUB[s] s1, s2, d <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 SUB R1, R2, R5 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 SUBs R1, R2, R5 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	R				u	0	0	0	0	0	0	1	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode					A	de	0	0	#	OP2					

Автокод: PROL <OP2> SUB[s] s1, s2, d <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> SUB[s] s1, s2, d <R/R.LβàR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 SUB R1, R2, R5 R8, R15 (AT), R0

LSRL R6, R0, R8 SUBs R1, R2, R5 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS/Rs				0	0	0	0	0	0	0	1	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> SUB[s] s1,s2,d <R/R.L $\bar{\beta}$ R/R.L> <YRAM $\bar{\alpha}$ R0>

Формат 8с: <OP2.сс> SUB[s] s1,s2,d <R.L $\bar{\beta}$ R.L>

Примеры: LSRL.eq R6,R0,R8 SUB R1,R2,R5 R8.L,R16.L

LSRL.eq R6,R0,R8 SUBs R1,R2,R5 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	RS				0	0	0	0	0	0	0	1	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2								

Автокод: PROL <OP2 сс> SUB[s] s1,s2,d <R.L $\bar{\beta}$ R.L>

Формат 8d: <OP2> SUB[s] s1,s2,d <R $\bar{\beta}$ RC>

Примеры: LSRL R6,R0,R8 SUB R1,R2,R5 CCR,R17

LSRL R6,R0,R8 SUBs R1,R2,R5 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s1				d				s2				M	Rs/Rd				0	0	0	0	0	0	0	1	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: PROL <OP2> SUB[s] s1,s2,d <R $\bar{\beta}$ RC>

**SUBL**
**Вычитание (long)**
**SUBL**
**Операция:**

3-адресный вариант: long S1, S2, D;      S2 - S1 → D

2-адресный вариант: long S, D;      D - S → D

**Описание:** В формате long вычисляется разность операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); результат помещается в операнд-приемник D. В качестве первого операнда-источника S1 в 3-адресном варианте может использоваться непосредственный операнд #32 с расширением знака до 32-х разрядов.

Особенностью данной операции является то, что в параллельной инструкции (формат 8) она может использоваться как операция типа OP1 и/или OP2, в остальных форматах – только как OP1. Мнемоническое включение режима “Scaling” (**SUBLs**) возможно только при исполнении операции типа OP1.

**Тип:** OP1, OP2

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1 (OP1): **SUBL**[.cc] S1, S2, D

Примеры:      **SUBL.ne** R2, R6, R12

**SUBL**      R2, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	1	0	0	1	1				

*Автокод:*      **SUBL cc, S1, S2, D**



Формат 2(ОП1): **SUBL[.cc] #32,S2,D**

Пример: **SUBL.ne 15,R6,R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	0	1	1	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: **SUBL2 cc, #32, S2, D**

Формат 4(ОП1): **SUBL S,D <XRAMBàR.L>**

Пример: **SUBL R2,R4 R8,(A0)+**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	1	0	0	1	1			

Автокод: **SUBL4 S, D, MLR,ea,A,R**

**SUBL4 S, D, MLM,ea,A,R**

Формат 5(ОП1): **SUBL S,D <R/R.LBàR/R.L>**

Пример: **SUBL R2,R4 R8.L,R12.L**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	0	0	1	1				

Автокод: **SUBL5 S, D, MR,rs,rd**

**SUBL5 S, D, MRL,RS,RD**

Формат 6(ОП1): **SUBL S,D <RBàRC>**

Пример: **SUBL R2,R6 R7,CCR**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	0	0	1	1				

Автокод: **SUBL6 S, D, MRC,rs,RC**

SUBL6 S, D, MRCd,RC,rd

Формат 7(OP1): **SUBL[.cc]** S,D <#16/32àRC/R/R.L>

Пример: **SUBL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	0	0	1	1	0	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **SUBL7** S, D, MIL,cc,#32,RD

**SUBL7** S, D, MIC,cc,#16,RC

Формат 8a(OP1): <OP2> **SUBL[s]** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 **SUBL** R2,R4,R6 R8,(A0)+ (AT),R0

LSRL R6,R0,R8 **SUBLs** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	0	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **SUBL[s]** S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8a(OP2): **SUBL** S3,S4,D2 <OP1> <XRAMβàR.L> <YRAMàR0>

Пример: **SUBL** R2,R4,R6 OR R6,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT		mode		A		de		0	0	1	1	1	1	1	1	1	1	1	1

Автокод: PROL SUBL S3,S4,D2 <OP1> <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b(OP1): <OP2> SUBL[s] S1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R6,R0,R8 SUBL R2,R4,R6 R8,R15 (AT),R0

LSRL R6,R0,R8 SUBLs R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M		RS/Rs				0	0	0	0	0	1	1	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> SUBL[s] S1,S2,D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8b(OP2): SUBL S3,S4,D2 <OP2> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: SUBL R2,R4,R6 ORL R6,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M		RS/Rs				0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				AT		RD/Rd				L	0	0	1	1	1	1	1	1	1	1	1	1	1

Автокод: PROL SUBL S3,S4,D2 <OP1> <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c(OP1): <OP2.cc> SUBL S1,S2,D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6,R0,R8 SUBL R2,R4,R6 R8.L,R16.L

LSRL.eq R6,R0,R8 SUBLs R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M		RS				0	0	0	0	0	0	1	1	0	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2								

Автокод: PROL <OP2 cc> SUBL[s] S1,S2,D <R.LBàR.L>

Формат 8с(OP2): SUBL[.cc] S3,S4,D2 <OP1> <R.LBàR.L>

Пример: SUBL.eq R6,R0,R8 ORL R2,R4,R6 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS			0	0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				cc		RD			0	cc		1	0	1	1	1	1	1	1	1	1	1	0

Автокод: PROL SUBL cc S3,S4,D2 <OP1> <R.LBàR.L>

Формат 8d(OP1): <OP2> SUBL S1,S2,D <RBàRC>

Примеры: LSRL R6,R0,R8 SUBL R2,R4,R6 CCR,R17

LSRL R6,R0,R8 SUBLs R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd			0	0	0	0	0	1	1	0	0	1	1					

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC			sc	de	1	1	#	OP2									

Автокод: PROL <OP2> SUBL[s] S1,S2,D <RBàRC>

Формат 8d(OP2): SUBL S3,S4,D2 <OP1> <RBàRC>

Пример: SUBL R2,R4,R6 ORL R6,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	Rs/Rd			0	0	0	0	OP1											

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3				D2				S4				0	0	RC				sc	de	1	1	1	1	1	1	1	1	1	1		

Автокод: PROL      SUBL S3,S4,D2      <OP1>      <R**B**àRC>

### *SUBLR Вычитание (long) с округлением SUBLR*

**Операция:**

3-адресный вариант: long S1,S2,D; ( S2 - S1 )<sub>Round</sub> à D

2-адресный вариант: long S,D; ( D - S )<sub>Round</sub> à D

**Описание:** В формате long вычисляется разность операндов-источников (S1,S2 - в 3-адресном варианте; S,D - в 2-адресном варианте); 32-х разрядное значение результата округляется и помещается в операнд-приемник D. В качестве первого операнда-источника S1 в 3-адресном варианте может использоваться непосредственный операнд #32.

Округление производится в соответствии с режимом округления, который устанавливается 9-м разрядом (бит RND) регистра CCR.

**Тип:** OP1

**Форматы:** 1, 2, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **SUBLR[.cc]** S1,S2,D

Примеры: **SUBLR.ne** R2,R6,R12

**SUBLR** R2,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc			0	0	0	1	0	1	1	1	1	0	0				

*Автокод:* SUBLR cc, S1, S2, D

Формат 2: **SUBLR[.cc]** #32,S2,D

Пример: **SUBLR.ne** 15,R6,R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				0	0	cc				0	0	1	0	0	1	1	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#32																															

Автокод: SUBLR2 cc, #32, S2, D

Формат 4: **SUBLR** S,D <XRAM $\beta$ R.L>

Пример: **SUBLR** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	1	1	0	0		

Автокод: SUBLR4 S, D, MLR,ea,A,R

SUBLR4 S, D, MLM,ea,A,R

Формат 5: **SUBLR** S,D <R/R.L $\beta$ R/R.L>

Пример: **SUBLR** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	1	1	0	0				

Автокод: SUBLR5 S, D, MR,rs,rd

SUBLR5 S, D, MRL,RS,RD

Формат 6: **SUBLR** S,D <R $\beta$ RC>

Пример: **SUBLR** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	1	1	0	0				

Автокод: SUBLR6 S, D, MRC,rs,RC

SUBLR6 S, D, MRCd,RC,rd

Формат 7: **SUBLR**[.cc] S,D <#16/32àRC/R/R.L>

Пример: **SUBLR** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	0	0	0	0	1	1	1	1	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **SUBLR7** S, D, MIL,cc,#32,RD

**SUBLR7** S, D, MIC,cc,#16,RC

Формат 8а: <OP2>**SUBLR**[s] S1,S2,D <XRAMβàR.L> <YRAMàR0>

Примеры: **LSRL** R6,R0,R8 **SUBLR** R2,R4,R6 R8,(A0)+ (AT),R0

**LSRL** R6,R0,R8 **SUBLRs** R2,R4,R6 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	1	1	0	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2								

Автокод: **PROL** <OP2> **SUBLR**[s] S1,S2,D <XRAMβàR.L> <YRAMàR0>

Формат 8b: <OP2> **SUBLR**[s] S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>

Примеры: **LSRL** R6,R0,R8 **SUBLR** R2,R4,R6 R8,R15 (AT),R0

**LSRL** R6,R0,R8 **SUBLRs** R2,R4,R6 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: **PROL** <OP2> **SUBLR**[s] S1,S2,D <R/R.LβàR/R.L> <YRAMàR0>



Формат 8с: <OP2.сс>                    **SUBLR** S1,S2,D <R.L**β**àR.L>

*Примеры*:    LSRL.eq R6,R0,R8            **SUBLR**    R2,R4,R6    R8.L,R16.L

                  LSRL.eq R6,R0,R8            **SUBLRs** R2,R4,R6    R8.L,R16.L

*Код инструкции*:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	1	1	1	0	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

*Автокод*: PROL            <OP2 сс>    SUBLR[s] S1,S2,D <R.L**β**àR.L>

Формат 8d: <OP2>                    **SUBLR** S1,S2,D <R**β**àRC>

*Примеры*:    LSRL R6,R0,R8            **SUBLR**    R2,R4,R6    CCR,R17

                  LSRL R6,R0,R8            **SUBLRs** R2,R4,R6    CCR,R17

*Код инструкции*:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	1	1	1	0	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

*Автокод*: PROL            <OP2>            SUBLR[s] S1,S2,D            <R**β**àRC>

**SUBLRTR      Вычитание (long) с округлением и  
преобразованием формата (в short)      SUBLRTR**

**Операция:**

3-адресный вариант: long S1, S2, d;  $(S2 - S1)_{\text{Round}} \rightarrow d[15:0]$

2-адресный вариант: long S, D;  $(D - S)_{\text{Round}} \rightarrow D[15:0]$

**Описание:** В формате long вычисляется разность операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте); 32-разрядное значение результата округляется до 16-разрядного значения и помещается в операнд-приемник D/d.

Округление производится в соответствии с режимом округления, установленным битом RND=CCR[9].

В форматах 1, 8 в качестве регистра приемника используется 16-разрядный регистр.

В форматах 4, 5, 6, 7 – 16-разрядный результат помещается в младшие 16 разрядов регистра приемника, старшие 16 разрядов регистра приемника не изменяются.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
√	√	√	√	√

**Синтаксис ассемблера и код инструкции:**

Формат 1: **SUBLRTR**[.cc] S1, S2, d

Примеры: **SUBLRTR**.ne R2, R6, R12

**SUBLRTR**      R2, R6, R12

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				0	0	cc				0	0	0	1	0	1	1	1	1	0	1			

Автокод: `SUBLRTR cc, S1, S2, d`

Формат 4: `SUBLRTR S,D<XRAM $\beta$ aR.L>`

Пример: `SUBLRTR R2,R4 R8,(A0)+`

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode	u	0	1	de	0	0	1	1	1	1	0	1			

Автокод: `SUBLRTR4 S, D, MLR,ea,A,R`

`SUBLRTR4 S, D, MLM,ea,A,R`

Формат 5: `SUBLRTR S,D<R/R.L $\beta$ aR/R.L>`

Пример: `SUBLRTR R2,R4 R8.L,R12.L`

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	1	1	1	1	0	1				

Автокод: `SUBLRTR5 S, D, MR,rs,rd`

`SUBLRTR5 S, D, MRL,RS,RD`

Формат 6: `SUBLRTR S,D<R $\beta$ aRC>`

Пример: `SUBLRTR R2,R6 R7,CCR`

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	1	1	0	1				

Автокод: `SUBLRTR6 S, D, MRC,rs,RC`

`SUBLRTR6 S, D, MRCd,RC,rd`

Формат 7: `SUBLRTR[.cc] S,D<#16/32 $\beta$ aRC/R/R.L>`

Пример: `SUBLRTR R2,R0 0x12345678,R16.L`

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	1	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: SUBLRTR7 S, D, MIL, cc, #32, RD

SUBLRTR7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **SUBLRTR[s]** S1, S2, d <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **SUBLRTR** R2, R4, R6 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 **SUBLRTRs** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				M	R				u	0	0	0	0	0	0	1	1	1	1	0	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode			A	de	0	0	#	OP2							

Автокод: PROL <OP2> SUBLRTR[s] S1, S2, d <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> **SUBLRTR[s]** S1, S2, d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **SUBLRTR** R2, R4, R6 R8, R15 (AT), R0

LSRL R6, R0, R8 **SUBLRTRs** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				M	RS/Rs				0	0	0	0	0	0	1	1	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	RD/Rd			L	0	0	1	#	OP2							

Автокод: PROL <OP2> SUBLRTR[s] S1, S2, d <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **SUBLRTR** S1, S2, d <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6, R0, R8 **SUBLRTR** R2, R4, R6 R8.L, R16.L

LSRL.eq R6, R0, R8 **SUBLRTRs** R2, R4, R6 R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				M	RS				0	0	0	0	0	0	1	1	1	1	0	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD				0	cc		1	0	#	OP2							

Автокод: PROL <OP2 cc>      SUBLRTR[s] S1,S2,d <R.L~~B~~R.L>

Формат 8d: <OP2>      SUBLRTR S1,S2,d      <R~~B~~RC>

Примеры: LSRL R6,R0,R8      SUBLRTR R2,R4,R6      CCR,R17

LSRL R6,R0,R8      SUBLRTRs R2,R4,R6      CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				d				S2				M	Rs/Rd				0	0	0	0	0	1	1	1	1	0	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL      <OP2>      SUBLRTR[s] S1,S2,d      <R~~B~~RC>

**SUBX**
**Вычитание комплексное (X16)**
**SUBX**
**Операция:**
3-адресный вариант: X16 S1, S2, D;

 $S2[31:16] - S1[31:16] \rightarrow D[31:16]$ 
 $S2[15:0] - S1[15:0] \rightarrow D[15:0]$ 
2-адресный вариант: X16 S, D;

 $D[31:16] - S[31:16] \rightarrow D[31:16]$ 
 $D[15:0] - S[15:0] \rightarrow D[15:0]$ 

**Описание:** В формате X16 вычисляется разность операндов-источников (S1, S2 - в 3-адресном варианте; S, D - в 2-адресном варианте), при этом независимо вычисляется разность старших и младших 16 разрядов операндов-источников; результат вычитания старших разрядов операндов помещается в старшие 16 разрядов операнда-приемника D, а результат вычитания младших разрядов помещается в младшие 16 разрядов операнда-приемника D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
Ur&Ui	Nr	Zr	Vr Vi	Cr

**Синтаксис ассемблера и код инструкции:**
Формат 1: **SUBX[.cc]** S1, S2, D

Пример: **SUBX.ne** R2, R6, R12

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1	D	S2	0	0	cc	0	0	0	1	0	1	1	1	0	1	1
----	---	----	---	---	----	---	---	---	---	---	---	---	---	---	---	---

Автокод: SUBX cc, S1, S2, D

Формат 4: **SUBX** S,D <XRAM $\beta$ àR.L>

Пример: **SUBX** R2,R4 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A				mode		u	0	1	de	0	0	1	1	1	0	1	1		

Автокод: SUBX4 S, D, MLR,ea,A,R

SUBX4 S, D, MLM,ea,A,R

Формат 5: **SUBX** S,D <R/R.L $\beta$ àR/R.L>

Пример: **SUBX** R2,R4 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	0	0	1	1	1	0	1	1			

Автокод: SUBX5 S, D, MR,rs,rd

SUBX5 S, D, MRL,RS,RD

Формат 6: **SUBX** S,D <R $\beta$ àRC>

Пример: **SUBX** R2,R6 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	0	1	1	1	0	1	1				

Автокод: SUBX6 S, D, MRC,rs,RC

SUBX6 S, D, MRCd,RC,rd

Формат 7: **SUBX[.cc]** S,D <#16/32 $\beta$ àRC/R/R.L>

Пример: **SUBX** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	RD/Rd/RC	L/sc	sr	cc	1	1	1	0	0	1	1	1	0	1	1
---	---	----------	------	----	----	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: SUBX7 S, D, MIL, cc, #32, RD

SUBX7 S, D, MIC, cc, #16, RC

Формат 8a: <OP2> **SUBX**[s] S1, S2, D <XRAM $\beta$ àR.L> <YRAMàR0>

Примеры: LSRL R6, R0, R8 **SUBX** R2, R4, R6 R8, (A0)+ (AT), R0

LSRL R6, R0, R8 **SUBXs** R2, R4, R6 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	R				u	0	0	0	0	0	0	1	1	1	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	mode			A	de	0	0	#	OP2									

Автокод: PROL <OP2> SUBX[s] S1, S2, D <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2>

Примеры: LSRL R6, R0, R8 **SUBX** R2, R4, R6 R8, R15 (AT), R0

LSRL R6, R0, R8 **SUBXs** R2, R4, R6 R8, R15 (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS/Rs				0	0	0	0	0	0	1	1	1	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> SUBX[s] S1, S2, D <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> **SUBX** S1, S2, D <R.L $\beta$ àR.L>

Примеры: LSRL.eq R6, R0, R8 **SUBX** R2, R4, R6 R8.L, R16.L

LSRL.eq R6, R0, R8 **SUBXs** R2, R4, R6 R8.L, R16.L



Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	RS				0	0	0	0	0	0	1	1	1	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2									

Автокод: PROL <OP2 cc> SUBX[s] S1,S2,D <R.L $\beta$ àR.L>

Формат 8d: <OP2> SUBX S1,S2,D <R $\beta$ àRC>

Примеры: LSRL R6,R0,R8 SUBX R2,R4,R6 CCR,R17

LSRL R6,R0,R8 SUBXs R2,R4,R6 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1				D				S2				M	Rs/Rd				0	0	0	0	0	1	1	1	0	1	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> SUBX[s] S1,S2,D <R $\beta$ àRC>

**SWL Перестановка (long) SWL**

**Операция:** long S,D; {S[15:0],S[31:16]} **à** D

**Описание:** Выполняется перестановка частей содержимого операнда-источника S: старшие 16 разрядов S пересылаются в младшие 16 разрядов операнда-приемника D, младшие 16 разрядов S пересылаются в старшие 16 разрядов D.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	-	-

**Синтаксис ассемблера и код инструкции:**

Формат 1: **SWL**[.cc] S,D

Пример: **SWL**.ne R0,R2

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S					D					0	0	0	0	0	0	0	cc					0	0	0	1	1	0	1	1	0	1	1

Автокод: SWL cc, S, D

Формат 4: **SWL** S,D <XRAM**β****à**R.L>

Пример: **SWL** R0,R2 R8,(A0)+

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					D					R			A			mode	u	0	1	de	0	1	0	1	1	0	1	1			

Автокод: SWL4 S, D, MLR,ea,A,R

SWL4 S, D, MLM,ea,A,R

Формат 5: **SWL** S,D <R/R.L**β****à**R/R.L>

Пример: **SWL** R1,R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	1	0	1	1				

Автокод: SWL5 S, D, MR,rs,rd  
 SWL5 S, D, MRL,RS,RD

Формат 6: **SWL** S,D <RβàRC>

Пример: **SWL** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	1	0	1	1				

Автокод: SWL6 S, D ,MRC,rs,RC  
 SWL6 S, D ,MRCd,RC,rd

Формат 7: **SWL[.cc]** S,D #16/32àRC/R/R.L>

Пример: **SWL** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	1	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: SWL7 S, D, MIL,cc,#32,RD  
 SWL7 S, D, MIC,cc,#16,RC

Формат 8а: <OP2> **SWL** S,D <XRAMβàR.L> <YRAMàR0>

Примеры: LSRL R5,R0,R8 **SWL** R0,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	1	0	1	1	0	1	1		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		mode			A		de	0	0	#	OP2								

Автокод: PROL <OP2> SWL S,D <XRAM $\beta$ aR.L> <YRAM $\alpha$ R0>

Формат 8b: <OP2> SWL S,D <R/R.L $\beta$ aR/R.L> <YRAM $\alpha$ R0>

Примеры: LSRL R5,R0,R8 SWL R0,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
S				D				0				0				0				0				RS/Rs				0	0	0	0	1	0	1	1	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT		RD/Rd			L	0	0	1	#	OP2									

Автокод: PROL <OP2> SWL S,D <R/R.L $\beta$ aR/R.L> <YRAM $\alpha$ R0>

Формат 8c: <OP2.cc> SWL S,D <R.L $\beta$ aR.L>

Примеры: LSRL.eq R5,R0,R8 SWL R0,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
S				D				0				0				0				0				RS				0	0	0	0	0	1	0	1	1	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2								

Автокод: PROL <OP2 cc> SWL S,D <R.L $\beta$ aR.L>

Формат 8d: <OP2> SWL S,D <R $\beta$ aRC>

Примеры: LSRL R5,R0,R8 SWL R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
S				D				0				0				0				Rs/Rd				0	0	0	0	1	0	1	1	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0		0		RC			sc	de	1	1	#	OP2							

Автокод: PROL <OP2> SWL S,D <R**B**ARC>

**TR**
**Транзит (short)**
**TR**
**Операция:**      short s,d;                      s à d

**Описание:** В формате short выполняется пересылка содержимого регистра источника s в регистр приемника d. Особенностью данной операции является то, что в параллельной инструкции (формат 8) она может использоваться как операция типа OP1 и/или OP2, в остальных форматах – только как OP1.

**Тип:**                      OP1, OP2

**Форматы:**              4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**
**Формат 4 (OP1):** **TR** s,d      <XRAM $\beta$ àR.L>

**Пример:**              **TR** R1,R2    R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				A				mode		u	0	1	de	0	1	0	0	1	0	1	0		

**Автокод:**      **TR**4 s, d, MLR,ea,A,R

**TR**4 s, d, MLM,ea,A,R

**Формат 5 (OP1):** **TR** s,d      <R/R.L $\beta$ àR/R.L>

**Пример:**              **TR** R1,R2    R8.L,R12.L

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RS/Rs				L	RD/Rd				1	1	0	0	1	0	0	1	0	1	0				

**Автокод:**      **TR**5 s, d, MR,rs,rd

**TR**5 s, d, MRL,RS,RD

Формат 6 (OP1): **TR** s,d <R**β**RC>

Пример: **TR** R1,R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				R				sc	RC				1	0	de	0	1	0	0	1	0	1	0				

Автокод: **TR6** s, d ,MRC,rs,RC

**TR6** s, d ,MRCd,RC,rd

Формат 7 (OP1): **TR**[.cc] s,d #16/32**α**RC/R/R.L>

Пример: **TR** R2,R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	0	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **TR7** s, d, MIL,cc,#32,RD

**TR7** s, d, MIC,cc,#16,RC

Формат 8a (OP1): <OP2> **TR** s,d <XRAM**β**R.L> <YRAM**α**R0>

Примеры: LSRL R5,R0,R8 **TR** R1,R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	R				u	0	0	0	0	1	0	0	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **TR** s,d <XRAM**β**R.L> <YRAM**α**R0>

Формат 8a (OP2): **TR** s,d <OP1> <XRAM**β**R.L> <YRAM**α**R0>

Примеры: **TR** R1,R2 ORL R5,R0,R8 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	R				u	0	0	0	0	OP1									

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
s				d				0	0	0	0	0	0	AT	mode				A				de	0	0	0	0	0	0	1	1	0

Автокод: PROL TR s,d <OP1> <XRAMβR.L> <YRAMαR0>

Формат 8b (OP1): <OP2> TR s,d <R/R.LβR/R.L> <YRAMαR0>

Примеры: LSRL R5,R0,R8 TR R1,R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	0	RS/Rs				0	0	0	0	1	0	0	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL <OP2> TR s,d <R/R.LβR/R.L> <YRAMαR0>

Формат 8b (OP2): TR s,d <OP1> <R/R.LβR/R.L> <YRAMαR0>

Примеры: TR R1,R2 ORL R5,R0,R8 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s				d				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	0	0	0	1	1	0			

Автокод: PROL TR s,d <OP1> <R/R.LβR/R.L> <YRAMαR0>

Формат 8c (OP1): <OP2.cc> TR s,d <R.LβR.L>

Примеры: LSRL.eq R5,R0,R8 TR R1,R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---



s	d	0	0	0	0	0	0	RS	0	0	0	0	0	1	0	0	1	0	1	0
---	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2								

Автокод: PROL <OP2 cc> TR s,d <R.LβàR.L>

Формат 8с (OP2): TR[.cc] s,d <OP1> <R.LβàR.L>

Примеры: TR.eq R1,R2 ORL R5,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS			0	0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
s				d				0	0	0	0	0	cc		RD			0	cc		1	0	0	0	0	0	1	1	0		

Автокод: PROL TR cc s,d <OP1> <R.LβàR.L>

Формат 8d (OP1): <OP2> TR s,d <RβàRC>

Примеры: LSRL R5,R0,R8 TR R1,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				d				0	0	0	0	0	0	Rs/Rd			0	0	0	0	1	0	0	1	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC			sc	de	1	1	#	OP2									

Автокод: PROL <OP2> TR s,d <RβàRC>

Формат 8d (OP2): TR s,d <OP1> <RβàRC>

Примеры: TR R1,R2 ORL R5,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	Rs/Rd			0	0	0	0	OP1											

6	6	6	6	5	5	5	5	5	5	53	52	51	50	49	48	47	4	4	4	4	4	41	40	39	38	37	36	35	34	33	32
3	2	1	0	9	8	7	6	5	4								6	5	4	3	2										
s				d						0	0	0	0	0	0	0	RC					sc	de	1	1	0	0	0	1	1	0

АВТОКОД: PROL      TR      s, d      <OP1>      <R**B**àRC>

**TRL**    *Транзит (long)*
**TRL**
**Операция:**    long S,D;

S → D

**Описание:** В формате long выполняется пересылка содержимого регистра источника S в регистр приемника D. Особенностью данной операции является то, что в параллельной инструкции (формат 8) она может использоваться как операция типа OP1 и/или OP2, в остальных форматах – только как OP1.

**Тип:**    OP1

**Форматы:**    4, 5, 6, 7, 8

**Признаки результата:**

<b>U</b>	<b>N</b>	<b>Z</b>	<b>V</b>	<b>C</b>
-	-	-	-	-

**Синтаксис ассемблера и код инструкции:**
**Формат 4 (OP1):**    **TRL** S,D    <XRAMβàR.L>

**Пример:**    **TRL** R2,R0    R8,(A0)+

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				A		mode		u	0	1	de	0	1	0	1	1	0	1	0				

**Автокод:**    **TRL4** S, D, MLR,ea,A,R

**TRL4** S, D, MLM,ea,A,R

**Формат 5 (OP1):**    **TRL** S,D    <R/R.LβàR/R.L>

**Пример:**    **TRL** R2,R4    R8.L,R12.L

**Код инструкции:**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RS/Rs				L	RD/Rd				1	1	0	0	1	0	1	1	0	1	0				

**Автокод:**    **TRL5** S, D, MR,rs,rd

**TRL5** S, D, MRL,RS,RD

Формат 6 (OP1): **TRL** S, D <R**β**àRC>

Пример: **TRL** R0, R2 R7, CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				R				sc	RC				1	0	de	0	1	0	1	1	0	1	0				

Автокод: **TRL6** S, D ,MRC,rs,RC

**TRL6** S, D ,MRCd,RC,rd

Формат 7 (OP1): **TRL**[.cc] S, D #16/32àRC/R/R.L>

Пример: **TRL** R2, R0 0x12345678, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	1	0	1	1	0	1	0			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: **TRL7** S, D, MIL,cc,#32,RD

**TRL7** S, D, MIC,cc,#16,RC

Формат 8а (OP1): <OP2> **TRL** S, D <XRAM**β**àR.L> <YRAMàR0>

Примеры: LSRL R5, R0, R8 **TRL** R0, R2 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	R				u	0	0	0	0	1	0	1	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> **TRL** S, D <XRAM**β**àR.L> <YRAMàR0>

Формат 8а (OP2): **TRL** S, D <OP1> <XRAM**β**àR.L> <YRAMàR0>

Примеры: **TRL** R1, R2 ORL R5, R0, R8 R8, (A0)+ (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	R				u	0	0	0	0	OP1									

6	6	6	6	5	5	5	5	5	5	53	52	51	50	49	4	4	4	4	4	4	4	4	4	40	39	38	37	36	35	34	33	32
3	2	1	0	9	8	7	6	5	4						8	7	6	5	4	3	2	1										
S				D				0	0	0	0	0	0	AT	mode				A	de	0	0	0	0	0	0	1	1	1	0		

Автокод: PROL      TRL      S, D      <OP1>      <XRAMβR.L>      <YRAMαR0>

Формат 8b (OP1):      <OP2>      **TRL** S, D <R/R.LβR/R.L>      <YRAMαR0>

Примеры:      LSRL R5, R0, R8      **TRL** R0, R2      R8, R15      (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	0	RS/Rs				0	0	0	0	1	0	1	1	0	1	0		

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2									

Автокод: PROL      <OP2>      TRL      S, D      <R/R.LβR/R.L>      <YRAMαR0>

Формат 8b (OP2):      **TRL** S, D      <OP1>      <R/R.LβR/R.L>      <YRAMαR0>

Примеры:      **TRL** R1, R2      ORL R5, R0, R8      R8, R15      (AT), R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS/Rs				0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S				D				0	0	0	0	0	0	AT	RD/Rd				L	0	0	1	0	0	1	1	1	0			

Автокод: PROL      TRL      S, D      <OP1>      <R/R.LβR/R.L>      <YRAMαR0>

Формат 8с (OP1):      <OP2.cc>      **TRL** S, d      <R.LβR.L>

Примеры:      LSRL.eq R5, R0, R8      **TRL** R0, R2      R8.L, R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	D	0	0	0	0	0	0	RS	0	0	0	0	0	1	0	1	1	0	1	0
---	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				cc		RD			0	cc		1	0	#	OP2								

Автокод: PROL <OP2 cc> TRL S,D <R.LBàR.L>

Формат 8с (OP2): TRL[.cc] S,D <OP1> <R.LBàR.L>

Примеры: TRL.eq R1,R2 ORL R5,R0,R8 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S1/s1				D1/d1				S2/s2				M	RS			0	0	0	0	0	OP1										

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S				D				0	0	0	0	0	cc		RD			0	cc		1	0	0	0	1	1	1	0			

Автокод: PROL TRL cc S,D <OP1> <R.LBàR.L>

Формат 8d (OP1): <OP2> TRL S,D <RBàRC>

Примеры: LSRL R5,R0,R8 TRL R0,R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				D				0	0	0	0	0	0	Rs/Rd			0	0	0	0	1	0	1	1	0	1	0				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC			sc	de	1	1	#	OP2									

Автокод: PROL <OP2> TRL S,D <RBàRC>

Формат 8d (OP2): TRL S,D <OP2> <RBàRC>

Примеры: TRL R1,R2 ORL R5,R0,R8 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S1/s1	D1/d1	S2/s2	M	Rs/Rd	0	0	0	0	OP1
-------	-------	-------	---	-------	---	---	---	---	-----

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S				D				0	0	0	0	0	0	0	RC				sc	de	1	1	0	0	1	1	1	0			

Автокод: PROL    TRL    S,D    <OP1>    <R**3**aRC>

### *TST* Определение признаков операнда (*short TST*)

**Операция:** short s; {U,N,Z} à CCR[4:2]

**Описание:** Определяются признаки 16-разрядного операнда-источника s.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	–

**Синтаксис ассемблера и код инструкции:**

Формат 1: **TST[.cc] s**

*Пример:* **TST.ne R12**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				s				s				0	0	cc			0	0	0	1	0	0	0	0	0	0	1	1			

*Автокод:* TST cc s

Формат 4: **TST s <XRAMβàR.L>**

*Пример:* **TST R2R8, (A0)+**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				s				R				A				mode		u	0	1	de	0	0	0	0	0	0	0	1	1	

*Автокод:* TST4 s, MLR, ea, A, R

TST4 s, MLM, ea, A, R

Формат 5: **TST s <R/R.LβàR/R.L>**

*Пример:* **TST R2 R8.L, R12.L**

*Код инструкции:*



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				s				RS/Rs				L	RD/Rd				1	1	0	0	0	0	0	0	0	0	0	0	1	1	

Автокод: TST5 d, MR,rs,rd

TST5 d, MRL,RS,RD

Формат 6: **TST** s <RβàRC>

Пример: **TST** R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				s				R				sc	RC				1	0	de	0	0	0	0	0	0	0	0	0	1	1	

Автокод: TST6 s ,MRC,rs,RC

TST6 s ,MRCd,RC,rd

Формат 7: **TST**[.cc] s #16/32àRC/R/R.L>

Пример: **TST** R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				s				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	0	0	0	0	0	0	1	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: TST7 s, MIL,cc,#32,RD

TST7 s, MIC,cc,#16,RC

Формат 8а: <OP2> **TST** s <XRAMβàR.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 **TST** R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				s				s				0	R				u	0	0	0	0	0	0	0	0	0	0	0	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> TST s <XRAM $\beta$ aR.L> <YRAMaR0>

Формат 8b: <OP2> TST s <R/R.L $\beta$ aR/R.L> <YRAMaR0>

Пример: LSRL R5,R0,R8 TST R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s				s				s				0	RS/Rs				0	0	0	0	0	0	0	0	0	0	0	0	1	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> TST s <R/R.L $\beta$ aR/R.L> <YRAMaR0>

Формат 8c: <OP2.cc> TST s <R.L $\beta$ aR.L>

Пример: LSRL.eq R5,R0,R8 TST R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				s				s				0	RS				0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2								

Автокод: PROL <OP2 cc> TST s <R.L $\beta$ aR.L>

Формат 8d: <OP2> TST s <R $\beta$ aRC>

Пример: LSRL R5,R0,R8 TST R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
s				s				s				0	Rs/Rd				0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: PROL <OP2> TST s <R $\beta$ aRC>

### TSTL Определение признаков операнда (long) TSTL

**Операция:** long S; {U,N,Z} à CCR[4:2]

**Описание:** Определяются признаки 32-разрядного операнда-источника S.

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
√	√	√	–	–

**Синтаксис ассемблера и код инструкции:**

Формат 1: **TSTL[.cc] S**

Пример: **TSTL.ne R12**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	0	cc				0	0	0	1	0	1	0	0	0	1	1			

Автокод: TSTL cc S

Формат 4: **TSTL S<XRAMβàR.L>**

Пример: **TSTL R2 R8,(A0)+**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				R				A				mode	u	0	1	de	0	0	1	0	0	0	1	1			

Автокод: TSTL4 S, MLR, ea, A, R

TSTL4 S, MLM, ea, A, R

Формат 5: **TSTL S<R/R.LβàR/R.L>**

Пример: **TSTL R2 R8.L,R12.L**

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	0	0	1	1				

Автокод: TSTL5 S, MR,rs,rd

TSTL5 S, MRL,RS,RD

Формат 6: **TSTL** S<R $\beta$ RC>

Пример: **TSTL** R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				R				sc	RC				1	0	de	0	0	1	0	0	0	1	1				

Автокод: TSTL6 S ,MRC,rs,RC

TSTL6 S ,MRCd,RC,rd

Формат 7: **TSTL**[.cc] S #16/32 $\beta$ RC/R/R.L>

Пример: **TSTL** R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: TSTL7 S, MIL,cc,#32,RD

TSTL7 S, MIC,cc,#16,RC

Формат 8а: <OP2> **TSTL** S <XRAM $\beta$ R.L> <YRAM $\beta$ R0>

Пример: LSRL R5,R0,R8 **TSTL** R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	R				u	0	0	0	0	0	1	0	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				AT	mode				A	de	0	0	#	OP2									

Автокод: PROL <OP2> TSTL S <XRAM $\beta$ aR.L> <YRAMaR0>

Формат 8b: <OP2> TSTL S <R/R.L $\beta$ aR/R.L> <YRAMaR0>

Пример: LSRL R5,R0,R8 TSTL R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	RS/Rs				0	0	0	0	0	0	1	0	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				AT	RD/Rd				L	0	0	1	#	OP2								

Автокод: PROL <OP2> TSTL S <R/R.L $\beta$ aR/R.L> <YRAMaR0>

Формат 8c: <OP2.cc> TSTL s <R.L $\beta$ aR.L>

Пример: LSRL.eq R5,R0,R8 TSTL R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	RS				0	0	0	0	0	0	1	0	0	0	1	1			

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				cc	RD				0	cc	1	0	#	OP2								

Автокод: PROL <OP2 cc> TSTL S <R.L $\beta$ aR.L>

Формат 8d: <OP2> TSTL S <R $\beta$ aRC>

Пример: LSRL R5,R0,R8 TSTL R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	Rs/Rd				0	0	0	0	0	1	0	0	0	1	1				

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2							

Автокод: PROL <OP2> TSTL S <R $\beta$ aRC>

**TSTX Определение признаков операнда (X16) TSTX**

**Операция:** (X16) S; {U,N,Z,V,C} à CCR[4:0]

**Описание:** Определяются признаки 32-разрядного операнда-источника S, рассматриваемого как комплексное число в формате X16..

**Тип:** OP1

**Форматы:** 1, 4, 5, 6, 7, 8

**Признаки результата:**

U	N	Z	V	C
*	*	*	*	*
Ur&Ui	Nr	Zr	Zi	Ni

**Синтаксис ассемблера и код инструкции:**

Формат 1: **TSTX[.cc] S**

*Пример:* **TSTX.ne R12**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				S				0	0	cc				0	0	0	1	0	1	0	1	0	1	1			

*Автокод:* TSTX cc S

Формат 4: **TSTX S<XRAMβàR.L>**

*Пример:* **TSTX R2 R8,(A0)+**

*Код инструкции:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S				S				R				A				mode	u	0	1	de	0	0	1	0	1	0	1	1			

*Автокод:* TSTX4 S, MLR, ea, A, R

TSTX4 S, MLM, ea, A, R

Формат 5: **TSTX S<R/R.LβàR/R.L>**

Пример: **TSTX** R2 R8.L,R12.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					S					RS/Rs				L	RD/Rd				1	1	0	0	0	1	0	1	0	1	1		

Автокод: TSTX5 S, MR,rs,rd  
TSTX5 S, MRL,RS,RD

Формат 6: **TSTX** S<R**β**àRC>

Пример: **TSTX** R2 R7,CCR

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					S					R				sc	RC				1	0	de	0	0	1	0	1	0	1	1		

Автокод: TSTX6 S ,MRC,rs,RC  
TSTX6 S ,MRCd,RC,rd

Формат 7: **TSTX[.cc]** S #16/32àRC/R/R.L>

Пример: **TSTX** R0 0x12345678,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					S					RD/Rd/RC				L/sc	sr	cc				1	1	1	0	0	1	0	1	0	1	1	

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
#16/32																															

Автокод: TSTX7 S, MIL,cc,#32,RD  
TSTX7 S, MIC,cc,#16,RC

Формат 8а: <OP2> **TSTX** S <XRAM**β**àR.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 **TSTX** R2 R8,(A0)+ (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

S	S	S	0	R	u	0	0	0	0	0	0	1	0	1	0	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	mode			A	de	0	0	#	OP2							

Автокод: PROL <OP2> TSTX S <XRAM $\beta$ àR.L> <YRAMàR0>

Формат 8b: <OP2> TSTX S <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Пример: LSRL R5,R0,R8 TSTX R2 R8,R15 (AT),R0

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					S					S					0	RS/Rs					0	0	0	0	0	1	0	1	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					AT	RD/Rd					L	0	0	1	#	OP2					

Автокод: PROL <OP2> TSTX S <R/R.L $\beta$ àR/R.L> <YRAMàR0>

Формат 8c: <OP2.cc> TSTX s <R.L $\beta$ àR.L>

Пример: LSRL.eq R5,R0,R8 TSTX R2 R8.L,R16.L

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S					S					S					0	RS					0	0	0	0	0	0	1	0	1	0	1	1

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5					D2/d2					S4/s4					cc	RD			0	cc	1	0	#	OP2							

Автокод: PROL <OP2 cc> TSTX S <R.L $\beta$ àR.L>

Формат 8d: <OP2> TSTX S <R $\beta$ àRC>

Пример: LSRL R5,R0,R8 TSTX R2 CCR,R17

Код инструкции:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
S					S					S					0	Rs/Rd					0	0	0	0	0	1	0	1	0	1	1



63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
S3/s3/#5				D2/d2				S4/s4				0	0	RC				sc	de	1	1	#	OP2								

Автокод: PROL <OP2> TSTX S <R**В**àRC>

## 8. РАБОТА ПРОГРАММНОГО КОНВЕЙЕРА И ВРЕМЯ ИСПОЛНЕНИЯ ИНСТРУКЦИЙ

Программный конвейер включает в себя четыре стадии (фазы): стадию формирования адреса программной памяти (A), стадию выборки команды из программной памяти (Fetch), стадию декодирования команды (Decode), стадию исполнения (Execute).

Конвейеризация приводит к тому, что в один и тот же момент времени происходит обработка нескольких инструкций, находящихся на разных стадиях. При этом для большинства инструкций скорость их выполнения в конвейерном режиме составляет одну инструкцию в течение одного командного цикла. Исключение составляют инструкции программных переходов.

В приводимой ниже таблице указано время исполнения инструкций DSP-ядра Elcore-26, измеряемое в командных циклах. Время исполнения дается в предположении, что командный конвейер заполнен, т.е. исполняемая команда уже выбрана и декодирована.

Инструкция	Время исполнения
Все инструкции, кроме B, BD, BS, J, JD, JS, RTS, ENDDO	1
B, BD, BS, J, JD, JS, RTS	2
ENDDO	3

### 8.1 Работа программного конвейера при последовательной выборке команд

Работа программного конвейера при последовательной выборке команд из программной памяти иллюстрируется временной диаграммой рис.8.1 (n – номер инструкции).

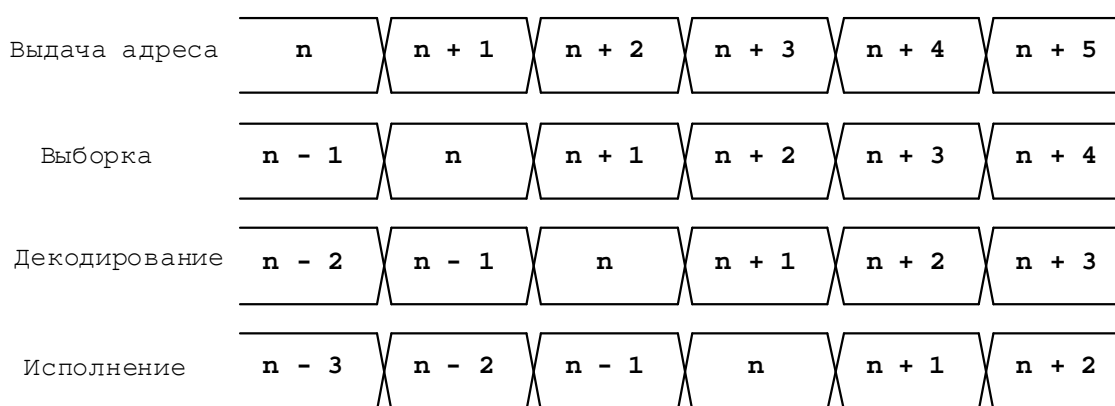


Рисунок 8.1 Работа программного конвейера при последовательной выборке команд

Начальное состояние всех трех регистров – нулевое:

$$PCA = PCF = PCD = PCE = 0x0000.$$

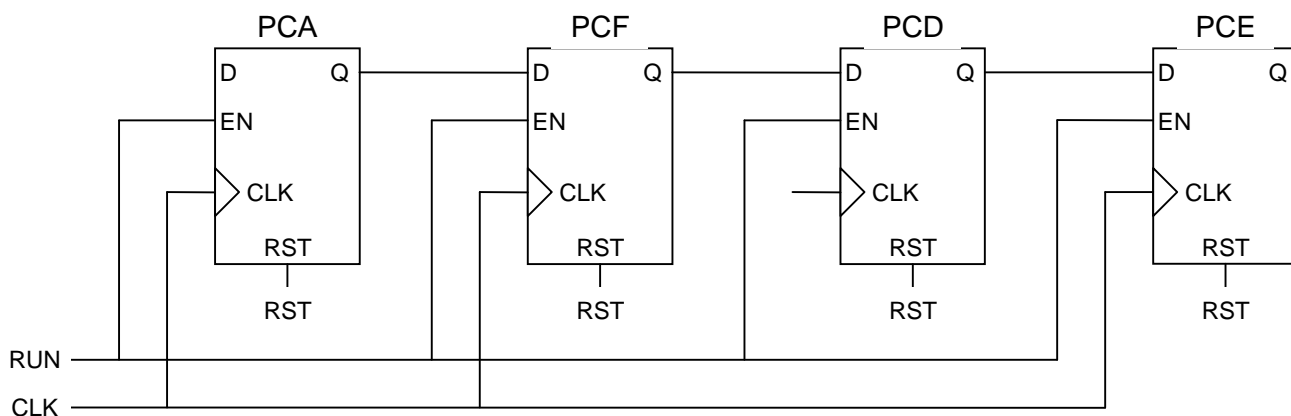


Рисунок 8.2 Аппаратная реализация программного счетчика

Состояния регистров PCA, PCF, PCD, PCE после начала работы иллюстрируются временной диаграммой рис.8.3.

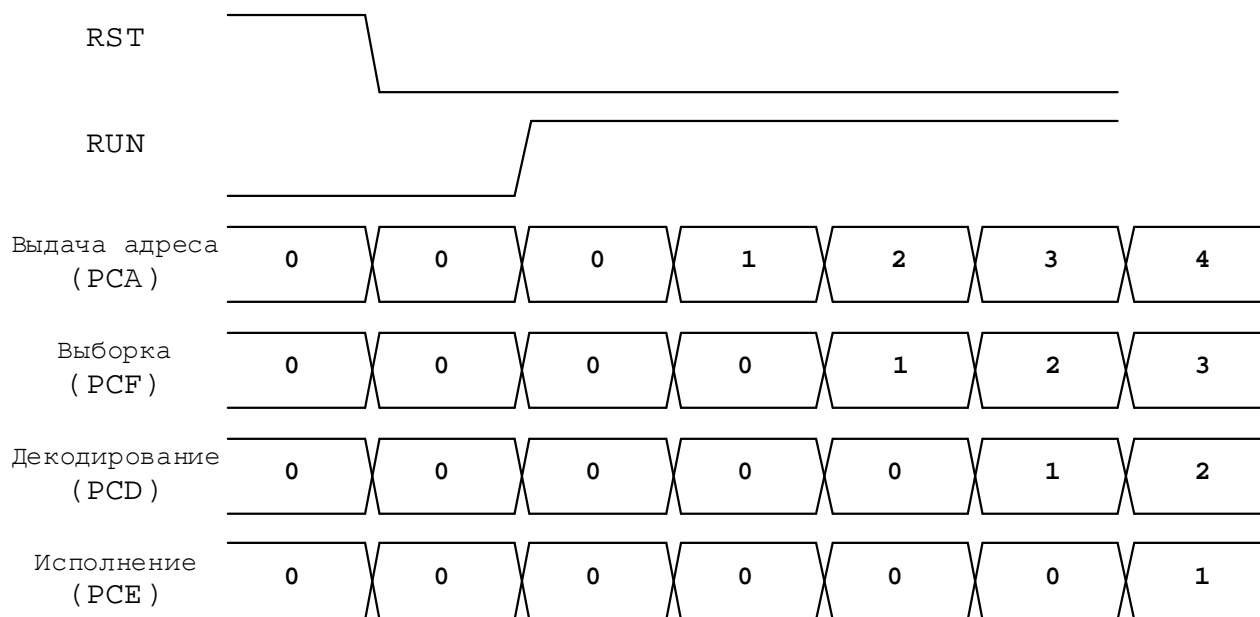


Рисунок 8.3 Работа программного конвейера после начальной установки

Обращения к программному счетчику происходят следующим образом: запись данных производится в первый регистр квадруплета – PCA, чтение производится из последнего регистра – PCE. Это обстоятельство следует иметь в виду программисту: если чтение PC производится сразу после записи в него, то прочитано будет предшествующее значение, а не только что записанное.

## 8.2 Работа программного конвейера при программных переходах

Работа программного конвейера при программных переходах (команды B, BS, J, JS, RTS) иллюстрируется временной диаграммой рис.8.4 (A – адрес перехода). Инструкция, следующая за инструкцией программного перехода, не выполняется (заменяется на NOP).

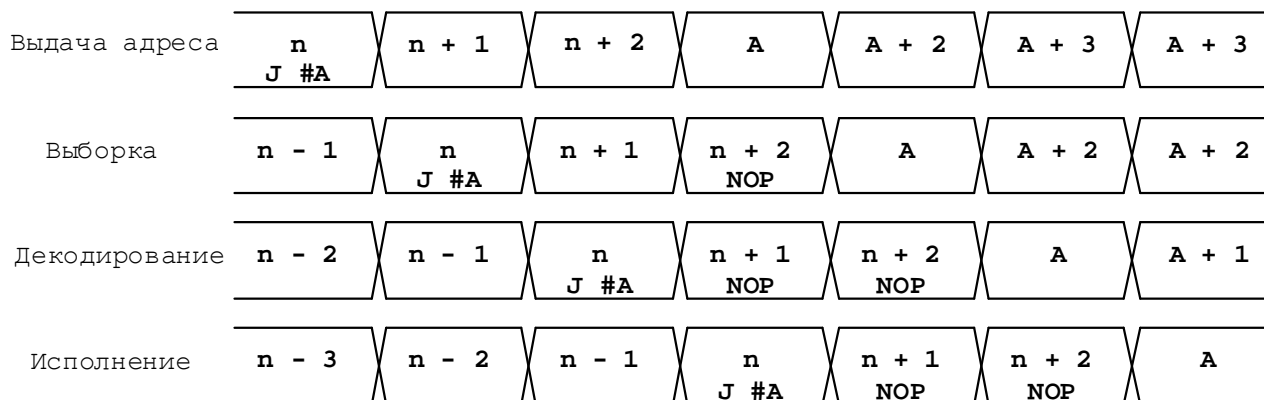


Рисунок 8.4 Работа программного конвейера при программных переходах (команды B, BS, J, JS, RTS).

При отложенных программных переходах (команды BD, JD) инструкция, следующая за инструкцией программного перехода, выполняется (рис.8.5).

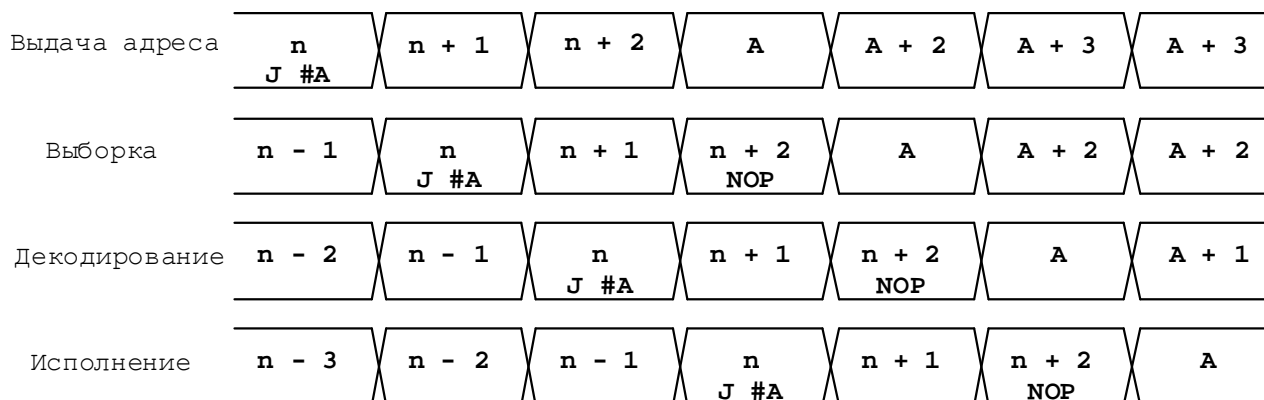


Рисунок 8.5 Работа программного конвейера при отложенных программных переходах (команды BD, JD).

Действие команды ENDDO иллюстрируется временной диаграммой рис.8.6 (LA – адрес последней инструкции цикла).

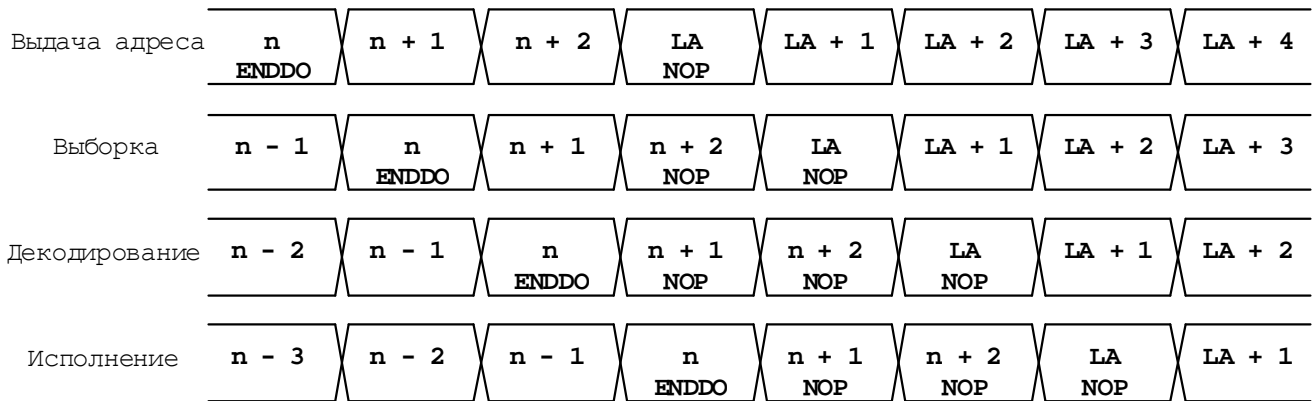


Рисунок 8.6 Работа программного конвейера при выполнении команды ENDDO

### 8.3 Работа программного конвейера при исполнении циклов

Работа программного конвейера при исполнении циклов иллюстрируется временной диаграммой рис.8.7 (LA – адрес последней инструкции цикла).

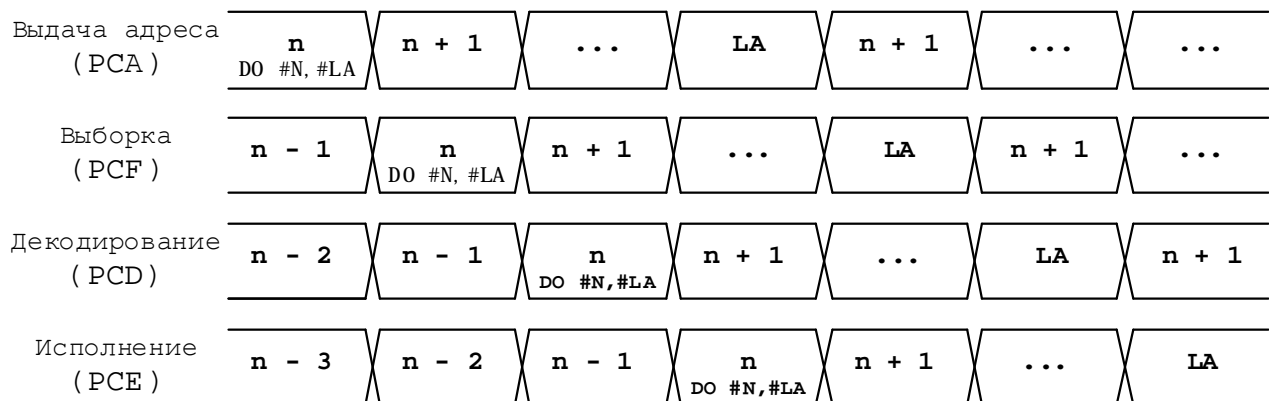


Рисунок 8.7 Работа программного конвейера при исполнении циклов.

## 8.4 Работа программного конвейера при исполнении команды STOP

Работа программного конвейера при исполнении команды STOP иллюстрируется временной диаграммой рис.8.9.

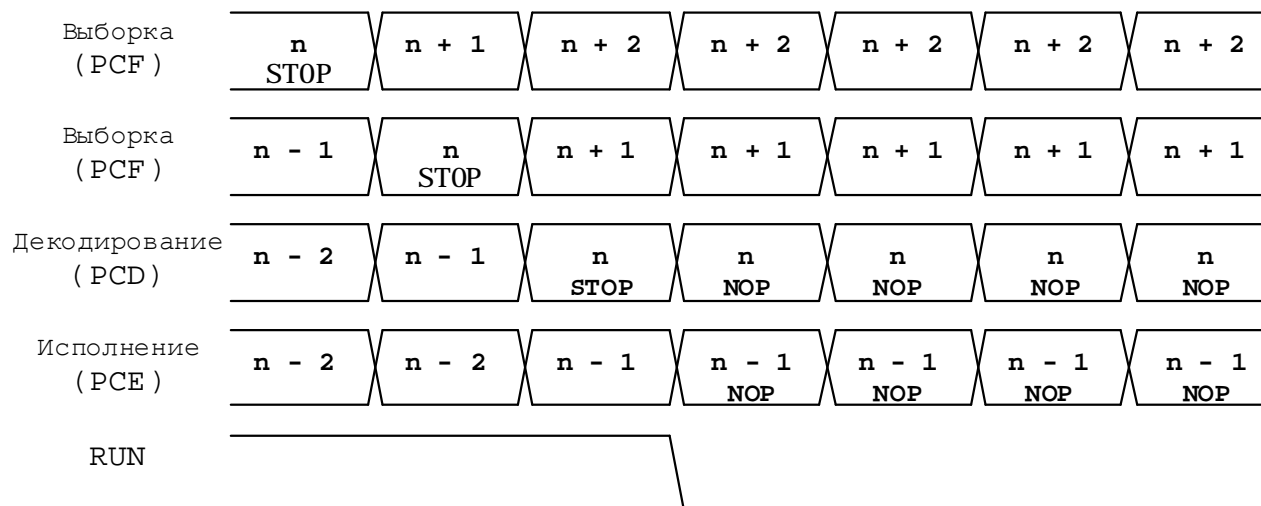


Рисунок 8.9 Работа программного конвейера при исполнении команды STOP

Команда STOP, в отличие от остальных, исполняется уже на стадии декодирования, за исключением тех случаев, когда непосредственно перед ней стоит команда программного перехода. В результате исполнения команды STOP бит “RUN” сбрасывается в “0”, программный конвейер останавливается, а инструкции, находящейся на стадии декодирования и исполнения, заменяются на NOP. Флаг STP (4-й разряд DCSR) по команде STOP устанавливается в «1» и стоит до тех пор, пока не будет сброшен по команде CPU.

## 9. ОГРАНИЧЕНИЯ ПРИ ИСПОЛНЕНИИ ИНСТРУКЦИЙ

### 9.1 Ограничение на адреса результатов одновременно исполняемых операций

- 1) Одновременно исполняемые вычислительные операции и пересылки не должны иметь одинаковые адреса операндов-приемников (регистров данных). Ассемблер DSP-ядра дает в этом случае предупреждение.

Примечание. Регистры CCR, PDNR, AC0, AC1 допустимо использовать в качестве приемника одновременно для вычислительной операции и пересылки. Пример:

MAC2 R0,R2,R4 R6,AC0

Приоритет в подобных случаях имеет операция пересылки.

### 9.2 Ограничения при выполнении инструкций программного управления

- 1) Заданное количество повторений цикла DO (регистр LC) должно находиться в пределах от 1 (от 2 - для циклов, состоящих из одной инструкции) до 16383.
- 2) Количество вложенных циклов DO, DOFOR не должно превышать семи (ограничение связано с глубиной стека циклов). Допускаются вложения только одноименных циклов – циклы DO вкладываются в циклы DO, циклы DOFOR - в циклы DOFOR;
- 3) Количество вложенных друг в друга циклов (DO, DOFOR) и подпрограмм (BScC, JScC) не должно превышать пятнадцать (ограничение связано с глубиной системного стека);
- 4) Цикл DO, DOFOR не может оканчиваться на команду программного управления - DO, DOFOR, B, J, BD, JD, BS, JS, RTS, ENDDO;
- 5) Цикл DO, DOFOR не может оканчиваться на ту же инструкцию, что и вложенный в него цикл. Если вложенный цикл состоит из одной инструкции, между его окончанием и последней инструкцией внешнего цикла должна быть еще хотя бы одна инструкция.
- 6) Адрес последней команды исполняемого цикла DO, DOFOR не может использоваться как адрес перехода для команд B, J, BD, JD, BS, JS. (Пояснение: переход на метку конца цикла возможен в тех случаях, когда данный цикл не запущен);
- 7) Непосредственно после команды отложенного перехода BD, JD не может следовать команда программного управления - DO, DOFOR, B, J, BD, JD, BS, JS, RTS, ENDDO;
- 8) Запись в регистры LA, LC, SP, запись/чтение из стеков SS, CSH, CSL во время исполнения цикла DO, DOFOR может привести к неправильной работе цикла; запись в регистр SP, запись/чтение из стека SS во время исполнения подпрограммы может привести к неправильной работе подпрограммы.
- 9) Если команда STOP стоит в середине текста программы, необходимо после нее вставить один NOP.

### 9.3 Ограничения при выполнении инструкций пересылок

К запрещенным комбинациям команд относятся следующие:

- 1) Регистры CCR, PDNR, AC0, AC1 недоступны для пересылок непосредственных данных (форматы 3, 7);
- 2) Команда ASRLE не может сочетаться с пересылкой, в которой источником является какой-либо регистр RF;
- 3) В параллельных условных инструкциях (формат 8с), в которых явно не указана пересылка, в операциях CS2, MAC, MAC2, MPYL, SAC2, MACL нельзя использовать в качестве адреса результата регистр R0.L. Рекомендация: надо явно указывать межрегистровую пересылку;
- 4) Запись бита YM в регистр SR дает эффект не на следующем такте, а через такт;
- 5) Непосредственно перед командой STOP нельзя ставить команду пересылки в DCSR (MOVE xxx, DCSR).



## 10. ОСОБЕННОСТИ ИСПОЛНЕНИЯ ИНСТРУКЦИЙ В РЕЖИМАХ SIMD И SCALAR

### 10.1 Управление режимом функционирования SIMD/SCALAR

Переключение режима SIMD/SCALAR осуществляется при помощи 15-го бита регистра SR (бит SI). Команда переключения имеет вид: MOVE <R>,SR или MOVE <#16>,SR и не должна сопровождаться вычислительной операцией.

В режиме SIMD (SR[15]=1) в каждой вычислительной секции одновременно выполняются одни и те же инструкции обработки и пересылок из общего потока инструкций. При этом в обоих вычислительных секциях формируются одинаковые адреса регистровых файлов и одинаковые адреса памяти данных.

В режиме SCALAR (SR[15]=0) работает *вычислительная секция 0*. При этом объем внутренней памяти XRAM увеличивается в 2 раза по сравнению с режимом SIMD.

### 10.2 Особенности выполнения обменов между регистрами управления и регистрами данных в режиме SIMD.

В режиме SIMD при пересылке данных из регистра управления в регистр данных пересылаемые данные записываются в соответствующие регистры обеих секций.

При обратной пересылке - из регистра регистрового файла в регистр управления источником пересылаемых данных является соответствующий регистр нулевой секции.

### 10.3 Особенности исполнения условных инструкций.

Каждая вычислительная секция содержит регистр кода условий (CCR), в который после исполнения каждой арифметической операции записываются признаки сформированного результата. Эти признаки называются *секционными (локальными)*.

На основе локальных признаков в регистре SR формируются *интегральные признаки*.

В скалярном режиме интегральные признаки совпадают с соответствующими разрядами регистра CCR0 0-й секции.

В режиме SIMD способ формирования интегральных признаков определяется в зависимости от управляющего кода SRSI (разряды 14-13 регистра SR) согласно приводимой ниже таблице.

SRSI	Способ определения интегральных условий
00	Использование CCR0 нулевой секции
01	Объединение секционных условий по “И”
10	Объединение секционных условий по “ИЛИ”
11	Резерв

При исполнении условных инструкций проверка истинности заданного условия производится:

- по локальным признакам – при исполнении арифметических инструкций и/или инструкций пересылок данных внутри регистрового файла;
- по интегральным признакам – во всех остальных случаях.