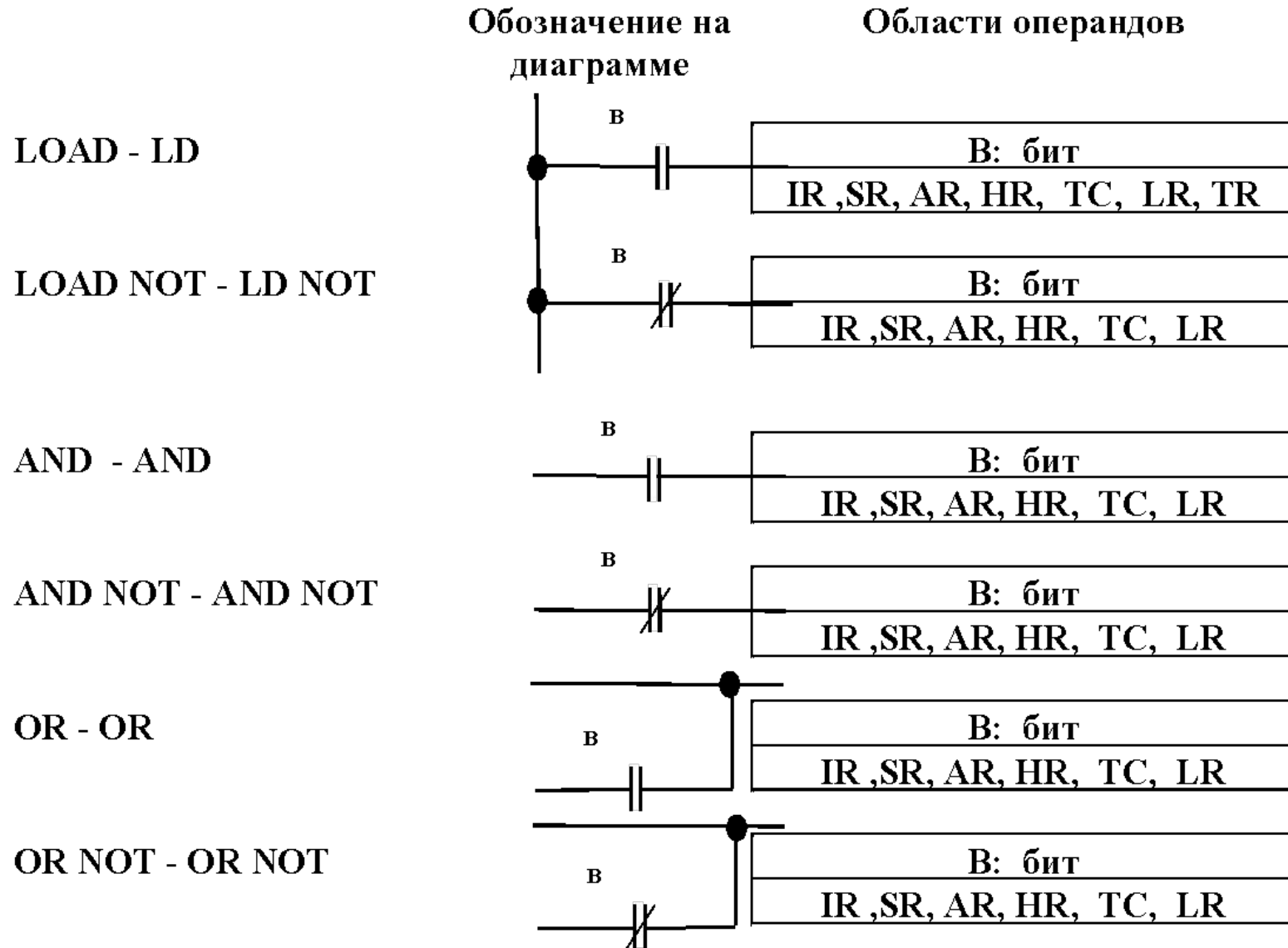


# Набор команд СРМ 1А

- ✓ Команды Ladder
- ✓ Базовые команды
- ✓ Команды данных
- ✓ Математические команды
- ✓ Логические команды
- ✓ Команды управления программой
- ✓ Коммуникационные команды
- ✓ Системные команды

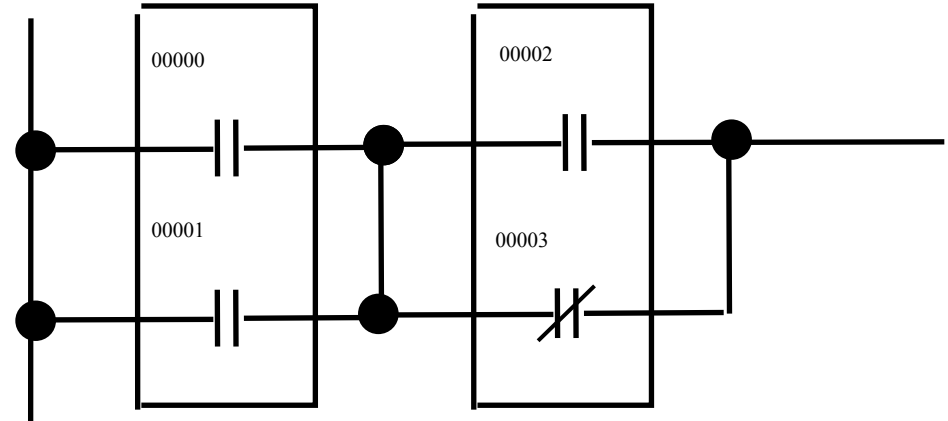
# Команды Ladder

## LOAD, LOAD NOT, AND, AND NOT, OR и OR NOT

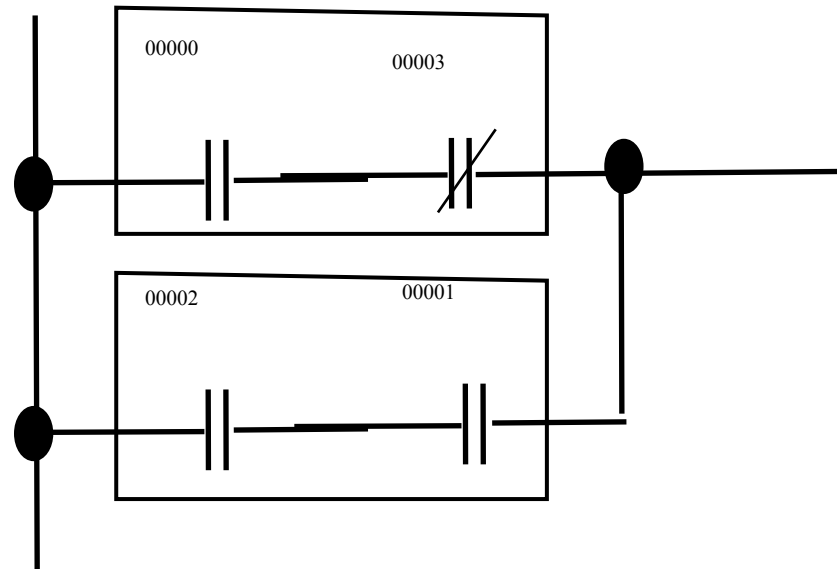


# Команды Ladder

AND LOAD и OR LOAD



AND LOAD - AND LD



OR LOAD - OR LD

# Базовые команды

- управление битами
- таймеры и счетчики
- управление программой
- подпрограммы

# Команды управления битами

Обозначение на диаграмме

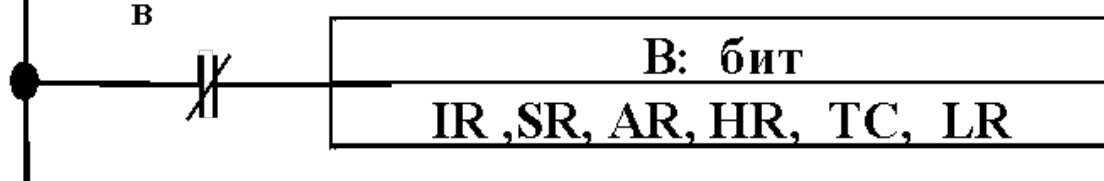
Области битов

## Вывод и Вывод инверсии

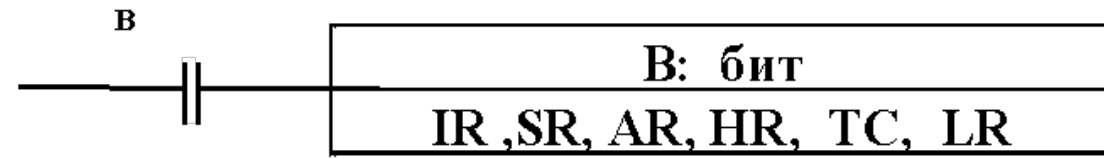
LOAD - LD



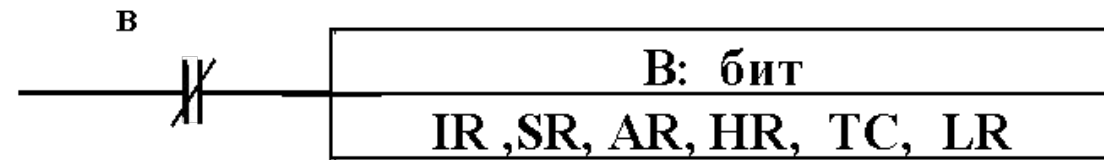
LOAD NOT - LD NOT



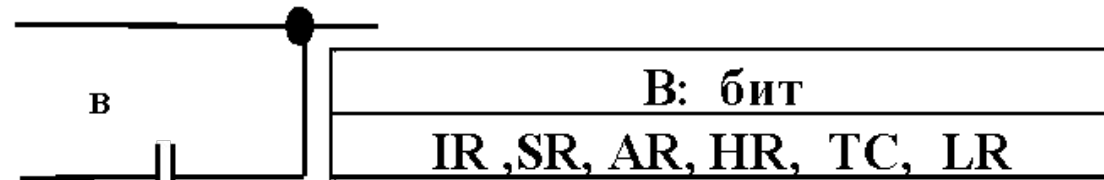
AND - AND



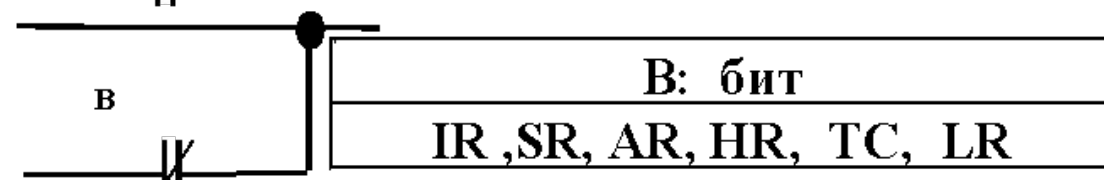
AND NOT - AND NOT



OR - OR



OR NOT - OR NOT



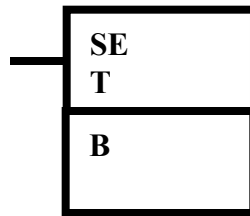
# Команды управления битами

## SET, RESET и KEEP

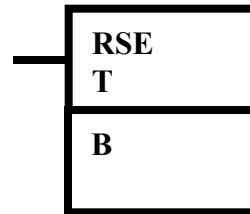
Обозначение на  
диаграмме

Области операндов

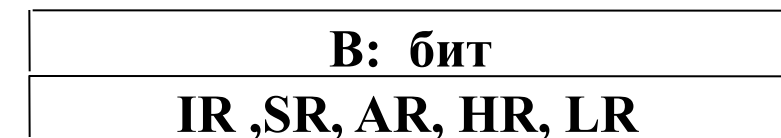
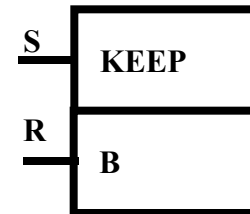
SET



RESET



KEEP



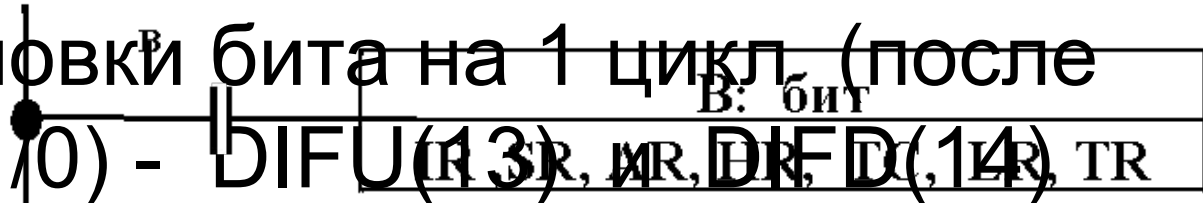
# Команды управления битами

Обозначение на диаграмме

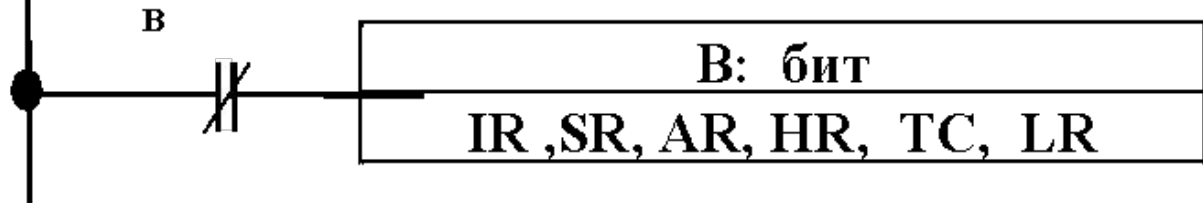
Области операндов

Команды установки бита на 1 цикл, (после фронта 0/1 и 1/0) -

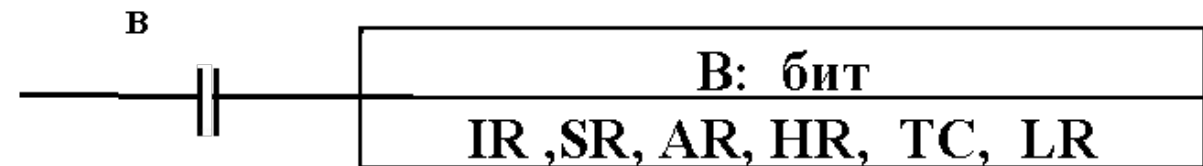
LOAD - LD



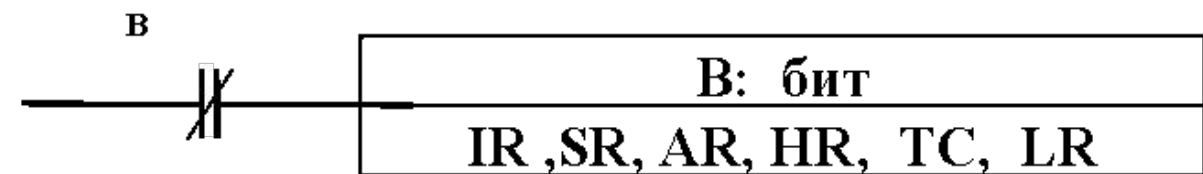
LOAD NOT - LD NOT



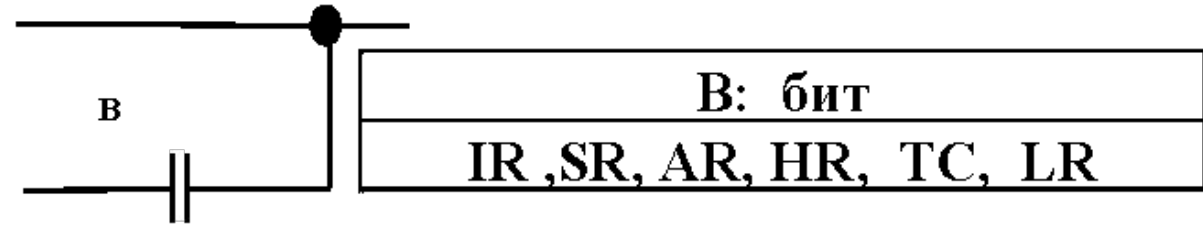
AND - AND



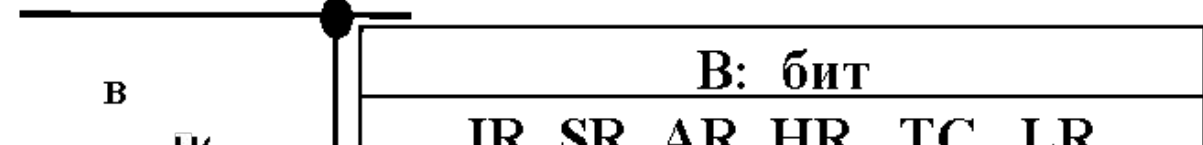
AND NOT - AND NOT



OR - OR



OR NOT - OR NOT



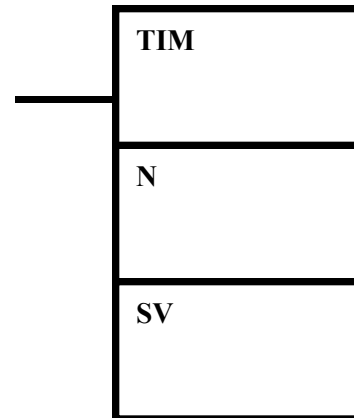
# Таймеры и счетчики

## Таймер с дискретой 100 ms.

Обозначение на  
диаграмме

Значения определителей

TIM



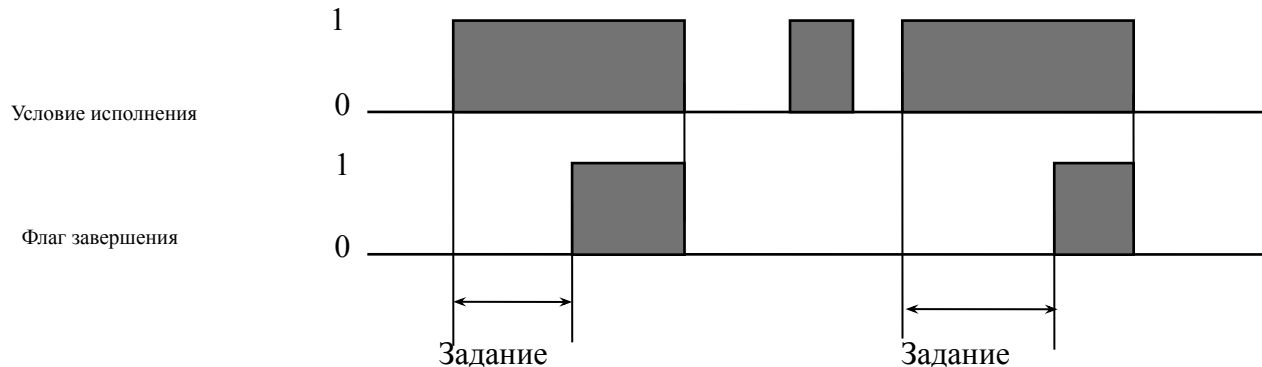
**N:** номер таймера

# (0 - 127)

Области данных операнда

**SV:** Заданное значение, (слово,  
в двоично-десятичном виде)

IR, SR, AR, DM, HR, LR, #  
(#0000 ... #9999)





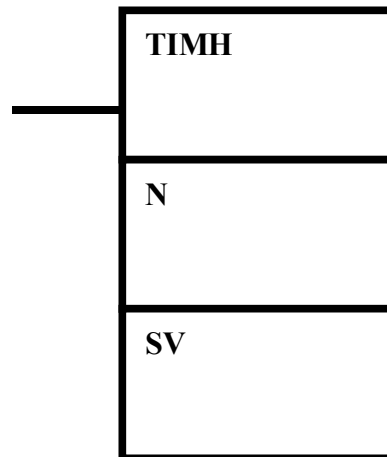
# Таймеры и счетчики

Высокоскоростной таймер с дискретой 10 ns.

Обозначение на  
диаграмме

Значения определителей

TIMH



**N:** номер таймера

# (0 - 3)

Области данных операнда

**SV:** Заданное значение, (слово,  
в -десятичном виде)

дв. и ш. AR, DM, HR, LR, #  
(#0000 ... #9999)

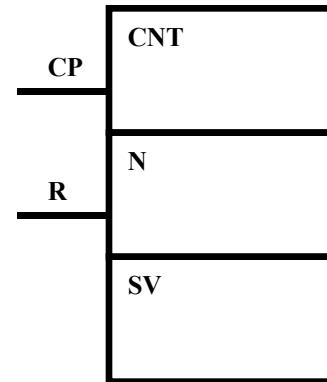
# Таймеры и счетчики

## Счетчик

Обозначение на  
диаграмме

Значения определителей

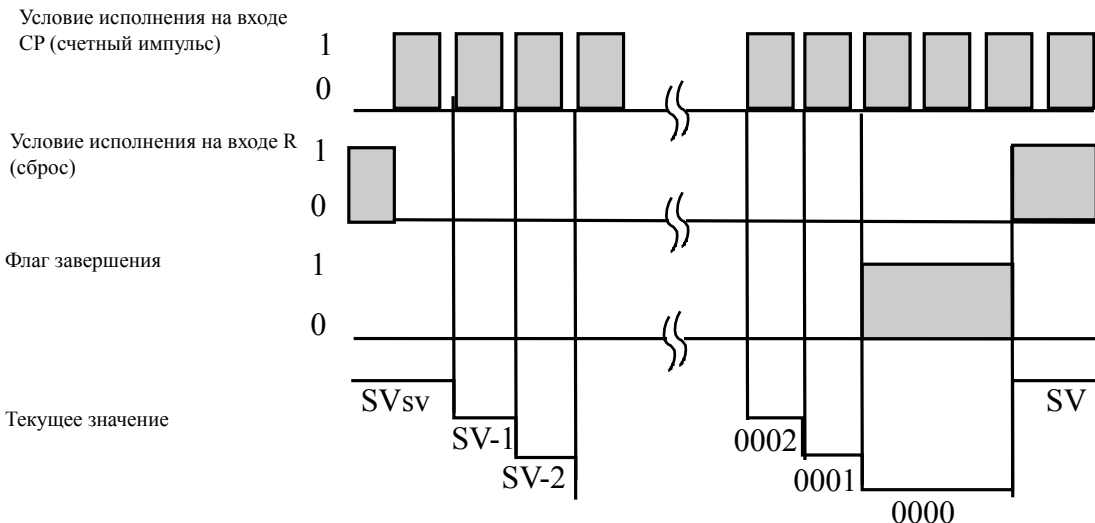
CNT



**N: номер счетчика**  
**# (0 - 127)**

Области данных операнда

**SV: Заданное значение, (слово,  
в двоично-десятичном виде)**  
**IR, SR, AR, DM, HR, LR, #**  
**(#0000 ... #9999)**



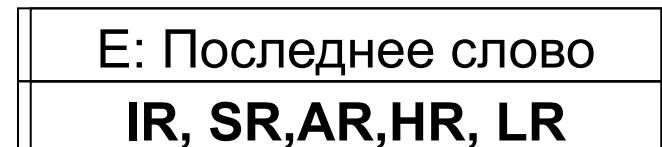
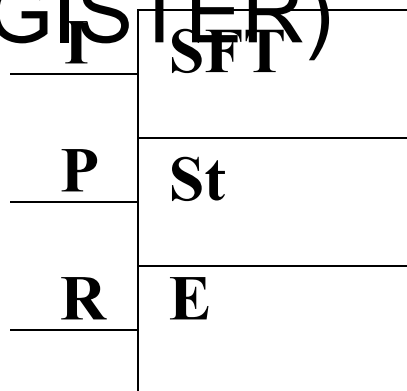


# Команды данных

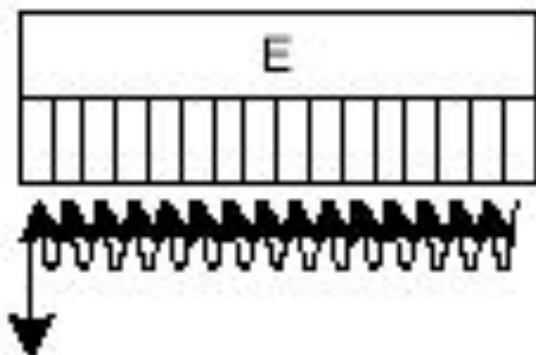
- команды сдвига
- команды пересылки данных
- команды сравнения
- команды преобразования

# Команды сдвига

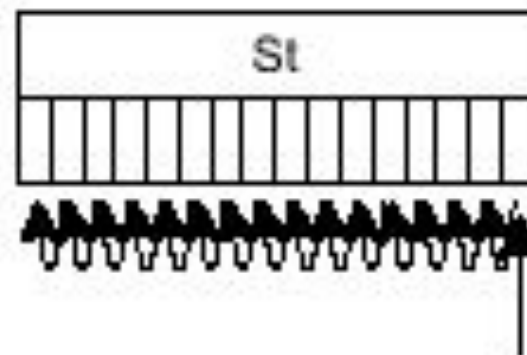
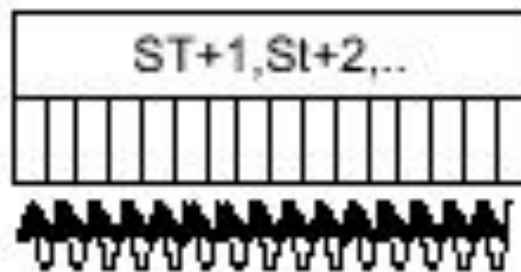
Сдвиговой регистр - SFT(10) (SHIFT REGISTER)



I – состояние, записываемое в свободный бит при сдвиге;  
P- бит управления сдвигом;  
R – бит обнуления регистра сдвига.



Потерянные данные

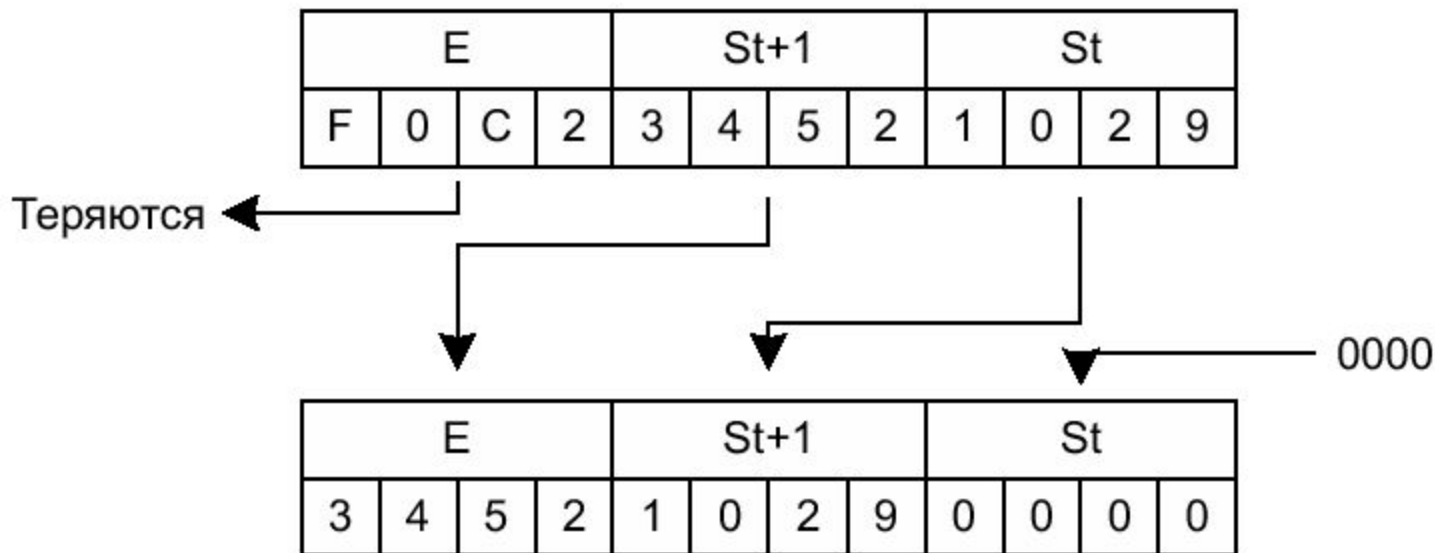
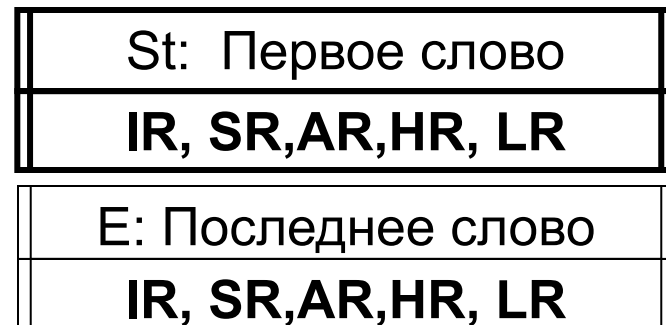
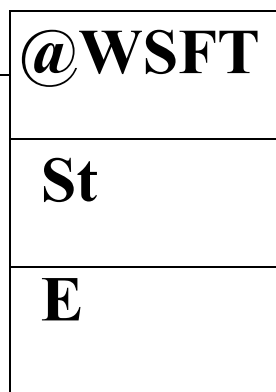
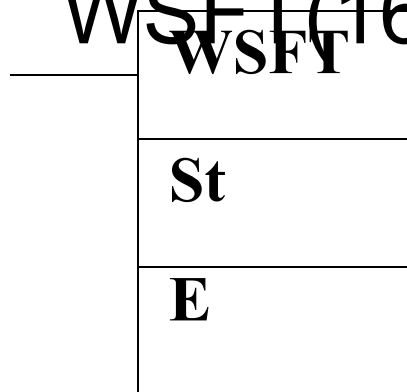


Условия на входе I

# Команды сдвига

Сдвиг слова (WORD SHIFT -

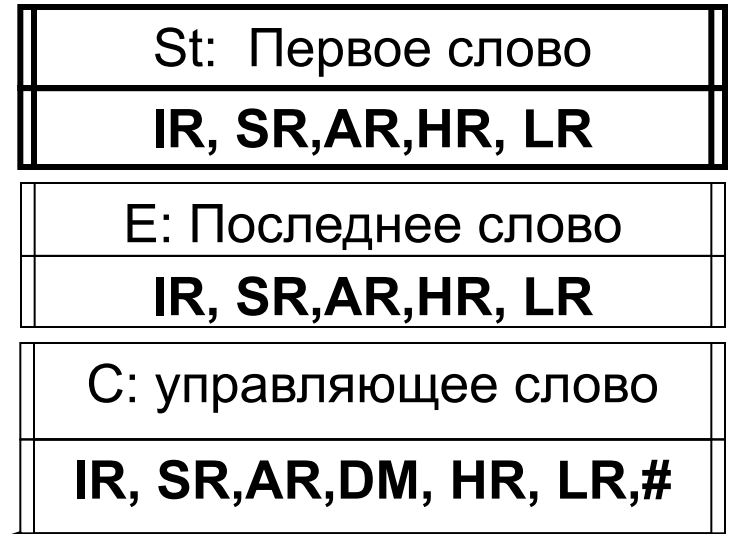
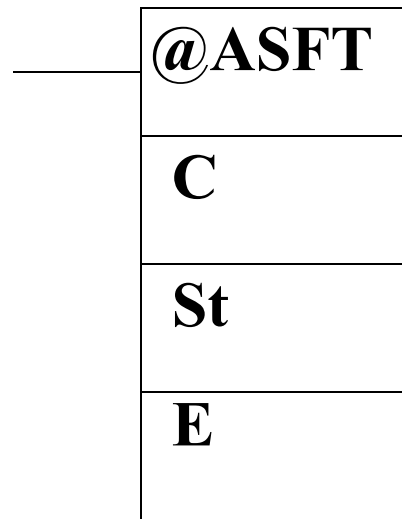
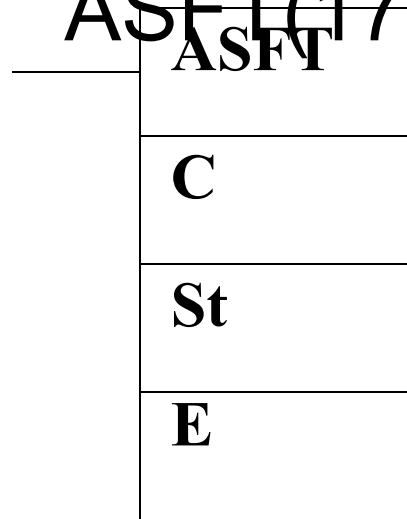
WSFT(16))



# Команды сдвига

Асинхронный регистр сдвига -

ASFT(17)



Биты 00....12	Не используются
Бит 13	Направление сдвига (1 - направление вниз, к младшим словам, 0 - к старшим словам).
Бит 14	Бит разрешения сдвига( 1 - разрешает регистру работать в соответствии с битом 13, 0 - запрещает).
Бит 15	Бит сброса: когда бит 15 = 1, регистр будет сброшен в 0 (между St и E). При нормальной работе бит 15 = 0.

# Команды сдвига

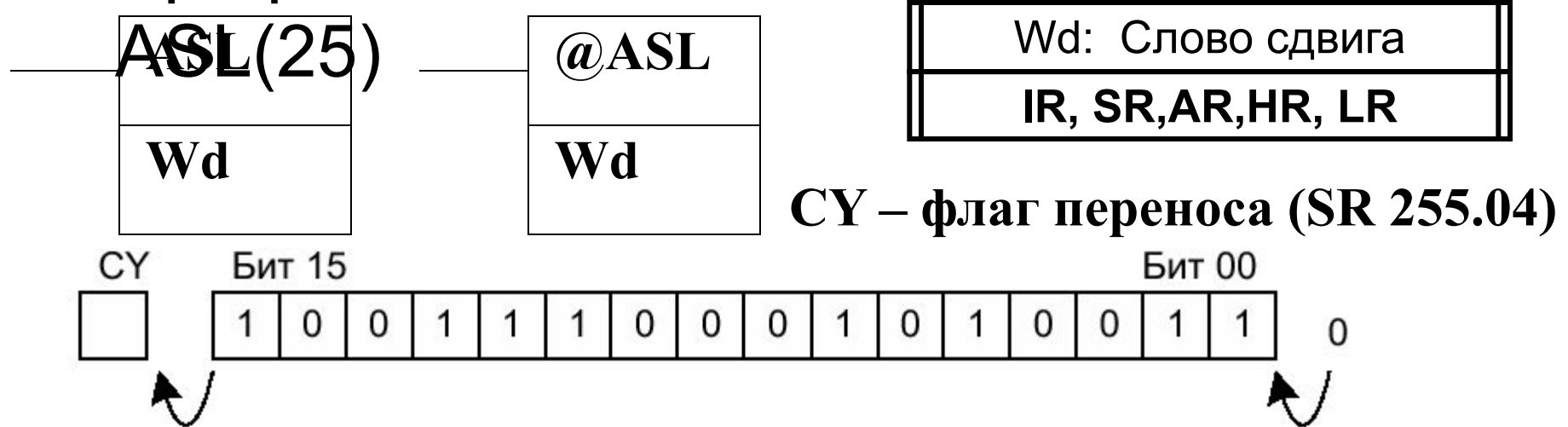
<b>ASFT</b>
<b>#6000</b>
<b>DM 0100</b>
<b>DM 0110</b>



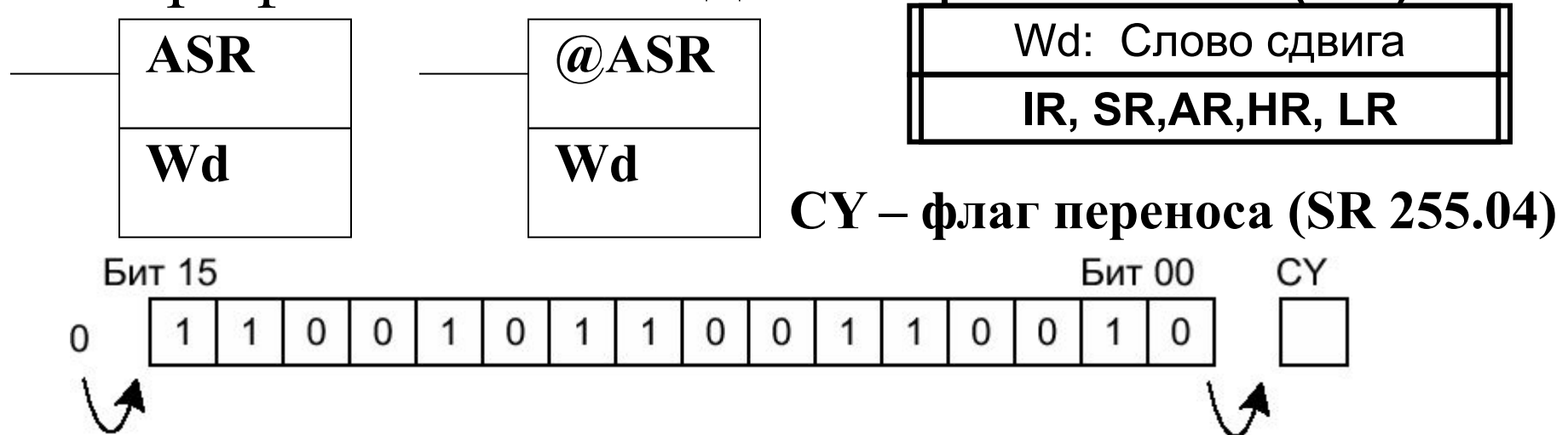


# Команды сдвига

Арифметический сдвиг влево -



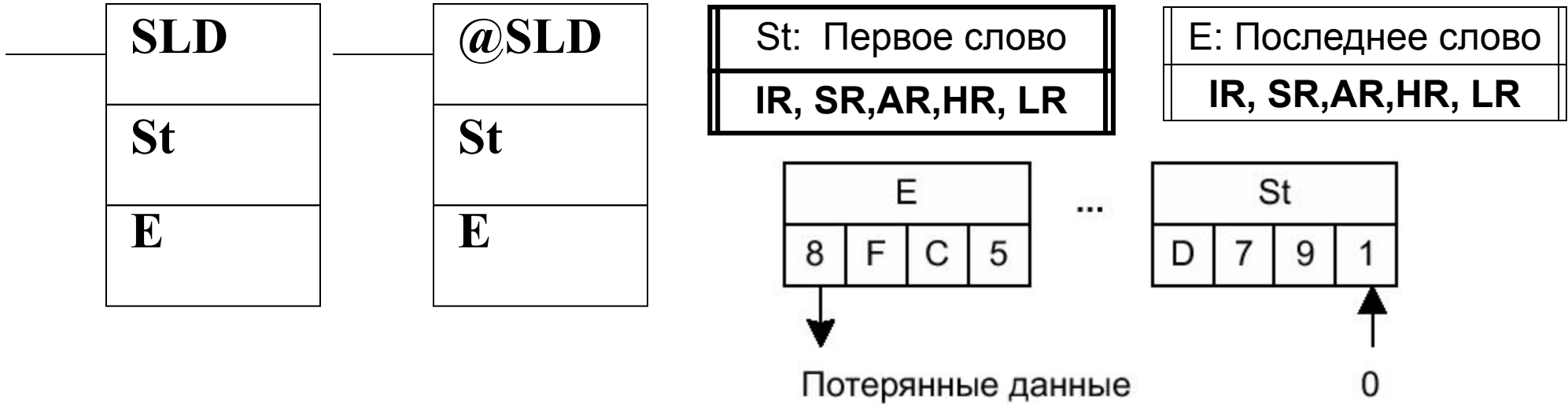
Арифметический сдвиг вправо - ASR(26)



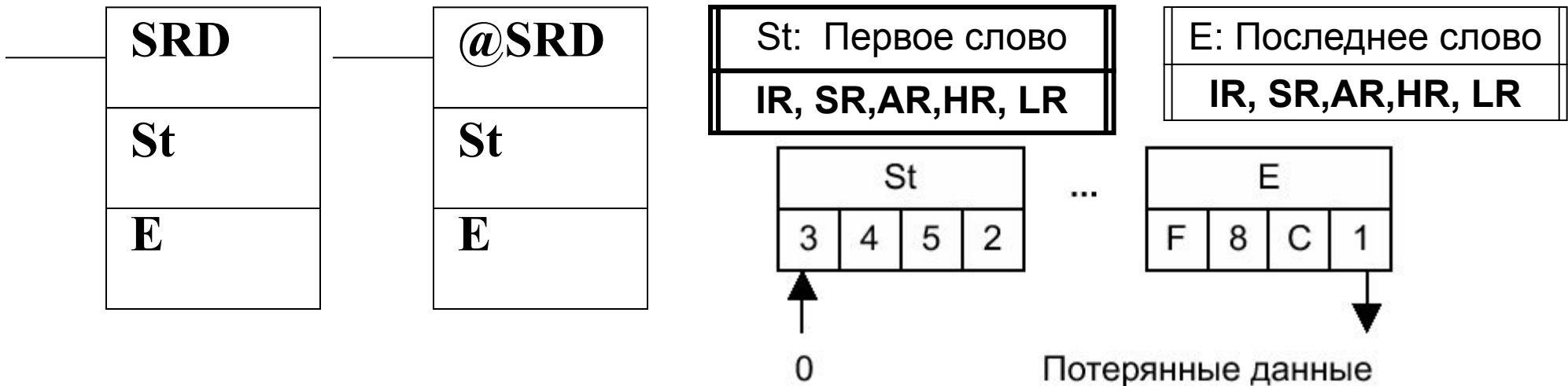


# Команды сдвига

## Сдвиг влево на одну цифру - SLD(74)



## Сдвиг вправо на одну цифру - SRD(75)



# Команды сдвига

Реверсивный регистр сдвига -

**SETR(84)**  
SFTR

C
St
E

@SFTR

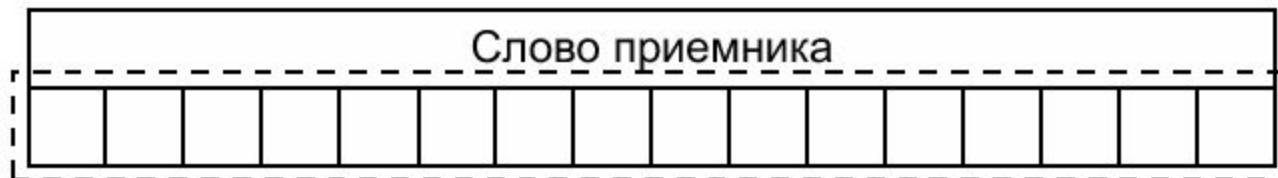
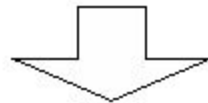
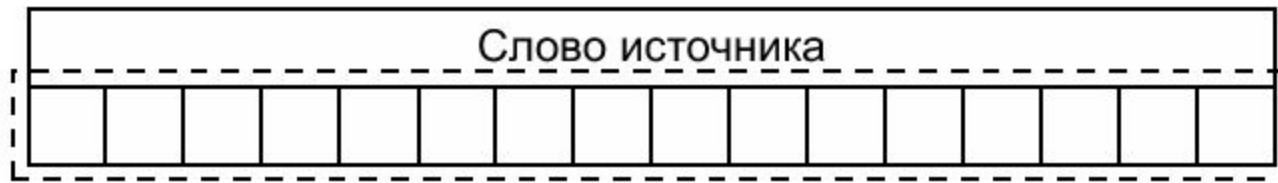
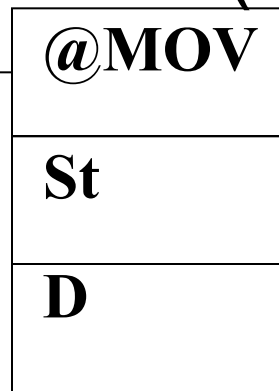
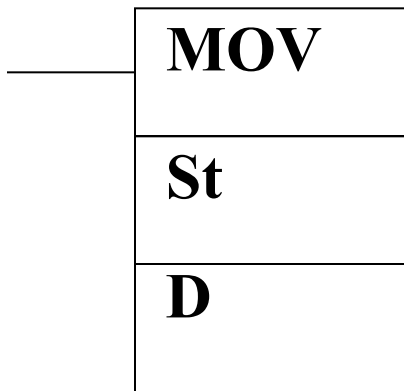
C
St
E

St: Первое слово
<b>IR, SR,AR,HR, LR</b>
E: Последнее слово
<b>IR, SR,AR,HR, LR</b>
C: управляющее слово
<b>IR, SR,AR,DM, HR, LR,#</b>

Бит 12	Направление сдвига (0 - к младшим битам, 1 - к старшим битам).
Бит 13	Состояние для ввода в регистр
Бит 14	Бит импульса сдвига
Бит 15	Бит сброса: когда бит 15 = 1, регистр будет сброшен в 0 (между St и E). При нормальной работе бит 15 = 0.

# Команды пересылки данных

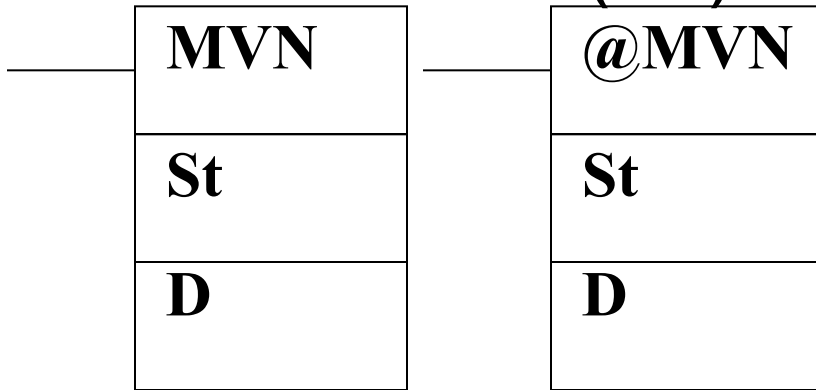
Пересылка -  
MOV(21)



Состояние битов не изменяется

# Команды пересылки данных

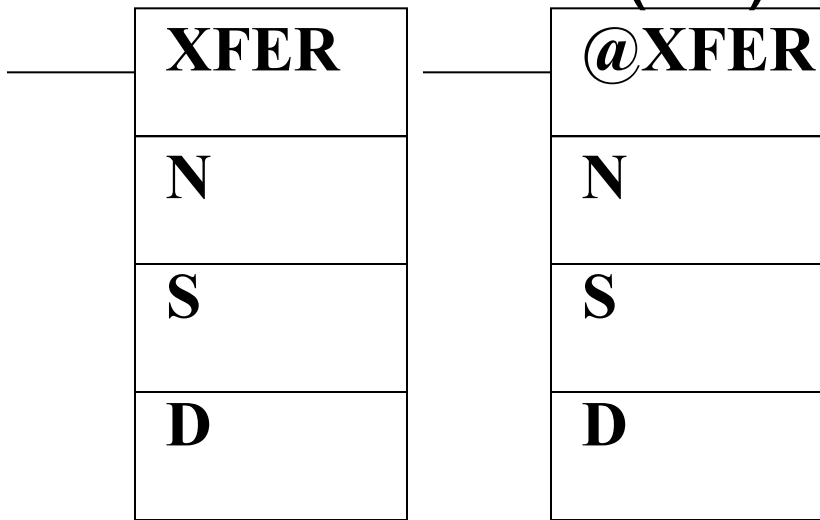
Пересылка инверсии -  
MVN(22)



Состояние битов инвертируется

# Команды пересылки данных

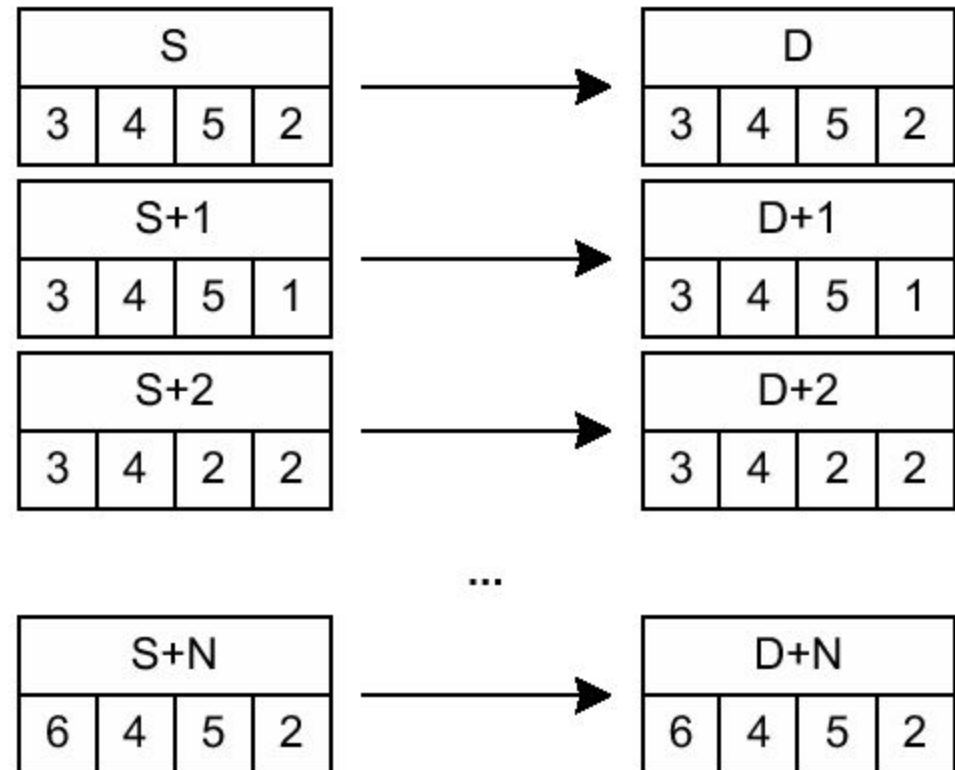
## Пересылка блока - XFER(70)



**N:** Количество слов (BCD)  
**IR, SR, AR, DM, HR, TC, LR, #**

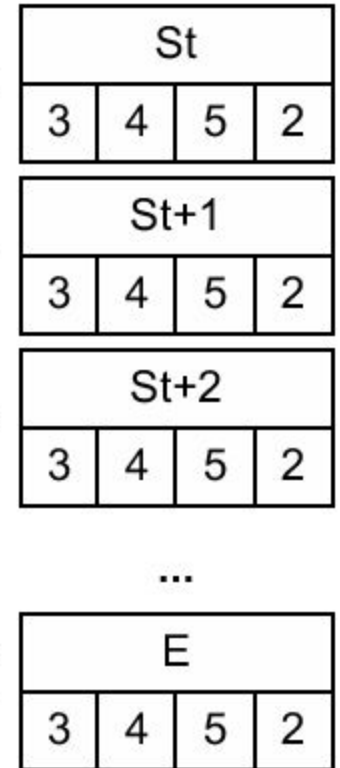
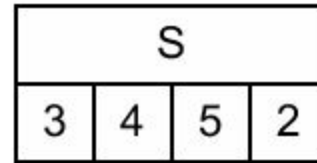
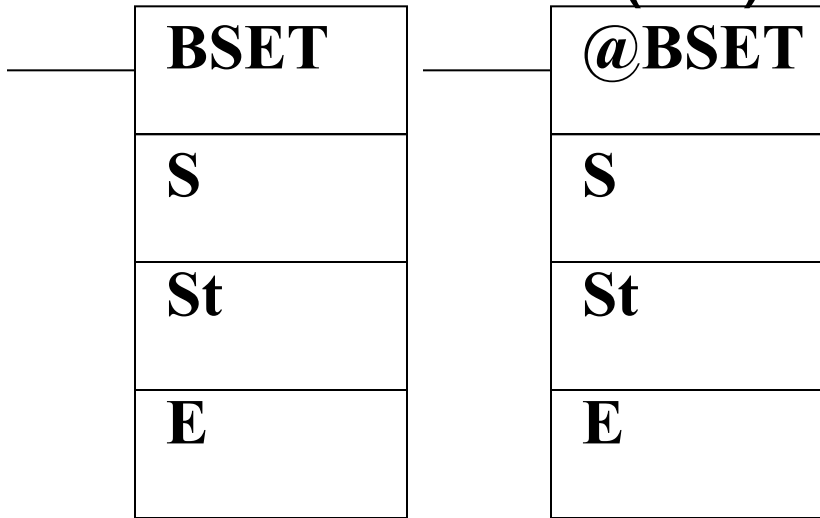
**S:** Первое слово источника  
**IR, SR, AR, DM, HR, TC, LR**

**D:** Первое слово приемника  
**IR, SR, AR, DM, HR, TC, LR**



# Команды пересылки данных

## Заполнение блока - BSET(71)



<b>S: Данные источника</b>
<b>IR, SR,AR,DM, HR,TC, LR,#</b>

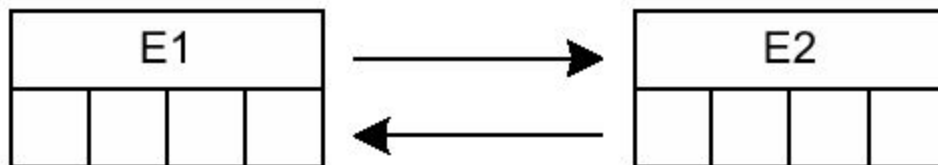
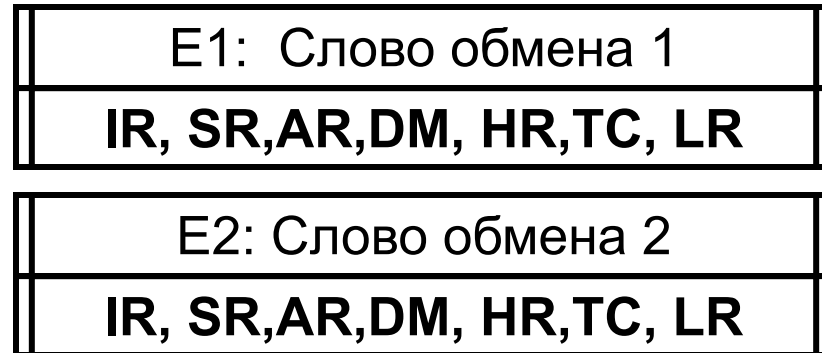
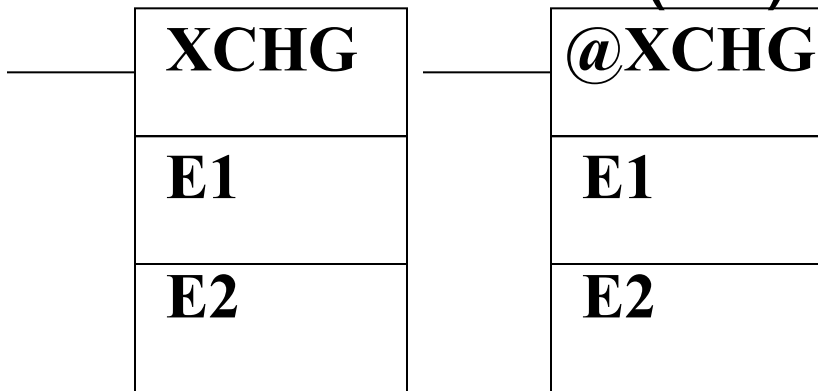
<b>S: Первое слово</b>
<b>IR, SR,AR,DM, HR,TC, LR</b>

<b>E: Последнее слово</b>
<b>IR, SR,AR,DM, HR,TC, LR</b>



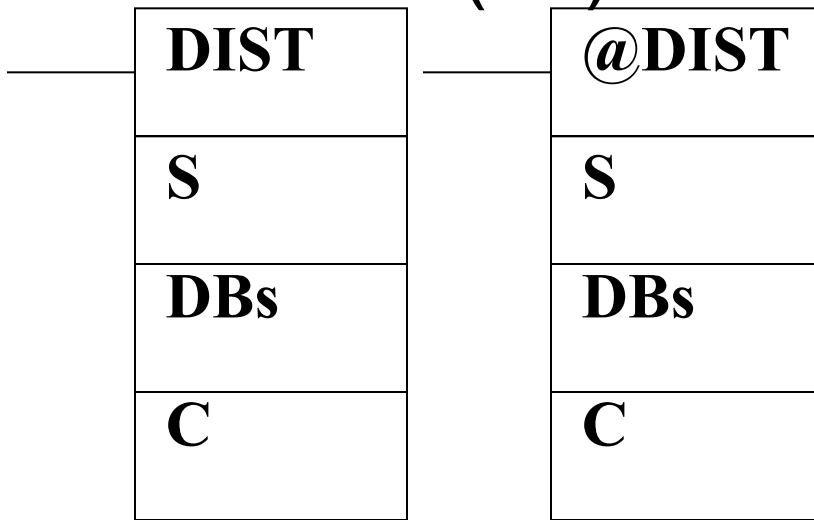
# Команды пересылки данных

Обмен данных -  
XCHG(73)



# Команды пересылки данных

## Распределение одного слова - DIST(80)



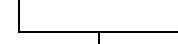
S: Данные источника
<b>IR, SR,AR,DM, HR,TC, LR,#</b>

DBs: Базовое слово приемника
<b>IR, SR,AR,DM, HR,TC, LR</b>

C: Слово управления (BCD)
<b>IR, SR,AR,DM, HR,TC, LR,#</b>

### Распределение одного слова:

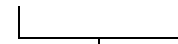
C - # 0 x x x



Смещение от базового слова приемника

### Операция со стеком:

C - # 9 x x x



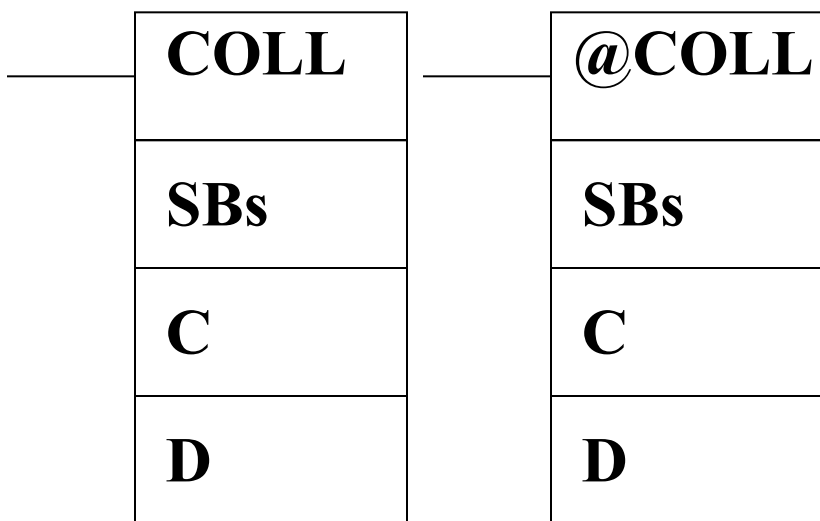
Размер стека

DBs – указатель стека

(перед использованием должен быть обнулен)

# Команды пересылки данных

## Сбор данных - COLL(81)



**Сбор данных :**

C - # 0 x x x

└──┬──┘

→ Смещение от базового слова источника

**Операция со стеком FIFO:**

C - # 9 x x x

└──┬──┘

→ Размер стека

**Операция со стеком LIFO:**

C - # 8 x x x

└──┬──┘

→ Размер стека

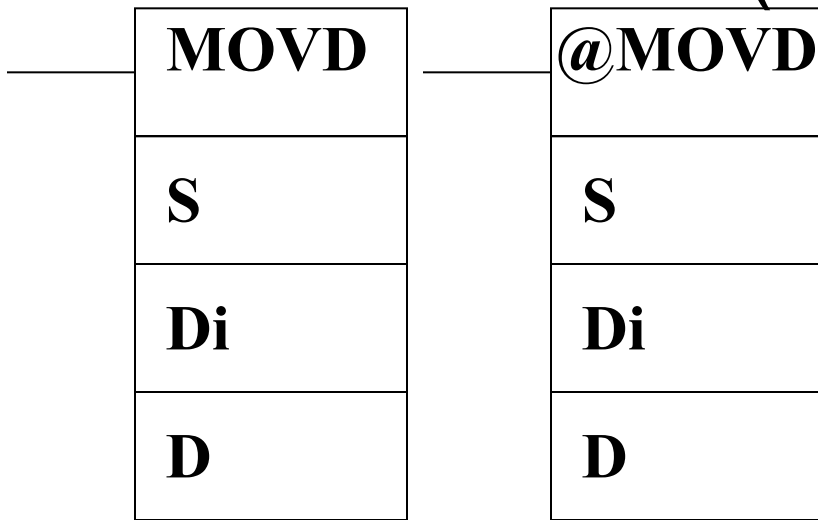
SBs – указатель стека

(перед использованием должен быть обнулен)

SBs: Базовое слово источника
<b>IR, SR,AR,DM, HR,TC, LR, #</b>
C: Слово управления (BCD)
<b>IR, SR,AR,DM, HR,TC, LR,#</b>
D: Слово приемника
<b>IR, SR,AR,DM, HR,TC, LR</b>

# Команды пересылки данных

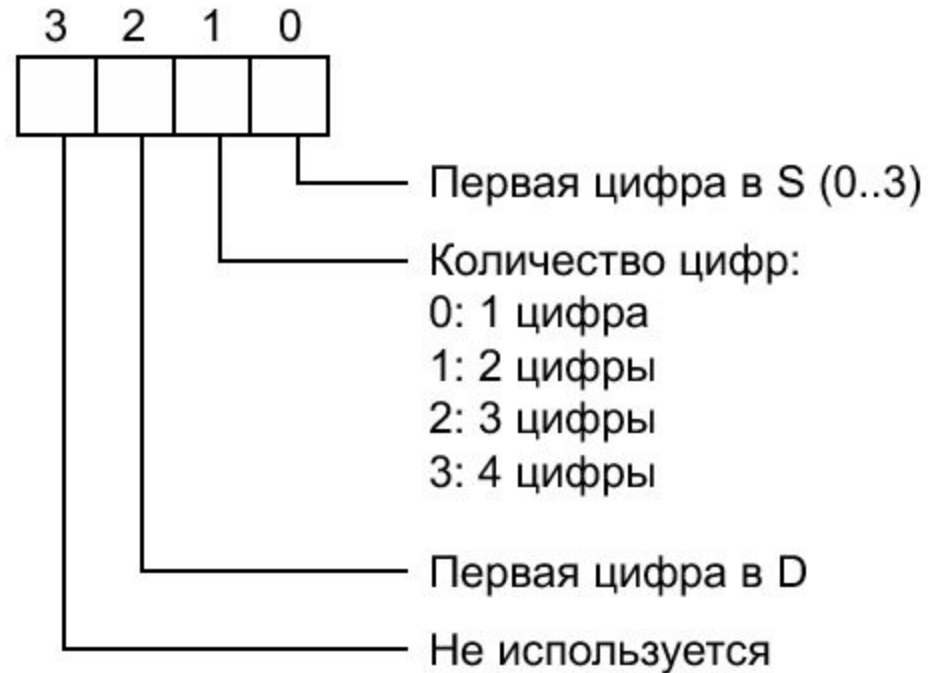
Переслать цифру -  
MOVD(83)



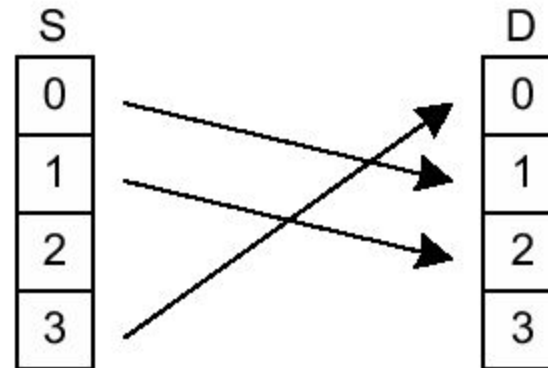
S: Слово источника
<b>IR, SR,AR,DM, HR, LR, #</b>

Di: Определитель слова (BCD)
<b>IR, SR,AR,DM, HR,TC, LR,#</b>

D: Слово приемник
<b>IR, SR,AR,DM, HR,TC, LR</b>



Di: 0023



# Команды сравнения

## Сравнение - CMP(20)

## Сравнение слов двойной длины -

## CMP(60)

<b>CMP</b>
<b>Ср1</b>
<b>Ср2</b>

Ср1: Первое слово сравнения
<b>IR, SR,DM, HR, TC,LR,#</b>
Ср2: Второе слово сравнения
<b>IR, SR,DM, HR, TC,LR,#</b>

<b>CMPL</b>
<b>Ср1</b>
<b>Ср2</b>

Флаг	Адрес	C1<C2	C1=C2	C1>C2
<b>GR</b>	<b>25505</b>	<b>0</b>	<b>0</b>	<b>1</b>
<b>EQ</b>	<b>25506</b>	<b>0</b>	<b>1</b>	<b>0</b>
<b>LE</b>	<b>25507</b>	<b>1</b>	<b>0</b>	<b>0</b>

# Команды сравнения

Сравнение с 16 значениями таблицы - ТСМР(85)

<b>ТСМР</b>	<b>@ТСМР</b>
<b>CD</b>	<b>CD</b>
<b>ТВ</b>	<b>ТВ</b>
<b>R</b>	<b>R</b>

**CD – #210, ТВ – DM0000, R - 216**

<b>CD: Данные сравнения</b>
<b>IR, SR,DM, HR, TC,LR,#</b>
<b>ТВ: Первое слово таблицы сравнения</b>
<b>IR, SR,DM, HR, TC,LR</b>
<b>R: Слово результат</b>
<b>IR, SR,AR,DM, HR, TC,LR</b>

DM 0000	0100		IR 21600	0
DM 0001	0200		IR 21601	0
DM 0002	0210	→	IR 21602	1
DM 0003	0400		IR 21603	0
DM 0004	0500		IR 21604	0
DM 0005	0600		IR 21605	0
DM 0006	0210	→	IR 21606	1
DM 0007	0800		IR 21607	0
DM 0008	0900		IR 21608	0
DM 0009	1000		IR 21609	0
DM 0010	0210	→	IR 21610	1
DM 0011	1200		IR 21611	0
DM 0012	1300		IR 21612	0
DM 0013	1400		IR 21613	0
DM 0014	0210	→	IR 21614	1
DM 0015	1600		IR 21615	0

# Команды сравнения

## Сравнение блока -

<b>BCMP</b>
<b>CD</b>
<b>CB</b>
<b>R</b>

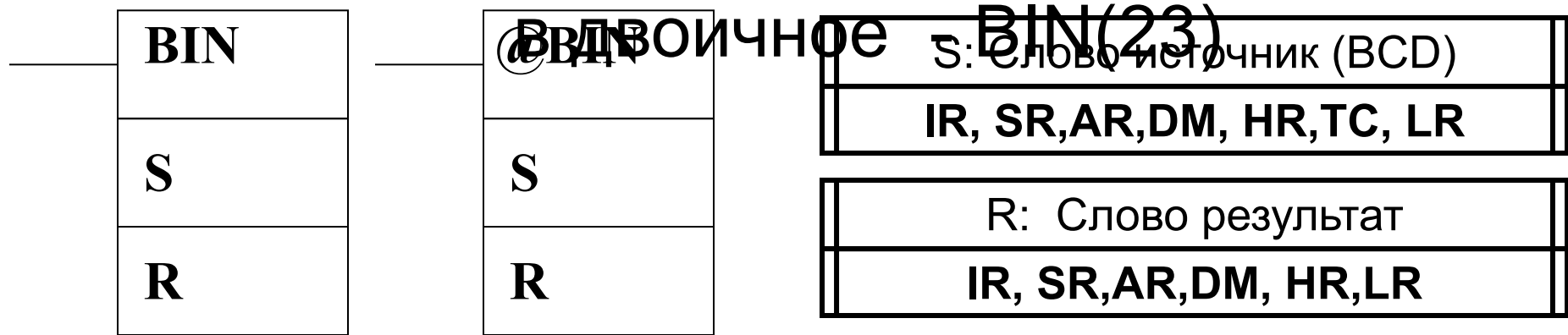
<b>@BCMP</b>
<b>CD</b>
<b>CB</b>
<b>R</b>

<b>BCMP(68):</b> Данные сравнения
<b>IR, SR,AR,DM, HR, TC,LR,#</b>
CB: Первое слово блока сравнения
<b>IR, SR,DM, HR, TC,LR</b>
R: Слово результат
<b>IR, SR,AR,DM, HR, TC,LR</b>

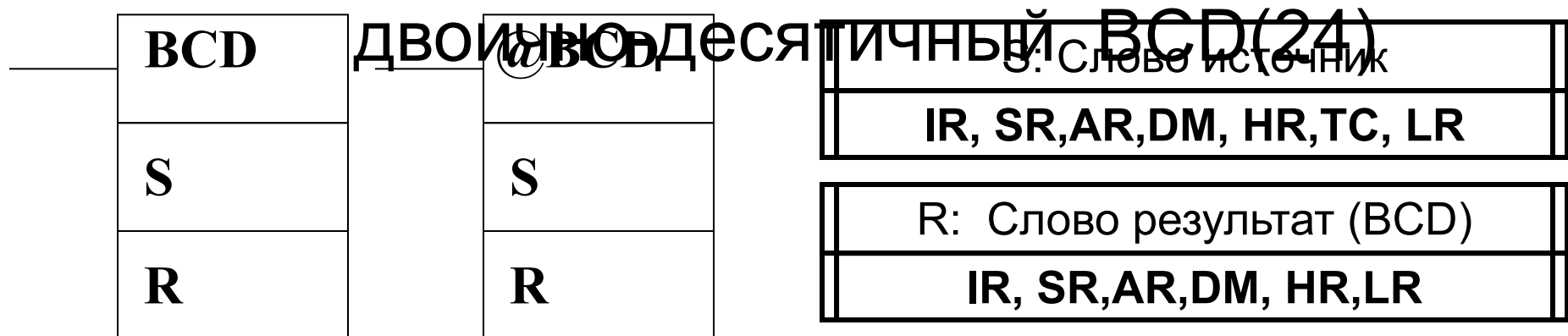
$CB \leq CD \leq CB+1$	Бит 00
$CB+2 \leq CD \leq CB+3$	Бит 01
$CB+4 \leq CD \leq CB+5$	Бит 02
$CB+6 \leq CD \leq CB+7$	Бит 03

# Команды преобразования

## Преобразование двоично-десятичного числа



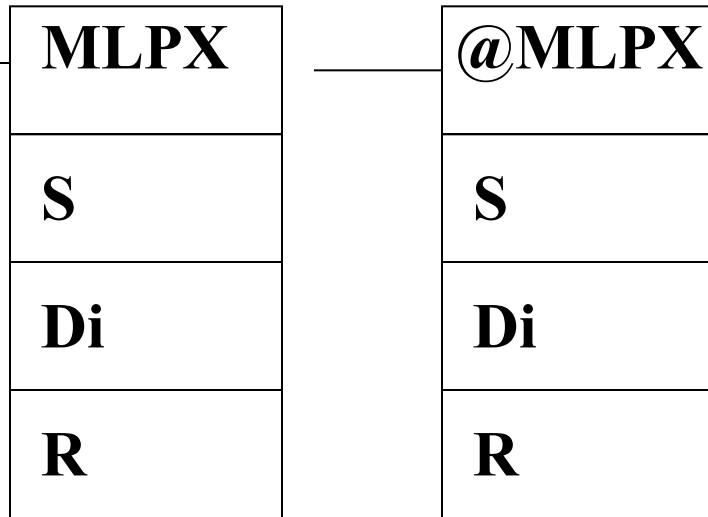
## Преобразование из двоичного вида в



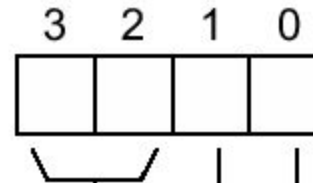
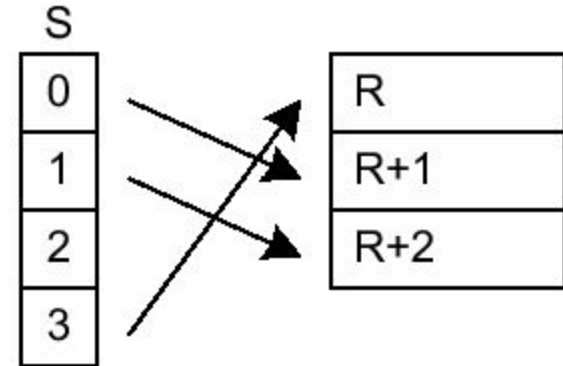


# Команды преобразования

## Преобразовать 4-в-16 MLPX(76)



**Di: 0023**



Задаёт первую цифру, подлежащую преобразованию (0..3)

Количество цифр, подлежащих преобразованию (0..3)

0: 1 цифра

1: 2 цифры

2: 3 цифры

3: 4 цифры

Не используется (установите в 0)

S: Слово источник

**IR, SR,AR,DM, HR, LR,TC**

Di: Определитель цифры

**IR, SR,AR,DM, HR, LR ,TC,#**

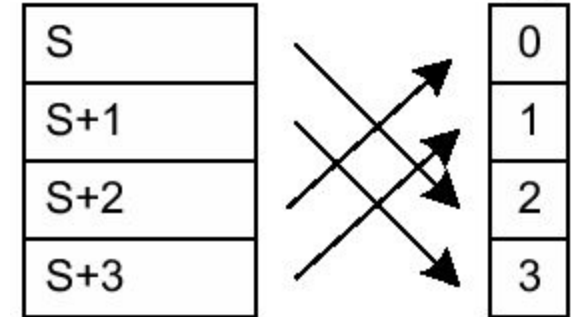
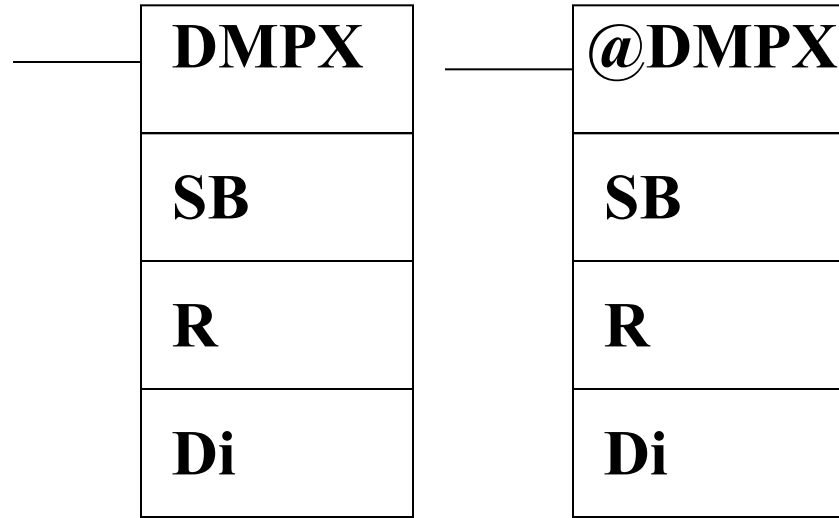
R: Первое слово результата

**IR, SR,AR,DM, HR, LR**

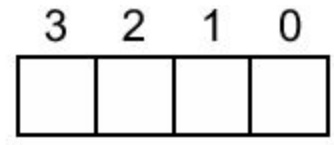
# Команды преобразования

Преобразовать 16-в-4 DMPX(77)

Di: 0023



SB: Первое слово источника
<b>IR, SR,AR,DM, HR, LR,TC</b>
DI: Определитель цифры
<b>IR, SR,AR,DM, HR, LR ,TC,#</b>
R: Первое слово результата
<b>IR, SR,AR,DM, HR, LR</b>



Задаёт первую цифру приемника преобразованных данных (0..3)

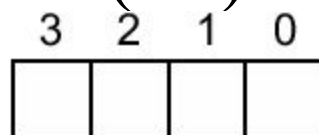
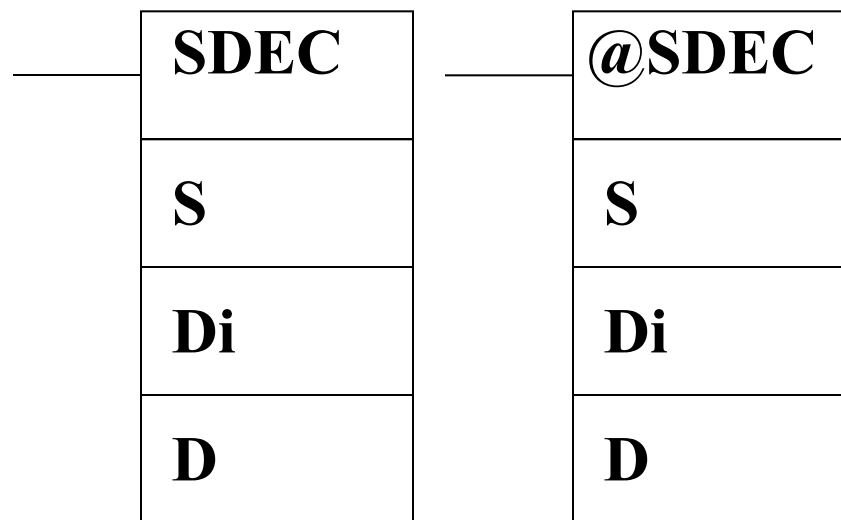
Количество слов, подлежащих преобразованию (0..3)  
 0: 1 цифра  
 1: 2 цифры  
 2: 3 цифры  
 3: 4 цифры

Не используется

# Команды преобразования

Преобразование в коды 7-сегментного индикатора -

## SDEC (78)



3 — Задает первую цифру в S, подлежащую преобразованию (0..3)

2 — Количество преобразуемых цифр (0..3)

0: 1 цифра

1: 2 цифры

2: 3 цифры

3: 3 цифры

1 — Первая половина в D

0: младшие 8 бит (1-я половина)

1: старшие 8 бит (2-я половина)

3 — Не используется (установлена в 0)

S: Слово источника

IR, SR,AR,DM, HR, LR,TC

Di: Определитель цифры

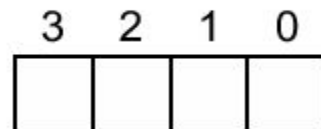
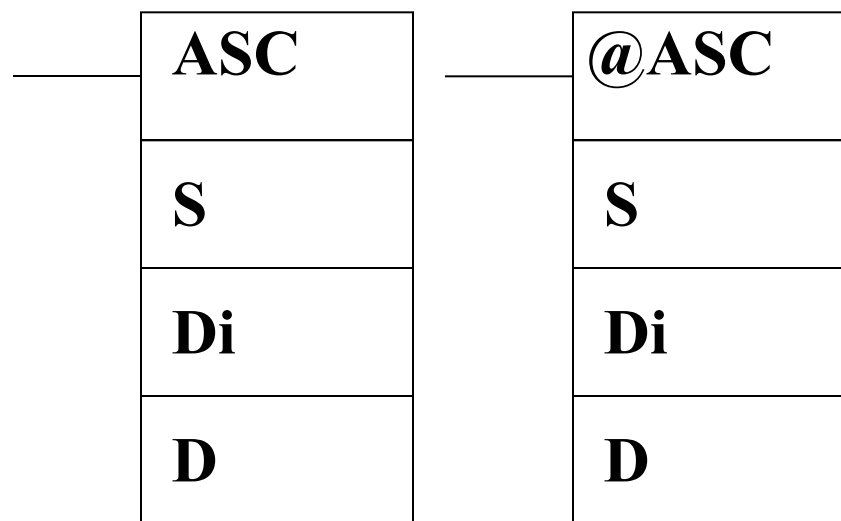
IR, SR,AR,DM, HR, LR ,TC,#

D: Первое слово приемника

IR, SR,AR,DM, HR, LR

# Команды преобразования

## Преобразование в коды ASCII - ASC(86)



3 —————  
2 —————  
1 —————  
0 —————

Задаёт первую цифру в S,  
подлежащую преобразованию  
(0..3)

Количество преобразуемых цифр  
(0..3)

0: 1 цифра  
1: 2 цифры  
2: 3 цифры  
3: 3 цифры

Первая половина в D  
0: младшие 8 бит (1-я половина)  
1: старшие 8 бит (2-я половина)

Контроль на четность  
0: нет  
1: четн.  
2: нечетн.

S: Слово источника

IR, SR,AR,DM, HR, LR,TC

Di: Определитель цифры

IR, SR,AR,DM, HR, LR ,TC,#

D: Первое слово приемника

IR, SR,AR,DM, HR, LR

# Команды преобразования

## Счетчик битов – BCNT (667)

<b>BCNT</b>	<b>@BCNT</b>
<b>N</b>	<b>N</b>
<b>SB</b>	<b>SB</b>
<b>R</b>	<b>R</b>

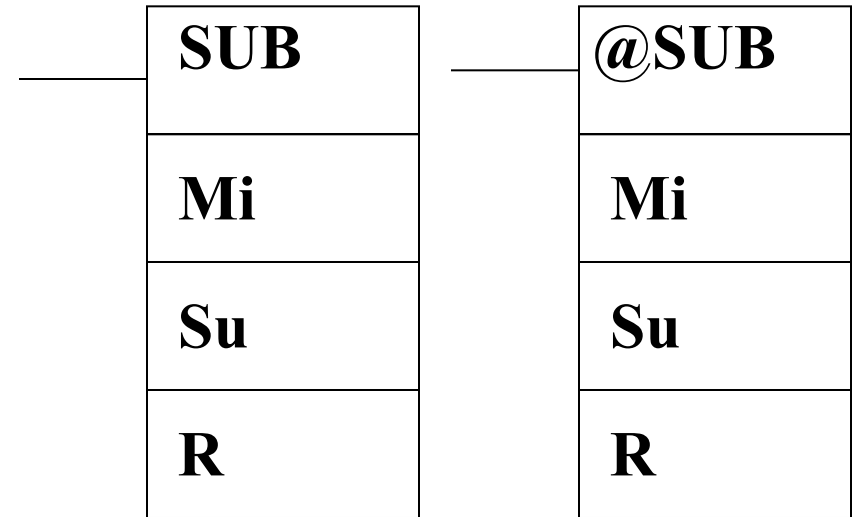
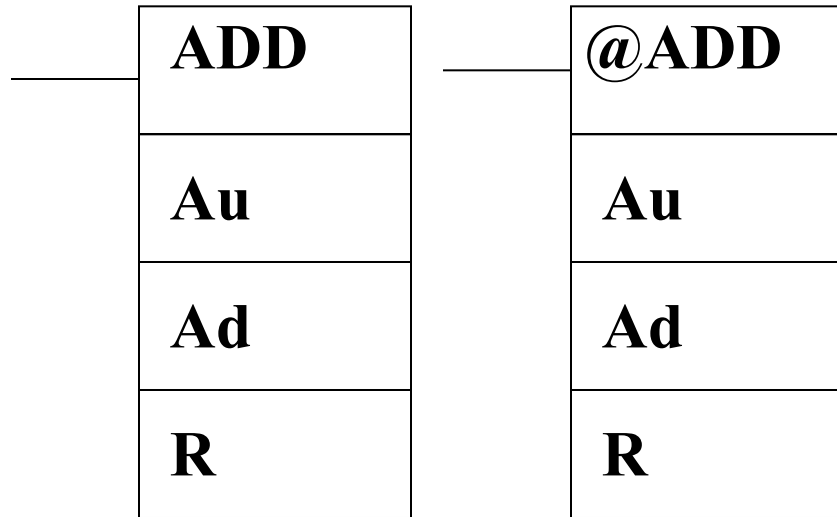
<b>N: Кол- во слов (BCD)</b>
<b>IR, SR,AR,DM, HR, LR,TC,#</b>
<b>SB: Первое слово источника</b>
<b>IR, SR,AR,DM, HR, LR ,TC</b>
<b>R: Слово приемник</b>
<b>IR, SR,AR,DM, HR, LR,TC</b>

# Математические операции

- команды двоично – десятичных вычислений
- команды двоичной арифметики

# Команды двоично – десятичных вычислений

Двоично-десятичное сложение ADD(30) и  
вычитание SUB(31)



**Au:** Первое слагаемое

**R, SR,AR,DM, HR, LR,TC,#**

**Ad:** Второе слагаемое

**IR, SR,AR,DM, HR, LR ,TC,#**

**R:** Слово результата

**IR, SR,AR,DM, HR, LR**

**MI:** Уменьшаемое

**R, SR,AR,DM, HR, LR,TC,#**

**SU:** вычитаемое

**IR, SR,AR,DM, HR, LR ,TC,#**

**R:** Слово результата

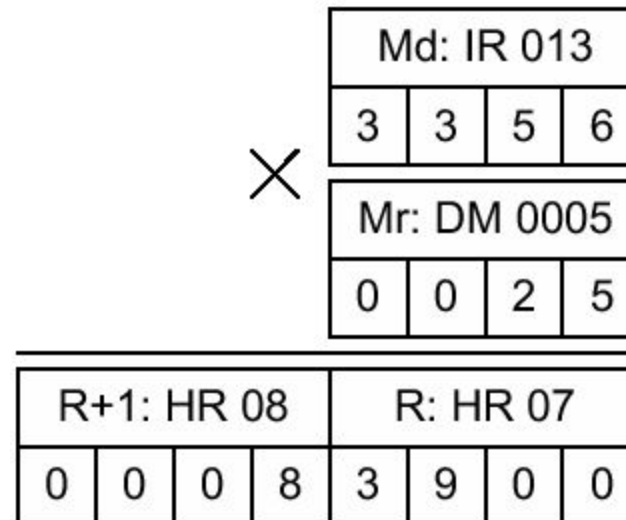
**IR, SR,AR,DM, HR, LR**

# Команды двоично – десятичных вычислений

## Двоично-десятичное умножение - MUL(32)

MUL	@MUL
Md	Au
Mr	Ad
R	R

MD: Множитель
R, SR, AR, DM, HR, LR, TC, #
MR: МНОЖИМОЕ
IR, SR, AR, DM, HR, LR, TC, #
R: Первое слово
IR, SR, AR, DM, HR, LR





# Команды двоично – десятичных вычислений

## Деление двоично-десятичных чисел DIV(33)

<b>DIV</b>	<b>@DIV</b>
<b>Dd</b>	<b>Dd</b>
<b>Dr</b>	<b>Dr</b>
<b>R</b>	<b>R</b>

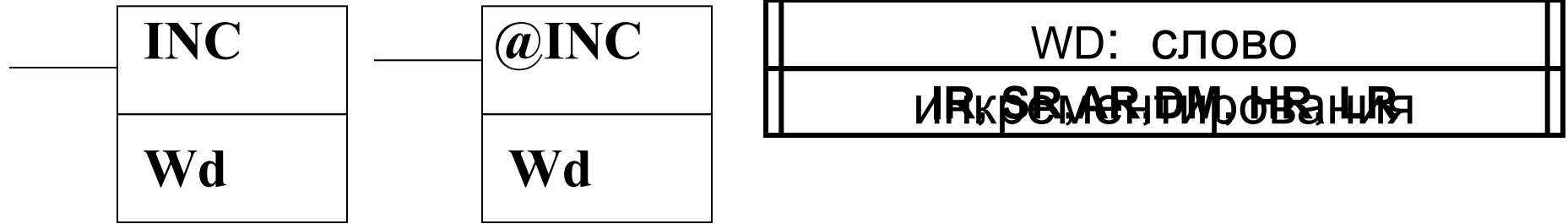
DD: делимое
R, SR, AR, DM, HR, LR, TC, #
DR: делитель
IR, SR, AR, DM, HR, LR, TC, #
R: Первое слово
IR, SR, AR, DM, HR, LR

Dd	Dr
R+1	R
Остаток	Частное

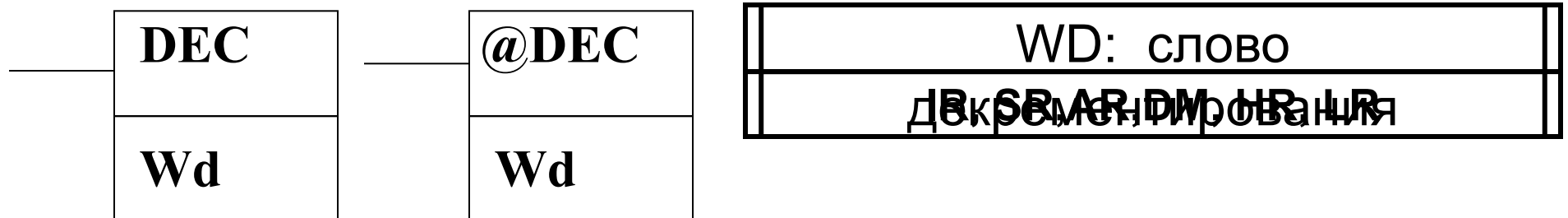
Dd: IR 216				Dr: HR 09			
3	4	5	2	0	0	0	3
R+1:DM 0018				R: DM 0017			
0	0	0	2	1	1	5	0

# Команды двоично – десятичных вычислений

## Инкремент двоично-десятичного числа - INC(38)



## Декремент двоично-десятичного числа - DEC(39)



# Команды двоичных вычислений

Двоичное сложение ADB(50) и вычитание SBB(51)

ADB	@ADB
Au	Au
Ad	Ad
R	R

SBB	@SBB
Mi	Mi
Su	Su
R	R

Au: Первое слагаемое

R, SR,AR,DM, HR, LR,TC,#

Ad: Второе слагаемое

IR, SR,AR,DM, HR, LR ,TC,#

R: Слово результата

IR, SR,AR,DM, HR, LR

MI: Уменьшаемое

R, SR,AR,DM, HR, LR,TC,#

SU: вычитаемое

IR, SR,AR,DM, HR, LR ,TC,#

R: Слово результата

IR, SR,AR,DM, HR, LR

# Команды двоичных вычислений

## Двоичное умножение MLB(52)

<b>MLB</b>	<b>@MLB</b>
<b>Md</b>	<b>Au</b>
<b>Mr</b>	<b>Ad</b>
<b>R</b>	<b>R</b>

<b>MD: Множитель</b>
<b>R, SR,AR,DM, HR, LR,TC,#</b>
<b>MR: МНОЖИМОЕ</b>
<b>IR, SR,AR,DM, HR, LR ,TC,#</b>
<b>R: Первое слово</b>
<b>IR, SR,AR,DM, HR, LR</b>



# Команды двоичных вычислений

## Двоичное деление DVB(53)

<b>DVB</b>	<b>@DVB</b>
<b>Dd</b>	<b>Dd</b>
<b>Dr</b>	<b>Dr</b>
<b>R</b>	<b>R</b>

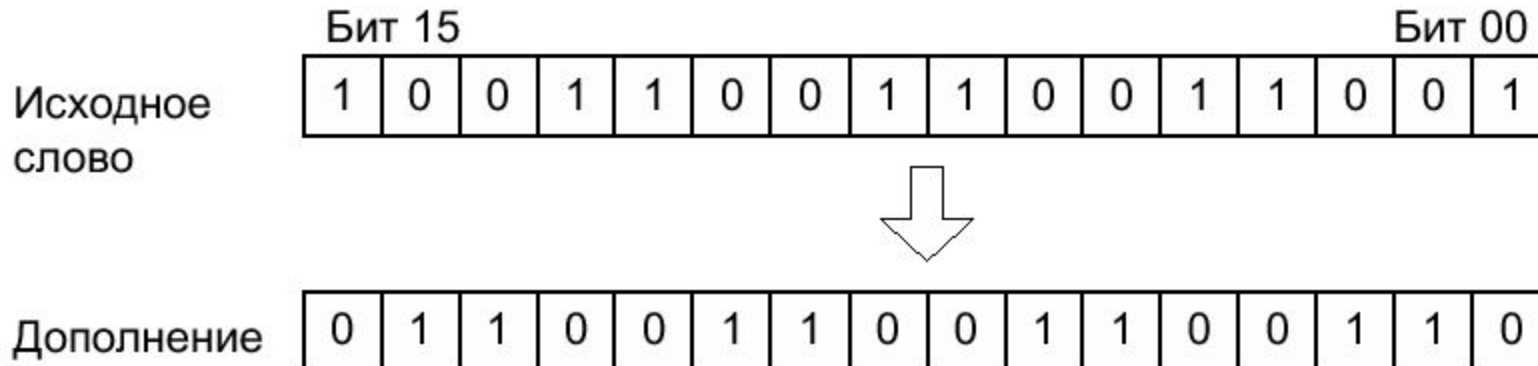
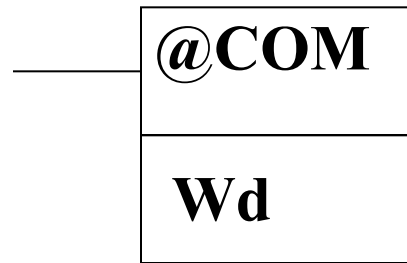
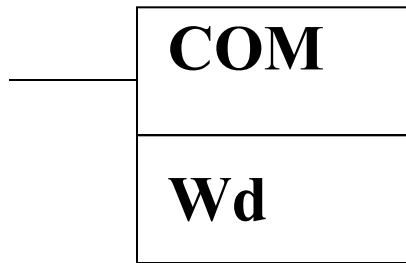
DD: делимое
R, SR, AR, DM, HR, LR, TC, #
DR: делитель
IR, SR, AR, DM, HR, LR, TC, #
R: Первое слово
IR, SR, AR, DM, HR, LR

Dd	Dr
R+1	R
Остаток	Частное

Dd: IR 216				Dr: HR 09			
3	4	5	2	0	0	0	3
R+1:DM 0018				R: DM 0017			
0	0	0	2	1	1	5	0

# Логические команды

## Дополнение - COM(29)



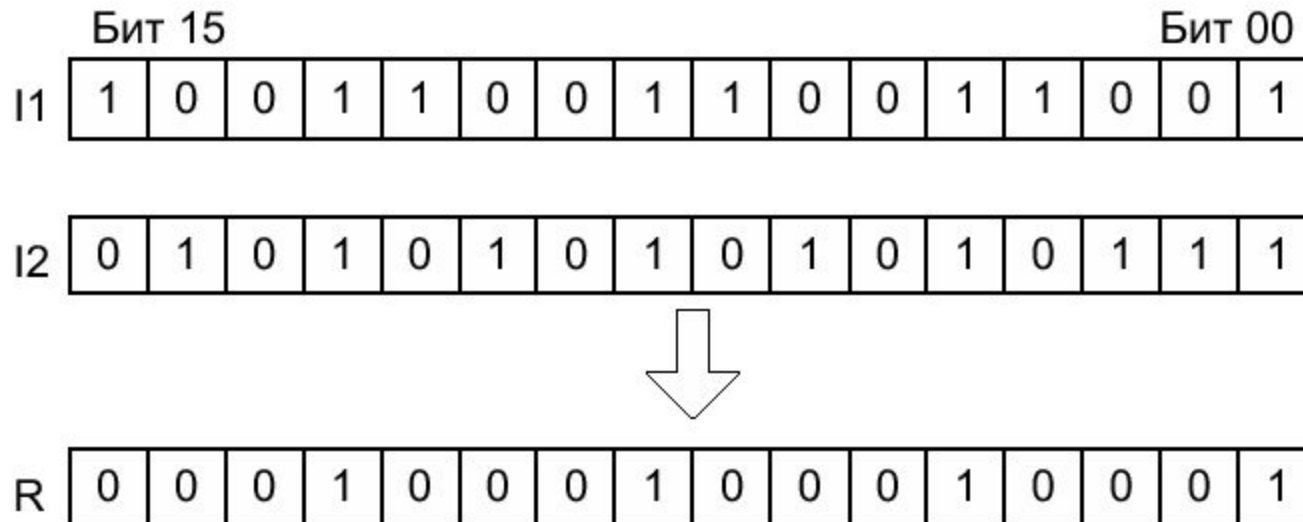
# Логические команды

## Логическое И - AND(34)

<b>AND</b>
<b>I1</b>
<b>I2</b>
<b>R</b>

<b>@AND</b>
<b>I1</b>
<b>I2</b>
<b>R</b>

<b>I1: Вход 1</b>
<b>IR, SR,AR,DM, HR, LR,TC,#</b>
<b>I2: Вход 2</b>
<b>IR, SR,AR,DM, HR, LR ,TC,#</b>
<b>R: Слово результат</b>
<b>IR, SR,AR,DM, HR, LR</b>



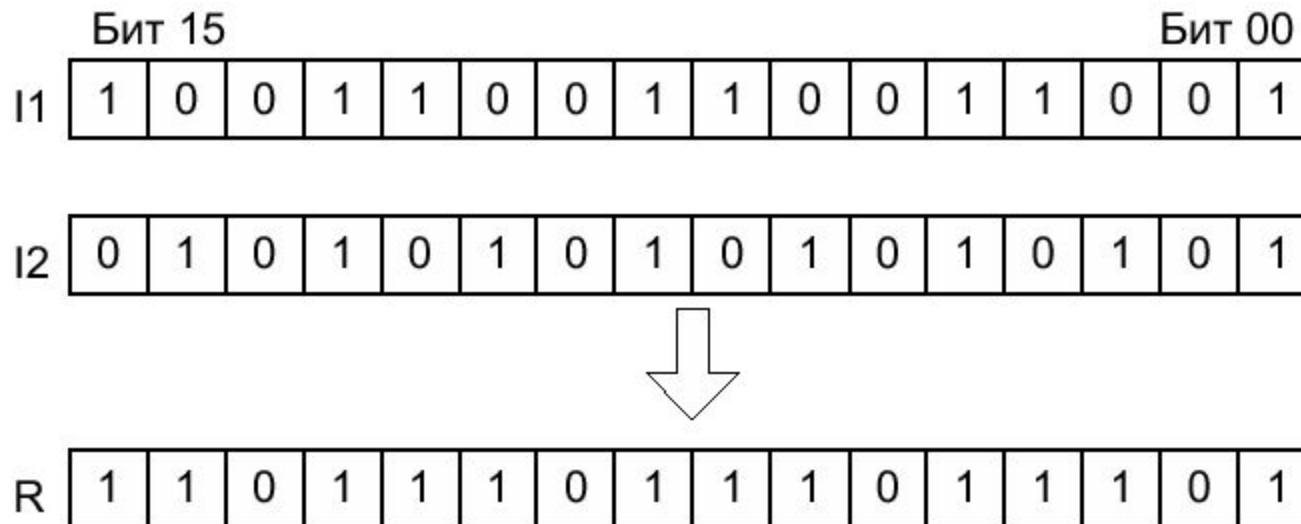
# Логические команды

## Логическое ИЛИ - OR(35)

<b>OR</b>
<b>I1</b>
<b>I2</b>
<b>R</b>

<b>@OR</b>
<b>I1</b>
<b>I2</b>
<b>R</b>

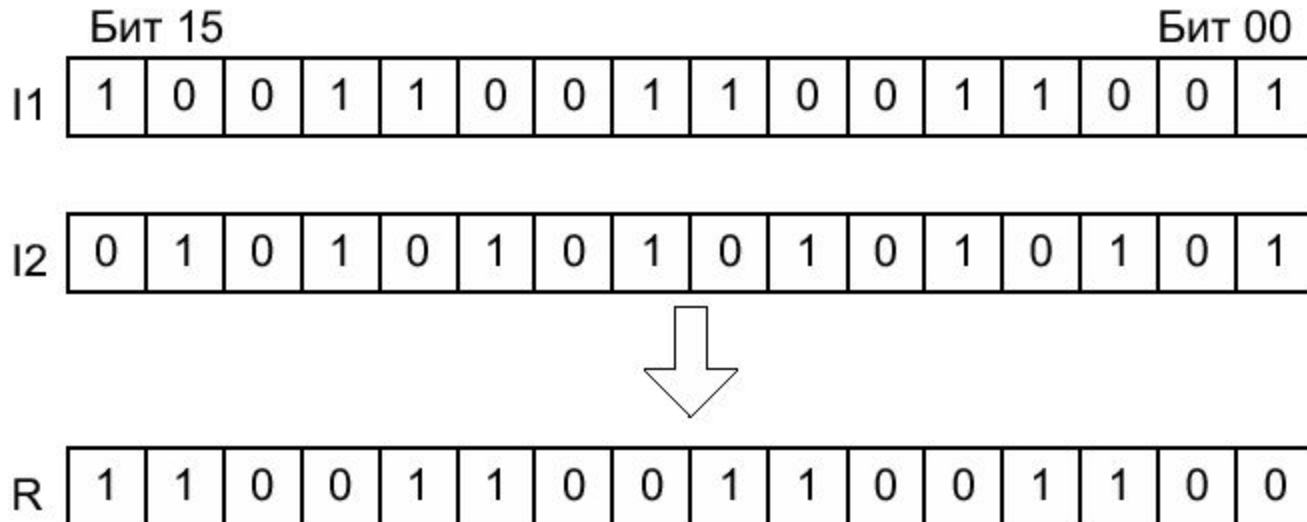
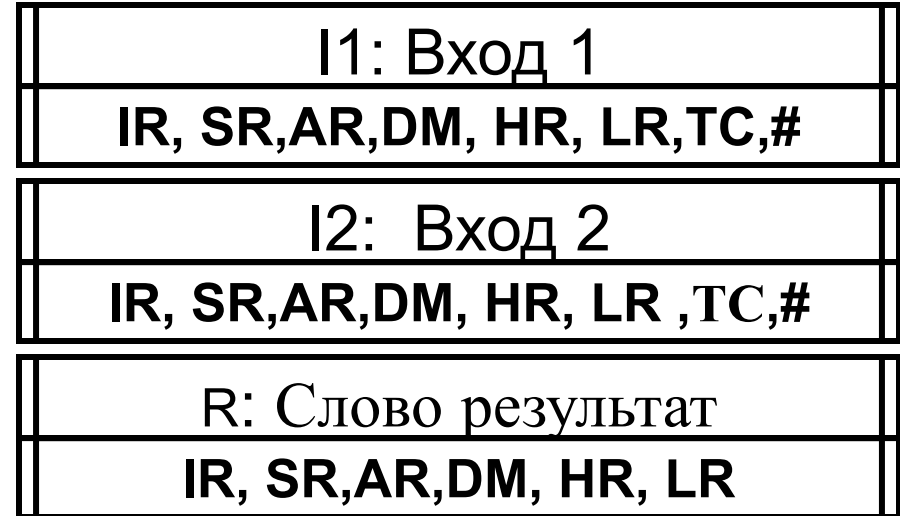
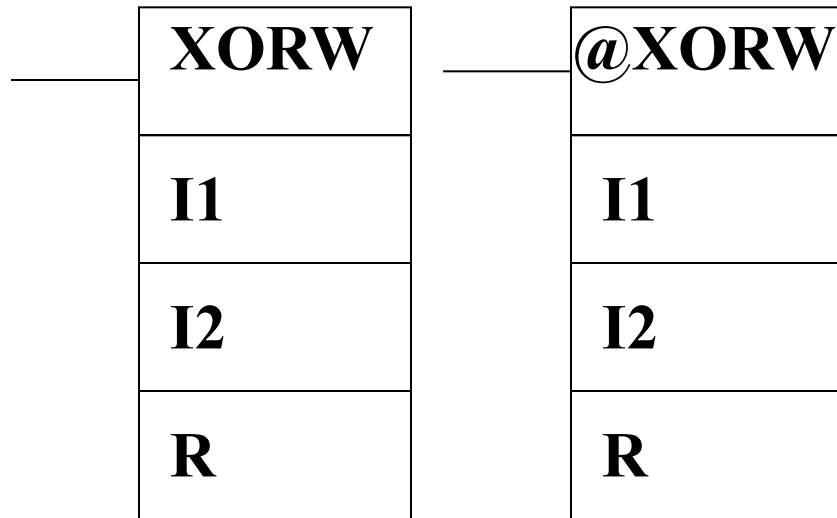
<b>I1: Вход 1</b>
<b>IR, SR,AR,DM, HR, LR,TC,#</b>
<b>I2: Вход 2</b>
<b>IR, SR,AR,DM, HR, LR ,TC,#</b>
<b>R: Слово результат</b>
<b>IR, SR,AR,DM, HR, LR</b>





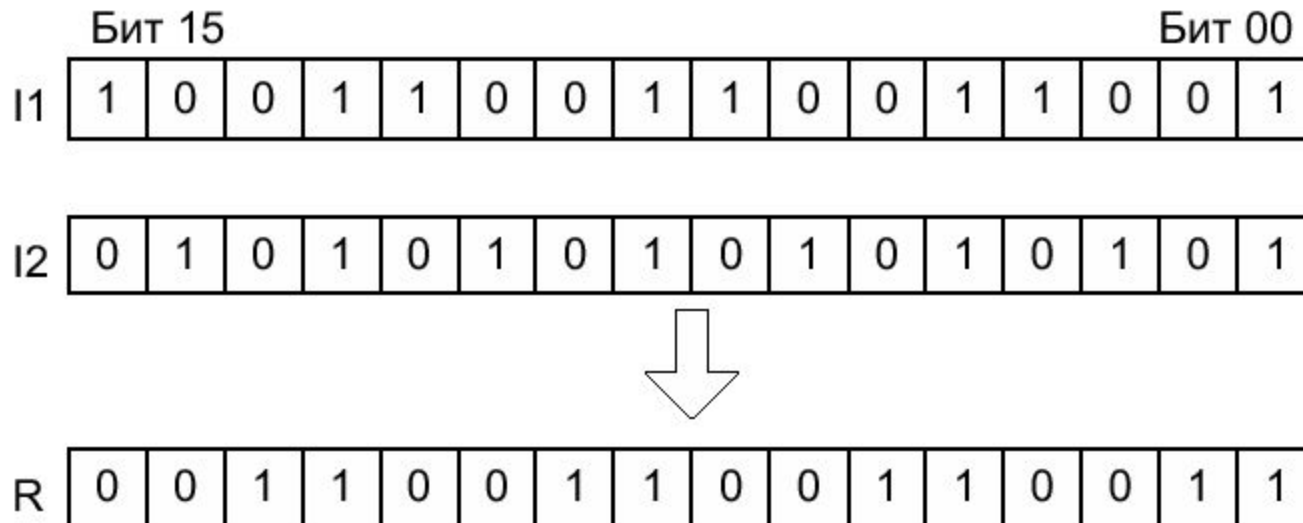
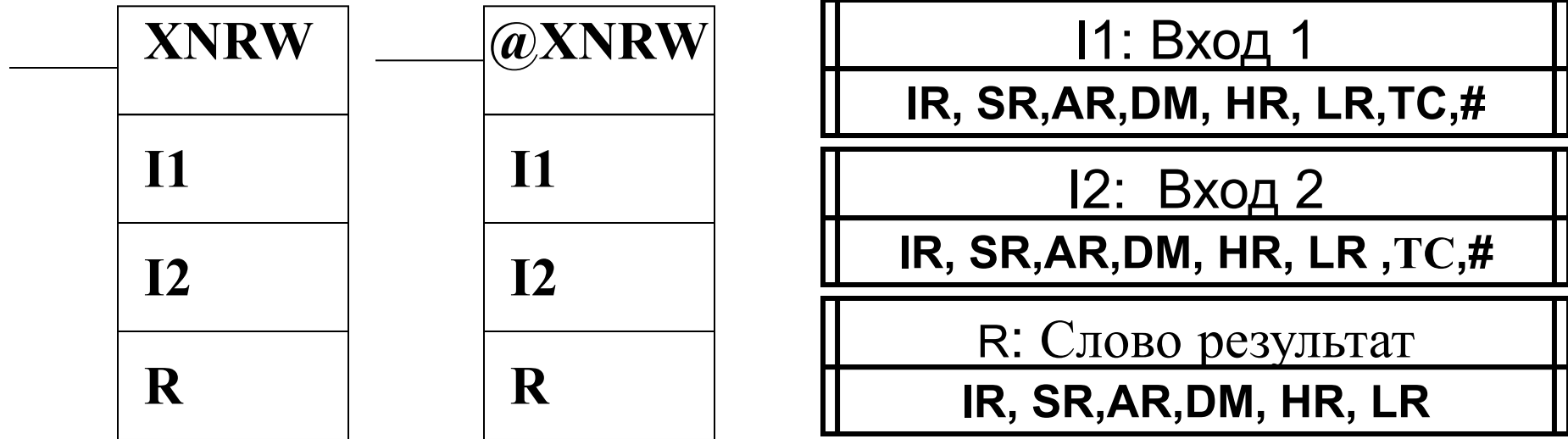
# Логические команды

## Исключающее ИЛИ - XORW(36)



# Логические команды

## Исключающее ИЛИ -НЕ - XNRW(37)



# Команды управления программой

## Нет операции – NOP (00)



**NOP**

Когда NOP(00) обнаруживается в программе, действий не производится и программа переходит к следующей команде. Когда память очищена перед программированием, во всех адресах записана команда NOP(00).

## Конец программы - END (01) (END)



**END**

END требуется в качестве последней командной линии программы. Если есть подпрограммы, END помещается после последней подпрограммы. Команды записываемые после END не выполняются, что можно применить при отладке программы.

Если в программе отсутствует END (01), никакие команды не выполняются и появляется сообщение NO END INST.

**END (01) сбрасывает в 0 флаги ER, CY, GR, EQ, LE.**

# Команды управления программой

## Секция сблокированных выходов INTERLOCK и INTERLOCK CLEAR - IL(02) и ILC(03)

**IL**

Если условие исполнения для IL(02) = 1, то программа между IL и ILC будет выполняться как написано.

**ILC**

Если условие для IL(02) = 0, блок выходных команд между IL(02) и ILC(03) будет обрабатываться, как указано в следующей таблице:

Команда	Обработка
OUT и OUT NOT	Заданные биты устанавливаются в 0
TIM TIMH(15)	Сброс
CNT, CNTR(12)	Сохраняется текущее значение
KEEP(11)	Сохраняется состояние бита
DIFU(13), DIFD(14)	Не выполняются (смотри ниже)
Все другие команды	Команды не выполняются.

# Команды управления программой

Переход и Конец перехода - JMP(04) и JME(05)

JMP
N

JME
N

<b>N: Номер перехода</b>
<b> #(0...49)</b>

Когда условие исполнения для JMP = 1, перехода не происходит и программа выполняется без пропусков. Когда условие исполнения для JMP = 0, происходит переход к JME с номером, таким же, как и у JMP, и далее выполняются команды, находящиеся после JME.

# Команды управления программой

## Команды секции STEP: STEP(08) и SNXT(09)

STEP
B

<b>B: Бит управления</b>
<b>IR, AR, HR, LR</b>

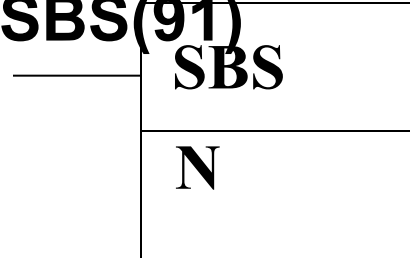
SNXT
B

Команды STEP(08) и SNXT(09) совместно служат для задания точек прерывания между секциями в длинных программах, чтобы секция могла быть отработана как единый блок и сброшена после исполнения. Секция программы обычно определяется соответствующей фактическому процессу. Команды секции STEP аналогичны обычным командам, за исключением того, что некоторые команды (END(01), FAL(06), FALS(07), JMP(04) /JME(05) и SBN(92)) нельзя включать в секцию.

# Команды подпрограмм

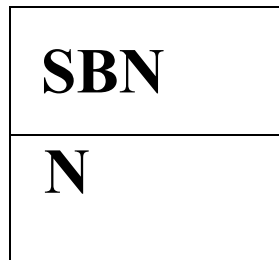
Войти в подпрограмму -

**SBS(91)**



Начало подпрограммы

**-SBN(92)**



Возврат - **RET(93)**



**N: Номер подпрограммы**

**#(0...49)**



# Команды подпрограмм

## Функция Макро - MCRO(99)

<b>MCRO</b>	<b>@MCRO</b>
<b>N</b>	<b>N</b>
<b>I1</b>	<b>I1</b>
<b>O1</b>	<b>O1</b>

<b>N: номер подпрограммы</b>
<b>00...49</b>
<b>I1: Первое слово входа</b>
<b>IR, SR,AR,DM, HR, TC,LR</b>
<b>O1: Первое слово выхода</b>
<b>IR, SR,AR,DM, HR, LR</b>

Команда MCRO позволяет написать одну подпрограмму вместо нескольких подпрограмм, которые имеют одинаковую структуру, но разные операнды. Есть 4 слова входа IR 232 .... IR 235 и 4 слова выхода IR 236 .... IR 239. Данные 8 слов используются в подпрограмме и берут свое содержимое из I1 ... I1+3 и O1 .... O1+3 при исполнении подпрограммы.

Когда условие исполнения = 1, MCRO копирует содержание I1 .... I1+3 в IR 232 .... IR 235 , содержание O1 .... O1+3 в IR 236 .... IR 239 и затем вызывает и исполняет подпрограмму, заданную в N. По завершении подпрограммы содержимое IR 096 .... IR 099 передаются обратно в O1 .... O1+3.



# Прерывания

- входные прерывания
- интервальные прерывания
- прерывания высокоскоростного счетчика

# Управление прерываниями

Обозначение на схеме

INT(89)
CC
000
D

@INT(89)
CC
000
D

CC	код управления	#(000..003, 100, 200)
000	нет функций	
D	слово состояния	IR, SR, AR, HR, TC, LR, TR, DM, #

<b>Функции INT(89)</b>	<b>CC</b>
Маскировать/размаскировать входные прерывания	000
Очистить входные прерывания	001
Читать текущее состояние маски	002
Возобновить заданное значение счетчика	003
Замаскировать все прерывания	100
Размаскировать все прерывания	200

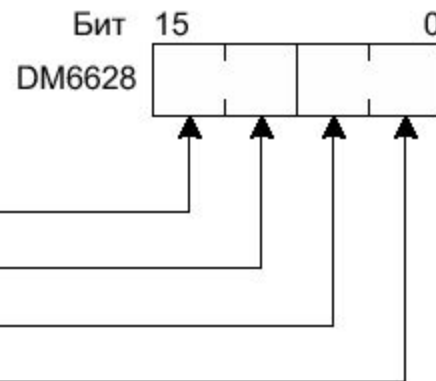
# Входные прерывания

## 1. Установочные параметры входных прерываний

Модель ПК	Вход	Номер прерывания	Время реакции	
			Режим прерывания	Режим счетчика
CPM1-10CDR-_ CPM1A-10CDR-__	0003	00	0.3 ms max (время до выполнения программы прерывания)	1 kHz
	0004	01		
CPM1-20CDR-_ CPM1A-20CDR-__	0003	00		
	0004	01		
CPM1-30CDR-_ CPM1A-30CDR-__	0003	02		
	0004	03		

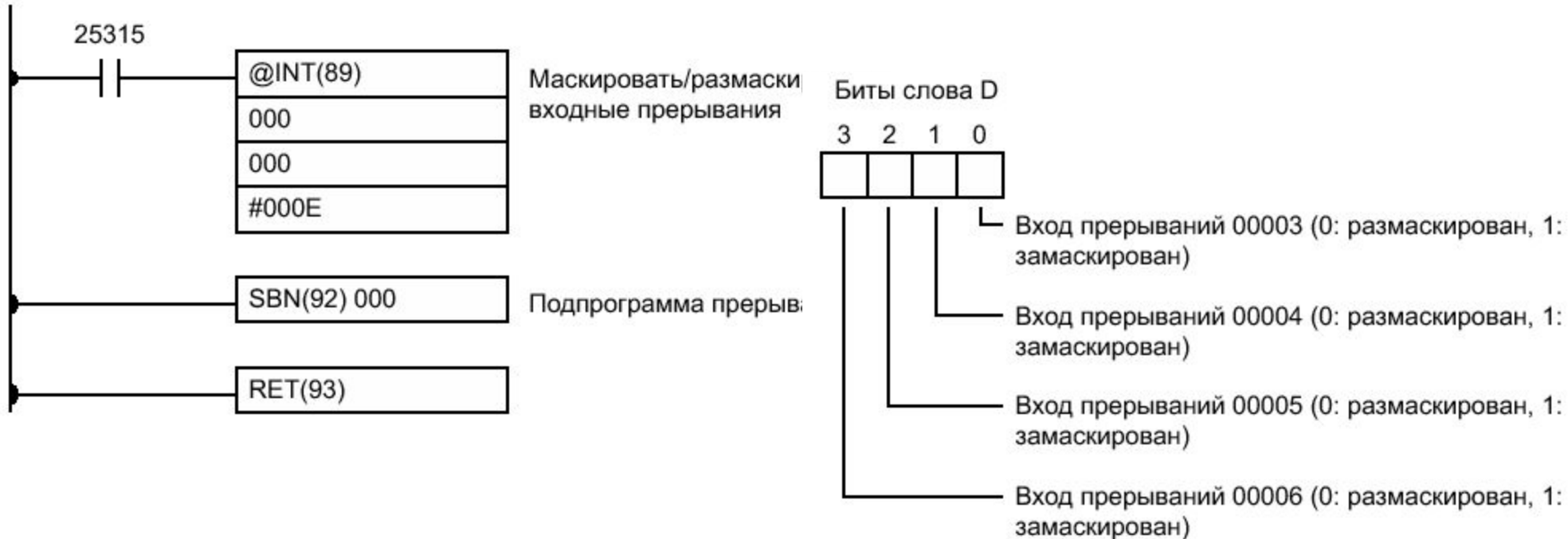
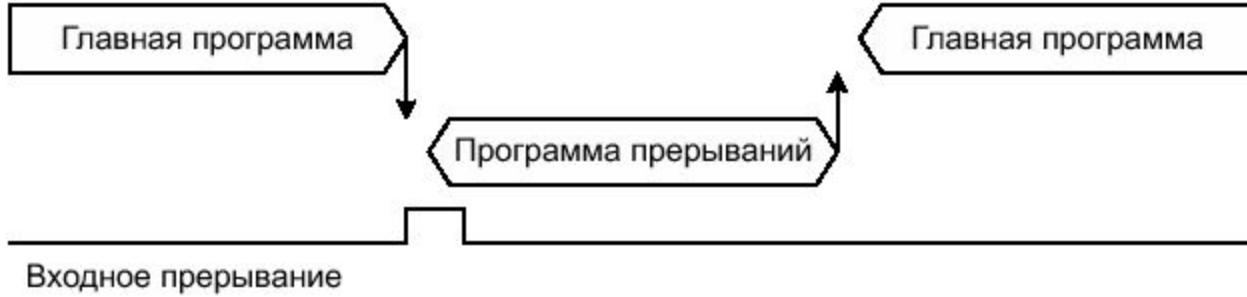
Слово	Значение параметра
DM 6628	0: Обычный вход (по умолчанию) 1: Вход прерывания 2: Вход быстрого реагирования

- Задание для входа 6: установите в 1
- Задание для входа 5: установите в 1
- Задание для входа 4: установите в 1
- Задание для входа 3: установите в 1



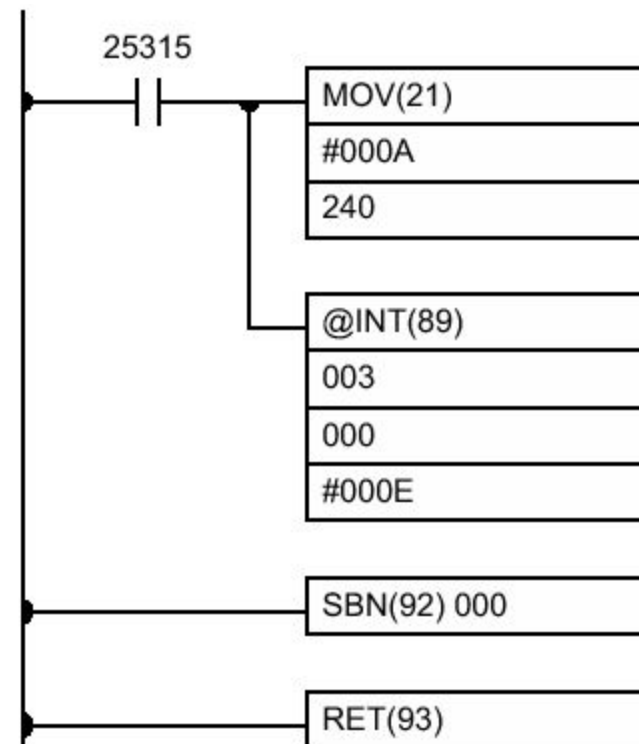
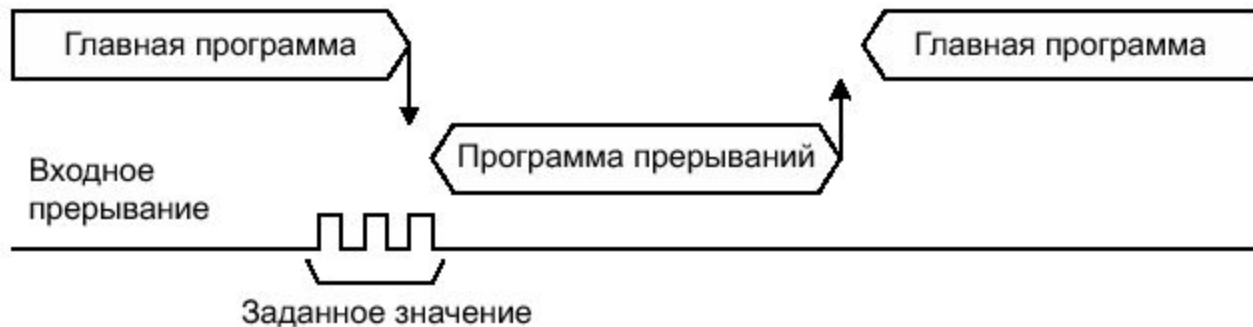
# Входные прерывания

## 2. Режим прерывания по входу



# Входные прерывания

## 3. Режим счета



Прерывание	Слово, содержащее задание	Слово, содержащее текущее значение -1
Вход 00003 (Вх. прерывание 0)	SR 240	SR 244
Вход 00004 (Вх. прерывание 1)	SR 241	SR 245
Вход 00005 (Вх. прерывание 2)	SR 242	SR 246
Вход 00006 (Вх. прерывание 3)	SR 243	SR 247

# Интервальные прерывания

## Управление интервальным таймером

Обозначение на схеме

STIM(69)
C1
C2
C3

@STIM(69)
C1
C2
C3

C1	данные управления #1	000..008, 010..012
C2	временной интервал	IR, SR, AR, HR, TC, LR, DM, #
C3	номер программы прерывания	IR, SR, AR, HR, TC, LR, DM, #

Функция	Таймер	Задание	ПК
Пуск таймеров	0	000	CQM1/CPM1/CPM1A/ SRM1
	1	001	Только CQM1
	2	002	
Пуск прерываний по расписанию	0	003	CQM1/CPM1/CPM1A/ SRM1
	1	004	Только CQM1
	2	005	
Чтение текущего значения таймера	0	006	CQM1/CPM1/CPM1A/ SRM1
	1	007	Только CQM1
	2	008	
Остановка таймеров	0	010	CQM1/CPM1/CPM1A/ SRM1
	1	011	Только CQM1
	2	012	

# Прерывания высокоскоростного счетчика

Режим	Функции входа	Метод входа	Частота счета	Диапазон счета	Метод управления
Плюс/минус	00000: Вход фазы A 00001: Вход фазы B 00002: Вход фазы Z	Разность фаз, 4x входа	2.5 kHz макс.	-32767.. 32767	Управление конечным значением: можно записать до 16 конечных значений и номеров подпрограмм прерываний. Управление сравнением с зоной: можно записать до 8 зон и номеров подпрограмм прерываний.
Инкрементальный	00000: Счетный вход 00001: См. Зам. 00002: Вход сброса	Индивидуальные входы	5 кГц макс.	0.. 65535	

Биты DM 6642	Функция	Значения		
		Инкремент	Плюс/минус	Не используются
00 - 03	Задаёт режим счетчика 0: Плюс/минус 4: Инкрементальный	4	0	0 или 4
04 - 07	Задаёт Метод сброса 0: Сброс фазой Z + системной программой 1: Сброс системной программой	0 или 1	0 или 1	0 или 1
08 - 15	Задаёт счетчик: 00: Счетчик не используется 01: Счетчик используется	01	01	00

# Прерывания высокоскоростного счетчика

@CTBL(63)
P
C
TB

C: три цифры BCD  
000: задать таблицу конечных значений и начать сравнение  
001: задать таблицу зон и начать сравнение  
002: задать только таблицу конечных значений  
003: задать только таблицу зон  
TB: начальное слово таблицы сравнений

Текущее значение высокоскоростного счетчика хранится в SR 248 и SR 249  
Сброс счетчика осуществляется установкой в й 1 бита SR 252.00

<b>CTBL</b>
<b>0</b>
<b>0</b>
<b>DM0</b>

Структура таблицы в режиме сравнения конечных значений:

DM 0000	0002	Число условий сравнения:- 2
DM 0001	1000	Конечное значение 1: 1000
DM 0002	0000	
DM 0003	0030	Номер подпрограммы обработки прерывания сравнения 1: 30
DM 0004	2000	Конечное значение 2: 2000
DM 0005	0000	
DM 0006	0031	Номер подпрограммы обработки прерывания сравнения 2: 31



# Прерывания высокоскоростного счетчика

Использование  
счетчика  
в режиме  
сравнения с зоной

<b>CTBL</b>
<b>0</b>
<b>1</b>
<b>DM0</b>

Структура таблицы в режиме сравнения с зоной

DM 0000	1500	
DM 0001	0000	Нижняя граница 1: 1 500
DM 0002	3000	
DM 0003	0000	Верхняя граница 1: 3 000
DM 0004	0040	Номер подпрограммы прерывания зоны 1: 40
DM 0005	7500	
DM 0006	0000	Нижняя граница 2: 7 500
DM 0007	0000	
DM 0008	0001	Верхняя граница 2: 10 000
DM 0009	0041	Номер подпрограммы прерывания зоны 2: 41
DM 0010	0000	
DM 0011	0000	
DM 0012	0000	
DM 0013	0000	
DM 0014	FFFF	Подпрограмма прерывания зоны 3 не исполняется
...		
DM 0035	0000	
DM 0036	0000	
DM 0037	0000	
DM 0038	0000	
DM 0039	FFFF	Подпрограмма прерывания зоны 8 не исполняется