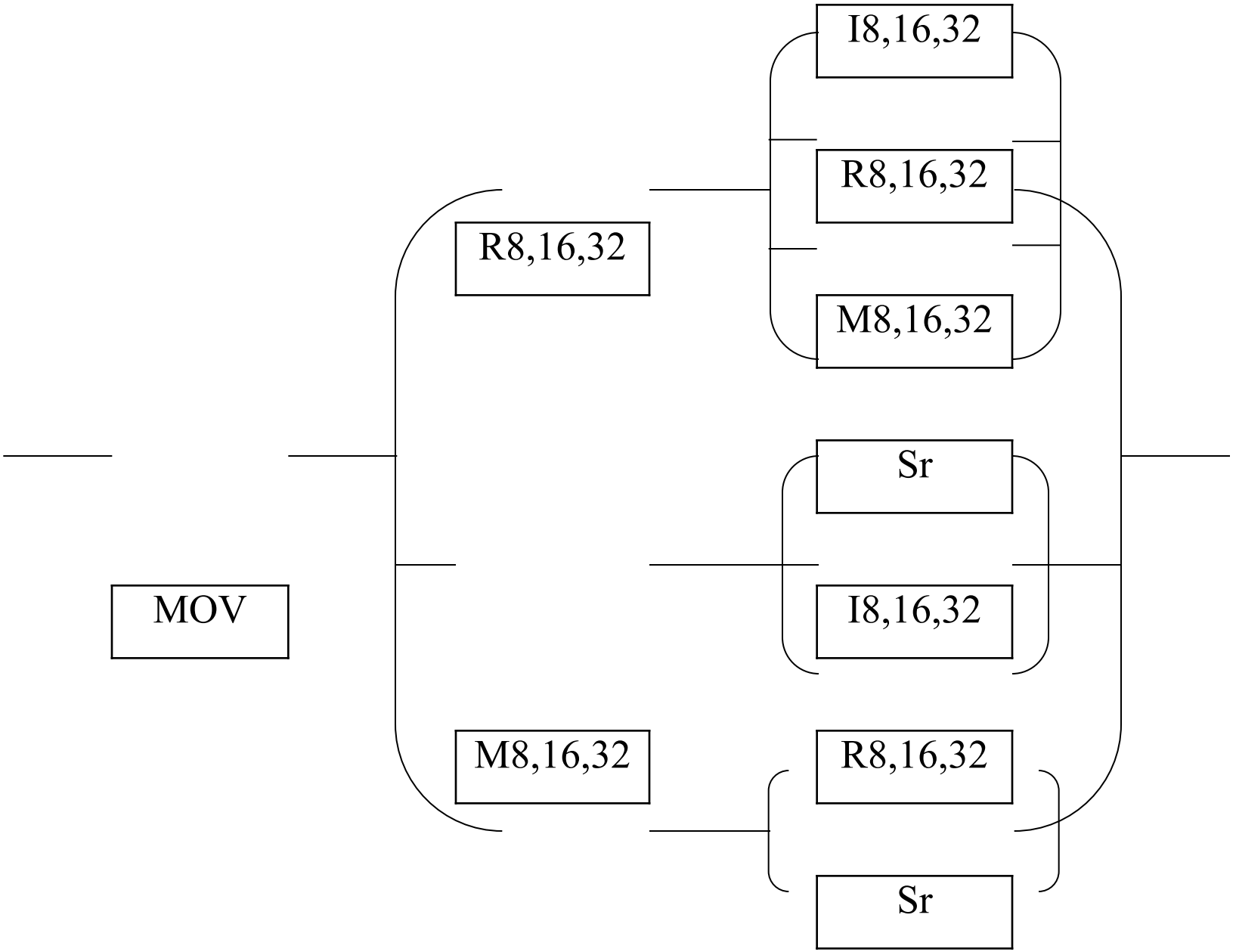
Основные команды ассемблера

Пересылки данных	Арифметические	Логические
Передачи управления	Обработки цепочек	Управления работой ЦП

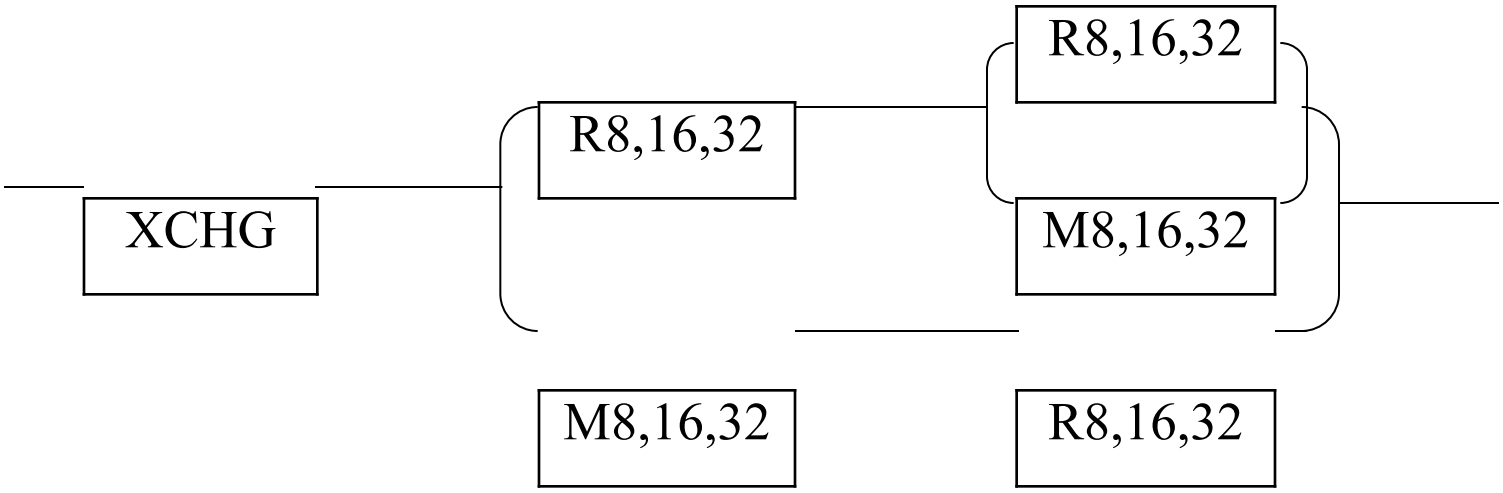
Команды пересылки данных

Общего назначения	Работы с адресами	Работы со стеком	Преобразования данных
Mov	Lea	Puch	Xlat
Xchg	Lss	Pop	
	Lds	Pucha	
	Les	Popa	
	Lfs	Puchf	
	Lgs	Popf	

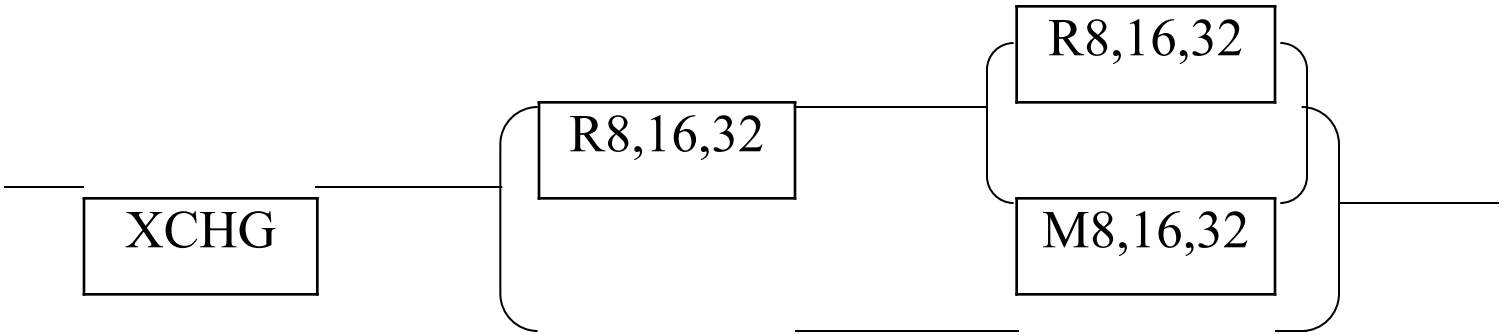
Инструкция MOV



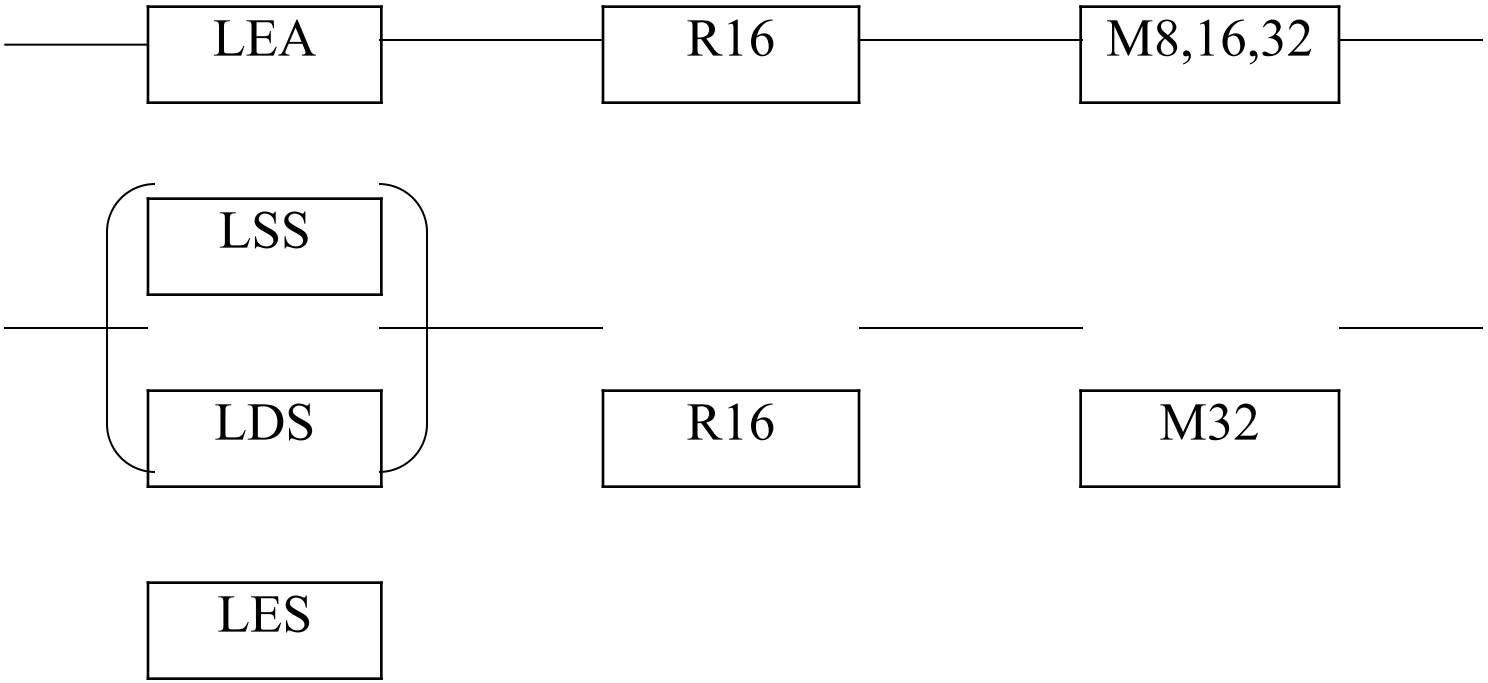
Инструкция обмена данными



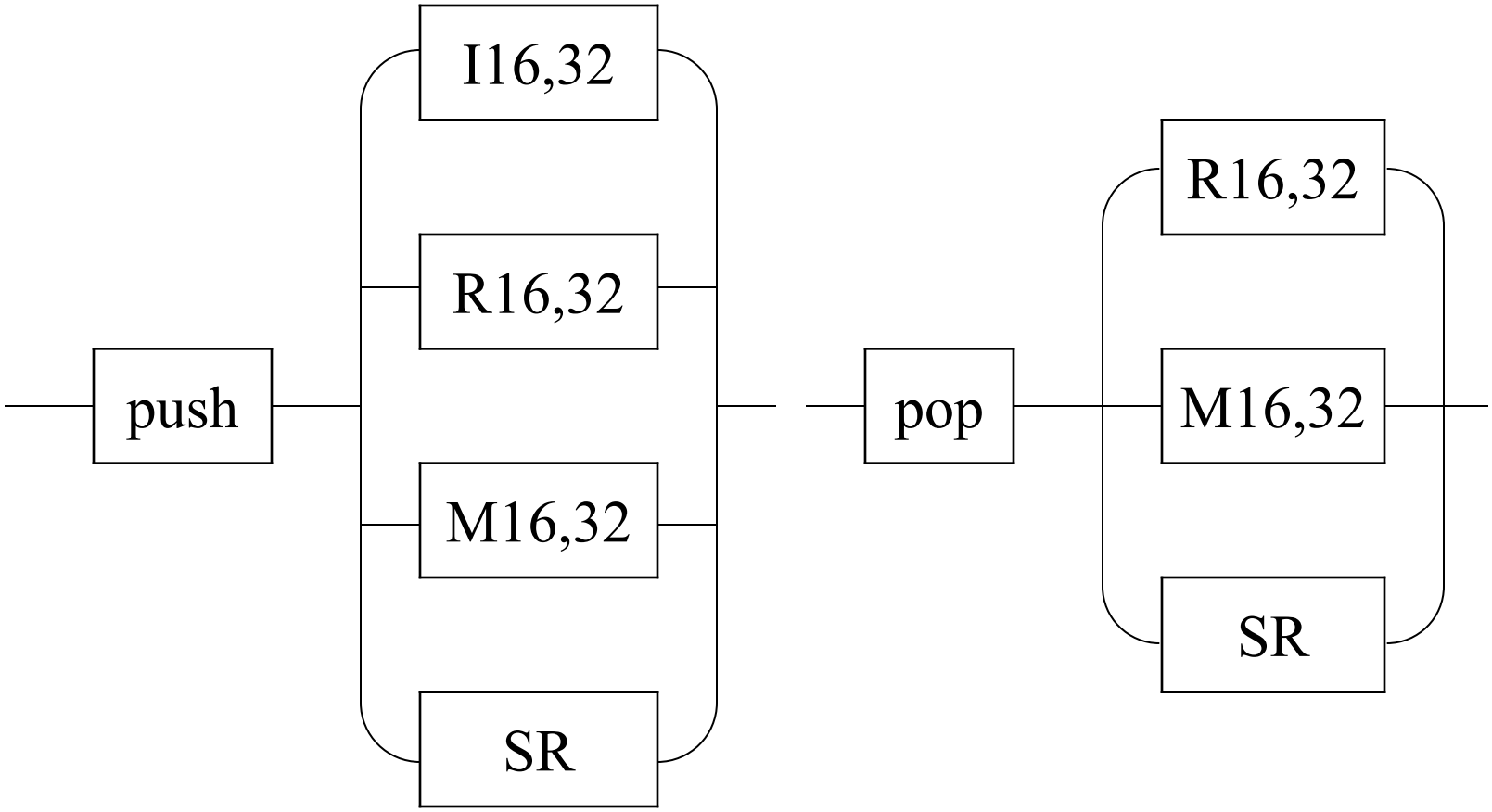
Инструкция обмена данными



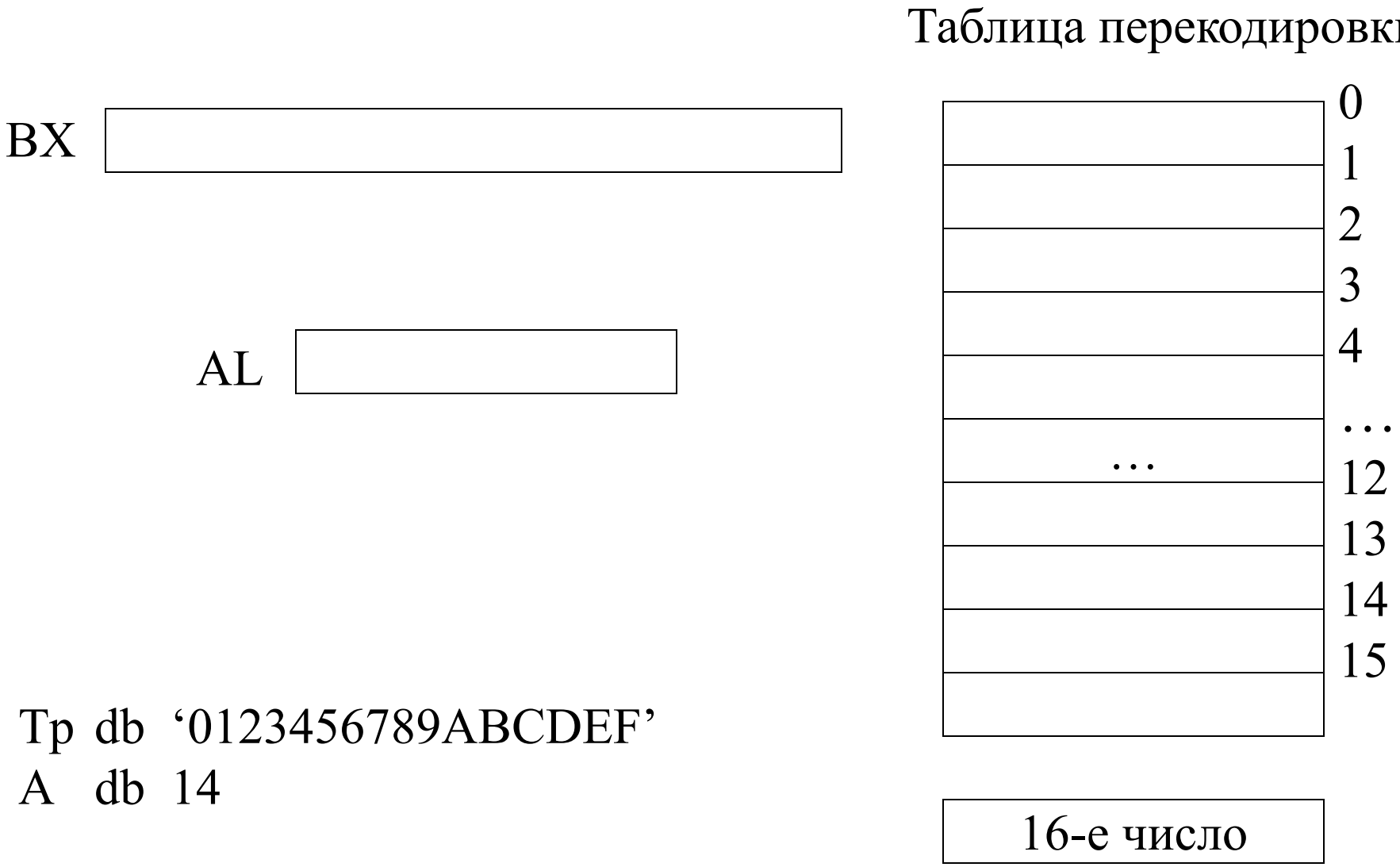
Инструкции загрузки адреса



Инструкции работы со стеком

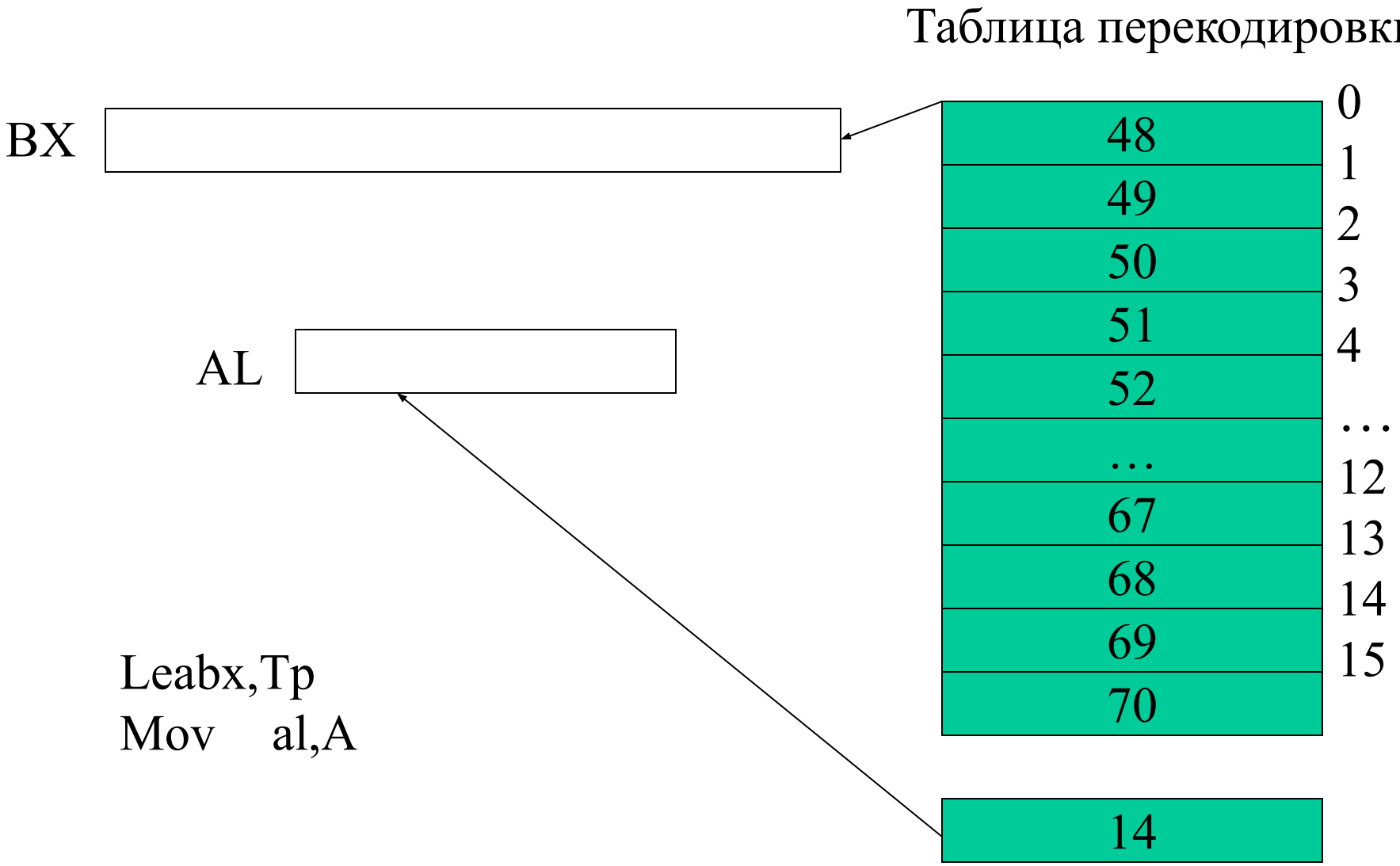


Инструкция перекодировки xlat

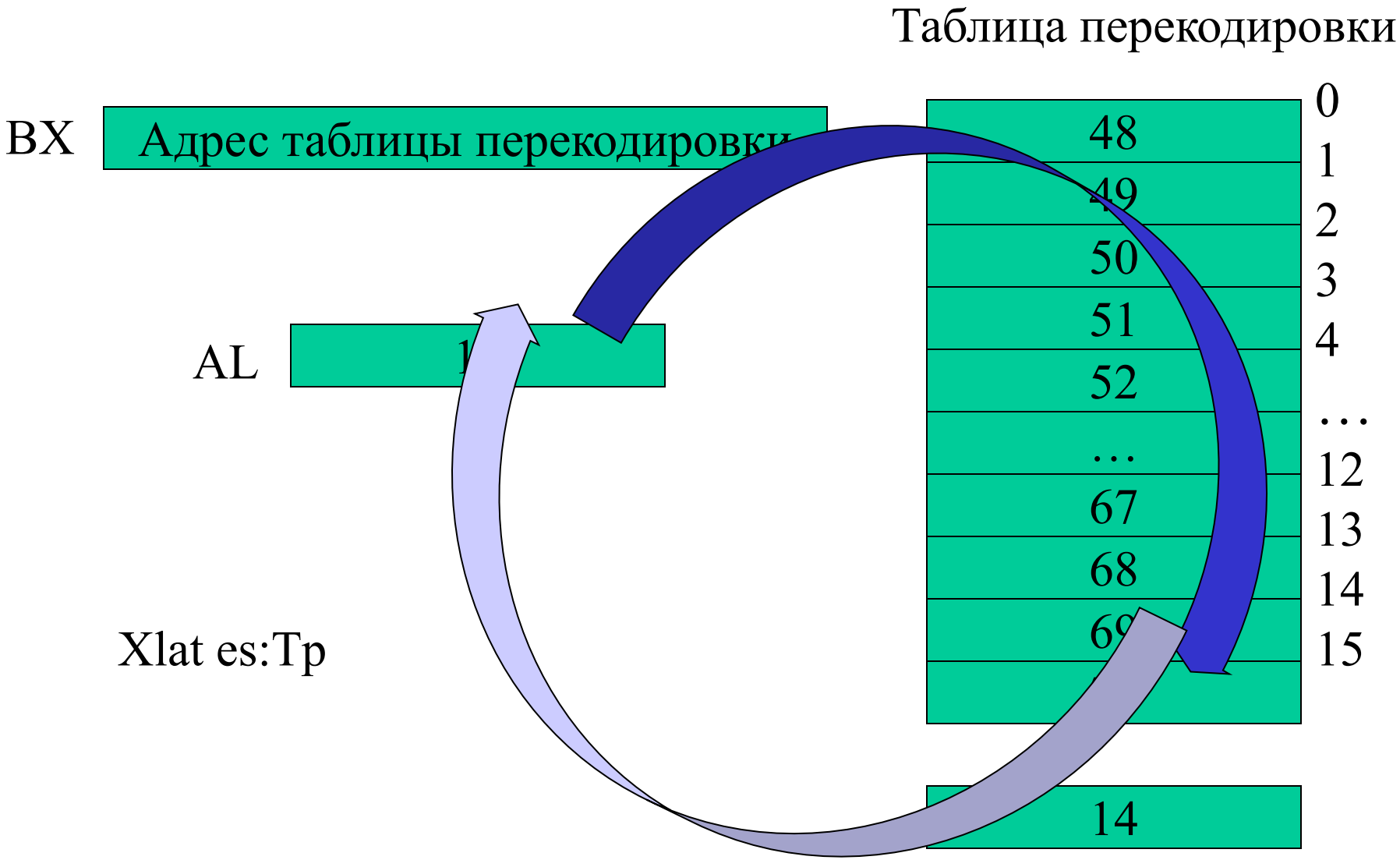



```
Tp db  '0123456789ABCDEF'
A  db  14
```

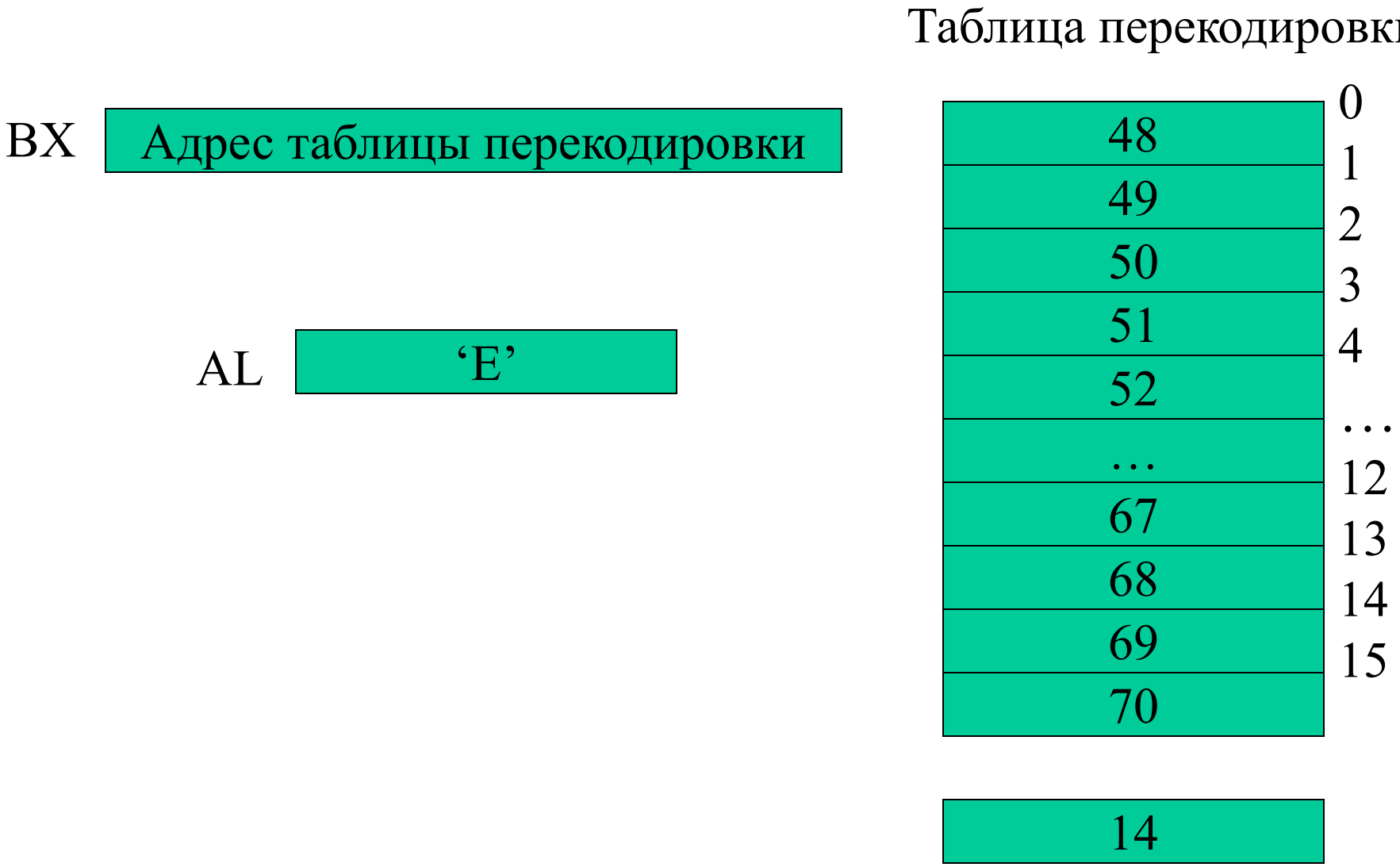
Инструкция перекодировки xlat



Инструкция перекодировки xlat



Инструкция перекодировки xlat

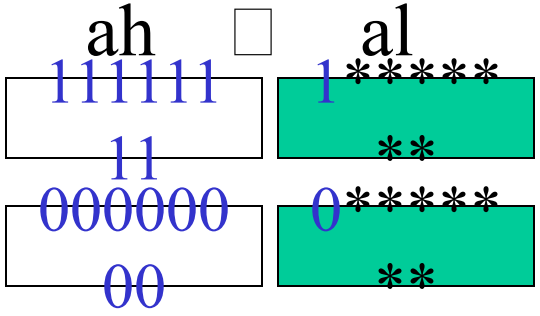


Арифметические команды

Преобразования типов	Двоичной арифметики		Десятичной арифметики	Прочие
Cbw	Add	Imul	Aaa	Cmp
Cwd	Adc	Mul	Daa	Set*cc*
Cwde	Inc	Idiv	Aas	
Cdq	Sub	Div	Das	
Movsx	Sbb	Neg	Aam	
Movzx	Dec		Aad	

Преобразование

- Байта в слово

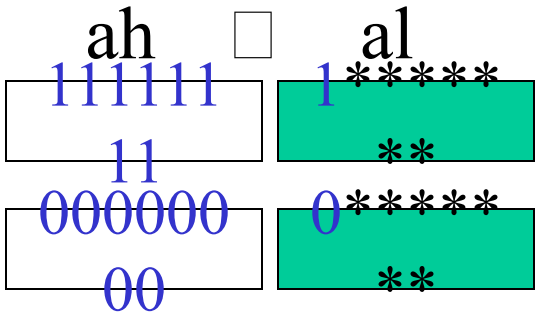


- Слова в двойное слово
  - Cwd: ax dx
  - Cwde: ax eax
- Двойного слова в учетверенное
  - Cdq: eax edx

Пересылка

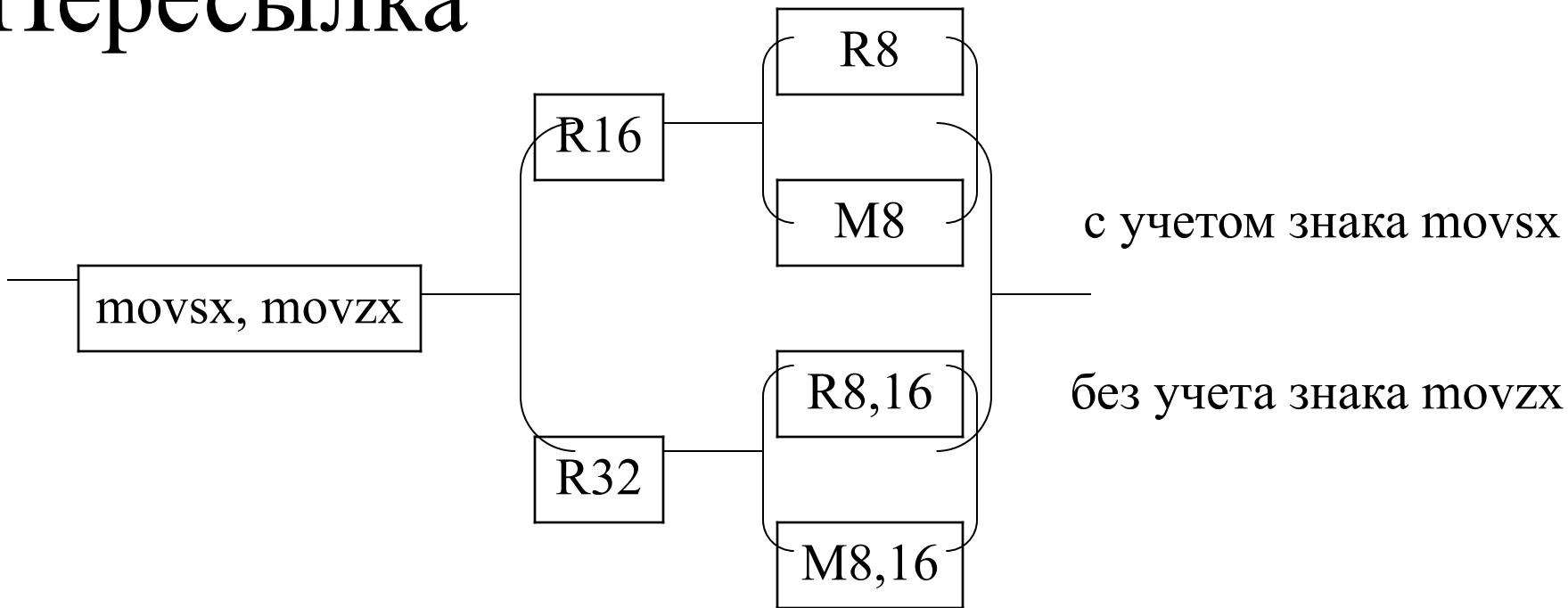
Преобразование

- Байта в слово



- Слова в двойное слово
  - Cwd: ax dx
  - Cwde: ax eax
- Двойного слова в учетверенное
  - Cdq: eax edx

Пересылка

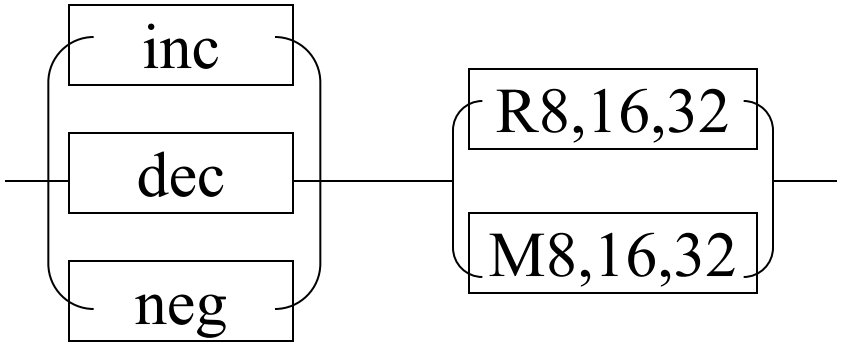


Двоичная арифметика

Inc – увеличение на 1*)

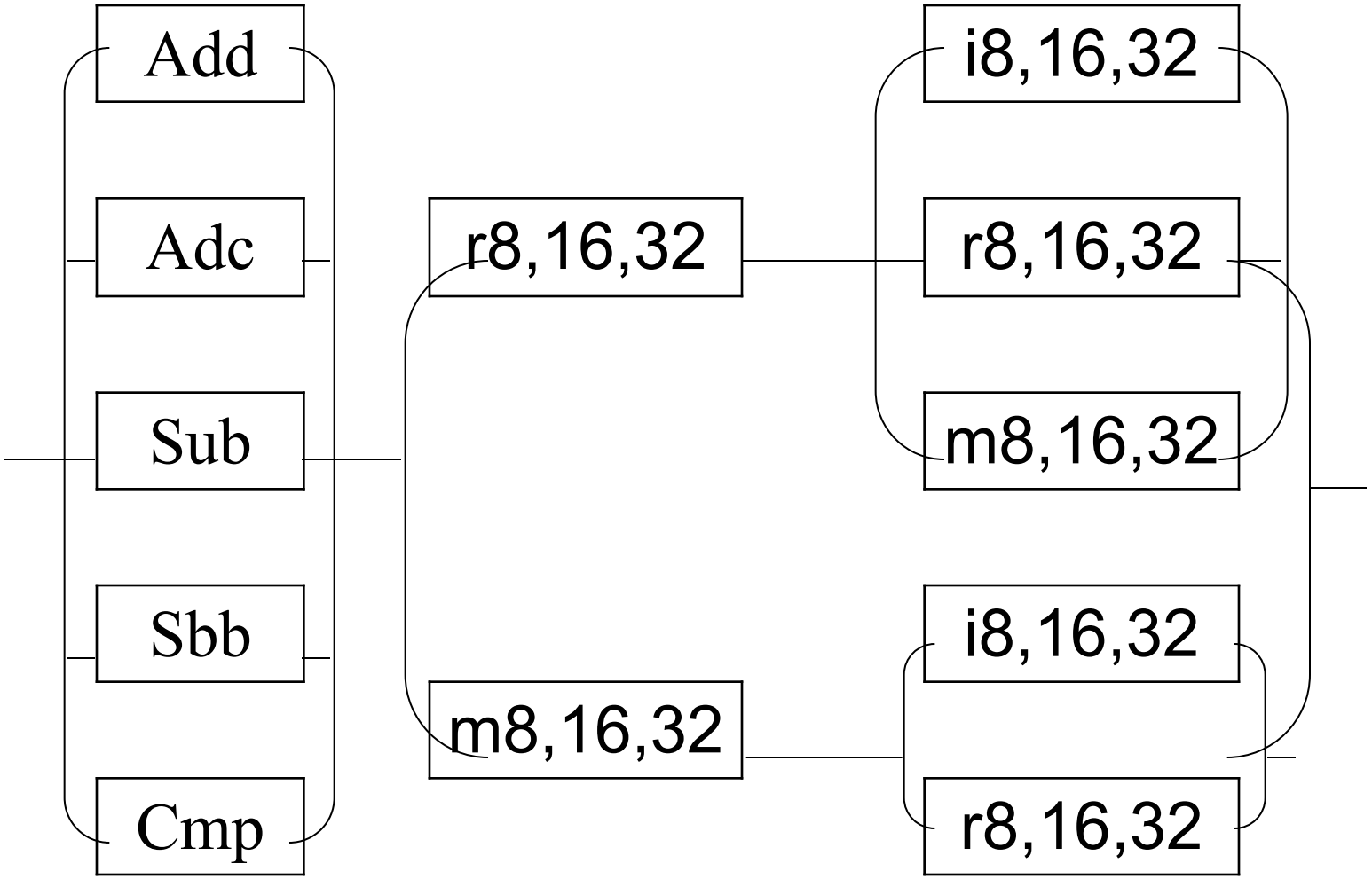
Dec – уменьшение на 1*)

Neg – смена знака

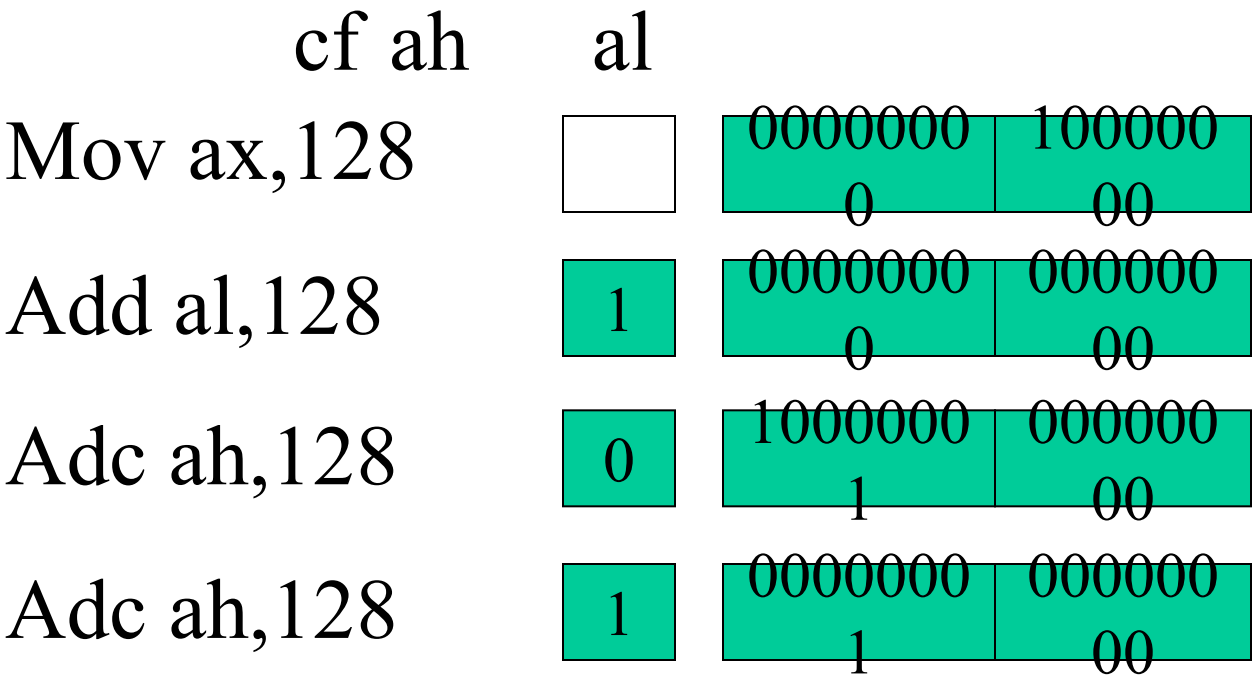


*) Не изменяет флага cf.

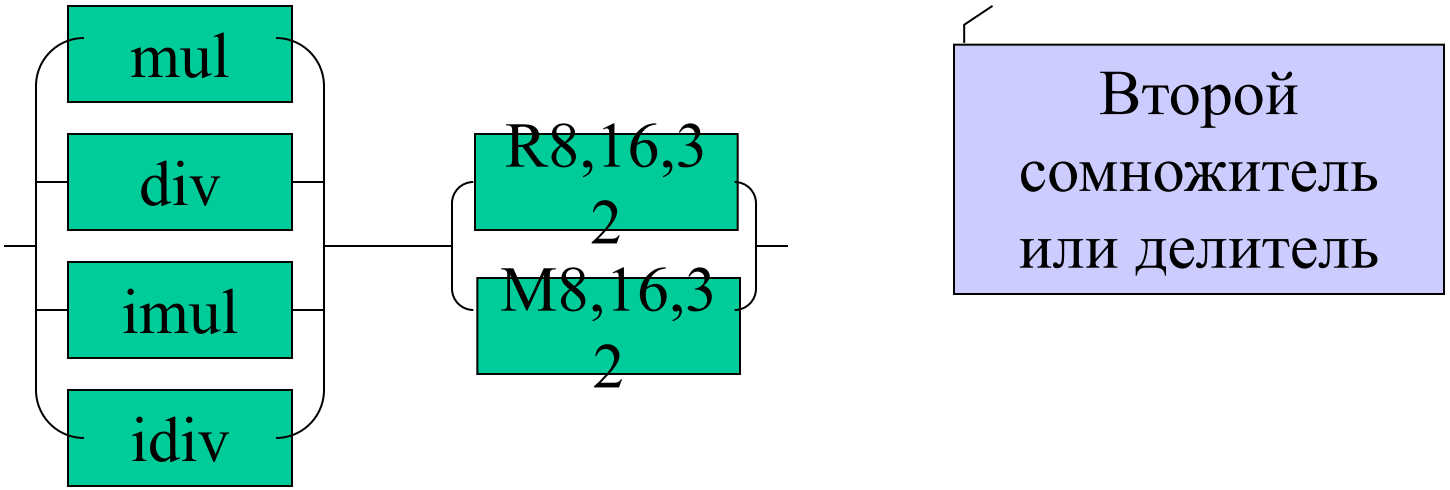
Сложение, вычитание сравнение



Пример


cf ah
al
Mov ax,128
Add al,128
Adc ah,128
Adc ah,128
0000000 0
100000 00
1
0000000 0
000000 00
0
1000000 1
000000 00
1
0000000 1
000000 00

Умножение, деление



Тип операнда	Первый сомножитель	Результат	Делимое	Результат	
				Частное	Остаток
8	al	ax	ax	al	ah
16	ax	dx:ax	dx:ax	ax	dx
32	eax	edx:eax	edx:eax	eax	edx

Делитель 0 или частное велико – исключительная ситуация

Десятичная арифметика

Имя	Содержание
Для неупакованных BCD чисел в регистре al	
Aaa	ASCII-коррекция после сложения
Aas	ASCII-коррекция после вычитания
Aam	ASCII-коррекция после умножения
Aad	ASCII-коррекция перед делением
Для упакованных BCD чисел в регистре al	
Daa	Десятичная коррекция после сложения
Das	Десятичная коррекция после вычитания

AAA и AAS: примеры

- mov ax,8
  add al,5
  aaa

```
00 08
00 05
00 0d   al > 9, af = 0
01 03   af = cf = 1
```

- mov ax,’18’
  add al,’9’
  aaa

```
31 38
00 39
31 71   af = 1
32 07   af = cf = 1
```

- mov ax,’18’
  sub al,9
  aas

```
31 38
00 09
31 2f   af = 1
30 09   af = cf = 1
```

AAM и AAD: примеры

- mov al,9 — ** 09
  mov bl,9 — ** 09
  mul bl — 00 51
  aam — 08 01
- mov al,99 — 00 63
  aam — 09 09
- mov ax,0703h — 07 03
  aad — 00 49
  mov bl,9 — 00 09
  div bl — 01 08
- mov ax,’99’ — 39 39
  aad — 00 73
  sub al,10h — 00 10
  — 00 63

DAA и DAS: примеры

- mov ax,44h
  add al,37h
  daa

  00 44
  00 37
  00 7B $al_1 > 9$, af=0
  00 81 af = 1

- Mov ax,57h
  sub al,19h
  das

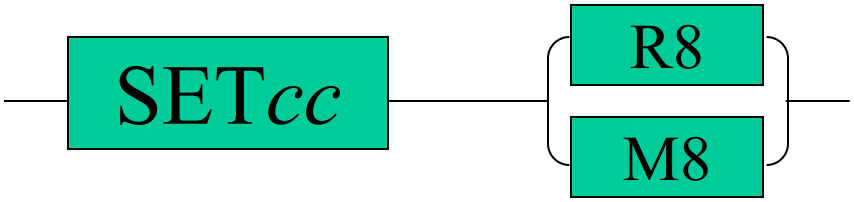
  00 57
  00 19
  00 3E af = 1
  00 38

- Mov ax,88h
  add al,12h
  daa

  00 88
  00 12
  00 9A $al_1 > 9$
  00 00 af = cf =1

- Mov ax,88h
  sub al,99h
  das

  00 88
  00 99
  00 EF af = cf =1
  00 89 af = cf =1

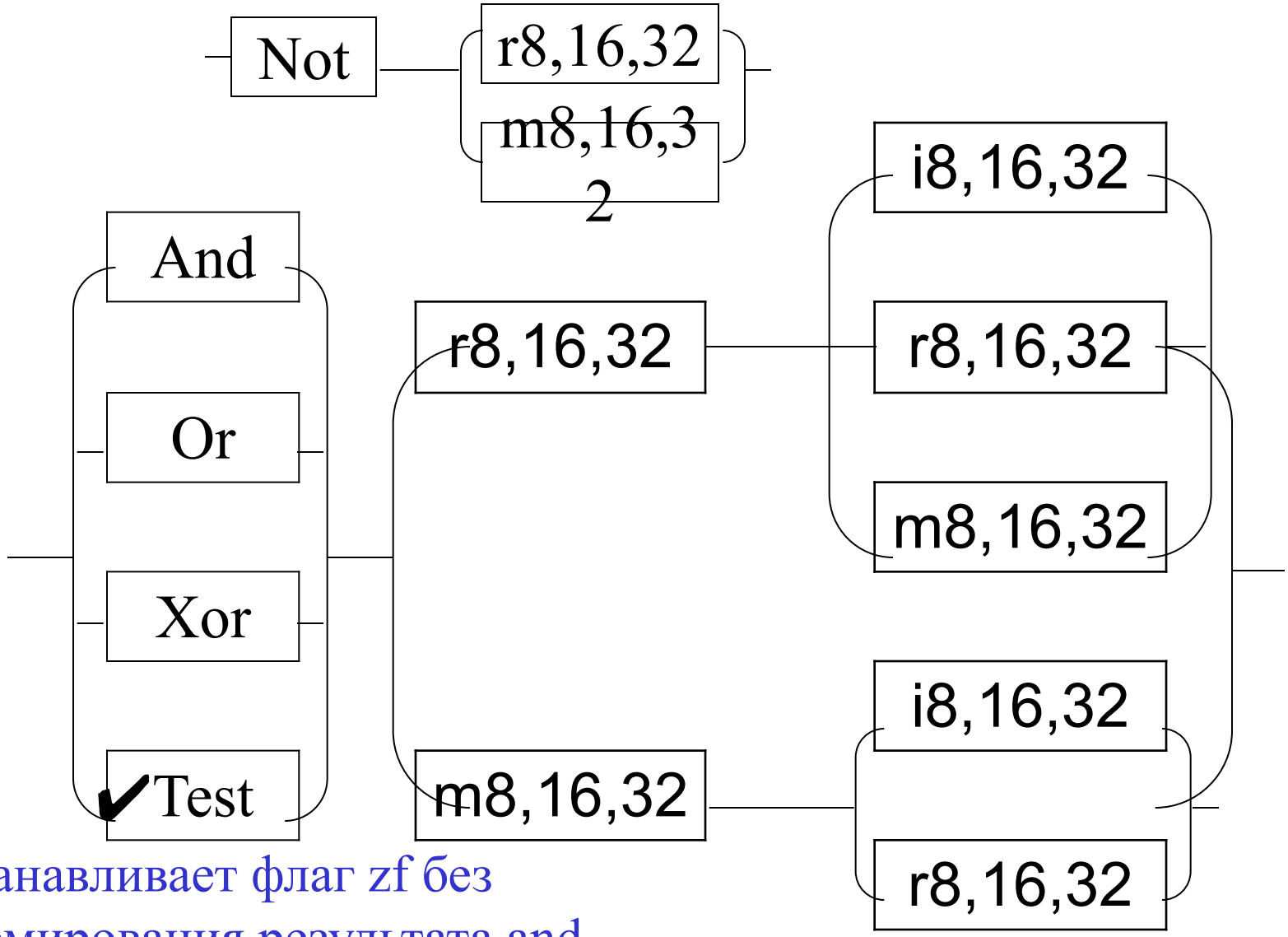


Команда	Условие	Команда	Условие
seta/setnbe	cf=0 и zf=0	setle/setng	zf=1 или sf≠of
setae/setnb	cf=0	setnc	cf=0
setb/setnae	cf=1	setne/setnz	zf=0
setbe/set	cf=1 или zf=1	setno	of=0
setc	cf=1	setnp/setpo	pf=0
sete/setz	zf=1	setns	sf=0
setg/setnle	zf=0 или sf=of	seto	of=1
setge/setnl	sf=of	setp/setpe	pf=1
setl/setnge	sf≠of	sets	sf=1

Логические команды

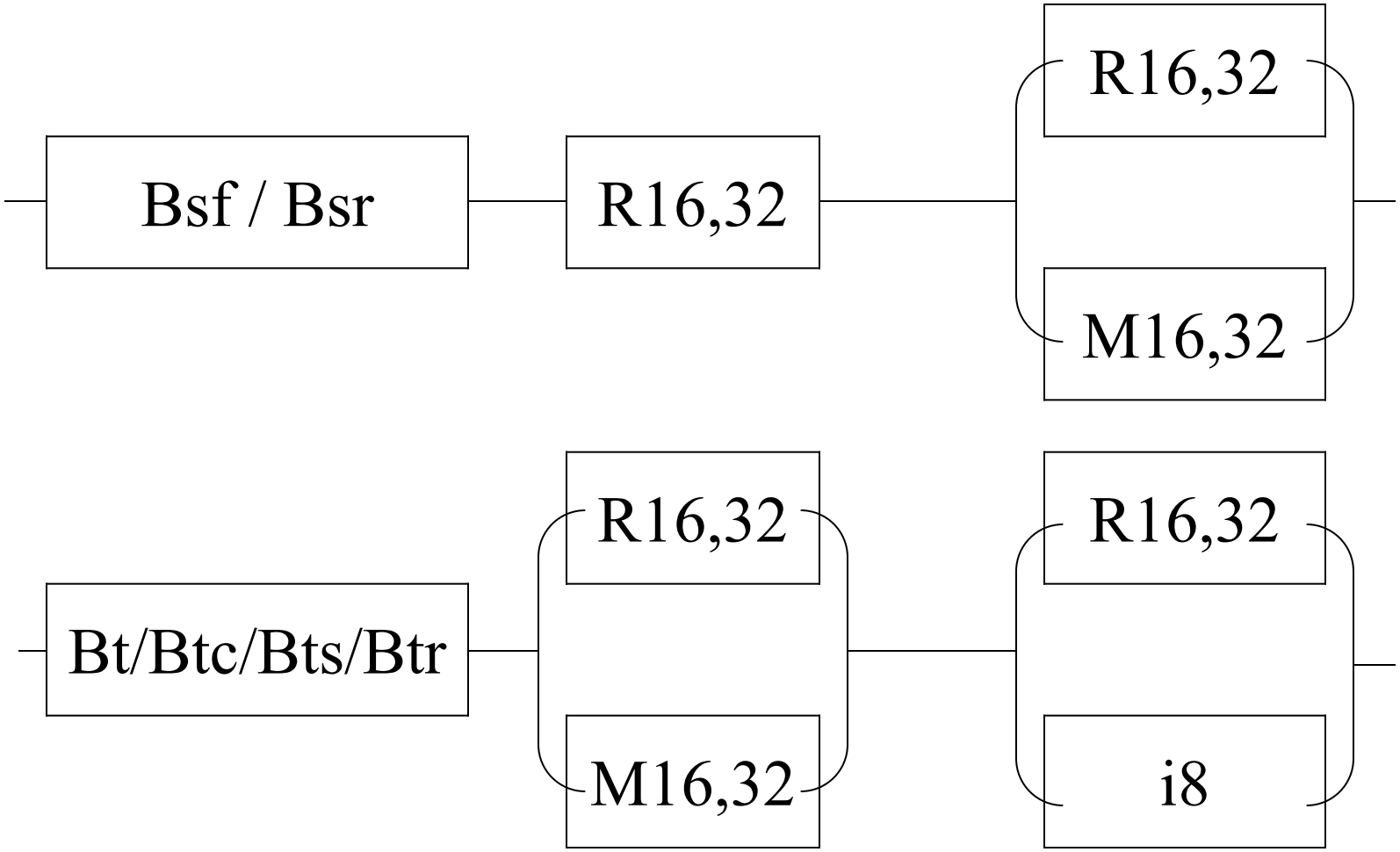
Логические	Обработки бит	Сдвига
And	Bsf	Sar
Or	Bsr	Sal
Xor	Bt	Shl
Not	Btc	Shr
Test	Btr	Shld
	Bts	Shrd
		Rcl
		Rcr
		Rol
		Ror

Побитовые булевские операции



✔устанавливает флаг zf без формирования результата and

Операции с битами



Примеры

- A dw0000h
  B dw0110h

  …

```
.386
   bsf  ax,A
   bsf  ax,B
   bsr ax,B
   bt   b,ax
   btc b,ax
   btr  b,4
   bts  b,1
```

zf=1

ax=4, zf=0

ax=8, zf=0

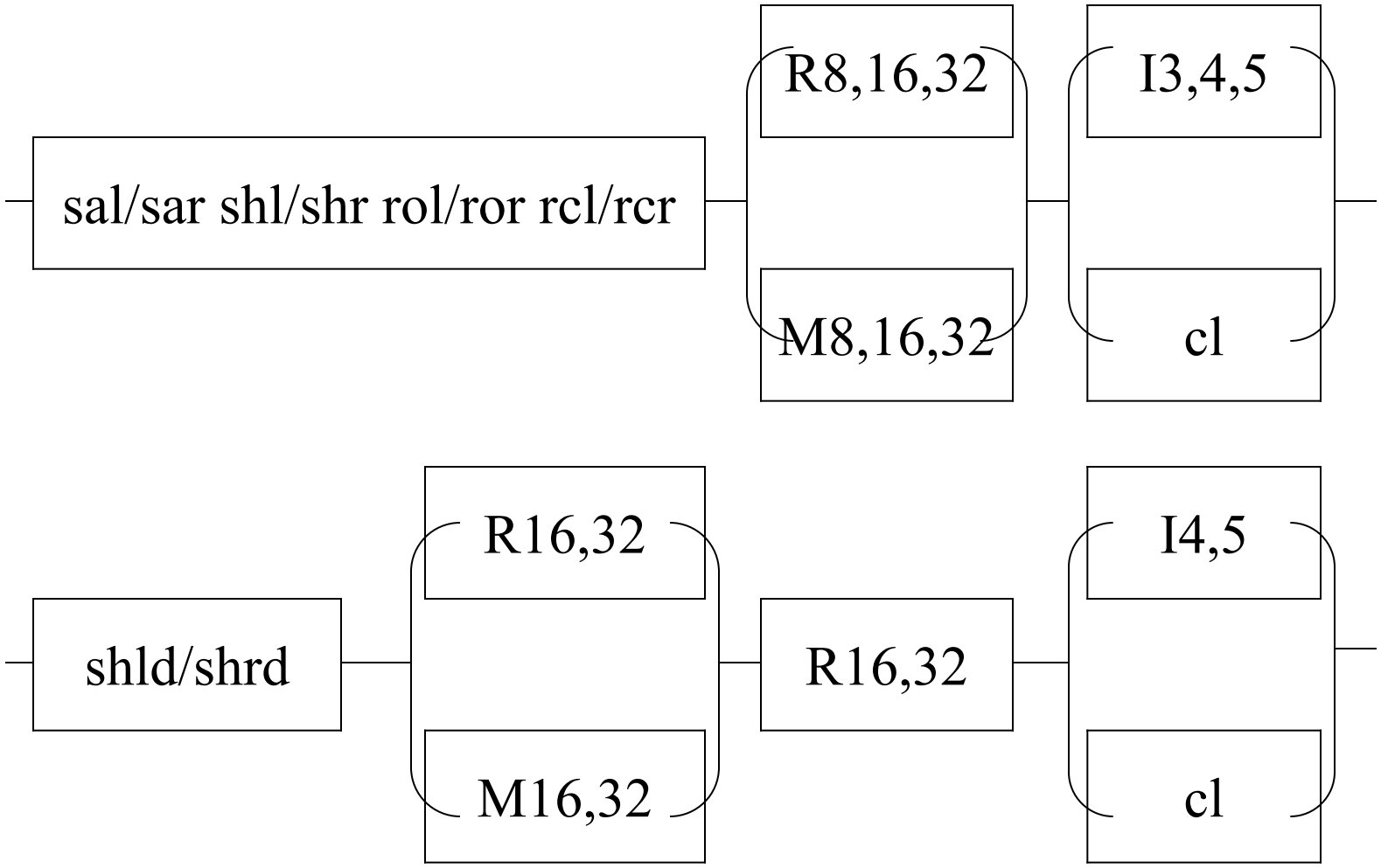
cf=1

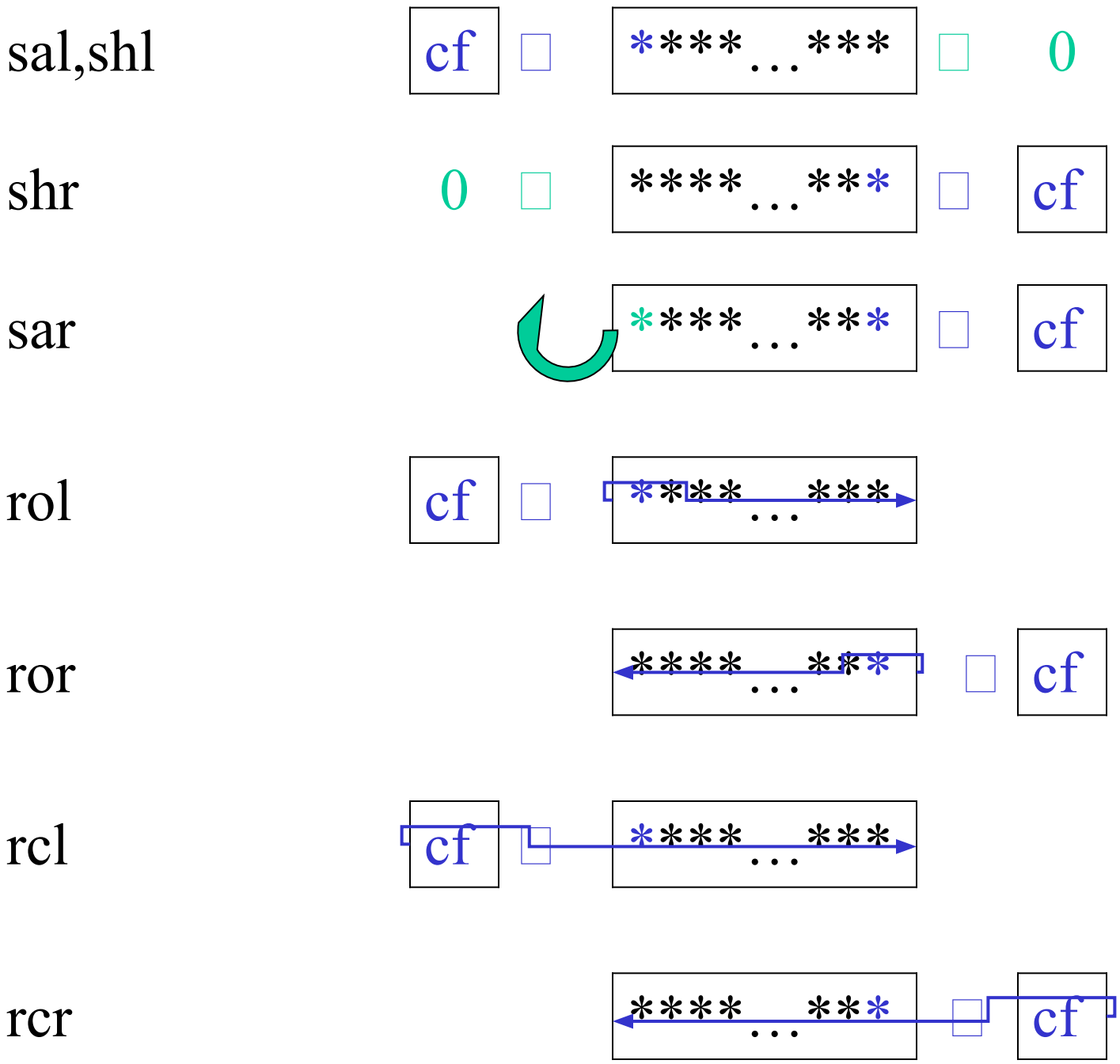
cf=1, b=0010h

cf=1, b=0000h

cf=0, b=0002h

Операции сдвига




sal,shl
cf
****...***
0
shr
0
****...***
cf
sar
****...***
cf
rol
cf
****...***
ror
****...***
cf
rcl
cf
****...***
rcr
****...***
cf

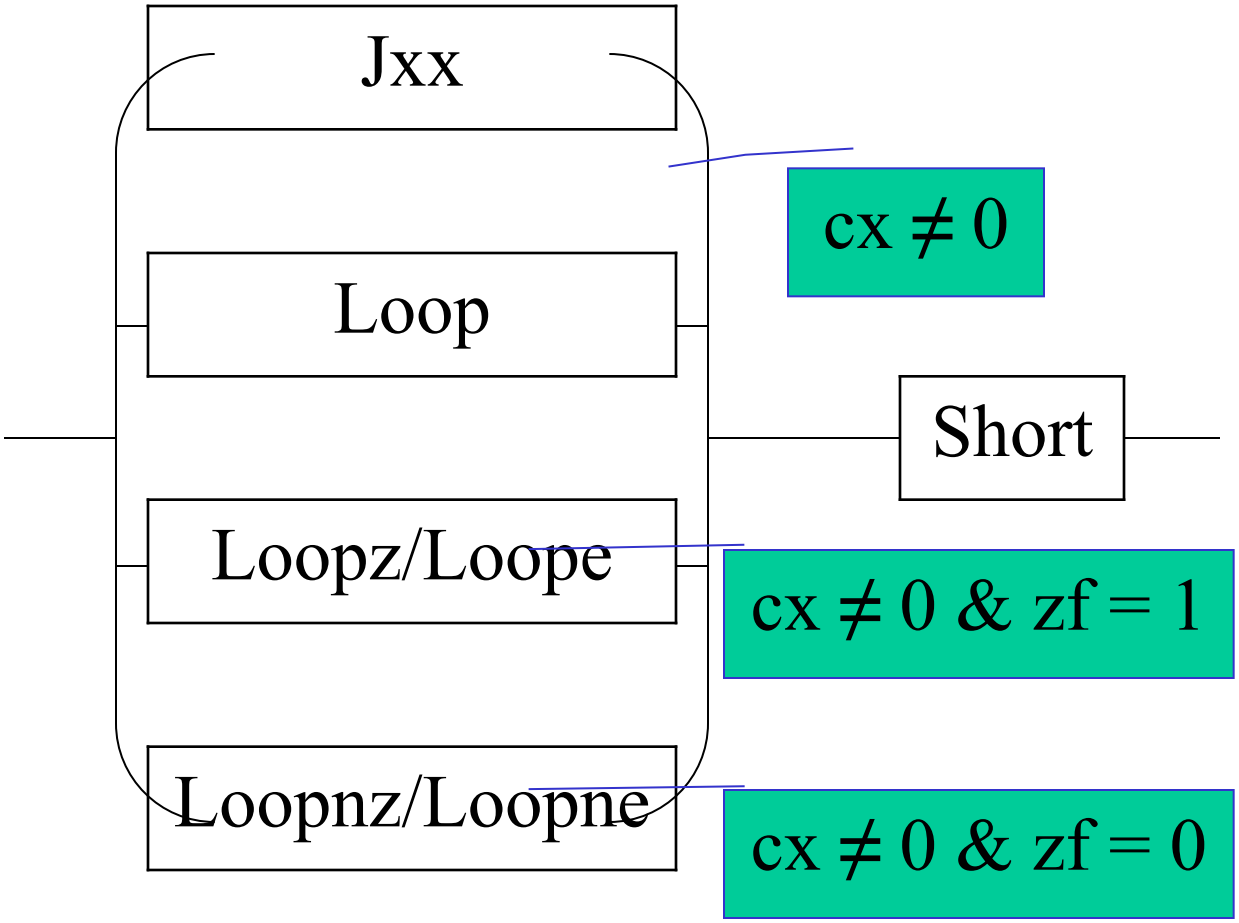
Команды передачи управления

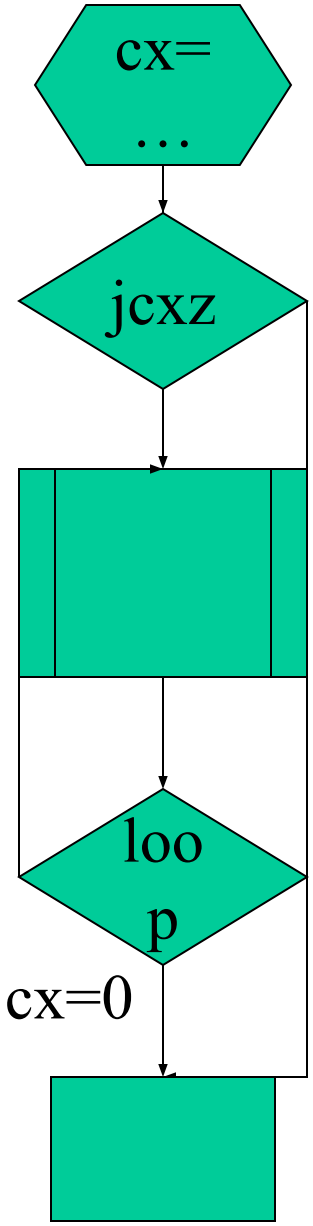
Безусловная	Взаимодействия с процедурами	Условные			Циклы
Jmp	Call	Jl=Jnge	Jc	Jnc	Loop
	Ret	Jle=Jng	Jp	Jnp	Loope
		Jg=Jnle	Jz	Jnz	Loopz
	Int	Jge=Jnl	Js	Jns	Loopne
	Iret	Jb=Jnae	Jo	Jno	Loopnz
		Jbe=Jna			
		Ja=Jnbe	Jcxz		
		Jae=Jnb	Jecxz		

«выше» - «ниже» - для чисел без знака

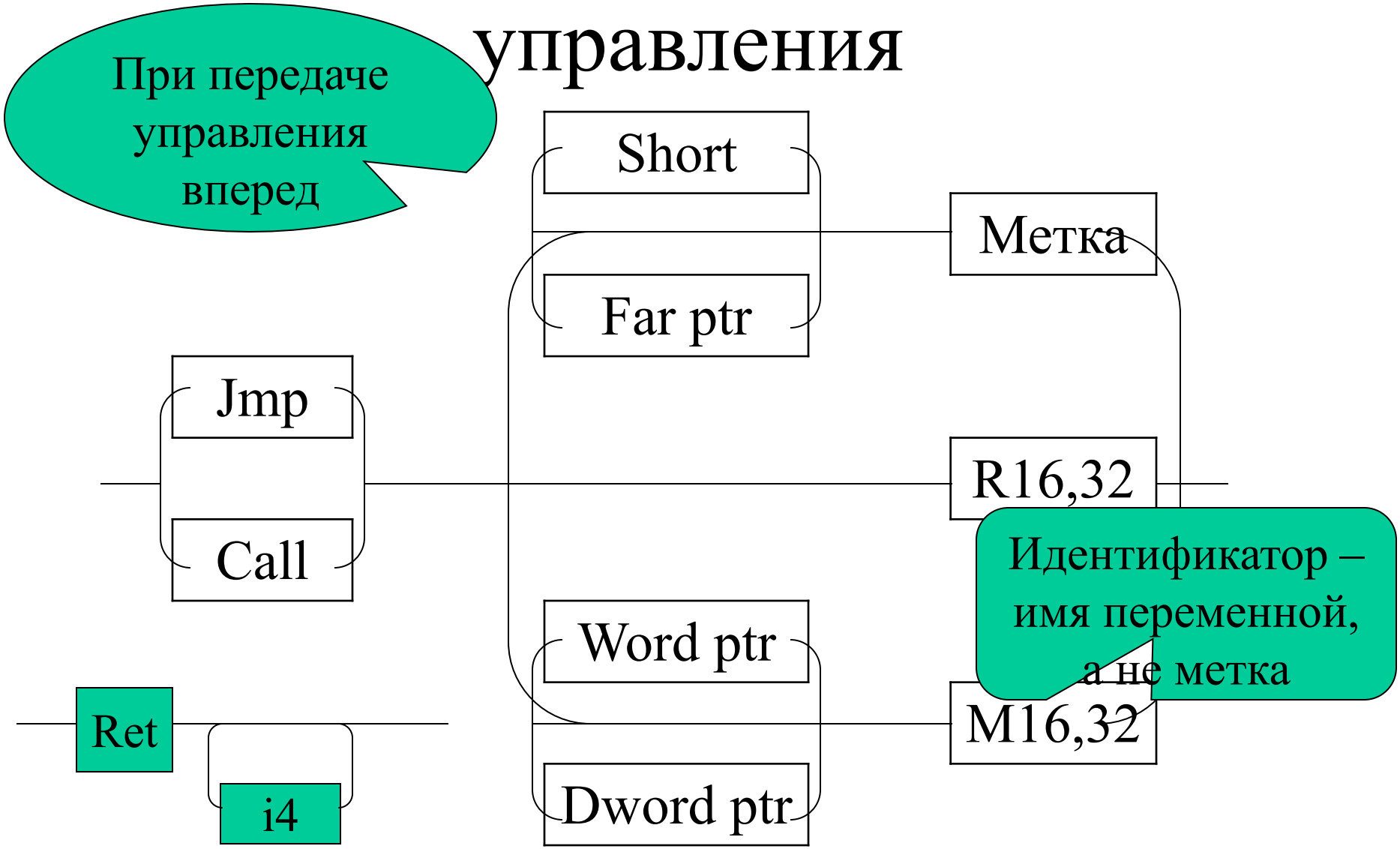
«больше» - «меньше» - для чисел со знаком

Операторы перехода по условию





Инструкции передачи управления



Инструкции работы с прерываниями

- Int i8 – вызов процедуры обслуживания прерывания с номером, заданным операндом команды:

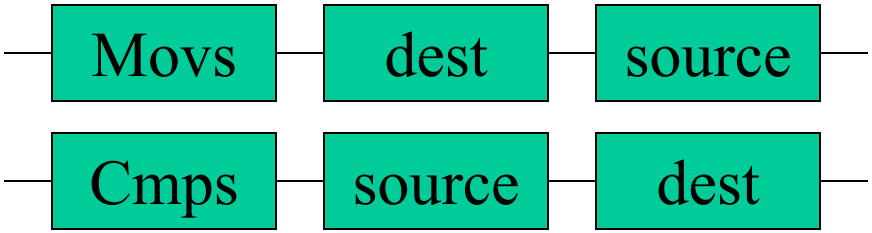
  pushf cli
  puch cs jmp …
  push ip

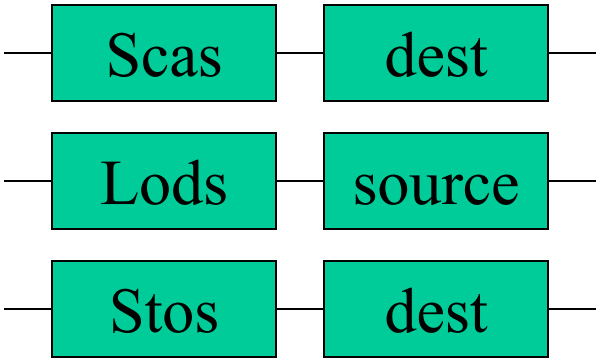
- Iret – возврат из программы обработки прерывания в прерванную программу:

  pop ip popf
  pop cs jmp …

Команды работы с цепочками

Пересылка	Сравнение	Сканирование	Загрузка	Сохранение
Movs	Cmps	Scas	Lods	Stos
Movsb	Cmpsb	Scasb	Lodsb	Stosb
Movsw	Cmpsw	Scasw	Lodsw	Stosw
Movsd	Cmpsd	Scasd	Lodsd	Stosd
Rep	Repz	Repe		
	Repnz	Repne		





	Источник	Приемник	Результат
Cmps	Ds:si	Es:di	Cf, zf, si,di ± 1
Lods	Ds:si	Eax/ax/al	Ds:[si]□al, si ± 1
	Приемник	Источник	Результат
Movs	Es:di	Ds:si	[si]□[di], si,di ± 1
Scas	Es:di	Eax/ax/al	Zf, di ± 1
Stos	Es:di	Eax/ax/al	al□[di], di ± 1

```
.model small
.stack 256
.data
   a    db '1234567890','$‘
   b    db '0987654321','$'
.code
main  proc
   assume es:@data
   mov ax,@data
   mov ds,ax
   mov es,ax
   mov ah,9
   lea dx,b
   int 21h
   lea si,a
   lea di,b
   mov cx,10
   rep movsb
   mov ah,9
   lea dx,b
   int 21h
.exit 0
main  endp
     end    main
```

Результат:

0987654321234567890

Команды управления ЦП

Флаг переноса	Флаг направления	Флаг прерываний	Регистр флагов
Stc	Std	Sti	Lahf
Clc	Cld	Cli	Sahf
Cmc			

Загрузка регистра флагов

- LAHF – в регистр AH
- SAHF – из регистра AH