

8086 → 8088

20 cimbbit

16bit

IP

AX

BX

CX

DX

AX



SP

SI

DI

BP

FLAG

CS

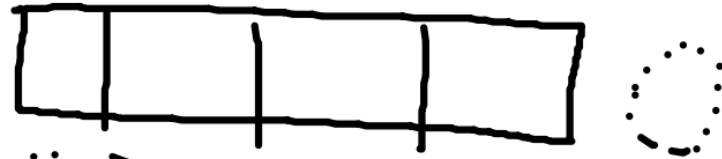
--	--	--	--

DS — || —

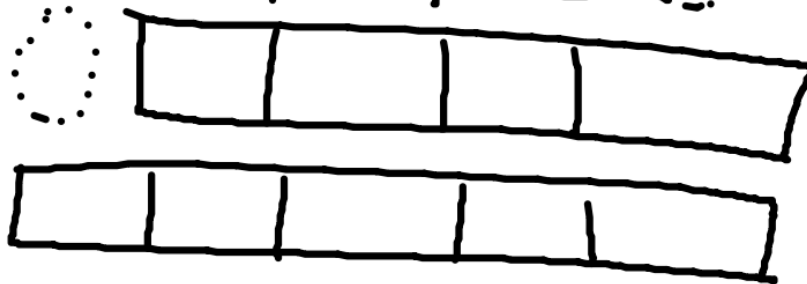
SS — || —

ES — || —

SEGMENT



OFFSET +



ADATMOZ GATÖ:

MOV

MOL

AX, BX
BH, 0

Mov BX, AX

~~Mov OIB, H~~

Mov AL, 1AH

Mov CL, 00001111B

MOL $A \times \phi B A B A H$

MOL $AL, (43-5)*2$

MOL $DH, 'Q'$

MOL $CL, 'A' - E$

MOL $Cx, 'AB'$

Mov

DX, [1234H]

Mov

[1236], AX

Mov

ES: [1236], AX

Mov

AX, [BX]

MOV [BX+1234], CL

MOV AH, [BX+DI]

PUSH word PTR [BX]

POP SI

pushf

popf

XCHG

AX, BX

IN A_x^L , $0..255$
 DX

OUT DX
 $0..255$) A_x^L

INT $0..255$
[]

INPUT

1. pushf (INT)

2. push CS

3. push IP

4. CS:IP \leftarrow újérték

1. pop IP

(IRET)

2. pop CS

3. popf

Aritmetikai utasítások

ADD

AX, 8

ADC

[BX], AL

SUB

CL, DL

SBB

AX, [1234H + BX]

INC	BX		
DEC	WORD PTR	[SI]	
CMP	AL, 'Q'		
NEG	BP		

MUL

8 bites * AL \rightarrow AX

16 bites * AX \rightarrow DX:AX

IMUL

— || —

DIV

AX: 8bit

AL halfword

DX:AX: 16bit

AH word

IDIV

— || —

AX halfword
DX word

IP-t módosító utasítások

JMP WORD PTR [BX]

JMP LOCK

JMP DWORD PTR [SI]

CALL LACIKA

RET

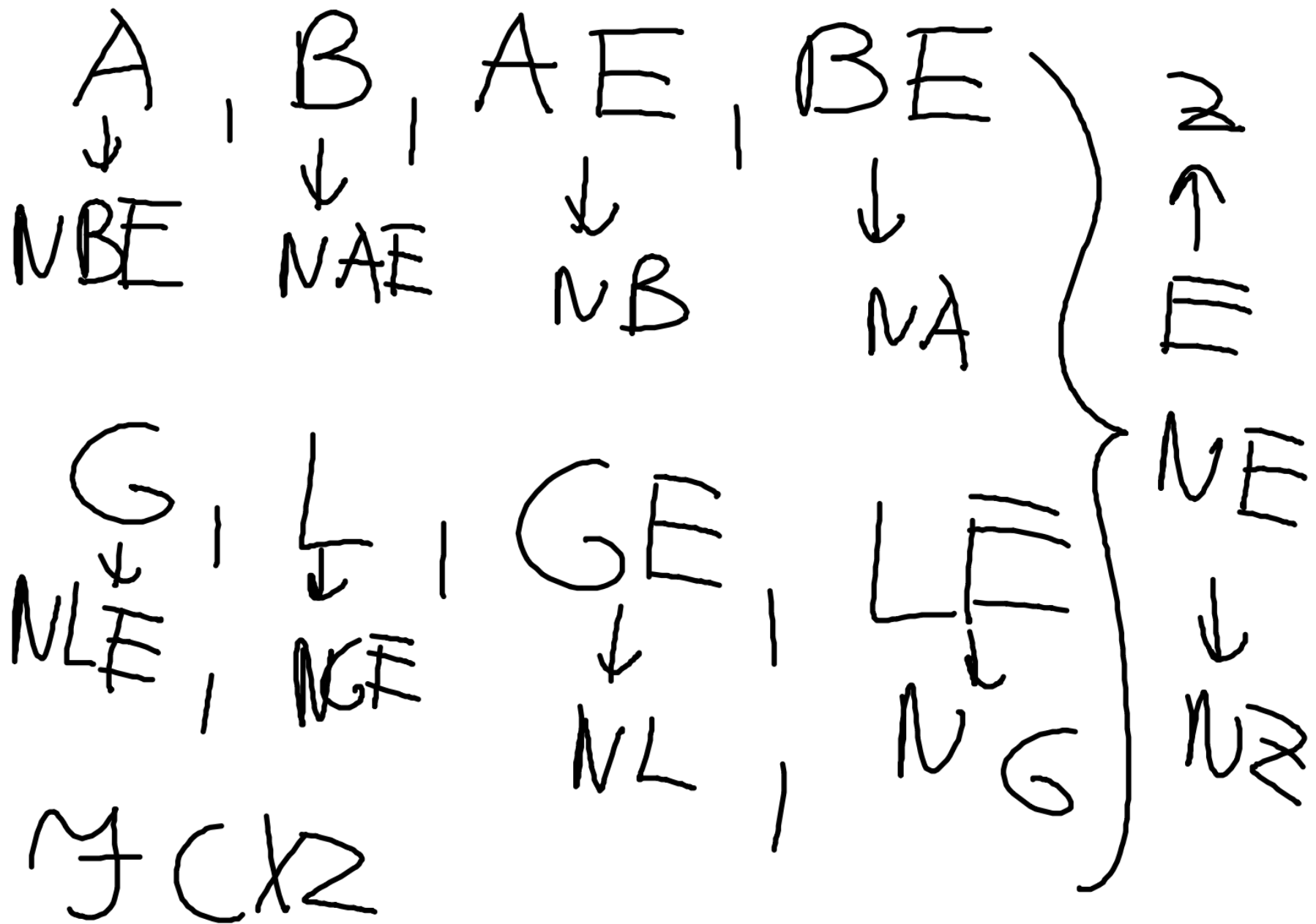
CALL DWOR PTR [EMOKE]

RET

$\mathcal{F}_{IF} \quad \underline{\quad}$

$\mathcal{F}_{IF} NC \quad \underline{\quad}$

$\mathbb{Z}, \mathbb{N}, \mathbb{Z}, S, NS, PO, PE$
 $\quad \quad \quad \downarrow \quad \downarrow$
 $\quad \quad \quad NPE \quad NPO$



Loop

C1C1C1: MOV CX, 10

⋮

Loop

C1C1C1

MOV SI, 10

VISSLA:

⋮

DEC SI

JNZ VISSZA

7C MESSZE : JNC K02
JMP MESSZE

K02:

MESSZE:

KULSO: MOV CX, 100

BELSO: MOV CX, 20
;
LOOP BELSO
LOOP KULSO

KULSO: MOV CX, 100

⋮

PUSH CX

BELSO: MOV CX, 20

⋮

LOOP BELSO

POP CX

LOOP KULSO

~~CALL BEND~~
~~MOV AX, DX~~
~~INC DX~~
BEND
RET

SLOVEG DB 'VALAMIG'

BENDE:

RET

LOGIKAI UTASÍTÁSOK

AND	CX, 3
OR	[BX], SI
XOR	AL, 00001111B
TEST	BX, DX
NOT	AL

$$\begin{array}{r}
 01011100 \\
 \text{AND } 00111100 \\
 \hline
 00011100
 \end{array}$$

$$\begin{array}{r}
 01011100 \\
 \text{XOR } 00111100 \\
 \hline
 01100000
 \end{array}$$

$$\begin{array}{r}
 01011100 \\
 \text{OR } 00111100 \\
 \hline
 01111100
 \end{array}$$

$$\begin{array}{r}
 01011100 \\
 \text{NOT} \\
 \hline
 10100011
 \end{array}$$

$$|0\rangle - 0 = |0\rangle$$

$$|1\rangle - 1 = |0\rangle$$

$$\vdots \quad \vdots \quad \vdots$$

$$|g\rangle - g = |n\rangle$$

MOV AH, 2

MOV DL, CL

ADD DL, 'Ø'

INT 21H

CX Kiirattasa
decimälisan

MOV AX, CX

MOV SI, 10

XOR DX, DX

DIV SI

MOV AX, CX

XOR CX, CX

PUSH CX

MOV SI, 10

KIKL:

XOR DX, DX

DIV SI

ADD DL, 01

PUSH DX

OR AX, AX

KIKL

UCHAR:	MOV	AX, 2
	POP	DX
	OR	DL, DL
	JZ	MARK1
	INT	21H
	JMP	UCHAR
MARK1:		

0011 0000 → 30H

0011 0001

101

111

⋮

Decimális érték beolvasás

VSLEUT:
XOR
MOV

DI, DI

AH, 8

INT

21H

CMP

AL, 13

JZ

KESZ

CMP

7B

CMP

YA

MOL

MOL

,NT

AL,'O'

HIBA

AL,'g'

HIBA

OL,AL

AH,2

21H

AND

DX, 00001111B

MOV

AX, DX

XCHG

AX, DI

MOV

CX, 10

MUL

CX

ADD

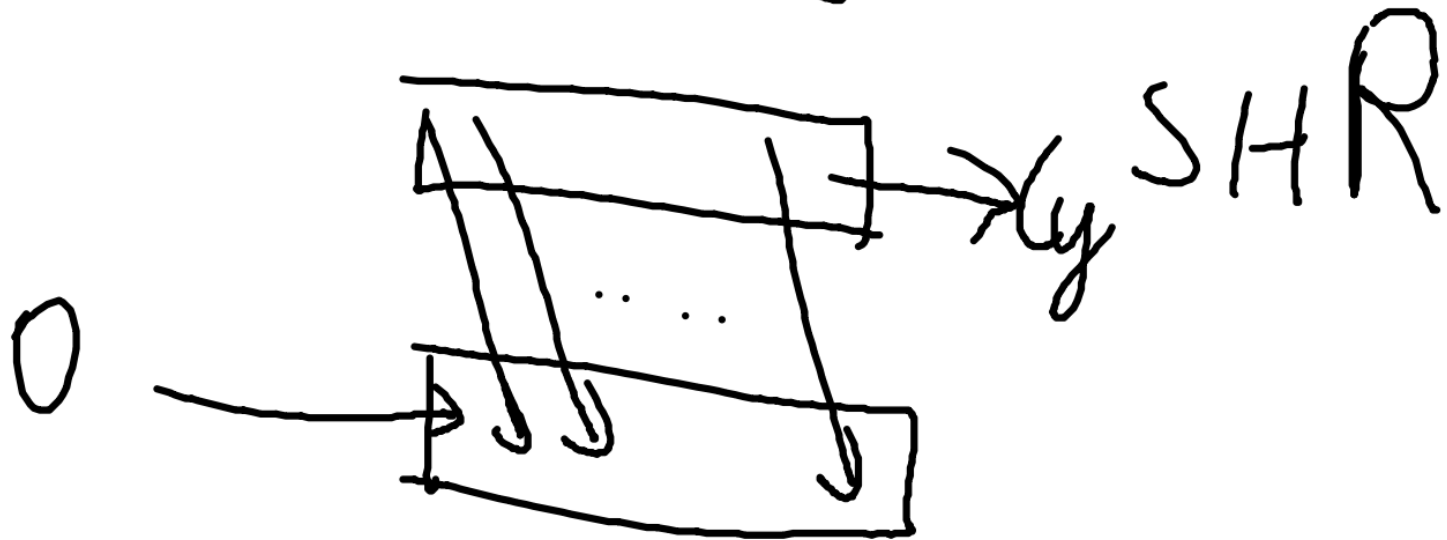
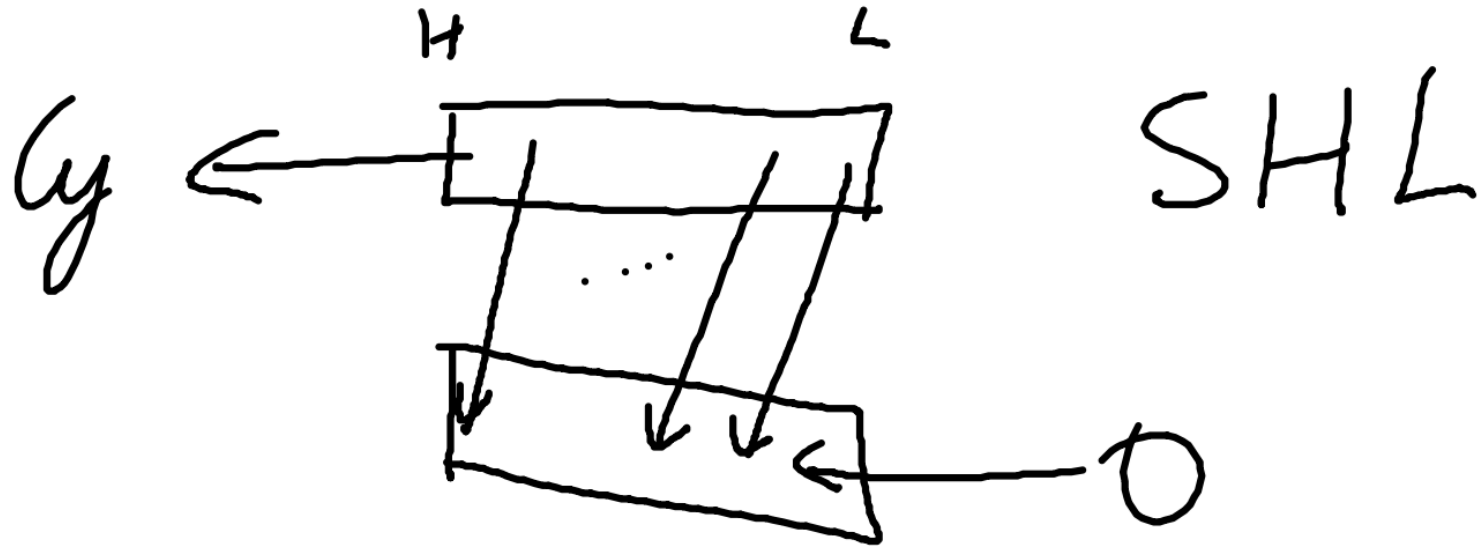
DI, AX

JMP

EXIT

RES:

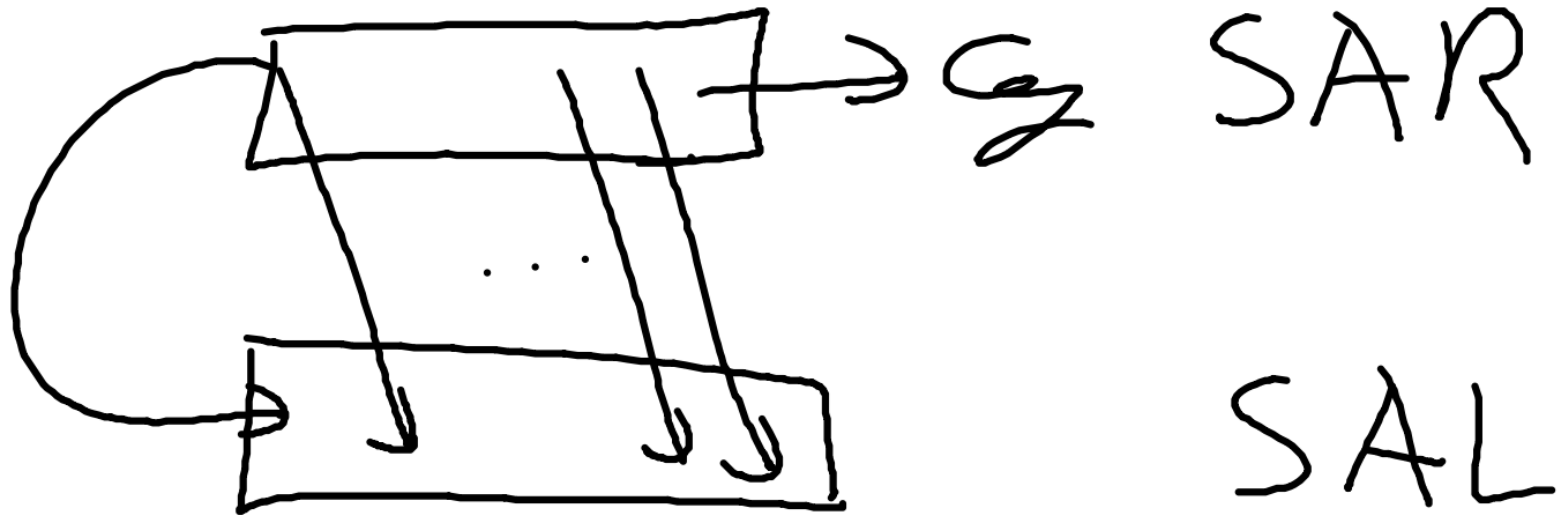
Léptető, forgató utasítások



SHR AX, 1

SHR AX, CL

SHL DWORD PTR [DI], 1



SHL

MOV

SHL

SHL

ADD

DI, 1

SI, DI

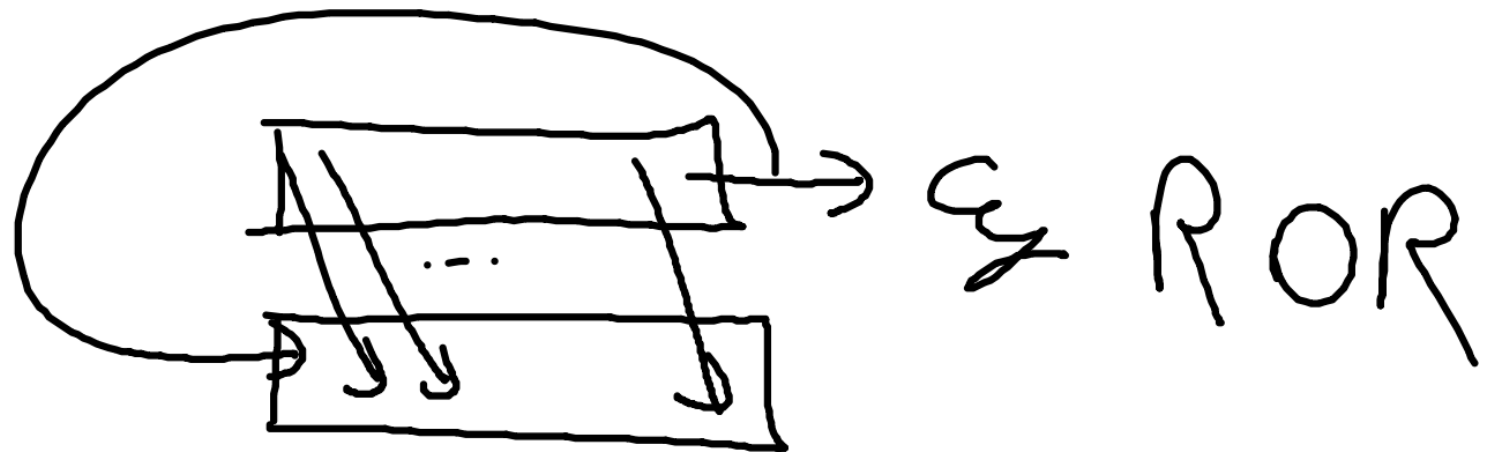
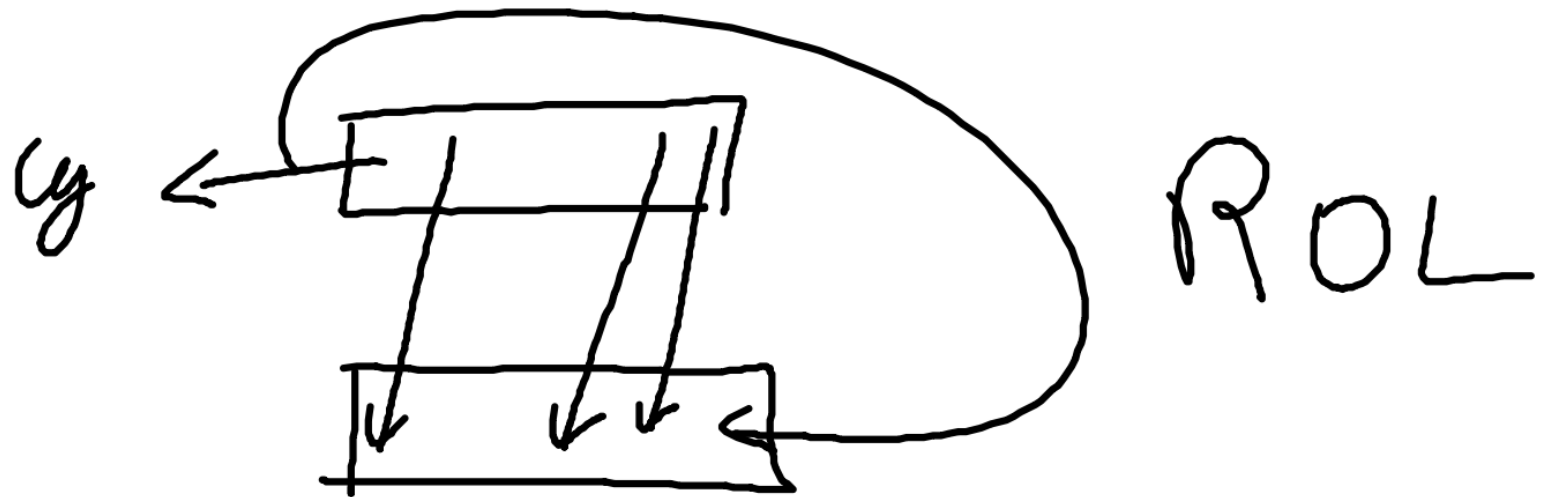
DI, 1

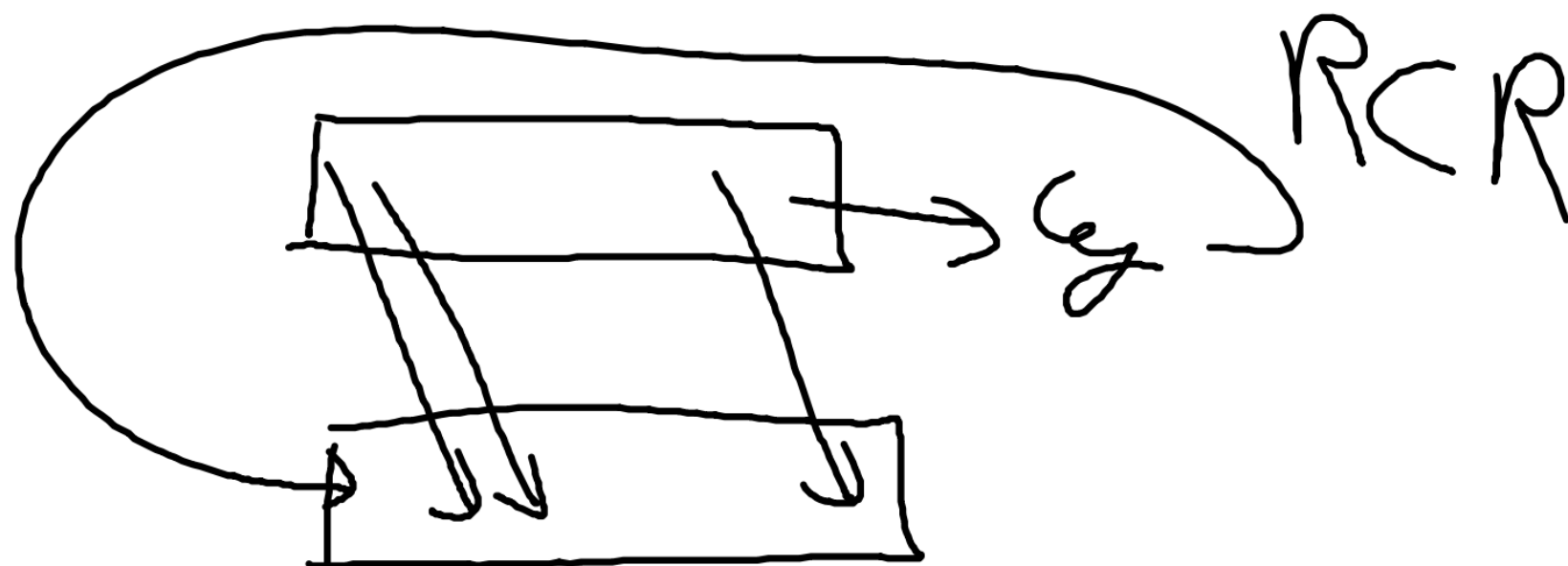
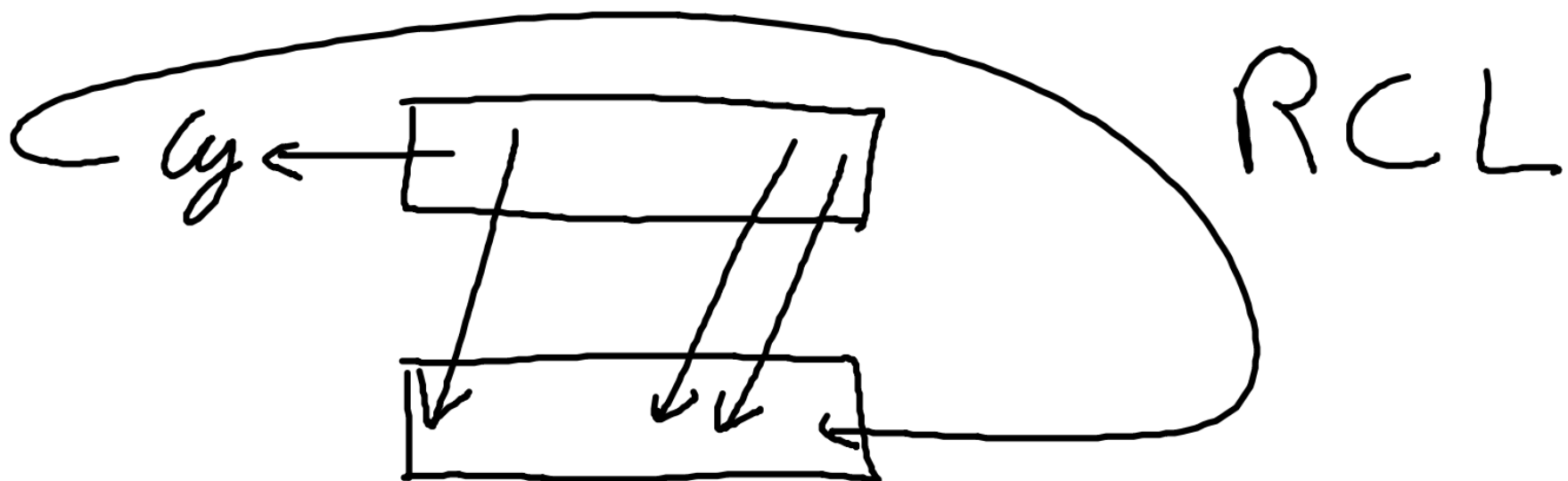
DI, SI

i * 2

i * 4

i * 8





31 16

15 8



② RCL

④ SHR

15 0

7 0



⑦ SHL

② RCR

EGYEB

NOP

CLC

STC

CMC

HLT

CLI

STI

CLE

STD



MOV SI, OFFSET INNEN

MOV DI, OFFSET IDE

MOV CX, TERMER

MACIK:

MOV AL, [SI]
MOV [DI], AL
INC SI
INC DI
LOOP MACIK

} MOVSB
DI
ESI
SZERINT

$\text{MOVSB} \rightarrow \begin{matrix} \text{SI} \leftarrow \text{SI} \pm 1 \\ \text{DI} \leftarrow \text{DI} \pm 1 \end{matrix}$
 $\text{MOVSW} \rightarrow \begin{matrix} \text{SI} \leftarrow \text{SI} \pm 2 \\ \text{DI} \leftarrow \text{DI} \pm 2 \end{matrix}$

REP

MOV SI, ...,
 MOV DI, ...,
 MOV CX, ...,
 CLD
 REP MOVSB

STOS^B_W

AL
AX} → ES:[DI]

LODS^B_W

DS:[SI] → { AL
AX

CMPS^B_W

CMP DS:[DI], ES:[DI]

SCAS^B_W

CMP AL
AX} ES:[DI]

REPZ, REPE

REPZ, REPNZ

```
MOV AX, 8  
CALL FAKT ; DX:AX
```

$$N! = (N-1)! \cdot N$$

FAKT:

CMP

AX, 1

JNZ

NEMEGY

XOR

DX, DX

RET

NEMEGY:

PUSH

AX

DEC	AX
CALL	FAKT
JC	HIBA
Mov	SI, AX
Mov	AX, DX
POP	BX

MUL BX

JC HIBA

XCHG AX, SI

MUL BX

ADD DX, SI

HIBA:

RET

INT 16H

(AH) 0 → leütés várás
↓
AL ← leütés kódja

1 → leütés vizsgálata

2 esetén nincs leütés
NZ -11- van, kódja \rightarrow AL

MOV AH, 1

INT 16H

JZ nincs

MOV AH, 0

INT 16H

2 → SHIFT status

MEGVAR:

beolvasás

MOV AH, 2

INT 16H

AND

AL, 00000011B

CMP

AL, 3

JNZ

MEGVAR

INT 33H

AX \rightarrow 0 Utezik-e a driver
↓

AX = 0 \rightarrow nin(S)
AX \neq 0 \rightarrow von } driver

$AX \leftarrow 1$ Show mouse cursor

$AX \leftarrow 2$ Hide mouse cursor

$AX \leftarrow 3$ Beolvassza a állapot

$BX \leftarrow$ gombok bitjei

$CX \leftarrow X$
 $DX \leftarrow Y$ } pozíció 8!

ITTVAR:

MOV AH, 1

INT 16H

JZ EGEROL

MOV AH, 0

INT 16H

CMP AL, 27

JZ KILERP

EGEROL:

MOV AX, 3

INT 33H

AND BL, 00000011B

MOV

DX, OFFSET EGYSEN

JZ

TXTK11

MOV

DX, OFFSET BALGOM

DEC

BL

JZ

TXTK11

MOV DX, OFFSET 70B60M

DEC BL

JZ TXTK11

MOV DX, OFFSET MINDE

TXTK11:

MOV AH, 9

INT 21H

JMP ITTVAR

EGYSEM DB 'EGYKSEM'
DB '9 DUP(8)' '\$'

BALGOM DB 'BALGOMB'
DB '9 DUP(8)' '\$'

JOBGOM DB 'JOBGB'
DB '9 DUP(8)' '\$'

MINDKE

DB

'MINDKETTŐ'

DB

{ DLP(8), '8' }

//

8, 8, 8, 8, 8, 8, 8, 8

MOV SI, 0FFH

:

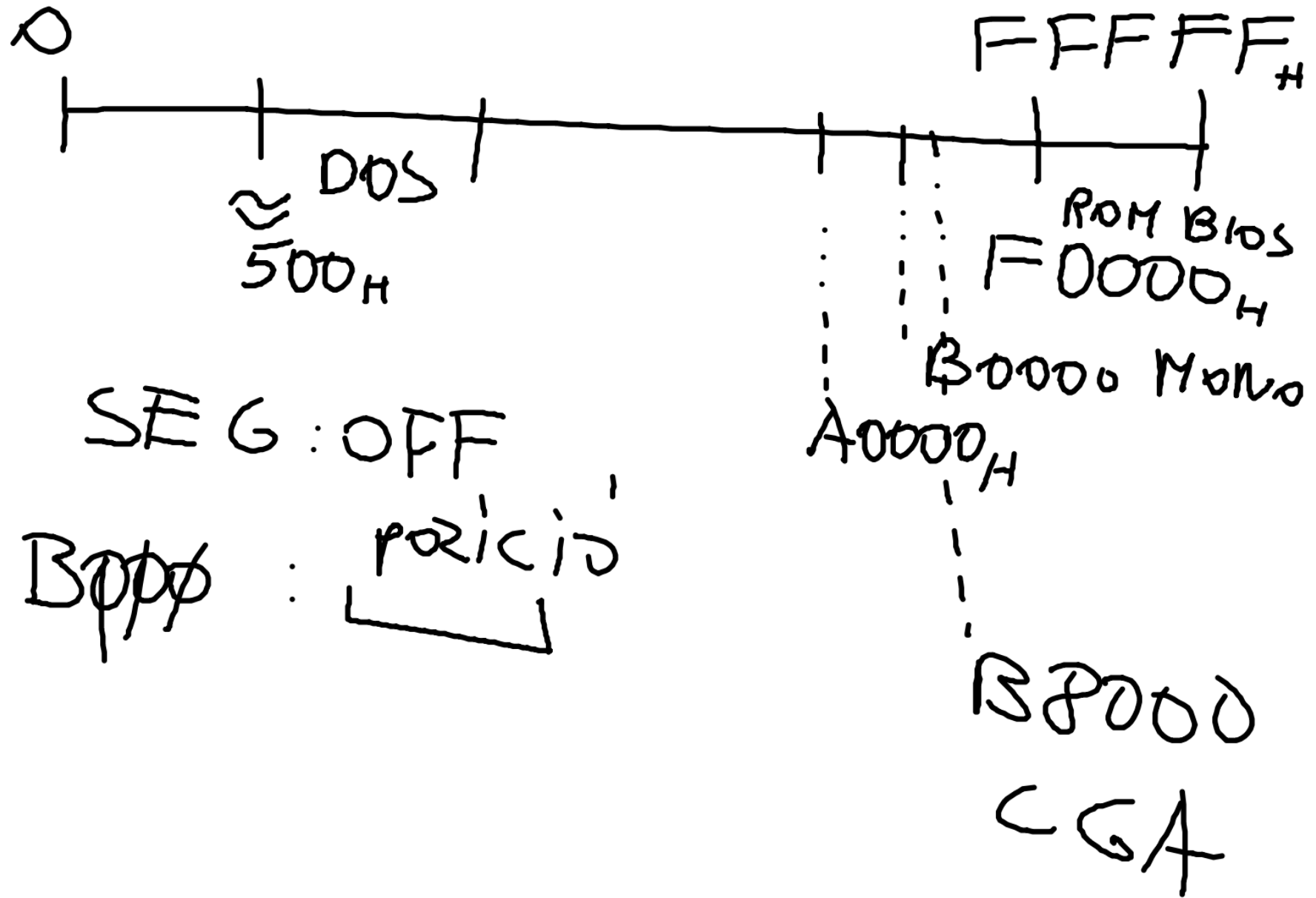
CMP SI, BX

JZ ITVAR

MUL SI, BX

AND BX, ...

DEC BX



00 (01)	02 (03)
160 (161)	

158 (159)

Statusz bajt

7



Villags

Hatter
szín

karakter
szín

CM → BX

MOV AX, 0B800H ; CGA

MOV ES, AX

XOR

DI, DI

PÄIKL:

MOV AL, [BX]

OR

AL, AL

42

KITEVE

.
| ES ← CGA
| OFFSET
| Kiteendõ
|. Nollag?
| itenivege

	INC BX	; következőre
STOSB	Mov ES:[DI], AL	; be másolom
	INC DI	az állapotokra
	INC DI*	; következőre
	JMP PACIKL	vissza beol

KITEVE:

* ADD DI, 2

EIÖHe !!

CLD

MOV AH, ES: [1]

:

STOSW

GRAFIKUS ÜZEM:

320X200 pont

A000 Szegmens

Mód → 13H

MOV AX, 13H

INT 10H

10 címek:

Szinszám → 3C8H

Összetevők → 3C9H

1-es zöld keyen
MOV DX, 3C8H

MUL AL, 1

OUT DX, AL

INC DX

MUL AL, 0

OUT DX, AL

MUL AL, 63

OUT DX, AL

```
MOV    AL, 0
OUT    DX, AL
```

```
Hdttér    leyyen kdl
MOV        DX, 3C8H
MOV        AL, 0
OUT        DX, AL
INC        DX
OUT        DX, AL
OUT        DX, AL
MOV        AL, 63
OUT        DX, AL
```

IRQ0 → Timer INT8

IRQ1 → kbu; INT9

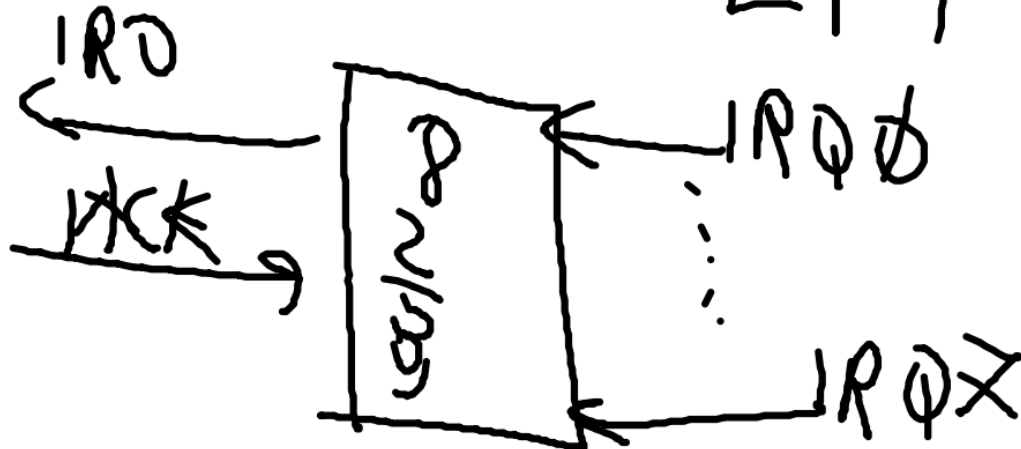
⋮

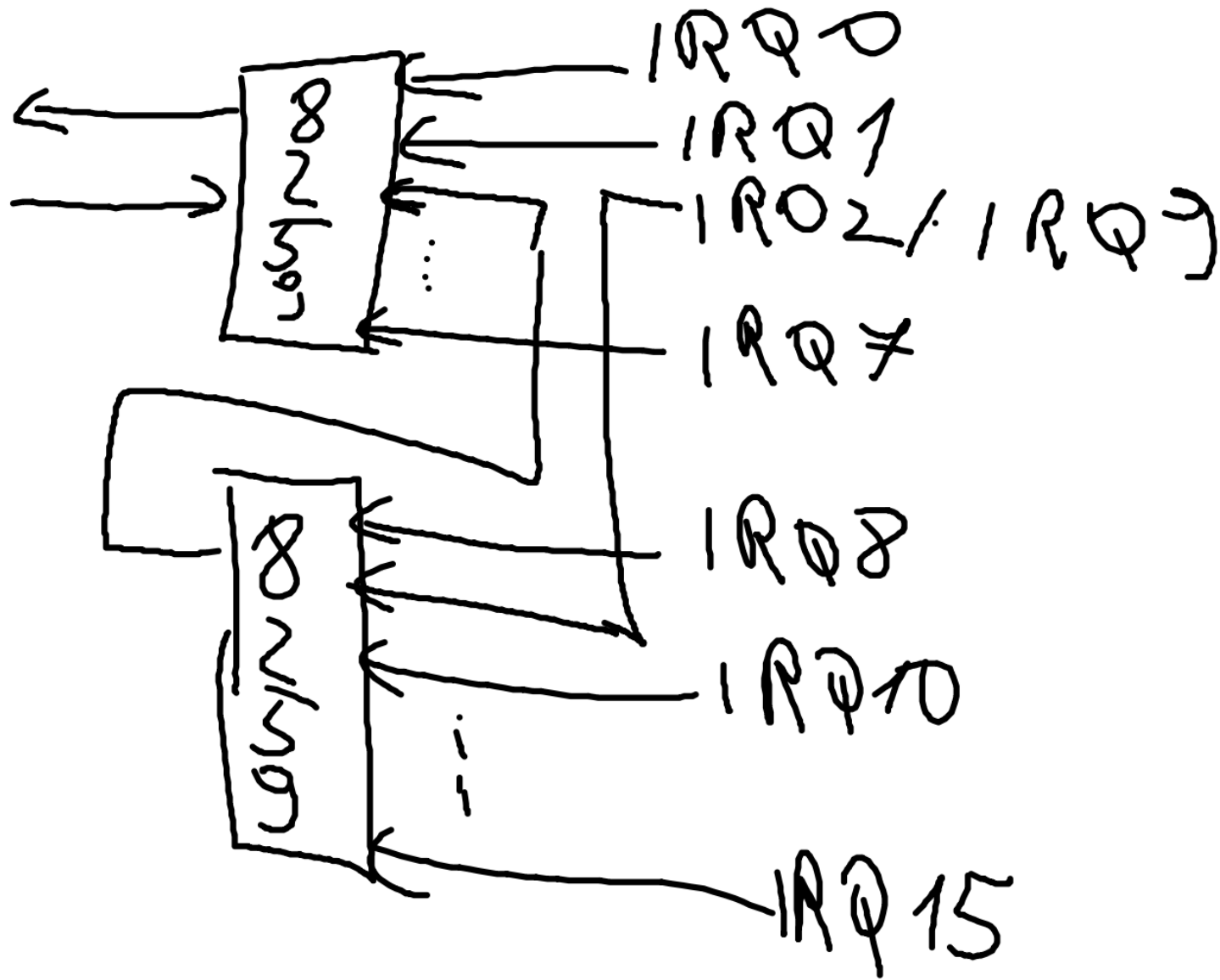
⋮

IRQ7

LPT

INT OF





①.

⋮
IRET

②A

CALL EREREDET)

⋮
IRET

2B

⋮

JMP #REDT1

XOR AX, AX

MOV ES, AX

MUL AX, ES:[8*4]

MUL DX, ES:[(8*4)+2]

MUL

WORD PTR[C1M], AX

MOV

WORD PTR[C1M+2], DX

CL)

MOV ES: [8*4], OFFSET SAYI1

MOV ES: [8*4+2], CS

STI

...

→ RESIDENS KILİPİS

SABATI:

PUSHF

CALLS: [CIM]

...

IRET

CIM

DD 2

SAMAT:

⋮

JMP CS:[C_M]

C_{1M}

DD ?