

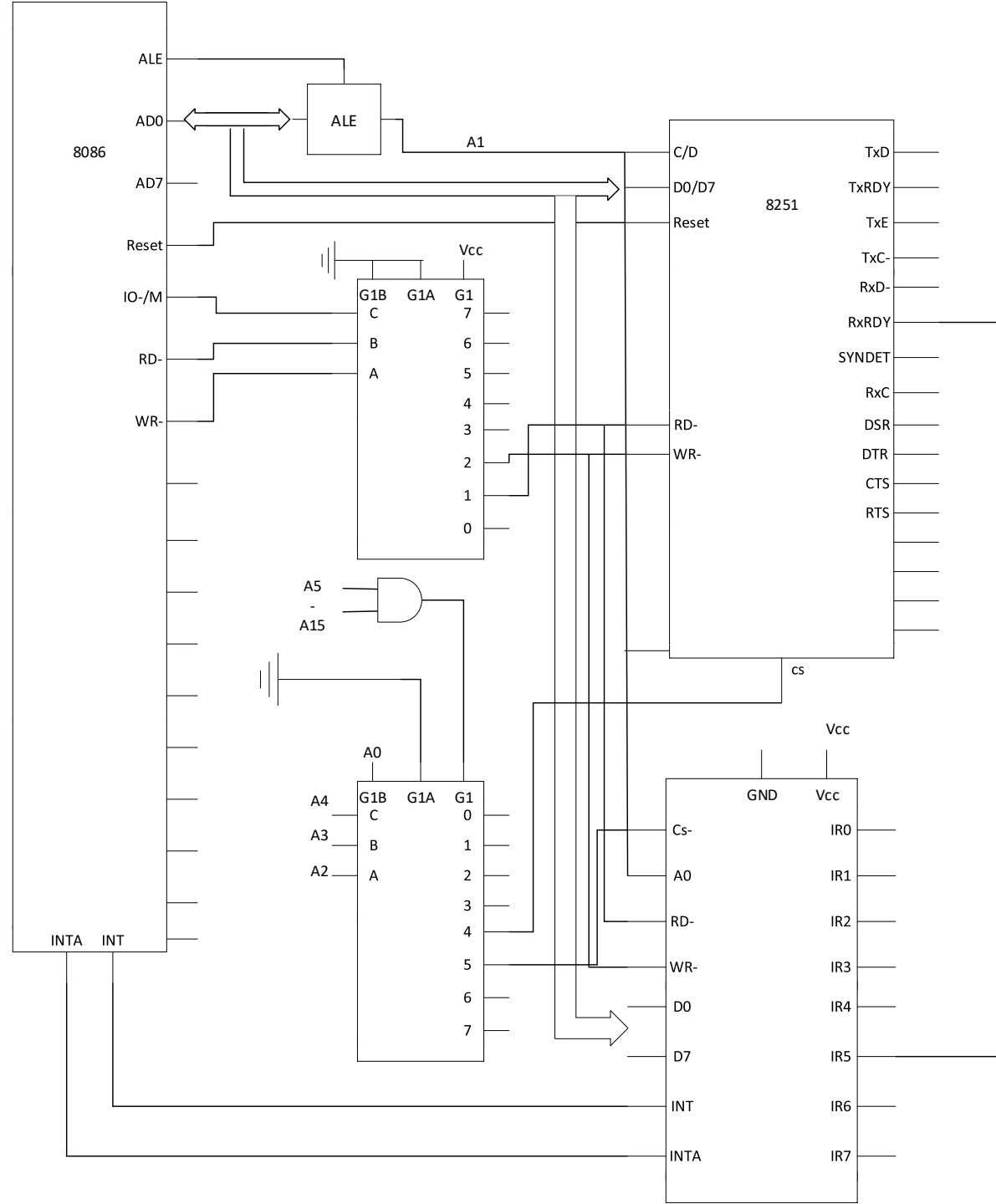
# Zadatak 3

- Za mikroprocesor iAPX 8086 projektovati mikroračunarski sistem za predaju i prijem podataka koristeći komponentu 8251A. Podaci se šalju od adrese SEND, a primaju od adrese REC. Broj podataka za predaju i prijem je 64. Komponentu treba isprogramirati za istovremeni prijem i predaju 5-bitnih podataka sa parnim bitom parnosti, 1.5 stop bitom i brzinom 16x. Predaju organizovati ispitnom petljom, a prijem preko prekida 173, koji se generiše na priključku RxRDy. Komponenta 8251 nalazi se na U/I adresi 0xFFFF0.

# Adrese

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	8251 (FFF0)
1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	0	8251 (FFF2)
1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	8259 (FFF4)
1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	0	8259 (FFF6)

# Šema povezivanja



# Mode word (8251)

- 0xB2

S2	S1	EP	PEN	L2	L1	B2	B1
Broj stop bitova: 01 – 1 stop bit 10 – 1.5 stop bit 11 – 2 stop bita		Bit parnosti: 00 – disable 01 – neparna 10 – disable 11- parna		Broj bitova po karakteru: 00-5b 01-6b 10 -7b 11-8b		Baud rate factor: 00-Sync 01-1x 10-16x 11-64x	
1	0	1	1	0	0	1	0

# Command word (8251)

- 0x15

EH	IR	RTS	ER	SBRK	RXE	DTR	TXEN
0:Normal 1:Hunt mode	0:Normal 1:Internal reset	0:DTR->1 1: DTR->0	0:Normal 1: Reset error flag	0:Normalan režim 1:Slanje karaktera prekida	Omogućiti prijemnik	0:DTR->1 1: DTR->0	Omogućiti predajnik
0	0	0	1	0	1	0	1

# ICW1

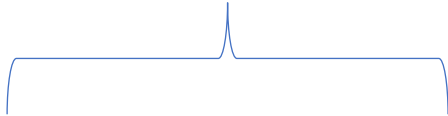
- 0x13

A7	A6	A5	1	LTIM	ADI	SNGL	IC4
A7-A5 adrese vektora interapta (samo za 80/85 mod) Za 8086 nebitno – stavljamo sve 0				0:Okidanje na ivicu (uzlaznu) 1:Okidanje na nivo (jedinica)	Nije relevantno za 8086	0:više od 1, kaskadno 1: jedan 8259	0: ICW4 nije potreban 1: ICW4 potreban
0	0	0	1	0	0	1	1

# ICW2

- 173 = 10101101(bin)
- 0xA8

Nije relevantno u toku  
inicijalizacije



Interrupt	A7	A6	A5	A4	A3	D2	D1	D0
IR0	1	0	1	0	1	0	0	0
IR1						0	0	1
IR2						0	1	0
IR3						0	1	1
IR4						1	0	0
IR5						1	0	1
IR6						1	1	0
IR7						1	1	1

# ICW4

- 0x03

D7	D6	D5	SFNM	BUF	M/S	AEOI	mPM
0	0	0	0: NOT SPECIAL FULLY NESTED 1: SPECIAL FULLY NESTED	0X – NON BUFFERED 10 – BUFFERED MODE/SLAVE 11 – BUFFERED MODE/MASTER		0: NORMAL 1: AUTO	0: MCS-80/85 MODE 1: 8086/8088 MODE
0	0	0	0	0	0	1	1



# OCW1

- 0xDF
- 173 = 10101101(bin)
- 168+5=173

M7	M6	M5	M4	M3	M2	M1	M0
1	1	0	1	1	1	1	1

# Program – inicijalizacija i konfiguracija

```
EXTRN PREKID:FAR ;pomocna procedura za obradu prekida
PUBLIC RECV, BRULAZ ;vidljivo i iz drugih modula
```

```
DATA SEGMENT
```

```
    BRULAZ DB 0
```

```
    ;baferi primljenih i poslatih podataka
```

```
    SEND DB 64 DUP (3Fh)
```

```
    RECV DB 64 DUP (?)
```

```
DATA ENDS
```

```
STEK SEGMENT
```

```
    BOS DW 256 DUP (?)
```

```
    TOS LABEL WORD
```

```
STEK ENDS
```

```
CODE SEGMENT
```

```
ASSUME CS:CODE, DS:DATA, SS:STEK
```

```
START:
```

```
    ;inicijalizacija segmenata
```

```
    MOV AX, DATA
```

```
    MOV DS, AX
```

```
    MOV AX, STEK
```

```
    MOV SS, AX
```

```
    LEA SP, TOS
```

```
    ;inicijalizacija komponenti
```

```
    CLI
```

```
    ;8251
```

```
    MOV AL, 00h
```

```
    MOV DX, FFF2h
```

```
    OUT DX, AL
```

```
    OUT DX, AL
```

```
    OUT DX, AL
```

```
    MOV AL, 40h
```

```
    OUT DX, AL
```

```
    MOV AL, B2h ;slanje mode kontrolne reci
```

```
    OUT DX, AL
```

```
    MOV AL, 15h ;slanje komandne kontrolne reci
```

```
    OUT DX, AL
```

```
    ;8259
```

```
    MOV DX, FFF4h
```

```
    MOV AL, 13h ;ICW1, A0=0
```

```
    OUT DX, AL
```

```
    MOV DX, FFF6h
```

```
    MOV AL, A8h ;ICW2
```

```
    OUT DX, AL
```

```
    MOV AL, 03h ;ICW4
```

```
    OUT DX, AL
```

```
    MOV AL, DFh ;OCW1
```

```
    OUT DX, AL
```

```
    ;inicijalizacija tabele vektora prekida
```

```
    MOV AX, OFFSET PREKID
```

```
    MOV [173*4], AX
```

```
    MOV AX, SEG PREKID
```

```
    MOV [173*4+2], AX
```

```
    MOV CX, 64
```

```
    MOV SI, 0
```

```
    STI ;dozvola prekida
```

1kB Interrupt  
Vector Table

CSi

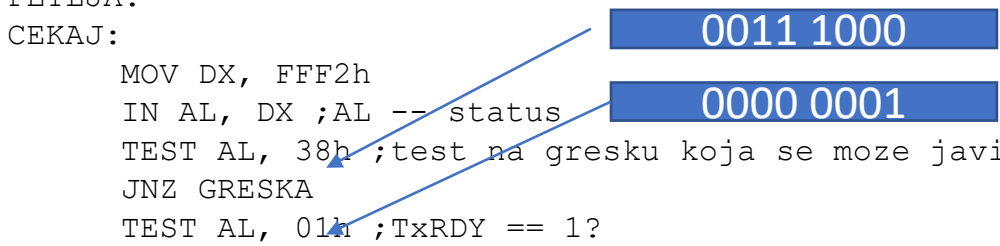
IPi

Zapis i

# Program - glavna petlja

```
PETLJA:
CEKAJ:
    MOV DX, FFF2h
    IN AL, DX ;AL -- status
    TEST AL, 38h ;test na gresku koja se moze javiti u prijemu
    JNZ GRESKA
    TEST AL, 01h ;TxRDY == 1?
    JZ CEKAJ ;ako nije spreman, cekaj da bude
    MOV DX, FFF0h
    MOV AL, SEND[SI]
    OUT DX, AL
    INC SI
    LOOP PETLJA ;64 prolaza
    CEKAJ1: ;cekanje na prekidnu proceduru
        MOV DX, 0FFF2h ;testiranje na gresku
        IN AL, DX
        TEST AL, 38h
        JZ GRESKA
        CMP BRULAZ, 64
        JL CEKAJ1

    GRESKA:
        ;ovde ide kod za obradu greske
    KRAJ:
    MOV AH, 4Ch
    INT 21h
CODE ENDS
END START
```



# Eksterna procedura za prekide

```
;Radi paralelno i vrsi prijem podataka
EXTRN RECV:BYTE, BRULAZ:BYTE
PUBLIC PREKID
PROCED SEGMENT
PREKID PROC FAR
ASSUME CS:PROCED
    ;cuvanje konteksta procesora
    PUSHF; automatski se stavlja na stek
    PUSH AX
    PUSH DI
    PUSH DX
    MOV DX, FFF0h ;bez testiranja na spremnost jer je RxRDY sigurno 1
    IN AL, DX
    MOV DI, BRULAZ
    MOV RECV[DI], AL
    INC BRULAZ ;uvecanje brojaca primljenih
    ;vracanje konteksta procesora
    POP DX
    POP DI
    POP AX
    POPF
    IRET ;vracanje iz procedure
PREKID ENDP
PROCED ENDS
```