



Proiectarea cu Micro-Procesoare

Lector: Mihai Negru

An 3 – Calculatoare și Tehnologia Informației

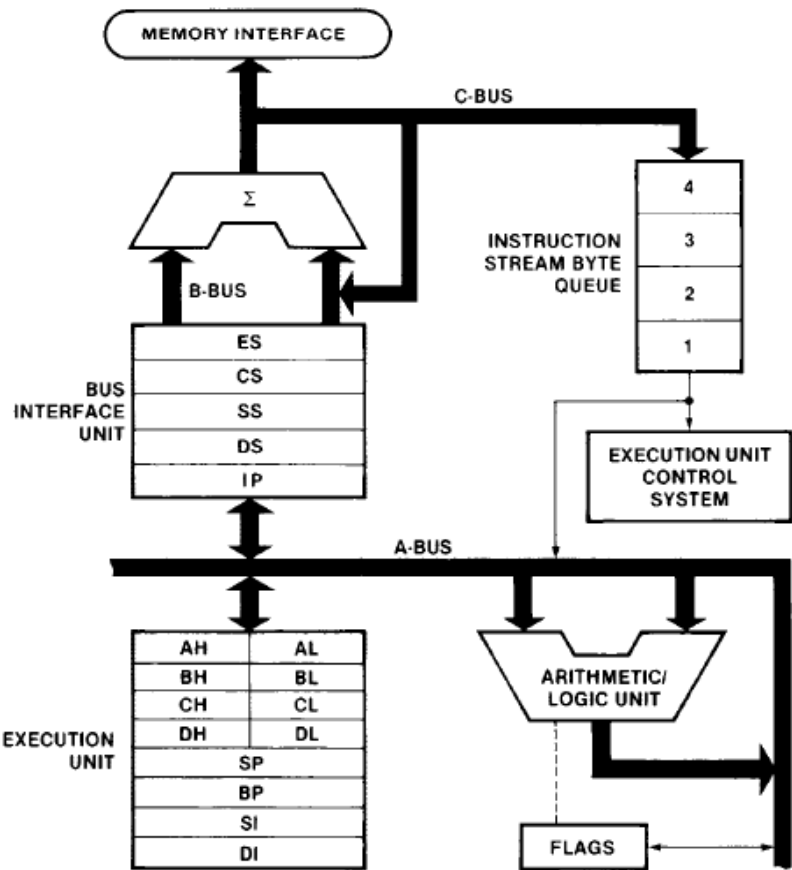
Seria B

Curs 10: Intel 8086 – I/O și întreruperi

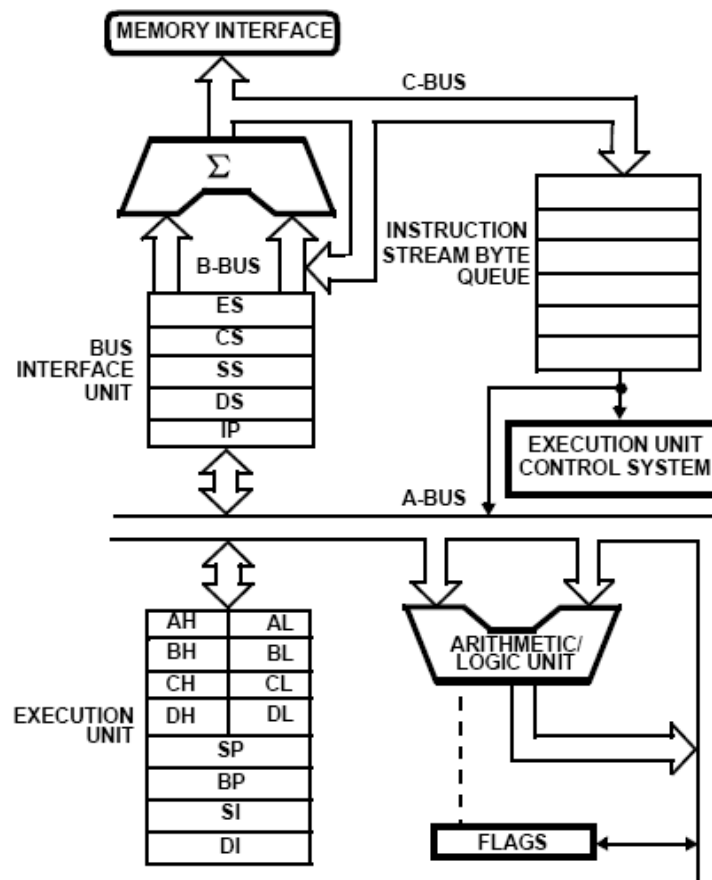
<http://users.utcluj.ro/~negrum/>



Diagrame bloc 8088 si 8086



8088



8086

Pipeline cu două etaje, EU (execution unit) și BIU (Bus Interface Unit) – decuplate printr-un FIFO (4/6 bytes)

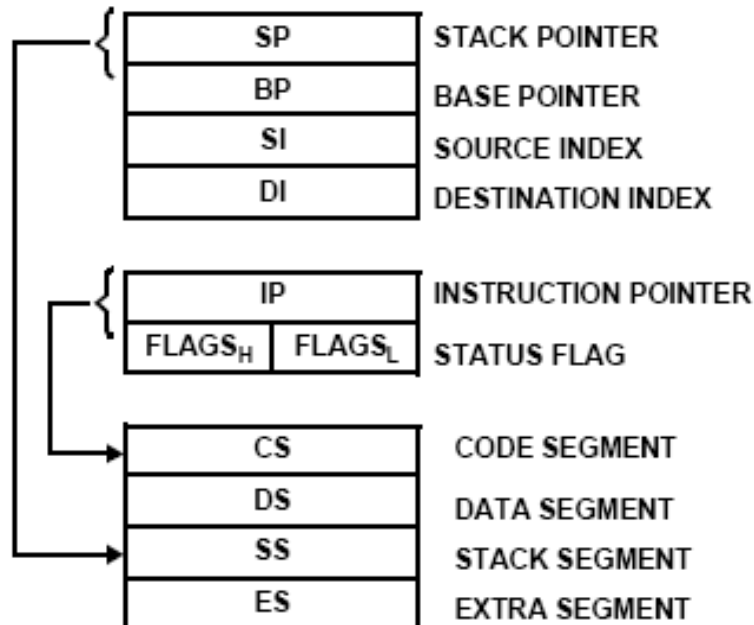


Regiștri 8086



AX	AH	AL	ACCUMULATOR
BX	BH	BL	BASE
CX	CH	CL	COUNT
DX	DH	DL	DATA

Regiștri de uz general (GPR)



Regiștri pointer

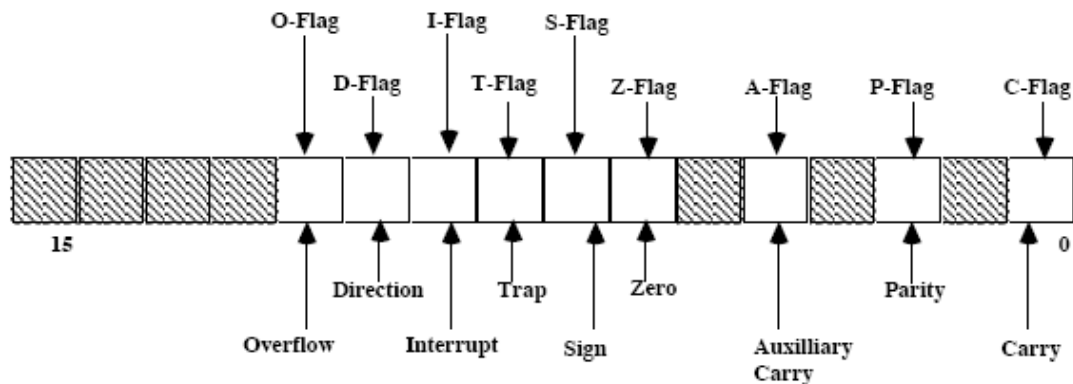
Regiștri segment



Registrul de stare (Flags)



- Overflow Flag (OF) – dacă rezultatul unei operații este prea mare (ca pozitiv) sau prea mic (ca negativ) pentru a putea fi reprezentat în locația destinație
- Direction Flag (DF) – dacă este '1', operațiile cu șiruri vor decrementa automat regiștrii index. Dacă e '0', indecșii se vor autoincrementa.
- Interrupt-enable Flag (IF) – setarea acestui bit activează întreruperile mascabile.
- Single-step Flag (TF) – Generează o întrerupere după fiecare instrucțiune dacă e setat. Folosit cu programele de tip debugger (Trap Flag).
- Sign Flag (SF) – Este setat dacă rezultatul unei operații e negativ.
- Zero Flag (ZF) – Setat dacă rezultatul unei operații este zero.
- Auxiliary carry Flag (AF) – Setat dacă se produce carry între biții 0-3 și biții 4-7 din AL
- Parity Flag (PF) – dacă numărul biților cu valoare '1' din rezultatul unei operații este par, are valoarea '1'.
- Carry Flag (CF) – dacă se produce carry în urma unei operații.

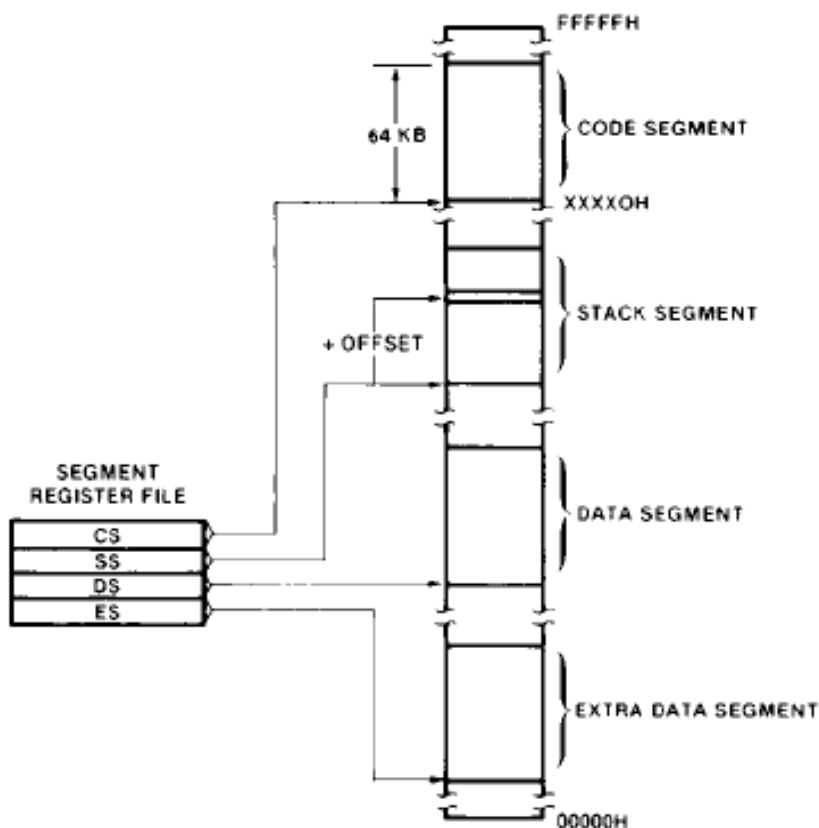




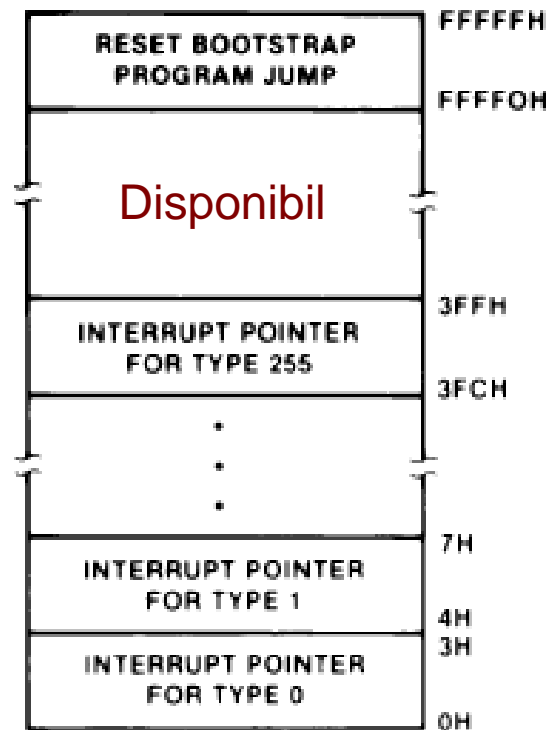
Spații de adresă – Memorie



- Memoria totala adresabilă: 1 MB
 - 20 linii de adresă
- Adresare la nivel de byte



Adresa cod RESET
CS: FFFFh, IP: 0000h



Adrese ISR, de la adresa 00000h
4 bytes / tip întrerupere,
256 tipuri întrerupere posibile



Moduri de adresare – Memorie

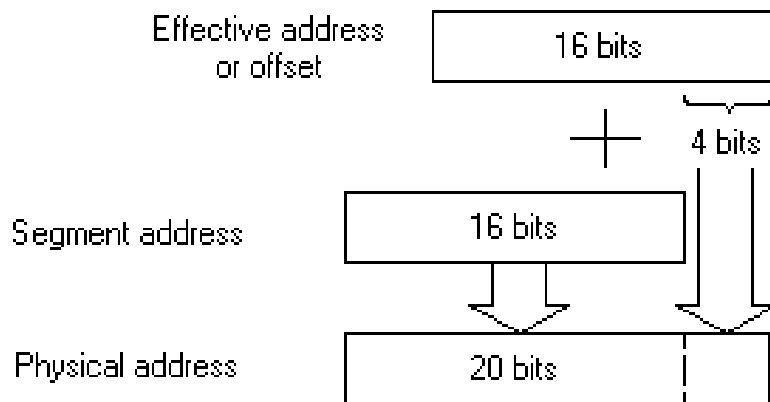


- Adresă efectivă = Bază + Index + Deplasament constantă
- Există multiple combinații, fiecare din cei trei termeni este opțional
- Adresa fizică: Segment x 16 + Adresa efectivă
- Adresa efectivă: 16 biți
- Adresa fizică: 20 biți

$$PA = \left\{ \begin{array}{c} CS \\ SS \\ DS \\ ES \end{array} \right\} : \left\{ \begin{array}{c} BX \\ BP \end{array} \right\} + \left\{ \begin{array}{c} SI \\ DI \end{array} \right\} + \left\{ \begin{array}{c} 8\text{-bit displacement} \\ 16\text{-bit displacement} \end{array} \right\}$$

Exemple:

```
mov al, [bx]
mov al, [si]
mov al, [bp][di]
mov al, [bx][di]200
mov al, ss: [di]470
mov al, [bp+si+100]
```

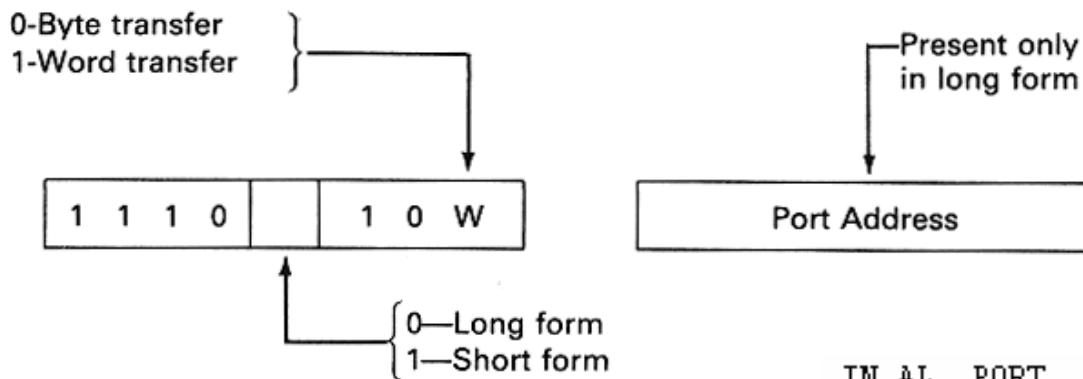




Adresare I/O



- I/O (spațiu separat), domeniu adresabil 0000 – FFFF (64 K-octeți)
- Două tipuri de instrucțiuni – scurtă și lungă. Instrucțiunea lungă permite specificarea portului ca o constantă între 0 și 255
- Registrul sursă sau destinație pentru operația I/O este întotdeauna AX (AL)
- Portul este specificat de o constantă, sau adresa lui este în DX



Exemple

```
IN AL, PORT  
IN AX, PORT  
IN AL, DX  
IN AX, DX
```

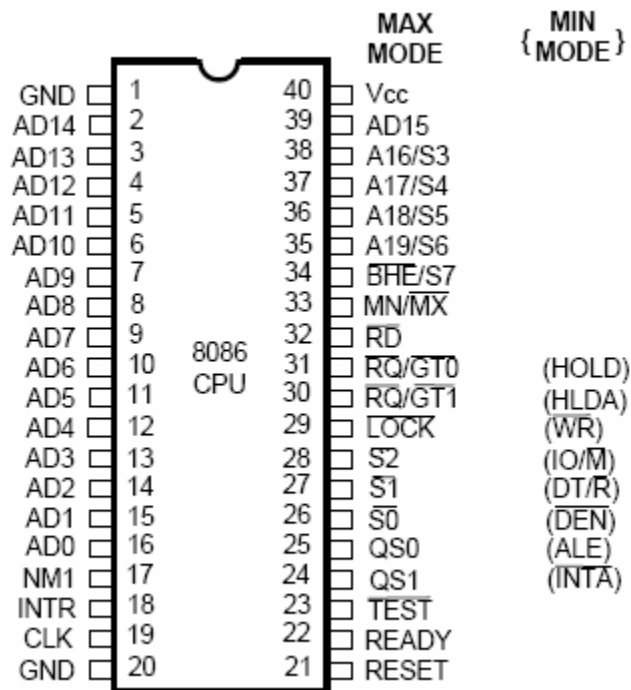
```
OUT PORT, AL  
OUT PORT, AX  
OUT DX, AL  
OUT DX, AX
```

```
(AL) <- (PORT)  
(AX) <- (PORT+1:PORT)  
(AL) <- ((DX))  
(AX) <- ((DX)+1:(DX))
```

```
(PORT) <- (AL)  
(PORT+1:PORT) <- (AX)  
((DX)) <- (AL)  
((DX)+1:(DX)) <- (AX)
```



Diagrama pinilor



AD15:0 – Adrese și date, multiplexate pe aceiași pini
A19:A16 – Linii de adresă superioară

BHE – Byte high enable

INTR – Interrupt request

MN/MX – Selecție între modul minim și maxim

WR, RD – write, read

DT/R – data transmit/receive

DEN – Data enable

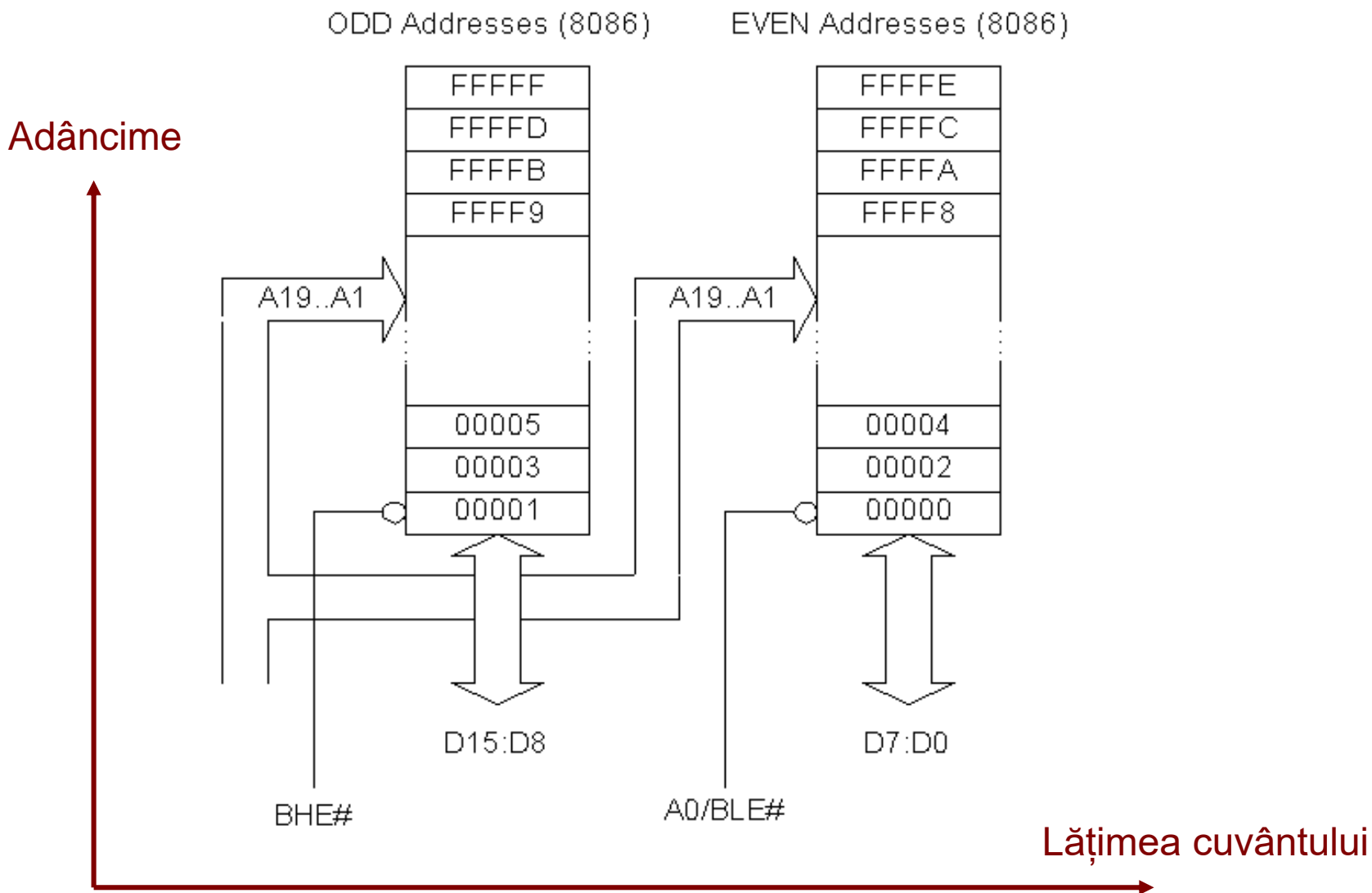
ALE – Address latch enable

IO/M – semnal care specifică dacă liniile de adresă indică o locație de memorie sau de I/O

#BHE	A0	Explicatie
0	0	Acces pe 16 biți (aliniat)
0	1	Byte superior, de la adresă impară
1	0	Byte inferior, de la adresă pară
1	1	Combinație nepermisă

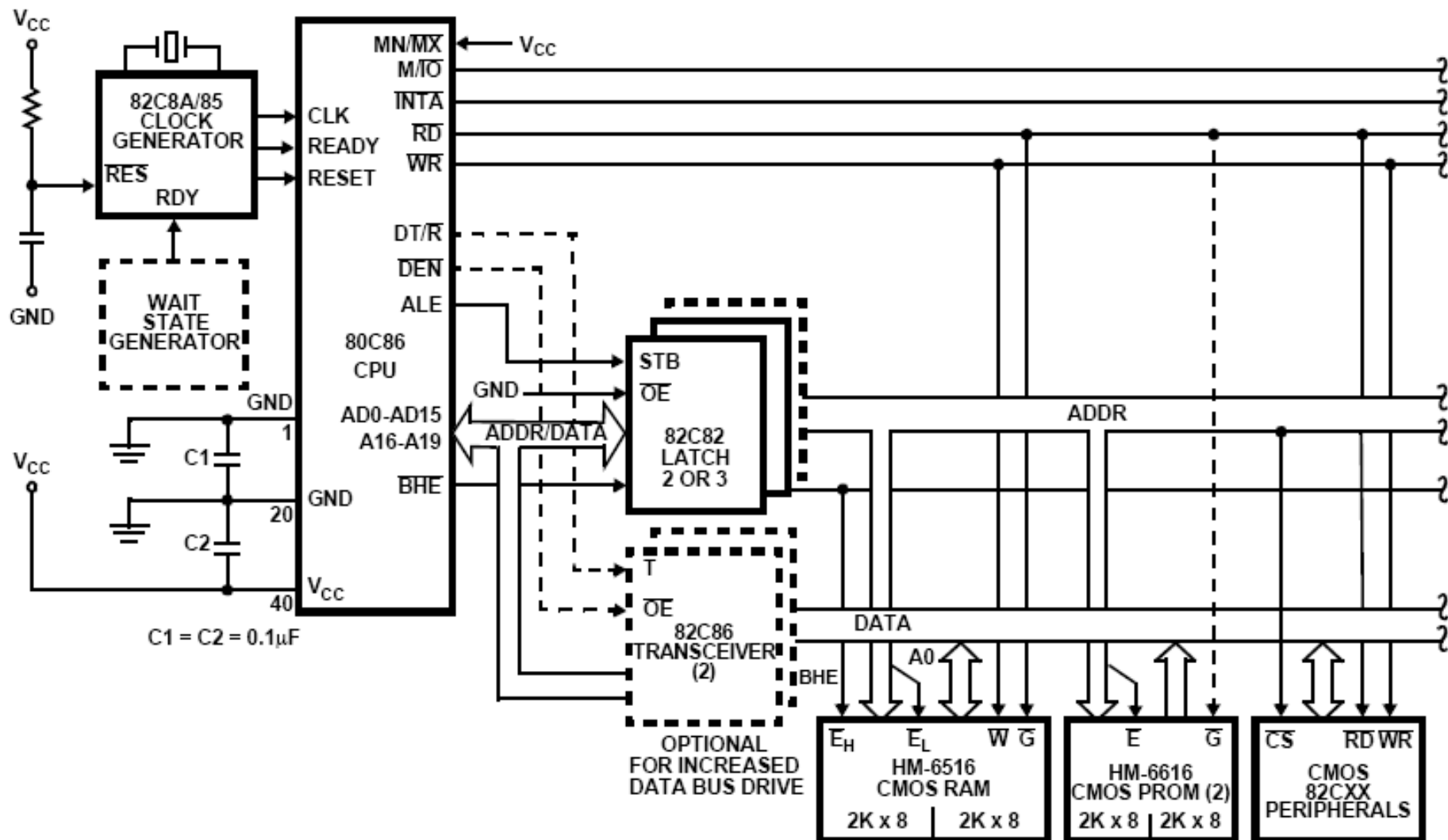


Organizarea memoriei



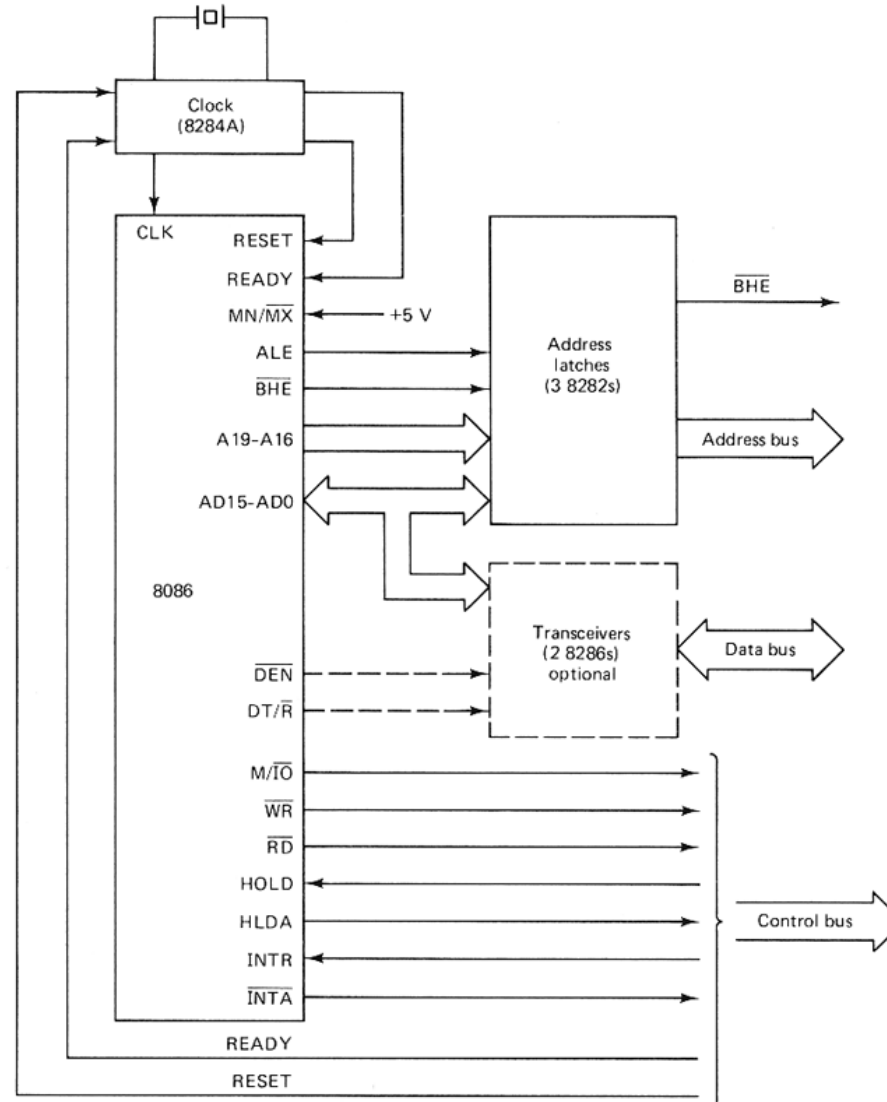


Sistem 8086 în mod minim



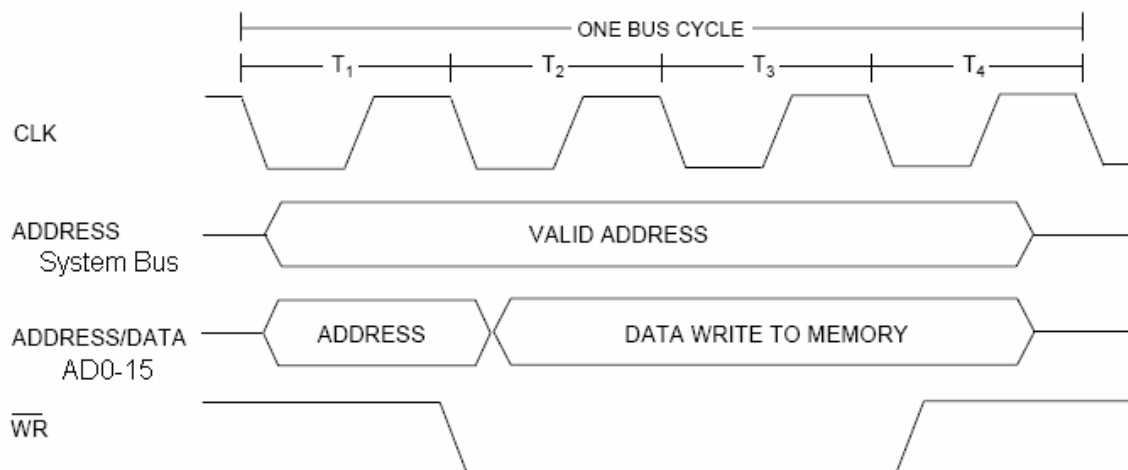
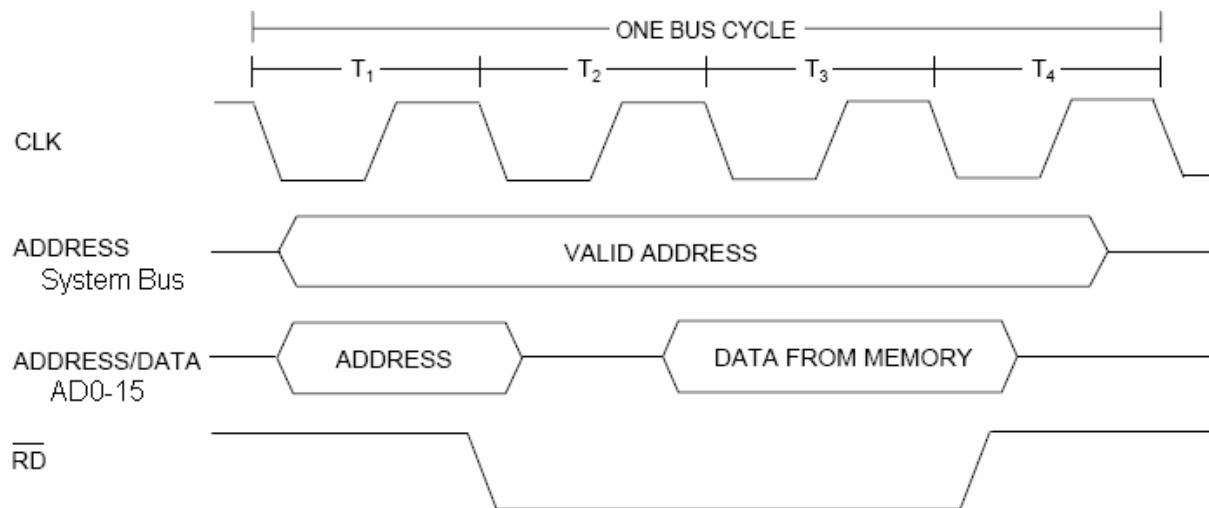


Formarea magistralelor (Bus) la 8086, mod minim



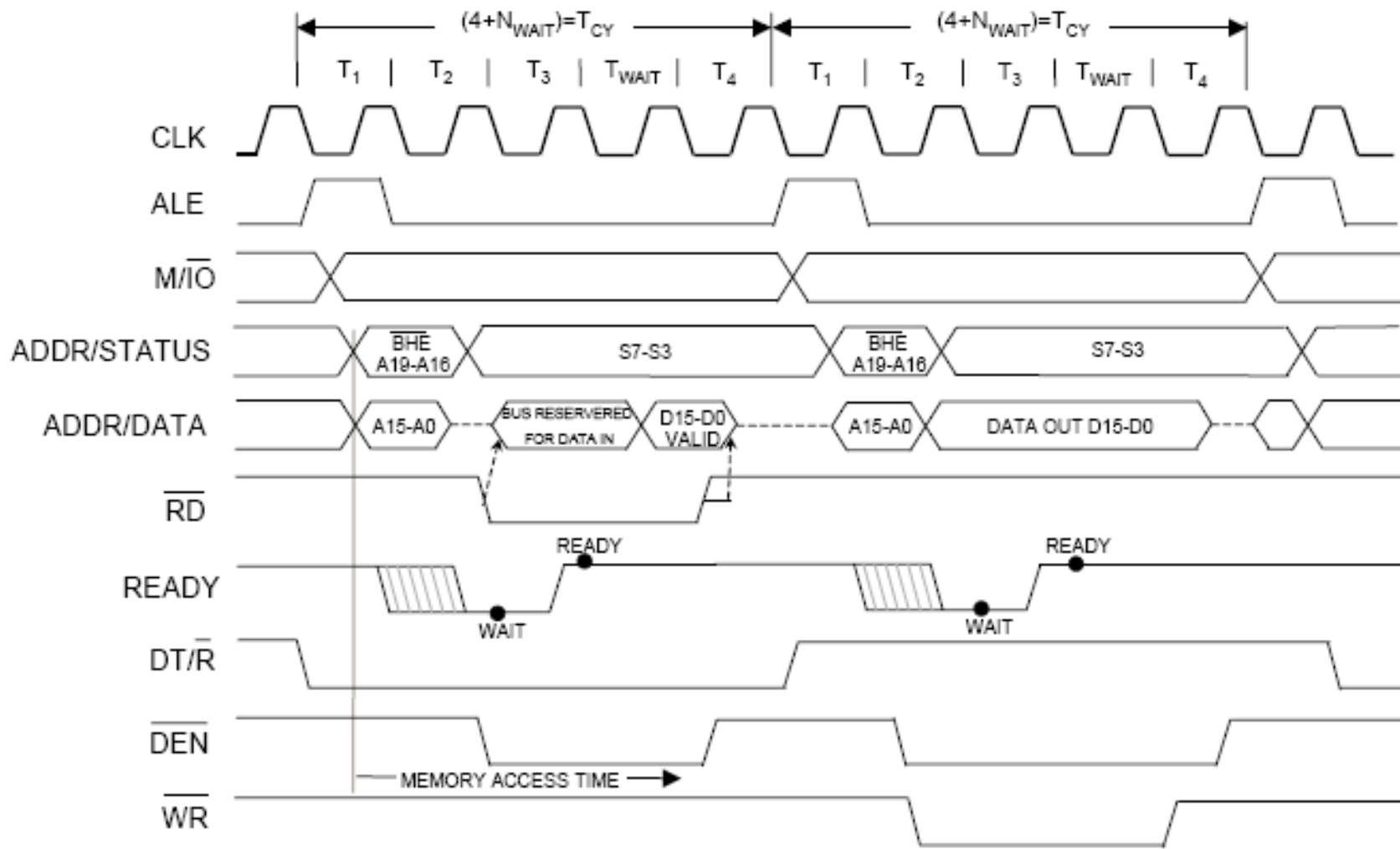


Diagrame de timp simplificate



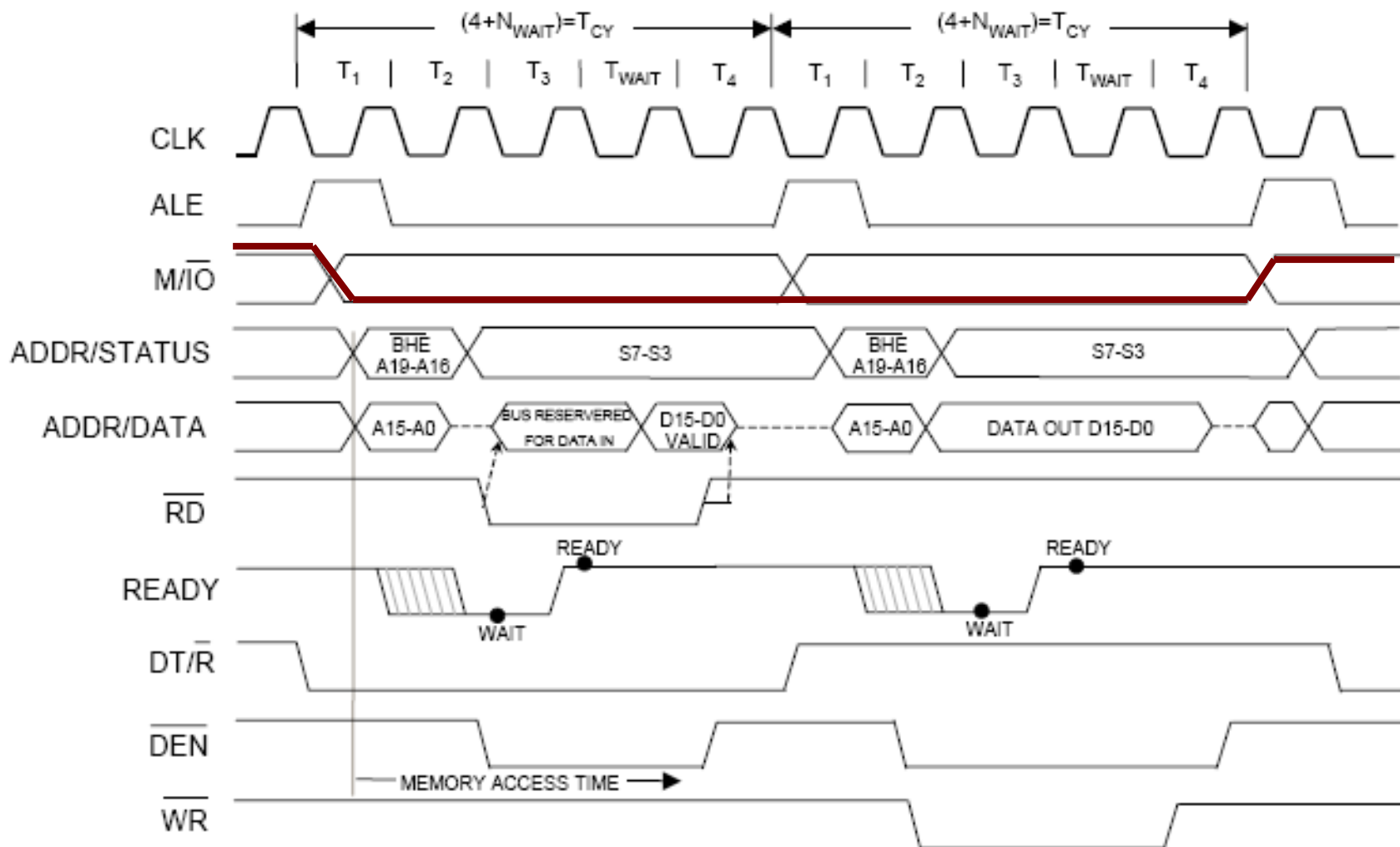


Diagrame de timp – detaliu





I/O Citire și Scriere – Diagrame de Timp



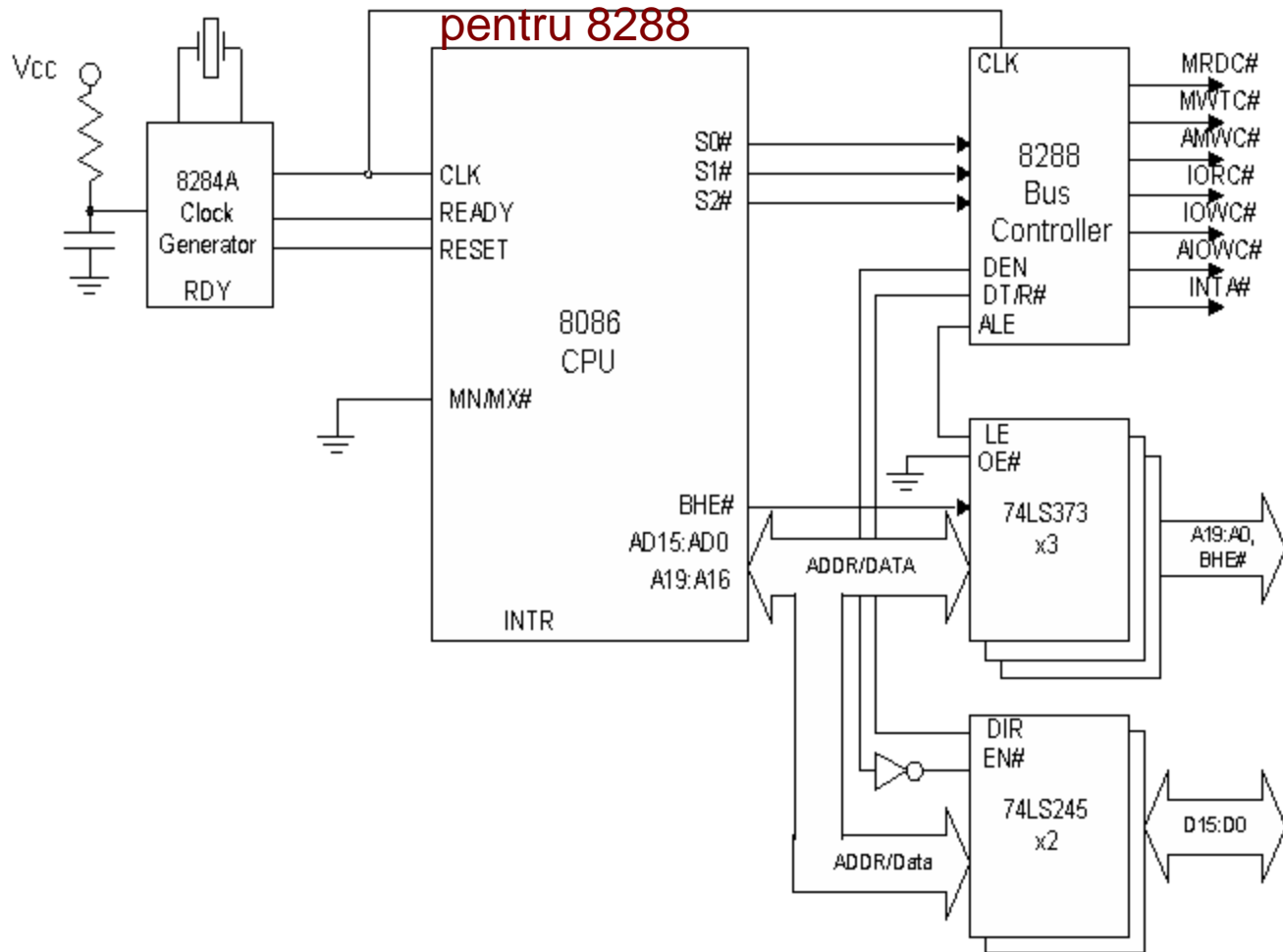
În modul minim, linia M/I $\bar{0}$ face diferența dintre accesul la memorie sau I/O, restul semnalelor fiind identice



Sistem 8086 în mod maxim



Procesorul indică starea, prin S2:S0, care este comandă pentru 8288



Comenzile pe bus sunt generate de controllerul 8288

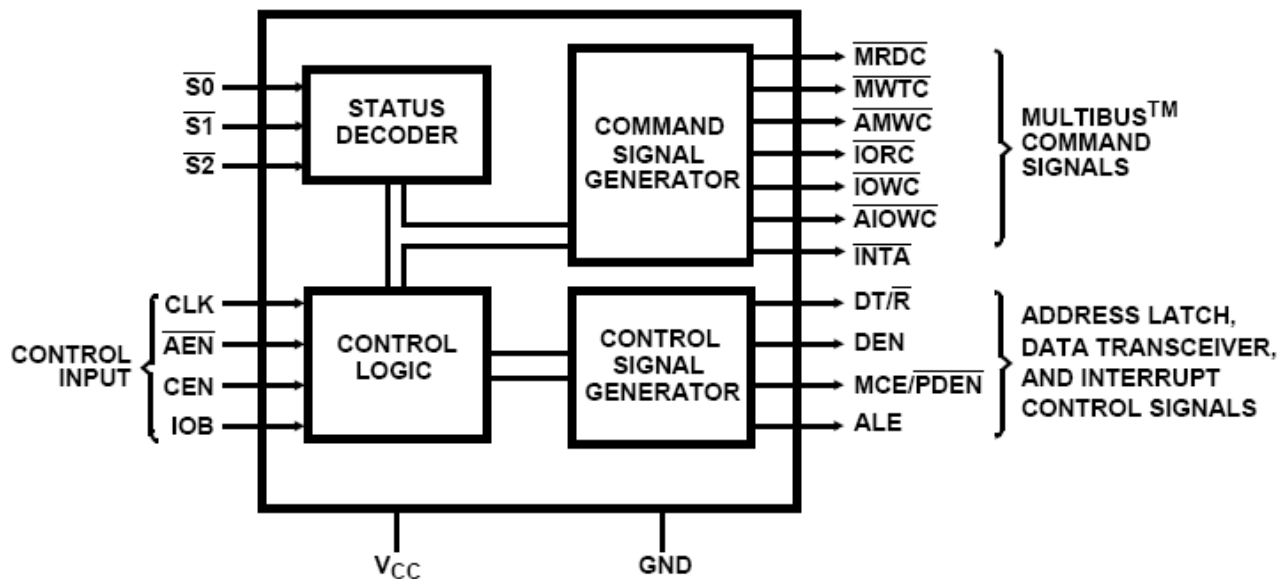
Bus de date și adrese, similar cu modul minim



Semnalele modului maxim



#S2	#S1	#S0	Explicație
0	0	0	Interrupt Acknowledge
0	0	1	Citire port I/O
0	1	0	Scriere port I/O
0	1	1	Halt
1	0	0	Instruction Fetch
1	0	1	Citire din memorie
1	1	0	Scriere în memorie
1	1	1	Pasiv (Nu se accesează bus-ul)



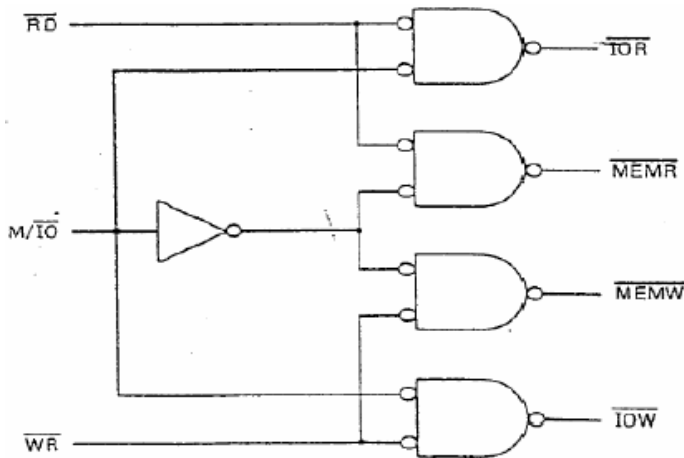


Interfațare dispozitive I/O

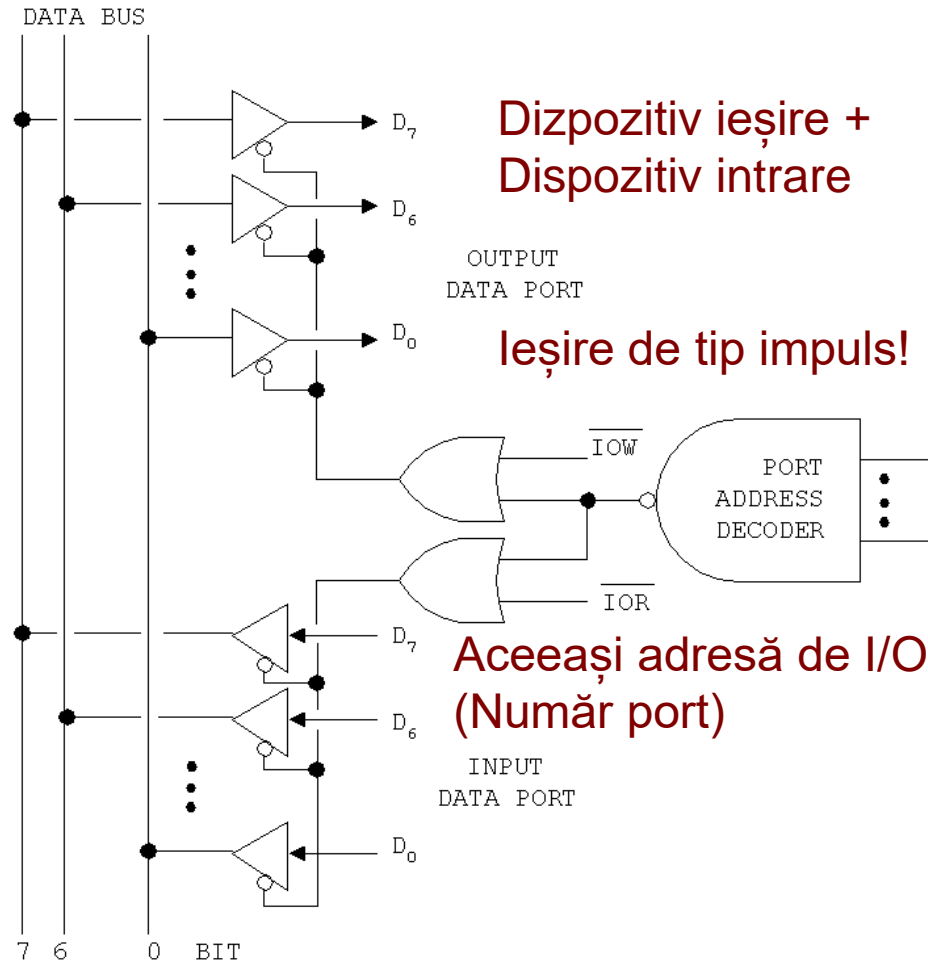


Dispozitivul este activat cand:

- Adresa transmisă de 8086 corespunde cu adresa proprie – Decodificare!
- Comanda transmisă de 8086 corespunde cu natura dispozitivului – I/O
- Două dispozitive pot avea aceeași adresă dacă sunt de tipuri diferite

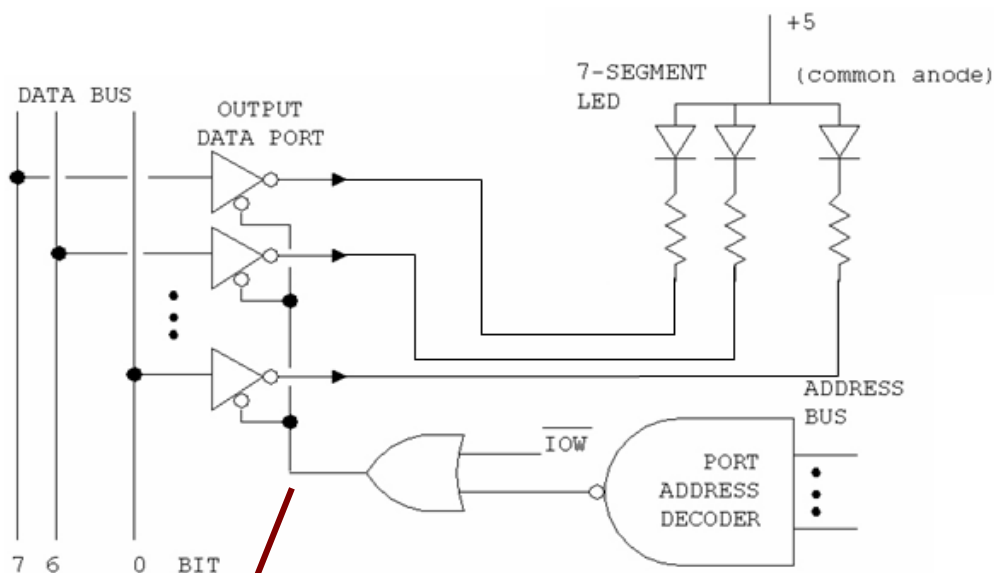


Din combinația semnalelor RD, WR, și M/IO se pot genera semnale explicite pentru IOR, IOW, MEMR, MEMW, similare cu cele din modul maximal → se poate face o prezentare unitară a celor două moduri

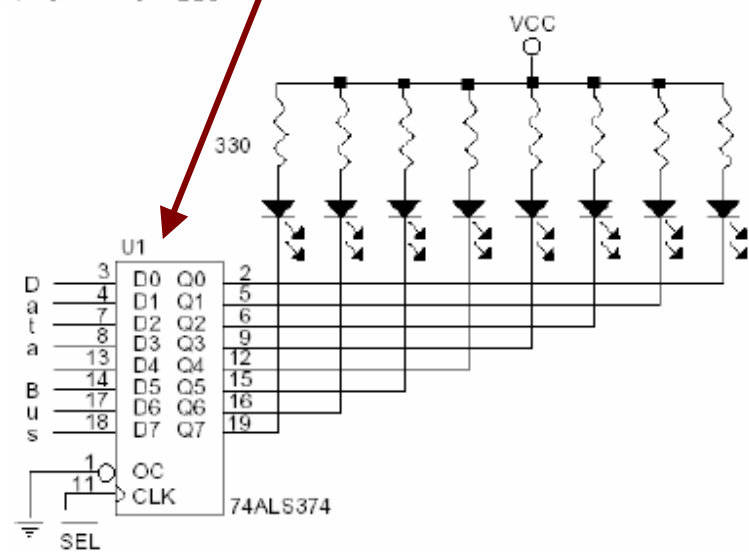
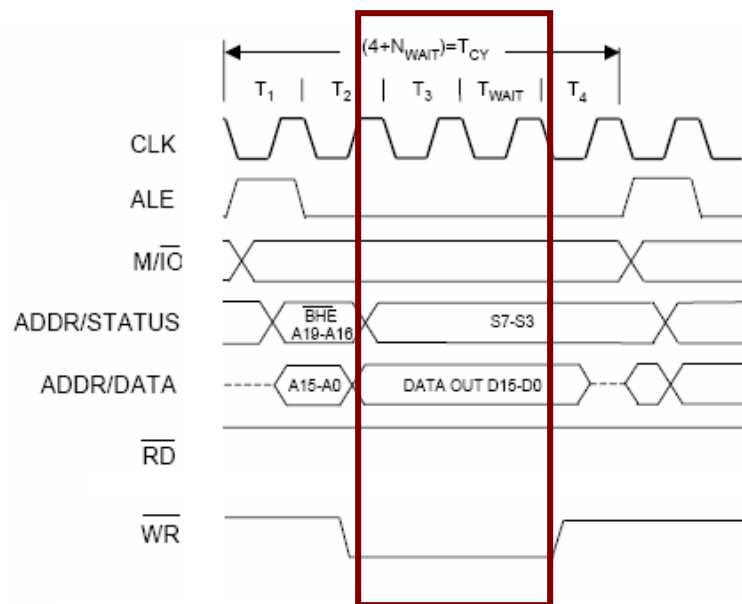




Problema ieșirii – persistența



Led-urile vor fi activate doar pe timpul unui ciclu de bus 8086 → invizibil!

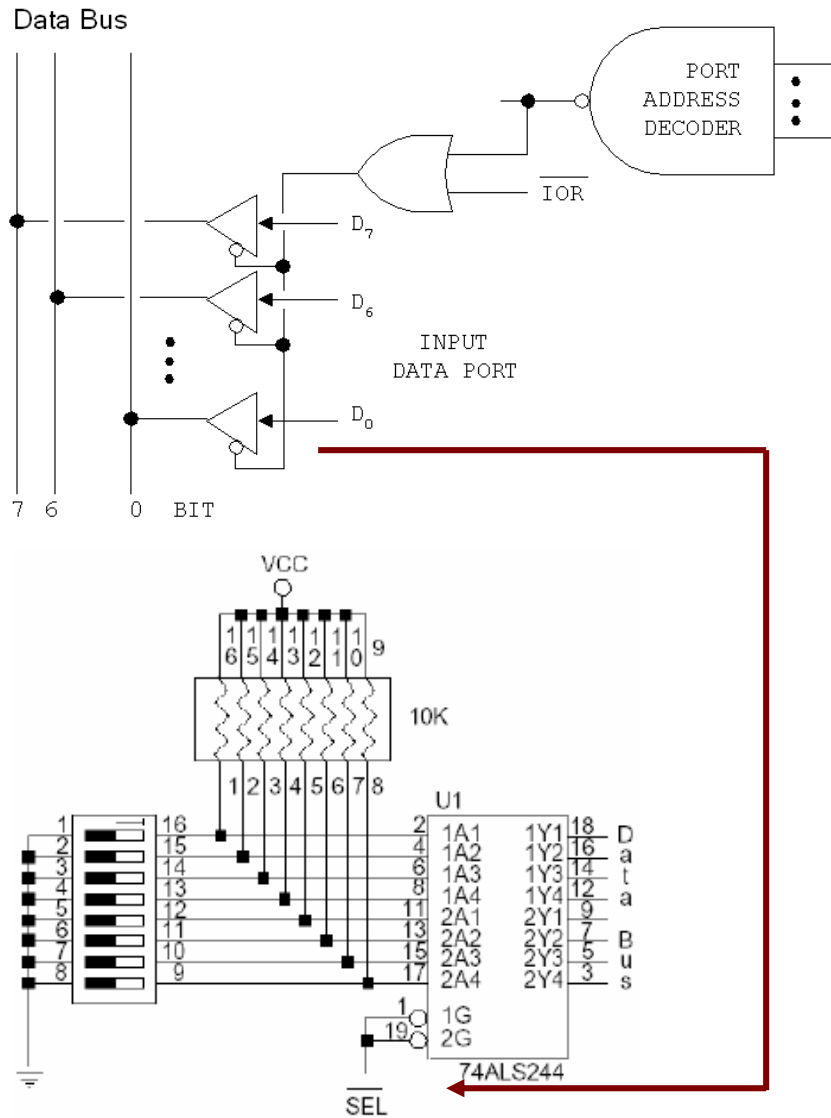


Soluția – utilizarea unui registru pentru mentinerea datelor după terminarea ciclului de bus

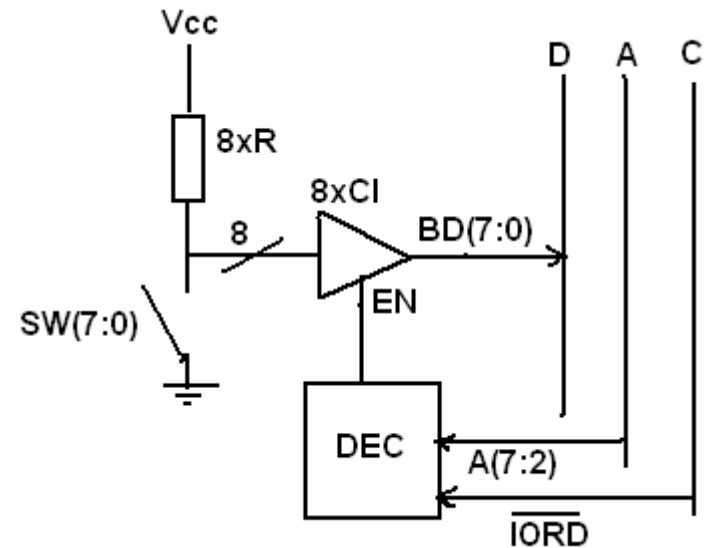
Latch sau Flip-Flop?



Problema intrării – decuplarea

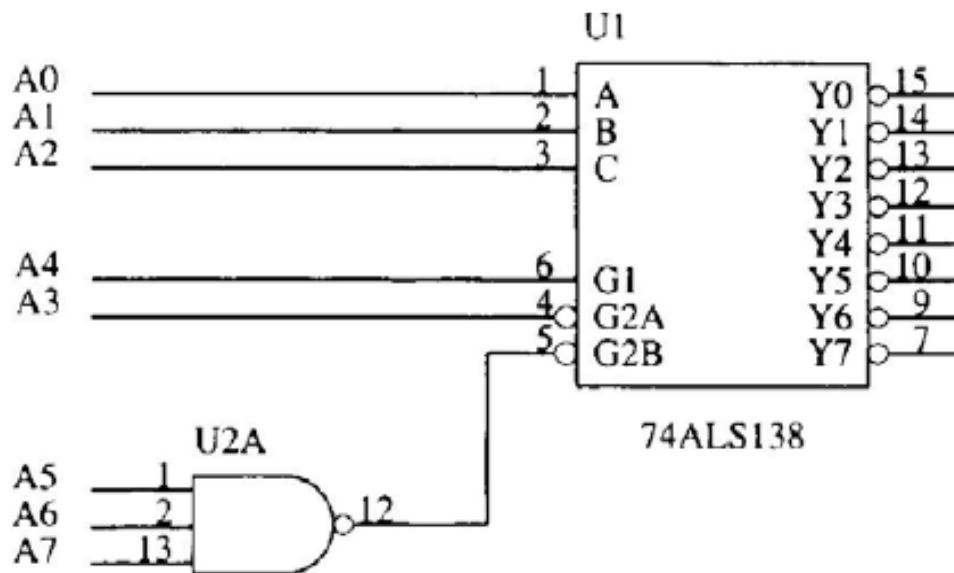
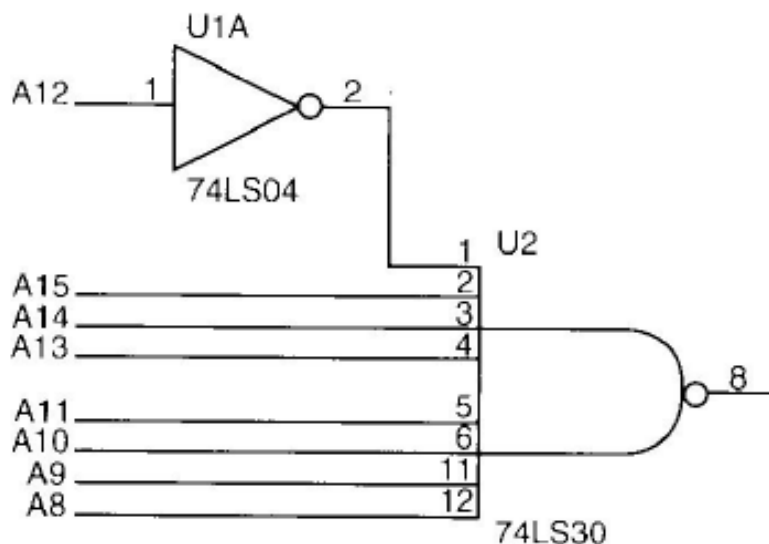


- Bus-ul de date este partajat (mai multe dispozitive impart aceleași linii de intrare)
- Un dispozitiv scrie date pe bus doar atunci cand este solicitat prin instructiuni I/O Read (in) !
- Folosire buffere 3-state
- Datele trebuie să fie stabile în momentul citirii.





Decodificarea adreselor I/O



Decodificarea cu circuite logice discrete:
AND, OR, NAND, NOR, NOT

Adresa EFXh = 11101111XXXXXXXX

Decodificarea cu un decodicator 74LS138

- G1, G2A, G2B – validari
- A, B, C – selecția unei linii
- Permite selectia a 8 dispozitive I/O

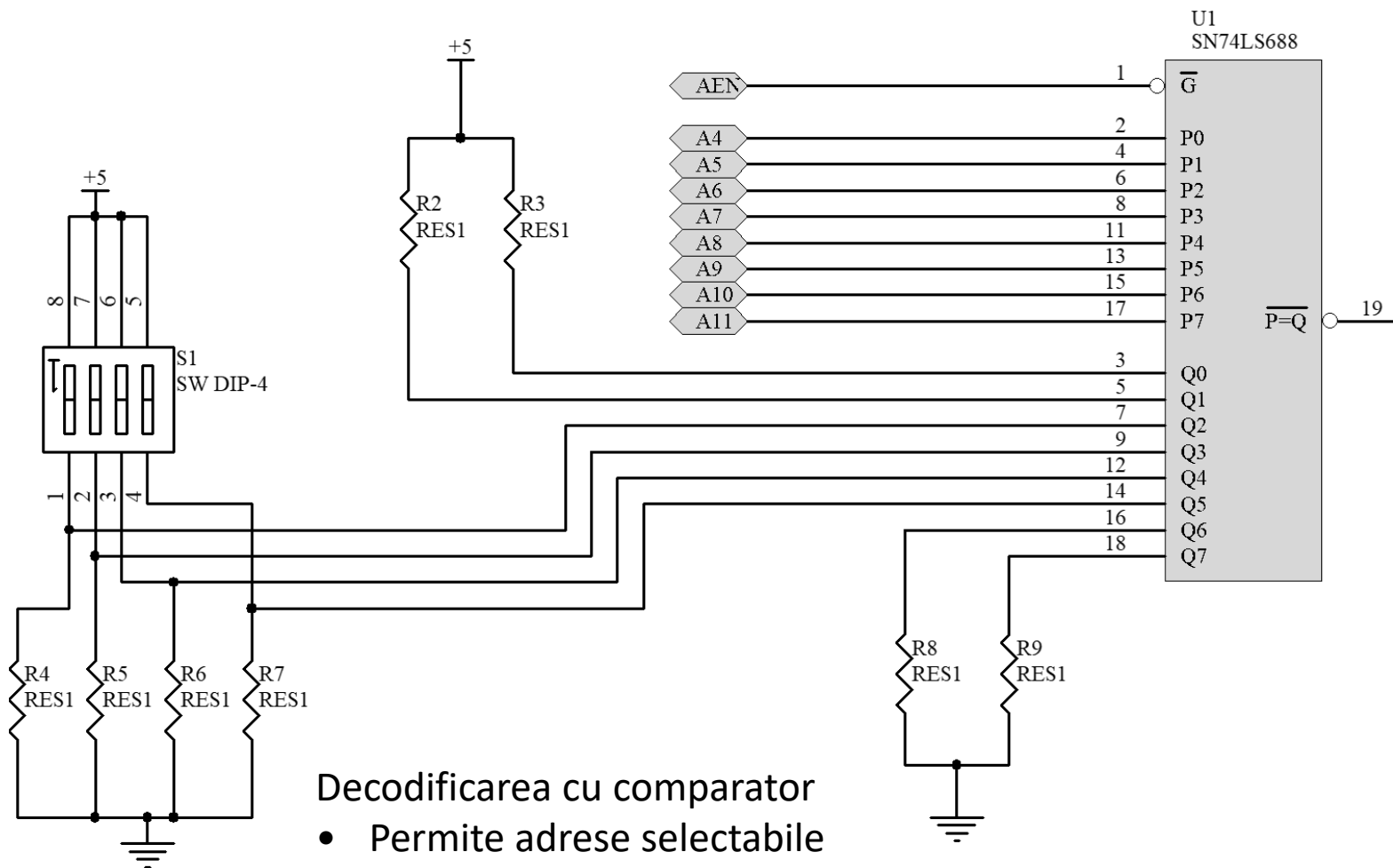
- A7:A5 trebuie sa fie '1'
- A3 trebuie sa fie '0'
- A4 trebuie sa fie '1'



Decodificarea adreselor I/O



A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	Sw	sw	sw	Sw	1	1	X	X	X	X

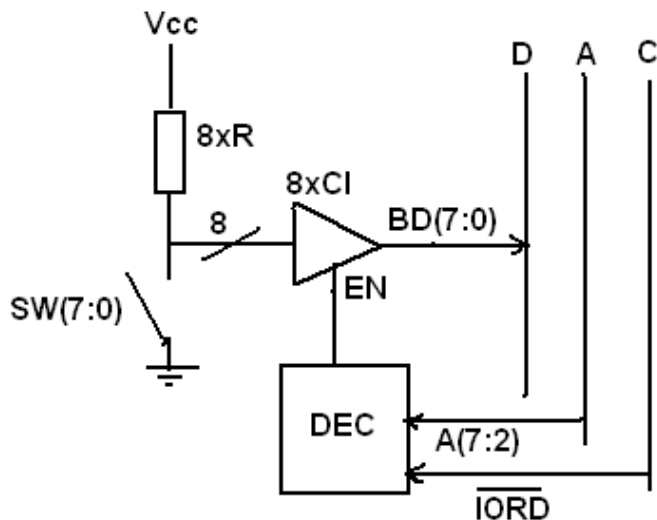




Acces pe Byte, acces pe Word

- Combi-nația BHE/A0 se aplică oricarui transfer pe bus – memorie sau I/O

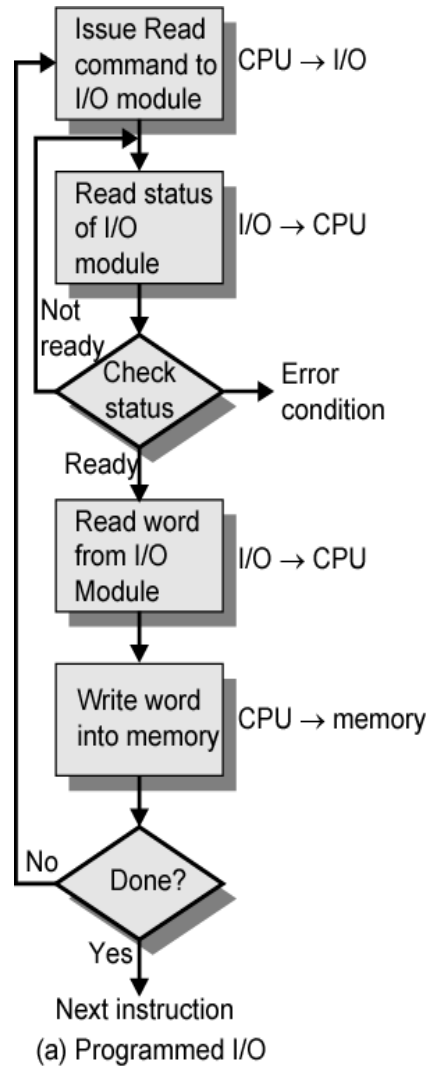
#BHE	A0	Explicatie
0	0	Acces pe 16 biți (aliniat)
0	1	Byte superior, de la adresă impară
1	0	Byte inferior, de la adresă pară
1	1	Combi-nație nepermisă



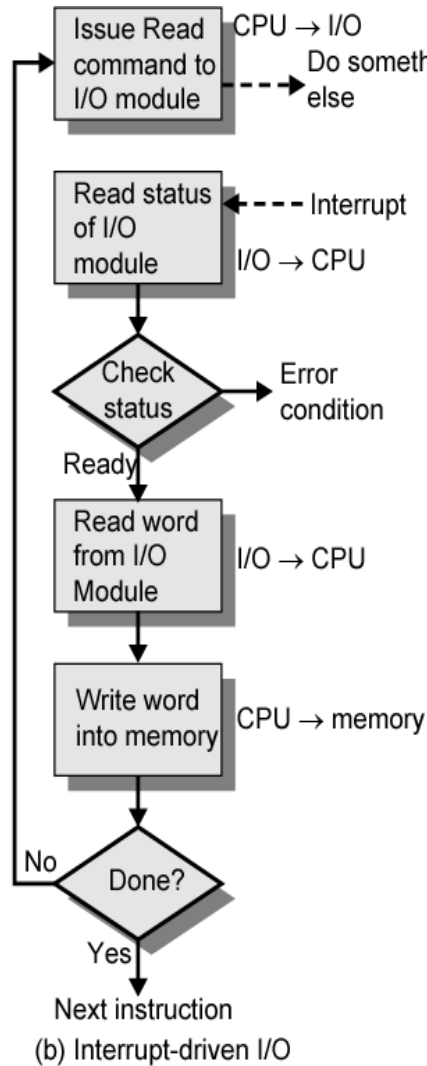
Conectarea la BD7:0 obligă A0 să fie '0'



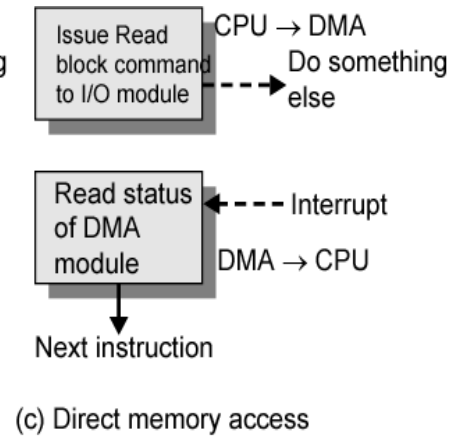
Tipuri de transfer



Programat



Întrepereri



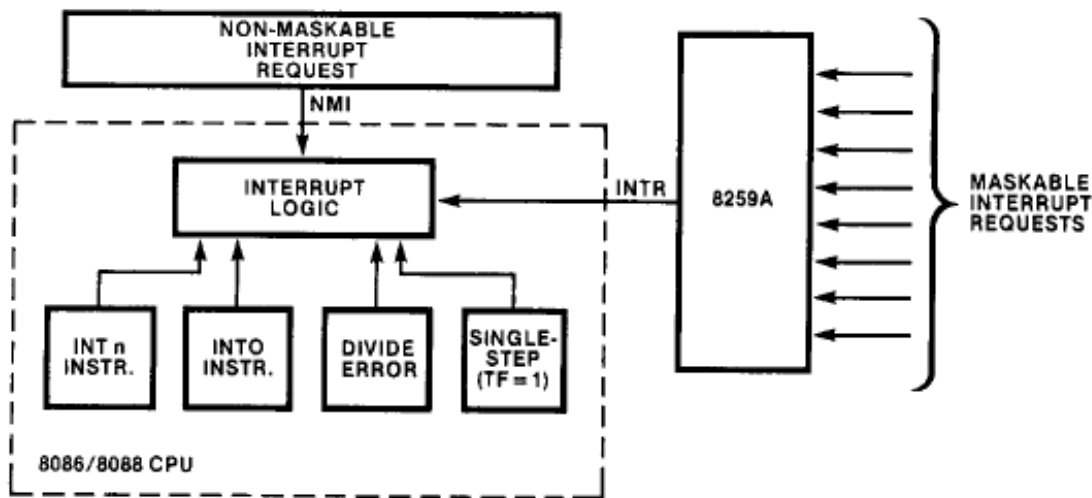
DMA



Întreruperi – Surse de întrerupere



- Inițiate prin Software – instrucțiuni INT n
- Inițiate prin Hardware
 - Excepții – semnalizare erori, (impartire cu 0,..)
 - Externe – dispozitive periferice, memorii
- Întreruperile pot fi:
 - Mascabile (Afectate de flag-ul IF)
 - Nemascabile



Prioritatea întreruperilor

INTERRUPT	PRIORITY
Divide error, INT n, INTO NMI INTR Single-step	highest lowest

Timpi

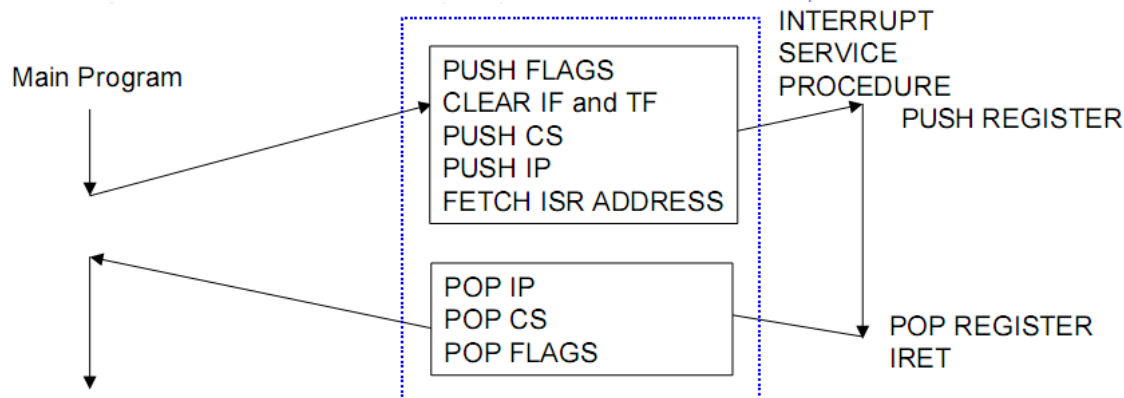
Interrupt Class	Processing Time
External Maskable Interrupt (INTR)	61 clocks
Non-Maskable Interrupt (NMI)	50 clocks
INT (with vector)	51 clocks
INT Type 3	52 clocks
INTO	53 clocks
Single Step	50 clocks



Înteruperi – Instrucțiuni ASM



Mnemonic	Meaning	Format	Operation	Flags Affected
CLI	Clear interrupt flag	CLI	0 → (IF)	IF
STI	Set interrupt flag	STI	1 → (IF)	IF
INT n	Type n software interrupt	INT n	(Flags) → ((SP - 2) 0 → TF, IF (CS) → ((SP) - 4) (2+4·n) → (CS) (IP) → ((SP) - 6) (4·n) → (IP)	TF, IF
IRET	Interrupt return	IRET	((SP)) → (IP) ((SP) + 2) → (CS) ((SP) + 4) → (Flags) (SP) + 6 → (SP)	All
INTO	Interrupt on overflow	INTO	INT 4 steps	TF, IF
HLT	Halt	HLT	Wait for an external Interrupt or reset to occur	None
WAIT	Wait	WAIT	Wait for $\overline{\text{TEST}}$ input to go active	None





Vectori de Întrerupere

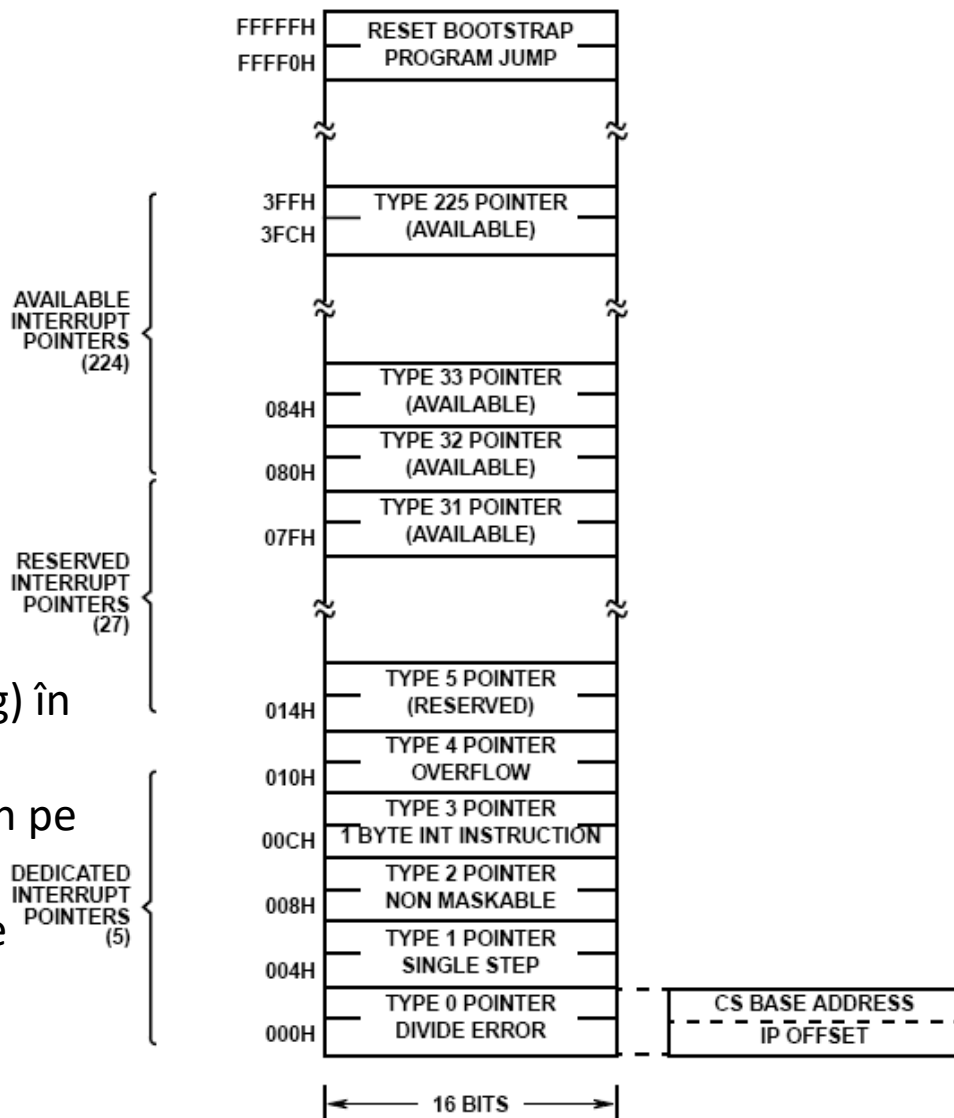


Tabela de vectori se află în RAM

- adresele de salt pentru fiecare tip de întrerupere se pot configura de către programator

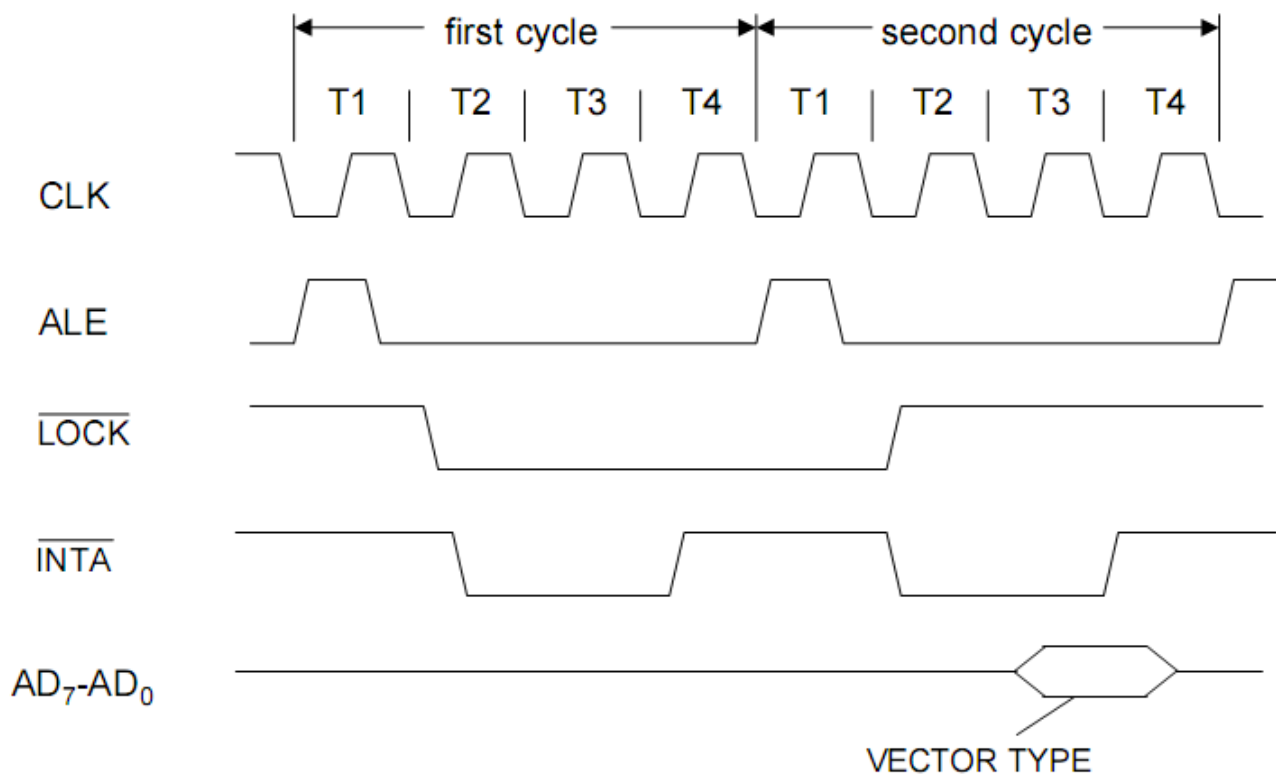
Tratarea întreruperilor

- Mascare – bitul IF (interrupt enable flag) în registrul de stare
- INTR – level triggered, sincronizat intern pe CLK ↑
- Durata INTR High – inclusiv perioada de ceas înainte de terminarea instrucțiunii curente





Ciclul INTA pentru întreruperi externe mascabile

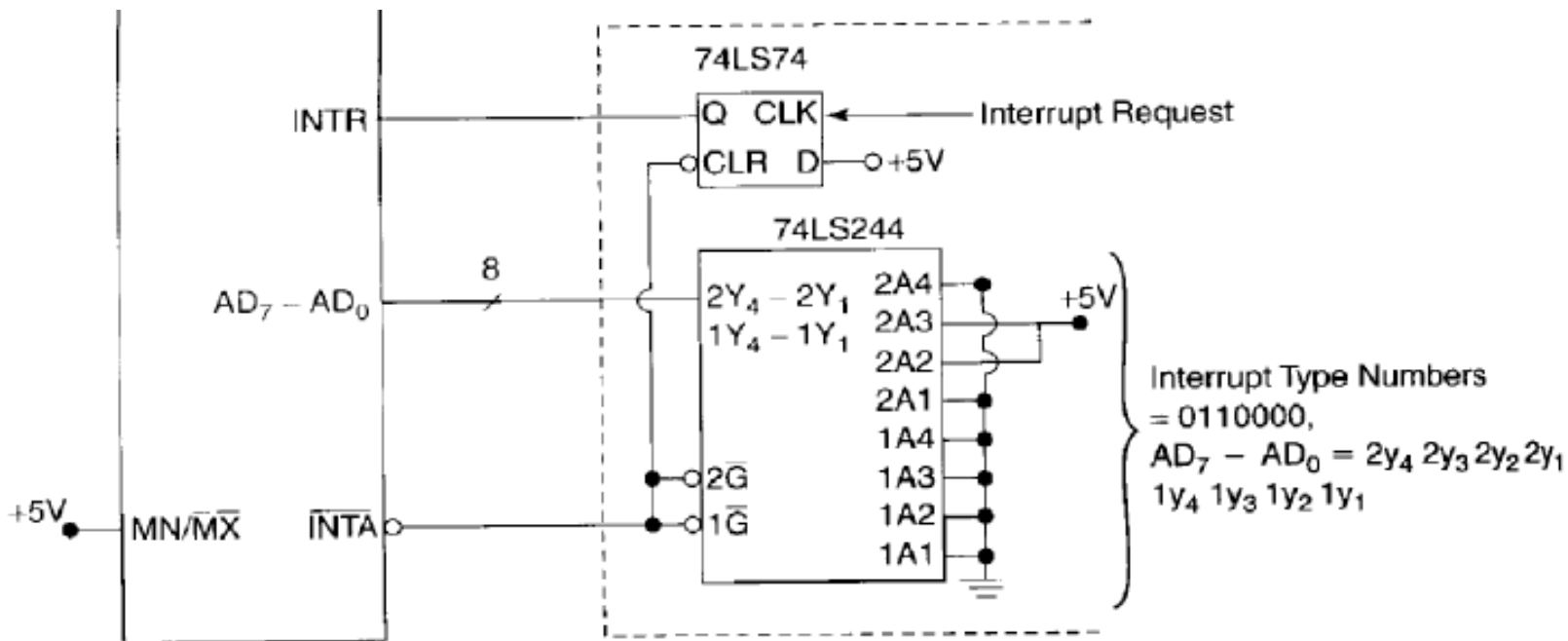


Un dispozitiv care generează întreruperi trebuie să fie capabil să:

- Scribe pe BUS-ul de date tipul intreruperii (indexul in tabela de vectori)
- Să dezactiveze INTR cand primește confirmarea INTA



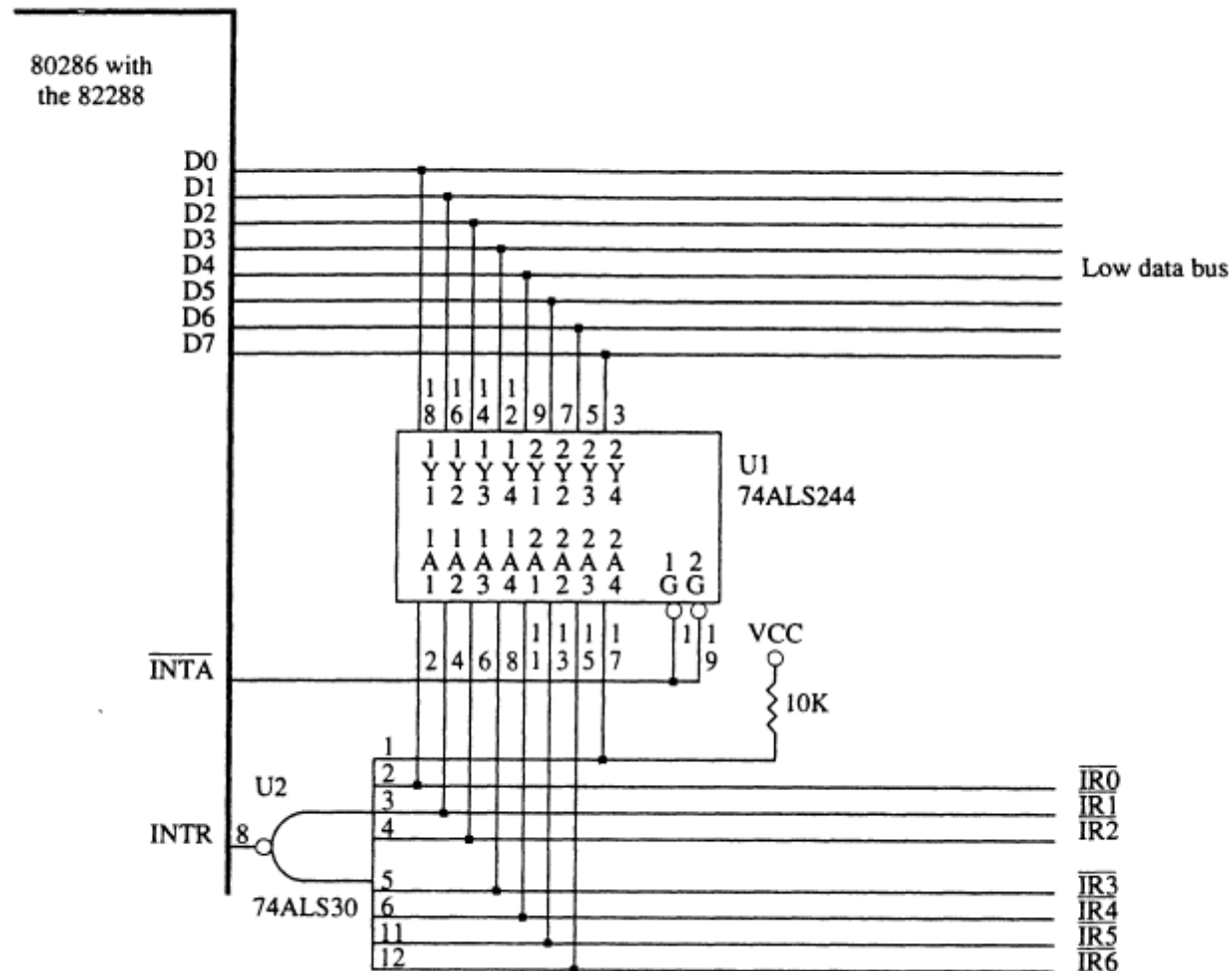
Întreruperi – Conectare dispozitive simple



- Generarea tipului de **întrerupere 60h**, când linia “Interrupt Request” este activă
- INTA produce scrierea tipului de întrerupere pe BUS, și ștergerea bistabilului care produce INTR
- 74LS244 – 3-STATE Buffer/Line Driver/Line Receiver



Întreruperi – Conectare dispozitive simple



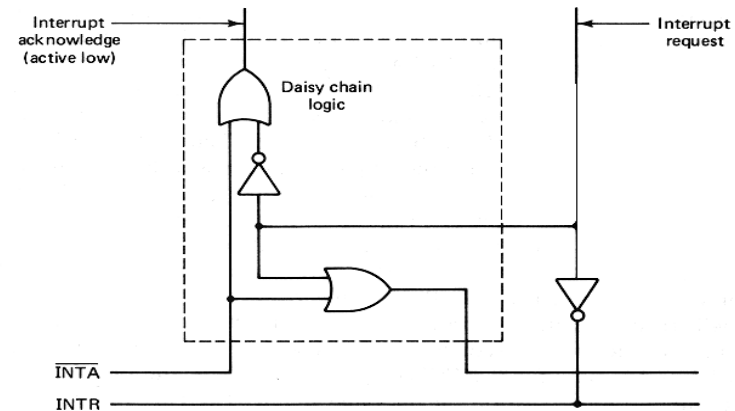
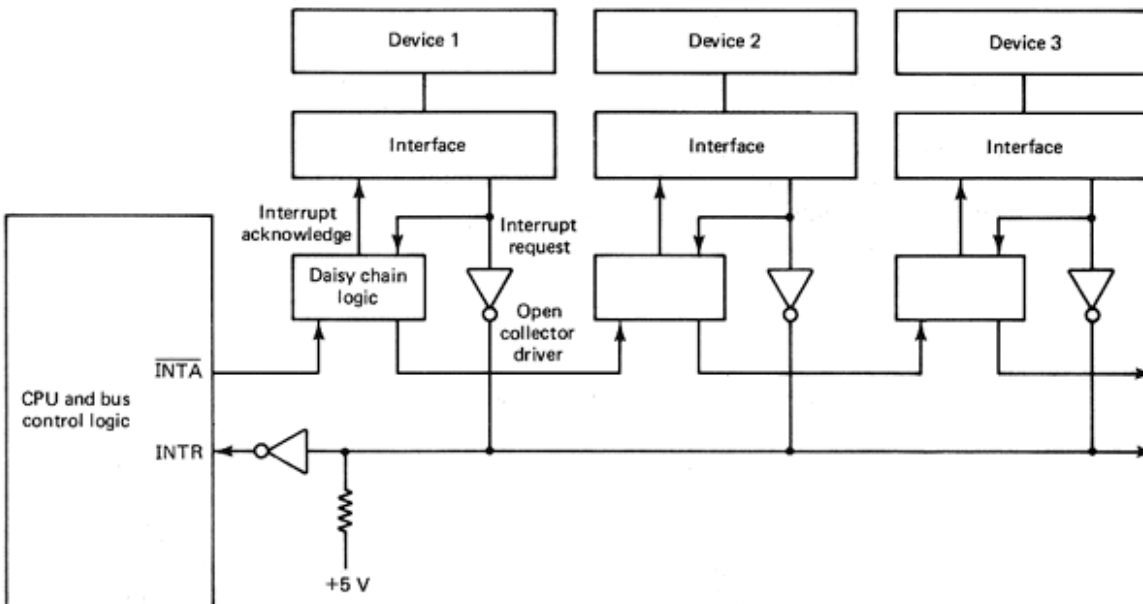
Generare de multiple tipuri de întrerupere, în funcție de dispozitiv



Tratarea cererilor multiple



- **Sondare (Polling)** – la apariția unei cereri de întrerupere, procesorul interoghează fiecare sursă potențială, iar aceasta raspunde. Ordinea de interogare a dispozitivelor este ordinea priorității întreruperilor.
- **Daisy Chain** – implementarea metodei polling in hardware. Limitare la lungimea lanțului din cauza întârzierilor de propagare

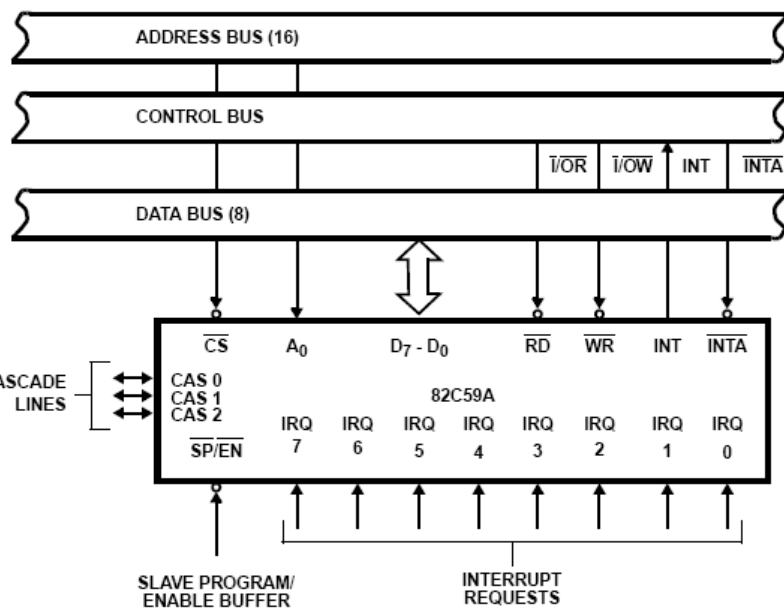
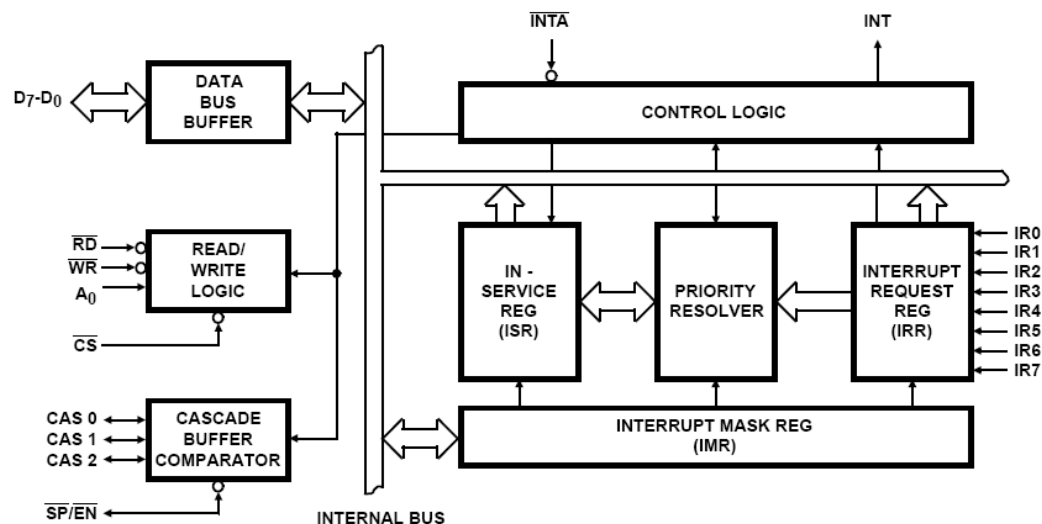




Controller de întreruperi 8259A



- 8 linii de cerere de întrerupere
- Prioritățile sunt programabile
- Se pot cascada – 8 slaves, 64 cereri de intrerupere in total





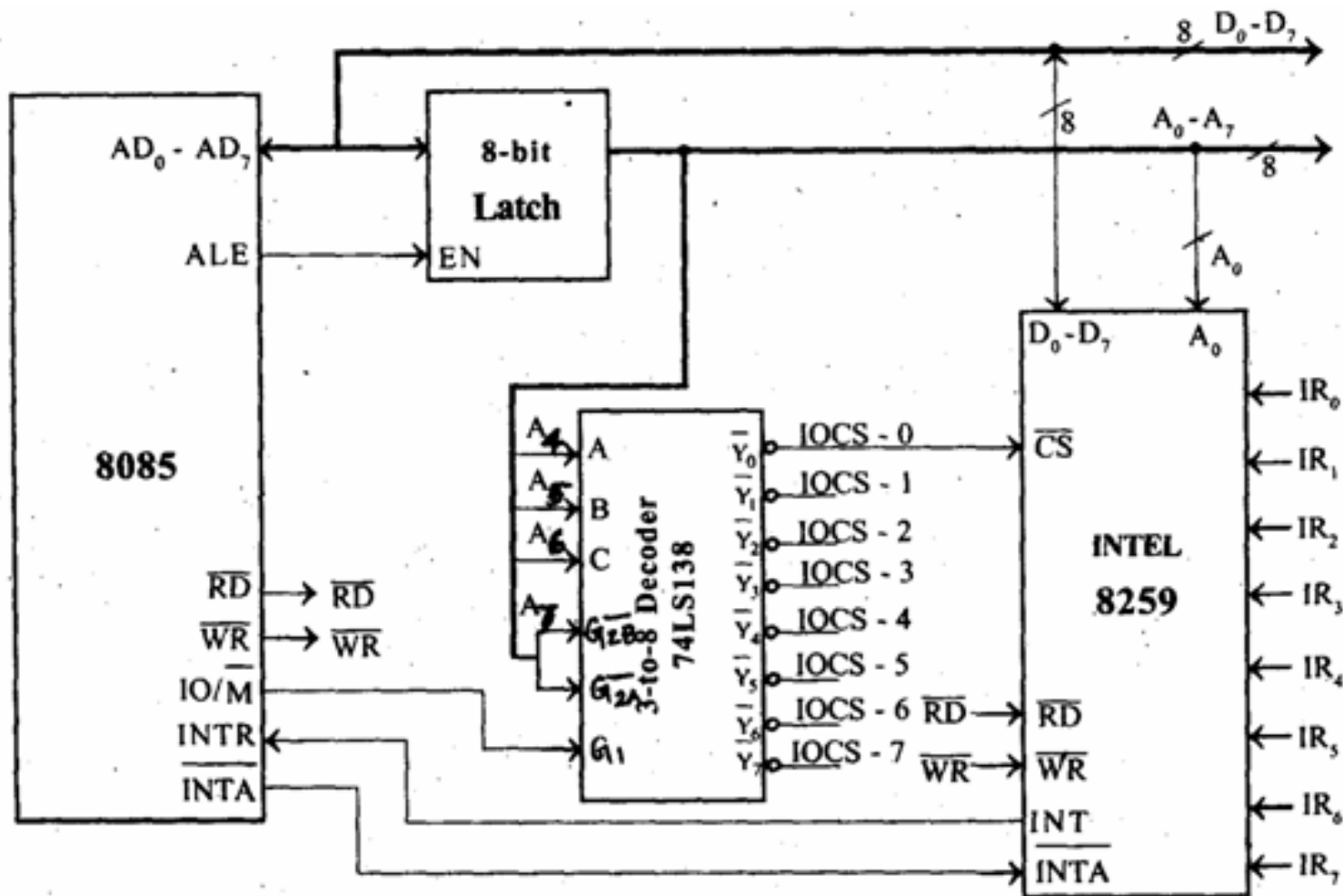
Controller de întreruperi 8259A



- PIC 8259 informează procesorul despre cererea de întrerupere, activând pinul INTR al procesorului
- Procesorul termină execuția instrucțiunii curente
- Procesorul trimite semnalul de Acknowledgment (INTA) către PIC 8259
- PIC 8259 transmite procesorului numărul vectorului pentru întreruperea cerută
- Procesorul folosește acest vector pentru a determina adresa unde este stocată rutina de tratare a întreruperii (ISR)
- Procesorul salvează pe stivă flagurile, CS și IP (în aceasta ordine).
- Procesorul pune IF pe zero.
- Procesorul setează CS:IP la adresa ISR și începe execuția acestei subrutine



Controller de întreruperi 8259A





Referințe



1. 8086/8088 datasheet
2. 8259 datasheet