

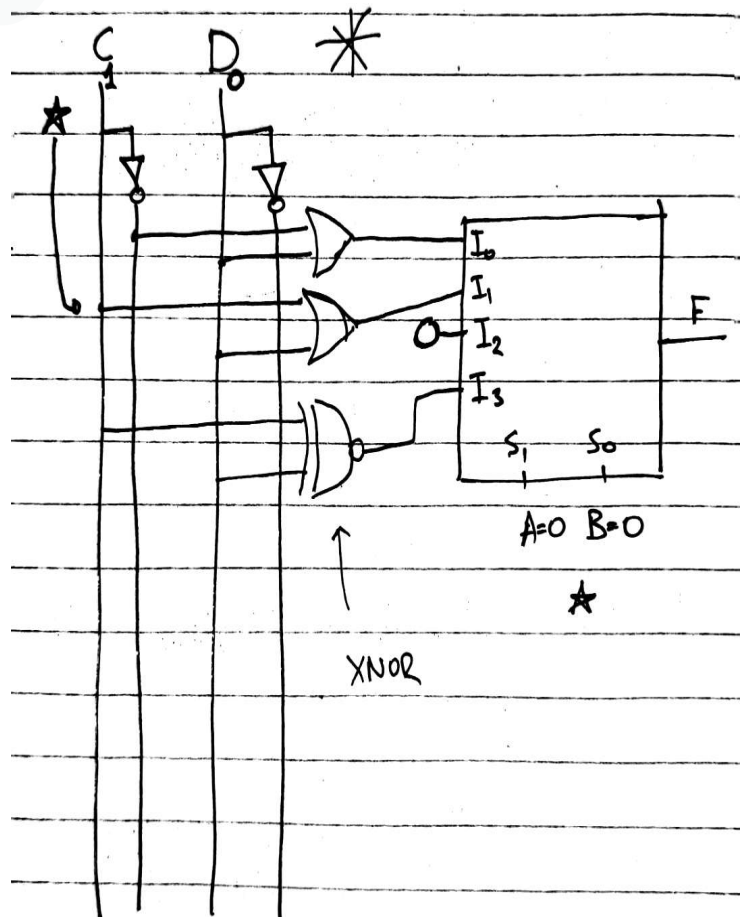
Αρχιτεκτονική Διαλεξη 3

Πολυπλέκτες:

Να υλοποιηθει με πολυπλέκτη MUX 4x1 $F=(A,B,C,D)=\Sigma(0,1,2,3,5,6,7,12,15)$ όπου τα σήματα A,B συνδέονται με τις γραμμές επιλογή και τα C,D με τις εισόδους

4x1: I0-I3 , 2 γραμμές επιλογής S0,S1

	A	B	C	D	F
A=0 B=0 F=I0	0	0	0	0	1
	0	0	0	1	1
	0	0	1	0	1
	0	0	1	1	1
A=0 B=1 F=I1	0	1	0	0	0
	0	1	0	1	1
	0	1	1	0	1
	0	1	1	1	1
A=1 B=0 F=I2	1	0	0	0	0
	1	0	0	1	0
	1	0	1	0	0
	1	0	1	1	0
A=1 B=1 F=I3	1	1	0	0	1
	1	1	0	1	0
	1	1	1	0	0
	1	1	1	1	1



Είσοδοι Πολυπλέκτη:

MUX 4x1

S1 τροφοδοτείται από το A
S0 τροφοδοτείται από το B

S1	S0	F
0	0	$F=I_0$
0	1	$F=I_1$
1	0	$F=I_2$
1	1	$F=I_3$

εδώ έχουμε 4 και στο προηγούμενο πίνακα 16 άρα Σπάει ο πίνακας αληθείας στα 4 (τελευταία σελίδα το κοκκίνο)

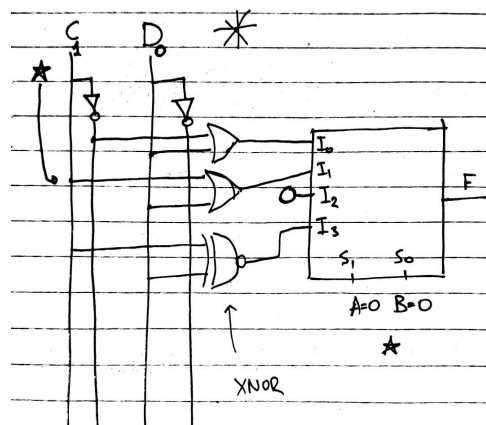
Κανόνας: Γενικά πίνακας αληθείας χωρίζεται σε $\frac{M}{N}$ τμήματα όπου M είναι όλοι οι συνδυασμοί του πίνακα αληθείας και N είναι το πλήθος των εισόδων του πολυπλέκτη $M=16$ $N=4$

1) $A=B=0$ Ψάχνω τι θα συνδέσω στο I_0 εκφράζω το I_0 ως συνάρτηση των C,D

C\D	0	1
0	1	1
1		1

1η δυνάδα το C'
2η δυνάδα το D
 $I_0 = C' + D$

Σχήμα πρώτη σελίδα →



2) $A=0$ και $B=1$ $F=I1$

$C \backslash D$	0	1
0		1
1	1	1

$I1 = C + D$

3) $I2=0$

4) $I3$ $A=B=1$

$C \backslash D$	0	1
0	1	
1		1

$C'D' + CD$
 \uparrow
 Xnor

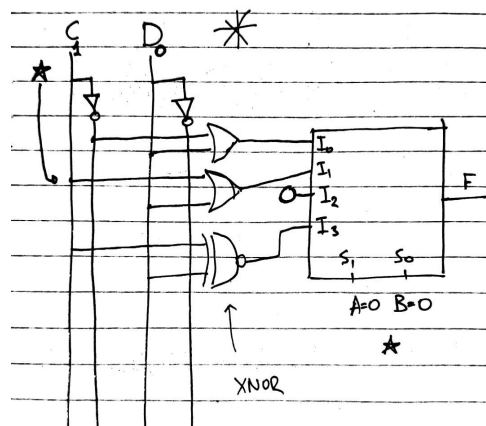
Εστω οτι την χρονική στιγμή E

$A=B=0$

$C=1$

$D=0$

$A=B=0$ σημαίνει $F=I0$ κοίτα σχήμα το αστεράκι



Μνήμη

RAM-Μέγεθος : Εξαρτάται από

1. Το πλήθος των Modules/τσιπ(Chip)
2. Πόσες λέξεις χωράει κάθε τσιπ
3. Από το μέγεθος της λέξης

Μια μνήμη με 8 Chip

1M λέξεις ($2^{20}=1M$) → 32 Mbyte (8x1Mx4)
4 byte μέγεθος λέξεις

Θα μπορούσε όμως να ήταν και άλλος συνδυασμός π.χ.

16 Chip

1M λέξεις ($2^{20}=1M$) → 32 Mbyte (16x1Mx2)
2 byte μέγεθος λέξεις

4 Chip

2M λέξεις ($2^{20}=1M$) → 32 Mbyte (4x2Mx4)
4 byte μέγεθος λέξεις

Φτιάχνουμε μνήμη :

Έστω ότι έχουμε μία μνήμη με 4 chip

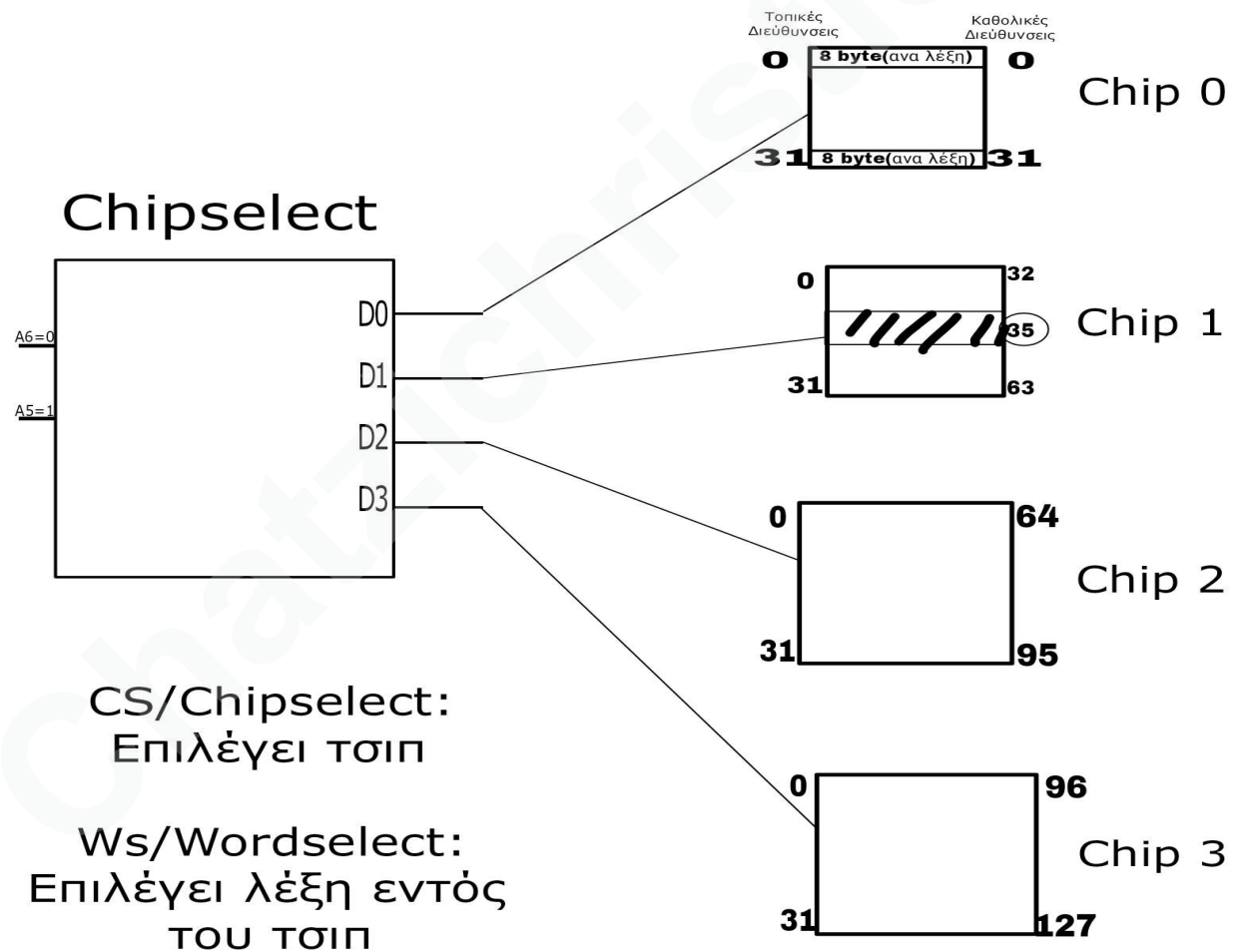
1 byte ανα λέξη

32 byte ανα Chip

$4 \times 32 = 128$ bytes

Θα δούμε

1. Εξωτερική οργάνωση/ Αποκωδικοποίηση μνήμης
2. Εσωτερική οργάνωση σε επίπεδο chip(Το chip απο μέσα)
3. Σύνδεση με CPU



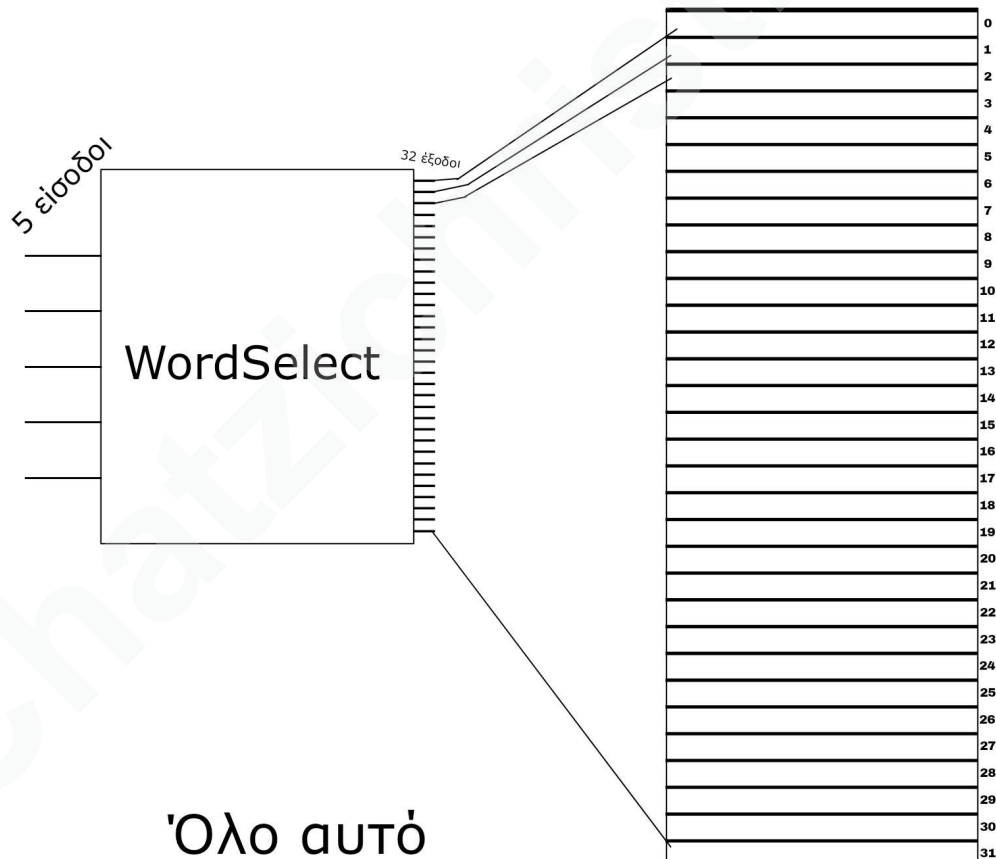
Το μέγεθος CS εξαρτάται από το πλήθος των Chip

Αν έχω να επιλέξω ανάμεσα σε X τσιπ, θέλω αποκωδικοποιητή με X εξόδους
άρα $X=4$ π.χ 4 chip 4 έξοδοι D0-D3
 $\log_2 4=2$ είσοδοι 8 chip 8 έξοδοι και $\log_2 8=3$ άρα 3x8

Οι εξοδοι λειτουργούν ως επιτρεψει
Το κάθε chip εχει ενα Wordselect

Αν το Chip εχει Y λέξεις το WordSelect πρέπει να εχει Y εξοδους και
 $\log_2 Y$ εισόδους

Στο παράδειγμα έχει $Y=32$ (λέξεις ανα Chip)
 $\log_2(32)=5$ Άρα $WS=5 \times 32$



Όλο αυτό
ονομάζεται Chip

Π.χ. Έστω η μνήμη του παραδείγματος

1. Ποσα bit απαιτούνται για την διευθυνσιοδότηση των λέξεων της μνήμης
2. Να αναλύσετε τη διεύθυνση μνήμης και πως χρησιμοποιούνται τα bit
3. Η CPU ζητάει να γράψει στην διεύθυνση 35. Να εξηγήσετε την διαδικασία

1. Μνήμη με 128 λέξεις (Να βρώ την δύναμη του 2 που δίνει $2^x=128$)
 $\log_2(128) = 7$ bit
2. Address 2 : A6-A0 Επειδη έχω 4 chip θέλω 2 bit για είσοδο στο CS. Άρα το (A6 και το A5) A6 και A5 τα πιο σημαντικά

Εχω 32 λέξεις/chip τα 5 τελευταία bit θα χρησιμοποιηθούν ως WS(A4,A3,A2,A1,A0). Τα A4-A0 θα μουν κοινές ως είσοδοι για όλα τα 4 WordSelect

MAR : Memory Address Register

Αποθηκεύει τη διεύθυνση μιας μνήμης που θα χρειαστεί η CPU για ανάγνωση η εγγραφή

MDR:Memory Data Register

Φέρνει δεδομένα από/προς την μνήμη

Η αποκωδικοποίηση γίνεται από τον MAR

Απάντηση στο 3(Η CPU ζητάει να γράψει στην διεύθυνση 35. Να εξηγήσετε την διαδικασία)

35:γράφεται με 7 bit γιατί το μήκος της διεύθυνσης της λέξης είναι 7 bit

01|00011 επειδή το 35=0100011 στην δυαδική μορφή

CS |Ws Για πολλές διευθύνσεις δεν χρειάζεται να κάνω το σχήμα το λύνω όπως το έλυσα εδώ.

Τώρα δείχνουμε την σύνδεση αποκωδικοποιητή με CPU και μνήμη

1) Ο MAR=35

2) Ο MAR περνάει την διεύθυνση 35 στο Address Bus

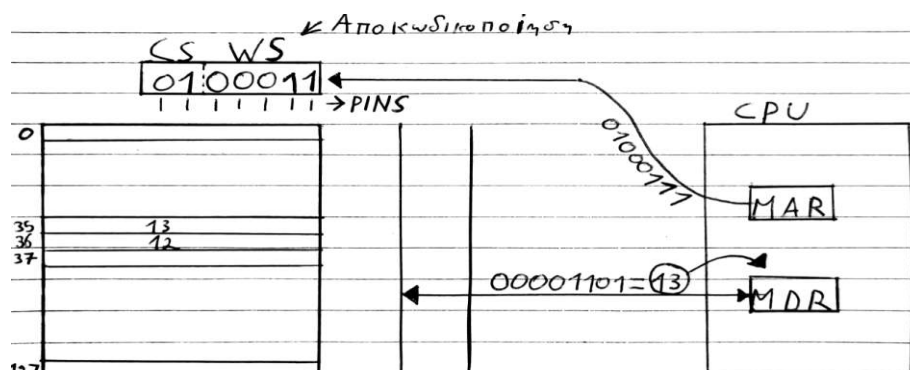
7 γραμμές bit Οι 2 πάνε στο ChipSelect:01

Οι 5 πάνε σε όλα τα WordSelect: 00011

Αρα 0100011

Και επίσης στέλνεται ένα σήμα R/W: 0→R

1→W



3) Η μνήμη στέλνει τα περιεχόμενα της λέξης 35 στο δίαυλο δεδομένων για να το διαβάσει ο MDR

Μικρολειτουργία \rightarrow $MDR \leftarrow M[MAR]$ —Μεταφράση—

Ο MDR θα πάρει απο την μνήμη τα περιεχόμενα που έχει ο MAR

SOS: Γενικά ο MDR λαμβάνει απο την μνήμη τα περιεχόμενα θέσης που υποδεικνύει ο MAR

Μόνο ο MDR μιλάει με μνήμη, όταν το 13 περάσει θα μπει από το MDR στον καταχωρητή

Άρα ο καταχωρητής δεν μιλάει μνήμη του ο MDR

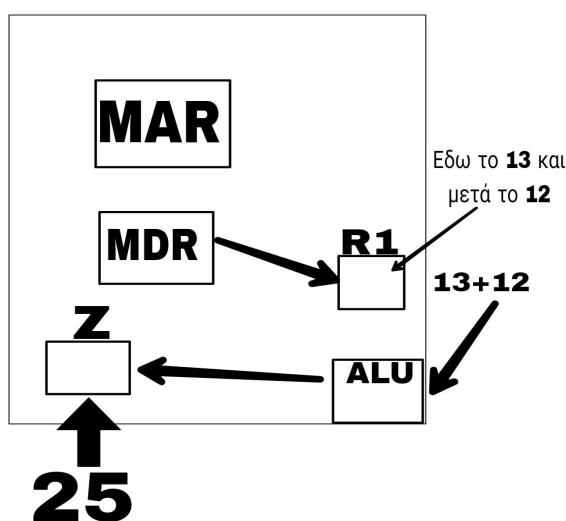
Με τον ίδιο τρόπο φέρνουμε στο MDR την τιμη 12 (θέση 36)

$MAR=01|00100 \rightarrow$ Δίαυλος δεδομένων

↓ CS|WS

↓

MDR, R1 θα πάνε στον αθροιστή, θα προστεθούν θα πάρει το αποτέλεσμα 25 και το αποτέλεσμα θα αποθηκευτεί σε έναν άλλον καταχωρητή, εστω Z



$$Z \leftarrow MDR + R1$$

$$25 \leftarrow 12 + 13$$

Τώρα εξήγηση της εγγραφής στην θέση 37

1)MAR=37

01|00101 Adress Bus μαζί με ενσα σήμα

CS| WS R/W=1

Ο Ζ δεν επικοινωνεί με το data bus, δίνει τα δεδομένα στον MDR και το data Bus

00011001 -> 25

Συνολικά

M[MAR]<-- MDR. Η θέση μνήμης που Υποδεικνύει ο MAR(37) θα πάρει τα περιεχόμενα του MDR(25)

ΠΑΡΑΔΕΙΓΜΑ 2

Δίνεται για RAM 32MB οργανωμένοι σε 8 chips τον 4mb. να δείξετε την αποκωδικοποίηση της λέξης M-1, αν κάθε λέξη έχει μέγεθος 2 bytes

η μνήμη έχει μέγεθος 32 mb για 8 chip

$32/8 = 4$ MB μέγεθος καθε chip

Κάθε λέξη 2 byte αρα καθε chip $4\text{mb}/2\text{byte} = 2\text{M}$ λέξεις σε καθε τσιπ

Συνολικά έχω $8 \times 2\text{m} = 16\text{m} = 2^{24}$

Αρα 24 Bit

3 bit για cs (8 chip) 8 chip αφού 3 bit είναι 000-111 το 111=8

WS $\rightarrow 21 \times 2^{21}$

εισόδοι x εξόδοι

$M-1 \rightarrow 2^{20} - 1 = 11111111111111111111$ (20 ασοι)

M-1= 000|11111111111111111111

↑ ↑
CS WS