



ΤΕΧΝΟΛΟΓΙΚΟ
ΕΚΠΑΙΔΕΥΤΙΚΟ
ΙΔΡΥΜΑ
ΤΕΙ ΗΠΕΙΡΟΥ

Αρχιτεκτονική Υπολογιστών

Ενδεικτικές ασκήσεις-απαντήσεις

Βαρζιώτης Φώτης, Καθηγητής Εφαρμογών Τμ. Μηχανικών Πληροφορικής Τ.Ε.

Άδειες Χρήσης

Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons. Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα. Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Ηπείρου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.



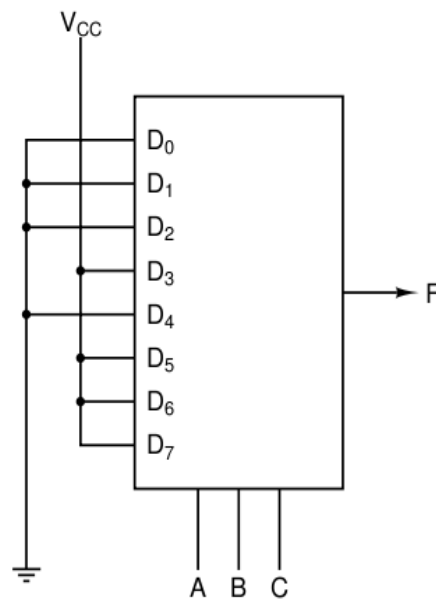
Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Επίπεδο Ψηφιακής Λογικής (The Digital Logic Level)

Ερωτήσεις Επανάληψης

- Ένας καθηγητής λογικής μπαίνει σε ένα εστιατόριο και λέει : “Θέλω ένα σάντουιτς ή ένα σουβλάκι και τηγανητές πατάτες”. Δυστυχώς, ο μάγειρας δεν είχε βγάλει το γυμνάσιο, και δεν ξέρει (ούτε θέλει να μάθει) αν έχει προτεραιότητα το “και” ή το “ή”. Όλες οι ερμηνείες του φαίνονται το ίδιο καλές. Ποίες από τις ακόλουθες περιπτώσεις είναι έγκυρες ερμηνείες της παραγγελίας; (Σημειώστε ότι στην φυσική γλώσσα το “ή” σημαίνει “αποκλειστικό ή”.)
 α. Μόνο σάντουιτς.
 β. Μόνο σουβλάκι.
 γ. Μόνο τηγανητές πατάτες.
 δ. Σουβλάκι και τηγανητές πατάτες.
 ε. Σάντουιτς και τηγανητές πατάτες.
 στ. Σουβλάκι και σάντουιτς.
 ζ. Και τα τρία.
 η. Τίποτα από τα παραπάνω.
- Ένας ιεραπόστολος που έχει χαθεί στην νότια Καλιφόρνια σταματά σε ένα σταυροδρόμι. Ξέρει ότι στην περιοχή υπάρχουν δύο συμμορίες μοτοσικλετιστών, από τις οποίες η μία λέει πάντα την αλήθεια και η άλλη λέει πάντα ψέματα. Θέλει να μάθει ποίος δρόμος οδηγεί στην Disneyland. Τι ερώτηση πρέπει να κάνει;
- Χρησιμοποιήστε έναν πίνακα αληθείας για να δείξετε ότι $P = (P \text{ AND } Q) \text{ OR } (P \text{ AND NOT } Q)$.
- Υπάρχουν τέσσερις συναρτήσεις Boole μίας μεταβλητής και 16 συναρτήσεις δύο μεταβλητών, Πόσες συναρτήσεις τριών μεταβλητών υπάρχουν; Πόσες συναρτήσεις n μεταβλητών;
- Δείξτε πώς μπορεί να κατασκευαστεί η συνάρτηση AND από δύο πύλες NAND.
- Χρησιμοποιώντας το τσιπ πολυπλέκτη τριών μεταβλητών της εικόνας 3-12, υλοποιήστε μία συνάρτηση που η έξοδος της είναι η ισοτιμία (parity) των εισόδων της, δηλαδή, η έξοδος είναι 1 εάν και μόνο εάν **άρτιος** αριθμός εισόδων είναι 1.

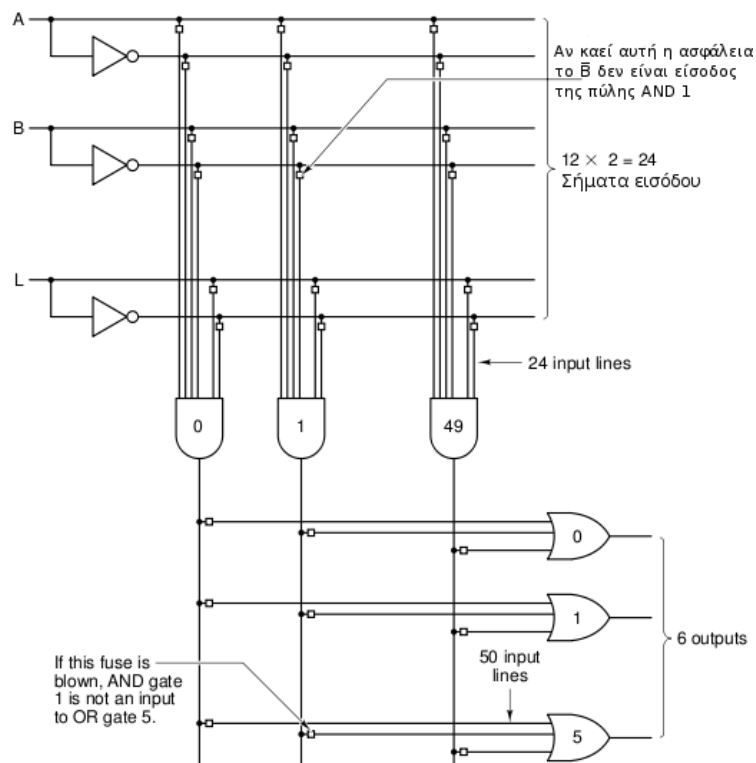


(b)

7. Βάλτε το μυαλό σας να δουλέψει. Το τσιπ πολυπλέκτη τριών μεταβλητών της εικόνας 3-12 μπορεί στην πραγματικότητα να υπολογίσει οποιαδήποτε συνάρτηση τεσσάρων μεταβλητών Boole. Περιγράψτε πώς, και δώστε ένα παράδειγμα, σχεδιάζοντας το λογικό διάγραμμα μιας συνάρτησης η οποία είναι 0 εάν ο αριθμός που αντιστοιχεί στη γραμμή του πίνακα αληθείας αντιστοιχεί σε λέξη με άρτιο αριθμό γραμμάτων και 1 εάν έχει περιττό αριθμό γραμμάτων (π.χ. 0000 = μηδέν = πέντε γράμματα => 1, 0111 = επτά = τέσσερα γράμματα => 0, 1101 = δεκατρία = οκτώ γράμματα => 0).
Υπόδειξη: Αν ονομάσουμε την τέταρτη μεταβλητή εισόδου D, οι οκτώ γραμμές εισόδου μπορούν να χρησιμοποιηθούν για να υλοποιηθούν σωστά τη τέταρτη μεταβλητή με τη βοήθεια της D του Vcc (1) και της γείωσης (0).
8. Σχεδιάστε το λογικό διάγραμμα ενός αποπολυπλέκτη των 2 bit, δηλαδή ενός κυκλώματος του οποίου η μοναδική γραμμή εισόδου οδηγείται σε μία από τις τέσσερις γραμμές εξόδου, ανάλογα με την κατάσταση των δύο γραμμών ελέγχου.
9. Σχεδιάστε το λογικό διαγράμματος κωδικοποιητή των 2 bit, δηλαδή ενός κυκλώματος με τέσσερις γραμμές εισόδου από τις οποίες ακριβώς μία είναι στην υψηλή κατάσταση οποιαδήποτε δεδομένη στιγμή, και με δύο γραμμές εξόδου των οποίων η δυαδική τιμή των 2 bit προσδιορίζει ποια είσοδος είναι στην υψηλή κατάσταση.
10. Ξανασχεδιάστε το PLA της εικόνας 3-15 με αρκετές λεπτομέρειες, για να δείξετε πως μπορεί να υλοποιηθεί η λογική συνάρτηση πλειοψηφίας της εικόνα 3-3. Ειδικότερα, φροντίστε να φανείτε ποιες συνδέσεις υπάρχουν και στα δύο μητρώα.

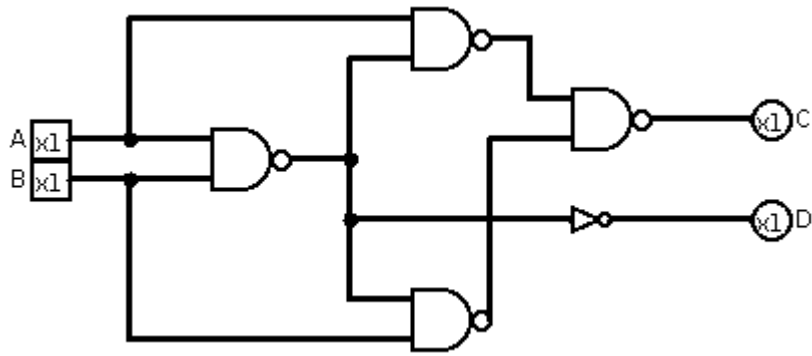
A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Εικόνα 3-3



Εικόνα 3-15

11. Τι κάνει το παρακάτω κύκλωμα;



12. Ένα κοινό τσιπ MSI είναι ένας αθροιστής των 4 bit. Τέσσερα τέτοια τσιπ μπορούν να συνδεθούν για να σχηματίσουν έναν αθροιστή των 16 bit. Πόσους ακροδέκτες νομίζετε ότι θα έχει το τσιπ αθροιστή των 4 bit; Γιατί;
13. Ένας αθροιστής των n bit μπορεί να κατασκευαστεί με επάλληλη τοποθέτηση n πλήρων αθροιστών σε σειρά, όπου το εισερχόμενο κρατούμενο του σταδίου i , το C_i , προέρχεται από την έξοδο του σταδίου $i-1$. Το εισερχόμενο κρατούμενο του σταδίου 0, το C_0 , είναι 0. Αν κάθε στάδιο χρειάζεστε T nsec για να δώσει το άθροισμα και το κρατούμενο του το εισερχόμενο κρατούμενο του σταδίου i δεν θα είναι έγκυρο μέχρι να περάσουν iT nsec από την αρχή της πρόσθεσης. Για μεγάλα n , ο χρόνος που χρειάζεται για να διαδοθεί το κρατούμενο μέχρι το στάδιο υψηλής τάξης μπορεί να είναι απαράδεκτα μεγάλος. Σχεδιάστε ένα αθροιστή που να δουλεύει γρηγορότερα. Υπόδειξη: Κάθε C_i μπορεί να εκφραστεί ως συνάρτηση των bit-τελεστών A_{i-1} και B_{i-1} καθώς και του κρατούμενου C_{i-1} . Χρησιμοποιώντας αυτήν την σχέση, μπορείτε να εκφράσετε το C_i ως συνάρτηση των εισόδων των σταδίων από 0 έως $i-1$, ώστε όλα τα κρατούμενα να παράγονται ταυτόχρονα.
14. Αν όλες οι πύλες της εικόνας 3-19 έχουν καθυστέρηση μετάδοσης 1 nsec, και όλες οι άλλες καθυστερήσεις μπορούν να αγνοηθούν, ποιος είναι ο συντομότερος χρόνος στον οποίο ένα κύκλωμα που χρησιμοποιεί αυτόν τον σχεδιασμό θα έχει σίγουρα ένα έγκυρο bit εξόδου;

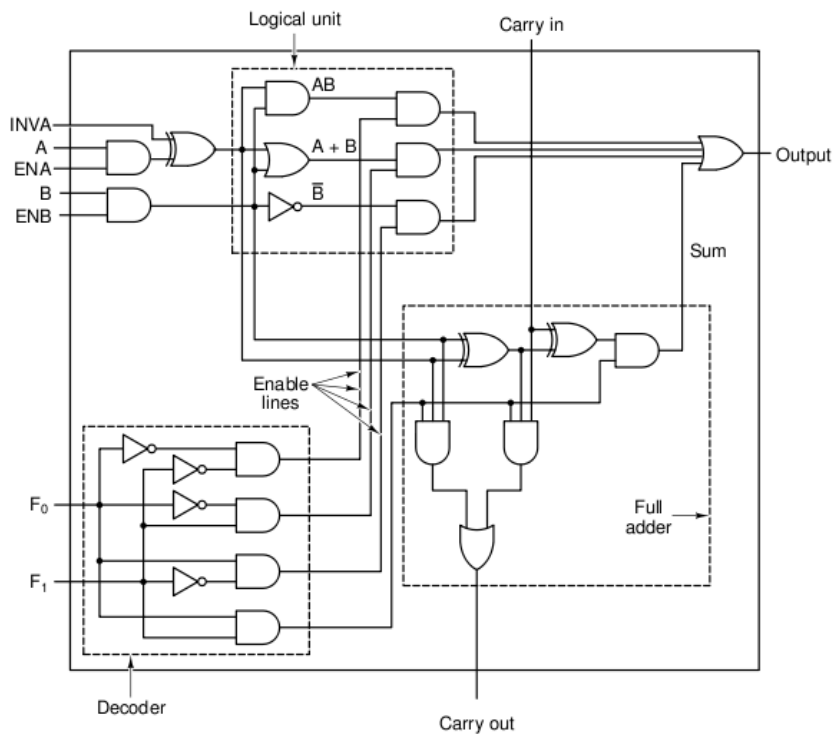
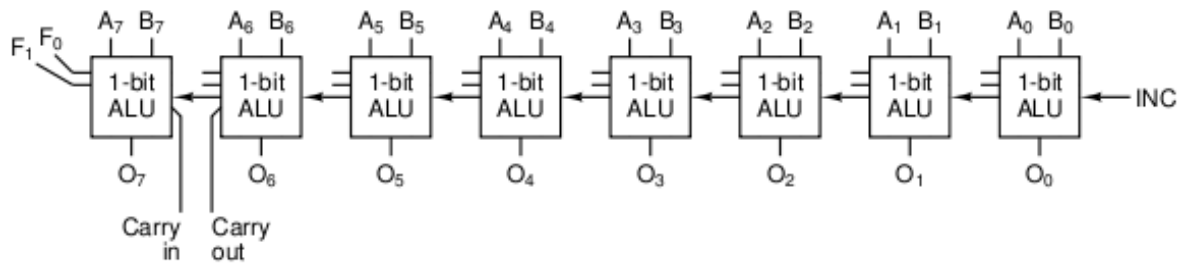


Figure 3-19. A 1-bit ALU.

15. Μερικές φορές, είναι χρήσιμο μία ALU των 8 bit, σαν εκείνη της εικόνας 3-20, να δίνει ως έξοδο την σταθερά -1. Βρείτε δύο διαφορετικούς τρόπους με τους οποίους μπορεί να γίνει αυτό. Για τον κάθε τρόπο, προσδιορίστε τις τιμές των έξι σημάτων ελέγχου.



16. Μία ALU των 16 bit αποτελείται από 16 ALU του 1 bit, κάθε μία από τις οποίες έχει χρόνο πρόσθεσης 10 nsec. Αν υπάρχει μία πρόσθετη καθυστέρηση 1 nsec για διάδοση από την κάθε ALU στην επόμενη, πόσος χρόνος χρειάζεται για να εμφανιστεί το αποτέλεσμα μίας πρόσθεσης των 16 bit;
17. Ποία είναι η κατάσταση ηρεμίας των εισόδων S και R ενός κυκλώματος μανδάλωσης SR που αποτελείται από δύο πύλες NAND;
18. Το κύκλωμα της εικόνας 3-26 είναι ένα δισταθές κύκλωμα που ενεργοποιείται στην ακμή ανόδου του παλμού του ρολογιού. Τροποποιήστε για να δημιουργήσετε ένα δισταθές κύκλωμα που να ενεργοποιείται στην ακμή καθόδου.

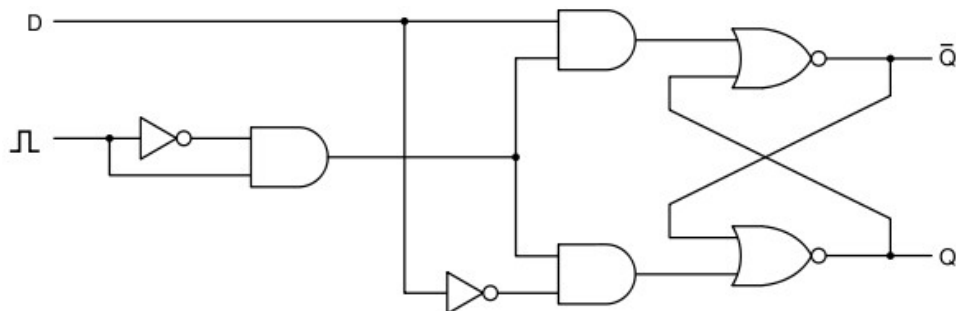
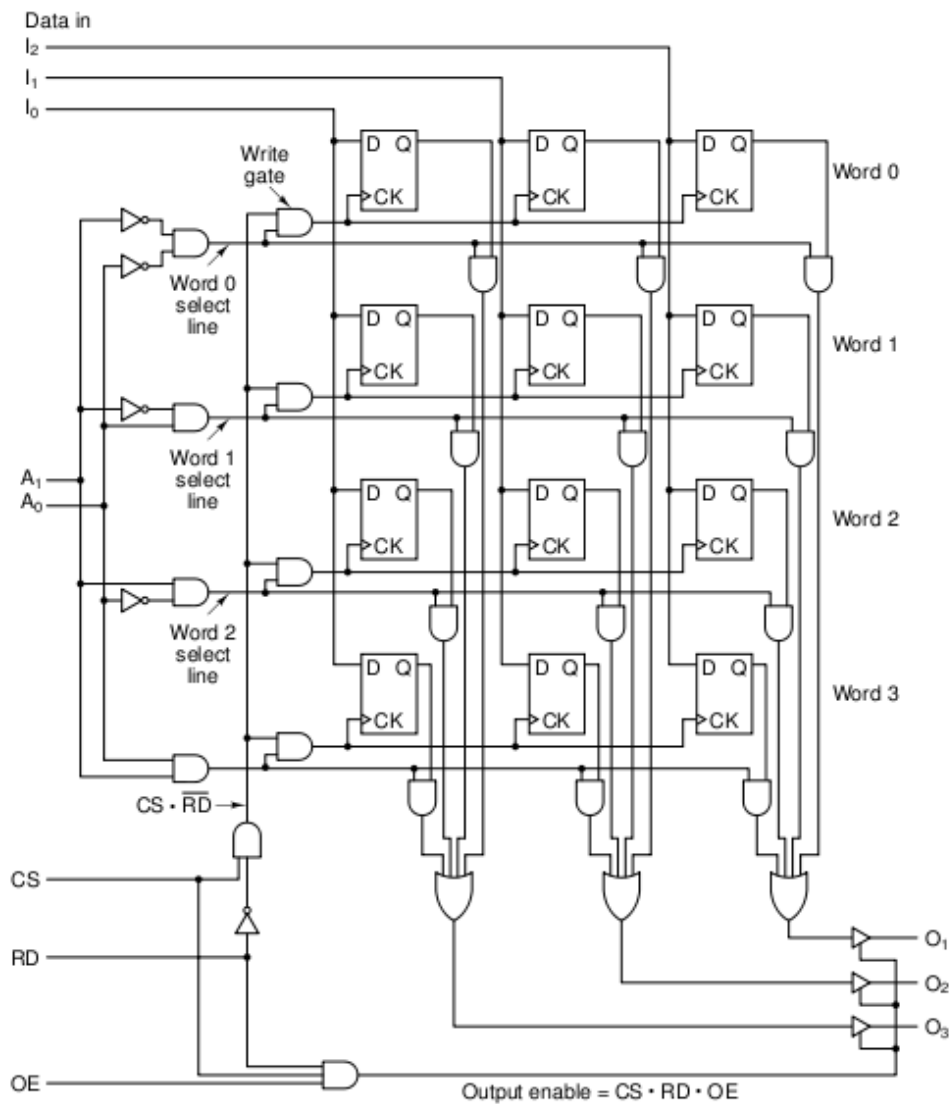
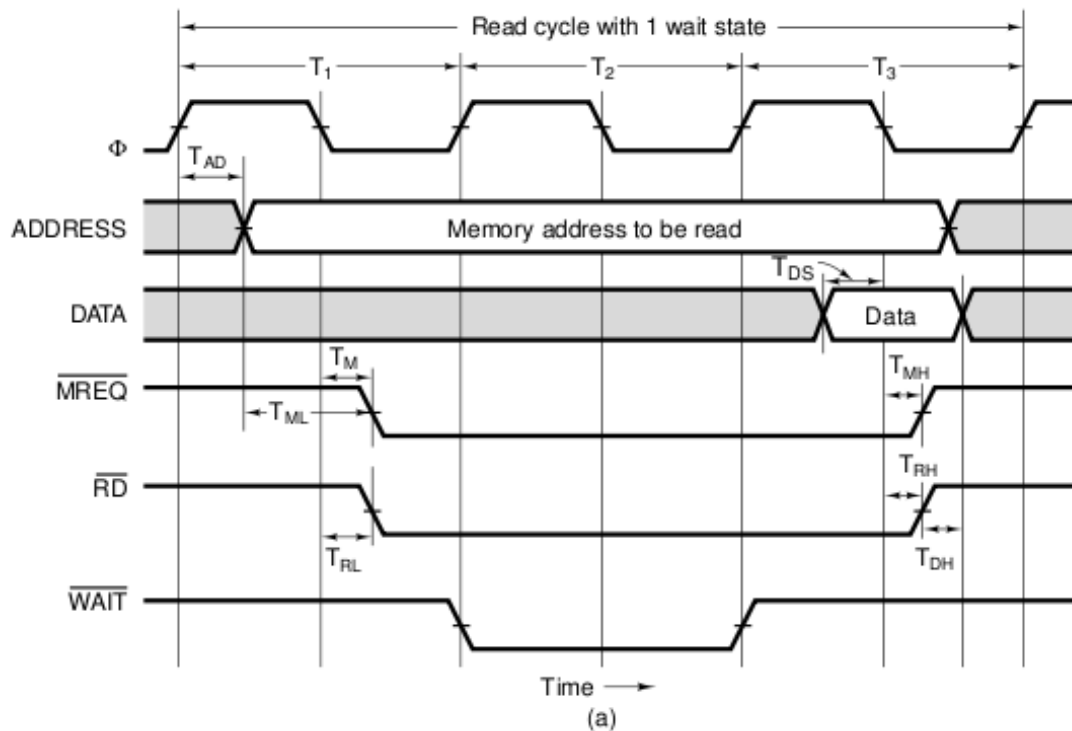


Figure 3-26. A D flip-flop.

19. Για να πληρώσετε τις δόσεις του καινούργιου σας προσωπικού υπολογιστή, κάνετε το σύμβουλο για νέους κατασκευαστές τσιπ SSI. Ένας από τους πελάτες σας, σκέφτεται να κυκλοφορήσει ένα τσιπ που θα περιέχει τέσσερα δισταθή κύκλωμα D, το κάθε ένα με γραμμές Q και \bar{Q} , κατ' απαίτηση ενός σημαντικού υποψήφιου πελάτη του. Ο προτεινόμενος συνδυασμός έχει και τα τέσσερα σήματα ρολογιού συνδεδεμένα μαζί, επίσης κατ' απαίτηση του πελάτη. Δεν υπάρχει ούτε γραμμή φόρτισης (preset) ούτε γραμμή καθαρισμού (clear). Η δουλειά σας είναι να κάνετε μια επαγγελματική αξιολόγηση του σχεδιασμού.
20. Η μνήμη 4 x 3 της εικόνας 3-29 χρησιμοποιεί 22 πύλες AND και τρεις πύλες OR. Πόσες πύλες από το κάθε είδος χρειάζονται για να επεκταθεί το κύκλωμα σε 256 x 8;



21. Καθώς τοποθετείται όλο και περισσότερο μνήμη σε ένα τσιπ, ο αριθμός των ακροδεκτών που χρειάζονται για την διευθυνσιοδότηση της αυξάνεται. Συχνά, δεν είναι βολικό να υπάρχουν πάρα πολλοί ακροδέκτες διευθύνσεων σε ένα τσιπ. Επινοήστε ένα τρόπο για να απευθύνεστε σε 2^n λέξεις μνήμης χρησιμοποιώντας λιγότερους από n ακροδέκτες.
22. Ένας υπολογιστής με δίαυλο δεδομένων εύρους 32 bit χρησιμοποιεί τσιπ δυναμικής μνήμης RAM 1M x 1. Πόση είναι η μικρότερη μνήμη που μπορεί να έχει αυτός ο υπολογιστής;
23. Στο διάγραμμα χρονισμού της εικόνας 3-37, ας υποθέσουμε ότι επιβραδύνουμε το ρολόι ώστε να έχει περίοδο 40 nsec αντί 25 nsec, αλλά οι περιορισμοί του χρονισμού παραμένουν αμετάβλητοι. Πόσο χρόνο θα έχει η μνήμη για να τοποθετήσει τα δεδομένα στο δίαυλο κατά τον κύκλο T_3 αφού γίνει θετικό το \overline{MREQ} , στη χειρότερη περίπτωση;



Symbol	Parameter	Min	Max	Unit
T_{AD}	Address output delay		11	nsec
T_{ML}	Address stable prior to \overline{MREQ}	6		nsec
T_M	\overline{MREQ} delay from falling edge of Φ in T_1		8	nsec
T_{RL}	\overline{RD} delay from falling edge of Φ in T_1		8	nsec
T_{DS}	Data setup time prior to falling edge of Φ	5		nsec
T_{MH}	\overline{MREQ} delay from falling edge of Φ in T_3		8	nsec
T_{RH}	\overline{RD} delay from falling edge of Φ in T_3		8	nsec
T_{DH}	Data hold time from negation of \overline{RD}	0		nsec

(b)

3-37

24. Και πάλι στην εικόνα 3-37, ας υποθέσουμε ότι το ρολόι παραμένει στα 40 Mhz, αλλά το T_{DS} αυξάνεται στα 16 nsec. Θα μπορούσαν πάλι να χρησιμοποιηθούν τσιπ μνήμης των 40 nsec;
25. Υποθέστε ότι η μεταφορά ενότητας (block transfer) της εικόνας 3-41 γίνεται πάνω στο δίαυλο της εικόνας 3-37. Πόσο μεγαλύτερο εύρος ζώνης θα έχουμε αν χρησιμοποιήσουμε μεταφορά ενότητας αντί για μεμονωμένες μεταφορές για τις μεγάλες ενότητες; Υποθέστε τώρα ότι ο δίαυλος έχει εύρος 32 bit αντί 8 bit. Απαντήστε πάλι στην ίδια ερώτηση.

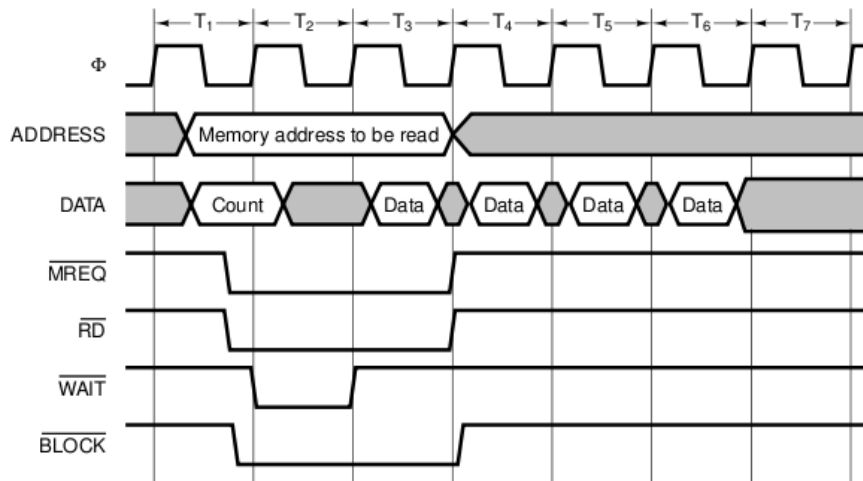


Figure 3-41. A block transfer.

26. Οι περισσότεροι δίαυλοι των 32 bit επιτρέπουν αναγνώσεις και εγγραφές των 16 bit. Υπάρχει καμία ασάφεια σχετικά με το που θα τοποθετηθούν τα δεδομένα; Επιχειρηματολογήστε.
27. Πολλές CPU έχουν έναν ειδικό τύπο κύκλου διαύλου για επιβεβαίωση των διακοπών. Γιατί;
28. Ένας 64-bit υπολογιστής με 200 Mhz δίαυλο χρειάζεται τέσσερις κύκλους για να διαβάσει μία λέξη 64-bit. Πόσο εύρος ζώνης του διαύλου καταναλώνει η CPU;

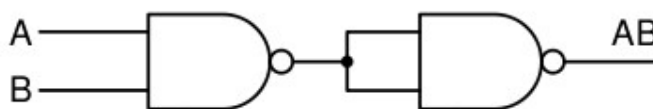
Απαντήσεις

1. Η παραγγελία είναι “σάντουιτς ή ένα σουβλάκι και τηγανητές πατάτες” η οποία μπορεί να αναλυθεί ως
 “σάντουιτς ή (σουβλάκι και τηγανητές πατάτες)”
 ή ως
 “(σάντουιτς ή σουβλάκι) και τηγανητές πατάτες”.
 Ο πρώτος κανόνας ικανοποιεί το ερώτημα α και δ. Ο δεύτερος τα ε και δ. Έτσι α, δ και ε είναι οι πιθανές απαντήσεις.
2. Θα πρέπει να εστιάσει σε έναν από τους δρόμους και να ρωτήσει: “Εάν επρόκειτο να ρωτήσω την άλλη συμμορία εάν αυτός είναι ο δρόμος που οδηγεί στην Disneyland, τι θα απαντούσαν;” Εάν η απάντηση είναι όχι, πρέπει να πάρει το δρόμο εάν η απάντηση είναι ναι, δεν πρέπει να την πάρει.
3. Ο πίνακας αληθείας είναι ο ακόλουθος:

P	Q	P AND Q	P AND NOT Q	(P AND Q) OR (P AND NOT Q)
0	0	0	0	0
0	1	0	0	0
1	0	0	1	1
1	1	1	0	1

Άρα από τον παραπάνω πίνακα βλέπουμε ότι η σχέση $P = (P \text{ AND } Q) \text{ OR } (P \text{ AND NOT } Q)$ ισχύει.

4. Με τρεις μεταβλητές, ο πίνακας αληθείας θα έχει 8 γραμμές, και η συνάρτηση μπορεί να εκφραστεί από ένα 8-bit αριθμό. Έτσι υπάρχουν $2^8 = 256$ συναρτήσεις, οι οποίες αντιστοιχούν στις 256 δυνατές ακολουθίες αποτελεσμάτων 8-bit. Με n μεταβλητές, ο πίνακας αληθείας θα έχει $k = 2^n$ γραμμές και θα υπάρχουν 2^k συναρτήσεις.
5. Έχουμε τις δύο μεταβλητές A και B. Τις συνδέουμε στην πρώτη πύλη NAND. Παίρνουμε την έξοδο της πρώτης πύλης και τροφοδοτούμε τις δύο εισόδους της δεύτερης πύλης NAND. Έτσι κατασκευάζουμε την πύλη AND από 2 πύλες NAND.



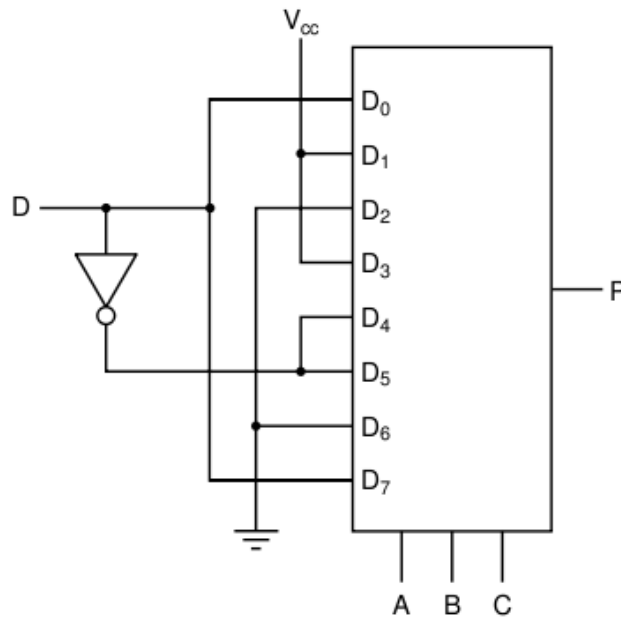
6. Ο πίνακας αληθείας της συνάρτησης είναι:

A	B	C	M
0	0	0	1
0	0	1	0
0	1	0	0

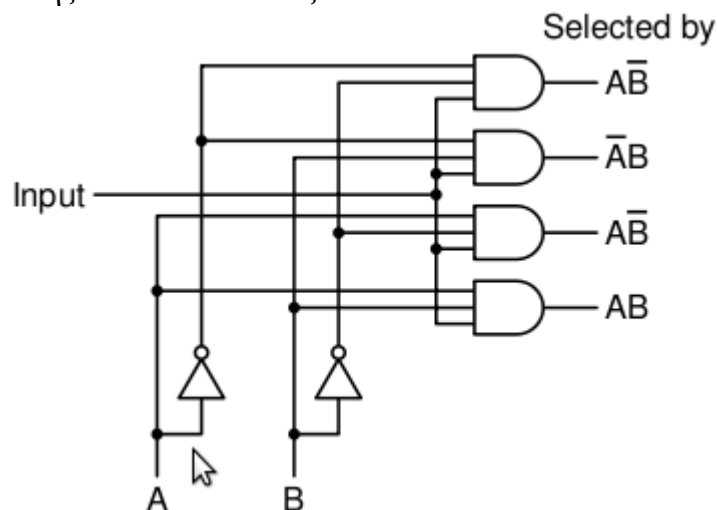
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Οι είσοδοι D1 , D2 , D4 και D7 συνδέονται στην γείωση και οι υπόλοιπες τέσσερις είσοδοι συνδέονται στο Vcc.

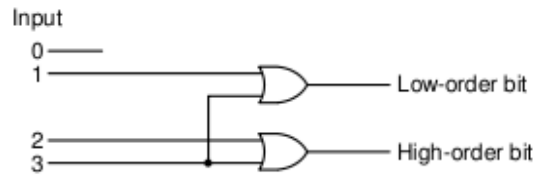
7. Η είσοδος D0 τροφοδοτεί την έξοδο για τον πίνακα αληθείας στις γραμμές 0000 και 0001. Η είσοδος D1 τροφοδοτεί την έξοδο για τον πίνακα αληθείας στις γραμμές 0010 και 0011, και ούτω καθεξής. Για κάθε περίπτωση, οι τιμές της συνάρτησης για τις δύο γραμμές μπορεί να είναι 00, 01, 10 και 11. Εάν είναι 00, απλά συνδέουμε το καλώδιο με την γείωση; αν είναι 11, απλά το συνδέουμε με το Vcc. Εάν είναι 01, παρατηρήστε ότι είναι ακριβώς το ίδιο με την τέταρτη μεταβλητή εισόδου, έτσι το συνδέουμε με το D. Εάν είναι 10, το συνδέουμε με το D. Ο πίνακας αληθείας για αυτό το παράδειγμα, έχει τιμές από 0000 έως 1111, 0111001110100001.



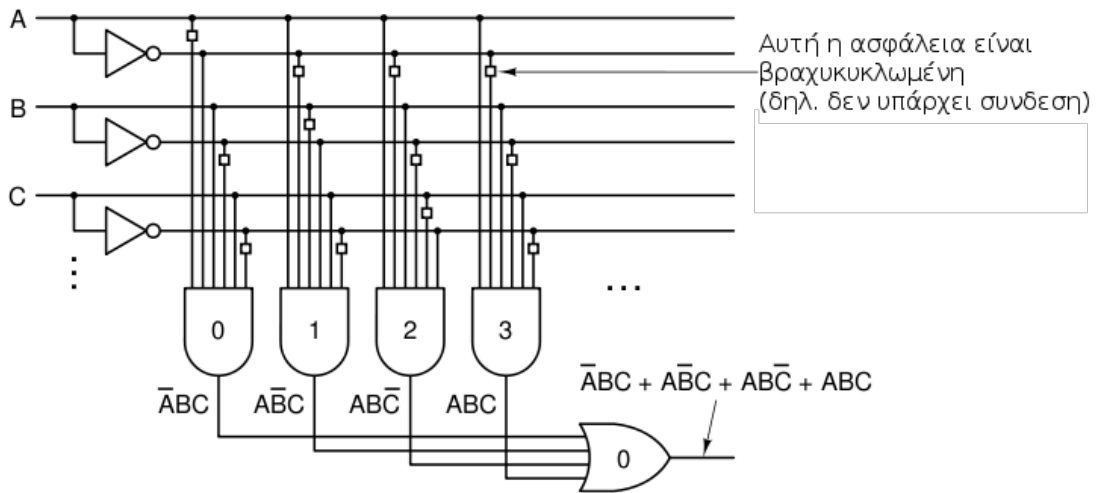
8. Ο αποπολυπλέκτης είναι ο ακόλουθος :



9. Ο κωδικοποιητής είναι ο παρακάτω. Βλέπουμε ότι η είσοδος 0 δεν χρησιμοποιείται.



10. Το σχετικό τμήμα του PLA έχει ως εξής. Όπου υπάρχουν κουτάκια, αυτό σημαίνει ότι εκείνα τα σημεία είναι βραχυκυκλωμένα και από κει δεν περνάει η τάση.



11. Το κύκλωμα είναι ένας ημιαθροιστής (half adder), με την έξοδο C ως το άθροισμα (sum) και την έξοδο D ως το υπόλοιπο (carry).

12. Το τσιπ αθροιστή 4-bit χρειάζεται τέσσερις ακροδέκτες για τον πρώτο τελεστή, τέσσερις ακροδέκτες για το δεύτερο τελεστή, τέσσερις ακροδέκτες για το αποτέλεσμα, ένα ακροδέκτη για το εισερχόμενο κρατούμενο και ένα ακροδέκτη για το εξερχόμενο. Επίσης χρειάζεστε τροφοδοσία και γείωση, άρα σύνολο από 16 ακροδέκτες.

13. Το κρατούμενο του σταδίου i μπορεί να εκφραστεί ως $C_i = P_{i-1} + S_{i-1}C_{i-1}$, όπου P_{i-1} είναι το αποτέλεσμα του $A_{i-1}B_{i-1}$ και S_{i-1} είναι το άθροισμα των όρων $A_{i-1}+B_{i-1}$. Αυτό το αποτέλεσμα προκύπτει άμεσα από το γεγονός ότι το κρατούμενο παράγεται από ένα βήμα εάν και οι δύο τελεστές είναι 1, ή εάν ένας από του τελεστές και το κρατούμενο είναι και οι δύο 1, για παράδειγμα:

$$C_0 = 0$$

$$C_1 = P_0 + S_0C_0 = P_0$$

$$C_2 = P_1 + S_1C_1 = P_1 + P_0S_1$$

$$C_3 = P_2 + S_2C_2 = P_2 + P_1S_2 + P_0S_1S_2$$

$$C_4 = P_3 + S_3C_3 = P_3 + P_2S_3 + P_1S_2S_3 + P_0S_1S_2S_3$$

Μόλις οι είσοδοι A και B είναι διαθέσιμοι, όλα οι όροι P και S μπορούν να παραχθούν ταυτόχρονα με καθυστέρηση μια πύλης. Στην συνέχεια οι διάφοροι όροι AND όπως $P_0S_1S_2$ μπορούν να παραχθούν με μία δεύτερη καθυστέρηση πύλης. Τέλος, όλα τα κρατούμενα είναι διαθέσιμα με μία τρίτη καθυστέρηση πύλης. Έτσι όλα τα κρατούμενα είναι διαθέσιμα μετά από καθυστέρηση τριών πυλών, χωρίς να παίζει ρόλο πόσα στάδια έχει ο αθροιστής. Το κόστος όμως για την ταχύτερη λειτουργία έχει να κάνει με την τοποθέτηση πρόσθετων πυλών.

14. Το χρονοδιάγραμμα του κυκλώματος μπορεί να βρεθεί γράφοντας το 0 σε κάθε μία από τις

γραμμές εισόδου, και έπειτα εντοπίζοντας τες μέσω του κυκλώματος, προσθέτοντας το 1 σε κάθε πύλη. Η είσοδος A παίρνει 2 nsec για να γίνει διαθέσιμη, έτσι η έξοδος της λογικής μονάδας παίρνει 4 nsec και η τελική έξοδος για μία Boolean λειτουργία παίρνει 5 sec. Οι γραμμές αποκωδικοποιητή που οδηγούν στην λογική μονάδα έχουν από δύο καθυστερήσεις πυλών, έτσι οι ενεργές γραμμές σετάρονται σε πολύ χρόνο. Ο αθροιστής επίσης παίρνει 3 nsec για να παράγει τη συμβολή του στην πύλη εξόδου μετά την απόκτηση του A. Η χειρότερη περίπτωση σε όλο το κύκλωμα είναι 6 nsec.

15. Μια λύση είναι να εισάγουμε $ENB = 0$, θέτοντας ταυτόχρονα B ίσο με 0, και επιλέγοντας κωδικό λειτουργίας (function select) 01 ώστε το B να αποτελεί την έξοδο της ALU. Το αποτέλεσμα θα είναι όλα τα bits ίσα με 1 (δηλαδή, το -1 στο συμπλήρωμα του 2). Το INC (Carry In) μηδενίζεται και οι γραμμές ελέγχου του A είναι αδιάφορες. Ένας δεύτερος τρόπος να θέσουμε $A = 1$ και να το αντιστρέψουμε ($ENA, INVA = 1$). Επίσης θέτουμε $ENB = 0$ ώστε $B = 0$. Με $A = -1$ και $B = 0$, μπορούμε να κάνουμε είτε πρόσθεση ή OR.
16. Ένας βασικός κύκλος είναι 11 nsec, συμπεριλαμβανομένης της διάδοσης. Δεκαέξι κύκλοι είναι 176 nsec. Ωστόσο, η τελευταία διάδοση δεν χρειάζεται, έτσι η σωστή απάντηση είναι 175 nsec μετά την εκκίνηση.
17. Ένα κύκλωμα μανδάλωσης SR (SR latch) κατασκευασμένο με πύλες NAND, είναι δομημένο με τον ίδιο τρόπο που είναι ένα και με πύλες NOR. Κανονικά, οι δύο εισοδοί θα πρέπει να είναι 1 για να επιτευχθεί η συνέπεια μεταξύ των εισόδων και των εξόδων.
18. Χρησιμοποιήστε το ίδιο κύκλωμα, αντικαθιστώντας την πύλη AND στην γεννήτρια παλμών με μία πύλη NOR.
19. Δεν είναι αποδοτικό. Κάθε ένα από τα τέσσερα δισταθή κυκλώματα D χρειάζονται τρεις ακροδέκτες για D, Q και \bar{Q} . Επιπλέον το τσιπ χρειάζεται ρολόι, τροφοδοσία και γείωση. Αυτό σημαίνει ότι θα χρειαστείτε 15 ακροδέκτες, και θα πρέπει να βγάλετε σε πρώτη παγκόσμια κυκλοφορία τέτοιο τσιπ 15 ακροδεκτών, ή να χρησιμοποιήσετε ένα υπάρχον με 16 ακροδέκτες, αναξιοποιώντας έτσι ένα ακροδέκτη. Κατ' αυτό τον τρόπο η σχεδίαση δεν είναι πολύ αποτελεσματική. Δεδομένου ότι ένας ακροδέκτης μας περισσεύει και είναι διαθέσιμος μπορούμε να τον αξιοποιήσουμε κάνοντας έτσι το τσιπ πιο αποδοτικό. Για παράδειγμα θα μπορούσαμε να αξιοποιήσουμε τον δέκατο έκτο ακροδέκτη τοποθετώντας την επαναφορά (reset) για τα δισταθή κυκλώματα.
20. Ο σχεδιασμός χρησιμοποιεί δυο πύλες AND για την ενεργοποίηση εξόδου και άλλες δύο AND πύλες ανά γραμμή επιλογής λέξης και τέλος μία AND πύλη ανά bit δεδομένων. Για μνήμη 256×8 οι AND πύλες είναι $2 + 2 * 256 + 8 * 256 = 2 + 512 + 2048 = 2562$. Το κύκλωμα επίσης χρησιμοποιεί μία πύλη OR για κάθε bit στην κάθε λέξη, άρα για λέξεις των 8 bit που αναφέρεται η ερώτηση, θα χρειαστούν 8 πύλες OR.
21. Οι ακροδέκτες μπορούν να πολυπλεχθούν στον χρόνο. Για παράδειγμα, με n/2 ακροδέκτες, τα μισά από τα bits αναπαριστώνται στον επόμενο κύκλο. Πολλές RAM λειτουργούν με αυτόν τον τρόπο. Ακόμη ποιο ακραίο είναι να τροφοδοτούμε την διεύθυνση σειριακά στο τσιπ, bit κάθε φορά χρησιμοποιώντας έναν ακροδέκτη.
22. Ο δίαυλος δεδομένων εύρους 32 bit, σημαίνει ότι θα χρησιμοποιήσει 32 τσιπ παράλληλα, κάθε τσιπ παρέχει 1 bit. Έτσι η μικρότερη μνήμη που αποτελείται από 32 τσιπ, είναι 32 Megabits ή 4 Mbytes.
23. Με περίοδο ρολογιού 40 nsec, το \overline{MREQ} μπορεί να επιβεβαιωθεί το αργότερο σε 28 nsec στον κύκλο T1. Τα δεδομένα λαμβάνονται 5 nsec πριν από την ακμή καθόδου στον κύκλο T3, η οποία καθοδος συμβαίνει 20 nsec μετά την έναρξη του κύκλου. Από το μέσο του κύκλου T1 έως το μέσο του κύκλου T3 είναι 80 nsec. Δεδομένου ότι η μνήμη δεν μπορεί να ξεκινήσει έως και 8 nsec μετά την μετάβαση στον πρώτο κύκλο, και πρέπει να ολοκληρωθεί 5 nsec πριν από τη μετάβαση στον τρίτο κύκλο, στη χειρότερη περίπτωση η μνήμη έχει 67 nsec για να αποκριθεί.

24. Η μνήμη θα είχε τώρα $12.5 * 4 - 8 - 16 = 26$ nsec για να αποκριθεί αφότου ληφθούν τα MREQ και RD. Δεν υπάρχουν πολλά περιθώρια αριστερά, άλλα αν όλα τα τσιπ μπορούν να αποκριθούν σε 40 nsec, το σύστημα θα μπορεί ακόμη να λειτουργήσει.
25. Σε κανονική κατάσταση χρειάζονται τρεις κύκλοι ανά μεταφορά. Σε μεταφορά ενοτήτων, χρειάζεται ένας κύκλος ανά μεταφορά μόλις η μεταφορά ξεκινήσει. Συνεπώς, οι μεταφορές ενοτήτων (block transfers), έχουν περίπου 3 φορές το εύρος ζώνης. Το εύρος του διαύλου δεν παίζει ρόλο, έτσι είναι ακόμα τρεις φορές περισσότερο ακόμη και για 32-bit μεταφορές.
26. Ναι υπάρχει. Όταν διαβάζεται το κάτω μισό μιας λέξης, οι κάτω μισές γραμμές δεδομένων χρησιμοποιούνται. Ωστόσο, κατά την ανάγνωση του άνω μισού της λέξης, ο δίαυλος έχει επιλογή χρησιμοποίησης του πάνω μισού των γραμμών ή μετατοπίζοντας τα όλα προς τα κάτω έως το κάτω μισό.
27. Ο κύκλος επιβεβαίωσης διακοπών χρειάζεται ώστε η συσκευή που προκαλεί την διακοπή να μπορεί να δει ότι έγινε αποδεκτή και η CPU να έχει ένα διάνυσμα διακοπών (interrupt vector), για να βρει τη διεύθυνση της διαδικασίας που πρέπει να εκτελεστεί για την εξυπηρέτηση της διακοπής.
28. Στα 200 MHz, ο κύκλος είναι 5 nsec. Τέσσερις κύκλοι χρειάζονται 20 nsec. Μία λέξη είναι 64 bit, δηλαδή 8 bytes, έτσι ο υπολογιστής χρειάζεται 8 bytes κάθε 20 nsec, δηλαδή 400 Mbytes/sec.