

Επίπεδο Ψηφιακής Λογικής (The Digital Logic Level)

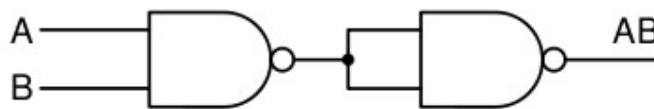
Απαντήσεις

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 να, 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.



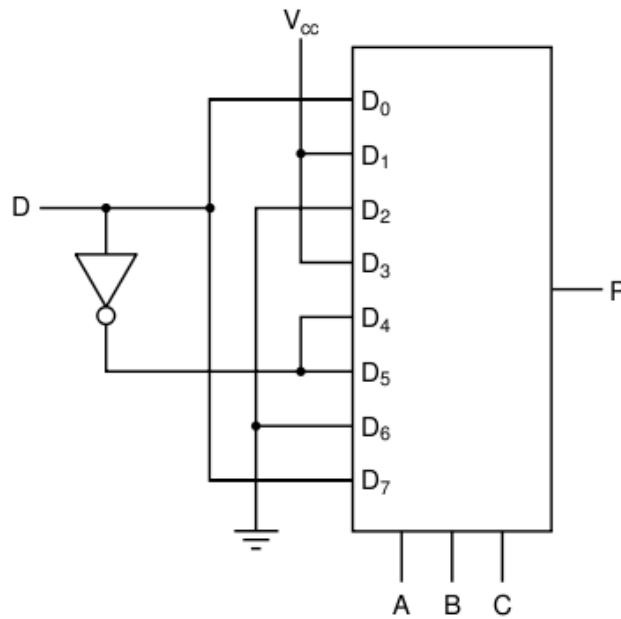
6. Ο πίνακας αληθείας της συνάρτησης είναι:

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

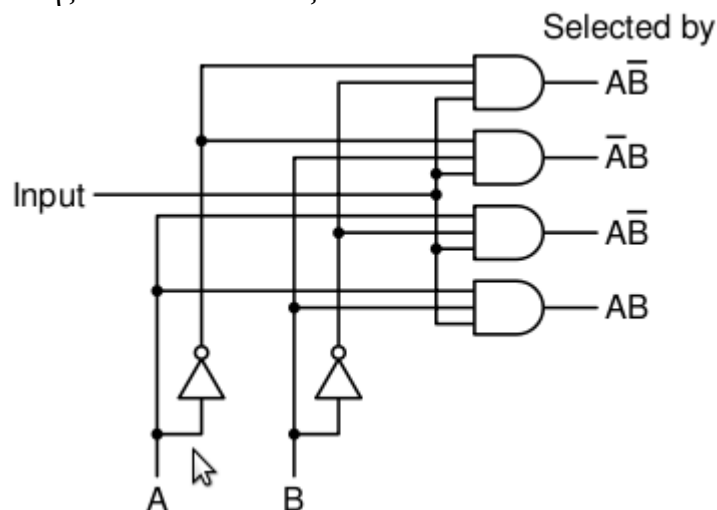
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

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

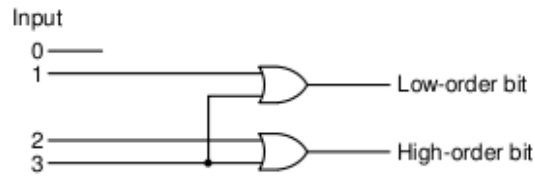
7. Η είσοδος D0 τροφοδοτεί την έξοδο για τον πίνακα αληθείας στις γραμμές 0000 και 0001. Η είσοδος D1 τροφοδοτεί την έξοδο για τον πίνακα αληθείας στις γραμμές 0010 και 0011, και ούτω καθεξής. Για κάθε περίπτωση, οι τιμές της συνάρτησης για τις δύο γραμμές μπορεί να είναι 00, 01, 10 και 11. Εάν είναι 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.



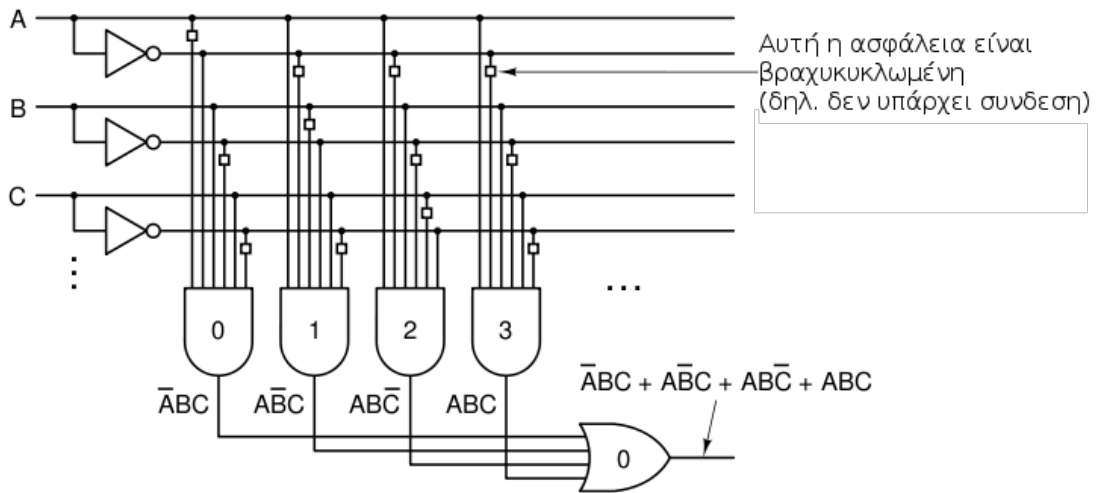
8. Ο αποπολυπλέκτης είναι ο ακόλουθος :



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 αποτελέσματα, 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 \overline{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.