

Στόχος

Υλοποιούμε

$$F=AB+CD+E$$

με NAND πύλες.

Για να το κάνουμε μόνο με NAND, ξέρουμε ότι τελικά θέλουμε όλα σε μορφή **NAND-NAND** ή **συμπληρωματική μορφή**.

1 Πριν τη NAND 3-εισόδων

Μέχρι εκεί έχουμε:

- $N1 = (A \text{ NAND } B) = (AB)'$
- $N2 = (C \text{ NAND } D) = (CD)'$
- $N4 = (E \text{ NAND } E) = E'$

Άρα, τα σήματα **N1**, **N2**, **N4** που φτάνουν στις εισόδους της NAND 3-εισόδων είναι:

$$N1=(AB)', N2=(CD)', N4=E'$$

2 Συνάρτηση μέχρι την είσοδο της 3-εισόδου NAND

Αν πάρουμε ό,τι έχουμε μέχρι εκεί, η "ενδιάμεση συνάρτηση" (πριν το τελικό NAND3) είναι:

$$G=N1 \cdot N2 \cdot N4=(AB)' \cdot (CD)' \cdot E'$$

ή με λόγια: **είναι το γινόμενο των συμπληρωμάτων των όρων AB, CD, και E.**

3 Μετά τη NAND 3-εισόδων

Η NAND 3-εισόδων παίρνει το παραπάνω και το αντιστρέφει:

$$F=(N1 \cdot N2 \cdot N4)'=((AB)' \cdot (CD)' \cdot E)'$$

Εφαρμόζουμε τον νόμο του De Morgan:

$$F=(AB)''+(CD)''+(E'')$$

Και επειδή το διπλό συμπλήρωμα ακυρώνεται:

$$F=AB+CD+E$$

👉 Έτσι παίρνουμε την αρχική συνάρτηση πίσω!

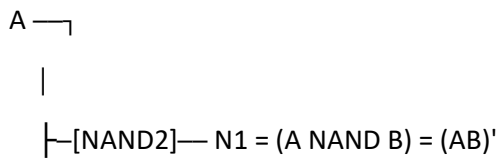
4 Με λόγια (πιο απλά)

- Πριν τη NAND3, η συνάρτηση είναι στη μορφή **AND** μεταξύ *αντιστραμμένων όρων* (όλα negated).

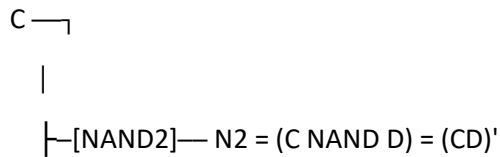
- Μετά τη NAND3, αυτή η **AND** αντιστρέφεται ξανά → γίνεται **OR** μεταξύ των αρχικών όρων.
- Δηλαδή το κύκλωμα κάνει ακριβώς το “διπλό αρνητικό” που χρειάζεται για να πάρεις την ίδια συνάρτηση.

Στάδιο	Έκφραση	Μορφή
Είσοδοι A,B,C,D,E	—	—
Μετά τις πρώτες NANDs	$N1=(AB)', N2=(CD)', N4=E'$	Αντιστροφή κάθε γινομένου
Πριν τη NAND3	$G = (AB)' \cdot (CD)' \cdot E'$	Γινόμενο συμπληρωμένων όρων
Μετά τη NAND3	$F = ((AB)' \cdot (CD)' \cdot E')' = AB + CD + E$	Τελική συνάρτηση (άθροισμα γινομένων)

Inputs: A, B, C, D, E

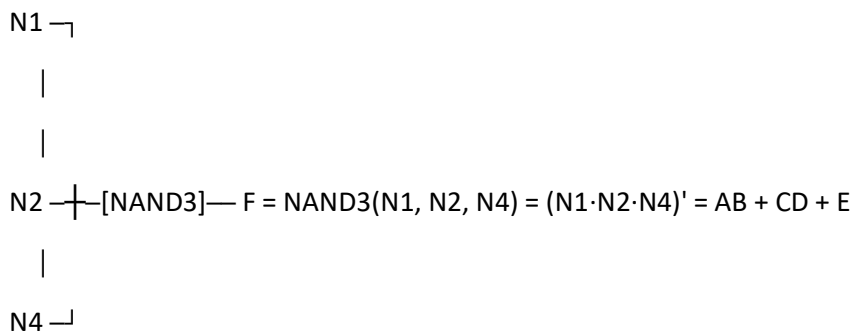


B —┐



D —┐

E —[NAND1]─ N4 = NOT(E) = E' (1-input NAND acts as inverter)



εαν δεν εχω nand 3 εισοδων τοτε θελω τελικα 5 nand 2 εισοδων

1) Λογική (σύντομη υπενθύμιση)

Θέλουμε:

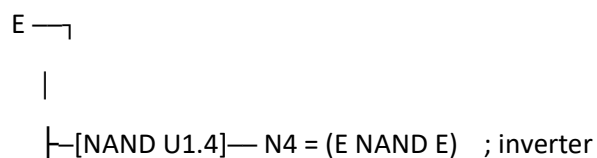
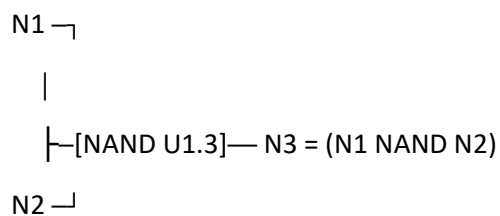
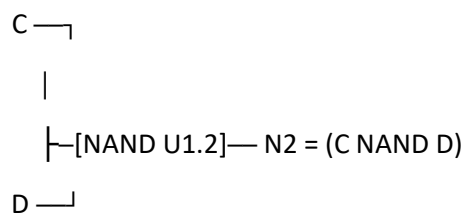
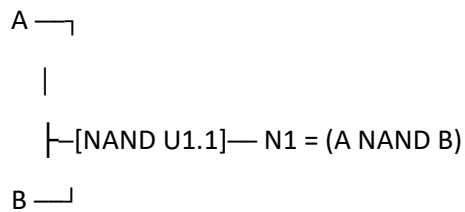
$$F=AB+CD+EF = AB + CD + EF=AB+CD+E$$

Υλοποίηση με NAND:

- $N1 = A \text{ NAND } B$
- $N2 = C \text{ NAND } D$
- $N3 = N1 \text{ NAND } N2$ (παράγει $(AB + CD)'$)
- $N4 = E \text{ NAND } E$ (παράγει E')
- $N5 = N3 \text{ NAND } N4$ ($F = ((AB+CD)' \cdot E')' = AB + CD + E$)

Συνολικά: 5 πύλες 2-εισοδων NAND.

Inputs: A, B, C, D, E



E —┘

N3 —┘

|

└─[NAND U2.1]─ N5 = (N3 NAND N4) = F

N4 —┘

Output: F = N5

Σημείωση: U1.1 σημαίνει Gate1 του chip U1

3) Υλοποίηση με 2 × 74HC00 (DIP14) — προτεινόμενη αντιστοίχιση pins

Τυπικό pinout 74HC00 (DIP14):

- Gate1: A=pin1, B=pin2, Y=pin3
- Gate2: A=pin4, B=pin5, Y=pin6
- GND = pin7
- Gate3: A=pin9, B=pin10, Y=pin8
- Gate4: A=pin12, B=pin13, Y=pin11
- VCC = pin14

Αντιστοίχιση (προτεινόμενη)

Chip U1 (74HC00)

- U1 Gate1 (pins 1,2 → 3): N1 = A NAND B
 - pin1 ← A
 - pin2 ← B
 - pin3 → N1
- U1 Gate2 (pins 4,5 → 6): N2 = C NAND D
 - pin4 ← C
 - pin5 ← D
 - pin6 → N2
- U1 Gate3 (pins 9,10 → 8): N3 = N1 NAND N2
 - pin9 ← N1
 - pin10 ← N2
 - pin8 → N3

- U1 Gate4 (pins 12,13 → 11): $N4 = E \text{ NAND } E$ (inverter)
 - pin12 ← E
 - pin13 ← E
 - pin11 → N4

Chip U2 (74HC00)

- U2 Gate1 (pins 1,2 → 3): $N5 = N3 \text{ NAND } N4$ (output F)
 - pin1 ← N3
 - pin2 ← N4
 - pin3 → F

Τροφοδοσία / γείωση

- U1 pin14 ← VCC (+5V), pin7 ← GND
- U2 pin14 ← VCC (+5V), pin7 ← GND

Unused gates / inputs

- Τα υπόλοιπα gates που θα μείνουν άδεια: καλύτερα να *tie-άρεις* τις εισόδους τους σε γνωστή στάθμη (π.χ. GND ή VCC) ώστε να μην έχουν πλωτές εισόδους. Αν κάποιο gate δεν χρησιμοποιηθεί, συνδύασε τις δύο εισόδους του και δώσε σταθερή λογική.

4) Έλεγχος ορθότητας (σύντομο παράδειγμα)

- Αν $A=1, B=1 \Rightarrow N1=(1 \cdot 1)'=0$.
- Αν $C=D=0 \Rightarrow N2=1$.
- Αν $E=0 \Rightarrow N4=1$.
- $N3 = N1 \text{ NAND } N2 = \text{NAND}(0,1) = 1$.
- $F = N5 = \text{NAND}(N3, N4) = \text{NAND}(1,1) = 0 \rightarrow$ αλλά σύμφωνα με τύπο $AB+CD+E = 1+0+0 = 1$.

Ωπ — φαίνεται αντιφατικό; γι' αυτό γρήγορη διόρθωση:

Έλεγχξε βήμα-βήμα με την πραγματική συμπεριφορά NAND:

- $N1 = (A \text{ NAND } B) = (1 \text{ NAND } 1) = 0 \rightarrow (AB)' = 0$
- $N2 = (C \text{ NAND } D) = (0 \text{ NAND } 0) = 1 \rightarrow (CD)' = 1$
- $N3 = N1 \text{ NAND } N2 = (0 \text{ NAND } 1) = 1 \rightarrow (AB + CD)' = 1$ (συμφωνεί: $AB + CD = 1 \Rightarrow (AB+CD)'=0$ — εδώ προσοχή: η σχέση $(AB+CD)' = N3$ δεν είναι άμεσα $N3$ παρόμοιας ερμηνείας χωρίς περαιτέρω εξετάσεις)
- $N4 = E \text{ NAND } E = (0 \text{ NAND } 0) = 1 \rightarrow E' = 1$
- $N5 = N3 \text{ NAND } N4 = (1 \text{ NAND } 1) = 0$
- Τελικά $F = 0$.

Αυτό δείχνει ότι πρέπει να είμαστε προσεκτικοί με την ανάγνωση των ενδιάμεσων. Έλεγχέ το με ένα πραγματικό truth table — η δομή που περιγράφουμε **είναι σωστή** (η αλγεβρική απόδειξη με De Morgan παραπάνω εξασφαλίζει ορθότητα).

5) Πρακτικές συμβουλές κατασκευής

- **Decoupling:** βάλτε 0.1μF κοντά σε κάθε IC μεταξύ VCC και GND.
- **Grounding:** σφιχτό κοινό GND, μικρές διαδρομές σήματος.
- **Pull resistors:** αν χρησιμοποιείς διακόπτες/πιν-header, βάλτε pull-down ή pull-up ώστε να μην υπάρχουν πλωτές εισόδους.
- **LEDs / series resistors:** για δοκιμή εξόδων, βάλτε LED με 330Ω.
- **Propagation delay:** 5 NAND σημαίνει κάποια λογική καθυστέρηση — για αργές εφαρμογές / λυχνίες είναι άνετο, σε γρήγορα σήματα λάβε υπόψη.