

7 Circuite logice combinaționale complexe din categoria MSI

7.1 Obiective

Se analizează și se verifică funcționarea unor componente integrate MSI care implementează funcții mai complexe precum: multiplexorul cu calea de date pe mai mulți biți, sumatorul, unitatea aritmetică-logică și decodificatorul BCD-7 segmente. Se proiectează un sumator-scăzător cu ajutorul unui sumator și se testează funcționalitatea acestuia. Se extinde numărul de biți prin cascada.

7.2 Considerații teoretice

Circuitele MSI complexe au o largă utilizare în aplicațiile bazate pe sisteme numerice. Operațiile întâlnite cel mai frecvent sunt cele aritmetice și logice, în consecință, o unitate aritmetică-logică este practic nelipsită. Creșterea numărului de biți de calcul este o abordare naturală în contextul în care aceste sisteme prelucrează date reprezentate pe multiplu de 8 biți (1 octet / byte). În general, afișarea rezultatelor se realizează în format zecimal. Decodificatoarele BCD-7 segmente facilitează afișarea valorilor binare în baza 10, cu ajutorul afișoarelor cu 7 segmente.

7.2.1 Multiplexoare și demultiplexoare cu mai multi biți pe calea de date

Multiplexoarele realizează selecția datelor de la un set de intrări către o singură ieșire, în funcție de valorile semnalelor de selecție. Demultiplexoarele realizează plasarea datelor de la o intrare unică la una din ieșirile indicate de semnalele de selecție. Multiplexoarele și demultiplexoarele se pot proiecta astfel încât intrările și ieșirile de date să fie reprezentate pe mai mulți biți. De exemplu, un MUX 4:1 cu calea de date pe 3 biți, are intrările și ieșirile de date pe 3 biți. El poate fi interpretat ca 3 unități MUX 4:1 cu calea de date pe 1 bit, ca în Figura 7. 1 – stânga, care funcționează concomitent și partajează semnalele de selecție.



Figura 7. 1 MUX 4:1 cu calea de date pe 3 biți – implementarea cu unități MUX 4:1 având calea de date pe 1 bit (stânga) și simbolul asociat (dreapta)

Similar, un DMUX 1:4 cu calea de date pe 3 biți poate fi implementat cu un set de 3 unități DMUX 1:4 cu calea de date pe 1 bit având selecțiile partajate (Figura 7. 2).

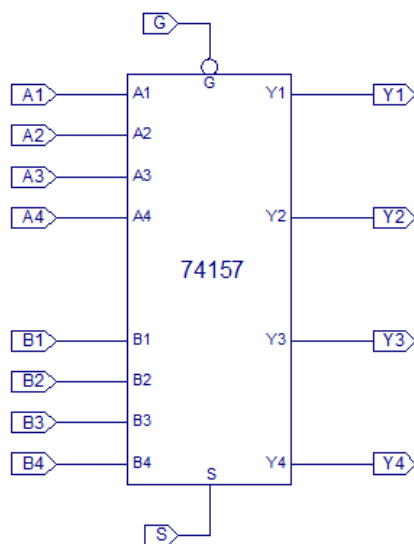


Figura 7. 2 DMUX 1:4 cu calea de date pe 3 biți – implementarea cu unități DMUX 1:4 având calea de date pe 1 bit (stânga) și simbolul asociat (dreapta)

Circuitul 74157 implementează un MUX 2:1 cu calea de date pe 4 biți. Simbolul circuitului este prezentat în Figura 7. 3. Intrările de date $A_{4:1}$, $B_{4:1}$ și ieșirea $Y_{4:1}$ sunt reprezentate pe 4 biți. Selecția S pe 1 bit stabilește intrarea care va fi înaintată pe ieșire, după regula:

$$Y_{4:1} = \begin{cases} A_{4:1}, & \text{dacă } S = 0 \\ B_{4:1}, & \text{dacă } S = 1 \end{cases} \quad (7.1)$$

Intrarea G (Strobe) activează circuitul dacă $G=0$, altfel ieșirile sunt menținute la 0.



Intrări				Ieșire
G (Strobe)	S (Select)	A _i	B _i	Y _i
1	X	X	X	0
0	0	0	X	0
0	0	1	X	1
0	1	X	0	0
0	1	X	1	1

Figura 7. 3 Multiplexorul 74157: simbolul (stânga) și tabelul de adevăr (dreapta)

7.2.2 Circuite care realizează operații aritmetice și logice

Operațiile aritmetice de bază sunt implementate cu sumatoare și scăzătoare. Circuitul 74283, prezentat în Figura 7. 4, realizează adunarea pe 4 biți a două numere binare $A_{4:1}$ și $B_{4:1}$. Rezultatul este calculat pe ieșirile $S_{4:1}$. Circuitul prezintă o intrare de transport (Carry In), la nivelul biților de rang 0, denumită C_0 , și o ieșire de transport (Carry Out), de rang 4, denumită C_4 , cu rol de semnalizare a depășirii. Liniile de transport se pot folosi la implementarea de sumatoare pe un număr extins de biți, pentru cascada mai multor circuite 74283. **Notă:** Activarea intrării C_0 are ca efect incrementarea rezultatului cu valoarea 1. Operația aritmetică efectuată de sumatorul 74283 este:

$$(C_4 S_4 S_3 S_2 S_1)_2 = (A_4 A_3 A_2 A_1)_2 + (B_4 B_3 B_2 B_1)_2 + (000 C_0)_2 \quad (7.2)$$

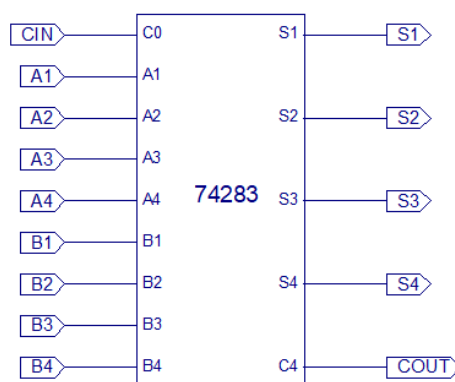


Figura 7. 4 Sumatorul 74283 pe 4 biți

7.2.2.1 Implementarea unui sumator-scăzător pe 4 biți

La reprezentarea în Complement față de 2, operația de scădere se poate realiza prin adunarea primului termen cu complementul față de 2 al celui de-al doilea termen, care se poate exprima la rândul său prin complementul față de 1: $A - B = A + \bar{B} = A + \bar{B} + 1$. La numere pe 4 biți expresia devine:

$$(A_4A_3A_2A_1)_2 - (B_4B_3B_2B_1)_2 = (A_4A_3A_2A_1)_2 + (\bar{B}_4\bar{B}_3\bar{B}_2\bar{B}_1)_2 + (0001)_2 \quad (7.3)$$

În algebra booleană, pentru o expresie φ avem relațiile: $\varphi \oplus 0 = \varphi$ și $\varphi \oplus 1 = \bar{\varphi}$. În consecință, operațiile de adunare și scădere pot fi exprimate unitar, cu ajutorul XOR:

$$\begin{cases} (A_4A_3A_2A_1)_2 + (B_4B_3B_2B_1)_2 = (A_4A_3A_2A_1)_2 + (B_4B_3B_2B_1 \oplus 0000)_2 + (0000)_2 \\ (A_4A_3A_2A_1)_2 - (B_4B_3B_2B_1)_2 = (A_4A_3A_2A_1)_2 + (B_4B_3B_2B_1 \oplus 1111)_2 + (0001)_2 \end{cases}$$

Cele 2 operații se pot contopi în expresia $(C_4S_4S_3S_2S_1)_2 = (A_4A_3A_2A_1)_2 + (B_4B_3B_2B_1 \oplus \text{Sel Sel Sel Sel})_2 + (0\ 0\ 0\ \text{Sel})_2$, unde **Sel=0 pentru adunare** și **Sel=1 pentru scădere**. Expresia poate fi implementată cu sumatorul 74283 și cu 4 porți XOR, în felul următor:

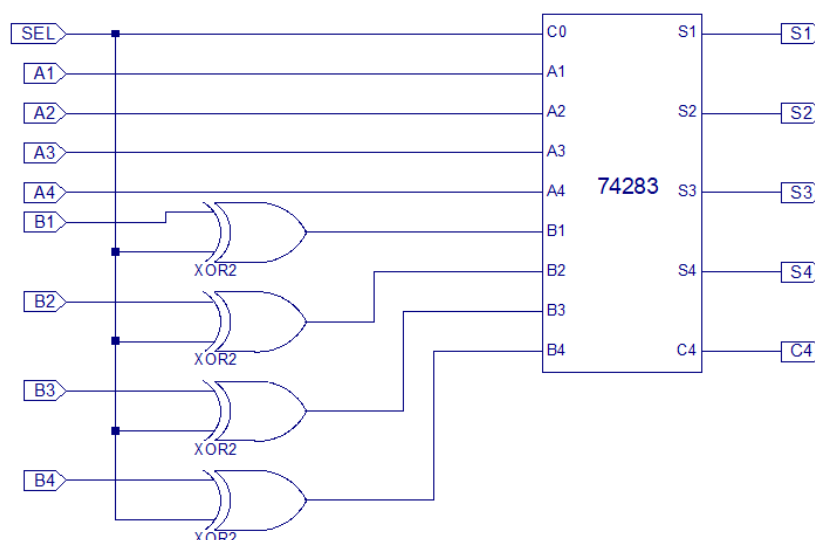


Figura 7. 5 Implementarea unui sumator-scăzător pe 4 biți, folosind circuitul 74283

7.2.2.2 Implementarea unui sumator-scăzător pe 8 biți

Pentru proiectarea pe 8 biți, expresia se poate extinde în felul următor:

$$(C_8 S_8 S_7 S_6 S_5 S_4 S_3 S_2 S_1)_2 = (A_8 A_7 A_6 A_5 A_4 A_3 A_2 A_1)_2 + (B_8 B_7 B_6 B_5 B_4 B_3 B_2 B_1 \oplus \text{Sel Sel Sel Sel Sel Sel Sel Sel})_2 + (0 0 0 0 0 0 0 \text{ Sel})_2 \quad (7.4)$$

Se vor utiliza 2 sumatoare 74283 cascade și 8 porți XOR. Unul dintre sumatoare va realiza calculul pe biții de rang 1÷4, cu semnalul Sel conectat la intrarea C₀. Cel de-al doilea sumator va realiza calculul pe biții de rang 5÷8, cu transportul C₄ de la sumatorul anterior conectat la C₀:

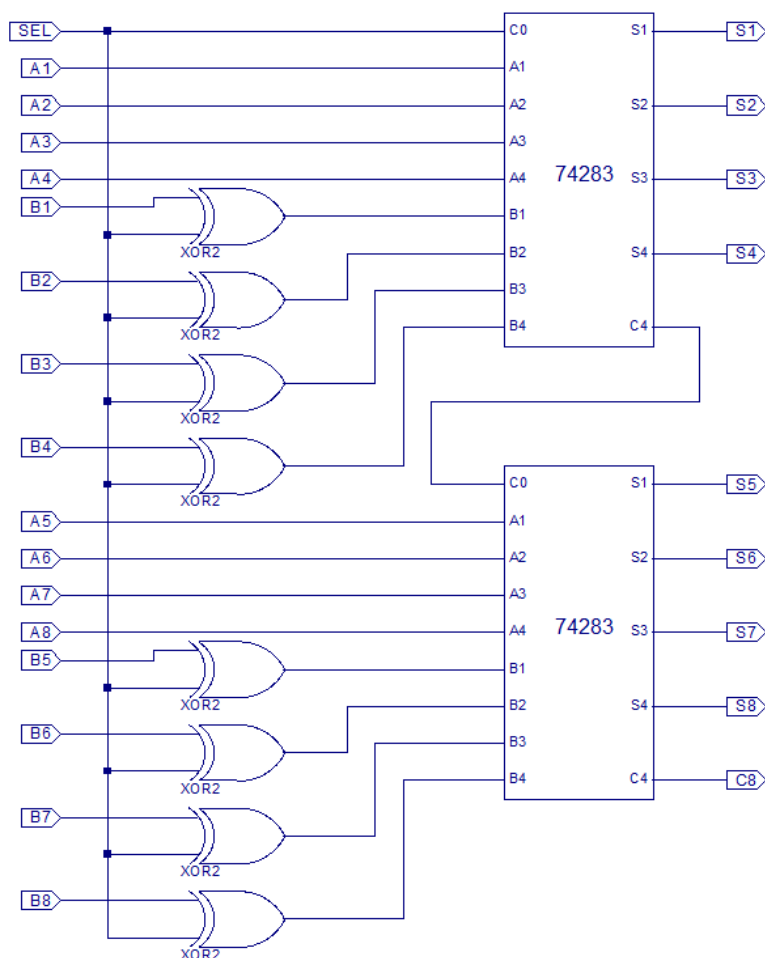


Figura 7. 6 Implementarea unui sumator-scăzător pe 8 biți prin cascaderă

7.2.2.3 Unități aritmetice-logice

Circuitul 74181 (Figura 7. 7) implementează operații aritmetice-logice pe 4 biți între operandii A_{3:0} și B_{3:0}, reprezentați în complement față de 2. Operația efectuată se stabilește cu ajutorul codului aplicat pe intrările S_{3:0}. Pentru fiecare cod se pot realiza două categorii de operații: o operație logică bit-cu-bit, dacă intrarea M=1 sau una aritmetică, dacă M=0. Rezultatul este calculat pe ieșirile F_{3:0}. Semnalele de transport C_n (C_{in}) și C_{n+4} (C_{out}) sunt active pe 0. Acestea ajută la extinderea numărului de biți prin cascaderă. Ieșirea EQ (sau A=B) testează egalitatea între A_{3:0} și B_{3:0} la efectuarea operației **A minus B minus 1**. Setul complet de operații poate fi urmărit în tabelul următor:

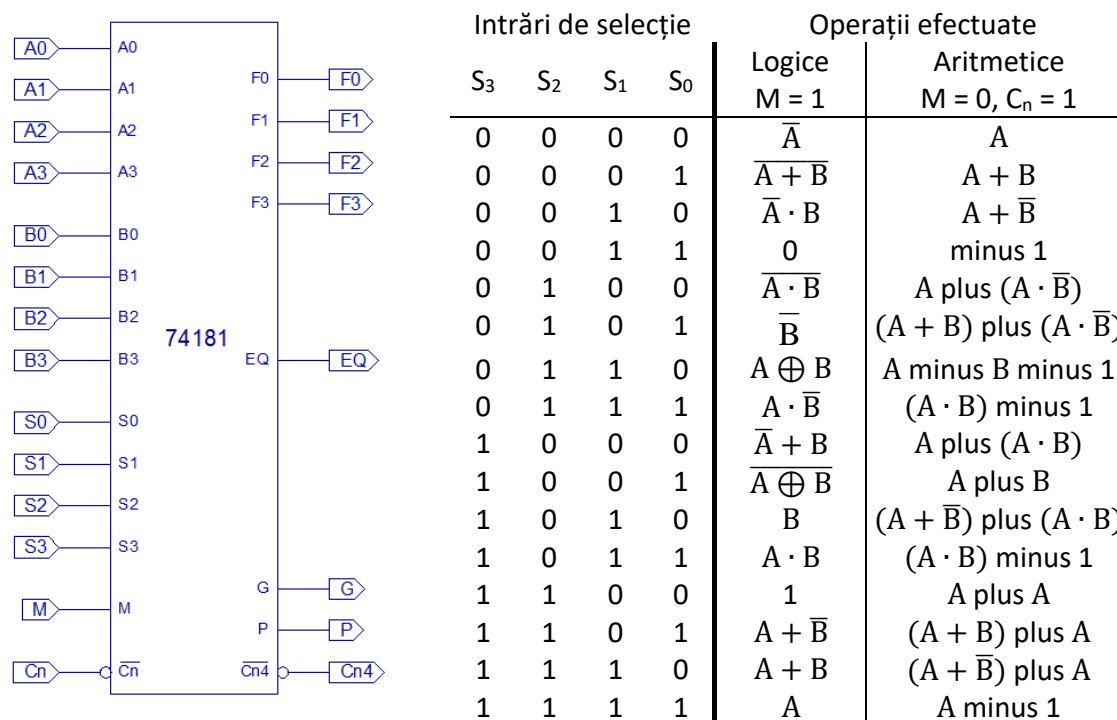


Figura 7. 7 Circuitul 74181: simbolul (stânga) și operațiile pe care le efectuează (dreapta)
(+ = SAU, · = ȘI, ⊕ = XOR, plus = adunare, minus = scădere)

7.2.3 Afișarea cifrelor zecimale

Placa de dezvoltare prezintă 8 afișoare cu 7 segmente pentru afișarea în baza 10 (Figura 7. 8) [1]. Fiecare afișor poate afișa o cifră zecimală, construită din cele 7 segmente, prin activarea sau inactivarea lor (Figura 7. 9). Segmentele sunt controlate de 7 semnale, denumite *catozi*, indexate de la A la G. Catozii sunt partajați de cele 8 afișoare. Fiecare afișor are un semnal de activare, denumit *anod*. Anozii sunt numerotați de la 0 la 7, fiind asociați afișoarelor de la dreapta la stânga. **Notă:** Datorită catozilor partajați, pe afișoarele active va apărea aceeași cifră, dar există tehnici prin care se pot afișa cifre diferite pe fiecare afișor, astfel încât să se poată vizualiza valori cu mai multe cifre zecimale. Catozii și anozii sunt activi pe 0. Implicit, catozii au valoarea 1 și anozii au valoarea 0, ceea ce înseamnă că afișoarele sunt active, dar nu afișează nimic pe segmente, fiind inactive.



Figura 7. 8 Afișoarele cu 7 segmente de pe placa de dezvoltare

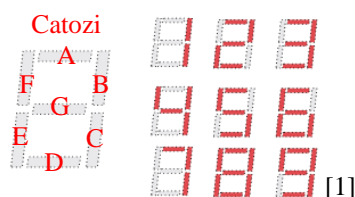


Figura 7. 9 Poziția segmentelor pe afișor și configurarea cifrelor zecimale

Circuitul 7447 este un decodificator BCD-7 segmente, care primește valori binare pe 4 biți și activează catodii care alcătuiesc cifra zecimală corespunzătoare, în vederea afișării acestora pe afișorul cu 7 segmente. Asocierea este prezentată în Figura 7. 10.



Figura 7. 10 Asocierea codurilor binare la simbolurile afișate

Ieșirile decodificatorului asociate catodilor A÷G funcționează în logica negativă (Figura 7. 11). Intrările \overline{LT} , \overline{RBI} , $\overline{BI}/\overline{RBO}$ sunt rezervate pentru configurații de test, motiv pentru care, la o funcționare normală, acestea se dezactivează prin conectare la VCC.

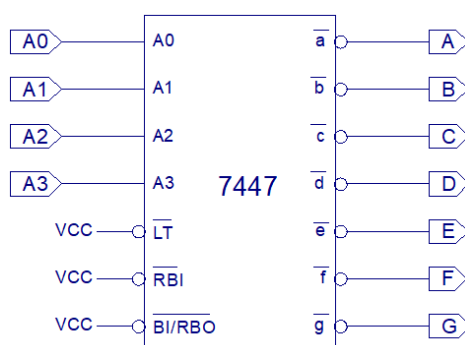


Figura 7. 11 Decodificatorul BCD-7 segmente 7447

În Logisim, afișorul pe 7 segmente se află în librăria *Input/Output* și funcționează în logica pozitivă, deci ieșirile decodificatorului 7447 trebuie inversate.

În Project Navigator, în fișierul .ucf, catodii și anozii se află într-o secțiune separată:

7 segment display

NET "A" LOC=T10 | IOSTANDARD=LVCMOS33; # cat a

NET "B" LOC=R10 | IOSTANDARD=LVCMOS33; # cat b

...

7.3 Activități practice

1. Implementați și testați pe placă multiplexorul 74157 cu calea de date pe 4 biți.
2. Implementați și testați pe placă sumatorul 74283 pe 4 biți.
3. Implementați și testați pe placă un sumator-scăzător implementat cu circuitul 74283.
4. Implementați și testați pe placă decodificatorul BCD-7 segmente 7447.
5. Implementați și testați în Logisim un sumator pe 8 biți cu sumatoare 74283.
6. Implementați și testați în Logisim un sumator-scăzător pe 8 biți cu sumatoare 74283.
7. Implementați și testați în Logisim unitatea aritmetică-logică 74181 pe 4 biți.

7.4 Bibliografie

[1] Digilent, Nexys A7 Reference Manual, [Online]. Available:

<https://digilent.com/reference/programmable-logic/nexys-a7/reference-manual>