

2 EDITAREA SCHEMATICĂ ȘI SIMULAREA FUNCȚIONĂRII CIRCUITELOR CU SOFTWARE SPECIALIZAT (I)

2.1 Scopul lucrării


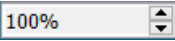
Se prezintă utilitarul Logisim care pune la dispoziție facilități de proiectare și simulare a circuitelor numerice digitale. Sunt prezentate elementele componente ale utilitarului și modul de lucru cu controalele active din componența simulatorului. Sunt redate câteva exemple de circuite, atât din punct de vedere al modului de implementare cât și în ceea ce privește procesul de simulare a funcționalității acestora. Se prezintă editorul de simboluri și posibilități de organizare a circuitelor în cadrul librăriei proiectului.

2.2 Considerații teoretice

Utilitarul Logisim - versiunea 2.7.1 (necesită instalarea Java Runtime Environment - versiunea 5 în sus) oferă suport pentru descrierea circuitelor numerice digitale și simularea funcționalității acestora prin aplicarea de stimuli pe terminalele de intrare și observarea rezultatelor care apar pe terminalele de ieșire. Descrierea unui circuit se face sub forma unei scheme de simboluri grafice interconectate, care poartă denumirea de *schemă logică*. Logisim pune la dispoziție un set de circuite logice de bază și oferă posibilitatea extinderii acestora cu alte circuite printr-o implementare în stil ierarhic și grupare în librării de circuite reutilizabile. Implementarea unui circuit presupune două faze de dezvoltare:


1. Descrierea schemei logice în editorul schematic prin care se definește structura circuitului. În modul de descriere a schemei logice se poate activa sau dezactiva modulul de simulare a funcționalității circuitului.
2. Definirea simbolului reprezentativ circuitului în vederea reutilizării acestuia în componența altor circuite. În acest fel se facilitează dezvoltarea ierarhică la niveluri de abstractizare de complexitate progresivă.

2.2.1 Editorul schematic

Editorul schematic este fereastra principală care devine disponibilă din primul moment al lansării în execuție a utilitarului Logisim. Editorul prezintă o zonă de lucru în care se poate realiza schema logică a circuitului (Figura 2. 1). În funcție de preferințe, aspectul punctat al fundalului se poate schimba prin apăsarea butonului  situat în colțul stânga-jos, la dreapta de zonei de control a dimensiunii de vizualizare a schemei . Prin schimbarea dimensiunii schemei se pot mări sau micșora (zoom in/zoom out) elementele de pe aceasta.

2.2.1.1 Uneltele de lucru și organizarea pe librării (Toolbox)

În partea laterală a zonei de lucru se află uneltele de lucru pentru realizarea unei scheme logice (zona Toolbox). Acesta conține lista librăriilor disponibile începând din partea superioară cu librăria proiectului denumită implicit *Untitled* (acesta este de altfel

și numele proiectului curent). Fiind la început, în cadrul acesteia se află doar circuitul curent – cu denumirea implicită *main* – a cărui schemă logică este în dezvoltare. Apăsând simbolul  din dreptul unei librării se pot vizualiza uneltele și circuitele din componența sa sub formă de simboluri grafice asociate la numele lor. Astfel, în cadrul librăriei *Gates* se pot accesa porțile logice fundamentale, iar în celelalte librării se pot descoperi circuite logice de complexitate mai ridicată sau diverse unelte necesare în realizarea schemei logice.

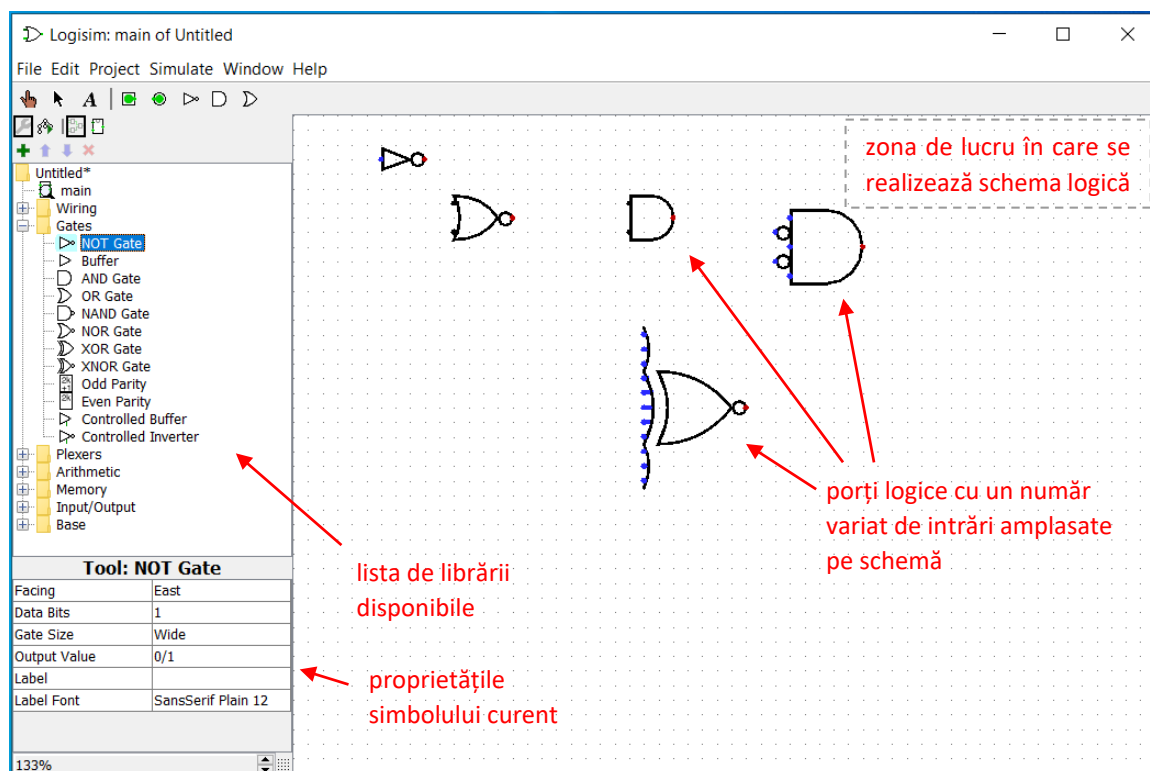


Figura 2. 1 Editorul schematic al utilitarului Logisim

2.2.1.2 Simbolurile, amplasarea pe schemă și proprietățile acestora




Cu ajutorul simbolurilor puse la dispoziție de librăriile existente se poate realiza schema logică a circuitului curent. Acestea se pot introduce pe schemă prin selectarea lor urmată de un click (cu mouse-ul) în poziția dorită a zonei de lucru. Ulterior se pot reamplasa într-o altă poziție prin selectare și deplasare cu mouse-ul (drag-and-drop).

Pe schemă se pot introduce oricâte simboluri de același tip sau de tipuri diferite. Un simbol de pe schemă se poate multiplica cu opțiunile de copiere (**Copy**) și lipire (**Paste**) disponibile în secțiunea **Edit** a meniului din partea superioară a ferestrei principale sau cu combinația de taste **Ctrl+C**, urmată de **Ctrl+V** sau simplu **Ctrl+Insert**. Ori de câte ori un simbol este selectat va apărea în partea inferioară a listei de librării numele simbolului și o listă de proprietăți specifice acestuia. De exemplu prin selectarea inversorului NOT (Figura 2. 1) setul de proprietăți cuprinde:


- Facing – direcția de amplasare pe schemă (rotirea simbolului);
- Data Bits – numărul de biți pentru care se aplică funcția logică NOT;

- Gate Size – dimensiunea simbolului pe schemă;
- Output Value – comportamentul la ieșire: poate avea “0 logic” sau “1 logic”;
- Label – eticheta de identificare pe schemă, care este opțională;
- Label Font – dimensiunea textului etichetei.

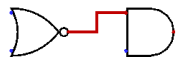
Proprietățile diferă de la simbol la simbol, dar o parte dintre acestea sunt comune la majoritatea. Valorile proprietăților se pot modifica, ceea ce poate avea efect asupra funcționalității circuitului asociat simbolului și/sau asupra aspectului acestuia în zona de lucru, în funcție de caz. De exemplu: o poartă ȘI-NU (NAND) cu 2 intrări (Number Of

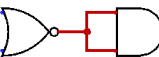
Inputs = 2) are simbolul ; dacă are 5 intrări simbolul devine ; dacă (de exemplu) intrările 2 și 4 sunt inversate (Negate 2 = Yes; Negate 4 = Yes) atunci apare un cerculeț în dreptul acestora .

2.2.1.3 Realizarea conexiunilor pe schema logică

Se observă faptul că intrările și ieșirile apar de regulă sub forma unor puncte mai proeminente amplasate la extremitățile simbolului având culori diferite. Pentru o localizare fiabilă a acestora amplasarea mouse-ului în dreptul lor duce la apariția unui cerculeț verde în jur: .

Conectarea intrărilor și ieșirilor se realizează prin deplasarea mouse-ului pe schemă de la un pin la celălalt în timp ce se menține apăsat butonul din stânga:



Dacă o intrare/ieșire are mai mulți biți atunci în mod automat și conexiunea creată are același număr de biți. Pe schemă se pot conecta numai intrări/ieșiri cu același număr de biți, în caz contrar apare o atenționare de eroare. În terminologia de specialitate o conexiune care are mai mulți biți se numește *magistrală* sau *bus*, dar în Logisim poartă denumirea de fir (*wire*) indiferent de numărul de biți. Crearea unei ramificații într-un punct al unui fir se poate realiza prin trasarea unui nou fir începând cu acest punct până la o intrare/ieșire neconectată: . Ramificația se distinge pe fir printr-o bulină de culoarea acestuia. Astfel tipurile de conexiuni posibile pot fi:

- conexiune directă de la o ieșire la o intrare;
- conexiune de la o ieșire la un fir existent sau de la un fir existent la o intrare; de menționat faptul că două fire separate devin unul singur prin conectare.

Notă: indiferent de numărul de ramificații, la fiecare moment de timp valoarea fiecărui bit dintr-o conexiune trebuie să fie aceeași pe toate segmentele firului. Ca urmare nu este corectă conectarea printr-un fir a doi sau mai mulți pini de ieșire fiindcă pot avea concomitent valori diferite în timpul funcționării circuitului.

Orice element de pe schemă (simbol sau segment de fir) se poate șterge prin click-dreapta pe acesta și alegerea opțiunii de ștergere (Delete) sau apăsând tasta *Del* după selectare.

În librăria *Wiring* se află elementul de circuit **Splitter** cu ajutorul căruia se pot grupa mai multe fire de dimensiune (număr biți) redusă într-un singur fir sau invers, un fir de dimensiune ridicată se poate împărți mai multe ramuri. Proprietatea *Bit Width In* definește numărul total de biți, iar *Fan Out* definește numărul de ramuri. Așadar dimensiunea unei ramuri este dată de relația $\text{Fan Out} / \text{Bit Width In}$. De exemplu pentru a comasa 4 fire de 1 bit într-un singur fir de 4 biți sau pentru a despărți un fir de 4 biți în 4 fire de 1 bit se va folosi un Splitter cu *Bit Width In* = 4 și *Fan Out* = 4. Un circuit care conține 2 astfel de Splitter-e este prezentat în Figura 2. 2. Pentru aliniere, cele două Splitter-e au fost rotite corespunzător cu ajutorul proprietății *Facing*.

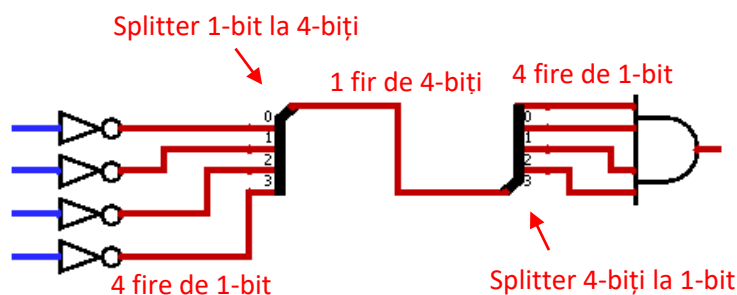


Figura 2. 2 Utilizarea Splitter-elor pentru modificarea dimensiunii firelor

Conexiunea între inversoare și poarta AND se putea realiza și direct, fără a fi nevoie de Splitter-e, folosind 4 fire de 1-bit, precum în Figura 2. 3. De reținut totuși faptul că folosirea Splitter-elor și comasarea mai multor fire în unul singur poate simplifica circuitul din punct de vedere vizual prin reducerea numărului de fire trasate.



Figura 2. 3 Conexiunea bit cu bit a elementelor de circuit fără Splitter-e

Atunci când o schemă logică devine încărcată datorită complexității sale eliminarea trasării firelor de legătură de la un capăt la altul poate ușura vizibilitatea circuitului și analiza acestuia în timpul simulării. În acest scop elementul de circuit **Tunnel** din librăria *Wiring* realizează **conexiunea virtuală** a firelor cu care au contact, dacă au aceeași etichetă în circuit (același nume al proprietății *Label*). Deasemenea este necesar ca firele conectate și elementele Tunnel să transporte un număr identic de biți (Figura 2. 4).

2.2.1.4 Elemente de circuit cu valori constante

Printre alte elemente de circuit puse la dispoziție de librăria *Wiring* se pot remarca **Power**, **Ground** și **Constant**. **Power** reprezintă sursa de curent care are valoarea "1

logic" (VCC), iar Ground reprezintă masa care are valoarea "0 logic" (GND). Ambele pot fi folosite cu rol de valori constante în circuit. În afară de Power și Ground se pot introduce și alte valori constante reprezentate pe unul sau mai mulți biți folosind elementul Constant. Valoarea generată de acesta în circuit se configurează schimbându-i atributul Value. Valorile se pot introduce în baza 10 (fără prefix) sau în baza 16 (folosind prefixul 0x). Utilizarea acestor elemente este exemplificată în Figura 2. 4. Datorită faptului că simularea este activată implicit în Logisim, valorile constante se propagă prin schemă producând efecte vizibile la ieșirile circuitelor și pe firele de legătură. Din acest motiv firele care transportă valoarea "1 logic" au culoarea verde-deschis, iar cele care transportă valoarea "0 logic" sunt colorate cu verde-închis.

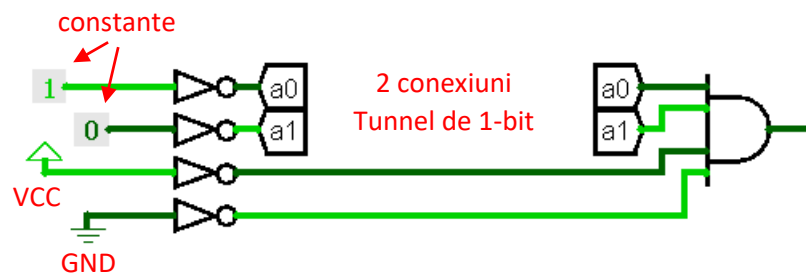


Figura 2. 4 Conexiuni între porți logice realizate cu 2 elemente Tunnel și cu 2 fire explicite de 1-bit. Intrările sunt conectate la elemente de circuit cu valori constante

2.2.1.5 Unealta de text

Unealta **A Text Tool** este disponibilă în cadrul librăriei *Base* sau în bara de unelte (Toolbar) situată imediat sub meniul . Cu ajutorul ei se pot insera pe schema logică elemente de text fără o funcționalitate propriu-zisă. Textul poate conține explicații sau etichete relevante pentru cei care analizează schema, având în vedere faptul că aceasta se poate salva ca imagine cu comanda **Export Image...** din secțiunea **File** a meniului sau se poate tipări la imprimantă cu comanda **Print...**

2.2.1.6 Terminale de intrare și de ieșire

Orice schemă logică a unui circuit necesită pe lângă valori constante, atât terminale de intrare prin care se transmit valori către circuit cât și terminale de ieșire prin care se pot prelua rezultatele produse de acesta. Numărul de terminale de intrare și de ieșire depinde de structura circuitului. În timpul simulării terminalele evidențiază valorile lor logice. În Logisim terminalul de intrare are simbolul și este accesibil în librăria *Wiring*. Atributul său numit **Output?** definește tipul terminalului: de intrare sau de ieșire. În cazul în care terminalul este de ieșire (**Output?** = **True**) simbolul adoptă o formă circulară (Figura 2. 5). Un terminal poate avea un număr variabil de biți definit cu ajutorul proprietății **Data Bits**. Schimbarea numărului de biți va schimba în consecință aspectul terminalului permițând vizualizarea valorii fiecărui bit.

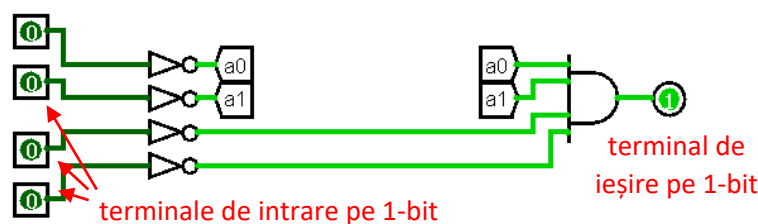


Figura 2. 5 Circuit cu terminale de intrare și ieșire pe care se pot identifica valorile logice în timpul simulării

2.2.1.7 Simularea combinațională în Logisim

După cum s-a precizat în capitolele anterioare simularea este activată implicit în Logisim, motiv pentru care din momentul plasării terminalelor de intrare, valorile acestora se propagă prin circuit producând efectele corespunzătoare. Tot implicit este activat și modul de editare a schemei logice, ceea ce înseamnă că se pot aduce modificări schemei, dar nu se pot modifica valorile asociate terminalelor de intrare. Pentru a trece în modul de modificare a terminalelor de intrare trebuie să se acceseze unealta **Poke Tool** situată atât în cadrul librăriei *Base*, cât și în bara de unelte . După accesarea unelei **Poke Tool** orice click (de mouse) realizat asupra unui terminal de intrare va schimba valoarea acestuia din "0" în "1 logic" sau invers. Revenirea în modul de editare a schemei logice se face accesând una din uneltele **Select Tool** sau **Edit Tool** situate tot în librăria *Base* sau în bara de unelte sau .

Dacă modulul de simulare detectează erori sau conexiuni incomplete în circuit atunci terminalele de ieșire pot indica valori necunoscute sau eronate . Toate comenzile legate de sesiunea de simulare se află la secțiunea **Simulate** a meniului. De exemplu oprirea simulării se poate realiza cu comanda **Simulation Enabled** sau cu combinația de taste **Ctrl+E**. Tot cu această comandă se realizează reactivarea simulării. Resetarea simulării de la momentul zero se realizează cu comanda **Reset Simulation** sau cu combinația de taste **Ctrl+R**. Aceasta este necesară după oprirea simulării, dacă se dorește resetarea ultimelor semnale valorice înregistrate în circuit și forțarea tuturor terminalelor la "0 logic".










2.2.1.8 Bara de unelte (Toolbar) a editorului schematic

Bara de unelte este situată dedesubtul meniului și conține un subset de elemente de circuit și unelte care pot fi accesate cu ușurință de utilizator. După instalarea utilitarului Logisim bara de unelte a editorului schematic are următoarea componență:



. Este posibilă modificarea componenței acesteia accesând opțiunea **Options** din secțiunea **Project** a meniului. În fereastra de opțiuni care apare pe ecran se va accesa tab-ul **Toolbar** care permite adăugarea și ștergerea de elemente din bară în funcție de preferințe.

2.2.2 Editorul de simboluri

Circuitul descris de schema logică se poate încapsula într-un simbol reprezentativ de forma unui dreptunghi pe laturile căruia vor fi situate terminalele de intrare și de ieșire sub formă de pini. Acest simbol se salvează în librăria proiectului și poate fi utilizat ulterior în cadrul altor scheme logice, păstrând în același timp funcționalitatea circuitului pe care îl încapsulează. Interacțiunea cu circuitul se va face strict prin terminalele simbolului. Comutarea de la modulul de editare a schemei logice la modulul de editare a simbolului asociat și invers se realizează cu comenzile **Edit Circuit Appearance**, respectiv **Edit Circuit Layout** din secțiunea **Project** a meniului. Pentru o comutare rapidă, cele două comenzi apar și dedesubtul barei de unelte având simbolurile:  . Pentru circuitul din Figura 2. 5 se generează automat simbolul prezentat în Figura 2. 6. Acesta poate suferi oricâte modificări ulterioare sunt necesare. Se observă faptul că terminalele de intrare sunt marcate cu un dreptunghi albastru, iar cele de ieșire cu un cerculeț. Pentru identificarea unui terminal, selectarea acestuia pe simbol va evidenția poziția sa în schema logică. Opțional, denumirile terminalelor sau chiar a circuitului se pot amplasa pe simbol folosind unealta de text din Toolbar-ul editorului de simboluri:       .

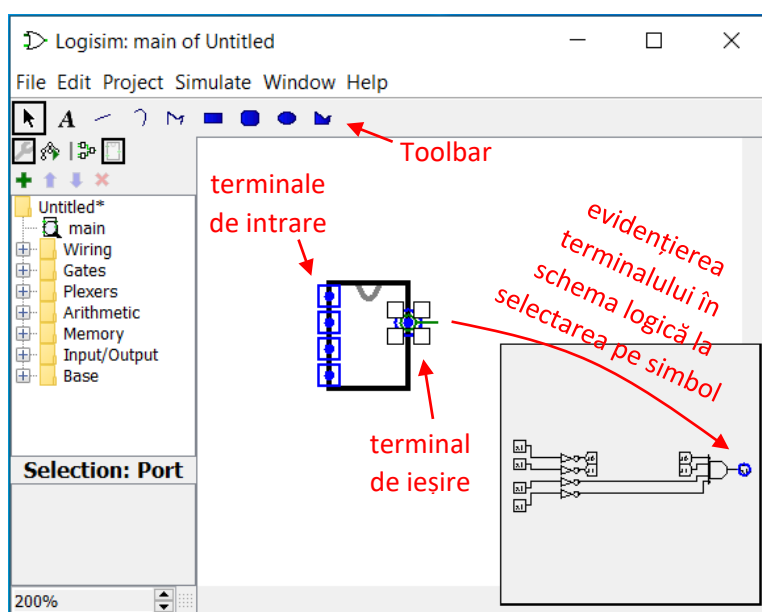






Figura 2. 6 Modulul de editare a simbolului asociat schemei logice curente

2.2.3 Organizarea circuitelor în cadrul librăriei proiectului

În cadrul librăriei proiectului se pot crea unul sau mai multe circuite. Adăugarea unui circuit se realizează cu comanda Add circuit  situată deasupra listei de librărie. Programul va solicita numele noului circuit și se va crea o schemă logică fără elemente. Alături de această comandă se află comenzile de deplasare   a circuitului curent într-o altă poziție în cadrul librăriei și comanda de ștergere  a circuitului. Cu aceste comenzi se pot organiza schemele logice în cadrul librăriei.

Se poate comuta între circuitele librăriei proiectului apăsând dublu-click pe numele circuitului în Toolbox. Va apărea schema logică a circuitului în zona de lucru și se

vor putea aduce modificări. În meniu la secțiunea **File** sunt comenzi de salvare a proiectului - **Save**, de încărcare a unui alt proiect - **Open** sau de creare a unui nou - **New**. Un proiect se va salva cu tot cu librăria de circuite a acestuia.

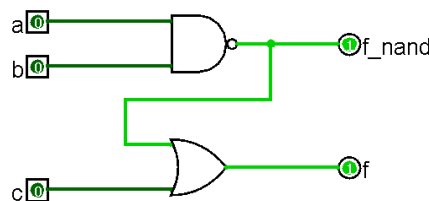
Obs: Salvați proiectul din când în când și la final pentru a nu pierde circuitele realizate.

Prin selectarea unei scheme din librăria curentă se vor putea vizualiza și modifica proprietățile acestora printre care și numele indicat de atributul **Circuit Name**. Așadar modificarea acestei proprietăți va determina modificarea numelui circuitului în cadrul librăriei.

2.3 Desfășurarea lucrării

1. Creați un proiect nou în Logisim și realizați un circuit care să conțină porțile logice fundamentale NOT, AND, XOR, OR din librăria *Gates*. Configurați porțile să fie pe 1-bit, cu 2 intrări. Introduceți 2 terminale de intrare folosite la comun pentru toate porțile și 4 terminale de ieșire, câte unul pentru fiecare poartă. Testați funcționarea porților logice cu ajutorul simulatorului.

2. Adăugați un nou circuit în cadrul proiectului în care să realizați schema logică din figura următoare. Realizați tabelul de adevăr al circuitului. Testați circuitul aplicând toate combinațiile posibile de valori pe intrările a, b și c și confrunțați rezultatele cu valorile din tabel. Vizualizați simbolul generat pentru circuit și adăugați nume corespunzătoare în dreptul terminalelor de intrare și ieșire folosind unealta de text din Toolbar.



3. Adăugați câte un circuit separat în cadrul proiectului pentru următoarele funcții logice și testați-le în simulator confruntând rezultatele cu cele din tabelul de adevăr pentru toate combinațiile valorilor de intrare:

- a) $f_1 = a + \bar{b} + c$
- b) $f_2 = (a + b) \cdot (a + c)$
- c) $f_3 = a + \bar{b} \cdot \bar{c}$
- d) $f_4 = b \cdot (\overline{a + c} + b \cdot \bar{c})$
- e) $f_5 = \overline{a \cdot \bar{b} \cdot c}$
- f) $f_6 = \bar{a} + \bar{b} + \overline{a + c}$
- g) $f_7 = \overline{a \cdot b} + b \cdot \bar{c}$