

UTILIZAREA EDITORULUI PENTRU AUTOMATE DE STARE STATECAD

1. Scopul lucrării

Lucrarea prezintă principalele aspecte legate de utilizarea editorului pentru automate de stare StateCAD, care permite specificarea sub formă grafică a automatelor de stare în cadrul sistemului CAD Xilinx WebPACK. Lucrarea descrie de asemenea utilizarea programului StateBench, care permite verificarea funcționării automatelor descrise cu ajutorul editorului StateCAD. Se prezintă apoi un exemplu de proiectare utilizând programele StateCAD și StateBench.

2. Considerații teoretice

2.1. Prezentare generală

StateCAD este un modul integrat în sistemul de proiectare Xilinx WebPACK care permite crearea și editarea sub formă grafică a automatelor cu stări finite. Acest modul permite definirea stărilor, a semnalelor de ieșire care trebuie activate în fiecare stare (acțiunile stărilor) și a tranzițiilor dintre stări. Acțiunile stărilor și condițiile tranzițiilor pot fi specificate prin ecuații cu o sintaxă simplă, fără a fi necesară cunoașterea limbajelor de descriere hardware. Modulul StateCAD permite și adăugarea la automatele de stare create a unor circuite logice uzuale, ca porți logice, multiplexoare, sumatoare, comparatoare, bistabile, registre de deplasare și numărătoare.

StateCAD permite validarea diagramelor de stare înainte de simularea funcționării acestora. Proiectantul poate descoperi condiții nedeterminate, erori de sintaxă și porțiuni specificate incomplet din cadrul diagramelor de stare. După validarea unei diagrame, aceasta poate fi compilată în codul echivalent într-un limbaj de descriere hardware (VHDL, Verilog sau ABEL). Codul obținut poate fi utilizat ulterior pentru simularea automatului de stare, pentru sinteza și implementarea automatului, sau pentru crearea unui simbol (macro) din automatul de stare, care poate fi plasat apoi într-o schemă.

După crearea și validarea unui automat de stare, se poate verifica funcționarea acestuia utilizând programul StateBench. Acest program permite crearea unui banc de test pentru automatul care trebuie testat, prin specificarea semnalelor de intrare ale automatului și a semnalelor de ieșire care trebuie generate în fiecare stare. StateBench utilizează bancul de test creat pentru simularea funcționării automatului, indicând diferențele față de rezultatele așteptate. După simulare, se poate genera un banc de test care conține constrângeri de timp. Acest banc de test se poate utiliza pentru verificarea constrângerilor de timp după sinteza automatului.

2.2. Definirea variabilelor și constantelor

Pentru definirea automatelor de stare, trebuie să se asigneze un nume fiecărei stări și trebuie să se specifice valorile semnalelor de ieșire în fiecare stare. Pentru aceasta se utilizează variabile și constante. Operatorii sunt de asemenea utili pentru definirea unor ecuații mai complexe care definesc valorile semnalelor de ieșire. Operatorii recunoscuți sunt prezentați în secțiunea 2.5.3, în care este descrisă sintaxa ecuațiilor.

Numele variabilelor recunoscute de editorul StateCAD trebuie să respecte următoarele cerințe:

- Numele variabilelor trebuie să înceapă cu un caracter alfabetic (A..Z, a..z).
- Următoarele caractere pot fi caractere alfanumerice (A..Z, a..z, 0..9) și caracterul de subliniere (). Nu sunt admise două caractere de subliniere consecutive.
- Ultimul caracter nu poate fi caracterul de subliniere.
- Cuvintele rezervate nu pot fi utilizate ca nume de variabile.
- Nu se diferențiază între ele literele mici și cele mari.
- Numele semnalelor de tip bit pot avea o lungime de până la 16 caractere, iar numele vectorilor pot avea o lungime de până la 14 caractere.

Constantele pot fi utilizate pentru inițializarea semnalelor sau pentru compararea valorii vectorilor. Aceste constante pot fi definite în baza 2, 8, 10 sau 16. Baza implicită este cea zecimală. Constantele sunt definite prin valori numerice precedate opțional de un specificator al bazei, care începe cu caracterul `^`. Constantele binare de un singur bit pot fi specificate și prin includerea lor între ghilimele simple, iar constantele binare de mai mulți biți pot fi specificate și prin includerea lor între ghilimele duble. Tabelul 9.1 indică specificatorii bazelor de numerație și exemple de constante definite în diferite baze.

Tabelul 9.1. Definirea constantelor recunoscute de editorul StateCAD.

Baza	Specificator al bazei	Exemple
Binară (2)	<code>^b</code> sau <code>^B</code>	<code>^b11110000</code> , <code>"11110000"</code> , <code>'1'</code> , <code>'0'</code>
Octală (8)	<code>^o</code> sau <code>^O</code>	<code>^o360</code>
Zecimală (10)	<code>^d</code> sau <code>^D</code>	<code>^d240</code> , <code>240</code>
Hexazecimală (16)	<code>^h</code> sau <code>^H</code>	<code>^hF0</code> , <code>^hf0</code>

Există două constante predefinite, **TRUE** și **FALSE**, care pot fi utilizate pentru a indica faptul că o condiție este întotdeauna adevărată, respectiv falsă. Aceste constante pot fi utilizate în ecuații similare altor constante sau variabile.

2.3. Crearea unui fișier pentru o diagramă de stare

Un fișier sursă care va conține o nouă diagramă de stare poate fi creat din fereastra *Project Navigator* a sistemului Xilinx WebPACK. Pentru aceasta, se execută următoarele operații:

1. Se selectează comanda *Project* → *New Source*.
2. În fereastra de dialog *New*, se selectează opțiunea *State Diagram*.
3. În câmpul *File Name* se introduce numele fișierului în care se va păstra diagrama de stare.
4. Se selectează butonul *Next*.
5. În fereastra de dialog *New Source Information*, se selectează butonul *Finish*.

Editorul StateCAD va fi lansat în execuție într-o nouă fereastră (Figura 9.1).

Observații

- Numele fișierului introdus în câmpul *File Name* trebuie să înceapă cu un caracter alfabetic.
- Numele fișierului poate conține până la opt caractere alfanumerice. Extensia fișierului va fi adăugată în mod automat de către editor; extensia implicită este `.dia`.

- Opțiunea *Add to Project* din fereastra de dialog *New* nu are efect în cazul creării diagramelor de stare. Aceste diagrame nu pot fi adăugate în mod automat la proiectele create cu sistemul WebPACK, fiind necesară adăugarea explicită a acestora cu comanda *Project* → *Add Source*.

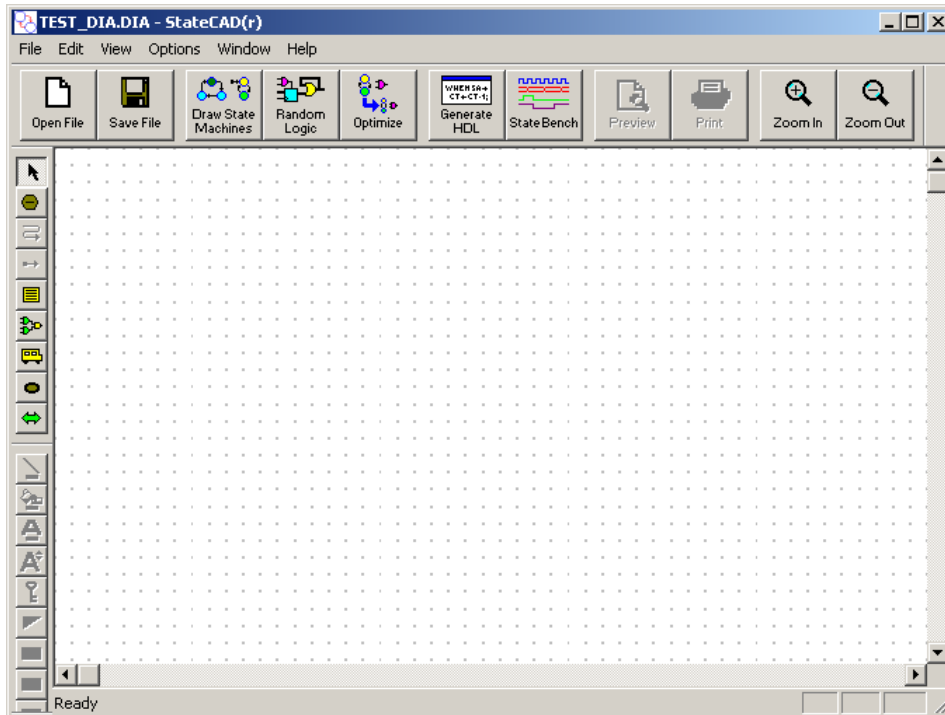


Figura 9.1. Fereastra editorului pentru automate de stare StateCAD.


2.4. Crearea și editarea automatelor de stare

2.4.1. Crearea unui automat de stare

Un automat de stare poate fi creat prin adăugarea stărilor automatului, specificarea acțiunilor pentru fiecare stare (o acțiune reprezentând activarea unui semnal de ieșire) și adăugarea tranzițiilor între stări. Stările indică situația curentă în care se află automatul de stare. O stare este reprezentată de către editorul StateCAD printr-un cerc sau o elipsă, în interiorul căreia se indică numele stării și, sub acest nume, semnalele de ieșire care sunt active în starea respectivă. Tranzițiile între stări sunt reprezentate prin săgeți trasate între aceste stări.

Crearea unui nou automat de stare este simplificată prin utilizarea utilitarului *State Machine Wizard*, care creează un schelet al automatului de stare. Acest automat conține numărul de stări specificat de proiectant, stările având nume asignate în mod implicit. De asemenea, automatul conține tranzițiile între stări și implementează logica de resetare specificată de proiectant. Ulterior, se pot edita stările automatului, se pot adăuga noi stări sau tranziții și se pot specifica acțiunile care trebuie executate în fiecare stare.

Pentru crearea unui automat de stare cu ajutorul utilitarului *State Machine Wizard*, se procedează astfel:

1. Se selectează butonul *Draw State Machines* . Se va deschide fereastra *State Machine Wizard* (Figura 9.2).
2. În câmpul *Shape of state machine* din fereastra afișată se selectează forma sub care se va desena diagrama de stare: pe o coloană (*Column*), pe două, trei sau patru coloane (*Multi-Column*), pe linie (*Row*), sau ca o figură geometrică cu forma corespunzătoare numărului de stări (*Geometric*). Forma selectată este vizualizată în câmpul *Preview*.

3. În câmpul *Number of States* se selectează numărul de stări ale automatului.

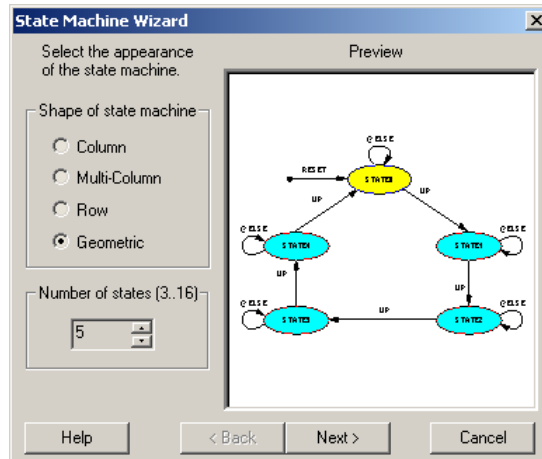


Figura 9.2. Fereastra de dialog *State Machine Wizard*.

4. Se selectează butonul *Next*. Se va deschide fereastra de dialog *Reset The State Machine* (Figura 9.3).

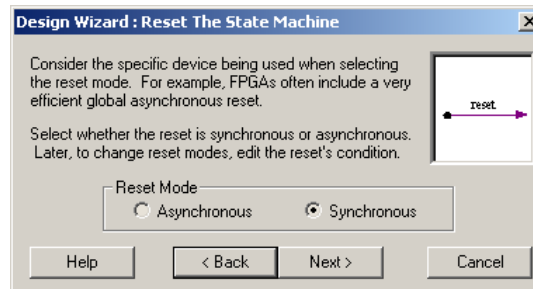


Figura 9.3. Fereastra de dialog *Reset The State Machine*.

5. În câmpul *Reset Mode* al ferestrei afișate, se selectează modul în care se va genera logica de resetare a automatului de stare. Acest mod poate fi asincron (*Asynchronous*), când automatul se va reseta imediat ce semnalul de resetare devine activ, sau sincron (*Synchronous*), când automatul se va reseta la următorul front al semnalului de ceas după activarea semnalului de resetare. Selectarea modului asincron poate fi avantajoasă pentru circuitele FPGA, care conțin de obicei o logică eficientă pentru resetarea globală a circuitului în mod asincron.
6. Se selectează butonul *Next*. Se va deschide fereastra de dialog *Setup Transitions* (Figura 9.4).

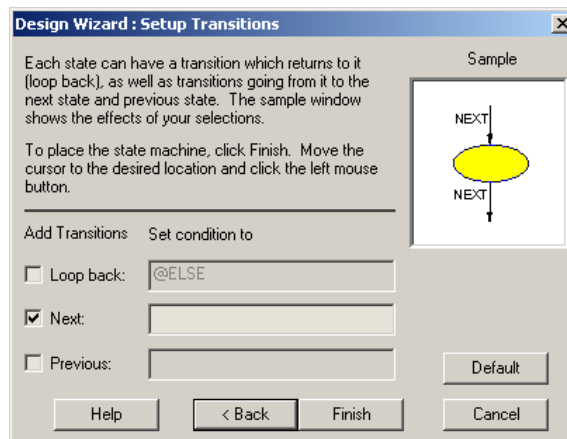


Figura 9.4. Fereastra de dialog *Setup Transitions*.

7. În câmpul *Add Transitions* se selectează tipul tranzițiilor care vor fi prezente în diagrama de stare. În mod implicit, este selectată doar opțiunea *Next*, prin care diagrama va conține tranziții de la fiecare stare la starea următoare a acesteia. Dacă se selectează și opțiunea *Loop back*, diagrama va conține și tranziții de buclare, prin care automatul va rămâne în aceeași stare dacă o anumită condiție nu este îndeplinită. Dacă se selectează și opțiunea *Previous*, diagrama va conține și tranziții de la fiecare stare la starea precedentă a acesteia.
8. Se selectează butonul *Finish*. Se va afișa un dreptunghi cu conturul de culoare verde, care indică marginile diagramei.
9. Se deplasează dreptunghiul în poziția dorită din fereastra editorului și se execută un clic cu butonul din stânga. Se va afișa diagrama de stare în fereastra editorului StateCAD, în modul ilustrat în Figura 9.5 pentru un automat cu patru stări pentru care s-a selectat forma ca o figură geometrică.

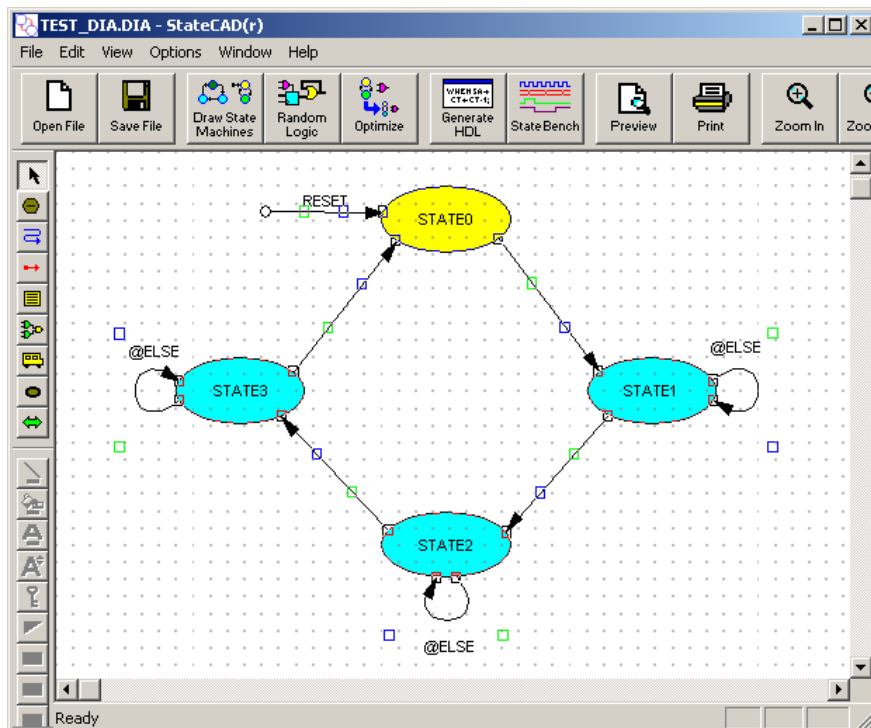



Figura 9.5. Exemplu de diagramă creată cu utilitarul *State Machine Wizard*.

2.4.2. Adăugarea unor noi stări

Pentru adăugarea unor noi stări la un automat de stare creat anterior, se procedează astfel:

1. Se selectează butonul *Draw States* , aflat în partea stângă a ecranului.
2. Se deplasează cursorul în poziția în care trebuie adăugat simbolul stării și se execută un clic cu butonul din stânga. StateCAD atribuie în mod implicit un nume noii stări, de exemplu, STATE4.
3. Se editează starea pentru modificarea numelui stării și specificarea semnalelor de ieșire care trebuie activate în starea respectivă. Aceste operații sunt descrise în secțiunea următoare.

2.4.3. Editarea stărilor

După crearea unui automat de stare cu utilitarul *State Machine Wizard* sau adăugarea unor noi stări, se pot modifica numele stărilor și se pot asigura valorile semnalelor de ieșire care trebuie generate

în stările respective (în cazul automatelor de tip Moore). Pentru aceasta, se poate utiliza fereastra de dialog *Edit State*, în modul prezentat mai jos.

Numele fiecărei stări trebuie să fie un nume unic de variabilă. Atunci când asignarea stărilor se realizează prin metoda cu un bistabil pe stare, numele stărilor se utilizează ca variabile de stare.

Editorul StateCAD asignează în mod automat valori implicite semnalelor de ieșire. Pentru ca semnalele să aibă valorile corecte, trebuie să se specifice valorile semnalelor de ieșire în fiecare stare a automatului. În cazul automatelor de tip Mealy, valorile semnalelor de ieșire se pot specifica la adăugarea condițiilor pentru tranzițiile între stări (secțiunea 2.4.5). Dacă anumite semnale de ieșire nu sunt specificate în mod explicit într-o anumită stare (sau la tranzițiile în acea stare), valoarea care va fi atribuită semnalelor în starea respectivă depinde de modul în care este setată opțiunea de configurație *Retain Output Values*. Această opțiune poate fi setată în secțiunea *Options* a ferestrei de dialog *Configuration*, fereastră care poate fi deschisă prin comanda *Options* → *Configuration*. Valorile semnalelor de ieșire care nu sunt specificate vor fi asigurate de editorul StateCAD astfel:

- Dacă opțiunea *Retain Output Values* nu este activată, semnalele de ieșire care nu sunt specificate în mod explicit vor fi setate la valorile lor inactive.
- Dacă opțiunea *Retain Output Values* este activată, semnalele de ieșire care nu sunt specificate în mod explicit își păstrează valorile de la ultima asignare, până la asignarea unei noi valori.

Editorul StateCAD permite specificarea valorii semnalelor de ieșire în mai multe moduri:

- *Printr-o listă de semnale de tip bit*. Semnalele din listă trebuie separate printr-un spațiu. Semnalele din listă vor fi activate în starea respectivă. Dacă un semnal este precedat de caracterul '!', care indică complementul logic, semnalul va fi dezactivat. De exemplu, considerăm că se specifică următoarea listă:

```
OUT_A !OUT_B
```

Presupunând că semnalele *OUT_A* și *OUT_B* sunt active în starea '1' logic, semnalul *OUT_A* va fi setat la valoarea '1', iar semnalul *OUT_B* va fi setat la valoarea '0'. Nivelul activ al semnalelor poate fi setat prin comanda *Options* → *Variable*, comandă care este descrisă în secțiunea 2.7.

- *Prin asignarea unor valori constante*. Se pot asigna valori constante atât semnalelor de tip bit, cât și vectorilor. Pentru asignare se poate utiliza fie caracterul '=', fie combinația de caractere '<='. Asignările succesive trebuie separate între ele fie prin spații, fie prin caracterul ';'. În exemplele următoare, *OUT_A* și *OUT_B* sunt semnale de tip bit, iar *VEC_A* și *VEC_B* sunt vectori (magistrale):

```
OUT_A = '1'; OUT_B = '0'; VEC_A = 7; VEC_B = "1010";
```

- *Prin ecuații*. Ecuațiile permit asignarea unor expresii booleene semnalelor de tip bit și vectorilor. Ecuațiile pot conține atât semnale de intrare sau de ieșire, cât și variabile de stare. Ecuațiile trebuie separate prin caracterul ';'. Două exemple de ecuații sunt prezentate în continuare:

```
OUT_A <= (IN_A OR IN_B) AND (NUM > 9); NUM <= NUM+1;
```

Editorul StateCAD permite specificarea unor ecuații complexe pentru semnalele de ieșire. De exemplu, se pot introduce ecuațiile unor multiplexoare, sumatoare sau numărătoare care trebuie activate într-o anumită stare. Pentru simplificarea introducerii acestor ecuații, se poate utiliza utilitarul *Logic Wizard*, care poate fi invocat prin selectarea butonului *Output Wizard* din fereastra *Edit State* (Figura 9.6). Utilitarul *Logic Wizard* este descris în secțiunea 2.5.2.

Pentru asignarea sau modificarea numelui unei stări și a valorii semnalelor de ieșire în starea respectivă, se procedează astfel:

1. Se execută un clic dublu pe cercul sau elipsa reprezentând starea care trebuie editată. Se va afișa fereastra de dialog *Edit State* (Figura 9.6).

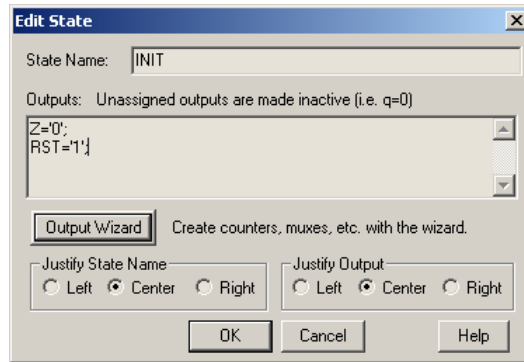


Figura 9.6. Fereastra de dialog *Edit State*.

2. În câmpul *State Name* se introduce numele stării sau se modifică numele existent.
3. În câmpul *Outputs* se introduce lista semnalelor de ieșire care trebuie activate sau dezactivate în starea respectivă, separate prin spații, sau se introduc ecuațiile semnalelor de ieșire, separate prin caracterul ';'.
4. Se selectează butonul *OK*. Ecuațiile semnalelor de ieșire vor fi afișate sub numele stării, separate de acest nume printr-o linie orizontală (Figura 9.7).

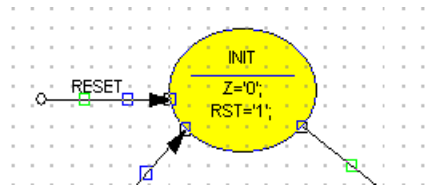



Figura 9.7. Simbolul stării STATE0 din Figura 9.5 după editarea acesteia.


2.4.4. Adăugarea tranzițiilor

Editorul StateCAD permite adăugarea simplă a tranzițiilor între stări. Tranzițiile pot fi reprezentate fie prin linii drepte, fie prin linii curbate. Fiecare linie de tranziție este definită prin două puncte reprezentând capetele liniei, și alte două puncte de control, care permit curbarea liniei. Punctele de capăt sunt reprezentate prin mici pătrate cu contururi de culoare neagră. Primul punct de control este reprezentat printr-un pătrat cu conturul de culoare verde, iar al doilea punct de control este reprezentat printr-un pătrat cu conturul de culoare albastră.


Modul cel mai simplu pentru adăugarea unei tranziții este prin trasarea unei linii drepte. În acest caz, utilizatorul indică doar punctele de capăt, cele două puncte de control fiind plasate în mod automat de către editorul StateCAD. Ulterior, linia dreaptă poate fi transformată într-o linie curbă prin re poziționarea punctelor de control. Pentru adăugarea unei tranziții sub forma unei linii drepte, se procedează astfel:


1. Se selectează butonul *Draw Transitions* , aflat în partea stângă a ecranului.
2. Se selectează starea de la care trebuie efectuată tranziția, executând un clic în interiorul stării respective, iar apoi se execută un nou clic în interiorul stării, în apropierea marginii de la care trebuie să pornească tranziția. Va apare un pătrat de culoare roșie pe marginea simbolului stării, iar un alt pătrat va urmări deplasarea cursorului.
3. Se deplasează cursorul deasupra stării destinație. Va apare un pătrat de culoare roșie pe marginea simbolului stării și o linie între cele două stări, sursă și destinație. Se fixează poziția punctului de capăt al liniei și se execută un clic în poziția respectivă. Editorul va adăuga în mod automat punctele de control pe linia de tranziție.

Pentru adăugarea unei tranziții sub forma unei linii curbe, se procedează astfel:

1. Se selectează butonul *Draw Transitions* , aflat în partea stângă a ecranului.
2. Se selectează starea de la care trebuie efectuată tranziția, iar apoi se execută un clic în interiorul stării, în apropierea marginii de la care trebuie să pornească tranziția. Va apare un pătrat de culoare roșie pe marginea simbolului stării, iar un alt pătrat va urmări deplasarea cursorului (acesta va fi primul punct de control).
3. Se execută un clic în afara stării pentru a fixa poziția primului punct de control. La capătul cursorului se va atașa al doilea punct de control.
4. Se fixează poziția celui de-al doilea punct de control și se execută un clic în acea poziție. La capătul cursorului se va atașa al doilea punct de capăt.
5. Se deplasează cursorul deasupra stării destinație, se fixează poziția punctului de capăt al liniei și se execută un clic în poziția respectivă.

Pentru adăugarea unei tranziții de buclare, care începe și se termină în aceeași stare, se procedează în felul următor:

1. Se selectează butonul *Draw Transitions* .
2. Se selectează starea pentru care trebuie adăugată tranziția de buclare, iar apoi se execută un clic dublu în apropierea marginii unde se dorește să apară tranziția. Editorul va adăuga linia pentru tranziția de buclare și condiția @ELSE pentru această tranziție. Condiția @ELSE este descrisă în secțiunea următoare.

După adăugarea tranzițiilor dorite, se selectează butonul *Select Objects*  pentru a termina adăugarea tranzițiilor.

2.4.5. Adăugarea condițiilor pentru tranziții

O condiție a unei tranziții determină executarea tranziției respective atunci când condiția devine adevărată. Pentru adăugarea condiției care va determina executarea unei tranziții, se execută un clic dublu pe linia tranziției respective. Se va deschide fereastra de dialog *Edit Condition* (Figura 9.8). Condiția trebuie introdusă în câmpul *Condition*.

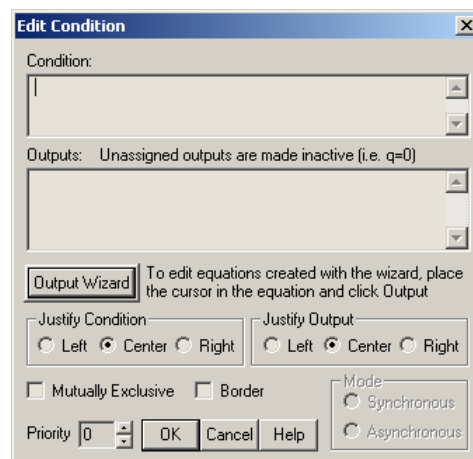


Figura 9.8. Fereastra de dialog *Edit Condition*.

O condiție poate fi specificată fie ca o ecuație booleană, fie ca o expresie booleană care se evaluează la valoarea TRUE sau FALSE. O ecuație booleană poate conține semnale de intrare, semnale de ieșire și variabile logice. Erorile de sintaxă din ecuațiile condițiilor și condițiile nedeterminate sunt detectate la compilarea diagramei de stare. Un exemplu simplu de condiție specificată printr-o expresie booleană constă dintr-un singur semnal de tip bit; condiția va fi adevărată atunci când semnalul este activ. Dacă semnalul este precedat de caracterul '!', condiția va fi adevărată atunci când

semnalul este inactiv. În Figura 9.9, tranziția din starea A în starea B se efectuează atunci când semnalul X_{in} devine activ. Alte expresii booleene simple constau din semnale conectate prin operatori logici.

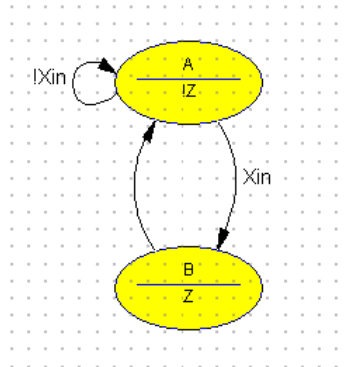


Figura 9.9. Condiții de tranziție specificate prin expresii booleene constând dintr-un singur semnal.

În cazul automatelor de stare de tip Mealy, se pot specifica semnalele de ieșire care trebuie activate pentru fiecare tranziție. Ecuatiile acestor semnale trebuie introduse în câmpul *Outputs*. Aceste ecuații nu pot conține variabile de stare sau variabile logice. Se pot defini însă ecuații complexe pentru circuite logice cum sunt multiplexoare, registre sau numărătoare.

Există două condiții speciale: condiția adevărată în mod implicit și condiția ELSE. O tranziție căreia nu i se asociază nici o condiție se execută întotdeauna, astfel încât condiția unei asemenea tranziții este adevărată întotdeauna. Această condiție este deci *adevărată în mod implicit*. O condiție poate fi specificată ca adevărată în mod explicit dacă pentru condiția respectivă se indică valoarea TRUE.

Condiția adevărată în mod implicit se utilizează atunci când automatul de stare trebuie să efectueze tranziții într-o anumită ordine care este independentă de semnalele de intrare. Un asemenea automat de stare poate fi inițializat totuși cu un semnal de resetare. Atunci când condiția de resetare este adevărată, se execută tranziția de resetare, care este mai prioritară decât condiția adevărată în mod implicit.

De multe ori, o stare poate avea mai multe tranziții în alte stări, iar una din tranziții trebuie executată atunci când nici o condiție a unei alte tranziții nu este adevărată. Aceasta este tranziția ELSE, iar condiția acestei tranziții este numită *condiție ELSE*. Condiția ELSE este indicată printr-un operator special, **@ELSE**. Tranziția ELSE are prioritatea maximă dintre toate tranzițiile dintr-o stare.

Editorul StateCAD permite de asemenea *condiția implicită ELSE*, care permite ca un automat să rămână într-o anumită stare dacă nici una din condițiile tranzițiilor din acea stare nu este adevărată. Astfel, nu este necesară adăugarea unei tranziții de buclare la starea respectivă, ceea ce simplifică diagramele de stare. Această facilitate este activată în mod implicit, dar poate fi dezactivată prin deselectarea opțiunii *Implied Else* din secțiunea *Options* a ferestrei de dialog *Configuration* (deschisă prin comanda *Options* → *Configuration*).

2.4.6. Adăugarea condiției de resetare a automatului



Editorul StateCAD permite specificarea unei condiții de resetare a automatului de stare. Atunci când condiția de resetare a automatului de stare devine adevărată, indiferent de starea în care se află automatul, se execută tranziția într-o stare de resetare specificată. Această stare este de obicei și starea inițială a automatului. La tranziția în starea de resetare, semnalele de ieșire vor fi setate la valorile specificate pentru această stare. Pentru un automat de stare se poate specifica o condiție de resetare sincronă și o condiție de resetare asincronă.

La adăugarea unei condiții de resetare, trebuie să se specifice dacă resetarea este sincronă sau asincronă. După selectarea modului de resetare, editorul StateCAD setează în mod automat ca și condiție de resetare ultima condiție utilizată pentru o resetare cu același mod. Pentru modificarea condiției, se execută un clic dublu pe condiția respectivă și se editează.

La compilarea automatului de stare, se execută operația ȘI logic între complementul logic al condiției de resetare sincronă și fiecare din condițiile de tranziție ale automatului de stare. Deci, com-

plementul logic al condiției de resetare este implicit pentru fiecare tranziție și nu trebuie adăugat manual.

Pentru adăugarea condiției de resetare a automatului, se execută următoarele operații:

1. Se selectează butonul *Draw Reset* , aflat în partea stângă a ecranului.
2. Se execută un clic în apropierea stării în care trebuie realizată tranziția la resetarea automatului. Va apare punctul de început al liniei de tranziție, iar primul punct de control va fi atașat la cursor.
3. Se continuă în același mod ca și la adăugarea unei tranziții obișnuite, fie sub forma unei linii drepte, fie sub forma unei linii curbe (secțiunea 2.4.4).
4. După terminarea adăugării tranziției de resetare, se va afișa un mesaj pentru specificarea modului de resetare. Se selectează butonul *Yes* dacă se dorește o resetare asincronă, sau butonul *No* dacă se dorește o resetare sincronă. Condiția de resetare va fi adăugată automat pe baza ultimei condiții utilizate.
5. Se selectează butonul *Select Objects*  pentru a termina adăugarea tranziției de resetare.
6. Dacă este necesar, se modifică condiția de resetare adăugată automat de editorul StateCAD. Pentru aceasta, se execută un clic dublu pe condiția de resetare, iar în fereastra *Edit Condition* se editează condiția în câmpul *Condition*. În câmpul *Mode* al aceleiași ferestre (câmp aflat în colțul din dreapta jos) se poate modifica și modul de resetare prin selectarea uneia din opțiunile *Asynchronous* sau *Synchronous*.

Observație

Pentru a fi posibilă sinteza unui automat cu resetare asincronă, circuitul programabil care va fi utilizat pentru implementare trebuie să dispună de semnale de resetare asincronă. În caz contrar, sinteza nu va putea fi executată. În acest caz, trebuie să se utilizeze resetarea sincronă. Circuitele FPGA Xilinx dispun de semnale globale de setare și resetare asincronă.

2.5. Adăugarea unor circuite logice suplimentare

2.5.1. Utilizarea circuitelor logice

Utilizarea diagramelor de stare este avantajoasă în special pentru proiectarea blocurilor de control ale sistemelor digitale în general și ale automatelor de stare în particular. Pe lângă aceste blocuri, sistemele digitale necesită circuite logice suplimentare pentru implementarea funcțiilor de ieșire și a unor elemente de prelucrare care nu pot fi specificate în mod simplu cu ajutorul diagramelor de stare. De aceea, este utilă completarea diagramelor de stare cu descrierea prin alte metode a circuitelor logice suplimentare necesare pentru proiectarea unor sisteme digitale complexe.

Editorul pentru automate de stare StateCAD permite adăugarea unor circuite logice suplimentare la sistemul digital proiectat. Aceste circuite sunt adăugate sub forma unor blocuri logice combinaționale sau secvențiale, care pot fi descrise prin ecuații cu o sintaxă relativ simplă. Pentru simplificarea definirii ecuațiilor care descriu funcționarea circuitelor logice adăugate, există un utilitar numit *Logic Wizard*. Cu ajutorul acestuia se pot adăuga în mod simplu circuite combinaționale sau secvențiale dintr-un set predefinit, cum sunt: porți logice, multiplexoare, sumatoare, comparatoare, bistabile, registre, numărătoare, registre de deplasare. Unele din aceste circuite sunt definite doar pentru semnale de tip bit, iar altele pot fi configurate cu ajutorul utilitarului *Logic Wizard* atât pentru semnale de tip bit, cât și pentru vectori. Acest utilitar definește în mod automat ecuațiile circuitelor care au fost configurate de proiectant. Circuitele adăugate la sistemul digital pot fi reprezentate fie sub formă grafică, fie sub formă de text, prin ecuațiile lor.

Proiectantul poate edita ecuațiile circuitelor predefinite pentru modificarea funcției circuitelor, sau poate defini noi circuite prin specificarea ecuațiilor acestora. Aceste ecuații pot conține operatori logici pentru definirea unor elemente combinaționale utilizate în mod frecvent pentru generarea sem-



nalelor de intrare ale automatelor de stare, cum sunt decodificatoarele de adresă. De asemenea, ecuațiile pot conține nume de stări sau variabile de stare, ceea ce permite specificarea unor circuite complexe într-o formă concisă. La compilarea ecuațiilor logice care conțin numele unor stări, aceste nume sunt înlocuite cu asignarea stărilor respective.

La editarea unei ecuații logice, există posibilitatea specificării unui semnal care, atunci când este activat, va determina trecerea semnalului descris prin ecuația respectivă în starea de înaltă impedanță. Circuitele secvențiale descrise prin ecuațiile logice pot fi resetate în mod sincron sau asincron.

2.5.2. Adăugarea circuitelor logice cu utilitarul Logic Wizard

Utilitarul *Logic Wizard* permite adăugarea simplă a unor circuite logice combinaționale și secvențiale uzuale. Acest utilitar permite proiectantului selectarea dintr-un set de circuite predefinite a circuitului care trebuie adăugat, iar apoi parametrizarea circuitului prin indicarea numelui semnalelor de intrare și de ieșire ale acestuia, a tipului semnalelor (biți sau vectori) și a dimensiunii semnalelor care sunt vectori. Pe baza setărilor proiectantului, utilitarul generează în mod automat ecuațiile circuitului, definește porturile de I/E ale acestuia și adaugă semnalele de control necesare funcționării circuitului.

Pentru adăugarea unui circuit cu ajutorul utilitarului *Logic Wizard*, se execută următoarele operații:

1. Se selectează butonul *Random Logic*  aflat în partea de sus a ecranului sau butonul *Logic Wizard*  aflat în partea stângă a ecranului.
2. Se deplasează cursorul deasupra unei zone libere de pe ecran și se execută un clic cu butonul din stânga. Se va deschide fereastra de dialog *Logic Wizard*.
3. Se selectează tipul circuitului care trebuie adăugat din lista afișată în partea din stânga sus a ferestrei. Conținutul ferestrei se va actualiza în funcție de circuitul selectat. Astfel, în zona centrală a ferestrei se afișează simbolul grafic al circuitului selectat, iar în partea din dreapta sus a ferestrei se afișează descrierea textuală a circuitului. Figura 9.10 ilustrează fereastra *Logic Wizard* în care s-a selectat un registru de deplasare la stânga (circuitul *Shift Left*).

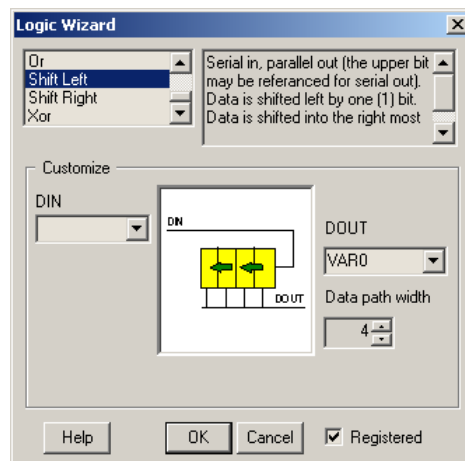


Figura 9.10. Fereastra de dialog *Logic Wizard*.

4. În partea din stânga simbolului grafic al circuitului se completează numele semnalului sau al semnalelor de intrare ale circuitului. În Figura 9.10, numele semnalului de intrare se introduce în câmpul *DIN*.
5. În partea din dreapta simbolului grafic al circuitului se completează numele semnalului de ieșire al circuitului. În Figura 9.10, numele semnalului de ieșire se introduce în câmpul *DOUT*.

6. În câmpul *Data path width* se selectează dimensiunea semnalelor de intrare și/sau de ieșire care sunt vectori. Această dimensiune poate fi de până la 32 de biți. Unele circuite sunt disponibile doar pentru semnale de tip bit, iar la acestea dimensiunea semnalelor este setată la 1 și nu poate fi modificată.
7. Se selectează tipul semnalului de ieșire. Dacă opțiunea *Registered* nu este selectată, semnalul de ieșire va fi combinațional, iar dacă această opțiune este selectată, semnalul de ieșire va fi secvențial. Valoarea unui semnal de ieșire combinațional se modifică imediat ce se modifică semnalele de intrare de care depinde semnalul de ieșire. Valoarea unui semnal de ieșire secvențial se modifică doar la apariția unui front al semnalului de ceas, după modificarea semnalelor de intrare de care depinde semnalul de ieșire. Există cazuri tipice când semnalul de ieșire este combinațional (de exemplu, în cazul multiplexoarelor), și altele când semnalul de ieșire este secvențial (de exemplu, în cazul numărătoarelor).
8. Se selectează butonul *OK*. Utilitarul *Logic Wizard* verifică validitatea numelor semnalelor, generează ecuația logică a circuitului și definește vectorii noi care au fost specificați. Se va deschide apoi fereastra de dialog *Edit Equation* (Figura 9.11).

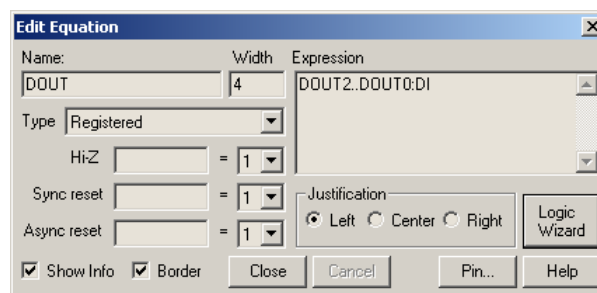



Figura 9.11. Fereastra de dialog *Edit Equation*.

9. În fereastra *Edit Equation* se poate edita ecuația logică generată. În câmpul *Name* se afișează numele semnalului de ieșire al circuitului care a fost specificat în fereastra *Logic Wizard*. În Figura 9.11, acest nume este *DOUT*. În câmpul *Width* apare dimensiunea semnalului, iar în câmpul *Expression* se afișează ecuația semnalului din câmpul *Name*. Tipul semnalului (combinațional sau secvențial) se afișează în câmpul *Type*. În mod normal, nu este necesar să se modifice conținutul acestor câmpuri. Dacă se dorește modificarea funcției circuitului, trebuie să se modifice ecuația acestuia care a fost generată în mod automat. Sintaxa ecuațiilor logice este descrisă în secțiunea următoare.
10. Opțional, în câmpul *Hi-Z* se poate introduce numele unui semnal de control care, atunci când este activ, determină trecerea semnalului de ieșire al circuitului în starea de înaltă impedanță. În caseta alăturată se selectează nivelul activ al semnalului de control (1 sau 0).
11. În câmpurile *Sync reset* și *Async reset* se pot introduce numele unor semnale pentru resetarea sincronă, respectiv asincronă a circuitului. În casetele alăturate se specifică nivelul activ al semnalului de resetare respectiv (1 sau 0).
12. Se selectează butonul *Close* pentru închiderea ferestrei *Edit Equation*. Dacă ecuația din câmpul *Expression* nu a fost modificată, simbolul grafic al circuitului va fi afișat pe ecran (Figura 9.12).
13. Se selectează butonul *Select Objects*  dacă nu mai trebuie adăugate alte circuite.

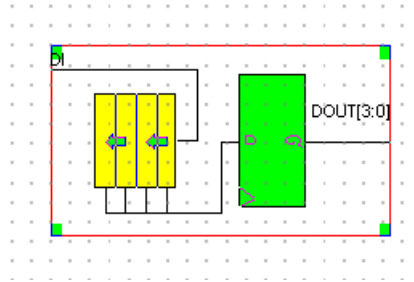


Figura 9.12. Simbolul unui registru de deplasare la stânga de 4 biți generat cu utilitarul *Logic Wizard*.

Observație

Utilitarul *Logic Wizard* poate fi invocat și la editarea stărilor pentru introducerea semnalelor de ieșire activate într-o anumită stare (la automatele de tip Moore) sau la adăugarea condițiilor pentru tranziții pentru introducerea semnalelor de ieșire activate la efectuarea unei anumite tranziții (la automatele de tip Mealy). Pentru acesta, se selectează butonul *Output Wizard* în fereastra *Edit State*, respectiv *Edit Condition*. Circuitele necesare pentru generarea acestor semnale nu vor fi afișate în mod explicit pe ecran.

Circuitele logice adăugate cu utilitarul *Logic Wizard* sunt afișate pe ecran fie sub formă grafică, fie sub formă textuală. În mod implicit, circuitele sunt afișate sub formă grafică. Modul de afișare se poate modifica prin comanda *View* → *Show Logic Graphics*. Dacă opțiunea de afișare sub formă grafică este dezactivată, fiecare circuit este reprezentat printr-o casetă conținând ecuația circuitului respectiv. Circuitele ale căror ecuații au fost modificate de proiectant vor fi afișate întotdeauna sub formă textuală.

Pentru editarea ecuației unui circuit adăugat anterior, se selectează reprezentarea circuitului (fie cea grafică, fie cea textuală) și se execută un clic în interiorul dreptunghiului selectat. Va apare fereastra *Edit Equation*, care permite editarea ecuației circuitului.

2.5.3. Ecuații și operatori

Ecuațiile se utilizează pentru specificarea unor condiții și pentru descrierea funcționării unor circuite logice diferite de cele care pot fi adăugate cu utilitarul *Logic Wizard*. Ecuațiile pot conține variabile (de tip bit sau vectori), constante și operatori. Dacă o ecuație conține variabile de tip bit și vectori, variabilele de tip bit sunt expandați automat în vectori cu dimensiunea corespunzătoare vectorului destinație. Precedența operatorilor trebuie indicată prin utilizarea parantezelor.

Operatorii pot fi grupați în următoarele categorii:

- Operatori logici (booleeni);
- Operatori de comparație;
- Operatori de adunare și scădere;
- Operatorul de specificare a domeniului;
- Operatorul de concatenare.

Operatorii logici permit efectuarea unor operații booleene între două variabile de tip bit sau între pozițiile binare de același rang a doi vectori. Un bit individual al unui vector este indicat prin adăugarea numărului bitului la numele vectorului. De exemplu, ADR_{15} indică bitul 15 al vectorului ADR . Operatorii logici sunt prezentați în Tabelul 9.2.

Tabelul 9.2. Operatori logici.

Operator	Semnificație
# sau sau OR	SAU logic
& sau AND	ȘI logic
! sau NOT	Complement logic

Operatorii de comparație permit compararea variabilelor de tip bit sau a vectorilor cu constante. Rezultatele comparațiilor sunt valori booleene (TRUE sau FALSE), care pot fi combinate prin operatori logici pentru a forma condiții mai complexe. Operatorii de comparație au o precedență mai ridicată decât operatorii logici SAU, ȘI. La combinarea operațiilor de comparație prin operatori logici, comparațiile trebuie incluse în paranteze, ca în exemplul următor:

$$(V > 9) \ \& \ (V < 16)$$

Tabelul 9.3 prezintă operatorii de comparație.

Tabelul 9.3. Operatorii de comparație.

Operator	Semnificație
= sau ==	Egal
<> sau !=	Diferit
<	Mai mic
<=	Mai mic sau egal
>	Mai mare
>=	Mai mare sau egal

Operatorii de adunare și scădere (+, -) se pot utiliza pentru adunarea sau scăderea a doi vectori. De asemenea, acești operatori permit adunarea unei constante la un vector sau scăderea unei constante dintr-un vector.

Operatorul de specificare a domeniului este reprezentat prin două puncte succesive (. .). Acest operator indică un grup de biți dintr-un vector. De exemplu, A1 . . A0 indică biții 1 și 0 din vectorul A.

Operatorul de concatenare, reprezentat prin caracterul ':', permite gruparea unor variabile de tip bit sau a unor vectori pentru a forma vectori de dimensiune mai mare. Vectorii obținuți pot fi utilizați pentru compararea lor cu valori constante, ca în exemplul următor:

$$(D3:D2:D1:D0 > 9) \ \& \ (D3:D2:D1:D0 < 16)$$

Operatorul de concatenare poate fi utilizat și pentru specificarea operațiilor de deplasare și de rotire a registrelor. Exemplul următor indică operația de rotire la stânga a unui registru de 8 biți:

$$D = D6..D0:D7$$

Constantele 0 și 1 pot fi concatenate cu variabile de tip bit sau cu vectori, ca în exemplul următor:

$$REZ = 0:0:A1:A0$$

2.6. Adăugarea și setarea valorii vectorilor

Vectorii (magistralele) permit gruparea unor variabile de tip bit care vor fi referite prin același nume. Astfel, se simplifică setarea valorii unor variabile, compararea a două variabile sau compararea unei variabile cu o constantă. La adăugarea unor circuite logice cu utilitarul *Logic Wizard*, vectorii necesari pentru utilizarea acestor circuite sunt adăugați în mod automat. La adăugarea unor circuite logice prin specificarea ecuației acestora sau la adăugarea unor condiții pentru tranziții care utilizează vectori, proiectantul trebuie să adauge în mod explicit vectorii utilizați, specificând dimensiunea acestora prin domeniul biților.

2.6.1. Adăugarea vectorilor

Pentru adăugarea unui vector se procedează astfel:

1. Se selectează butonul *Draw Vectors* , aflat în partea stângă a ecranului.

2. Se deplasează cursorul deasupra unei zone libere de pe ecran și se execută un clic cu butonul din stânga. Va apărea simbolul unui vector cu numele implicit VAR_n , unde n este un întreg, și cu dimensiunea implicită 8 (Figura 9.13). Dimensiunea este indicată prin domeniul biților, sub forma $[7:0]$.

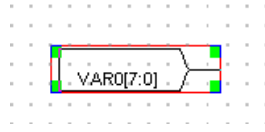


Figura 9.13. Simbolul grafic al unui vector cu numele și dimensiunea implicită.

3. Pentru modificarea numelui și dimensiunii vectorului, se execută un clic în interiorul dreptunghiului care încadrează simbolul vectorului care este selectat în mod automat după adăugarea acestuia, sau se execută un clic dublu deasupra simbolului acestuia. Se va deschide fereastra de dialog *Edit Vector* (Figura 9.14).

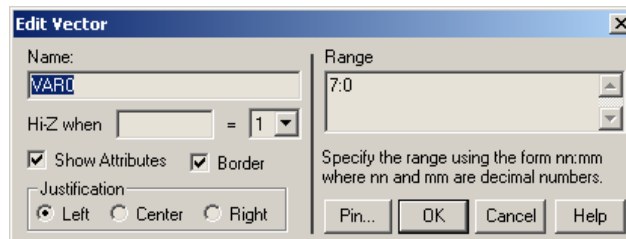



Figura 9.14. Fereastra de dialog *Edit Vector*.

4. Se modifică numele vectorului în câmpul *Name*: și domeniul biților în câmpul *Range*.
5. Opțional, în câmpul *Hi-Z* se poate introduce numele unui semnal care, atunci când este activ, determină trecerea magistralei care reprezintă vectorul în starea de înaltă impedanță. Nivelul activ al semnalului poate fi setat la 1 sau la 0 în caseta alăturată.
6. Se selectează butonul *OK* pentru închiderea ferestrei de dialog.
7. Se selectează butonul *Select Objects*  pentru terminarea adăugării vectorilor.

În cazul în care opțiunea *Show Logic Graphics* din meniul *View* este dezactivată, în locul simbolului grafic al vectorului se afișează un dreptunghi conținând definiția vectorului. De exemplu, vectorul din Figura 9.13 are definiția următoare:

$$VAR0 [] = 7:0$$

Aceasta este definiția vectorului prin indicarea domeniului biților. Bitul cel mai semnificativ al vectorului este indicat de prima cifră; în exemplul anterior, acesta este bitul 7. Dacă se dorește ca bitul cel mai semnificativ să fie bitul 0, domeniul trebuie indicat sub forma 0:7.

Un vector poate fi definit și prin listarea numelui unor variabile care vor fi grupate în vectorul respectiv, separate prin spații. Lista nu poate conține variabila de ceas, nume de stări sau alți vectori. O variabilă poate apărea o singură dată în listă. Exemplele următoare indică definițiile echivalente ale unui vector N de 4 biți:

$$\begin{aligned} N[] &= 3:0 \\ N[] &= N3 N2 N1 N0 \end{aligned}$$

2.6.2. Setarea valorii vectorilor

Pentru setarea valorii unui vector se poate utiliza unul din formatele următoare:

$$\begin{aligned} &vector[constantă] \\ &vector <= ecuație \\ &vector = ecuație \end{aligned}$$

unde *vector* este numele vectorului, *constantă* este o constantă având o valoare mai mică sau egală cu valoarea maximă posibilă a vectorului, iar *ecuație* este o ecuație logică. Un exemplu utilizând primul format este următorul:

$$A[\wedge 5A]$$

Dacă vectorul A are valoarea 5Ah și vectorul B are valoarea 3Bh, următoarea expresie este adevărată:

$$A[\wedge 5A] \ \& \ B[\wedge 3B]$$

Al doilea și al treilea format se poate utiliza pentru asignarea unor ecuații la vectorii de ieșire; acești vectori nu pot conține variabile de stare. Ecuația din partea dreaptă a unei asignări trebuie să se evalueze la un vector cu același număr de biți ca și vectorul din partea stângă a asignării. Ecuațiile pot conține operatori logici, operatori de adunare și scădere, operatori de comparație și expresii de tip bit (care sunt expandați automat la dimensiunea corectă).

Biții individuali ai unui vector pot fi setați utilizând ecuații care conțin măști corespunzătoare pentru fiecare bit. De exemplu, pentru a seta bitul 3 al vectorului N la valoarea A#B, bitul 2 la valoarea C&D, bitul 1 la valoarea !A, iar bitul 0 la valoarea TRUE, se poate scrie ecuația următoare:

$$N = (8 \ \& \ (A\#B)) \ \# \ (4 \ \& \ (C\&D)) \ \# \ (2 \ \& \ !A) \ \# \ (1)$$

Atunci când un vector se utilizează ca semnal de ieșire al unei stări sau al unei tranziții, se pot asigna biții individuali ai vectorului, fără a fi necesară scrierea ecuației complete pentru întregul vector. Ecuația din exemplu anterior poate fi scrisă sub forma a patru ecuații separate pentru biții individuali ai vectorului:

```
N3 = A#B;
N2 = C&D;
N1 = !A;
N0 = '1';
```

2.7. Adăugarea variabilelor și modificarea atributelor acestora

Variabilele se utilizează pentru identificarea semnalelor și a stărilor. Toate variabilele utilizate pentru descrierea unui automat de stare trebuie definite. Variabilele pot fi definite (adăugate) fie în mod automat de către editorul StateCAD, fie în mod explicit de proiectant. Fiecare variabilă are mai multe atribute. Principalele atribute sunt descrise în continuare.

- *Type*. Reprezintă tipul variabilei, care indică dacă variabila identifică un semnal de intrare, un semnal de ieșire, un semnal de ceas sau o stare. Semnalele de intrare se utilizează pentru specificarea condițiilor pentru tranziții și în ecuațiile logice. Semnalele de ieșire sunt setate de automatul de stare ca rezultat al trecerii în anumite stări sau al anumitor tranziții. Semnalul de ceas este utilizat de circuitele secvențiale din cadrul automatului. Acest semnal este adăugat în mod automat de editorul StateCAD. Variabilele de stare identifică stările automatului. Un set de variabile de stare va avea o valoare unică pentru fiecare stare. Semnalele de ieșire pot fi combinaționale (*Combinational*) sau secvențiale (*Registered*). Semnalele combinaționale își modifică valoarea imediat ce se modifică semnalele de care depind. Semnalele secvențiale nu își modifică valoarea imediat, ci numai la apariția unui front activ al semnalului de ceas.
- *Active*. Indică nivelul activ al variabilei (0 sau 1). Dacă nivelul activ al unei variabile este 0, într-o ecuație valoarea logică a variabilei este adevărată dacă variabila este 0. În mod implicit, variabilele adăugate în mod automat vor avea nivelul activ 1. Nivelul activ implicit al variabilelor poate fi setat în fereastra *Variables*, descrisă mai jos. Variabilele sunt inițializate implicit cu nivelele lor inactive.
- *Pin/Node*. Acest atribut indică dacă semnalul la care se referă variabila este vizibil din exteriorul automatului (este un pin de I/E) sau este un semnal intern (nod) care nu este vizibil din exterior.

- *Pin Number*. Indică numărul pinului din circuitul programabil asignat semnalului la care se referă variabila. Acest atribut se poate utiliza numai dacă limbajul în care va fi translatată diagrama de stare este ABEL.
- *Hi-Z*. Acest atribut indică dacă semnalul la care se referă variabila poate fi trecut în starea de înaltă impedanță prin activarea unui semnal de control. Nivelul activ al semnalului de control poate fi setat la 1 sau la 0.

Implicit, variabilele sunt adăugate în mod automat de către editorul StateCAD. În acest caz, atunci când se compilează un automat de stare și se întâlnește numele unei noi variabile, aceasta va fi adăugată în mod automat, tipul variabilei adăugate fiind determinat din contextul în care se întâlnește numele variabilei. Adăugarea automată a variabilelor poate fi dezactivată prin deselectionarea opțiunii *Add Variables Automatically* din secțiunea *Options* a ferestrei de dialog *Configuration* (deschisă prin comanda *Options* → *Configuration*). Dacă adăugarea automată a variabilelor este dezactivată, la compilarea automatului și întâlnirea unei noi variabile se va afișa un mesaj prin care proiectantului i se solicită să confirme adăugarea variabilei. Astfel, proiectantul poate verifica dacă numele variabilei este corect și va cunoaște tipul care va fi atribuit variabilei (de intrare, de ieșire sau de stare).

Pentru adăugarea unei variabile sau modificarea atributelor unei variabile existente, se procedează astfel:

1. Se selectează comanda *Options* → *Variable*, sau se execută un clic cu butonul din dreapta în orice zonă a ecranului și se selectează opțiunea *Variable*. Se va deschide fereastra de dialog *Variables* (Figura 9.15). În această fereastră sunt afișate numele variabilelor definite și atribuțiile lor.

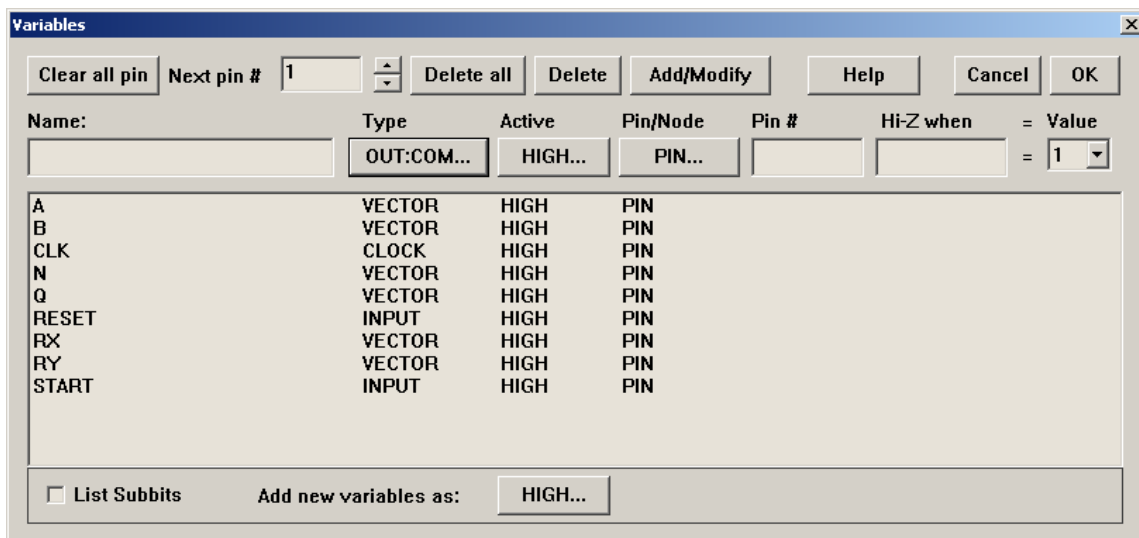


Figura 9.15. Fereastra de dialog *Variables*.

2. Se introduce numele variabilei care trebuie adăugată în câmpul *Name:*, sau se selectează variabila ale cărei atribute trebuie modificate.
3. Se selectează tipul variabilei prin selectarea succesivă a butonului de sub câmpul *Type*, până când se afișează tipul dorit. Tipul va fi modificat succesiv în **INPUT** (intrare), **OUT:COM** (ieșire combinațională), **OUT:REG** (ieșire secvențială) sau **STATE** (stare). Pentru asignarea unei variabile ca variabilă de stare (atributul STATE), este necesară selectarea asignării manuale a stărilor sau a asignării binare. Metoda de asignare a stărilor poate fi selectată în fereastra de dialog *Configuration*, deschisă prin comanda *Options* → *Configuration*. În secțiunea *State Assignment* a acestei ferestre se selectează opțiunea *Manual* pentru asignarea manuală a stărilor, sau opțiunea *Binary Encode* pentru asignarea binară. Tipul variabilei de ceas (CLOCK) și cel al variabilelor de stare (STATE) sau al vectorilor (VECTOR) nu poate fi modificat.

4. Dacă este necesar, se modifică nivelul activ al variabilei prin selectarea butonului de sub câmpul *Active*. Nivelul va fi modificat din **HIGH** (1) în **LOW** (0) sau invers. Nivelul activ implicit al variabilelor care vor fi adăugate ulterior poate fi modificat prin comutarea butonului din dreapta câmpului *Add new variables as:* din partea de jos a ferestrei.
5. Dacă este necesar, se modifică atributul *Pin/Node* al variabilei prin selectarea butonului de sub câmpul *Pin/Node*. Atributul va fi modificat din **PIN** în **NODE** sau invers.
6. În mod opțional, se poate introduce în câmpul *Hi-Z when* numele unui semnal de control care, atunci când este activ, determină trecerea în starea de înaltă impedanță a semnalului la care se referă variabila selectată. Nivelul activ al acestui semnal de control poate fi selectat în caseta *Value*.
7. Se selectează butonul *OK* pentru închiderea ferestrei de dialog.

2.8. Compilarea diagramei de stare

După editarea diagramei de stare, aceasta poate fi compilată pentru generarea unui cod într-un limbaj de descriere hardware (VHDL, Verilog sau ABEL). Codul obținut poate fi utilizat apoi pentru simularea sau sinteza automatului de stare. La compilarea diagramei de stare, aceasta este analizată pentru a se detecta nu numai erorile sintactice, ci și un număr mare de erori logice (de proiectare).

Pentru compilarea diagramei de stare, se selectează butonul *Generate HDL* sau se execută comanda *Options* → *Compile*. Dacă în timpul compilării se detectează o eroare, compilarea este oprită, iar în fereastra *View Error* se afișează problema găsită și se propune o soluție pentru rezolvarea acesteia (Figura 9.16). Se poate selecta butonul *Help* pentru afișarea unor informații suplimentare despre eroarea detectată.

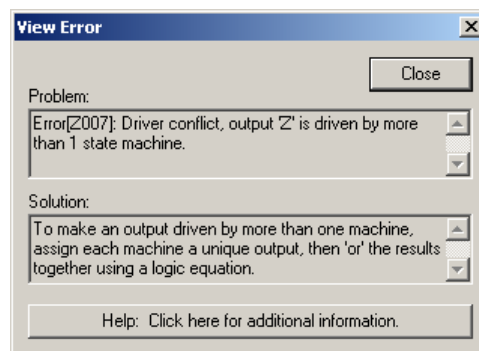


Figura 9.16. Fereastra *View Error* pentru afișarea unui mesaj de eroare la compilare.

Dacă în timpul compilării se detectează probleme de proiectare, acestea sunt afișate sub forma unor mesaje de avertisment în fereastra *View Warnings*. Se explică fiecare problemă detectată și se sugerează soluții pentru rezolvarea acestora.

Dacă opțiunea *Optimize Port I/O* din fereastra *Configuration* este validată, la compilare se testează toate semnalele pentru a determina dacă acestea sunt semnale interne de reacție din cadrul automatului de stare. Dacă se detectează un astfel de semnal care este vizibil în afara automatului (este un pin de I/E), se afișează fereastra de dialog *Optimize I/O Port*. În această fereastră se selectează semnalele care vor fi optimizate prin transformarea lor în semnale interne (noduri), după care se selectează butonul *Optimize*. De asemenea, optimizarea se poate dezactiva prin selectarea opțiunii *Disable Optimize Port I/O* sau optimizarea se poate abandona pentru compilarea curentă prin selectarea butonului *Cancel*.

Dacă nu se detectează erori sau avertismente, StateCAD generează descrierea HDL corespunzătoare diagramei de stare și se afișează fereastra *Results* cu informații despre rezultatele compilării (Figura 9.17). Se indică numărul liniilor sursă generate, timpul în care s-a executat compilarea, numă-

rul de stări și de tranziții ale diagramei de stare, ca și operațiile de analiză care au fost executate de compilator.

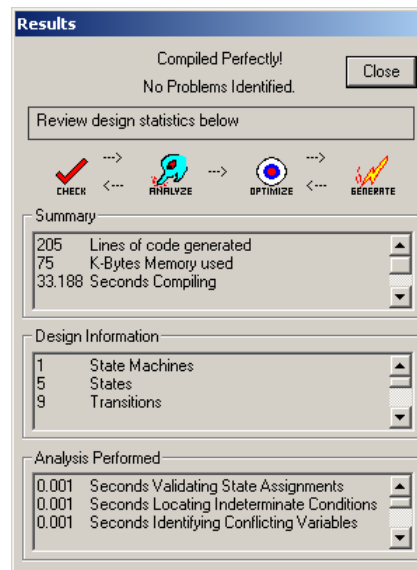


Figura 9.17. Fereastra *Results* pentru afișarea informațiilor despre rezultatele compilării.

După închiderea ferestrei *Results*, StateCAD deschide automat fereastra *StateCAD HDL Browser* pentru afișarea codului HDL generat.

2.9. Verificarea funcționării automatelor cu programul StateBench


2.9.1. Prezentarea programului StateBench

Programul StateBench permite verificarea funcționării automatelor de stare create cu editorul StateCAD și generarea bancurilor de test pentru aceste automate. Acest program realizează simularea funcțională a automatelor pe baza condițiilor inițiale specificate de proiectant și a rezultatelor care trebuie generate. StateBench compară rezultatele specificate ca fiind corecte cu rezultatele obținute la simulare și indică dacă automatul funcționează conform așteptărilor. În cazul obținerii unor rezultate diferite de cele așteptate, programul StateBench simplifică identificarea cauzei problemelor apărute.

Pe baza condițiilor inițiale specificate, a rezultatelor așteptate și, eventual, a unei secvențe de stări care trebuie urmărită, programul StateBench generează un banc de test în limbajul VHDL sau Verilog, care poate fi utilizat pentru simularea funcționării automatului de stare cu majoritatea simulatorilor, de exemplu, cu simulatorul *ModelSim*. Prin adăugarea unor constrângeri de timp, bancul de test generat poate fi utilizat pentru verificarea funcționării automatului după implementarea acestuia într-un circuit programabil. Programul StateBench poate genera un banc de test și în mod automat, aplicând automatului toate combinațiile semnalelor de intrare și încercând parcurgerea tuturor stărilor și tranzițiilor din diagrama de stare.

2.9.2. Lansarea programului StateBench

Programul StateBench poate fi lansat în execuție din fereastra editorului StateCAD, prin exe-

cuția comenzii *Options* → *StateBench* sau selectarea butonului *StateBench* . La lansarea programului StateBench, editorul StateCAD compilează diagrama de stare și verifică eventualele erori. În scopul generării unui banc de test funcțional și valid, se testează dacă în cadrul editorului StateCAD este validată verificarea principalelor erori, cum sunt tranziții nedeterminate sau semnale de ieșire forțate la valori constante. Aceste verificări sunt validate în mod implicit și setările corespunzătoare

sunt afișate în secțiunea *Check For* a ferestrei de dialog *Configuration*. Dacă se detectează invalidarea verificării acestor erori, se afișează o fereastră de dialog pentru confirmarea validării acestor verificări.

Dacă sunt detectate erori la compilarea diagramei, se afișează fereastra *View Error* și compilarea este oprită. Dacă nu sunt detectate erori, se deschide fereastra programului StateBench (Figura 9.18). Această fereastră se afișează în jumătatea de jos a ecranului, diagrama de stare fiind afișată în jumătatea de sus a ecranului. În fereastra programului StateBench sunt afișate toate semnalele utilizate în cadrul automatului de stare, inițializate în mod implicit cu valoarea 0. Semnalele de intrare sunt afișate cu culoarea albastră, iar cele de ieșire sunt afișate cu culoarea maro. Culoarele cu care se afișează semnalele și ordinea de afișare a acestora pot fi modificate prin comanda *Options* → *Signal Editor*.

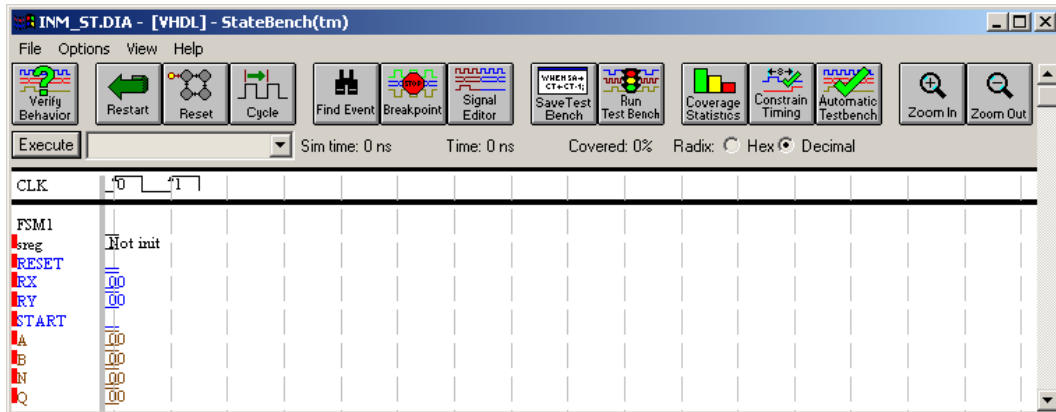



Figura 9.18. Fereastra programului StateBench.

2.9.3. Verificarea funcționării cu comanda *Verify Behavior*

Metoda cea mai simplă de verificare a funcționării automatului de stare constă în utilizarea comenzii *Verify Behavior*. Această comandă permite setarea condițiilor inițiale pentru automatul de stare, alegerea stării până la care să se execute simularea și setarea valorilor așteptate ale semnalelor în această stare. Programul StateBench va executa apoi simularea funcționării automatului, urmărind tranzițiile între stări pe baza condițiilor inițiale, și va afișa diagrama de timp (forma de undă) a semnalelor. Dacă se ajunge în starea selectată, se poate selecta o altă stare sau se poate opri simularea. Este posibil ca programul StateBench să nu poată dirija funcționarea automatului până în starea selectată, caz în care simularea poate fi continuată prin alte comenzi.

Pentru verificarea funcționării automatului de stare utilizând comanda *Verify Behavior*, se procedează astfel:

1. Se selectează comanda *Options* → *Verify Behavior* sau butonul *Verify Behavior* . Se va deschide fereastra de dialog *Verify Behavior* (Figura 9.19).
2. În secțiunea *Set initial conditions* se setează valorile inițiale ale semnalelor. Pentru aceasta, se execută un clic dublu pe numele unui semnal care trebuie inițializat. Dacă semnalul este de tip bit, valoarea acestuia va fi modificată (din 0 în 1 sau invers). Dacă semnalul este un vector, se va afișa fereastra *Assign Value*, care permite introducerea valorii vectorului în câmpul *Value*. Valoarea poate fi introdusă în zecimal sau hexazecimal. Se selectează butonul *OK*, după care se inițializează în mod similar celelalte semnale de intrare ale automatului.
3. În secțiunea *Choose the target state* se selectează starea până la care să se execute simularea (starea destinație).

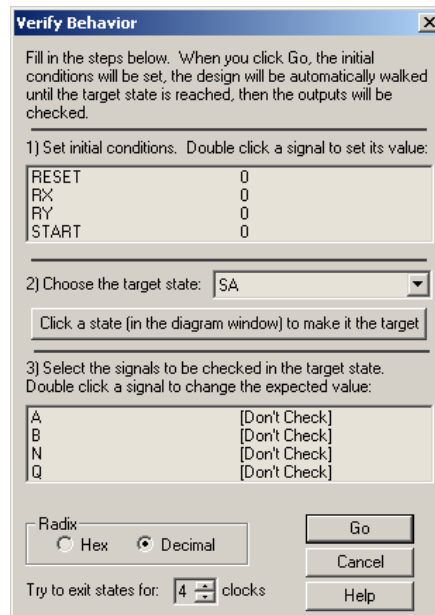




Figura 9.19. Fereastra de dialog *Verify Behavior*.

4. În mod opțional, se setează valorile așteptate ale semnalelor de ieșire în starea destinație. Pentru aceasta, în secțiunea *Select the signals ...* se execută un clic dublu pe numele semnalului care trebuie verificat dacă se ajunge în starea destinație. Se va afișa fereastra *Check Value*, care permite introducerea valorii semnalului în câmpul *Value*. Pentru a indica starea de înaltă impedanță a unui semnal, în câmpul *Value* se introduce caracterul Z. Se selectează butonul *OK*, după care se setează în mod similar valorile celorlalte semnale de ieșire ale automatului. Valorile așteptate ale semnalelor de ieșire vor fi salvate în bancul de test generat și vor fi indicate pe forma de undă a semnalelor. Dacă valoarea unui semnal obținută la simulare este aceeași cu valoarea așteptată, aceasta din urmă este indicată prin culoarea verde, iar dacă există diferențe valoarea așteptată este indicată prin culoarea roșie. De asemenea, dacă există diferențe între valorile semnalelor, se afișează un mesaj și zona corespunzătoare de pe diagrama de timp a semnalelor va fi marcată prin culoarea neagră.
5. Se selectează butonul *Go*. Programul *StateBench* va executa simularea cu valorile inițiale specificate ale semnalelor și va încerca dirijarea automatului în starea destinație indicată. Dacă se ajunge în această stare, programul va indica eventualele diferențe între valorile așteptate ale semnalelor și cele obținute la simulare. Stările care au fost parcurse vor fi indicate pe diagrama de stare prin culoarea verde, iar starea în care s-a ajuns va fi indicată prin culoarea galbenă. Se va afișa fereastra de dialog *Reached Target State*, care permite selectarea unei alte stări destinație sau oprirea simulării. Dacă nu a fost posibilă dirijarea automatului în starea destinație, se va afișa un mesaj pe ecran și simularea va fi oprită.


2.9.4. Alte comenzi pentru verificarea funcționării

Pe lângă comanda *Verify Behavior*, există mai multe comenzi utile care pot fi utilizate pentru verificarea funcționării automatului de stare. Principalele comenzi sunt descrise în continuare.

Comanda *Reset*, activată prin selectarea butonului *Reset* , execută un ciclu pentru resetarea automatului de stare. Dacă există un semnal de resetare sincronă sau asincronă, comanda activează acest semnal pe durata unei perioade de ceas, după care dezactivează semnalul de resetare. Dacă există o condiție mai complexă de resetare, comanda activează în mod automat semnalele necesare pentru resetarea automatului de stare. Programul *StateBench* trasează în mod automat diagrama de timp a semnalelor în ciclul de resetare al automatului și pune în evidență prin culoarea galbenă starea în care se realizează tranziția la resetarea automatului.

Comanda *Cycle*, activată prin selectarea butonului *Cycle* , simulează funcționarea automatului pe durata unei perioade de ceas și trasează diagrama de timp a semnalelor. Această comandă poate fi utilizată, de exemplu, pentru continuarea simulării după execuția comenzii *Verify Behavior*. Diagrama de timp a semnalelor este trasată pe măsura execuției simulării, iar starea curentă este indicată prin culoarea galbenă pe diagrama de stare. Pentru execuția simulării pe durata mai multor perioade de ceas, se poate introduce comanda `cycle n` în linia de comandă, unde n este numărul perioadelor de ceas. Comanda se introduce în caseta din dreapta butonului *Execute*, după care se apasă tasta **Enter** sau se execută un clic pe butonul *Execute*.

Execuția poate fi simulată pentru un timp specificat prin utilizarea comenzii `sim n`, unde n este numărul unităților de timp (implicit, în ns). Comanda este disponibilă numai din linia de comandă. Dacă timpul specificat este mai mic decât o perioadă de ceas, simularea se execută pe durata unei perioade de ceas. Unitatea de timp, durata perioadei de ceas și constrângerile de timp pot fi setate prin comanda *Options* → *Timing Constraints*.


Comanda *Restart*, activată prin selectarea butonului *Restart* , inițializează simularea automatului de stare. Această comandă este executată automat la lansarea programului StateBench. Comanda inițializează toate semnalele (implicit, cu valoarea 0), setează timpul de simulare la 0 și aduce automatul într-o stare necunoscută (registrul de stare nu este inițializat). După execuția acestei comenzi, trebuie executată comanda *Restart* dacă simularea va fi executată cu comanda *Cycle* sau comanda `sim`.

2.9.5. Salvarea bancului de test

În urma simulării funcționale a automatului de stare, programul StateBench creează un banc de test în limbajul VHDL sau Verilog. Acest banc de test generează semnalul de ceas pentru automatul de stare, inițializează semnalele de intrare cu valorile specificate și compară valorile semnalelor de ieșire obținute la simulare cu cele așteptate (dacă acestea au fost specificate de proiectant). Bancul de test creat poate fi utilizat împreună cu descrierea automatului de stare creată de programul StateCAD pentru simularea funcțională a automatului cu un alt simulator sau pentru simularea cu constrângeri de timp după implementare.

La salvarea unui banc de test, se poate salva și un fișier (“*regression file*”) conținând valorile semnalelor de intrare și de ieșire obținute la simularea curentă. Acest fișier poate fi utilizat ulterior cu programul StateBench pentru execuția simulării în aceleași condiții, fără a fi necesară specificarea din nou a condițiilor inițiale și a stărilor care trebuie parcurse.

Pentru salvarea bancului de test conținând rezultatele simulării curente, se execută comanda

File → *Save Test Bench* sau se selectează butonul *Save Test Bench* . Se va deschide fereastra de dialog *Save Test Bench/Regression* (Figura 9.20). În această fereastră se pot modifica numele implicite ale celor două fișiere care vor fi salvate și se pot selecta unele opțiuni.

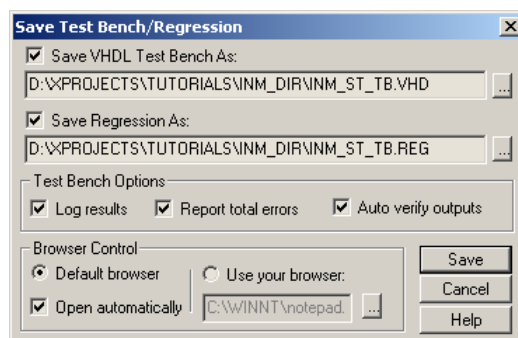


Figura 9.20. Fereastra de dialog *Save Test Bench/Regression*.

Dacă opțiunea *Log results* este validată, la simularea automatului cu un alt simulator mesajele vor fi scrise într-un fișier care poate fi vizualizat cu un editor de texte. Dacă opțiunea *Auto verify*


outputs este validată, în bancul de test generat vor fi inserate comenzi de verificare pentru toate semnalele de ieșire în fiecare ciclu de ceas, pe baza valorilor care au fost obținute la simularea curentă. Această opțiune este utilă dacă bancul de test va fi utilizat pentru verificarea funcționării după sinteza automatului. Pentru salvarea fișierelor se selectează butonul *Save*. Dacă opțiunea *Open automatically* este validată, bancul de test va fi vizualizat automat cu editorul HDL al programului StateCAD sau cu un alt editor selectat în fereastra *Save Test Bench/Regression*.

2.9.6. Modificarea valorii semnalelor și reexecutarea simulării

După execuția simulării cu anumite condiții inițiale și parcurgerea unei anumite succesiuni a stărilor, se pot modifica valorile unor semnale pentru a se verifica funcționarea cu alte condiții inițiale sau pentru alte valori ale semnalelor de intrare.

Metoda cea mai simplă pentru modificarea valorii unui semnal este prin executarea unui clic dublu pe forma de undă a semnalului, în ciclul de ceas de la care se dorește modificarea valorii semnalului. Dacă semnalul este de tip bit, valoarea acestuia va fi modificată din 0 în 1 sau invers. Dacă semnalul este un vector, se va deschide fereastra de dialog *Assign Value*. Se introduce valoarea vectorului în câmpul *Value*, după care se selectează butonul *OK*. În ambele cazuri (semnal de tip bit sau vector), înainte de modificarea valorii semnalului se afișează o fereastră de dialog. În această fereastră, se selectează butonul *Yes* dacă se dorește modificarea valorii semnalului numai în ciclul de ceas curent, sau butonul *No* dacă se dorește modificarea valorii în toate ciclurile de ceas începând cu ciclul curent. Valorile semnalelor de intrare vor fi modificate pe diagrama de timp, dar semnalele de ieșire nu vor fi actualizate decât după executarea din nou a simulării.

Pe lângă modificarea valorii unor semnale de intrare, poate fi necesară și modificarea valorii așteptate a semnalelor de ieșire. Pentru a introduce valoarea așteptată a unui semnal de ieșire în scopul comparării acestei valori cu valoarea obținută la simulare, se execută un clic dublu pe forma de undă a semnalului în starea în care trebuie verificată valoarea semnalului. Se va deschide fereastra de dialog *Assertions*, în care se selectează opțiunea *Add* pentru adăugarea valorii pe diagramă și se selectează butonul *OK*. Se va deschide fereastra de dialog *Assign Value*, în care se poate introduce valoarea dorită. Dacă în ciclul de ceas respectiv a fost setată anterior o valoare, aceasta poate fi modificată în fereastra de dialog *Assertions* prin selectarea opțiunii *Modify the expected value*, sau poate fi ștersă prin selectarea opțiunii *Delete*. Pentru ștergerea valorilor așteptate ale tuturor semnalelor, se poate executa comanda *Options* → *Clear Assertions*.

După modificarea valorii semnalelor, se poate reexecuta simularea. Pentru aceasta, se execută comanda *File* → *Run Test Bench* sau se selectează butonul *Run Test Bench* . Se va afișa o fereastră de dialog, în care se selectează butonul *Yes* pentru reexecutarea simulării curente cu valorile modificate ale semnalelor. Dacă se selectează butonul *No*, se poate executa o simulare salvată anterior într-un fișier.

2.10. Exemplu de proiectare

În continuare se descriu etapele care trebuie parcurse pentru proiectarea unui circuit pentru înmulțirea a două numere fără semn de câte 4 biți prin metoda directă. Atât unitatea de comandă a circuitului de înmulțire, cât și partea de prelucrare vor fi proiectate cu ajutorul editorului pentru diagrame de stare StateCAD.

2.10.1. Descrierea circuitului proiectat

Schema bloc a circuitului de înmulțire este prezentată în Figura 9.21. Registrele *RX* și *RY* se utilizează pentru comunicația cu circuitul de înmulțire. În aceste registre se încarcă deînmulțitul *X*, respectiv înmulțitorul *Y* înaintea începerii operației de înmulțire. Registrul *B* păstrează deînmulțitul în timpul operației de înmulțire. În registrul *Q* se încarcă înmulțitorul la începutul operației, iar acest registru va conține biții cei mai puțin semnificativi ai produsului la sfârșitul operației. Registrul *A* este

un registru acumulator, care este inițializat cu 0 la începutul operației. Acest registru păstrează produsul parțial în timpul operației și va conține biții cei mai semnificativi ai produsului la sfârșitul operației. Sumatorul de 4 biți se utilizează pentru adunarea repetată a deînmulțitului din registrul B la produsul parțial din registrul A . Unitatea de comandă conține un numărator N , care este inițializat cu 4 la începutul operației și este decrementat cu 1 în fiecare pas al operației. Atunci când număratorul N ajunge la valoarea 0, operația de înmulțire este terminată. Semnalul $START$ trebuie activat pentru a iniția operația de înmulțire. Semnalul $TERM$ este activat de circuitul de înmulțire pentru a indica terminarea operației.

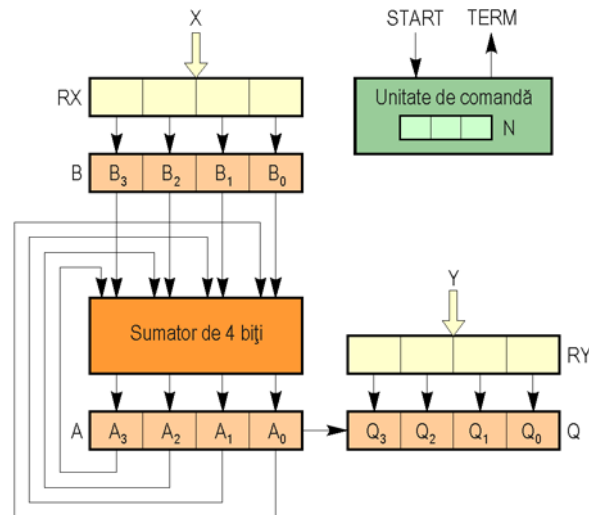


Figura 9.21. Schema bloc a circuitului de înmulțire prin metoda directă.

Înainte începerii operației de înmulțire, se încarcă deînmulțitul X în registrul RX și înmulțitorul Y în registrul RY . Pentru începerea operației, se activează semnalul $START$. Ca urmare, circuitul de înmulțire efectuează următoarele operații de inițializare: transferă conținutul registrului RX în registrul B , transferă conținutul registrului RY în registrul Q , inițializează registrul A cu 0 și inițializează număratorul N cu 4. În fiecare pas al operației, se testează bitul c.m.p.s. al înmulțitorului (Q_0), iar dacă acest bit este 1, se adună deînmulțitul (aflat în registrul B) la produsul parțial (păstrat în registrul A). În continuare, se deplasează registrele A și Q la dreapta cu o poziție, iar bitul c.m.p.s. al registrului A se transferă în poziția c.m.s. a registrului Q . Se decrementează număratorul N , iar dacă acesta nu este 0, se continuă operația cu testarea bitului c.m.p.s. al înmulțitorului. La sfârșitul operației, se activează semnalul $TERM$, care indică faptul că registrele A și Q conțin rezultatul înmulțirii.

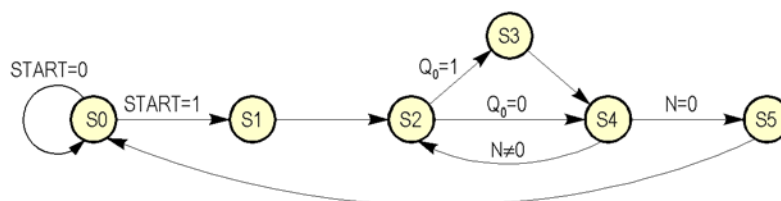


Figura 9.22. Diagrama de stare a circuitului de înmulțire prin metoda directă.

Circuitul de înmulțire poate fi considerat un automat de stare, astfel încât funcționarea acestuia poate fi descrisă printr-o diagramă de stare. Diagrama de stare a circuitului este ilustrată în Figura 9.22. Această diagramă conține 6 stări distincte. Starea S_0 este o stare de repaus, în care circuitul așteaptă activarea semnalului $START$ pentru începerea operației de înmulțire. Dacă semnalul $START$ devine 1, se trece în starea S_1 , în care se efectuează operațiile de inițializare, care pot fi descrise în mod simbolic astfel:

$$B \leftarrow RX; Q \leftarrow RY; A \leftarrow 0; N \leftarrow 4;$$

În starea S2 se decrementează numărătorul N . Decrementarea se efectuează înaintea fiecărui pas al operației de înmulțire și nu după fiecare pas, așa cum s-a descris mai sus, pentru simplificarea diagramei de stare. Dacă bitul Q_0 este 1, se trece în starea S3, în care se adună deînmulțitul la produsul parțial:

$$A \leftarrow A + B;$$

Din starea S3 se trece necondiționat în starea S4, în care se deplasează la dreapta registrele A și Q cu o poziție. În starea S4 se trece și din starea S2, dacă bitul Q_0 este 0. Din starea S4 se trece în starea S2 dacă numărătorul N nu este 0, sau în starea S5 dacă numărătorul N este 0. În starea S5 se activează semnalul $TERM$ care indică terminarea operației, după care se trece în starea de repaus S0.

2.10.2. Crearea proiectului

Pentru crearea unui nou proiect, se lansează în execuție programul Xilinx WebPACK, după care se execută următoarele etape:

1. În ecranul *Project Navigator* se selectează *File* → *New Project*. Se va deschide fereastra de dialog *New Project*.
2. În câmpul *Project Location* se selectează subdirectorul în care se va crea proiectul (un subdirector din directorul C:\Student\).
3. În câmpul *Project Name* se introduce numele proiectului, de exemplu, **lab9_1**.
4. Se verifică setările corecte în câmpurile *Device Family* și *Device* (**Spartan2**, respectiv **xc2s50-5tq144**).
5. În câmpul *Design Flow* se selectează fluxul de proiectare: **XST VHDL**.
6. Se selectează butonul *OK* pentru a crea proiectul.

2.10.3. Lansarea în execuție a programului StateCAD

Pentru lansarea în execuție a programului StateCAD și crearea unui fișier pentru diagrama de stare, se procedează astfel:

1. Se selectează comanda *Project* → *New Source*.
2. În fereastra de dialog *New*, se selectează opțiunea *State Diagram*.
3. În câmpul *File Name* se introduce numele fișierului în care se va păstra diagrama de stare, de exemplu, **inm_dir**.
4. Se selectează butonul *Next*.
5. În fereastra de dialog *New Source Information*, se selectează butonul *Finish*.

Se va lansa în execuție programul StateCAD într-o nouă fereastră.

2.10.4. Crearea automatului de stare

Pentru crearea automatului de stare se va folosi utilitarul *State Machine Wizard*.



1. Se selectează butonul *Draw State Machines*. Se va deschide fereastra *State Machine Wizard*.
2. În câmpul *Shape of state machine* se selectează opțiunea *Row* pentru plasarea stărilor automatului pe linie.

3. În câmpul *Number of States* se selectează 5 ca număr de stări ale automatului. Ulterior, se va introduce o stare suplimentară.
4. Se selectează butonul *Next*. Se va deschide fereastra de dialog *Reset The State Machine*.
5. În câmpul *Reset Mode* se selectează opțiunea *Asynchronous* pentru a se genera o logică asincronă de resetare a automatului de stare.
6. Se selectează butonul *Next*. Se va deschide fereastra de dialog *Setup Transitions*.
7. În câmpul *Add Transitions* se selectează tipul tranzițiilor care vor fi prezente în diagrama de stare. În mod implicit, va fi selectată doar opțiunea *Next*, opțiunile *Loop back* și *Previous* fiind neselectate. Astfel, diagrama va conține doar tranziții de la fiecare stare la starea următoare a acesteia. Se păstrează aceste setări.
8. Se selectează butonul *Finish*. În fereastra editorului StateCAD se va afișa un dreptunghi cu conturul de culoare verde, care indică marginile diagramei.
9. Se deplasează dreptunghiul în zona centrală a ferestrei și se execută un clic cu butonul din stânga. Se va afișa diagrama de stare cu forma ilustrată în Figura 9.23.

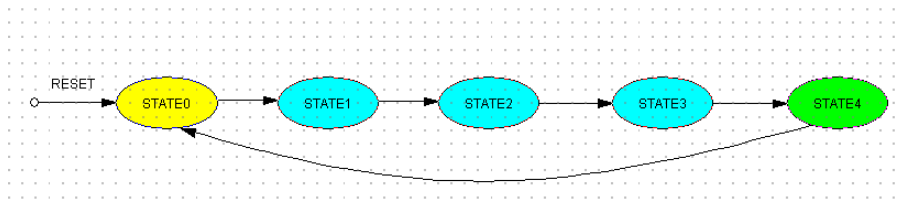




Figura 9.23. Forma inițială a diagramei de stare creată cu utilitarul *State Machine Wizard*.

2.10.5. Adăugarea unei stări suplimentare

Pentru adăugarea unei stări suplimentare la diagrama creată anterior, se procedează astfel:

1. Se selectează butonul *Draw States* , aflat în partea stângă a ecranului.
2. Se deplasează cursorul deasupra stării STATE2, după care se execută un clic cu butonul din stânga. Simbolul stării va fi plasat pe diagramă. StateCAD va atribui numele STATE5 noii stări.
3. Se selectează butonul *Select Objects* .

2.10.6. Editarea stărilor

În continuare se modifică numele stărilor astfel încât acestea să reflecte funcția lor și se specifică semnalele de ieșire care trebuie activate în stările respective. Pentru aceasta se execută următoarele operații:

1. Se execută un clic dublu pe simbolul stării STATE0. Se va afișa fereastra de dialog *Edit State*.
2. În câmpul *State Name*, se introduce noul nume al stării, **repaus**, după care se selectează butonul *OK*.
3. Se execută un clic dublu pe simbolul stării STATE1. În câmpul *State Name* se introduce numele **init**. În câmpul *Outputs* se introduc următoarele ecuații indicând valorile cu care se inițializează registrele și semnalul *TERM*:

B=RX;Q=RY;<Enter>

A=0;N=4;<Enter>

TERM='0';

Se selectează apoi butonul *OK*.

4. Se execută un clic dublu pe simbolul stării STATE2. În câmpul *State Name* se introduce numele **decrem_N** și se selectează butonul *OK*.
5. Se execută un clic dublu pe simbolul stării STATE3. În câmpul *State Name* se introduce numele **deplas_AQ** și se selectează butonul *OK*.
6. Se execută un clic dublu pe simbolul stării STATE4. În câmpul *State Name* se introduce numele **termin**. În câmpul *Outputs* se introduce valoarea semnalului *TERM*:



TERM='1';

Se selectează apoi butonul *OK*.

7. Se execută un clic dublu pe simbolul stării STATE5 (starea care a fost adăugată separat). În câmpul *State Name* se introduce numele **adun_B** și se selectează butonul *OK*.

2.10.7. Adăugarea unor tranziții suplimentare

Pentru completarea diagramei de stare cu tranzițiile care nu au fost generate în mod automat, se procedează astfel:

1. Se selectează butonul *Draw Transitions* , aflat în partea stângă a ecranului.
2. Se selectează starea **decrem_N** prin execuția unui clic în interiorul stării, iar apoi se execută un nou clic în interiorul stării, în apropierea marginii de sus. Va apare un pătrat de culoare roșie pe marginea simbolului stării, iar un alt pătrat va urmări deplasarea cursorului.
3. Se deplasează cursorul deasupra stării **adun_B**. Va apare un pătrat de culoare roșie pe marginea simbolului stării și o linie între stările **decrem_N** și **adun_B**. Se fixează poziția punctului de capăt al liniei și se execută un clic în poziția respectivă. Dacă o linie de tranziție nu a fost adăugată în mod corespunzător, aceasta va fi ștersă prin apăsarea tastei **Delete**, iar apoi va fi adăugată din nou.
4. În același mod se adaugă o tranziție de la starea **adun_B** la starea **deplas_AQ**.
5. Se procedează similar pentru adăugarea unei tranziții de la starea **deplas_AQ** la starea **decrem_N**. Pentru transformarea liniei de tranziție într-o linie curbă, se execută un clic pe unul din cele două puncte de control din interiorul liniei (puncte indicate prin pătrate cu contururi de culoare roșie), iar apoi se re poziționează punctul prin deplasarea cursorului în jos, menținând butonul apăsat. Se procedează similar cu al doilea punct de control.
6. Se adaugă o tranziție de buclare pentru starea **repaus**. Pentru aceasta, se execută un clic dublu în interiorul simbolului acestei stări, în apropierea marginii de sus. Editorul va adăuga linia pentru tranziția de buclare (care începe și se termină în starea **repaus**) și va adăuga condiția **@ELSE** pentru această tranziție.
7. Se selectează butonul *Select Objects*  pentru a termina adăugarea tranzițiilor.

2.10.8. Adăugarea condițiilor pentru tranziții

Pentru adăugarea condițiilor care determină execuția tranzițiilor, se execută următoarele operații:

1. Se execută un clic dublu pe linia tranziției de la starea **repaus** la starea **init**. Se va deschide fereastra de dialog *Edit Condition*. În câmpul *Condition* se introduce condiția **START** (sau **START='1'**), iar apoi se selectează butonul *OK*. Se deplasează dreptunghiul de culoare roșie deasupra liniei de tranziție.

2. Se execută un clic dublu pe linia tranziției de la starea **decrem_N** la starea **adun_B**. În câmpul *Condition* se introduce condiția $Q0=1$ (sau $Q0$) și se selectează butonul *OK*. Se deplasează dreptunghiul de culoare roșie în stânga liniei de tranziție.
3. Se execută un clic dublu pe linia tranziției de la starea **decrem_N** la starea **deplas_AQ**. În câmpul *Condition* se introduce condiția $Q0=0$ (sau $!Q0$) și se selectează butonul *OK*. Se deplasează dreptunghiul de culoare roșie deasupra liniei de tranziție.
4. Se execută un clic dublu pe linia tranziției de la starea **deplas_AQ** la starea **decrem_N**. În câmpul *Condition* se introduce condiția $N=0$ și se selectează butonul *OK*. Se deplasează dreptunghiul de culoare roșie sub linia de tranziție.
5. Se execută un clic dublu pe linia tranziției de la starea **deplas_AQ** la starea **termin**. În câmpul *Condition* se introduce condiția $N=0$ și se selectează butonul *OK*. Se deplasează dreptunghiul de culoare roșie deasupra liniei de tranziție.

După execuția acestor operații, diagrama de stare va arăta ca în Figura 9.24. Se salvează fișierul cu diagrama de stare prin selecția butonului *Save File*.

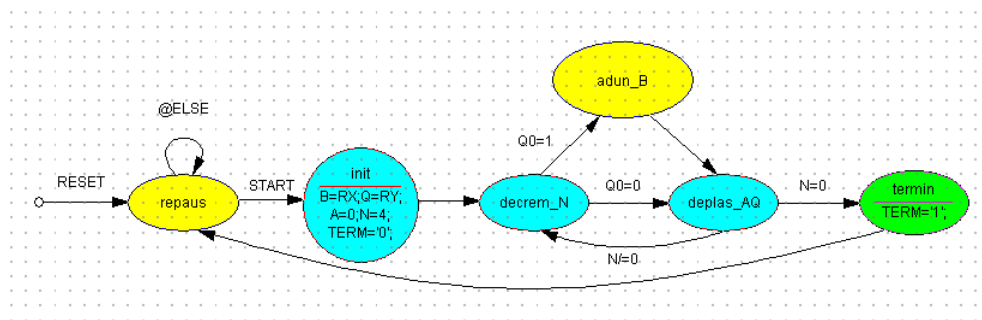


Figura 9.24. Forma diagramei de stare după editarea acesteia.

2.10.9. Adăugarea circuitelor logice

Pentru executarea operațiilor elementare din care se compune operația de înmulțire trebuie adăugate circuitele logice corespunzătoare. Aceste circuite vor fi adăugate în mod automat de editorul StateCAD dacă se specifică ecuațiile ieșirilor care trebuie generate în stările respective. Se vor adăuga următoarele circuite: numărătorul N , care se decrementează în fiecare pas al operației; sumatorul de 4 biți, utilizat pentru adunarea de înmulțitului din registrul B la produsul parțial din registrul A ; registrele A și Q , care se deplasează la dreapta cu o poziție în fiecare pas al operației.

Pentru adăugarea numărătorului N , se execută următoarele operații:

1. Se execută un clic dublu pe simbolul stării **decrem_N**. Se va deschide fereastra de dialog *Edit State*.
2. Se selectează butonul *Output Wizard*, prin care se va deschide fereastra de dialog *Logic Wizard*.
3. În lista afișată în partea din stânga sus a ferestrei se selectează circuitul **Count Down**.
4. În câmpul *COUNT* se introduce N ca nume al semnalului de ieșire al numărătorului.
5. În câmpul *Data path width* se selectează **2** ca dimensiune a vectorului reprezentând semnalul de ieșire al numărătorului.
6. Se verifică selectarea opțiunii *Registered*, prin care semnalul de ieșire va fi secvențial.
7. Se selectează butonul *OK*. Utilitarul *Logic Wizard* generează ecuația logică a numărătorului, definește vectorul care reprezintă semnalul de ieșire al acestuia și va deschide apoi fereastra de dialog *Edit Equation*. În câmpul *Outputs* al acestei ferestre va apare ecuația numărătorului: $N = N-1$.

8. Se selectează butonul *OK*. Ecuația numărătorului va apare sub numele stării **decrem_N**. Sub diagrama de stare va apare simbolul vectorului **N[1:0]**, indicând definirea acestui vector.

Pentru adăugarea sumatorului, se procedează astfel:

1. Se execută un clic dublu pe simbolul stării **adun_B**. Se va deschide fereastra de dialog *Edit State*.
2. Se selectează butonul *Output Wizard*, prin care se va deschide fereastra de dialog *Logic Wizard*.
3. În lista afișată se selectează circuitul **Adder**.
4. În câmpul *A* se introduce **A**, iar în câmpul *B* se introduce **B**. Aceștia sunt cei doi operanzi care vor fi adunați.
5. În câmpul *DEST* se introduce **A** ca destinație a operației de adunare.
6. În câmpul *Data path width* se selectează valoarea **4** ca dimensiune a vectorilor care vor fi adunați.
7. Se selectează butonul *OK*. Utilitarul *Logic Wizard* generează ecuația logică a sumatorului, definește vectorii care vor fi adunați și va deschide fereastra de dialog *Edit Equation*. În câmpul *Outputs* al acestei ferestre va apare ecuația sumatorului: **A = (A+B)**.
8. Se selectează butonul *OK*. Ecuația sumatorului va apare sub numele stării **adun_B**. În fereastra diagramei de stare vor apare simbolurile vectorilor **A[3:0]** și **B[3:0]**, indicând definirea acestor vectori.

Pentru adăugarea registrelor de deplasare *A* și *Q*, se procedează astfel:



1. Se execută un clic dublu pe simbolul stării **deplas_AQ**. Se va deschide fereastra de dialog *Edit State*.
2. Se selectează butonul *Output Wizard*, prin care se va deschide fereastra de dialog *Logic Wizard*.
3. În lista afișată se selectează circuitul **Shift Right**.
4. În câmpul *DIN* se introduce **A3**, iar în câmpul *DOUT* se introduce **A**.
5. În câmpul *Data path width* se setează valoarea **4** ca dimensiune a registrului de deplasare.
6. Se selectează butonul *OK*. Se va deschide fereastra de dialog *Edit Equation*. În câmpul *Outputs* al acestei ferestre va apare ecuația registrului de deplasare: **A = A3:A3..A1**.
7. Se selectează butonul *Output Wizard*, prin care se va deschide din nou fereastra de dialog *Logic Wizard*.
8. Se selectează circuitul **Shift Right**.
9. În câmpul *DIN* se introduce **A0**, iar în câmpul *DOUT* se introduce **Q**.
10. În câmpul *Data path width* se setează valoarea **4** ca dimensiune a registrului de deplasare.
11. Se selectează butonul *OK*. Se va deschide fereastra de dialog *Edit Equation*. În câmpul *Outputs* al acestei ferestre, sub ecuația registrului de deplasare *A*, va apare ecuația registrului de deplasare *Q*: **Q = A0:Q3..Q1**.
12. Se selectează butonul *OK*. Ecuațiile registrelor de deplasare vor apare sub numele stării **deplas_AQ**. În fereastra diagramei de stare va apare simbolul vectorului **Q[3:0]**, indicând definirea acestui vector.

Se salvează fișierul cu diagrama de stare prin selectarea butonului *Save File*.

2.10.10. Adăugarea unor noi vectori

Programul StateCAD definește în mod automat vectorii care au fost utilizați în ecuațiile circuitelor logice adăugate pentru generarea semnalelor de ieșire. Dimensiunea acestor vectori este cunoscută prin selectarea valorii câmpului *Data path width* al ferestrei de dialog *Logic Wizard*. Variabilele de tip vector care nu apar în ecuațiile circuitelor logice adăugate trebuie definite în mod explicit de proiectant, specificând dimensiunea acestora. În caz contrar, aceste variabile vor fi considerate de tip bit.

În cazul circuitului de înmulțire, variabilele de tip vector a căror dimensiune nu a fost specificată sunt *RX* și *RY*. Pentru definirea acestor variabile ca vectori, se procedează astfel:


1. Se selectează butonul *Draw Vectors* , aflat în partea stângă a ecranului.
2. Se deplasează cursorul într-o zonă liberă de sub diagrama de stare și se execută un clic cu butonul din stânga. Va apare simbolul unui vector **VAR0[7:0]**, cu dimensiunea 8.
3. Se plasează un al doilea vector în dreapta primului. Va apare simbolul vectorului **VAR1[7:0]**.
4. Se selectează butonul *Select Objects*  pentru terminarea adăugării vectorilor.
5. Pentru modificarea numelui și dimensiunii vectorului **VAR0**, se execută un clic dublu deasupra simbolului acestuia. Se va deschide fereastra de dialog *Edit Vector*.
6. În câmpul *Name* se modifică numele vectorului în **RX**, iar în câmpul *Range* se modifică domeniul biților în **3:0**. Se selectează apoi butonul *OK*. Se poate afișa o fereastră indicând faptul că variabila *RX* este utilizată. În acest caz, se selectează butonul *Yes* pentru ștergerea variabilei existente *RX* și redenumirea vectorului în *RX*.
7. Se procedează similar pentru modificarea numelui vectorului **VAR1** în **RY** și a domeniului biților în **3:0**.

2.10.11. Vizualizarea opțiunilor de configurație

În continuare se vizualizează opțiunile implicite care vor fi utilizate la compilarea automatului de stare și se modifică una din opțiuni.

1. Se selectează comanda *Options* → *Configuration*, prin care se va deschide fereastra de dialog *Configuration*.
2. În secțiunea *Options* a acestei ferestre se validează opțiunea *Retain Output Values*, prin care semnalele de ieșire care nu sunt specificate în mod explicit într-o stare își vor păstra valorile de la ultima asignare.
3. În secțiunea *State Assignment* se indică modul în care se va realiza asignarea stărilor. Opțiunea selectată în mod implicit este *One Hot*, care indică metoda de asignare cu un bistabil pe stare. Se poate păstra această opțiune, sau se poate selecta opțiunea *Enumerated*.
4. Se închide fereastra prin selectarea butonului *OK*.

2.10.12. Compilarea diagramei de stare și corectarea erorilor

Pentru compilarea diagramei de stare, se selectează butonul *Generate HDL* . Se va deschide fereastra *View Error* în care se afișează un mesaj de eroare. Mesajul afișat indică faptul că valoarea 4 asignată vectorului *N* este mai mare decât domeniul posibil al valorilor vectorului, care este de 2 biți. Pe diagrama de stare, este pusă în evidență starea în care s-a efectuat asignarea; aceasta este starea **init**. Se închide fereastra *View Error* prin selectarea butonului *Close*.

Pentru corectarea acestei erori și recompilarea diagramei de stare, se procedează astfel:

1. Se execută un clic dublu pe simbolul vectorului *N*. Se va deschide fereastra de dialog *Edit Vector*.

2. În câmpul *Range* se modifică domeniul vectorului din 1:0 în **2:0**. Astfel, numărătorul *N* va fi de 3 biți.
3. Se selectează butonul *OK*. Simbolul vectorului *N* va conține noul domeniu
4. Se compilează diagrama de stare prin selectarea butonului *Generate HDL*. Se va afișa o nouă eroare în fereastra *View Error*, indicând o comparație ilegală **Q0 = 1** sau **Q0 = 0**. Valoarea cu care se efectuează comparația trebuie să fie o constantă sau o variabilă booleană. Condiția în care apare comparația ilegală și tranziția corespunzătoare sunt puse în evidență prin culoarea roșie pe diagramă. Se închide fereastra *View Error*.
5. Se execută un clic dublu pe condiția **Q0 = 1**. Se va deschide fereastra de dialog *Edit Condition*.
6. În câmpul *Condition* se corectează condiția existentă **Q0 = 1** în **Q0 = '1'**, după care se selectează butonul *OK*.
7. Se procedează similar pentru modificarea condiției **Q0 = 0** în **Q0 = '0'**.
8. Se salvează diagrama de stare prin selectarea butonului *Save File*.
9. Se compilează din nou diagrama de stare prin selectarea butonului *Generate HDL*.
10. Dacă se afișează fereastra de dialog *Delete Unused Variables*, se selectează butonul *Delete* pentru ștergerea variabilelor neutilizate.

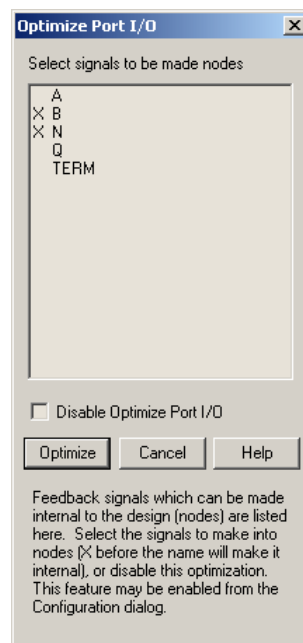






Figura 9.25. Fereastra de dialog *Optimize Port I/O* pentru selectarea semnalelor care vor deveni noduri.

11. Se va deschide fereastra de dialog *Optimize Port I/O*. În această fereastră se pot selecta semnalele care vor deveni noduri (semnale interne) și cele care vor deveni porturi de I/E. Semnalele interne permit efectuarea unor optimizări. Se execută un clic pe numele semnalelor *A*, *Q* și *TERM* pentru a elimina marcajul X din fața numelui acestora (Figura 9.25). Astfel, semnalele *A*, *Q* și *TERM* vor deveni porturi de I/E, iar *B* și *N* vor deveni noduri.
12. Se selectează butonul *Optimize*. Se va afișa fereastra *Results* indicând compilarea fără erori. Sunt afișate de asemenea diferite statistici despre automatul de stare generat și operațiile efectuate la compilare.
13. Se închide fereastra *Results*. Se va deschide fereastra *StateCAD HDL Browser* în care se afișează fișierul VHDL generat prin compilarea diagramei de stare. Se vizualizează codul generat, după care se închide această fereastră.


2.10.13. Verificarea funcționării cu programul StateBench

Pentru verificarea funcționării circuitului de înmulțire, se va lansa în execuție programul StateBench și se va utiliza mai întâi comanda *Verify Behavior*, iar apoi comanda *Cycle*. Pentru aceasta, se execută următoarele operații:

1. Se lansează în execuție programul StateBench prin selectarea butonului *StateBench* . În jumătatea de jos a ecranului se va afișa fereastra programului StateBench, diagrama de stare fiind afișată în jumătatea de sus a ecranului.
2. În fereastra programului StateBench se selectează butonul *Verify Behavior* . Se va deschide fereastra de dialog *Verify Behavior*.
3. În secțiunea *Set initial conditions* se setează valorile inițiale ale semnalelor. Mai întâi, se setează valoarea de înmulțitului din registrul *RX* prin execuția unui clic dublu pe numele semnalului *RX*. Se va deschide fereastra de dialog *Assign Value*, în care se introduce valoarea 6, iar apoi se selectează butonul *OK*. Se procedează similar pentru setarea valorii înmulțitorului din registrul *RY* la 5. Se execută apoi un clic dublu pe numele semnalului *START* pentru setarea acestui semnal la 1 logic.
4. În secțiunea *Choose the target state* se selectează starea **decrem_N**. Simularea va fi executată până când se ajunge în această stare.
5. Se selectează butonul *Go*. Programul StateBench va executa simularea cu valorile inițiale specificate ale semnalelor până când se ajunge în starea **decrem_N**. Stările care au fost parcurse vor fi indicate pe diagrama de stare prin culoarea verde, iar starea în care s-a ajuns va fi indicată prin culoarea galbenă. În fereastra cu diagramele de timp ale semnalelor se trasează forma acestora în ciclurile de ceas 0, 1 și 2. Se va afișa apoi fereastra de dialog *Reached Target State*. Aceasta indică faptul că s-a ajuns în starea specificată, fiind posibilă selectarea unei alte stări destinație sau oprirea simulării. Se selectează butonul *No* pentru închiderea ferestrei și oprirea simulării prin comanda *Verify Behavior*.
6. Se selectează butonul *Signal Editor* , prin care se va deschide fereastra de dialog cu același nume. În secțiunea *List in simulation* se selectează semnalul *Q*, iar apoi se selectează butonul *Up* de două ori pentru ca semnalul *Q* să fie afișat imediat sub semnalul *A* în fereastra diagramelor de timp. Astfel va fi mai simplă urmărirea rezultatului înmulțirii, rezultat care se formează în registrele *A* și *Q*. Se selectează butonul *OK*, iar în fereastra *Signal Editor* se selectează butonul *No*.
7. Se continuă simularea prin comanda *Cycle*, activată prin selectarea butonului *Cycle* . Prin această comandă, execuția va fi simulată pe durata unei perioade de ceas, fiind posibilă urmărirea parcurgerii stărilor pe diagrama de stare și a valorii semnalelor pe diagrama de timp. Se continuă simularea prin această comandă până când se ajunge în starea **termin**.
8. Se verifică valorile semnalelor (registrelor) *A* și *Q* în starea **termin**. Registrul *A* conține valoarea 1, iar *Q* conține valoarea 14 (în zecimal). Deoarece registrul *A* conține partea mai semnificativă a produsului, *Q* conține partea mai puțin semnificativă, iar registrele sunt de câte 4 biți, valoarea produsului este $1 \times 16 + 14 = 30$. Rezultatul obținut este corect ($6 \times 5 = 30$).
9. În continuare se vor modifica valorile inițiale ale registrelor *RX* și *RY* pentru a se verifica funcționarea și pentru alte valori ale de înmulțitului și înmulțitorului. Pentru aceasta, se execută un clic dublu pe valoarea 06 a semnalului *RX* de pe diagrama de timp, iar în fereastra *Assign Value* se introduce valoarea 9 și se selectează butonul *OK*. Se va afișa o fereastră de dialog, în care se selectează butonul *No* pentru modificarea valorii semnalului în toate ciclurile de ceas. Se procedează similar pentru modificarea valorii semnalului *RY* din 05 în 11.
10. Se introduc valorile așteptate (cele considerate corecte) ale semnalelor *A* și *Q* la terminarea operației (în starea **termin**), în scopul comparării acestor valori cu valorile obținute la simula-

re. Produsul așteptat este $9 \times 11 = 99$, care se poate scrie ca $6 \times 16 + 3$. Deci, valoarea așteptată a semnalului A este 6, iar cea a semnalului Q este 3. Pentru setarea acestor valori, se execută mai întâi un clic dublu pe forma de undă a semnalului A în starea **termin**. Se va deschide fereastra de dialog *Assertions*, în care se selectează opțiunea *Add* pentru adăugarea valorii pe diagramă, iar apoi se selectează butonul *OK*. Se va deschide fereastra de dialog *Assign Value*, în care se introduce valoarea 6 și se selectează butonul *OK*. Valoarea introdusă va fi marcată pe diagrama de timp prin culoarea roșie. Se procedează similar pentru introducerea valorii așteptate a semnalului Q (valoarea 3).

11. În continuare se va reexecuta simularea cu noile valori ale semnalelor. Pentru aceasta, se se-

lectează butonul *Run Test Bench* . Se va afișa o fereastră de dialog, în care se selectează butonul *Yes* pentru reexecutarea simulării curente cu valorile modificate ale semnalelor. Programul StateBench va reexecuta simularea până în starea **termin**, verificând valorile semnalelor A și Q . Se va afișa un mesaj de avertizare indicând faptul că au fost detectate diferențe între valorile semnalelor (“*Warning: Failing assertions found...*”). Se închide fereastra mesajului de avertizare. Zona în care au fost detectate diferențe între valorile semnalelor este marcată pe diagrama de timp (Figura 9.26). Se execută un clic pe diagrama de timp, observând că valoarea incorectă (indicată prin culoarea roșie) este cea a semnalului A .

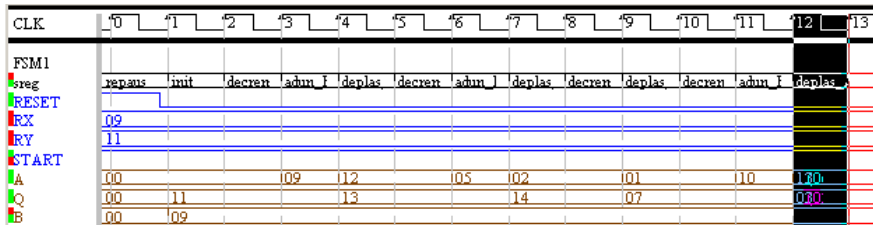



Figura 9.26. Diagramă de timp indicând diferențe între valorile așteptate ale semnalelor și cele obținute la simulare.

2.10.14. Detectarea și corectarea erorii

Pentru detectarea erorii existente, se va utiliza comanda *Find Event*. La execuția acestei comenzi, programul StateBench parcurge diagrama de timp și permite oprirea la o anumită stare, la un anumit timp de simulare, la detectarea unei diferențe între valorile așteptate și cele obținute ale semnalelor, sau atunci când o expresie este adevărată. La detectarea unei condiții specificate, programul StateBench permite identificarea sursei semnalului și editarea acesteia direct pe diagrama de stare.

Pentru a încerca detectarea erorii, se execută următoarele operații:

1. Se execută comanda *Find Event* prin selectarea butonului *Find Event* . Se va deschide fereastra de dialog *Find Event*.
2. În această fereastră, se selectează opțiunea *Find driver for output*. În caseta alăturată se selectează semnalul A pentru care s-a detectat diferența între valoarea așteptată și cea obținută (acest semnal poate fi selectat în mod automat). Se selectează apoi butonul *Find Next*. Se va afișa fereastra de dialog *Find*, în care se indică faptul că semnalul A este setat în starea **deplas_AQ**. Se selectează butonul *Yes* pentru editarea acestei stări pe diagrama de stare.
3. În fereastra *Edit State* se verifică ecuația registrului de deplasare A : $A = A3:A3..A1$. Această ecuație indică faptul că, la deplasarea la dreapta a registrului, bitul cel mai semnificativ nu este modificat. În mod normal, la deplasarea la dreapta a unui număr fără semn trebuie să se introducă valoarea 0 în poziția cea mai semnificativă.
4. Se corectează ecuația registrului de deplasare, astfel: $A = 0:A3..A1$. Se selectează apoi butonul *OK*.
5. În mod automat, programul StateBench va executa din nou simularea curentă, verificând valorile semnalelor. Se va afișa un mesaj care indică trecerea cu succes a testelor efectuate (“*All*

assertions pass.”). Se închide fereastra acestui mesaj. Se observă afișarea valorilor așteptate ale semnalelor cu culoarea verde.

6. Se elimină valorile așteptate ale semnalelor prin comanda *Options* → *Clear Assertions*. Se introduc alte valori în registrele *RX* și *RY*, se setează valorile așteptate ale semnalelor *A* și *Q*, după care se execută din nou simularea.



7. Se salvează bancul de test prin selectarea butonului *Save Test Bench*. Se va deschide fereastra de dialog *Save Test Bench/Regression*. În această fereastră se păstrează numele implicite ale fișierelor și se selectează butonul *Save*. Bancul de test generat în limbajul VHDL va fi afișat în fereastra *StateCAD HDL Browser*. Se vizualizează codul generat, după care se închide fereastra.

3. Desfășurarea lucrării

3.1. Executați etapele descrise în secțiunea 2.10 pentru proiectarea circuitului de înmulțire prin metoda directă.

3.2. Refaceți exemplul de proiectare descris în secțiunea 2.10 pentru un circuit de înmulțire a două numere cu semn reprezentate în complement față de 2 prin metoda Booth. Numerele sunt de câte 5 biți, din care unul este bitul de semn. Desenați mai întâi diagrama de stare a circuitului pe hârtie, după care parcurgeți aceleași etape ca și în exemplul de proiectare descris în lucrare.

3.3. Proiectați circuitul de înmulțire prin metoda directă utilizând o schemă pentru partea de prelucrare și o diagramă de stare pentru unitatea de comandă. Intrările circuitului sunt următoarele:

- *CLK*: Semnal de ceas;
- *RST*: Semnal de resetare asincronă;
- *START*: Semnal pentru începerea operației de înmulțire;
- *X*: Vector de 4 biți care conține de înmulțitul;
- *Y*: Vector de 4 biți care conține înmulțitorul.

Ieșirile circuitului sunt următoarele:

- *A*: Vector de 4 biți care va conține partea c.m.s. a produsului;
- *Q*: Vector de 4 biți care va conține partea c.m.p.s. a produsului;
- *TERM*: Semnal care indică prin valoarea logică 1 terminarea operației.

Pentru proiectarea circuitului de înmulțire, utilizați componentele care sunt descrise pe scurt în continuare.

- **FD4CE**: *Registru de 4 biți cu resetare asincronă*

Intrările de date sunt *D0..D3*, iar ieșirile de date sunt *Q0..Q3*. Dacă semnalul *CLR* (*Clear*) este 1, registru este resetat în mod asincron, ieșirile sale devenind 0. Dacă semnalul *CE* (*Clock Enable*) este 1, la următorul impuls de ceas se realizează încărcarea paralelă a registrului cu datele aplicate la intrare. Dacă semnalul *CE* este 0, ieșirile registrului rămân neschimbate.

- **SR4CLED**: *Registru de deplasare bidirecțional de 4 biți cu resetare asincronă*

Intrările paralele de date sunt *D0..D3*, iar intrările seriale de date sunt *SLI* (utilizată în cazul deplasării la stânga) și *SRI* (utilizată în cazul deplasării la dreapta). Ieșirile de date sunt *Q0..Q3*. Dacă semnalul *CLR* (*Clear*) este 1, registru este resetat în mod asincron, ieșirile sale devenind 0. Dacă semnalul *L* (*Load*) este 1, la următorul impuls de ceas se realizează încărcarea paralelă a registrului cu datele aplicate la intrare, indiferent de starea semnalului *CE*. Dacă semnalul *CE* (*Clock Enable*) este 1, la fiecare impuls de ceas se realizează deplasarea registrului cu o poziție în direcția indicată de semnalul *LEFT*. Dacă semnalul *LEFT* este 0, deplasarea se realizează la dreapta, iar dacă semnalul *LEFT* este 1, deplasarea se realizează la stânga. Dacă semnalul *CE* este 0, ieșirile registrului rămân neschimbate.

- **ADD4:** *Sumator de 4 biți*

Intrările sunt cei doi operanzi care trebuie adunați, $A0..A3$, respectiv $B0..B3$, și transportul de intrare CI (*Carry In*). Ieșirile sunt suma $S0..S3$, transportul de ieșire CO (*Carry Out*) și semnalul de depășire OFL (*Overflow*). Acesta din urmă se utilizează numai la adunarea numerelor cu semn.

- **CB4CLED:** *Numărător bidirecțional de 4 biți cu încărcare paralelă și resetare asincronă*

Intrările de date sunt $D0..D3$, iar ieșirile de date sunt $Q0..Q3$. Dacă semnalul CLR (*Clear*) este 1, numărătorul este resetat în mod asincron, ieșirile sale devenind 0. Dacă semnalul L (*Load*) este 1, la următorul impuls de ceas se realizează încărcarea paralelă a numărătorului cu datele aplicate la intrare, indiferent de starea semnalului CE . Dacă semnalul CE (*Clock Enable*) este 1, la fiecare impuls de ceas se realizează incrementarea sau decrementarea numărătorului cu 1, în funcție de semnalul UP . Dacă semnalul UP este 1, numărătorul este incrementat, iar dacă semnalul UP este 0, numărătorul este decrementat. Dacă semnalul CE este 0, ieșirile numărătorului rămân neschimbate. Dacă numărarea se realizează în sus, semnalul TC (*Terminal Count*) devine 1 atunci când numărătorul ajunge la valoarea 1111, iar dacă numărarea se realizează în jos, semnalul TC devine 1 atunci când numărătorul ajunge la valoarea 0000.

Pentru proiectarea circuitului de înmulțire, procedați astfel:

1. Pe baza operațiilor elementare care trebuie realizate în fiecare etapă (operații descrise în secțiunea 2.10.1) și a descrierii componentelor utilizate, definiți semnalele de comandă necesare. De exemplu, pentru validarea operațiilor executate în etapa de inițializare trebuie activat un semnal de comandă numit, de exemplu, $LDINIT$. Acest semnal validează inițializarea registrului acumulator A cu 0 (resetarea acestuia), încărcarea de înmulțitorului X în registrul B , încărcarea înmulțitorului Y în registrul Q și inițializarea numărătorului N cu 4. Alte semnale de comandă sunt necesare pentru încărcarea registrului A cu valoarea de la ieșirea sumatorului, pentru decrementarea numărătorului N și pentru deplasarea la dreapta a registrelor A și Q .
2. Creați un nou proiect, iar apoi creați un fișier pentru schema circuitului de înmulțire și editați schema. Plasați mai întâi componentele necesare în cadrul schemei, iar apoi realizați interconectarea lor în mod asemănător cu schema din Figura 9.21 (registrele RX și RY nu sunt utilizate în proiectul de față). Conectați apoi în mod corespunzător intrările de control ale componentelor, cu excepția acelor intrări la care trebuie conectate semnalele de comandă definite anterior. Aceste semnale vor fi conectate ulterior. Salvați schema circuitului.
3. Creați un fișier pentru diagrama de stare a unității de comandă. Editați diagrama cu ajutorul programului StateCAD; forma diagramei poate fi cea din Figura 9.22. Specificați condițiile pentru tranziții cu ajutorul semnalelor $START$, $Q0$ (bitul 0 al registrului Q) și NZ (semnal care va fi activat atunci când numărătorul N ajunge la zero). Specificați apoi semnalele de comandă care trebuie activate în fiecare stare. Compilați diagrama pentru a crea un fișier sursă VHDL și închideți programul StateCAD.
4. În fereastra *Project Navigator* includeți în proiect fișierul cu diagrama de stare și fișierul VHDL creat pe baza diagramei. Creați apoi un simbol din fișierul VHDL prin selectarea acestui fișier și execuția comenzii *Create Schematic Symbol*.
5. Deschideți din nou fișierul cu schema și plasați în cadrul schemei simbolul creat anterior, care reprezintă unitatea de comandă a circuitului. Conectați intrările și ieșirile acestei unități, după care definiți porturile de I/E ale circuitului de înmulțire.
6. Creați un banc de test pentru circuitul de înmulțire utilizând programul HDL Bench.
7. Simulați funcționarea circuitului de înmulțire cu simulatorul ModelSim.