

Capitolul 4

Circuite logice programabile

Obiectul acestui capitol sunt de fapt doar circuitele cunoscute sub numele de **FPGA** (Field-Programmable Gate Array) și respectiv **CPLD** (Complex Programmable Logic Device), cu accentul pus pe circuitele unuia din producătorii reprezentativi de astfel de circuite, firma **Xilinx**. Se presupune că cititorul este familiarizat cu circuitele programabile clasice de tip **PLD** (cum ar fi PAL sau GAL) cunoscute și sub denumirea de **SPLD** (Simple Programmable Logic Device).

Păstrând proporțiile am putea spune că circuitele FPGA și CPLD sunt echivalentul modern al plăcilor universale de testare (breadboard) pe care erau montate circuite numerice simple (porți, bistabile, etc.) care puteau fi apoi interconectate prin intermediul unor fire de către utilizator pentru a testa o anumită aplicație. Afirmatia va fi, evident, justificată în cele ce urmează.

Înainte de a aborda problematica specifică trebuie menționate și circuitele cunoscute sub denumire de **ASIC** (Application-Specific Integrated Circuit). ASIC este un circuit integrat complex dedicat implementării unei aplicații particulare, mai degrabă decât unei utilizări generale. Un circuit ASIC este proiectat de o anumită companie și este realizat pentru un singur beneficiar (client).

Creșterea exponențială a densității de integrare a circuitelor VLSI face posibilă integrarea aproape a tuturor sub sistemelor numerice imaginabile. Un circuit ASIC poate include un microprocesor de 32 de biți împreună cu memorie și periferice complexe, caz în care mai poate fi referit și ca un **SoC** (System on Chip) prin care se înțelege un sistem de calcul realizat pe un singur microcircuit. SoC este în realitate o noțiune foarte largă ținând cont că printre periferice se pot număra sisteme analogice complexe, senzori de tip MEMS (bazați pe micro sau nano tehnologii mecanice) și chiar sisteme de radiofrecvență performante. Un exemplu tipic de ASIC și SoC în același timp este un circuit cu ajutorul căruia se realizează cea mai mare parte a unui telefon mobil.

Argumentele în favoarea conceptului SoC seamănă acum cu cele care, cu mult timp în urmă, justificau dezvoltarea circuitelor integrate numerice: scăderea dimensiunilor, reducerea consumului energetic și creșterea fiabilității comparativ cu sistemele care utilizau mai multe componente, reducerea costurilor legate de montare, totul la o altă scară de complexitate.

ASSP (Application Specific Standard Product) este tot un circuit integrat similar ASIC dedicat unei anume piețe de aplicații, dar destinat mai multor utilizatori (de aici și "standard"). Circuitele de tip ASSP sunt destinate mai multor clienți pentru că sunt de uz general, dar numărul acestora este mic pentru că aplicația este una specifică.

Un aspect important legat de circuitele ASIC sunt costurile legate de pregătirea fabricației și parțial de proiectarea lui. Costurile respective referite drept costuri **NRE** (Non Recurring Engineering) sunt uzual foarte mari (x100 000USD)

chiar și pentru un circuit pur numeric astfel că este important să fii sigur că aplicația se va vinde într-un volum mare pentru ca aceste costuri să poată fi amortizate. Un alt aspect este că, în acest caz, pregătirea procesului tehnologic de fabricație referitor la (re)dimensionarea suprafeței de siliciu pentru produse de tip ASIC durează mult timp, astfel că eventualele întârzieri la lansarea produsului pe piață pot reduce profitabilitatea

Sistemul numeric dintr-un ASIC care este și cel mai important sub sistem poate fi implementat prin mai multe tehnologii, funcție și de scara de producție necesară. Principal pentru realizarea acestuia se utilizează ca principale tehnologii:

- circuite total dedicate (**Full custom**)
- circuite bazate pe celule standard (**Standard Cell**)
- circuite FPGA

Dezavantajele menționate anterior legate de NRE se referă la prima tehnologie, deși ea este totuși și cea mai performantă sub aspectul vitezei de lucru, al densității de integrare și, nu în ultimul rând, al securității proprietății intelectuale.

Pentru o producție pe scară mică circuitele FPGA reprezintă întotdeauna o alternativă mai ieftină.

Circuitele cu celule standard reprezintă o cale de mijloc în ce privește raportul preț-performanță.

4.1 Circuite FPGA

Xilinx Inc. a lansat pe piață în anul 1985 o nouă tehnologie numită FPGA (Field Programmable Gate Array). Această tehnologie permite implementarea unor aplicații complexe ce folosesc logică secvențială și combinațională folosind doar un singur circuit. Aceasta duce la scăderea considerabilă a prețului de cost având și mari avantaje legate de flexibilitatea sistemelor dezvoltate pe baza acestei tehnologii. Un circuit FPGA atinge maximum de flexibilitate atunci când este configurat pe baza unor celule de memorie RAM și poate fi reprogramat de un număr nedefinit de ori în vederea aducerii unor îmbunătățiri sistemului prin modificarea funcțiilor existente sau adăugarea unor noi funcții. Datele de configurare sunt volatile și trebuie reîncărcate de fiecare dată la punerea sub tensiune a circuitului. Tehnologia a fost îmbunătățită și diversificată pentru a acoperi un spectru cât mai larg de cerințe existente privind consumul redus de energie, alimentarea la tensiuni reduse cu menținerea unor viteze mari de lucru.

Una din ideile esențiale este că circuitele FPGA trebuie să poată fi programate de cu ușurință de utilizator. "Punctele de configurare" pot fi celule de memorare SRAM, EPROM-UV, EEPROM/FLASH sau de tip fuzibil (fuse) similare circuitelor PROM. Celulele de tip fuzibil pot fi programate doar o singură dată. Configurațiile FPGA cu fuzibile sunt de tip OTP (One Time Programmable), și pot suferi doar o singură programare. Cele de tip EEPROM/FLASH pot fi reprogramate după ce au fost șterse în prealabil. Ambele variante menționate au

avantajul că nu sunt volatile, în timp ce configurațiile pe bază de celule SRAM trebuie programate la fiecare conectare a tensiunii de alimentare a sistemului.

Circuitele FPGA cu celule de configurare de tip EEPROM/FLASH au dat de fapt naștere unei noi familii, circuitele CPLD.

Pentru circuitele FPGA pe bază de celule SRAM, în mod uzual configurația este memorată într-o memorie ROM externă din care se încarcă automat la inițializarea sistemului.

Aria de siliciu ocupată de o celulă SRAM este mult mai mare decât aria ocupată de o celulă de tip fuzibil, deci un FPGA pe bază de SRAM va avea mai puține puncte de configurare decât unul bazat pe celule de tip fuzibil, și deci o densitate de integrare mai mică din punct de vedere al resurselor.

Evoluția rapidă a tehnologiilor SRAM (RAM static) de la sfârșitul anilor 80 și începutul anilor 90 și creșterea densităților de integrare au permis dezvoltarea rapidă și a circuitelor FPGA cu celule de configurare SRAM. Astfel aceste circuitele FPGA au început să fie folosite în mod masiv și ele de fapt domină piața.

Xilinx a dezvoltat mai multe familii de circuite FPGA cu celule de configurare SRAM., reprezentative fiind familiile mari (în ordinea apariției pe piață): XC4000, Spartan și Virtex. Familiile care mai contează la ora actuală, Spartan și Virtex au cunoscut fiecare la rândul lor o evoluție spectaculoasă datorată în esență progreselor tehnologiei CMOS.

Circuitele FPGA aveau performanțe mai bune decât circuitele PLD chiar și numai pentru faptul că puteau implementa o logică pe mai multe nivele, în timp ce circuitele PLD erau specializate pentru o logică pe doar două nivele. Circuitele FPGA sunt optimizate pentru o logică organizată pe mai multe nivele. Acest lucru permite implementarea unor circuite mult mai complexe decât cele care se pot implementa într-un circuit PLD.

Esențial mai era însă din punct de vedere al utilizatorului că datorită memoriei RAM de configurare un circuit FPGA poate fi (re)programat pentru mai multe aplicații, asemănător unui microprocesor.

Un circuit FPGA plus câteva configurații memorate într-un circuit de tip ROM pot fi folosite pentru mai multe modele hardware, selectând doar la inițializare (sau power up) programul dorit.

Mai mult, un sistem bazat pe circuite FPGA poate deveni un hardware generic, putând fi adaptat ușor de la o aplicație la alta.

Unele dintre cele mai noi domenii de aplicații pentru circuitele FPGA este combinarea acestora în masive pentru calcule matematice. Ecuațiile mapate în aceste dispozitive nu trebuie neapărat să fie ecuații hardware, ele putând reprezenta de exemplu algoritmi de criptare, decriptare sau compresie, etc.

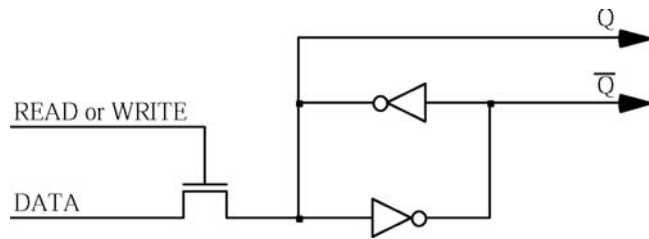


Figura 4.1 Celulă SRAM pentru configurarea FPGA

O celulă SRAM dintr-un FPGA are structura tipică din figura 4.1. După cum se poate observa există perechea de inversoare tipică celulei SRAM care va memora valoarea programată. O singură poartă de tranzistor de tip MOSFET n este folosită atât pentru scriere cât și pentru citire. Pentru controlul căii de configurare sunt folosite semnalele Q și \bar{Q} .

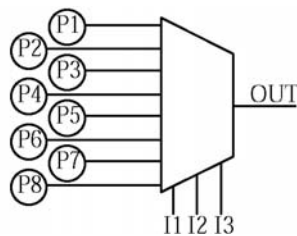


Figura 4.2 LUT cu 8 variabile de intrare

Una dintre cele mai folosite structuri în circuitele FPGA este tabela de căutare (Look-Up Table LUT). Prin conectarea a 2^N celule de memorie SRAM prin intermediul unui multiplexor se obține un LUT care poate implementa orice funcție de N variabile. Cu toate că este nevoie de un număr mare de celule SRAM, uzual se construiesc LUT-uri cu 5 variabile de intrare. Pentru funcții cu mai mult de cinci variabile se folosesc multiplexoare pentru a forma LUT-uri de 6 sau 7 variabile.

4.1.1 Seria Spartan-IIE (Xilinx)

Seria Spartan-IIE este una din ultimele generații de circuite FPGA Xilinx din familia Spartan (cea mai nouă este Spartan-3). Familia permite dezvoltarea aplicațiilor numerice de complexitate medie spre mare, cu costuri de dezvoltare și producție foarte mici. Caracteristicile principale, în termeni de resurse disponibile, pentru seria Spartan-IIE sunt date în tabelul 4.1.

Familia poate opera cu un semnal de ceas de până la 200 MHz și oferă resurse multiple: memorie RAM de configurare distribuită și organizată ca blocuri RAM, existența a 19 variante de interfețe electrice de intrare/ieșire configurabile de utilizator, patru bucle de întârziere digitale (DLL), etc..

Tabelul 4.1 Resursele familiei Spartan-IIE

Circuit	Celule Logice	Porti logice sistem	CLB Matrice (R x C)	Total CLB	Max I/O Utilizator	Max Perechi I/O Diferentiale	RAM biti Distribuiri	Blocuri RAM Bits
XC2S50E	1,728	23,000 - 50,000	16 x 24	384	182	83	24,576	32K
XC2S100E	2,700	37,000 - 100,000	20 x 30	600	202	86	38,400	40K
XC2S150E	3,888	52,000 - 150,000	24 x 36	864	265	114	55,296	48K
XC2S200E	5,292	71,000 - 200,000	28 x 42	1,176	289	120	75,264	56K
XC2S300E	6,912	93,000 - 300,000	32 x 48	1,536	329	120	98,304	64K
XC2S400E	10,800	145,000 - 400,000	40 x 60	2,400	410	172	153,600	160K
XC2S600E	15,552	210,000 - 600,000	48 x 72	3,456	514	205	221,184	288K

Resursele logice sunt reprezentate de blocurile logice configurabile **CLB** (Configurable Logic Blocks) organizate matricial (linii, coloane). Resursa de bază care se alocă unei aplicații este numită **slice**, astfel fiecare CLB conține două slice-uri, fiecare slice fiind compus din patru LUT-uri (fiecare LUT putând realiza o funcție logică de 4 variabile independente), o logică de transport rapid și un element de memorare (bistabil de tip D). În plus, fiecare CLB are două buffere dedicate cu ieșire de tip tri-state care pot fi folosite pentru a implementa magistrale interne de tip tri-state (3-S). LUT-urile pot fi legate în cascadă pentru a realiza funcții de 5 sau 6 variabile, pot fi configurate ca un bloc de memorie RAM cu dimensiunea de 1x16 biți sau ca un registru de deplasare. De exemplu se poate implementa direct un multiplexor 8:1 într-un singur CLB.

RAM-ul distribuit este o soluție ideală pentru aplicații care necesită memorii rapide, de dimensiune mică, amplasate cât mai aproape de de logica principală.

Blocurile RAM sunt reprezentate prin masive RAM cu dimensiunea de 4096 biți. Blocurile RAM pot fi configurate ca memorii de tip RAM sau ROM. Aceste memorii pot fi inter conectate pentru a obține masive de memorie mai mari.

Buclele de întârziere digitale (DLL) asigură distribuția semnalului de ceas eliminând întârzierile. Deoarece buclele de întârziere sunt complet digitale nu apar problemele legate de zgomot.

Arhitectura seriei Spartan-IIE este ilustrată de schema bloc a unui circuit FPGA din această serie dată din figura 4.3.

Pe schema bloc se observă componentele principale:

- blocurile **IOB** (Input Output Blocks) care asigură interfața între pinii circuitului și circuitele interne (logica I/O)
- blocurile CLB care conțin elementele funcționale pentru implementarea logicii
- blocurile de RAM dedicat, fiecare bloc având o dimensiune de 4096 biți
- blocurile DLL care asigură distribuția semnalului de ceas și compensarea întârzierilor

Toate aceste componente sunt plasate într-o matrice de interconectare multifuncțională (matrice de rutare GRM - General Routing Matrix) prin intermediul căreia se realizează configurarea la nivel de sistem.

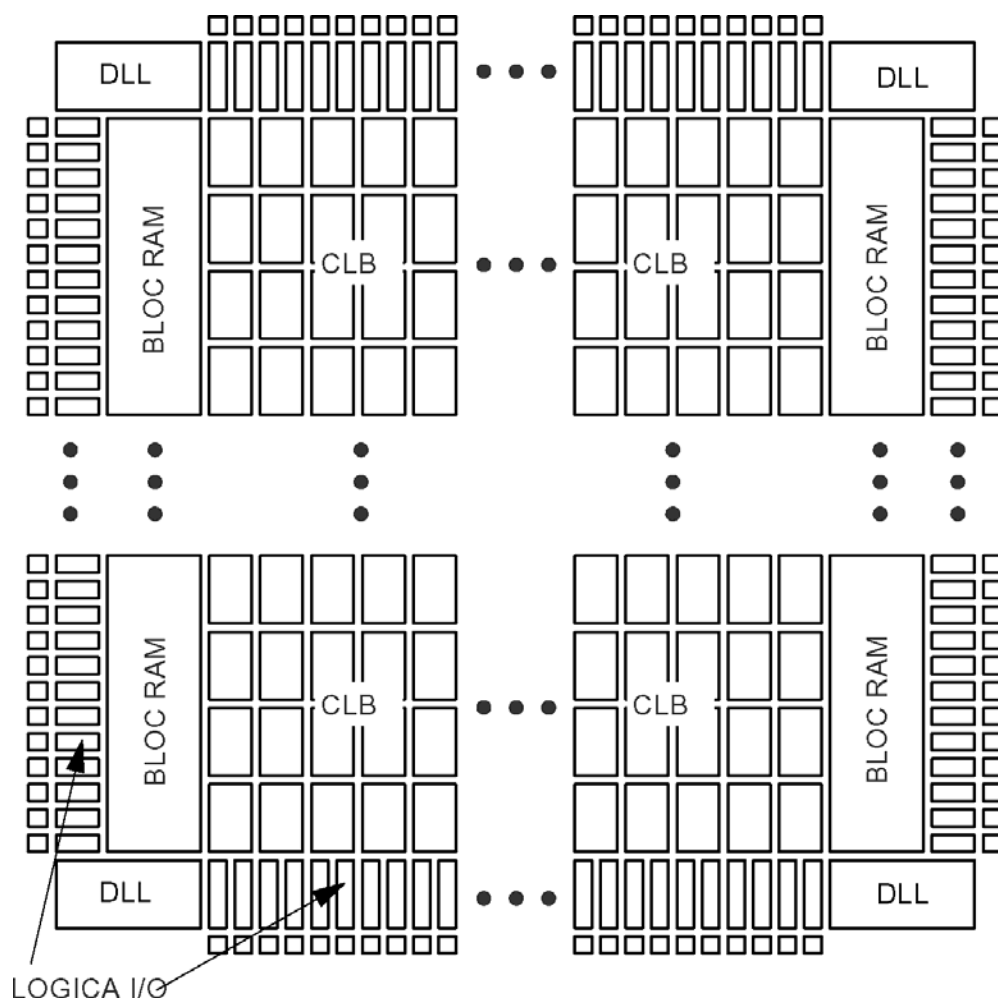


Figura 4.3 Schema bloc a unui circuit Spartan-IIE Xilinx

Blocurile CLB conțin logica centrală, având acces la matricea de rutare. Blocurile IOB sunt plasate în jurul memoriei și al CLB-urilor pentru a permite accesul la pinii circuitului.

Fiecare din aceste componente funcționale va fi discutat în detaliu în continuare.

4.1.1.1 Resursele Spartan-IIE

a. Blocul IOB

O schema detaliată a blocului IOB este dată în figura 4.4

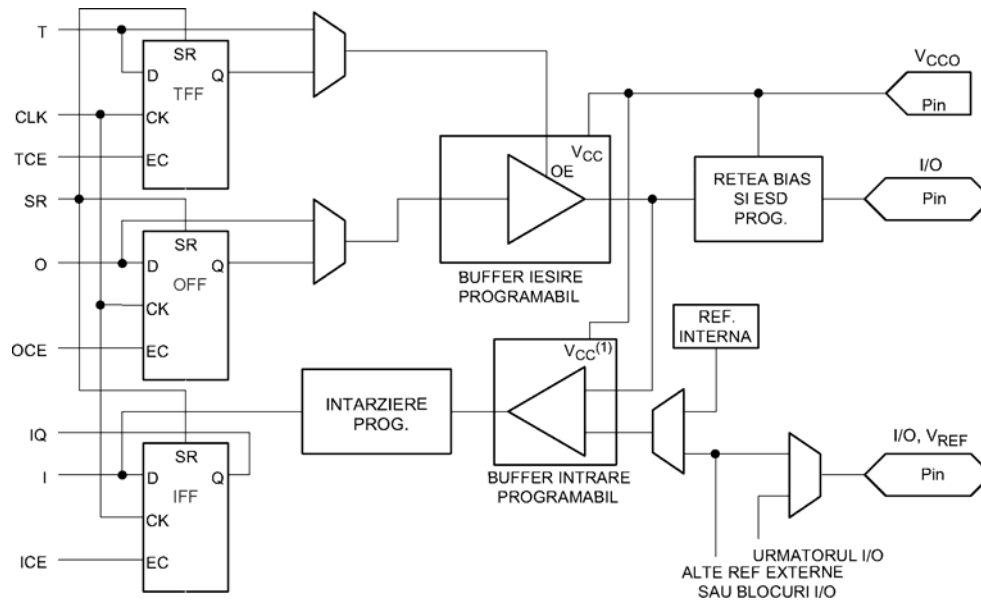


Figura 4.4 Blocul IOB

Una din caracteristicile importante ale acestui bloc este faptul că el permite implementarea unei mari varietăți de standarde de interfață electrică intrare-ieșire. Standardele suportate sunt date în tabelul 4.2.

Cele trei registre disponibile într-un bloc IOB pot funcționa fie ca registru de tip D, cu comutare pe front negativ sau pozitiv, fie ca latch transparent.

Toate bistabilele au același semnal de ceas (CLK), dar semnalul de validare a ceasului (CE) este independent pentru fiecare bistabil. În plus față de aceste două semnale mai există un semnal comun care poate fi configurat ca Set sau Reset (SR) pentru fiecare bistabil. Semnalul SR poate fi configurat individual, pentru fiecare registru, ca fiind unul sincron sau asincron.

O caracteristică care nu este prezentată în schema de mai sus o prezintă posibilitatea controlului polarității (direct sau inversat). Polaritatea buffer-elor de I/O poate fi selectată independent.

La nivelul IOB-ului se poate identifica o cale de intrare și respectiv una de ieșire care sunt parcurse de semnal.

Calea de intrare constă în esență într-un buffer care face legătura între logica internă și pinul de intrare prin intermediul și al unui bistabil.

Un element opțional de întârziere de tip bistabil D elimină timpul de hold. Această întârziere este sincronizată cu rețeaua internă de distribuție a semnalului de ceas. Fiecare buffer de intrare poate fi programat astfel încât intrarea respectivă să fie compatibilă cu unul din standardele prezentate mai sus. În unele cazuri este necesară asigurarea unei tensiune V_{REF} de către utilizator.

Tabelul 1.2 Standarde I/O suportate

Standard I/O	Tens. Ref. [V] (V _{REF})	Intrare [V] (V _{CCO})	Tens. iesire [V] (V _{CCO})	Tens. Term. [V] (V _{TT})
LVTTTL (2-24 mA)	N/A	3.3	3.3	N/A
LVC MOS2	N/A	2.5	2.5	N/A
LVC MOS18	N/A	1.8	1.8	N/A
PCI (3V, 33 MHz/66 MHz)	N/A	3.3	3.3	N/A
GTL	0.8	N/A	N/A	1.2
GTL+	1.0	N/A	N/A	1.5
HSTL Class I	0.75	N/A	1.5	0.75
HSTL Class III	0.9	N/A	1.5	1.5
HSTL Class IV	0.9	N/A	1.5	1.5
SSTL3 Class I and II	1.5	N/A	3.3	1.5
SSTL2 Class I and II	1.25	N/A	2.5	1.25
CTT	1.5	N/A	3.3	1.5
AGP	1.32	N/A	3.3	N/A
LVDS, Bus LVDS	N/A	N/A	2.5	N/A
LVPECL	N/A	N/A	3.3	N/A

Calea de ieșire include principal buffer de ieșire 3-state conectat cu ieșirea la pinul respectiv. Sursa semnalului de ieșire poate să fie ieșirea buffer-ului, poate să provină direct de la logica internă sau prin intermediul unui bistabil de tip D intermediar. Controlul 3-state este controlat de logica internă direct sau prin intermediul unui bistabil de tip D, asigurând sincronizarea.

Fiecare dintre driver-ele de ieșire poate debita un curent de până la 24 mA și absorbi un curent de maxim 48 mA (depinzând evident de tensiunea de alimentare!). Aceste valori de curentului precum și viteza de variație a semnalului (slew rate) sunt programabile implicit sau explicit .

Un circuit opțional numit weak-keeper poate fi conectat la fiecare pin. Circuitul respectiv este similar circuitului bus hold prezent la unele din circuitele pentru cuplarea la magistrală. Când acest circuit este activat, pinul circuitului va putea fi „tras” slab către „1” sau „0”, în funcție de starea anterioară a semnalului de intrare. Astfel în momentul în care ieșirea trece în High-Z circuitul weak-keeper va păstra pentru pin ultima valoare avută la intrare.

În afară de acest circuit mai există și posibilitatea conectării la pin a unor rezistențe interne de valoare mare către sursa de alimentare (pull-up) sau către masă (pull-down) pentru a fixa potențialul pinului respectiv.

După configurarea circuitului pinii care rămân nefolosiți de aplicație sunt forțați în starea High-Z (întă impedență) astfel că trebuie utilizați rezistori externi pentru fixarea potențialului, dacă nu s-au folosit facilitățile menționate mai sus.

Câteva din standardele descrise mai sus necesită o tensiune externă care trebuie aplicată pe pinii V_{CCO} sau V_{REF} . Aceste tensiuni trebuie asigurate din exterior și sunt conectate la pini dedicați pentru fiecare banc. Ca urmare apar o serie de restricții legate de modul în care pot fi combinate standardele de I/O în cadrul unui banc.

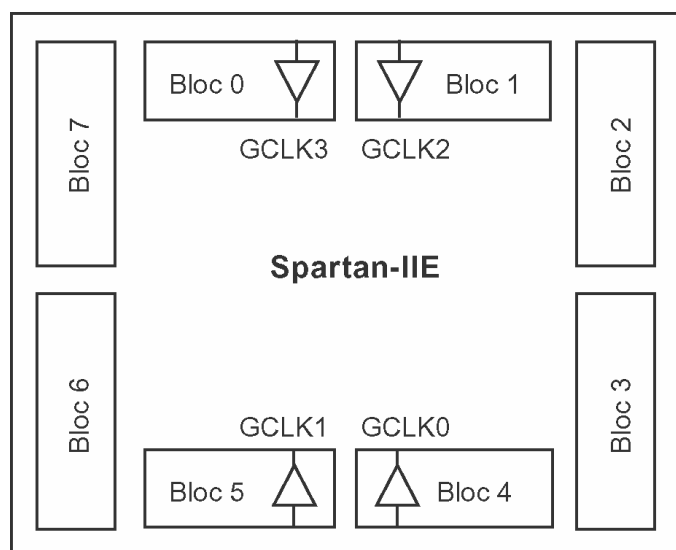


Figura 4.2 Bancurile I/O

Pentru utilizator este important de știut că fiecare latură a circuitului FPGA este împărțită în două bancuri, rezultând opt bancuri (figura 4.5).

Toate conexiunile externe sunt protejate împotriva descărcărilor electrostatice (ESD). După configurarea circuitului diodele din circuitele de protecție ESD sunt conectate la V_{CCO} dacă sunt folosite standardele LVTTTL, PCI, HSTL, SSTL, CTT și AGP. Toate blocurile IOB din familia Spartan-IIE suportă standardul IEEE 1149.1 (boundary-scan test).

Fiecare banc are asociați pinii V_{CCO} și V_{REF} , tensiunile la care se vor conecta aceștia de către utilizator vor fi determinate de standardele I-O folosite.

De exemplu pentru variantele de încapsulare TQ144 și PQ208 intrările V_{CCO} pentru cele opt bancuri sunt conectate împreună, iar intrările V_{REF} sunt conectate separat. **În cadrul unui banc se pot folosi numai standarde care necesită aceeași valoare a tensiunii V_{CCO} .** În tabelul 4.3 sunt prezentate standardele I-O compatibile din acest punct de vedere.

Tabelul 4.3 Standarde compatibile

V _{cco}	Standarde compatibile
3.3V	PCI, LVTTTL, SSTL3 I, SSTL3 II, CTT, AGP, LVPECL, GTL, GTL+
2.5V	SSTL2 I, SSTL2 II, LVCMOS2, LVDS, Bus LVDS, GTL, GTL+
1.8V	LVCMOS18, GTL, GTL+
1.5V	HSTL I, HSTL III, HSTL IV, GTL, GTL+

b. Blocul CLB

Celula de bază pentru CLB este reprezentată de celula logică **LC** (Logic Cell). O celulă logică este formată dintr-un generator de funcții cu 4 intrări, o logică pentru lanțul de transport (carry chain) și un element de memorare.

Ieșirea generatorului de funcții din fiecare celulă logică poate să fie legată la ieșirea CLB-ului sau la intrarea bistabilului de tip D. Fiecare CLB conține patru celule logice, organizate în slice-uri similare cu cel din figura 4.6.

În plus CLB-ul conține logica necesară pentru combinarea ieșirilor celor patru LC-uri pentru a obține un generator de funcții cu cinci sau șase intrări.

Generatoarele de funcții sunt implementate prin intermediul **LUT** (Look-Up Table). În plus, fiecare LUT poate fi folosit ca un RAM static sincron, cu dimensiunea de 16x1 bit. Două LUT-uri pot fi combinate pentru a crea un bloc RAM static sincron de 16x2 biți sau 31x1 bit. De asemenea un LUT poate fi folosit și ca registru de deplasare pe 16 biți.

Elementele de memorare din CLB pot fi configurate fie ca bistabile de tip D cu comutare pe front sau ca bistabile D transparente (latch). Intrările D ale bistabilelor pot fi conectate la generatorul de funcții sau la una dintre intrările LUT-ului. În plus față de semnalele CLK și CE, fiecare slice are semnale de set și reset sincrone (SR și BY). SR forțează elementul de memorare în starea inițială precizată la configurare, iar BY forțează circuitul în starea complementară celei de la configurarea inițială.

Aceste semnale pot fi de asemenea configurate pentru a acționa sincron sau asincron cu ceasul CLK. Cele două semnale pot fi complementate independent și sunt comune pentru cele două bistabile din fiecare slice.

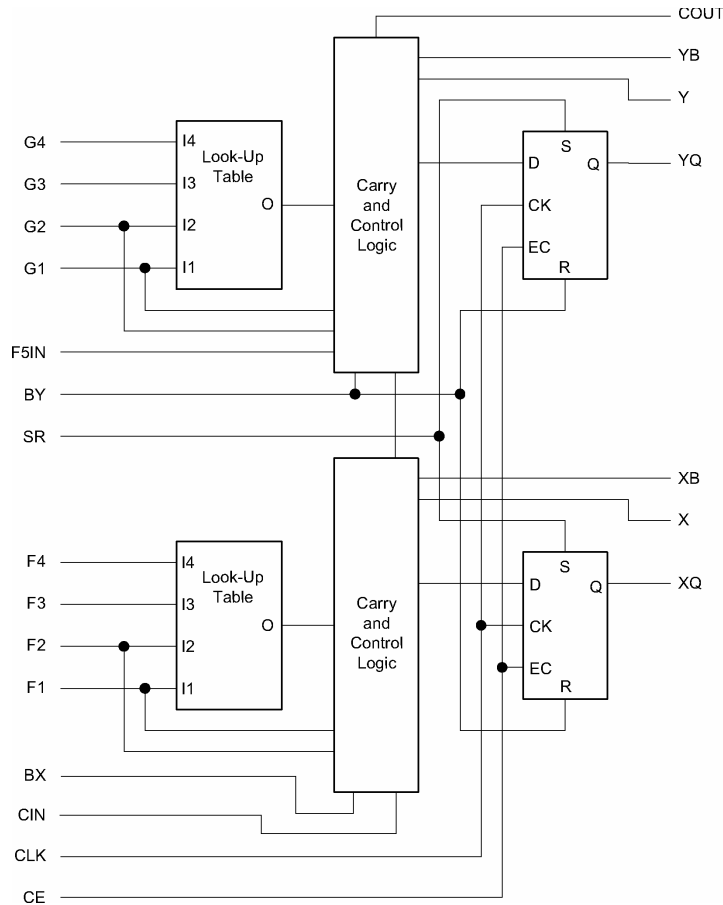


Figura 4.6 Slice Spartan-IIE

Circuitele dedicate de transport (Carry and Control Logic) asigură o viteză mare de lucru, permițând implementarea unor funcții aritmetice rapide. CLB-ul are două lanțuri de transport per slice. Lățimea unui lanț de transport este de 2 biți. Logica aritmetică include o poartă de tipul XOR care permite implementarea unui sumator complet în interiorul unei singure celule logice. În plus, o poartă AND dedicată îmbunătățește implementarea operațiilor de multiplicare. Călea de transport poate fi folosită și pentru cascada funcțiilor.

La nivelul CLB-ului mai există și **o logică adițională**. Astfel multiplexorul F5 din fiecare slice (figura 4.7) combină ieșirea celor două LUT-uri ale unui slice pentru a obține o funcție de cinci variabile sau un multiplexor 4:1. În mod similar multiplexorul F6 combină ieșirile celor două multiplexoare F5. Acest lucru permite realizarea unei funcții de 6 variabile sau realizarea unui multiplexor de 8:1.

Fiecare CLB mai conține și 4 căi de reacție locale pentru a nu consuma din resursele globale de rutare.

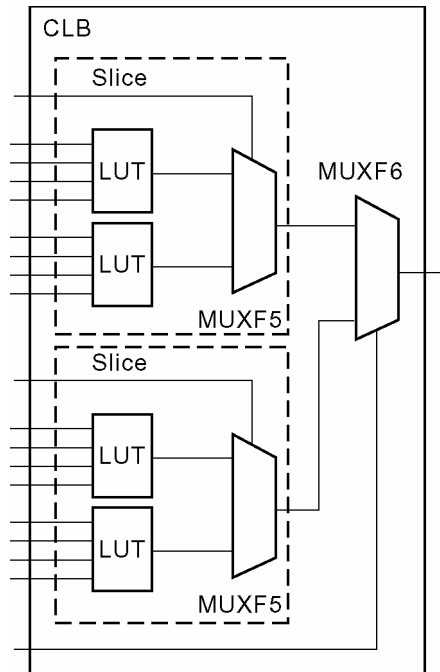


Figura 4.7 Logica adițională

Fiecare CLB conține două buffere tri-state prin care acesta se poate conecta la magistralele interne ale circuitului. Fiecare buffer are un control independent al stării 3-state (linia T activă în 0 logic). Când toate bufferele legate la magistrala internă sunt dezactivate, linia respectivă din magistrală este în starea „1”. Dacă două drivere conectate la aceeași linie din magistrala internă sunt în stări complementare nu va rezulta un conflict, linia fiind trasă în „0” (un ȘI cablat).

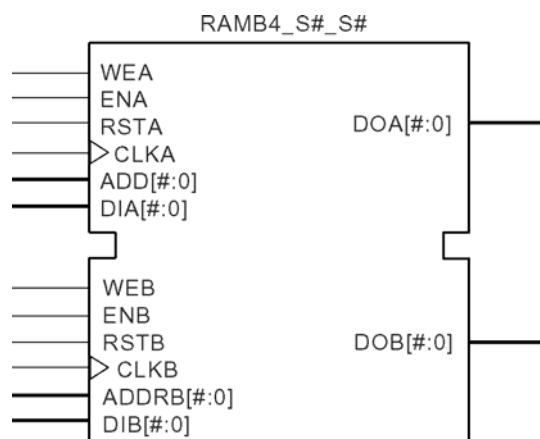
c. Blocul RAM

Spartan-III încorporează mai multe blocuri RAM care completează memoria RAM distribuită la nivelul CLB-urilor.

Blocurile de memorie RAM sunt organizate în coloane. Cele mai multe circuite din familia Spartan-III conțin două coloane, distribuite pe laturile „verticale”(figura 4.3). De exemplu XC2S400E are 4 coloane de blocuri RAM, iar XC2S600E are 6 coloane de blocuri RAM.

Fiecare bloc RAM, este complet sincron, de tip bi-port (portul A și portul B), cu o dimensiune de 4096 biți, având control independent (semnale dedicate) pentru fiecare port.

Lungimea cuvântului de date DIA/DOA și DIB/DOB pentru fiecare din cele două porturi poate fi configurată independent. În funcție de lungimea cuvântului de date se pot obține următoarele tipuri de memorii: 1x4096 biți, 2x2048 biți, 4x1024 biți, 8x512 biți sau 16x256 biți.



Figură 4.8 Bloc RAM

4.1.1.2 Rutarea programabilă

La modul general viteza unui circuit numeric complex este limitată de cea mai mare întârziere de propagare intrare-ieșire. Timpii de întârziere țin atât de resursele de tip componentă cât și de resursele de tip rețea de rutare. Arhitectura Spartan-IIe împreună cu mediul de programare asociat care face plasarea componentelor și rutarea interconexiunilor au fost optimizate pentru a minimiza aceste întârzieri. Cea mai bună cale de rutare obținabilă va fi aleasă automat pe baza unor restricții pentru timpul de întârziere definite de utilizator.

Un rol esențial în acest sens îl au, pentru orice circuit FPGA, resursele numite matrice de rutare sau, în cazul circuitelor FPGA Xilinx, GRM-General Routing Matrix. Există mai multe nivele de rutare: local, la nivelul I-O și global. Pentru rutare mai contează și organizarea matricială a CLB-urilor.

Rutarea locală este descrisă în figura 4.9. Există trei tipuri de conexiuni

- interconexiuni între LUT-uri, bistabile și matricea de rutare generală (GRM)
- conexiuni de mare viteză interne CLB-ului, care conectează LUT-uri aparținând aceluiași CLB
- conexiuni directe de mare viteză între CLB-uri adiacente orizontale care elimină întârzierile produse de GRM

Resursele pentru rutarea generală sunt localizate atât pe verticală cât și pe orizontală, fiind asociate cu liniile și coloanele de CLB-uri. Există mai multe posibilități de rutare generală.

Adiacent fiecărui CLB există un GRM. GRM reprezintă matricea generală de comutație prin care sunt conectate resursele orizontale și verticale constituind calea de acces a CLB-ului către calea de rutare generală.

Mai există 24 linii de rutare care asigură legătura între GRM-urile adiacente în toate cele 4 direcții.

96 de buffere multiple (grupat câte 6) rutează semnalele GRM-ului către alte 6 GRM-uri, în fiecare dintre cele patru direcții (stânga, dreapta, sus, jos). Aceste seturi de 6 linii pot fi conectate numai la capete. O treime dintre liniile multiple sunt bidirecționale, iar restul sunt unidirecționale.

Există 12 linii bidirecționale care distribuie eficient semnalele pe toată lungimea și lățimea microcircuitului.

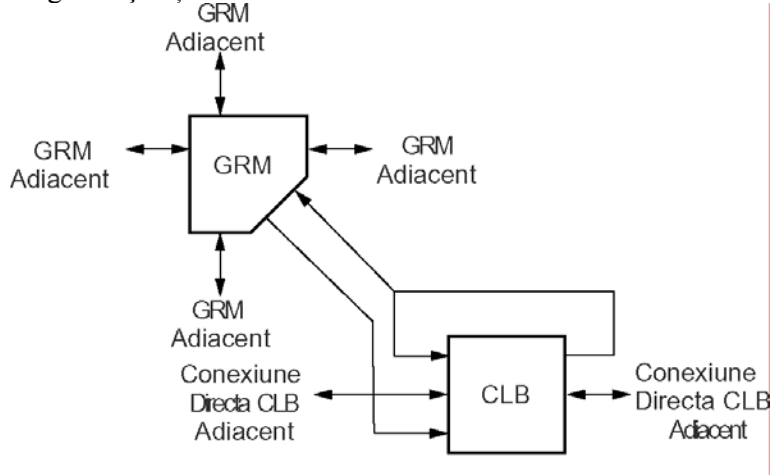


Figura 4.9 Rutarea locală

De-a lungul periferiei microcircuitului se află distribuite resurse care realizează legătura între matricea CLB și blocurile IOB, cu alte cuvinte care realizează **rutarea I-O**. Logica de rutare adițională poartă denumirea de rutare VersaRing. Există clase de semnale care necesită tehnici de rutare dedicate. În familia Spartan-III sunt oferite resurse de rutare speciale pentru două clase de semnale:

- **resurse de rutare orizontale** pentru magistrale de tip tri-state constând în 4 linii de magistrală care sunt alocate pentru fiecare rând de CLB-uri, după cum se poate observa în figura 4.10
- **două rețele verticale** sunt dedicate pentru propagarea semnalelor de transport între două CLB-uri adiacente

Resursele de **rutare globală** sunt folosite pentru distribuirea semnalelor de ceas și a altor semnale care necesită un fan-out mare. Spartan-III include două rețele de rutare globală, referite ca o rețea de rutare globală primară și o rețea de rutare globală secundară.

Rețeaua globală primară este formată din patru rețele globale, cu pini de intrare dedicați. Aceste rețele sunt folosite pentru distribuirea semnalului de ceas. Fiecare dintre aceste patru rețele poate distribui semnalul de ceas la toate CLB-urile, blocurile IOB și blocurile RAM din circuit.

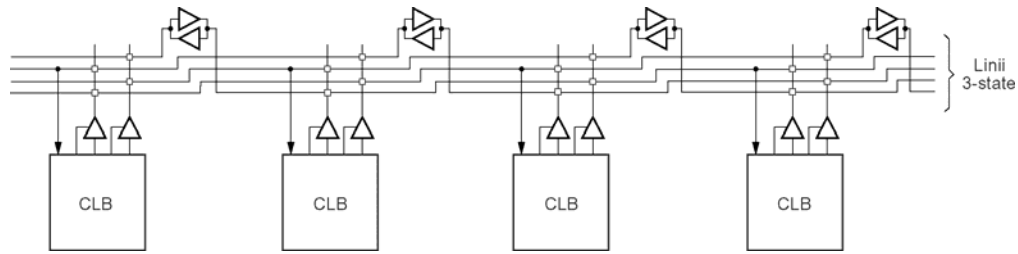


Figura 4.10 Resurse de rutare orizontale

Rețeaua globală secundară este formată din de 24 linii principale (backbone lines), 12 linii pe fața microcircuitului și 12 linii pe spatele microcircuitului. Prin aceste linii până la 12 semnale unice pe coloană pot fi distribuite prin intermediul celor 12 linii verticale. Aceste resurse secundare sunt mai flexibile decât cele primare deoarece nu sunt restricționate de conectarea lor la anumiți pini dedicați ai dispozitivului.

Distribuirea semnalului de ceas pentru Spartan-IIe se asigură prin intermediul rețelei globale primare. O rețea tipică de distribuție a semnalului de ceas este dată în figura 4.11.

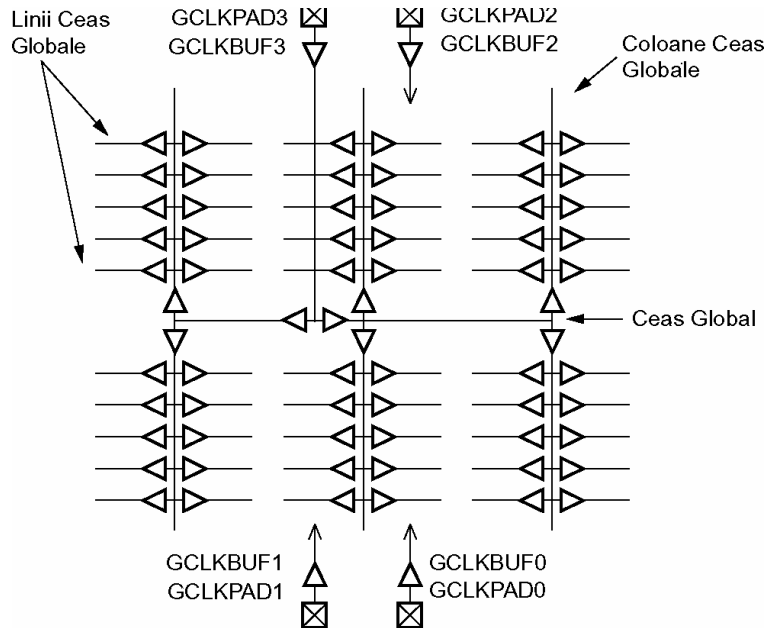


Figura 4.11 Rețeaua de distribuție a semnalului de ceas

Sunt prevăzute patru buffere globale, două în zona de centru față și două în zona de centru spate a microcircuitului (vezi și figura 4.5). Acestea comandă cele patru rețele globale primare. Sunt prevăzuți patru pini dedicați pentru semnalul de ceas care prin intermediul unor buffere pot comanda cele patru linii globale

primare. Semnalul pentru rețeaua globală primară este selectat fie dintre acești pini dedicați, fie din semnalele date de rețeaua generală de rutare.

Fiecare intrare de tip global pentru semnal de ceas (GCLK) are asociată și o buclă digitală de întârziere **DLL** (figura 4.12). Această buclă poate să elimine întârzierile între semnalul de ceas intern (CLKOUT) și semnalul de ceas extern (CLKIN). Fiecare circuit DLL poate comanda două rețele globale de distribuire a ceasului. Circuitul DLL monitorizează semnalul de ceas de intrare și ajustează automat elementul de întârziere pentru a asigura sincronizarea. El are în componență o linie de întârziere numerică auto-programabilă, astfel că o întârziere suplimentară este introdusă pentru ca semnalul de ceas să ajungă la bistabile exact după o perioadă de ceas, după ce acesta a ajuns la intrare. În acest mod se asigură comutarea tuturor circuitelor bistabile sincron cu același semnal de ceas. Acest sistem în buclă închisă elimină întârzierile (decalările temporale) care apar la distribuția semnalului de ceas.

Pentru a garanta o funcționare corectă la pornire (power-up), **DLL-ul poate întârzia terminarea procesului de configurare până la intrarea sa în regim de sincronism (lock).**

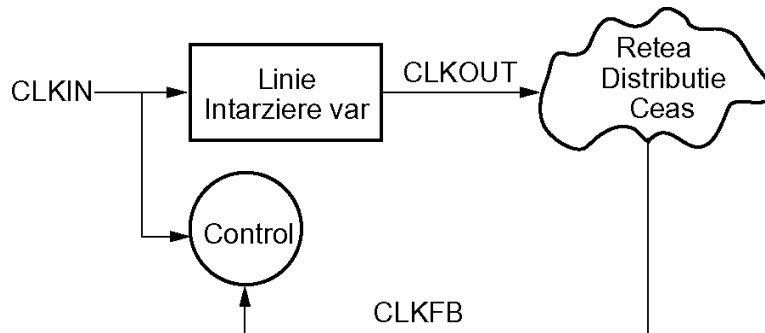


Figure 4.12 Circuitul DLL

În plus DLL-urile asigură controlul semnalului de ceas în mai multe domenii: asigură patru faze ale semnalului de ceas defazate între ele cu 90° , pot dubla frecvența semnalului de ceas și realiza o divizare a semnalului de ceas cu factori de 1.5, 2, 2.5, 3, 4, 5, 8, 16.

De asemenea circuitele DLL pot fi utilizate pentru a asigura sincronizarea în cazul când distribuția semnalului de ceas se face pentru mai multe circuite Spartan-IIe.

4.2 Circuite CPLD (Complex Programmable Logic Device)

Tehnologia CPLD se adresează unui domeniu larg de aplicații cu un grad de complexitate mic sau mediu (oricum mai mic decât la FPGA). Familiile CPLD ale firmei Xilinx sunt bazate pe o tehnologie combinată FLASH-EPROM și RAM. Configurația este memorată permanent într-o memorie de tip FLASH, fiind transferată la punerea sub tensiune (power-up) într-o memorie de configurare de tip RAM (oarecum similară cu cea de la FPGA). Structura de configurare astfel rezultată este nevolatilă.

Tehnologia CPLD oferă performanțe asemănătoare tehnologiei FPGA pentru aplicații de complexitate medie, la un preț de cost mai scăzut. În același timp circuitele CPLD au o rețea de interconexiuni de lungime fixă (o proprietate a matricii de interconectare) ceea ce face ca sistemele numerice dezvoltate cu un circuit CPLD să prezinte întârzieri predictibile în totalitate chiar din faza de proiectare (vezi figura 4.13). Întârzierile dintre fiecare două celule logice conținute în CPLD sunt fixe și se cunosc. Aceasta se datorează faptului că structura de interconexiuni dintr-un CPLD este formată din linii conductoare de lungime constantă ce străbat structura circuitului pe toată lungimea și lățimea acestuia. În contrast cu CPLD-urile, FPGA-urile au o structură de interconexiuni formată din segmente care străbat circuitul, iar capetele acestora sunt conectate de o matrice de interconectare permițând semnalelor să ajungă de la o celulă logică la alta. Numărul de segmente necesare pentru a conecta două celule logice nu este nici fix și nici predictibil, deci întârzierile nu se pot cunoaște decât după ce se face asignarea și plasarea celulelor (după implementare).

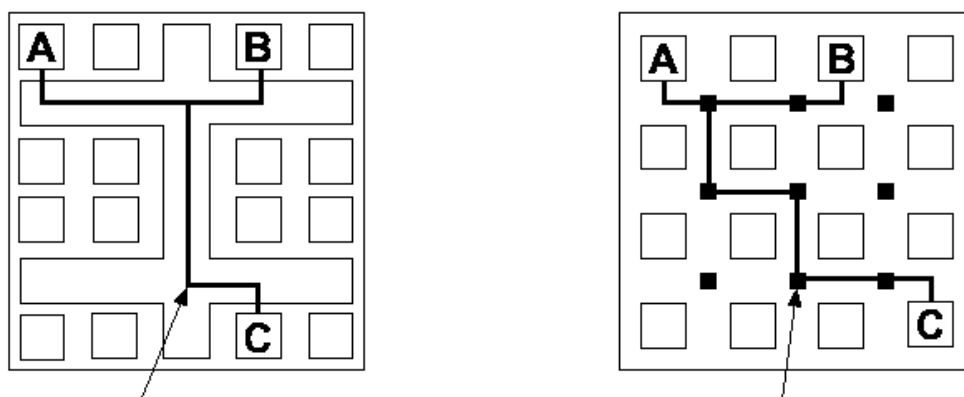


Figura 4.13 Comparatie CPLD-FPGA privind poziția punctului de conexiune

În figura 4.13 este ilustrat modul diferit de interconectare internă la circuitele CPLD față de circuitele FPGA.

Familiile reprezentative de circuite CPLD ale firmei Xilinx sunt: XC9500, XC9500XL, XC9500XV, Cool Runner II, Cool Runner XPLA3. Familiile Cool Runner sunt destinate unor aplicații unde consumul propriu trebuie să fie foarte mic și vitezele de funcționare trebuie să fie mari. Actualmente (2004) cea mai nouă familie CPLD Xilinx este Cool Runner II.

Seriile XC9500 (XL, XV) diferă între ele prin anumite valoarea tensiunii nominale de alimentare a circuitului propriu-zis ($V_{CC_{INT}}$) și respectiv ale interfeței externe ($V_{CC_{IO}}$).

Ele sunt împărțite în diverse clase din punct de vedere al frecvenței de lucru (al timpului de propagare) și sunt disponibile în diverse variante de încapsulare.

4.2.1 Seria Xilinx XC9500XL

Familia XC9500XL este realizată într-o tehnologie CMOS avansată (0.35 microni) numită FastFLASH și cuprinde circuite alimentate ($V_{CC_{INT}}$) la tensiunea nominală de 3,3V. Întârzierea minimă pin la pin este de 5ns, iar frecvența internă de lucru este de până la 200 MHz. Porturile I/O tolerează semnale de intrare provenind de la circuite alimentate la 5V, 3,3V și 2,5V și sunt capabile să furnizeze semnale de ieșire compatibile cu circuite TTL, sau CMOS alimentate la 3,3V și 2,5V.

Alte caracteristici sunt inversarea locală a semnalelor de ceas, validarea individuală a ieșirilor, controlul timpilor de front (slew rate), individual pe fiecare ieșire, capabilități sporite de securizare a datelor (protecție la citire, scriere, sumă de control - checksum). Circuitele sunt garantate la peste 10.000 de cicluri de scriere/ștergere, menținerea datelor fiind garantată 20 de ani.

Familia cuprinde circuite de patru densități (ca număr de macrocelule), compatibile pin la pin pentru același tip de încapsulare. Densitățile sunt cuprinse între 36 și 288 macrocelule cu 800 respectiv 6.400 de porți utilizabile (codurile sunt XC9536XL, XC9572XL, XC95144XL și XC95288XL). Numărul prezent în cod este de fapt numărul de macrocelule disponibile.

Circuitele din această familie sunt disponibile în mai multe variante de încapsulare și sunt compatibile pin la pin cu circuitele similare ca densitate din familia XC9500 sau XC9500XV (pentru același tip de capsulă). Mai mult, această compatibilitate pin la pin există și între circuitele de densități diferite din aceeași familie, atunci când au aceeași încapsulare. În momentul în care, din diverse motive, densitatea actuală nu este suficientă (proiectul nostru nu „încapă” în circuit), se poate trece fără probleme, la un circuit cu o densitate superioară, având același pin-out.

În tabelele următoare sunt prezentate sintetic circuitele din aceasta familie. În tabelul 4.4 sunt prezentate numărul de macrocelule, numărul de porți utilizabile (usable gates), numărul de circuite bistabile (registers), timpii de propagare t_{PD} și frecvența echivalentă a sistemului (f_{system}).

Tabelul 4.4 Resurse pentru seria 9500XL

	XC9536XL	XC9572XL	XC95144XL	XC95288XL
Macrocells	36	72	144	288
Usable Gates	800	1,600	3,200	6,400
Registers	36	72	144	288
T _{PD} (ns)	5	5	5	6
T _{SU} (ns)	3.7	3.7	3.7	4.0
T _{CO} (ns)	3.5	3.5	3.5	3.8
f _{SYSTEM} (MHz)	178	178	178	208

În tabelul 4.5 sunt prezentate variantele de încapsulare disponibile pentru fiecare circuit, împreună cu numărul de intrări/ieșiri (I/O pins) disponibile pentru fiecare variantă de încapsulare.

Tabelul 4.5 Variante de încapsulare pentru seria 9500XL

	XC9536XL	XC9572XL	XC95144XL	XC95288XL
44-pin PLCC	34	34	-	-
44-pin VQFP	34	34	-	-
48-pin 0.8 mm CSP	36	38	-	-
64-pin VQFP	36	52	-	-
100-pin TQFP	-	72	81	-
144-pin 0.8 mm CSP	-	-	117	-
144-pin TQFP	-	-	117	117
208-pin PQFP	-	-	-	168
256-pin BGA	-	-	-	192
256-pin FBGA	-	-	-	192
280-pin 0.8 mm CSP	-	-	-	192

Circuitele înglobează un anumit număr de Blocuri Funcționale (Functional Block) și de Blocuri de Intrare/Ieșire (I/O Block) interconectate prin intermediul unei matrici de interconectare numită FastConnect II. Fiecare Bloc Funcțional are maxim 54 de intrări și maxim 18 ieșiri. Fiecare Bloc Funcțional este compus din 18 macro celule (macrocell). La nivelul Blocului Funcțional implementarea este tip sumă de produse, fiind disponibili până la 90 de termeni produs (Product Terms).

Fiecare macro celulă conține un bistabil - register (numărul de bistabile disponibile este egal cu numărul de macro celule al circuitului) care poate fi configurat ca fiind de tip D sau T sau absent când este neutilizat, în cazul în care funcția implementată este pur combinațională.

Luând ca exemplu circuitul CPLD XC9572XL acesta este format din 72 de macro celule grupate în patru blocuri funcționale, numărul de porți utilizabile fiind de 1600. Circuitul CPLD-ul prezintă caracteristicile specifice familiei și anume intrări tolerante la 5V, 3,3V și 2,5V (V_{IH}), ieșiri capabile să furnizeze 3,3V sau 2,5V (V_{OH}). Structura funcțională de bază a circuitului CPLD XC9572XL este prezentată în figura 4.14.

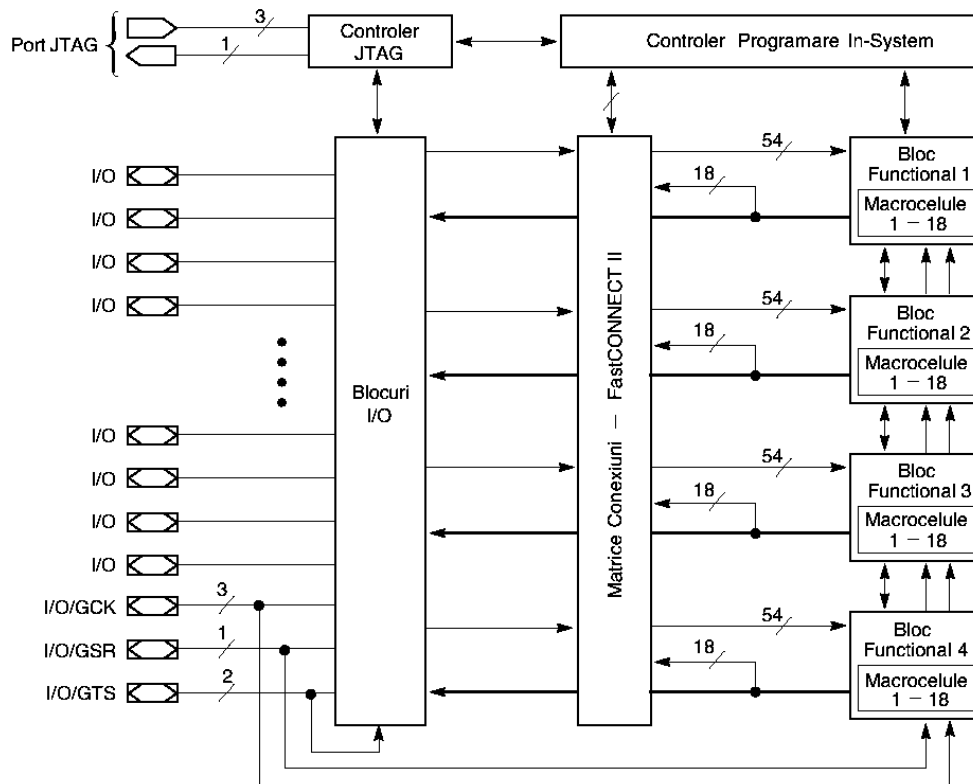


Figura 4.14 Structura circuitului XC9572XL

Puterea disipată poate varia substanțial în funcție de frecvența de lucru, aplicația implementată și încărcarea ieșirilor. Pentru a reduce consumul propriu, fiecare macro celulă dintr-un dispozitiv XC9500 (XL, XV) poate funcționa într-un mod de lucru cu putere scăzută (low power) dacă viteza de lucru nu este critică. Macro celulele neutilizate sunt implicit dezactivate, consumul lor fiind nesemnificativ.

Valorile maxime absolute admise pentru tensiunea de alimentare V_{CCINT} sunt între -0,5V și 4V față de masă. Pentru tensiunea de intrare V_{IN} valorile maxime admisibile sunt cuprinse -0,5 și 5V.

Valorile nominale de alimentare pentru logica internă $V_{CCINT} = 3V \dots 3,6V$, iar pentru circuitele (bufferele) de ieșire $V_{CCIO} = 2,3V \dots 3,6V$. Astfel circuitele de ieșire pot fi alimentate separat, de exemplu la tensiunea nominală de 2,5V. Indiferent de tensiunea de alimentare V_{CCIO} nivelele de intrare sunt $V_{ILmax} = 0,8V$ și $V_{IHmin} = 2V$ cu tensiunea de prag a intrărilor de cca 1.4 V (compatibilitate TTL). Nivelele logice de ieșire sunt $V_{OL} = 0$ și $V_{OH} = V_{CCIO}$ (ieșire CMOS, pentru curent de ieșire nul).

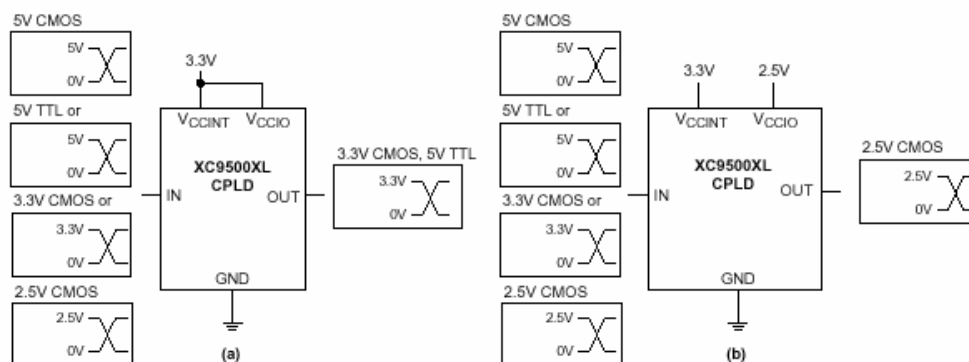


Figura 4.15 Interfețele I-O posibile pentru 9500XL

În figura 4.15 este prezentată interfața externă posibilă pentru un astfel de circuit, în situația în care (a) $V_{CCIO} = V_{CCINT} = 3.3V$ și respectiv (b) cu $V_{CCINT} = 3.3V$ și $V_{CCIO} = 2.5V$.

Configurarea (programarea) circuitului CPLD se face prin intermediul interfeței standard JTAG cu care se poate programa, șterge, sau citi (dacă nu a fost protejat la citire) conținutul datelor de configurare. Datele de configurare sunt descrise sub forma unui fișier standard JEDEC.

4.3 Aplicații ale circuitelor CPLD (Complex Programmable Logic Devices)

Circuitele CPLD utilizate și la care se vor face referiri în continuare sunt circuite ale firmei Xilinx și aparțin familiei **XC9500XL**. Mediul de programare la care se fac referiri particulare este **Xilinx WebPACK**.

Pentru dezvoltarea unei aplicații ce complexitate mică sau medie, cum sunt multe în domeniul mecatronic, în principiu sunt suficiente următoarele cunoștințe:

- definirea unui proiect CPLD asociat unei aplicații
- alegerea ca țintă a unui anumit tip de circuit CPLD (ca resurse și pin-out)
- editarea unei surse HDL folosind un editor HDL, detectarea și corectarea erorilor sintactice din sursa HDL
- sinteza logică a unui netlist din descrierea HDL
- plasarea și rutarea (fitting) netlist-ului obținut în circuitul CPLD țintă
- verificarea modului în care au fost utilizate resursele CPLD-ului
- utilizarea vectorilor de test pentru verificarea funcțională a aplicației
- generarea unui fișier JEDEC (implicit și a bitstream-ului) pentru CPLD
- încărcarea fișierului JEDEC (programarea circuitului CPLD)
- testarea aplicației (a circuitului CPLD programat)

Principalele probleme care **nu** vor fi abordate aici sunt:

- proiectarea logică complexă folosind un limbaj HDL (analiza timing, analiza puterii disipate)
- alegerea celui mai bun tip de CPLD pentru o anumită aplicație
- configurarea circuitului pentru cea mai eficientă utilizare a resurselor circuitului CPLD
- ce trebuie făcut dacă resursele logice ale CPLD-ului sunt insuficiente și fitter-ul nu poate plasa netlist-ul

Realizarea unui proiect (a unei aplicații, a unui circuit logic) cu ajutorul unui circuit CPLD presupune următorii pași (Design Flow):

- a. Descrierea circuitului logic folosind o schemă (Schematics) sau **un limbaj HDL** cum ar fi ABEL, VHDL sau Verilog. Etapa poartă numele și de **Design Entry**.
- b. Utilizarea unui program de sinteză logică pentru a ajunge de la sursa HDL la o descriere a porților logice utilizate și a modului în care acestea sunt conectate sau, cu alte cuvinte, pentru a compila sursa HDL într-un **netlist**. În continuare se poate face și o **verificare** a aplicației, prin simularea funcțională cu ajutorul vectorilor de test sau a unui test bench.
- c. Utilizarea unui program **fitter** pentru a plasa porțile logice și a ruta interconexiunile într-un anumit circuit CPLD țintă. Circuitul CPLD este compus din mai multe **blocuri funcționale** care la rândul lor sunt formate din **macrocelule**. Blocurile funcționale și macrocelulele sunt plasate în interiorul unor matrici de conectare- rutare (**routing matrices**). Fitter-ul asignează porțile din netlist diverselor macrocelule din blocurile funcționale și, în același timp, configurează matricile de conectare prin închiderea sau deschiderea elementelor de comutație (comutatoarelor) pentru a interconecta porțile conform conectivității descrise în netlist. După această etapă se poate face o verificare complexă a circuitului, luându-se în considerare și timpii de propagare- o analiza de timing (Timing Analysis).
- d. Odată faza de plasare terminată un alt program (generatorul fișierului de programare) va extrage starea elementelor de comutație din matricile de rutare și va genera o descriere sub forma unui șir de biți (**bitstream**). Într-o descriere simplificatoare biții în „0” corespund unui comutator deschis, iar cei în „1” unui comutator închis (în realitate lucrurile sunt mai complicate!). Bitstream-ul este incorporat unui **fișier JEDEC**, fișier conținând și alte informații, fișier a cărui formă și conținut este standard.
- e. Fișierul JEDEC poate fi acum **încărcat** (download) în circuitul fizic (conectat în circuit și alimentat!) cu ajutorul unui program dedicat și al unui dispozitiv de programare extern (**JTAG programmer**). Dispozitivul de programare se conectează cu circuitul CPLD prin intermediul interfeței JTAG standard și utilizează unul din porturile externe ale calculatorului gazdă (paralel, serial sau USB). După realizarea încărcării – programării, circuitul CPLD se va comporta conform descrierii din sursa HDL.

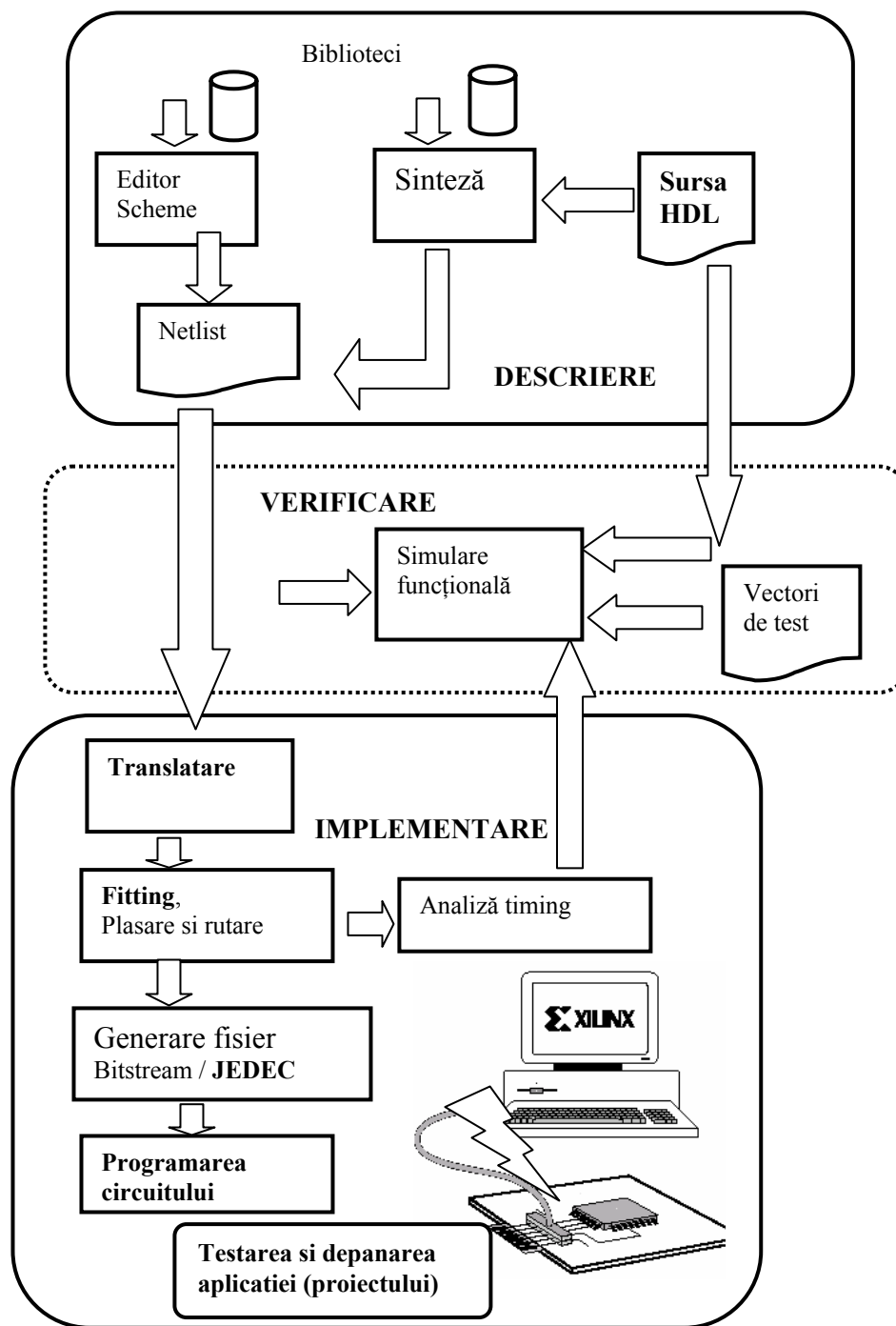


Figura 4.16 Etapele unui proiect CPLD

Configurarea (programarea) circuitelor CPLD Xilinx se face prin intermediul unei interfețe JTAG standard. Pentru aceasta se utilizează un modul de programare (JTAG programmer) cu o arhitectură deschisă, care se conectează la portul paralel PC și care este recunoscut de componenta de programare a mediului ISE. O schemă electrică a programatorului JTAG este prezentată în figura 4.17. Programatorul poate fi folosit atât pentru configurarea circuitelor CPLD cât și a circuitelor FPGA Xilinx. Bufferele utilizate sunt de tip 74HC125 care funcționează (ca de fapt toată seria 74HC) la tensiuni de alimentare de la 2V la 6 V (tensiunile nominale de 2.5V, 3,3V și 5V corespunzătoare seriilor 9500XV, 9500XL și 9500 sunt astfel acoperite). Procesul de programare verifică prezența modulului de programare, după care verifică prezența circuitului (sau a circuitelor) conectate la interfața JTAG. Cu ajutorul programatorului se pot programa, se pot șterge sau se pot citi datele de configurare. De asemenea se poate verifica dacă datele programate în circuit corespund cu fișierul de configurare, se poate citi un identificator al circuitului sau o sumă de control (checksum), se poate proteja circuitul la citire.

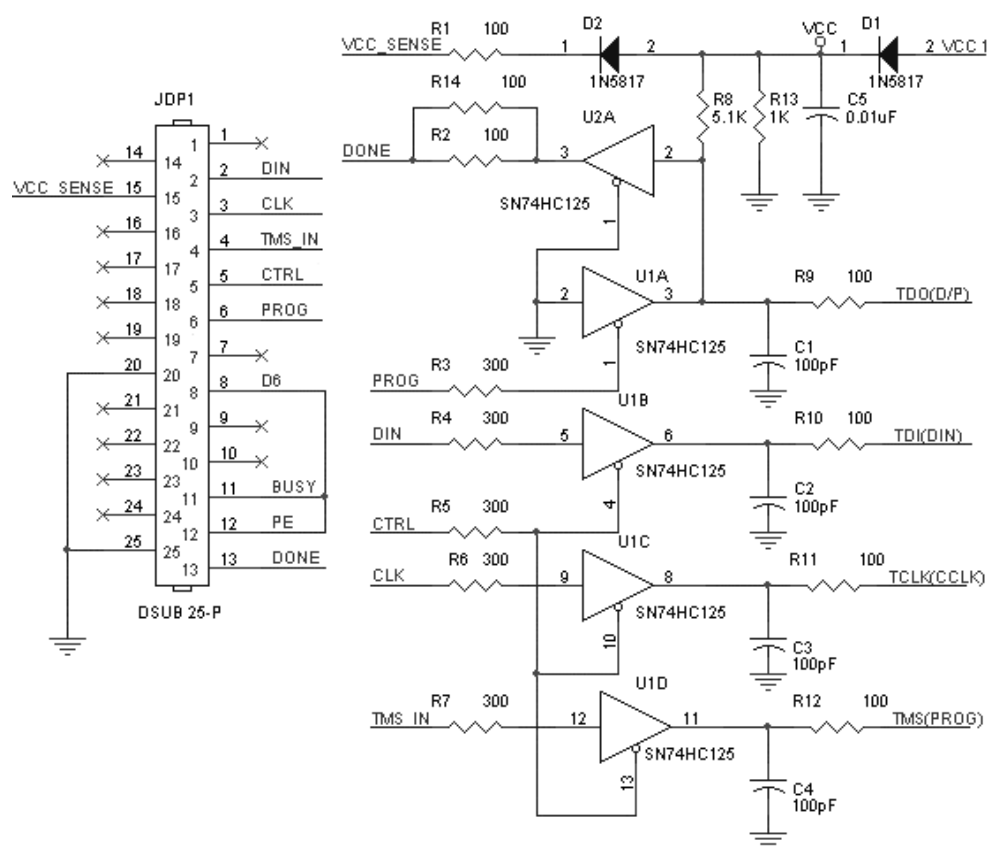


Figura 4.17 Schema programatorului JTAG Xilinx