

Sisteme cu dispozitive reconfigurabile

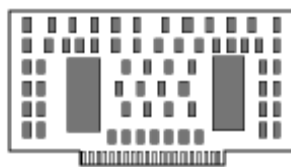
Arhitecturi de calcul

Dezvoltarea tehnologiei mijloacelor automate de calcul oferă mai multe soluții de implementare a sistemelor digitale:

1. Procesoare de uz general (GP), programate software;
2. Structuri hardware fixe (ASIC Application Specific Integrated Circuits) - circuite integrate specifice aplicației;
3. Circuite reconfigurabile (cele mai reprezentative fiind circuitele FPGA).

La începutul deceniului 9, al secolului trecut, cea mai mare parte a circuitelor logice, din sistemele numerice tipice, era realizată cu ajutorul unui număr relativ mic de circuite standard integrate pe scară largă (LSI): microprocesoare, controloare de magistrale, controloare de I/E, circuite de sincronizare etc.

Toate sistemele aveau încă nevoie de o logică “aleatoare”, atât pentru interconectarea circuitelor integrate pe scară largă, cât și pentru: generarea semnalelor de comandă globală (reset s.a.), formatarea datelor (serial/ paralel, paralel/serial, multiplexare) etc. Sistemele erau alcătuite dintr-un număr mic de componente LSI și din numeroase componente integrate pe scară mică (SSI) și medie (MSI).



Plachetă prevăzută cu componente LSI, MSI și SSI.

Pentru realizarea structurii de interconectare adesea se proiectau circuite la cerere (custom, ASIC), care conduceau la:

- reducerea complexității sistemului și a costurilor de fabricație, cât și la mărirea performanței;
- costuri ridicate de dezvoltare a circuitelor la cerere, creșterea timpului de proiectare și a timpului în care produsul ajungea pe piață.

În acest context apar două componente ale costurilor:

- costul de dezvoltare, denumit uneori: *non-recurring engineering* (NRE) și
- costurile de fabricație.

În realizarea sistemelor numerice, abordarea cu circuite la cerere este viabilă pentru produsele realizate în număr foarte mare, la care costurile de dezvoltare se pot amortiza și care nu sunt critice în raport cu timpul de lansare pe piață (TTM -Time To Market). FPGA-urile au fost introduse ca o alternativă la circuitele la cerere, pentru implementarea logicii de interconectare, ceea ce a permis mărirea densității de circuite de circa 10 ori în raport cu soluția SSI/MSI, reducerea costurilor de dezvoltare și scurtarea TTM. Cu ajutorul mijloacelor automate de proiectare (CAD), circuitele au putut fi implementate într-un timp foarte scurt, lipsind etapele de proiectare a măștilor și fabricare a circuitelor.

În conformitate cu legea lui Moore, densitatea (porți/suprafață) FPGA-urilor a crescut între anii 80-90, ai secolului 20, până la punctul în care funcțiile importante de prelucrare a datelor au fost implementate direct într-un singur circuit FPGA. FPGA este în continuare în competiție cu circuitele la cerere pentru funcții speciale de prelucrare și pentru logica de interconectare, dar este în competiție și cu microprocesoarele în aplicații dedicate și încorporate. FPGA-urile au avantajul performanței, în raport cu microprocesoarele, deoarece circuitele pot fi

adaptate ușor la aplicație. Microprocesoarele realizează funcțiile speciale în software, în condițiile operării în mai multe cicluri.

Comparație între soluțiile bazate pe ASIC, FPGA și procesoare pentru implementarea unui sistem numeric, referitor la: performanță, NRE, cost/unitate și TTM.

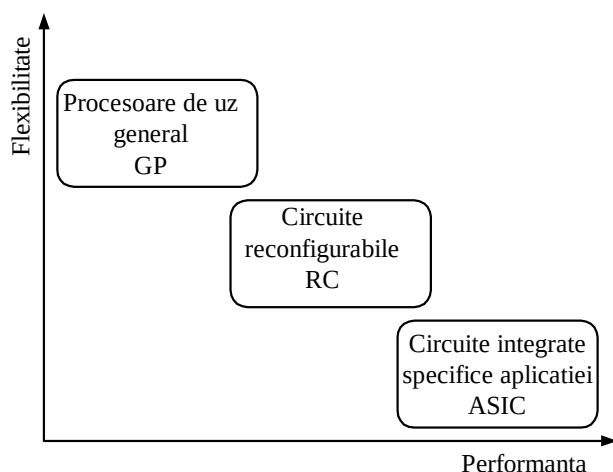
Performanță	NRE	Cost/unitate	TTM
ASIC	ASIC	FPGA	ASIC
FPGA	FPGA	Procesoare	FPGA
Procesoare	Procesoare	ASIC	Procesoare



Vom compara aceste arhitecturi și din punctul de vedere al criteriilor de flexibilitate și performanță.

Performanța determină capacitatea unui produs de a îndeplini o anumită sarcină. Există mai multe moduri de a măsura performanța unui produs, în general aceasta este exprimată în: MIPS (milioane de instrucțiuni per secundă), MMACS (milioane de multiplicări și acumulări per secundă) sau mai simplu în MHz (milioane de perioade de tact per secundă).

Flexibilitatea se referă la posibilitatea de a schimba caracteristicile unui produs pentru a îndeplini cerințe noi. Implementările software au cel mai scurt timp de lansare pe piață, datorită mării flexibilități și interfeței prietenoase a limbajelor de programare și a compilatoarelor. Acesta este motivul principal pentru care ele sunt mai prezente pe piață decât implementările hardware.



Procesoarele programate prin software

- Procesoarele (microprocesoare, procesoare digitale de semnal DSPs) furnizează resursele logice necesare implementării unor platforme de calcul **variabile**.
- Permit implementarea unor platforme de calcul foarte **flexibile**, capabile să execute aplicații din domenii diferite: comunicații, sisteme de comandă și control, prelucrarea semnalelor, etc.
- Aplicațiile sunt executate prin decodarea unui flux de instrucțiuni preluate dintr-un cod software, pe baza informațiilor stocate în blocuri de memorie.
- Caracterul secvențial de executare a aplicațiilor, limitele impuse de viteza de acces la blocurile de memorie și arhitectura fixă a procesoarelor, limitează performanța acestor sisteme.

Structurile hardware fixe (ASIC)

- ASIC-urile furnizează o soluție alternativă procesoarelor convenționale pentru implementarea unor aplicații performante.
- ASIC-urile sunt dispozitive hardware dedicate, proiectate pentru a implementa un număr foarte mic de aplicații sau chiar numai una singură.
- Pentru o aplicație dată, ASIC-urile obțin performanțe foarte bune, ocupă un spațiu mult mai mic și consumă mai puțină energie decât procesoarele.
- În cazul unor producții de serie mari, prețul de cost unitar devine foarte bun.
- Durata mare a ciclului de dezvoltare a circuitului.
- Prețul ridicat al proiectării circuitului integrat (50 000\$ - 1 000 000\$)
- Sunt excluse optimizări post-design.
- Sunt puțin flexibile.

Circuitele reconfigurabile (FPGA)

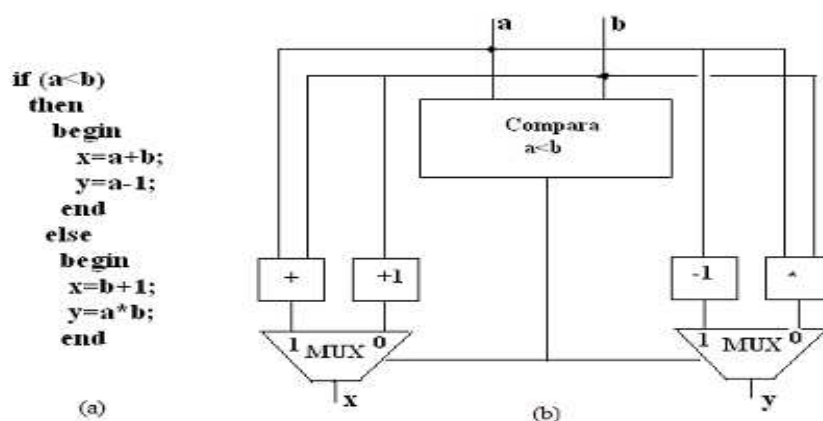
- Asigură un compromis între cele 2 tendințe: flexibilitate vs. performanță.
- Permit implementarea circuitelor specializate direct în hardware.
- Permit reconfigurarea, deoarece conțin resurse funcționale ce pot fi ușor modificate după implementarea în cadrul aplicației, ca urmare a schimbării parametrilor și datelor de lucru.
- Risc scăzut în faza de proiectare.
- Cost inițial redus.
- Timp de proiectare redus.
- Permit optimizări post-design.

Modalități de efectuare a calculului

Pornind de la posibilitățile de implementare a algoritmilor: prin software sau prin hardware (fix/reconfigurabil), se conturează două modalități de efectuare a calculului:

1. **Temporală** - executarea secvențială a algoritmului în baza instrucțiunilor din program, care necesită etapele de extragere, decodare și executare;
2. **Spațială** - executarea paralelă a algoritmului care se efectuează în baza unor blocuri funcționale care formează o structură hardware. Instrucțiunile, în acest caz, lipsesc. Distribuția algoritmului nu doar în timp dar și în spațiu permite atingerea unui grad înalt de paralelism.

Pentru exemplificare se presupune execuția următorului fragment de program pe un calculator convențional și într-o structură ASIC/FPGA (fig.1.1.).



(a) soluția temporală/software.

(b) soluția spațială.

Fragment de program pe un calculator vN (a) și într-o structură ASIC (b).

Timpul total de execuție pe un calculator vN este: $3 \cdot t_{\text{instrucțiune}}$, care poate cuprinde mai multe cicluri de ceas. Structura ASIC/FPGA efectuează calculele într-un interval de timp egal cu întârzierea cea mai mare în propagarea semnalului de la intrare la ieșire.

Circuite PLD

Circuitele logice programabile, cunoscute și sub forma acronimului PLD (Programmable Logic Device), sunt circuite integrate care conțin un număr mare de porți sau celule a căror interconexiune poate fi configurată sau “programată” pentru a implementa orice funcție combinațională sau secvențială dorită.

Pentru programarea circuitelor PLD se utilizează două tehnici:

- programarea prin măști, care se efectuează în timpul procesului de fabricație;
- programarea de către utilizator, pentru care se utilizează echipamente de programare cu costuri reduse.

Multe circuite PLD pot fi reprogramate de utilizator de multe ori, motiv pentru care ele sunt avantajoase pentru realizarea prototipurilor unui nou produs. Conexiunile programabile între elementele logice ale unui circuit PLD conțin comutatoare realizate de obicei cu tranzistoare sau antifuzibile (uneori fuzibile).

Firme producătoare de PLD: Xilinx, Altera, Lattice, Actel, Cypress, Atmel, QuickLogic.

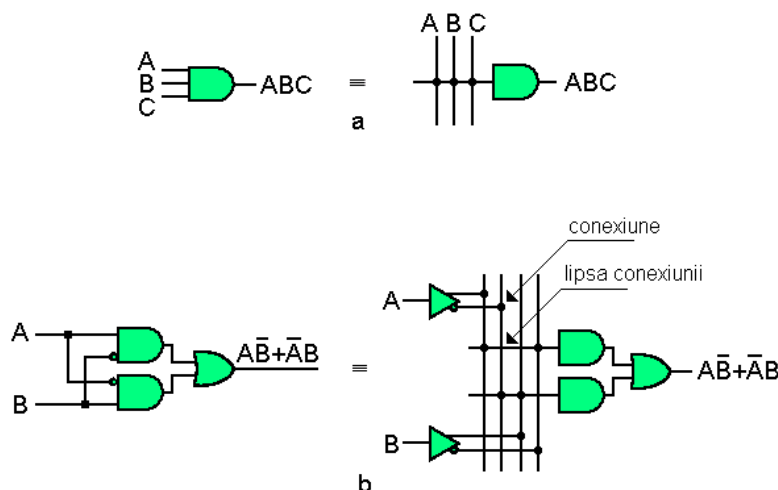
Există mai multe tipuri de circuite care sunt denumite în mod generic circuite logice programabile (PLD).

SPLD (Simple Programmable Logic Device)

CPLD (Complex Programmable Logic Device)

FPGA (Field Programmable Gate Array)

Porțile logice programabile ale unui circuit PLD pot fi reprezentate în mod simplificat: în locul unor linii de intrare multiple la fiecare din aceste porți s-a figurat o singură linie. Semnul $\times$ indică o conexiune programabilă a unei linii de intrare la o poartă logică. Absența semnului $\times$ indică faptul că respectiva conexiune a fost programată în starea deconectată.



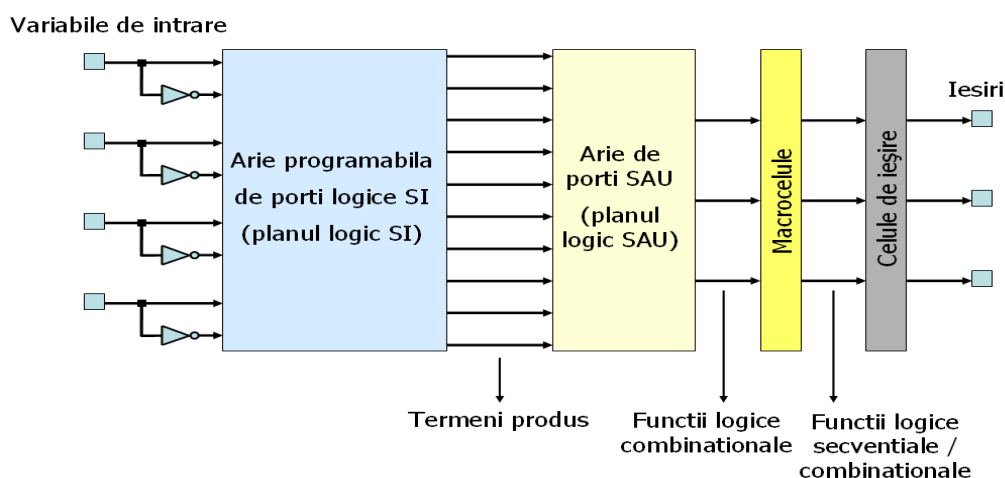
Circuite SPLD

Din punct de vedere arhitectural se clasifică în :

1. **PLA** – Programmable Logic Array.
2. **PAL/GAL** – Programmable Array Logic/Generic Array Logic

Ambele conțin arii de porți logice ȘI și SAU, conectate consecutiv, cu ajutorul cărora pot fi realizate formele disjunctive ale funcțiilor logice.

Arhitectura SPLD



Parametrii de bază:

n – numărul variabilelor logice

p – numărul termenilor produs

m – numărul funcțiilor logice realizate (numărul ieșirilor)

În planul logic ȘI variabilele intră atât în formă directă cât și în formă inversă. La ieșirile acestui plan se formează termenii produs. Acești termeni servesc drept intrări pentru planul logic SAU. În circuitele PLA acest plan este programabil, în circuitele PAL – este fix.

Complexitatea circuitului se determină din formula:

$$C = (2n + m)p$$

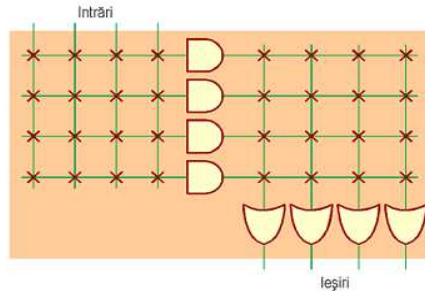
La intrările porților SAU din PLA poate fi aplicată orice combinație a termenilor produs, iar termenii pot fi utilizați de mai multe ori.

În cazul circuitelor PAL, la fiecare poartă SAU pot fi aplicați doar anumiți termeni produs. În cazul când este necesar de a utiliza același termen produs la altă poartă SAU, el este generat de mai multe ori. Dar programarea circuitelor PAL este mai simplă.

O rețea logică programabilă PLA (Programmable Logic Array) este similară ca și concept cu o memorie ROM, cu excepția faptului că nu realizează decodificarea completă a variabilelor și nu generează toți mintermii. Decodicatorul este înlocuit cu o rețea de porți ȘI care poate fi programată pentru a genera termenii produs ai variabilelor de intrare. Termenii produs sunt apoi conectați în mod selectiv cu porți SAU pentru a genera suma termenilor produs pentru funcțiile booleene necesare.

Un circuit PLA poate implementa în mod direct un set de funcții logice exprimate printr-un tabel de adevăr. Fiecare intrare pentru care valoarea funcției este adevărată necesită un termen produs, și acestuia îi corespunde o linie de porți ȘI din primul etaj al circuitului PLA. Fiecare ieșire corespunde la o linie de porți SAU din al doilea etaj al circuitului. Numărul de porți SAU corespunde cu numărul de intrări din tabelul de adevăr pentru care ieșirea este adevărată. Pentru proiectarea unui sistem digital cu un circuit PLA, nu este necesar să se indice conexiunile interne ale circuitului, ci trebuie să se specifice doar tabelul de programare.

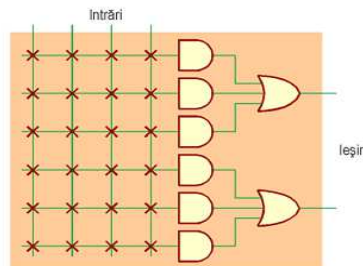
Structura de bază a unui circuit PLA:



Avantaj: flexibilitate

Dezavantaj: cost ridicat de fabricare, viteză de lucru redusă

Circuitele PAL (Programmable Array Logic) conțin o rețea de porți ȘI programabilă, și o rețea de porți SAU cu conexiuni fixe. Fiecare linie de ieșire este conectată la un set fix de linii ale rețelei de porți ȘI. O asemenea ieșire a circuitului PAL poate implementa o expresie pe două nivele conținând cel mult opt termeni. Avantajele circuitelor PAL sunt simplitatea utilizării în anumite aplicații și viteza mai ridicată. Aceste circuite sunt însă mai puțin flexibile decât circuitele PLA.



Avantaje: cost de fabricare mai redus decât PLA, viteza de lucru mai ridicată decât PLA

Dezavantaj: flexibilitate redusă

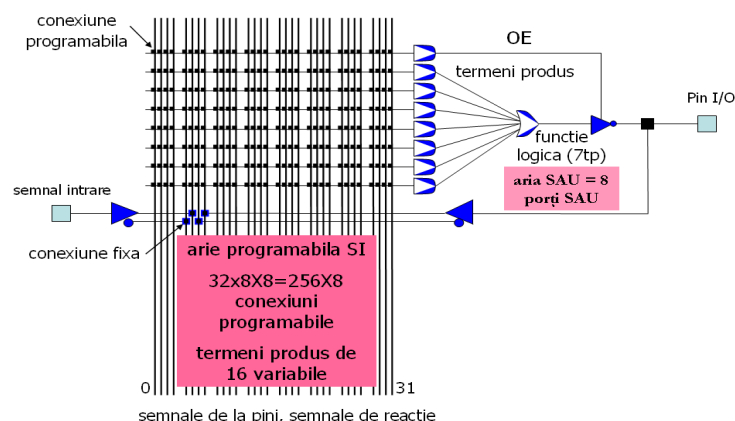
Există și circuite PLA sau PAL care conțin bistabile atașate prin conexiuni programabile la ieșirile rețelei de porți SAU, ceea ce permite implementarea unor circuite secvențiale de dimensiuni medii.

Unele circuite PAL conțin macrocelule, formate din bistabile, multiplexoare și porți logice.

Dispozitivul PAL16L8 (combinațional).

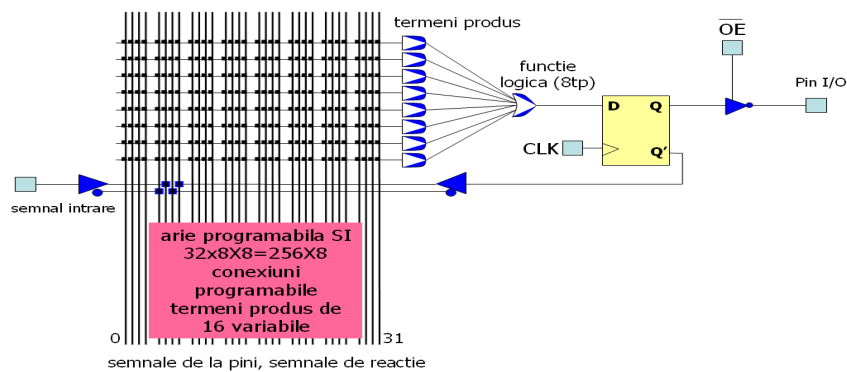
Rețeaua programabilă de porți ȘI conține 64 rânduri și 32 coloane ($64 \times 32 = 2048$ fuzibile). Fiecare din cele 64 porți ȘI are 32 intrări – 16 variabile în formă directă și inversă. =pt elemente ȘI sunt asociate cu fiecare pin de ieșire. Șapte dintre aceste porți sunt conectate la poarta fixă SAU. O poartă ȘI servește pentru validarea ieșirii (în cazul când primește valoarea „1” logic) și este conectată la un buffer cu trei stări. Circuitul conține 6 pini bidirecționali (13-18)

Structura dispozitivului PAL16L8

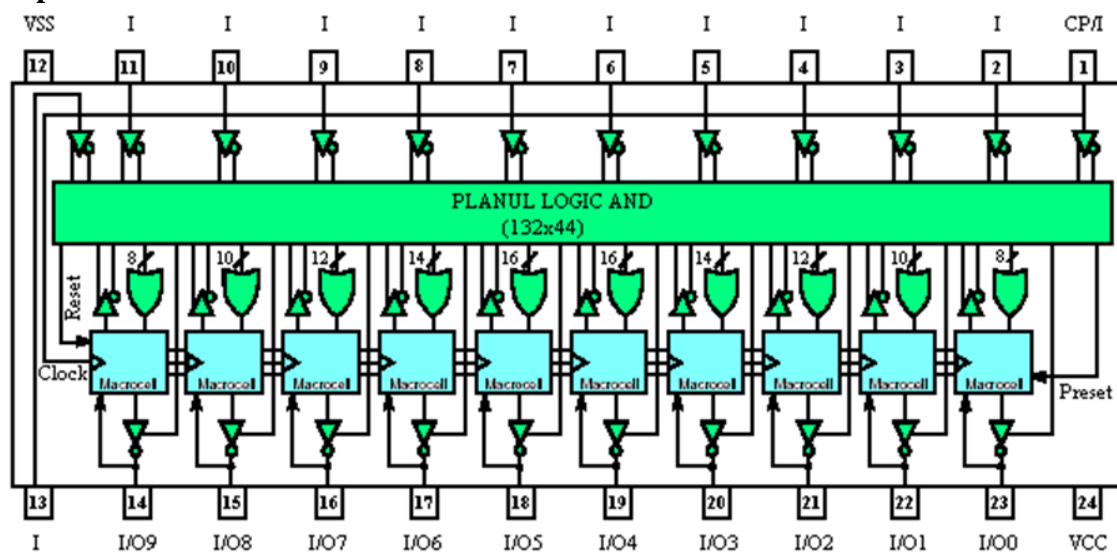


Dispozitivul PAL16R8 (secvențial)

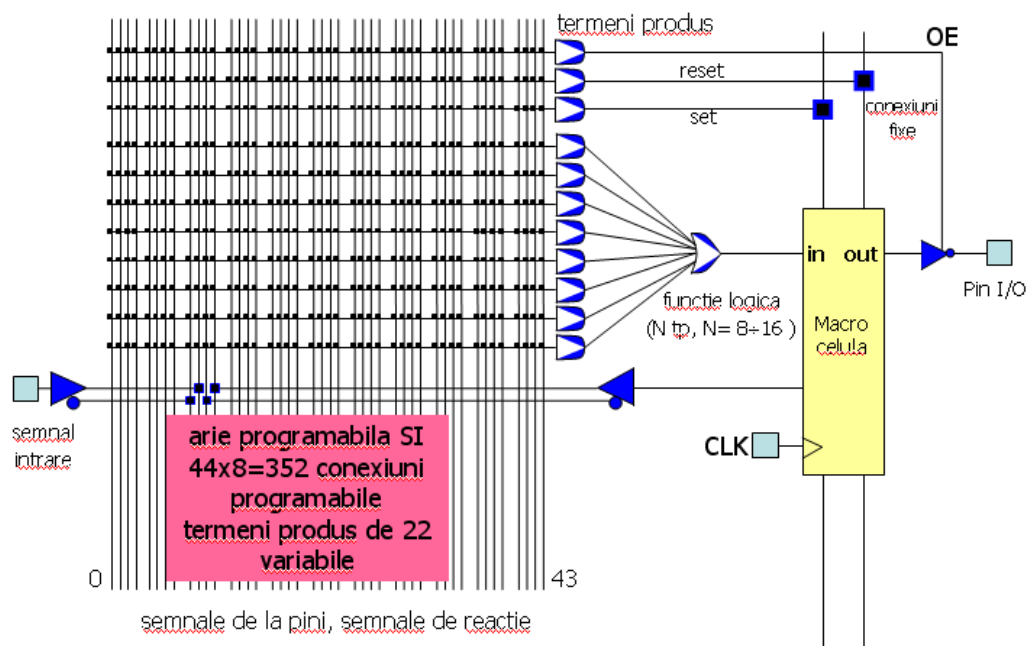
Structura dispozitivului PAL16R8



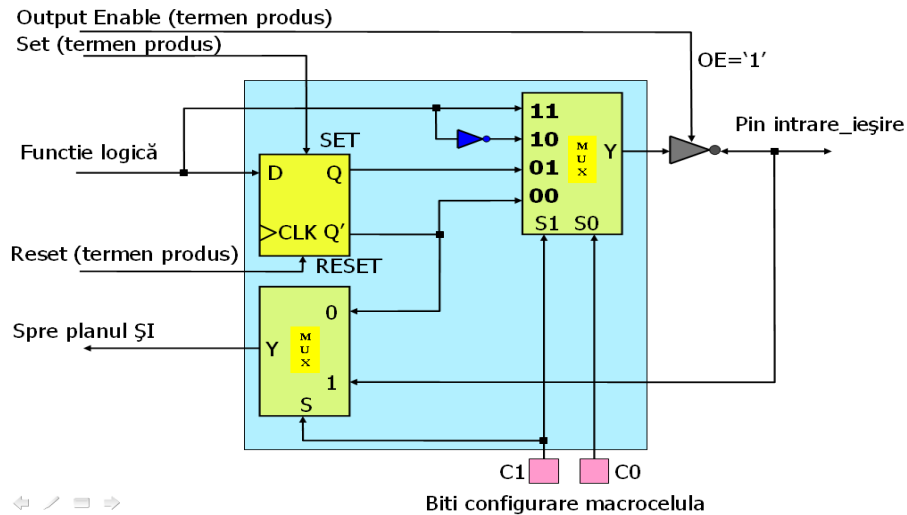
Dispozitivul PAL22V10



Structura dispozitivului PAL22V10



Structura macrocelulei PAL22V10

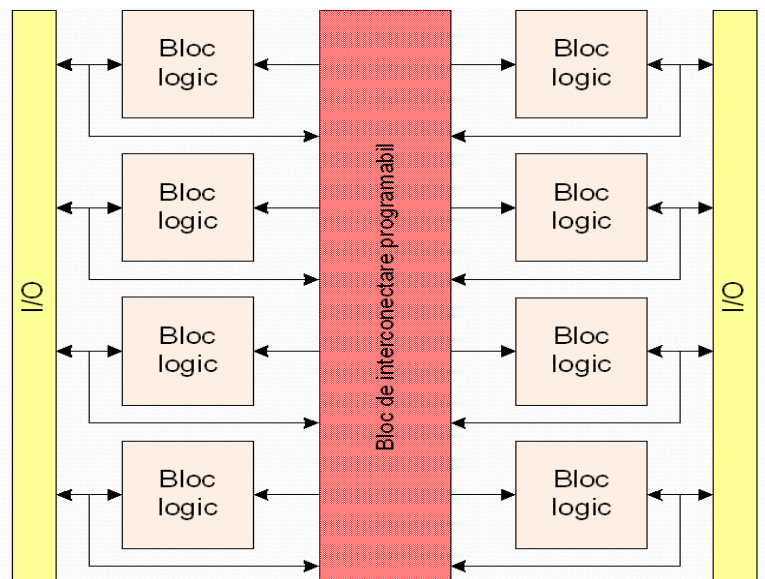


Circuite CPLD

Circuitele CPLD sunt circuite VLSI ale căror părți componente sunt:

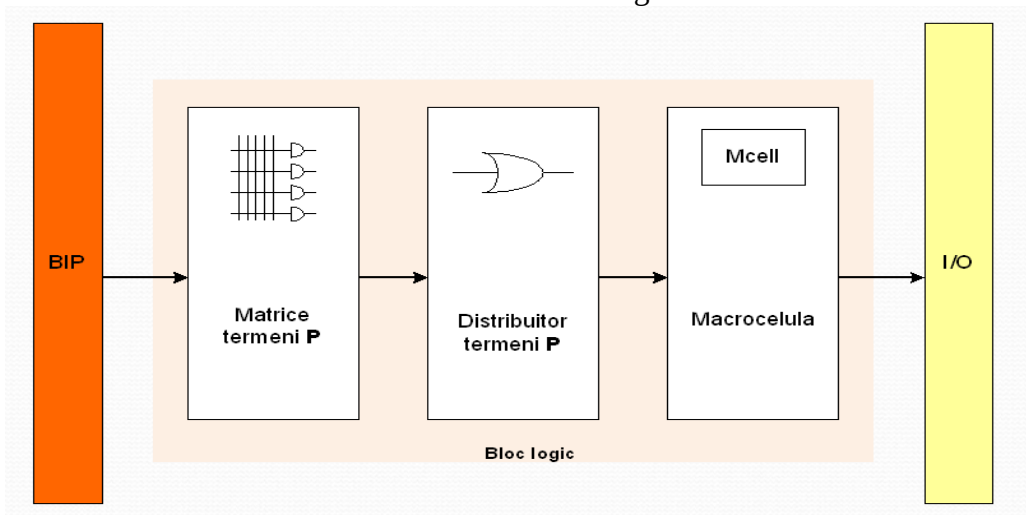
- PAL (GAL), care formează blocurile logice (funcționale), fiecare bloc fiind compus din mai multe macrocelule;
- Bloc de interconectare programabil;
- Blocuri de intrare/ieșire

Funcțiile logice simple pot fi implementate în cadrul unui singur bloc. Funcțiile mai complexe pot necesita mai multe blocuri, care vor fi interconectate prin matricea de rutare.



În circuitele CPLD, spre deosebire de circuitele PAL, matricea SAU nu este complet fixă și, prin intermediul distribuitorului termenilor produs permite de a forma funcții logice disjunctive, variind numărul termenilor, utilizând aceiași termeni pentru diferite funcții. Macrocelula reprezintă un bistabil care poate fi programat să lucreze în regim D sau T, multiplexează pentru alegerea modului de lucru (combinațional sau secvențial), elemente XOR pentru a obține funcția în forma directă sau inversă.

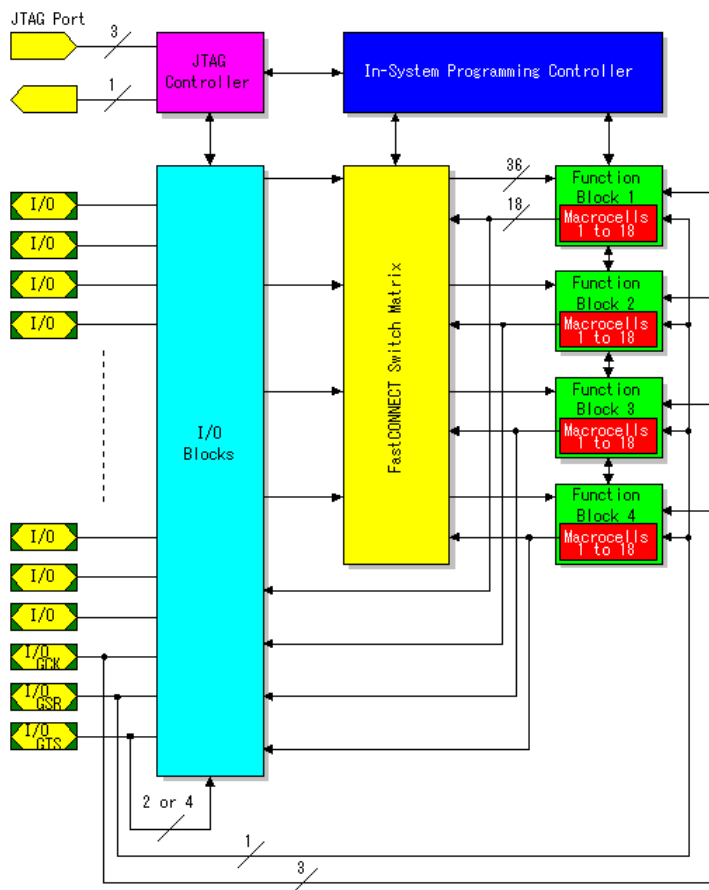
Structura blocului logic



Seria XC9500

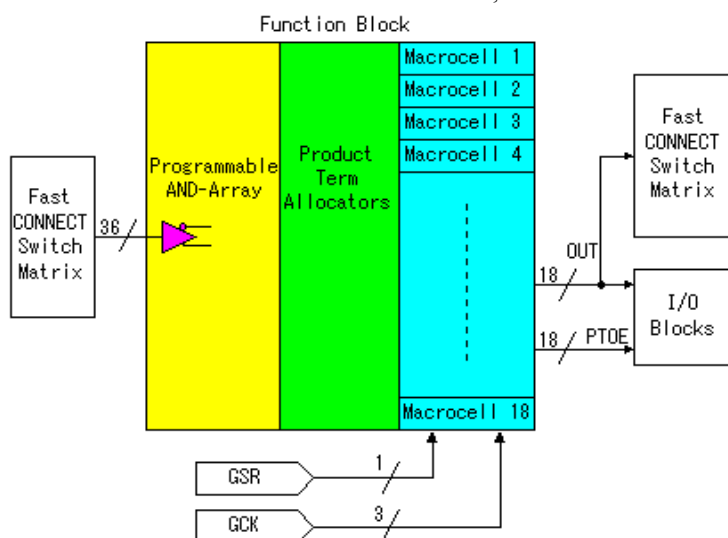
Circuitele înglobează un anumit număr de Blocuri Funcționale (Functional Block) și de Blocuri de Intrare/Ieșire (I/O Block) interconectate prin intermediul unei matrici de interconectare numită FastConnect. Fiecare Bloc Funcțional are maxim 36 de intrări și maxim 18 ieșiri. Fiecare Bloc Funcțional este 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.

Structura funcțională de bază a circuitului CPLD XC9572XL:



Fiecare bloc funcțional constă din aria programabilă de porți ȘI, distribuitorul de termeni și macrocelule. Macrocelula constă dintr-un bistabil de tip D/T. Semnalele de setare/resetare și ceas pentru bistabile sunt alocate de către distribuitorul de termeni. PTOE (product term output enable) este direcționat spre Blocul de intrare/ieșire, la fel, de către distribuitorul de termeni.

Structura unui bloc funcțional



Familiile CPLD 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.

Familiile reprezentative de circuite CPLD ale firmei Xilinx sunt: XC9500, XC9500XL, XC9500XV, Cool Runner II, Cool Runner XPLA3; ale firmei ALTERA: MAX7000, MAX3000A.

Circuite FPGA

Primul circuit *FPGA* a fost propus de Ross Freeman, unul din co-fondatorii firmei Xilinx Inc., în 1985. De atunci au fost elaborate diferite tipuri de circuite *FPGA* de un număr de alte companii ca Actel, Altera, Atmel, Texas Instruments etc. În comparație cu primele dispozitive programabile (*PLD*), care se deosebesc prin conexiuni rigide, *FPGA*-urile se bazează pe legături flexibile prin intermediul interconexiunilor programabile.

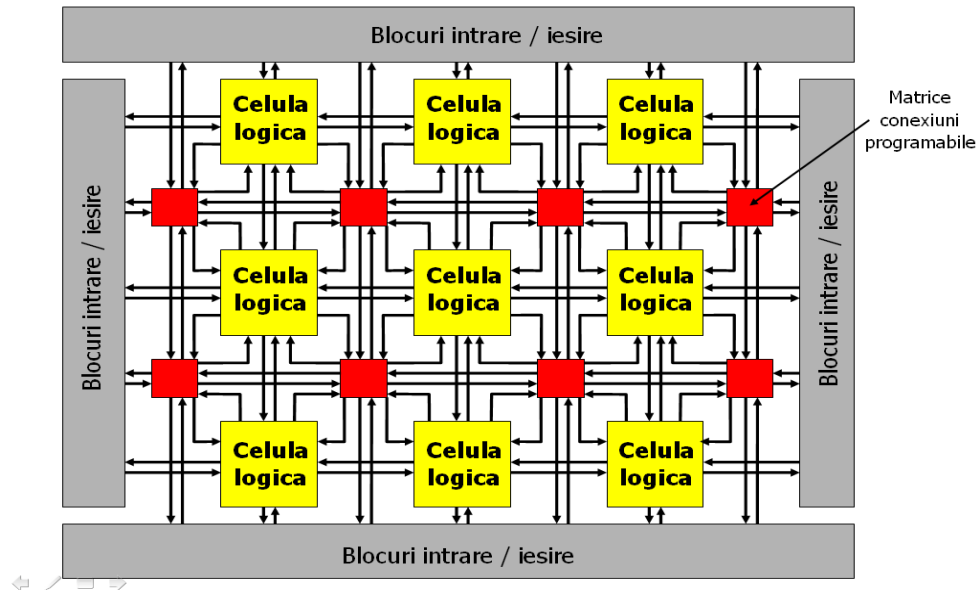
FPGA sunt circuite VLSI cu următoarele componente principale:

- celule logice (blocuri funcționale), amplasate în formă de matrice bidimensională;
- matrice de conexiuni programabile, amplasate în jurul fiecărui bloc funcțional;
- blocuri de intrare/ieșire.

Toate componentele FPGA sunt programabile (reconfigurabile) de către utilizator.

Fiecare celulă logică poate fi programată pentru a implementa orice funcție logică a intrărilor sale. De aceea, aceste blocuri sunt numite blocuri logice configurabile (*CLB* – *Configurable Logic Block*, Xilinx) Cele mai multe celule logice conțin de asemenea unul sau două bistabile. Fiecare celulă este înconjurată de interconexiuni programabile. Ansamblul acestor interconexiuni poartă numele de matrice de conexiuni programabile. Întreg ansamblul de celule și interconexiuni se află într-un inel format de blocurile de intrare / ieșire.

Arhitectura FPGA



Tehnologii de programare a circuitelor FPGA

În prezent sunt utilizate trei tehnologii de programare a circuitelor FPGA:

- Antifuzibile
- SRAM
- EPROM, EEPROM/FLASH

Circuite cu antifuzibile.

Un antifuzibil este un dispozitiv cu două terminale care în mod normal se află în starea de înaltă impedanță, iar atunci când este expus la o tensiune ridicată, trece în starea cu rezistență redusă (300-500 Ω). Antifuzibilele au dimensiuni reduse, astfel încât o arhitectură bazată pe antifuzibile poate conține sute de mii sau milioane de antifuzibile. Pentru simplificarea arhitecturii și a programării, circuitele FPGA bazate pe antifuzibile constau de obicei din rânduri de elemente logice configurabile cu canale de interconectare între ele, ca și rețelele de porți tradiționale.

Un bloc logic poate fi programat prin conectarea pinilor săi de intrare la valori fixe sau la rețele de interconectare. Există antifuzibile la fiecare punct de intersecție între interconexiuni și pini din canal și la toate punctele de intersecție între interconexiuni în locurile în care canalele se intersectează.

Din categoria circuitelor FPGA cu antifuzibile fac parte circuitele firmelor Actel, Quicklogic, Cypress.

Avantaje.

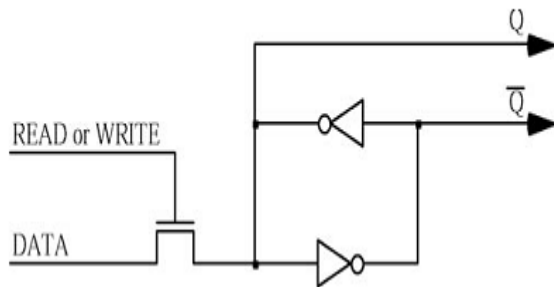
1. Sunt nevolatile
2. Timpul de rutare este mic și deci sunt mai rapide
3. Consum mai mic de putere
4. Securitate înaltă a datelor de configurare, deoarece circuitele sunt configurate o singură dată și informația de configurare nu este accesibilă concurenților.

Dezavantaje

1. Necesită un proces de fabricație nestandard complex.
 2. Necesită un programator extern și, o dată programate, nu mai pot fi reprogramate.
- Aceste dezavantaje au contribuit la dezvoltarea relativ lentă a acestor circuite.

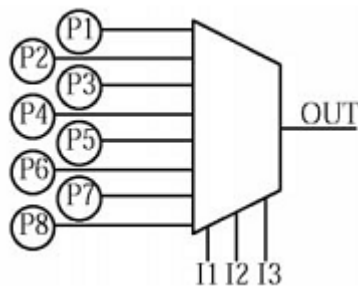
Circuite cu memorii SRAM

Programarea acestor circuite se realizează prin celule de memorie statică. O celulă SRAM dintr-un FPGA are structura:



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 (MOS Field-Effect-Transistors) este folosită atât pentru scriere cât și pentru citire. Pentru controlul căii de configurare sunt folosite semnalele Q și $\bar{Q}$.

Logica este implementată cu ajutorul unor tabele LUT (lookup table) realizate din celulele de memorie, intrările funcțiilor controlând liniile de adresă.



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.

Una sau mai multe tabele, combinate cu bistabile, formează un bloc logic configurabil. Aceste blocuri sunt aranjate într-un tablou bidimensional, segmentele de interconectare formând canale, similar cu rețelele de porți. Segmentele se conectează la pinii blocurilor logice din canale și la alte segmente din blocurile de comutare prin intermediul tranzistoarelor de trecere controlate de celule ale memoriei de configurare.

O secvență de configurare pentru circuitele cu memorii SRAM constă dintr-un singur cuvânt lung de programare. Logica din circuit încarcă cuvântul de programare, pe care îl citește serial dintr-o memorie externă de fiecare dată când circuitul este alimentat. Biții acestui cuvânt setează valorile tuturor celulelor memoriei de configurare din circuit, setând astfel valorile tabelor și selectând segmentele care se vor conecta între ele. Circuitele cu memorii SRAM sunt reprogramabile. Ele pot fi actualizate în sistem, punând la dispoziția proiectanților noi opțiuni și posibilități de proiectare.

Din această categorie de circuite FPGA fac parte cele ale firmelor Xilinx, Altera, AT&T.

Avantaje.

1. Necesită un proces de fabricație standard care permanent este optimizat.
2. Sunt reprogramabile, unele circuite și în timpul funcționării.

Dezavantaje

1. Sunt volatile și trebuie programate la fiecare conectare a tensiunii de alimentare a sistemului. 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.
2. Timpul de rutare este mare și deci sunt mai lente.
3. Consum mai mare de putere
4. Densitate de integrare mai mică
5. Au o securitate redusă a informației, deoarece ele trebuie configurate de fiecare dată la conectarea tensiunii de alimentare a sistemului iar codul de configurare poate fi citit.

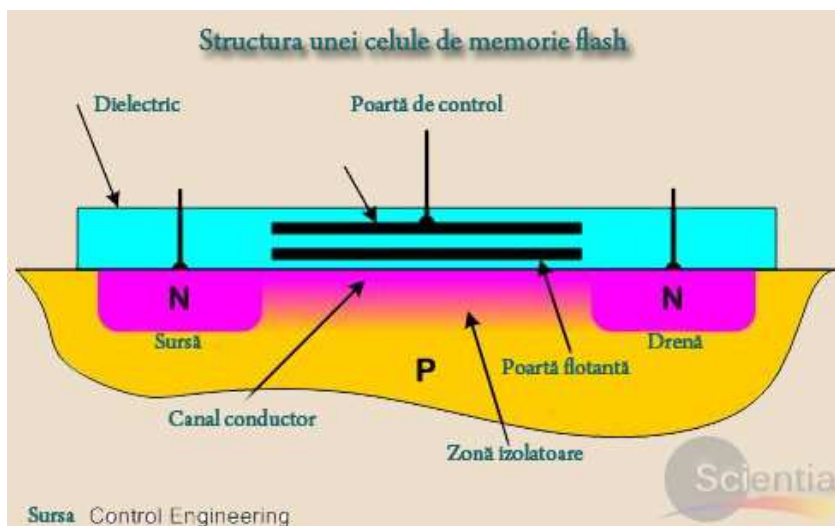
Totuși avantajele tehnologiei SRAM sunt mai importante decât neajunsurile și ele, de fapt, domină piața.

Circuite cu EEPROM/FLASH

Aceste circuite utilizează celule EEPROM/FLASH pentru fiecare element programat.

Memoria Flash este un tip de EEPROM (**E**lectrically **E**rasable **P**rogrammable **R**ead-**O**nly **M**emory), și este o memorie nevolatilă, ceea ce înseamnă că nu pierde datele stocate după ce este deconectată de la sursa de alimentare.

Memoria flash este formată dintr-o matrice de celule de memorie care, la rândul lor, sunt formate din perechi de tranzistori (*MOSFET - metal oxide semiconductor field effect transistor*) ce au între ei un strat subțire de oxid izolator. Un tranzistor este numit poartă flotantă (*floating gate*), iar celălalt - poartă de control (*control gate*). Atunci când este stabilită o legătură între cele două porți, celula de memorie are valoarea 1, schimbarea valorii în 0 are loc prin intermediul unui proces numit Fowler-Nordheim tunneling. **Efectul tunel** rezultă din capacitatea unui obiect cuantic de a străbate o barieră de potențial la scară atomică, fapt care ar fi imposibil după legile mecanicii clasice.



Elimina dificultățile de extragere din soclu și iradiere cu UV, aplicată la memoriile EPROM. Atât programarea cât și stergerea se fac electric. Prin perfecționarea tehnologiei și micșorarea grosimii stratului izolator al P_f există posibilitatea programării și stingerii electrice cu tensiuni mici aplicate între drenă și poartă. Polaritatea caderii de tensiune drenă-poartă este inversată la stergere față de programare.

Caracteristici:

- număr de stergeri și de reprogramări $> 10^4$;
- durata informației memorate mai mare de zece ani.
- se pot rescrie în timpul funcționării.

Din această categorie de circuite FPGA fac parte cele ale firmelor, Altera, Actel, Lattice.

Tehnologia EEPROM/FLASH combină avantajele tehnologiilor precedente.

1. Sunt nevolatile
2. Reprogramabile
3. Folosesc un proces de fabricație standard.
4. Consum redus de putere
5. Securitatea informației

FPGA seria XC4000 (Xilinx)

Blocurile componente

Structura internă programabilă de utilizator include trei elemente majore configurabile:

- **Blocurile logice configurabile (CLBs)**, care furnizează elementele funcționale și realizează structura logică proiectată;
- **Blocurile de intrare / ieșire (IOBs)**, care furnizează interfața între semnalele interne și exteriorul circuitului (legătură realizată fizic prin intermediul pinilor);
- **Matrici de comutatoare programabile** pentru interconectarea blocurilor.

Blocurile logice configurabile (CLB)

CLB-urile implementează majoritatea funcțiilor logice proiectate.

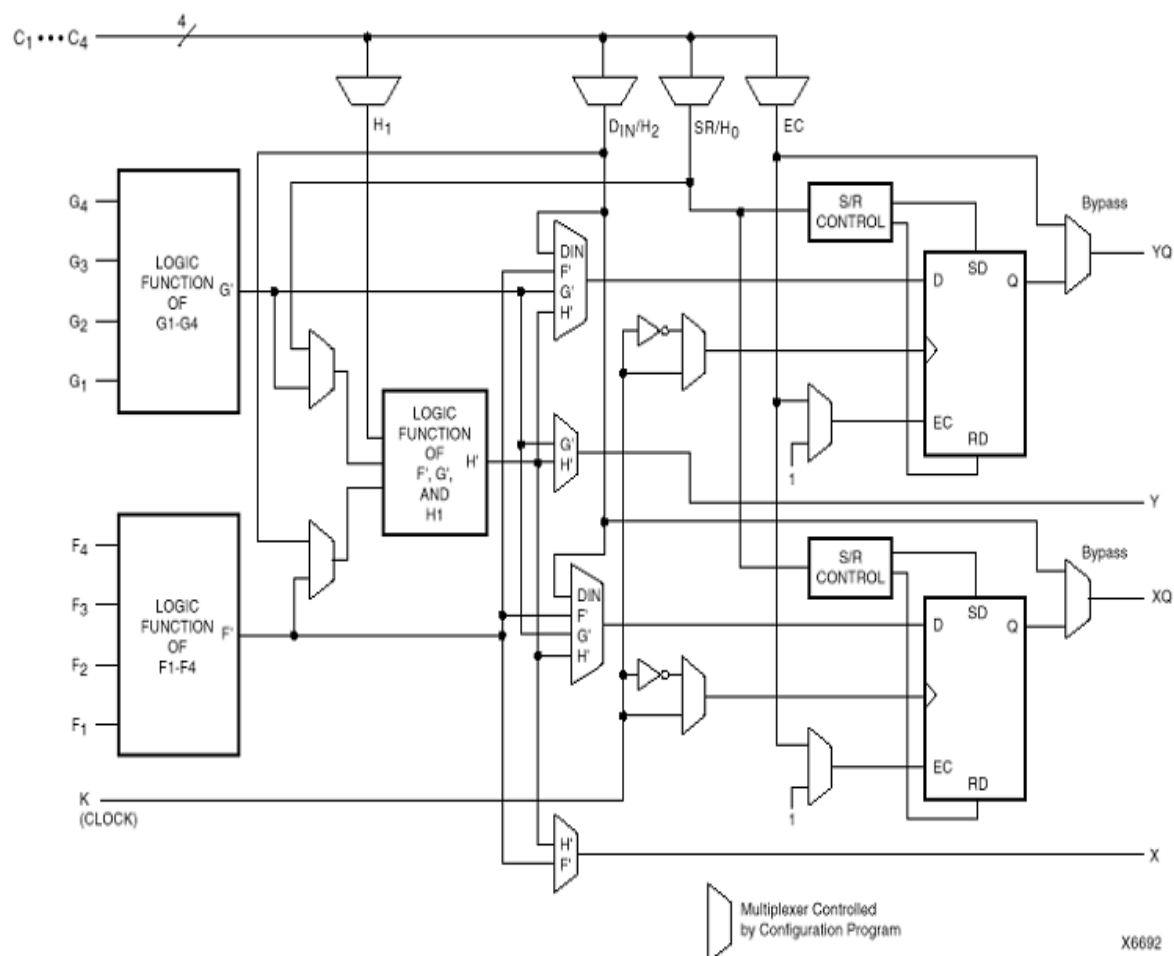
Elementele principale ale unui bloc configurabil sunt:

- F și G generatoare de funcții cu 2x4 intrări. H generator de funcții suplimentar ce posedă 3 intrări dintre care două provin de la generatoarele de funcții F și G, iar a treia provine din exterior de la unul din CLB-urile vecine. Astfel blocul configurabil poate implementa funcții logice cu până la 9 variabile de intrare. Prin implementarea funcțiilor logice, cu un număr mare de variabile într-un singur bloc, se reduce numărul de blocuri necesare pentru realizarea proiectului. De asemenea se reduc și timpii de propagare, astfel că va crește capacitatea de implementare și viteza de lucru. Generatoarele de funcții F' și G' se pot utiliza ca celule de memorie RAM/ROM.

- două elemente de stocare numite registre (bistabili D), care se pot utiliza pentru stocarea rezultatelor date de generatoarele de funcții.

Elementele de registru sau generatoarele de funcții se pot utiliza și independent. Intrările DIN, H1 sunt intrări directe de stocare. Ieșirile generatoarelor de funcții se pot utiliza ca ieșiri independente de ieșirile elementelor de stocare. Această flexibilitate mărește resursele logice și simplifică implementarea proiectelor. Astfel prin intermediul a 13 intrări și 4 ieșiri este asigurat accesul la fiecare bloc configurabil. Intrările și ieșirile respective legate la resursele programabile de interconectare vor realiza funcțiile logice.

Bistabilii interni pe lângă faptul că sunt ieșiri pentru rețeaua de interconexiuni, ieșirile combinaționale pot fi și sursă de intrare pentru bistabili interni ai CLB pentru realizarea circuitelor secvențiale. Bistabilii sunt de tip D și au tactul de înscriere pe front pozitiv prin semnalul comun de tact K. Validarea tactului se realizează prin semnalul EC (Enable Clock).

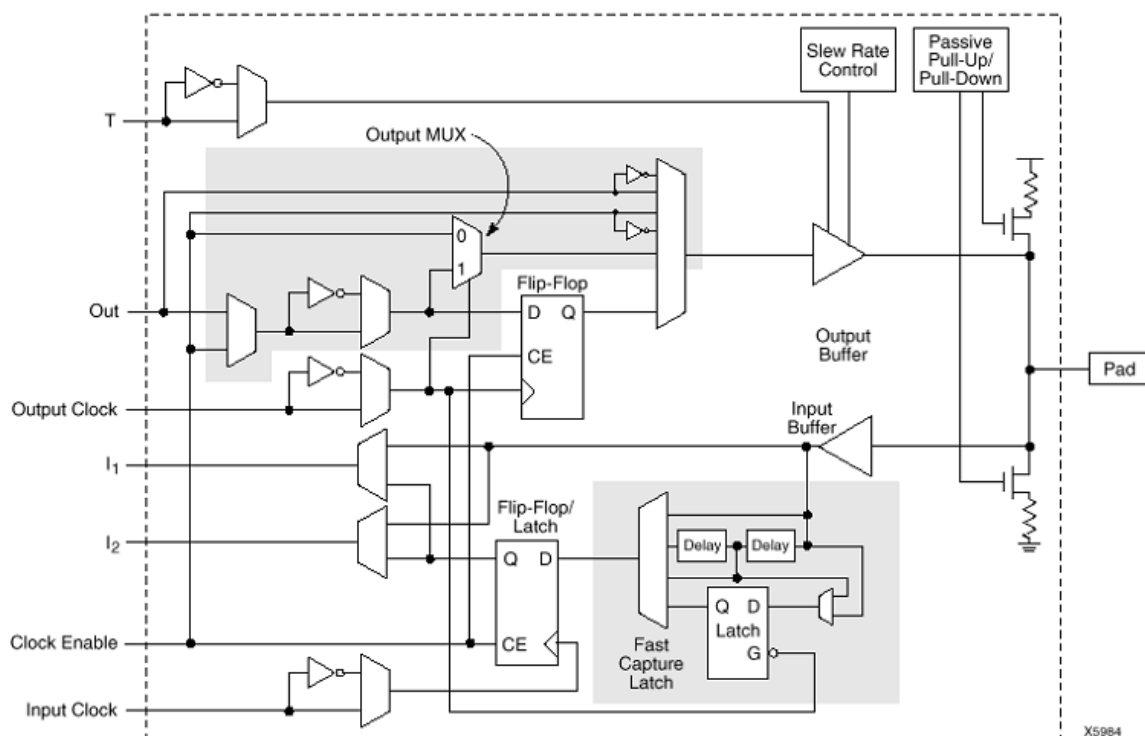


Circuitele XC4000 au caracteristici care permit integrarea unor sisteme complete. De exemplu, fiecare bloc CLB conține circuite care permit execuția eficientă a operațiilor aritmetice. Acestea implementează operații cu transport rapid pentru circuite de tip sumator. De asemenea, tabelele pot fi configurate ca celule RAM de tip R/W. Circuitele din seria XC4000E permit configurarea tabelor ca memorii RAM cu porturi duale, cu un singur port de scriere și două porturi de citire, existând posibilitatea ca blocurile RAM să fie sincrone. Fiecare circuit XC4000 conține planuri ȘI largi în jurul periferiei rețelei de blocuri logice pentru a facilita implementarea blocurilor de circuit cum sunt decodificatoarele de dimensiuni mari.

Blocurile de intrare / ieșire (IOBs)

Blocurile configurabile de intrare / ieșire realizează interfața între mediul exterior și structura internă a circuitului FPGA. Fiecare IOB controlează un pin (pad) al circuitului integrat. Blocurile de intrare ieșire se pot configura ca și port de intrare, port de ieșire sau port bidirecțional.

Diagrama simplificată a blocului IOB:



Căile I1 și I2 furnizează semnalele de intrare în IOB. Acestea sunt conectate la un bistabil al cărui tact de înscriere poate fi pe front sau pe nivel logic. Semnalele I1 și I2 se pot conecta la bistabilul de intrare al blocului IOB.

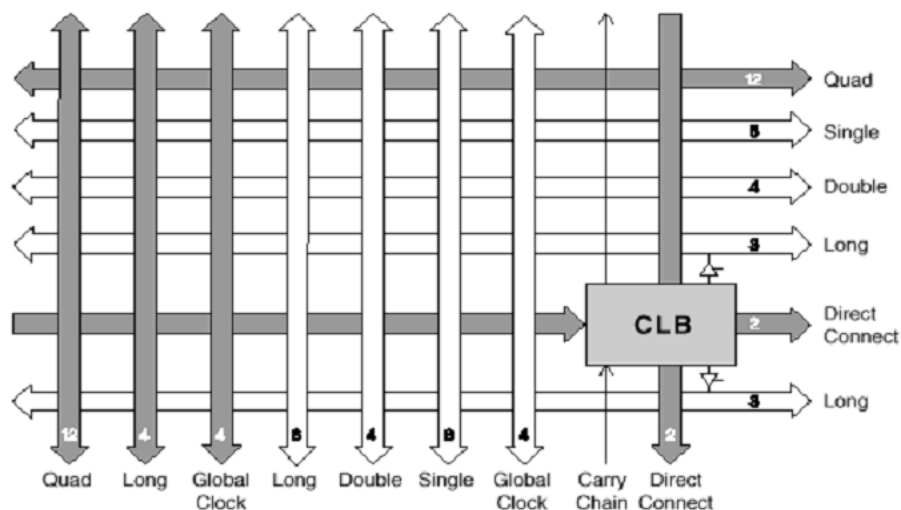
Semnalul de validare a bistabililor din IOB poate fi configurat astfel încât să fie comun sau separat pentru cele două registre. Acest semnal nu se poate inversa în interiorul IOB.

În mod opțional semnalele de ieșire se pot inversa în interiorul blocului de intrare / ieșire. Aceste semnale se pot conecta direct la ieșirea pinului sau la bistabilul de ieșire din IOB.

Slew Rate Control - Controlul vitezei de creștere a semnalului

Interconexiunile programabile

Toate conexiunile interne sunt compuse din segmente de metal cu puncte de cuplare programabile și matrice de cuplare pentru realizarea legăturilor interne.



Există următoarele tipuri de conexiuni:

- Realizarea legăturilor între blocurile CLB este asociată rândurilor și coloanelor matricei CLB;
- Realizarea legăturilor între blocurile IOB formează un cordon în jurul matricii CLB (VersaRing), acest cordon conectează pinii I/O cu blocurile logice interne;
- Conexiunile globale sunt compuse din rețele dedicate, proiectate pentru distribuirea rapidă a semnalelor de comandă și control utilizate în proiect.

Se disting 5 tipuri de linii de interconectare: linii de lungime simplă, linii de lungime dublă, linii de lungime quad-dublă, linii de lungime octală, linii lungi.

Întrările și ieșirile în blocurile configurabile sunt distribuite în toate cele patru direcții, pentru ca realizarea conexiunilor să fie cât mai flexibilă. Poziția intrărilor și ieșirilor CLB este interschimbabilă pentru evitarea congestiunilor, în timpul plasării și realizării conexiunilor.

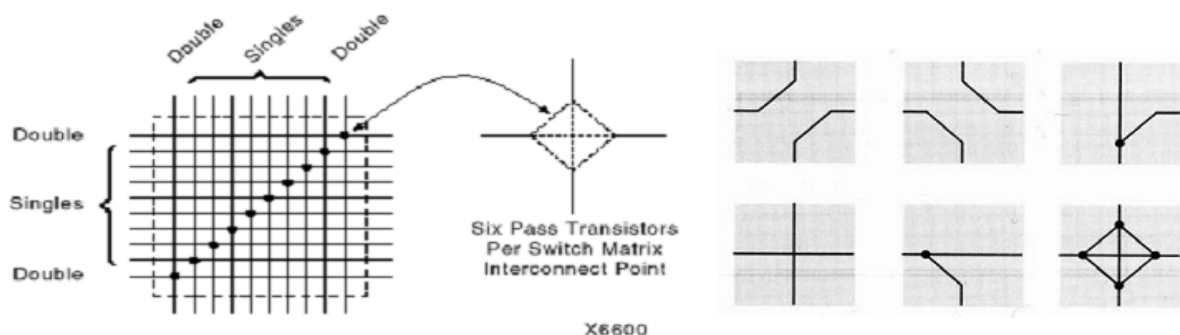
Liniile de simplă lungime permit interconectarea rapidă a blocurilor adiacente. Fiecărui bloc îi sunt asociate 8 linii verticale și 8 linii orizontale de simplă lungime. Aceste linii sunt conectate în matrice programabile situate la intersecțiile fiecărei linii și coloane. Liniile de simplă lungime introduc întârzieri în propagarea semnalelor ori de câte ori intră într-o matrice programabilă. Din această cauză nu sunt utilizabile pentru realizarea legăturilor la “distanță”. Ele sunt utilizate în mod normal pentru a ghida semnalele într-o arie restrânsă.

Liniile de dublă lungime realizează o grilă din segmente de metal. Fiecare linie are lungime dublă în comparație cu liniile simple. Liniile duble sunt grupate în perechi și intră în fiecare a doua matrice. Fiecărui CLB i se asociază patru linii verticale și patru linii orizontale, pentru realizarea conexiunilor.

Liniile lungi asemănător liniilor de dublă lungime formează o grilă de metal ce acoperă toată aria circuitului. Aceste linii cu un fun-out ridicat și timpi de propagare minimi pot conduce la semnale critice. La cele două linii orizontale fiecare CLB se poate conecta prin intermediul unui buffer 3-state. Astfel că aceste linii pot implementa bus-uri uni- sau bi-direcționale sau funcții cablate.

Matricea de cuplare programabilă. Liniile de simplă și dublă lungime verticale și orizontale se intersectează în așa numita matrice de cuplare programabilă (programmable switching matrix – PSM). Romburile de la intersecția a două linii reprezintă tranzistoarele de trecere utilizate pentru realizarea conexiunilor între linii

. Câteva posibilități de interconectare oferite de PSM:



Arhitecturi FPGA și CPLD combinate

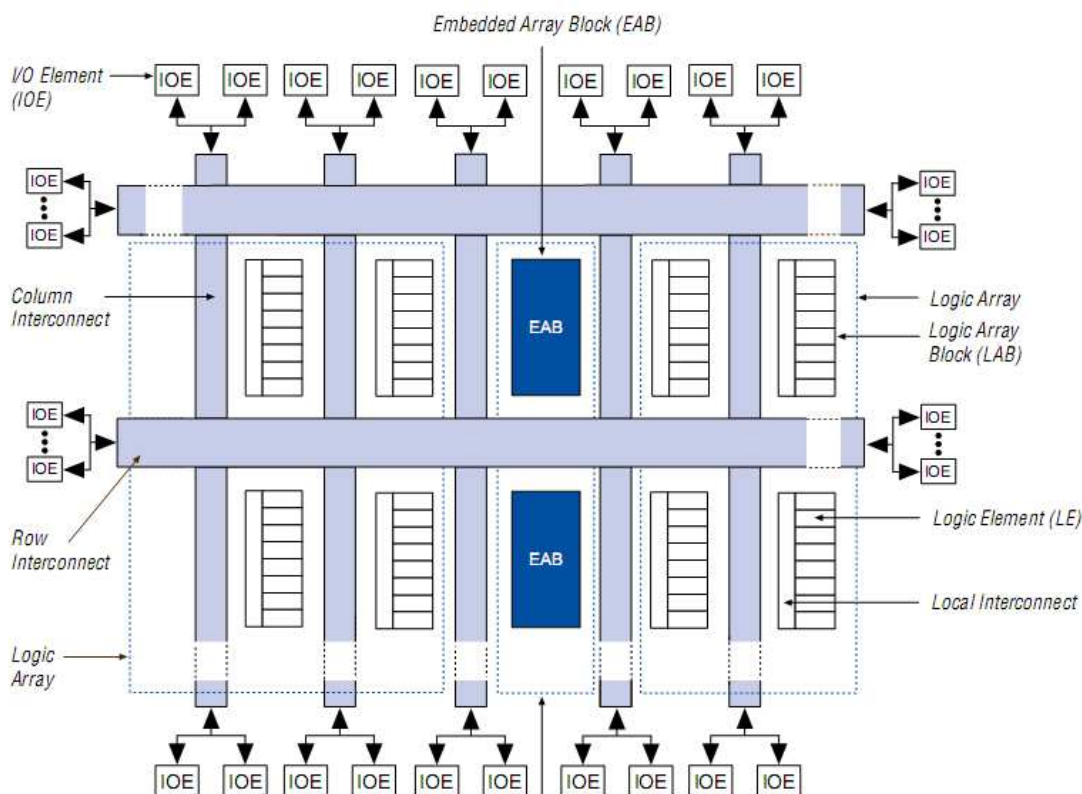
Tehnologia CPLD oferă performanțe asemănătoare tehnologiei FPGA pentru aplicații de complexitate medie, la un preț de cost mai scăzut. În același timp circuitele CPLD au o rețea de interconexiuni de lungime fixă (o proprietate a matricii de interconectare) ceea ce face ca sistemele numerice dezvoltate cu un circuit CPLD să prezinte întârzieri predictibile în totalitate

chiar din faza de proiectare. 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).

Odată cu creșterea nivelului de integrare a circuitelor au apărut arhitecturi care combină avantajele CPLD și FPGA. Un exemplu este familia de circuite FLEX (Flexible Logic Element matriX), firma ALTERA. Aceste arhitecturi sunt utilizate și în circuitele integrate de tipul „sistem pe chip” – SOPC (system on programmable chip).

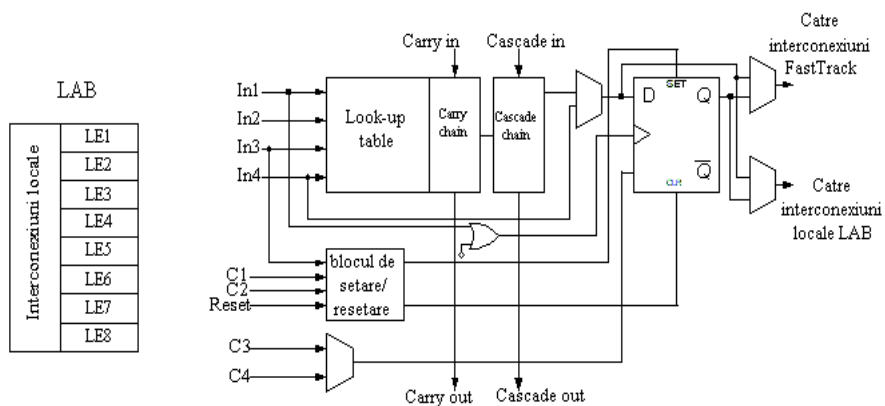
Schema de structură a circuitului FLEX10K:



Arhitectura generală a circuitelor Altera, care se bazează pe tehnologia de programare EPROM constă dintr-o rețea de celule programabile, numite blocuri ale rețelei logice (Logic Array Block - LAB), interconectate printr-o resursă de rutare numită rețea de interconectare programabilă (Programmable Interconnect Array - PIA).

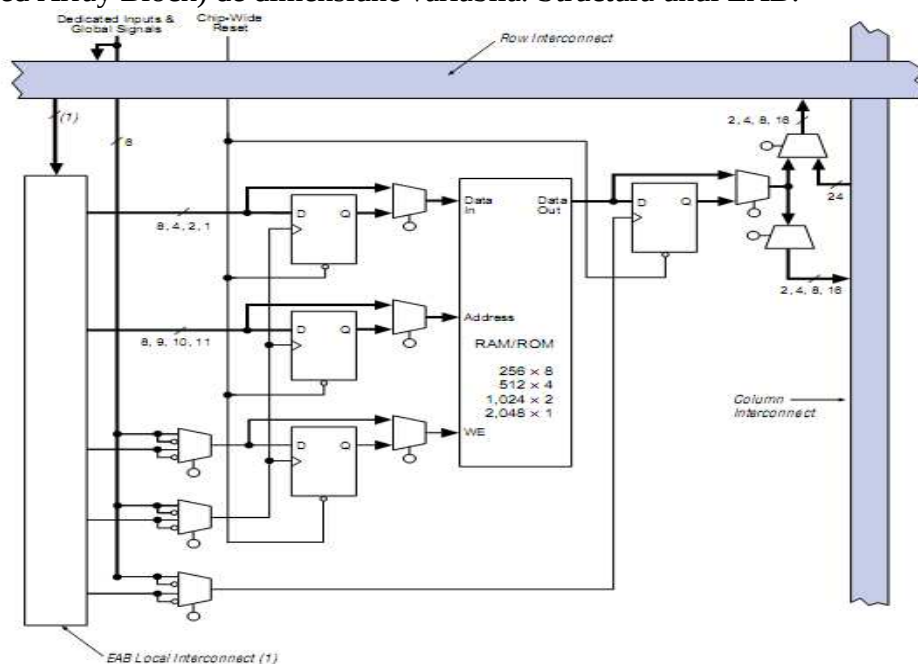
Interconexiunile sunt de două tipuri: locale (pentru conectarea elementelor logice din LAB) și globale (pentru conectarea LAB-urilor).

Un bloc LAB (Logic Array Blocks) al circuitului Flex10K constă din 8 elemente logice (LE) și interconexiuni locale prin care un element logic poate fi conectat cu oricare alt element logic din cadrul aceluiași bloc. Fiecare element logic constă dintr-o tabelă LUT cu patru intrări, care reprezintă un generator de funcții cu patru variabile, un bistabil programabil cu intrare de validare sincronă și două tipuri de interconexiuni dedicate pentru fluxul de date: lanțul de transport și lanțul de cascaderare.



Interconexiunile locale sunt legate la interconexiunile globale FastTrack ale circuitului. Ca și liniile lungi ale circuitului XC4000, fiecare conexiune FastTrack se extinde pe toată înălțimea sau lățimea circuitului. O diferență majoră între circuitele Flex10K și FPGA este însă că FastTrack conține numai linii lungi (tipic pentru CPLD), ceea ce permite configurarea simplă a circuitului. Toate liniile orizontale FastTrack sunt identice. De aceea, întârzierile de interconectare ale circuitului sunt mai predictibile decât cele ale altor circuite FPGA care utilizează segmente mai scurte, deoarece căile mai lungi conțin mai puține comutatoare programabile. Mai mult, conexiunile între liniile orizontale și verticale trec prin buffere active, îmbunătățind în plus predictibilitatea.

Familia de circuite Flex 10K dispune în plus de blocuri SRAM încorporate (EAB – Embedded Array Block) de dimensiune variabilă. Structura unui EAB:



Blocurile SRAM pot fi configurate pentru diferite aranjamente: 256×8, 512×4, 1K×2, sau 2K×1. Aceste blocuri pot fi configurate și pentru implementarea unui circuit logic complex, ca de exemplu un circuit de înmulțire.

Blocurile SRAM, în afară de modulul de memorie mai conțin câteva bistabile D sincrone și multiplexoare programabile. Interconexiunea locală EAB primește 22-26 semnale de la rândul de interconexiune globală. Înscrierea datelor în modulul de memorie poate fi sincronă – de la bistabili, și asincronă, direct de la interconexiunea locală. Semnalele de ieșire ale modulului de memorie pot fi transmise fie la rândul, fie la coloana de interconexiuni globale sincron – de la ieșirea bistabilului, asincron – direct de la modul.

Sisteme pe circuite integrate programabile - SOPC (System on programmable Chip)

Costul redus, performanța ridicată, densitatea mare a circuitelor programabile a făcut posibilă apariția circuitelor integrate, numite SOPC.

SOPC pot fi de 2 tipuri:

- cu structură uniformă și posibilitate de reconfigurare a tuturor unităților din sistem (nuclee programabile - soft cores);
- cu unități fixe în care sunt realizate anumite funcții și unități programabile (nuclee hard – hard cores). Inițial, nucleele hard erau destul de simple, în prezent ele reprezintă microprocesoare sau microcontrolere.

În SOPC cu nuclee programabile este posibilă realizarea părților componente a procesoarelor, memoriilor, dispozitivelor periferice. Resursele de proiectare permit formarea pe același circuit a componentelor virtuale – nucleele soft, IP (Intellectual Property) a diferitor producători. Dar în aceste sisteme nu se atinge viteza maximă a nucleelelor.

În SOPC cu nuclee hard, unitățile predefinite ocupă un spațiu de câteva ori mai mic, în comparație cu nuclee soft, deoarece ele nu conțin interconexiuni programabile și sunt optimizate pentru îndeplinirea funcțiilor concrete. În același timp, se pierde flexibilitatea funcțională. Nucleele hard sunt fixate pe suprafața circuitului și aceasta poate crea dificultăți în cazul plasării și rutării componentelor programabile, neatingându-se performanța maximă.

În prezent se dezvoltă pe larg ambele tipuri de SOPC. De exemplu, firma Altera utilizează nucleul programabil Nios pe circuitele APEX20KE și APEXII. Nivelul de integrare a acestor circuite este atât de înalt, că nucleul ocupă un spațiu foarte mic. Firma Xilinx a elaborat nucleul programabil Microblaze pentru familia VirtexII, care lucrează pe o frecvență de 125MHz.

Nucleele hard se bazează pe arhitectura RISC a procesoarelor firmelor ARM Limited, MIPS Technologies și IBM Microelectronics, lucrează pe o frecvență de peste 200 MHz, au inclus principiul pipeline, îndeplinesc operații scalare au un consum de putere redus și ocupă 2-3 mm². Arhitecturile standarde ale acestor procesoare permit utilizarea diverselor resurse CAD ce reduce complexitatea de proiectare și micșorează timpul de lansare pe piață.

Exemple de SOPC cu nuclee programabile:

Familia APEX20K, Altera;

Familia Virtex, Xilinx;

Familia ProASIC, Actel;

Familia Delta39K, Cypress Semiconductor.

Exemple de SOPC cu nuclee hard fără procesoare încorporate:

Familia ESP (Embedded Standard Products), firma QuickLogic, cu următoarele subfamilii: QuickRAM, QuickPCI, quickPC, QuickDSP, QuickSD. Părțile programabile a acestor circuite sunt realizate cu antifuzibile și au următoarele proprietăți: o singură programare, volum mai mic a interconexiunilor, viteză de lucru înaltă, sensibilitate mai mică la temperatură și radiație.

Exemple de SOPC cu nuclee hard cu procesoare încorporate:

1. FPSLIC (Field Programmable System-Level Integration Chip), Atmel, cu următoarele blocuri:

- microcontroler AVR și unitățile periferice;
- memorie SRAM;
- FPGA AT40K

2. Virtex4, Xilinx. Componente:

- procesoare PowerPC
- controlerul Ethernet MAC (Media Access Control) cu trei moduri de operare (10/100/1000 Mb/s)

- traneivere seriale cu viteza cuprinsă între 622 Mb/s și 11,1 Gb/s
- slice-uri dedicate procesării digitale a semnalelor DSP (XtremeDSP Slice) funcționând la 500 MHz. Acestea conțin:
- multiplicatoare pe 18x18 biți, dedicate,
- nivele de pipeline opționale pentru îmbunătățirea performanțelor,
- blocuri multiplicatoare-acumulatori sau multiplicatoare-sumatoare,

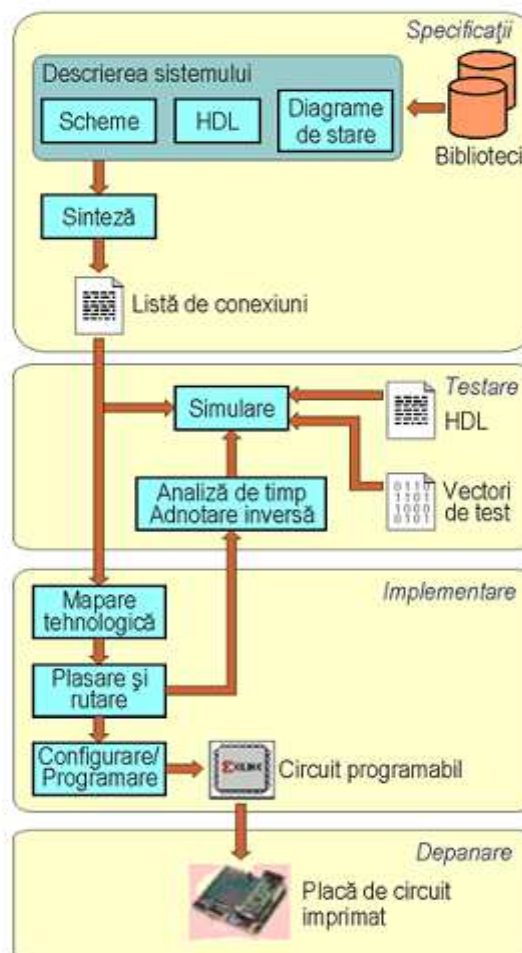
Procesul de proiectare cu circuite programabile

Etapele de proiectare

Pentru proiectarea sistemelor digitale utilizând circuite programabile, cum sunt circuitele FPGA și CPLD, se utilizează pachete de programe de proiectare asistată de calculator (CAD – Computer Aided Design). Aceste pachete de programe asistă proiectantul în toate etapele procesului de proiectare. Astfel, majoritatea pachetelor CAD pentru circuitele programabile asigură următoarele funcții principale:

- Specificarea (descrierea) sistemului digital;
- Sinteza descrierii, deci transformarea acesteia într-o listă de conexiuni conținând porți elementare și interconexiunile dintre ele;
- Simularea funcționării sistemului pe baza listei de conexiuni obținute, înainte de implementarea într-un anumit circuit;
- Implementarea sistemului într-un circuit prin adaptarea listei de conexiuni pentru a se utiliza în mod eficient resursele disponibile ale circuitului;
- Configurarea (programarea) circuitului pentru ca acesta să realizeze funcția dorită.

Etapele din cadrul procesului de proiectare a sistemelor digitale utilizând circuite programabile:



Descrierea sistemului

Principalele metode pentru descrierea sistemelor digitale:

- prin scheme logice,
- prin limbaje de descriere hardware (HDL – Hardware Description Language),
- prin diagrame de stare.

În mod tradițional, sistemele digitale sunt descrise prin scheme logice. Pentru aceasta se utilizează un editor schematic, care permite specificarea componentelor care trebuie utilizate și a modului în care acestea trebuie interconectate.

Pe lângă schemele logice, o altă posibilitate pentru descrierea sistemelor digitale este cu ajutorul limbajelor de descriere hardware. Aceste limbaje sunt din ce în ce mai utilizate, fiind preferate pentru descrierea sistemelor cu complexitate mai ridicată, datorită următoarelor avantaje principale:

- Posibilitatea descrierii funcționale a sistemelor, aceasta fiind o descriere la un nivel mai înalt, fără detalierea structurii la nivelul componentelor simple sau a porților elementare. Astfel, timpul necesar pentru descrierea sistemelor complexe se reduce în mod semnificativ.

- Independența descrierilor HDL față de diferitele tipuri de circuite. În timp ce schemele logice sunt realizate cu componente de bibliotecă specifice unei anumite familii de circuite, descrierile HDL sunt complet independente de un anumit circuit, astfel încât aceeași descriere se poate utiliza pentru implementarea sistemului într-un anumit circuit FPGA, dar și într-un alt tip de circuit programabil, de exemplu, într-o rețea logică programabilă.

- Posibilitatea modificării mai simple a descrierii HDL a unui sistem, datorită faptului că o asemenea descriere reprezintă în același timp o documentare a sistemului.

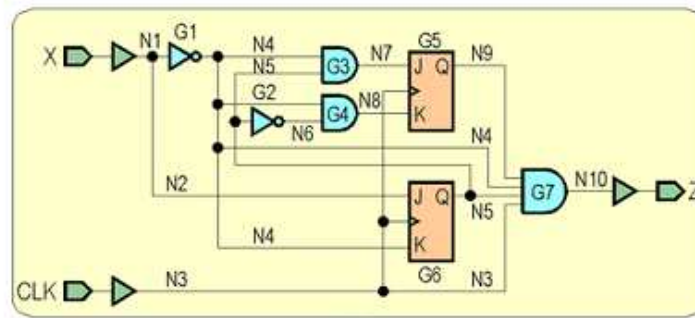
Există diferite limbaje de descriere hardware, dar mai utilizat este limbajul VHDL (VHSIC Hardware Description Language), VHSIC fiind acronimul pentru Very High Speed Integrated Circuit. Pe lângă acest limbaj, pentru proiectarea cu circuite FPGA se mai utilizează limbajul Verilog. Pentru proiectarea cu circuite CPLD, un limbaj utilizat în mod frecvent este ABEL (Advanced Boolean Expression Language). Limbajele VHDL și Verilog sunt standardizate de institutul IEEE.

Sinteza sistemului

După descrierea sistemului digital, etapa următoare din cadrul procesului de proiectare este cea de sinteză a sistemului. Sinteza constă în translatarea schemei logice, a descrierii HDL sau a diagramei de stare într-o listă de conexiuni. Această translatare se realizează cu ajutorul unui program de sinteză din cadrul sistemului CAD. Lista de conexiuni (“netlist”) este o descriere compactă a sistemului digital sub formă textuală, în care sunt specificate componentele sistemului, interconexiunile dintre acestea și pinii de intrare/ieșire. Această listă este prelucrată de celelalte componente ale sistemului CAD pentru realizarea etapelor următoare din cadrul procesului de proiectare.

Există diferite formate pentru listele de conexiuni, cel mai utilizat fiind formatul EDIF (Electronic Digital Interchange Format), acesta reprezentând un standard industrial. Pe lângă acest format standard, se pot utiliza diferite formate care sunt specifice anumitor producători de circuite. Un exemplu este formatul XNF (Xilinx Netlist Format), care este formatul propriu al firmei Xilinx, cel mai important producător de circuite programabile de tip FPGA și CPLD. O altă posibilitate este utilizarea unui limbaj de descriere hardware ca format pentru lista de conexiuni. De exemplu, sistemul CAD poate utiliza o reprezentare structurală a sistemului proiectat într-un limbaj de descriere hardware specificat de proiectant.

Relația dintre schema logică a unui circuit simplu și un format posibil al unei liste de conexiuni:



Component INV G1;
 Component INV G2;
 Component AND2 G3;
 Component AND2 G4;
 Component JKF G5;
 Component JKF G6;
 Component AND4 G7;

Net N1: X, G1.IN;
 Net N2: X, G6.J;
 Net N3: CLK, G5.C, G6.C, G7.IN4;
 Net N4: G1.OUT, G3.IN1, G4.IN1,
 G7.IN2, G6.K;
 Net N5: G6.Q, G2.IN, G3.IN2, G7.IN3;
 Net N6: G2.OUT, G4.IN2;
 Net N7: G3.OUT, G5.J;
 Net N8: G4.OUT, G5.K;
 Net N9: G5.Q, G7.IN1;
 Net N10: G7.OUT, Z;

În prima parte a listei de conexiuni sunt declarate componentele din cadrul schemei, iar în a doua parte sunt specificate conexiunile dintre componente. Denumirile componentelor sunt G1..G7, iar denumirile conexiunilor sunt N1..N10. Aceste denumiri sunt fie cele specificate de proiectant, fie cele asigurate în mod automat de sistemul CAD.

În circuitul ilustrat există două inversoare (G1 și G2), două porți ȘI cu două intrări (G3 și G4), o poartă ȘI cu patru intrări (G7) și două bistabile JK (G5 și G6). Inversoarele au un pin de intrare IN, un pin de ieșire OUT, un pin de alimentare Vcc și un pin de masă GND. Similar, porțile ȘI cu două intrări au doi pini de intrare IN1 și IN2, un pin de ieșire OUT, un pin de alimentare și un pin de masă. Bistabilele au doi pini pentru intrările de date J și K, un pin pentru intrarea de ceas C și un pin pentru ieșirea Q, pe lângă pinii de alimentare și masă. Pentru simplitate, pinii și semnalele de alimentare și masă au fost omiși în această figură. O conexiune este indicată prin listarea tuturor pinilor care sunt conectați împreună. Semnalele de intrare X și CLK sunt conectate la pinii de intrare cu aceleași nume ai circuitului, iar semnalul de ieșire Z este conectat la pinul de ieșire al circuitului.

Proiectantul poate specifica diferite criterii de optimizare de care să se țină cont în procesul de sinteză. Exemple de asemenea opțiuni sunt:

- minimizarea numărului de porți elementare necesare,
- obținerea vitezei maxime de funcționare a circuitului,
- minimizarea puterii consumate.

Proiectantul poate experimenta cu diferite criterii de optimizare pentru a obține soluția cea mai convenabilă pentru aplicația respectivă.

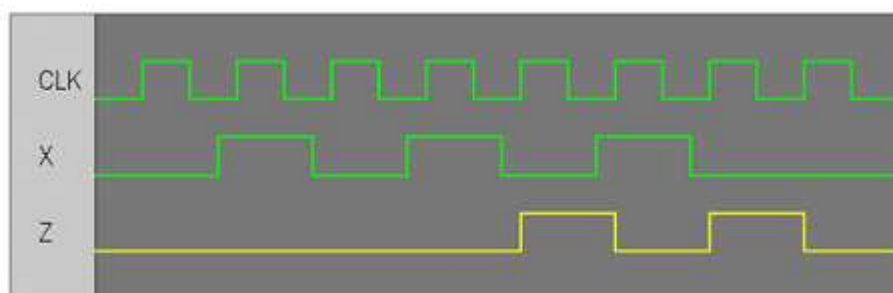
Simularea funcțională

În această etapă se utilizează un program simulator pentru verificarea funcționării sistemului proiectat, înainte de implementarea acestuia într-un circuit programabil. Această verificare se referă doar la aspectele funcționale ale sistemului, fără a se lua în considerare întârzierile semnalelor, care vor fi cunoscute numai după implementare. Pentru verificarea

funcțională proiectantul furnizează simulatorului mai multe combinații ale valorilor semnalelor de intrare, o asemenea combinație fiind numită **vector de test**. De asemenea, proiectantul poate specifica valorile semnalelor de ieșire care trebuie generate de sistem pentru fiecare vector de test.

Simulatorul aplică pe rând câte un vector de test la intrările sistemului, determină semnalele de ieșire care sunt generate de sistem și le compară cu valorile acestor semnale care au fost specificate de proiectant. În cazul în care apar diferențe, simulatorul afișează mesaje care indică diferențele apărute. Proiectantul va efectua modificările necesare ale descrierii sistemului pentru a corecta erorile apărute, va efectua sinteza descrierii modificate și va executa din nou simularea funcțională. Aceste etape vor fi repetate până când sistemul va funcționa conform cerințelor.

Modul în care pot fi vizualizate pe ecranul calculatorului semnalele de intrare și de ieșire ale unui circuit:



Maparea tehnologică

Etapele următoare din cadrul procesului de sinteză realizează implementarea sistemului proiectat într-un circuit programabil (FPGA sau CPLD). Prima etapă din cadrul implementării este cea de mapare tehnologică. Această etapă constă dintr-o serie de operații care realizează prelucrarea listei de conexiuni și adaptarea acesteia la particularitățile și resursele disponibile ale circuitului utilizat pentru implementare. Operațiile executate în această etapă diferă în funcție de sistemul de proiectare. Cele mai obișnuite operații sunt:

- adaptarea la elementele fizice ale circuitului,
- optimizarea și verificarea regulilor de proiectare (de exemplu, testarea depășirii numărului pinilor de I/E disponibili în cadrul circuitului). În timpul acestei etape, proiectantul selectează tipul circuitului programabil care va fi utilizat, capsula circuitului integrat, viteza și alte opțiuni specifice circuitului respectiv.

În urma execuției operațiilor din etapa de mapare tehnologică se generează un raport detaliat al rezultatelor tuturor programelor executate. Pe lângă mesaje de eroare și de avertizare, se creează de obicei o listă cu resursele utilizate din cadrul circuitului.

Plasarea și rutarea

Aceste operații sunt executate în cazul utilizării unui circuit FPGA pentru implementare.

Pentru proiectarea cu circuite CPLD, operația echivalentă este numită adaptare ("fitting").

Plasarea este procesul de selectare a unor module sau blocuri logice ale circuitului programabil care vor fi utilizate pentru implementarea diferitelor funcții ale sistemului digital.

Rutarea constă în interconectarea acestor blocuri logice utilizând resursele de rutare disponibile ale circuitului.

Majoritatea sistemelor CAD realizează operațiile de plasare și rutare în mod automat, astfel încât utilizatorul nu trebuie să cunoască detaliile arhitecturii circuitului utilizat pentru implementare.

Analiza de timp

Pachetele de programe CAD pentru proiectarea sistemelor digitale conțin de obicei un program numit analizor de timp, care poate furniza informații despre întârzierile semnalelor. Aceste informații se referă atât la întârzierile introduse de blocurile logice, cât și la întârzierile datorate interconexiunilor. Analizorul poate afișa aceste informații în diferite moduri, de exemplu, prin ordonarea conexiunilor în ordinea descrescătoare a întârzierilor semnalelor. Proiectantul poate utiliza informațiile despre întârzierile semnalelor pentru a realiza o nouă simulare a sistemului, în care să se țină cont de aceste întârzieri. Această operație prin care se furnizează simulatorului informații detaliate despre întârzierile semnalelor se numește adnotare inversă ("back-annotation").

Anumite sisteme permit utilizatorilor experți plasarea și rutarea manuală a unor porțiuni critice ale sistemului digital pentru a obține performanțe superioare.

Configurarea sau programarea circuitului

Operația de configurare se referă la circuitele programabile bazate pe memorii volatile SRAM (Static Random Access Memory) și constă din încărcarea informațiilor de configurare în memoria circuitului. Operația de programare se referă la circuitele programabile bazate pe memorii nevolatile (cum sunt circuitele care conțin antifuzibile). Această operație se execută similar cu cea de configurare, dar informațiile de configurare sunt păstrate și după întreruperea tensiunii de alimentare.

La sfârșitul operațiilor de plasare și rutare, se generează un fișier care conține toate informațiile necesare pentru configurarea circuitului. Aceste informații se referă atât la configurarea blocurilor logice ale circuitului, cât și la specificarea interconexiunilor dintre blocurile logice. Fișierul în care se înscriu aceste informații conține, în principiu, un șir de biți ("bitstream"), fiecare bit indicând starea închisă sau deschisă a unui comutator. Circuitele programabile conțin un număr mare de asemenea comutatoare, un comutator fiind realizat sub forma unui tranzistor sau a unei celule de memorie. Un bit de 1 din șirul de biți va determina închiderea unui comutator și, deci, stabilirea unei conexiuni.

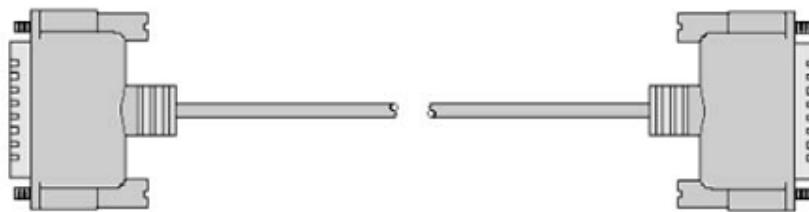
Biții din acest fișier de configurare sunt aranjați într-un anumit format pentru a realiza o corespondență între un bit și comutatorul corespunzător.

Conținutul fișierului de configurare se transferă la circuitul programabil, aflat de obicei pe o placă de circuit imprimat împreună cu alte circuite. Comutatoarele circuitului se închid sau rămân deschise în funcție de valorile biților din șirul de configurare. După terminarea configurării, circuitul va funcționa conform descrierii sistemului digital care a fost implementat.

Din cauza memoriei volatile, circuitul trebuie configurat din nou după fiecare întrerupere a tensiunii de alimentare. Informațiile de configurare pot fi păstrate într-o memorie nevolatilă PROM (Programmable Read Only Memory), existând posibilitatea configurării automate a circuitului din această memorie nevolatilă la aplicarea tensiunii de alimentare.

Configurarea sau programarea se pot realiza utilizând interfața paralelă a calculatorului. Pentru aceasta, este necesar ca placa cu circuitul programabil să conțină un conector pentru interfața paralelă, pentru transfer utilizându-se un cablu paralel.

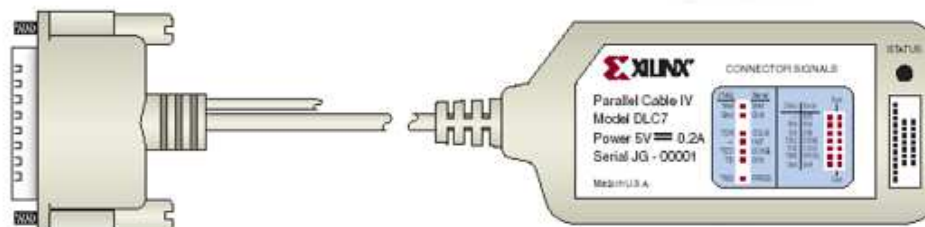
Exemplu de cablu paralel obișnuit, care conține conectori cu 25 de contacte DB25.



O altă posibilitate este utilizarea unui cablu special și a unei metodologii de configurare propuse de organizația JTAG (Joint Test Advisory Group). Această metodologie, cunoscută și sub numele de "Boundary-Scan", a fost standardizată de institutele IEEE (Institute of Electrical and Electronic Engineers) și ANSI (American National Standards Institute) ca standardul 1149.1,

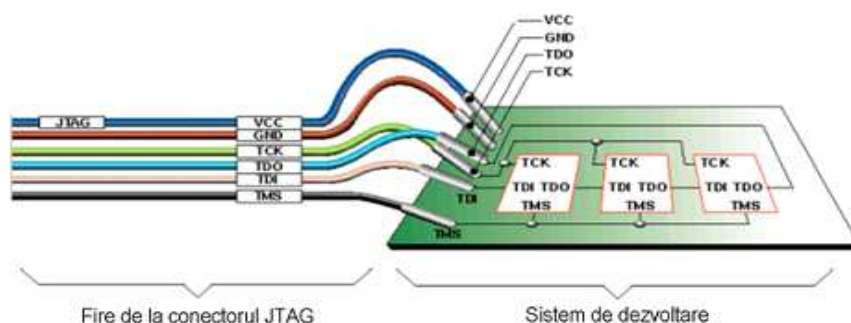
reprezentând un set de reguli de proiectare care facilitează configurarea sau programarea circuitelor, testarea și depanarea acestora. Un capăt al cablului JTAG se conectează la interfața paralelă a calculatorului, iar celălalt capăt se conectează la un număr de 5 pini speciali de pe placa circuitului programabil. Informațiile de configurare sunt preluate în paralel de la calculator și sunt transferate serial (bit cu bit) la circuitul programabil. Un asemenea cablu permite și testarea sistemului digital implementat prin citirea unor informații (valori ale semnalelor sau conținutul unor locații de memorie) de la circuitul programabil în timpul funcționării, transferul acestora la calculator și vizualizarea lor pe ecran.

Exemplu de cablu JTAG al firmei Xilinx (Parallel Cable IV):



Există mai multe variante de cabluri JTAG produse de această firmă. Cablul MultiLINX poate fi conectat fie la interfața serială RS232 a calculatorului, fie la interfața USB, prin intermediul unui cablu serial sau al unui cablu USB. Dispozitivul MultiPRO poate fi utilizat atât ca și cablu de configurare, cât și ca programator pentru memorii PROM și circuite CPLD CoolRunner II cu ajutorul unor adaptoare. Cablurile Parallel Cable III și Parallel Cable IV se conectează la interfața paralelă a calculatorului. Cablul Parallel Cable IV permite o rată de transfer superioară comparativ cu cablul Parallel Cable III (de până la 5 MB/s față de 500 KB/s).

Exemplu deconectare a unui cablu JTAG în modul JTAG (sau “Boundary-Scan”) la un sistem de dezvoltare conținând unul sau mai multe circuite programabile. Firele de legătură se conectează cu un capăt la pinii JTAG ai cablului, iar cu celălalt capăt la pinii JTAG corespunzători ai plăcii de dezvoltare. Un asemenea cablu poate fi utilizat fie pentru configurarea unui singur circuit programabil, fie a mai multor circuite conectate într-un lanț “Boundary-Scan”. De menționat că un cablu JTAG poate fi utilizat de obicei și pentru configurarea circuitelor în alte moduri decât modul JTAG, cum sunt modurile “Slave Serial” sau “Slave Parallel”.



Depanarea sistemului

În această ultimă etapă a procesului de proiectare se verifică funcționarea sistemului digital proiectat în condiții reale. O funcționare necorespunzătoare se poate datora nerespectării specificațiilor de proiectare, a specificațiilor circuitului utilizat pentru implementare, a unor aspecte legate de întârzierea semnalelor etc. Depanarea poate fi simplificată dacă circuitul se configurează astfel încât să conțină unele module speciale care permit citirea valorii unor semnale în timpul funcționării și transferul acestor informații la calculator, utilizând un cablu JTAG și un program special pentru vizualizarea semnalelor dorite.