

# **PRINCIPII GENERALE PRIVIND STRUCTURA ȘI FUNCȚIONAREA UNUI SISTEM DE PRELUCRARE NUMERICĂ**

## **1.1. INTRODUCERE**

În acest capitol se prezintă principiile generale privind structura și funcționarea unui sistem de prelucrare numerică cu procesor, SPN. Aceste principii sunt generale în sensul că se regăsesc în structura și funcționarea unui SPN indiferent de tipul circuitului integrat pe scară largă care stă la baza construcției SPN și care poate fi microprocesor, microcontroler sau procesor numeric de semnal.

## **1.2. STRUCTURA GENERALĂ ȘI FUNCȚIONAREA DE PRINCIPIU A UNUI SPN**

Unitatea de informație numerică sau logică utilizată în prelucrarea numerică este variabila binară (bitul) cu valorile numerice 0 și 1 sau logice DA și NU. Pentru reprezentarea informației numerice sau logice, în SPN se utilizează secvențe de 8, 16, 24 sau 32 de biți, numite cuvinte. Cuvintele de 8 biți se numesc octeți. Valorile numerice ale cuvintelor sunt funcție de codurile utilizate (cod binar natural CBN, cod binar deplasat CBD, cod complementul lui doi CCD) și de tipul reprezentării, în virgulă fixă sau în virgulă mobilă.

Structura generală a unui SPN este prezentată în figura 1.1. Componenta esențială a unui SPN este unitatea aritmetică și logică (UAL), care lucrează cu operanzi biți și cuvinte. Operațiile pe care le poate efectua UAL sunt: adunare, scădere, înmulțire, incrementare, decrementare, ȘI, SAU, SAU-exclusiv etc. UAL este un circuit combinațional, astfel încât rezultă necesitatea unor registre de memorare temporară a operanzilor și a rezultatului unei operații. Astfel, UAL generează rezultatul unei operații și, în funcție de acest rezultat, poziționează la nivel logic 0 sau 1 biții indicatori de condiții ai unui registru de stare, numit registru cu indicatorii de

condiții. Astfel, un bit al acestui registru poate indica una dintre condițiile: depășirea domeniului de valori corespunzător reprezentării operanzilor ca urmare a unei operații de adunare sau scădere, obținerea unui rezultat zero ca urmare a unei operații aritmetice sau logice, obținerea unui rezultat pozitiv sau negativ etc.

Operanzii și rezultatele operațiilor se numesc date. Datele se memorează în registre și în locații ale memoriei SPN. Un registru sau o locație de memorie care conține un operand pentru o operație se numește sursă pentru operația corespunzătoare. Un registru sau o locație de memorie în care se încarcă rezultatul unei operații se numește destinație pentru operația corespunzătoare. Transferul datelor între componentele SPN se realizează prin magistrala de date care interconectează aceste componente. Din punct de vedere fizic, magistrala de date constă în conexiuni cu 8, 16 sau 32 fire de legătură paralele. În general, funcție de numărul de biți ai magistralei de date, se spune că SPN este de 8, 16 sau 32 de biți.

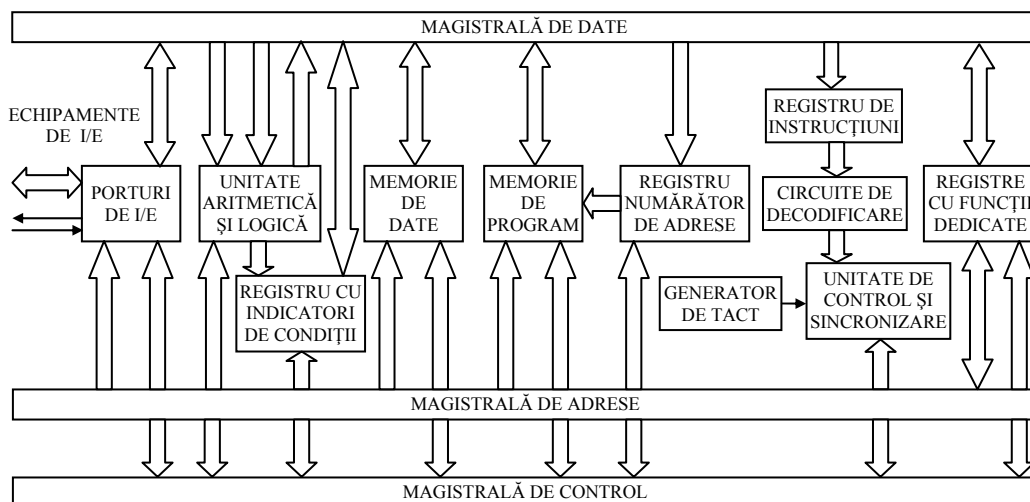


Fig. 1.1. Structura generală a unui sistem de prelucrare numerică.

Memoria de date din structura unui SPN se utilizează pentru memorarea datelor și este de tip RAM și ROM. Memoria de date de tip ROM se utilizează pentru memorarea unor constante care intervin în algoritmi de prelucrare numerică. Adresarea locațiilor memoriei de date se realizează prin magistrala de adrese a SPN de 16, 20, 24 sau 32 de biți. Fiecărei adrese referitoare la memoria de date îi corespunde o locație de memorie la care se poate efectua o operație de scriere a cuvântului încărcat pe magistrala de date de o altă componentă a SPN, sau o operație de citire de către o altă componentă a SPN a conținutului locației de memorie adresate, conținut încărcat de memorie pe magistrala de date. Magistralei de adrese de 16, 20, 24 sau 32 de biți îi corespunde un spațiu de adresare de 64 K, 1 M, 16 M și, respectiv, 4 Gcuvinte.

SPN cuprinde un grup de registre de 8, 16 sau 32 biți, numite interne, având funcții dedicate (speciale) prin structura sistemului. Aceste registre conțin date, adrese și informații de control. Astfel, există registre dedicate care se utilizează ca surse cu operanzi sau ca destinații ale rezultatelor pentru anumite operații. Un registru din grupul de registre cu funcții dedicate este registrul acumulator, notat cu A sau ACC, utilizat ca sursă și destinație în multe operații aritmetice, logice și de transfer.

Transferul de date între SPN și echipamentele de intrare/ieșire (I/E) se realizează prin porturi de I/E, care pot fi de tip paralel sau serie. În cazul unui port paralel, transferul unui cuvânt între SPN și un echipament de I/E se realizează printr-o magistrală de I/E cu 8, 16 sau 32 de linii, funcție de lungimea cuvântului. Transferul unui cuvânt de la un echipament la SPN se numește operație de intrare, iar transferul invers se numește operație de ieșire. În cazul unui port serial, transferul de date între SPN și un echipament de I/E se realizează prin două linii de comunicație, de transmisie și, respectiv, de recepție (din punctul de vedere al SPN). Biții corespunzători unui cuvânt se încarcă în ordine succesivă, cu frecvența de comunicație, pe linia de recepție sau transmisie, funcție de sensul transferului. Adresarea porturilor de I/E se realizează prin magistrala de adrese a SPN.

Din cele prezentate mai sus rezultă că magistrala de adrese conține cuvinte de adresare a datelor din: memoria de date, registrele cu funcții dedicate și porturile de intrare. Aceste componente ale SPN pot încărca magistrala de date. Deoarece la un moment dat o singură componentă a SPN poate încărca magistrala de date, rezultă necesitatea selecției componentelor sistemului funcție de operațiile executate de acesta. Această selecție se realizează prin magistrala de control a SPN de către unitatea de control și sincronizare.

Funcția de prelucrare numerică este realizată de către sistem prin execuția secvențială a unor operații aritmetice, logice și de transfer. Operațiile de transfer se realizează între componentele SPN sau între SPN și echipamentele de I/E. O operație se realizează prin execuția de către SPN a unei instrucțiuni. Rezultă că o succesiune de operații corespunde unei succesiuni de instrucțiuni, care formează un program. O instrucțiune este definită prin 1÷6 cuvinte de 8 sau 16 biți, care conțin codul operației de executat, operanzii sau adresele operanzilor și adresa destinației. Cuvintele care definesc o instrucțiune reprezintă codul mașină al instrucțiunii. Elaborarea unui program prin scrierea codurilor mașină ale instrucțiunilor corespunzătoare se numește programare în limbaj mașină.

Fiecărei instrucțiuni îi corespunde o scriere simbolică (cu caractere alfanumerice) care trebuie să precizeze aceleași informații ca și codul mașină, informații care constau în codul operației, operanzii sau adresele operanzilor și adresa destinației. Simbolul corespunzător codului operație se numește mnemonică. Elaborarea unui program prin scrierea simbolică a instrucțiunilor se numește programare în limbaj de asamblare.

Codurile mașină ale instrucțiunilor unui program sunt plasate la adrese succesive în memoria de program a SPN. Memoria de program, de tip ROM sau

RAM, este conectată, ca și memoria de date, la magistralele de date și de adrese ale SPN. Rezultă că magistrala de date se încarcă și cu cuvinte care reprezintă codurile instrucțiunilor.

Execuția unei instrucțiuni începe cu extragerea din memoria de program a primului cuvânt din codul mașină, cuvânt care precizează codul operației corespunzătoare instrucțiunii. Sub comanda unității de control și sincronizare, acest cuvânt este transferat prin magistrala de date în registrul de instrucțiuni al SPN. Registrul de instrucțiuni realizează memorarea temporară a cuvântului cod operație în scopul decodificării. Rezultatul decodificării este transmis la unitatea de control și sincronizare care comandă funcționarea componentelor SPN pentru execuția instrucțiunii identificată prin decodificare. Această comandă se realizează prin magistrala de control a SPN. În cazul în care codul mașină al instrucțiunii conține mai mult de un cuvânt, execuția instrucțiunii cuprinde și extragerea din memoria de program a celorlalte cuvinte conținând date și/sau adrese. Extragerea în ordine succesivă a cuvintelor reprezentând codurile mașină ale instrucțiunilor unui program se realizează prin adresarea memoriei de program cu registrul numărător de adrese ale programului (PC). Registrul PC (de 16 biți) se incrementează cu o unitate după fiecare extragere de cuvânt cod instrucțiune. Unitatea de control și sincronizare poate comanda încărcarea în registrul PC și a altor valori decât cele rezultate din numărare în ordine naturală, rezultând salturi în citirea memoriei de program. Efectuarea unui astfel de salt se numește transfer al controlului și poate rezulta ca urmare a execuției unei instrucțiuni de transfer al controlului (salt, apel de subrutină, revenire din subrutină) sau ca urmare a unei cereri de întrerupere.

Din cele prezentate rezultă că execuția unei instrucțiuni de către un SPN cuprinde următoarele operații de bază:

- *extragere cod operație* - transferul din memoria de program în registrul de instrucțiuni al primului cuvânt din codul mașină al instrucțiunii, cuvânt care conține codul operației de executat prin instrucțiune;

- *decodificare* - analiza cuvântului cod operație cu circuitele pentru decodificarea instrucțiunilor și transferul rezultatului decodificării la unitatea de control și sincronizare;

- *transfer operanzi* - transferul operanzilor între componentele SPN (memorie de program, memorie de date, registre, porturi de I/E) în scopul execuției instrucțiunii;

- *execuție* - execuția operației aritmetice, logice sau de transfer precizată de codul operație al instrucțiunii.

Execuția unei instrucțiuni începe cu extragere cod operație și decodificare, continuând cu o secvență specifică de operații de bază de transfer operanzi și execuție. Astfel, este necesară funcționarea secvențială și sincronizată a SPN. Această funcționare se obține prin comanda componentelor SPN de către unitatea de control și sincronizare. Viteza de execuție a instrucțiunilor este funcție de frecvența semnalului de la generatorul de tact al SPN. În general, o operație de bază se efectuează pe durata a unei perioade sau a mai multor perioade ale semnalului de tact. Intervalul

corespunzător efectuării unei operații de bază se numește un ciclu mașină al SPN. Ciclurile mașină corespunzătoare unei instrucțiuni definesc un ciclu instrucțiune. Execuția unei instrucțiuni durează câteva cicluri mașină incluzând: ciclu de extragere cod operație, ciclu de decodificare și, apoi, funcție de tipul instrucțiunii, cicluri de citire/scriere, cicluri de intrare/ieșire și cicluri de execuție. Efectuarea acestor cicluri poate implica componente diferite din structura SPN. Rezultă posibilitatea ca SPN să efectueze simultan cicluri diferite din execuția unei instrucțiuni sau din execuția unor instrucțiuni succesive. Această tehnică de suprapunere în timp a execuției ciclurilor mașină (operare paralelă) se numește tehnică pipeline și utilizarea ei în funcționarea unui SPN conduce la mărirea vitezei de lucru a acestuia.

În general, realizarea unui SPN se bazează pe utilizarea unui circuit integrat pe scară largă de tip microprocesor, microcontroler sau procesor numeric de semnal, care conține, pentru orice tip, o unitate centrală de prelucrare UCP cu următoarele componente din structura generală a unui SPN: unitate aritmetică și logică, registru cu indicatorii de condiții, registru numărator de adrese, registru de instrucțiuni, circuite de decodificare a instrucțiunilor, unitate de control și sincronizare și registre cu funcții dedicate. Toate aceste componente se numesc interne, relativ la circuitul integrat utilizat ca bază pentru realizarea SPN. În acest sens, registrele din structura SPN se numesc registre interne. Un circuit de tip microcontroler conține toate componentele din structura generală a unui SPN, figura 1.1, incluzând memorie internă, porturi de I/E, precum și alte periferice.

### **1.3. FUNCȚIONAREA UNUI SPN PENTRU EXECUȚIA INSTRUȚIUNILOR DE TRANSFER AL CONTROLULUI**

Transferul controlului constă în efectuarea unui salt în procesul de extragere a codurilor instrucțiunilor dintr-un spațiu de adresare al memoriei de program în altul. Rezultă că transferul controlului se realizează prin modificarea conținutului registrului numărator de adrese PC. Instrucțiunile de transfer al controlului sunt instrucțiunile de salt, apel de subrutine și revenire din subrutine, prin execuția cărora se realizează, în principal, încărcarea registrului PC cu adresa de salt, de început a subrutinei și, respectiv, de revenire din subrutină. În cazul instrucțiunilor de salt și de apel de subrutine, aceste adrese sunt indicate prin codurile mașină ale instrucțiunilor.

Subrutina este definită de programator ca un grup de instrucțiuni care realizează o funcție dată, funcție necesară a fi executată de mai multe ori în cadrul unui program. Execuția grupului de instrucțiuni ca subrutină permite scrierea codurilor mașină corespunzătoare o singură dată în memoria de program. Astfel, programul conține instrucțiuni de apel al subrutinei *CALL* în pozițiile în care se impune execuția funcției corespunzătoare. Subrutina se termină cu o instrucțiune de revenire *RET*, care determină revenirea în programul apelant la instrucțiunea următoare instrucțiunii care

a realizat apelul subrutinei. Procedura de apel și revenire din subrutină se desfășoară conform reprezentării din figura 1.2, în care:  $ADR1$  și  $ADR2$  sunt adresele locațiilor din memoria de program care conțin primul cuvânt din codul mașină al instrucțiunilor de apel de subrutină  $CALL$ ,  $n$  este numărul de cuvinte din codul mașină al unei instrucțiuni  $CALL$  și  $ASRT$  este adresa de început a subrutinei  $SRT$ .

Adresele de revenire din subrutină  $ADR1+n$ , respectiv  $ADR2+n$  sunt disponibile în registrul PC după extragerea din memoria de program a cuvintelor corespunzătoare instrucțiunii  $CALL$  apelante. Rezultă posibilitatea memorării adresei de revenire, înainte de înlocuirea conținutului registrului PC cu adresa  $ASRT$  de început a subrutinei. Adresa de revenire este necesară pentru execuția instrucțiunii de revenire din subrutină,  $RET$ . Deoarece o subrutină, la rândul ei, poate apela alte subrutine, rezultă necesitatea unei zone de memorie pentru memorarea adreselor de revenire. Această memorie a SPN se numește stivă, deoarece adresele de revenire sunt depuse în ea în ordine succesivă, una peste cealaltă. Ordinea de scoatere din memoria stivă a adreselor de revenire este inversă în raport cu ordinea de depunere în stivă, adică ultima introdusă este prima extrasă. Altfel spus, memoria stivă este de tip LIFO.

### MEMORIE DE PROGRAM

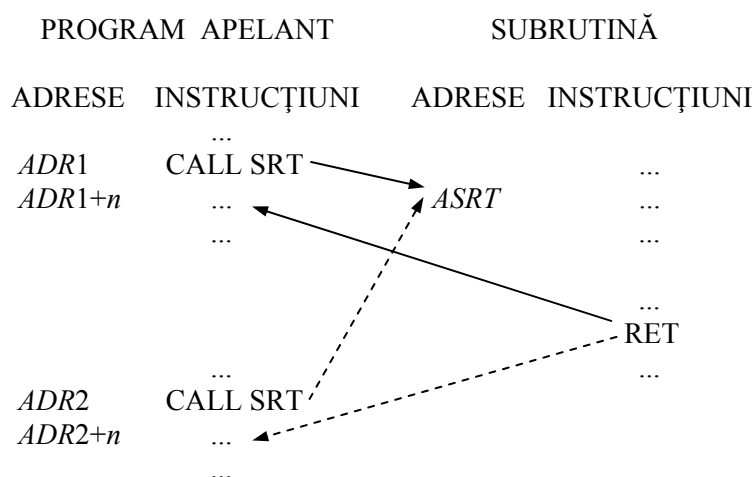


Fig. 1.2. Reprezentare a procedurii de apel și revenire din subrutină.

Un SPN poate conține în structura sa o memorie LIFO dedicată pentru stivă. O altă variantă, des utilizată, constă în organizarea stivei în memoria de date a SPN prin utilizarea registrului indicator de stivă (SP) din grupul registrelor cu funcții dedicate. Registrul SP este destinat pentru adresarea memoriei stivă. Astfel, conținutul registrului SP este inițializat la conectarea SPN sau poate fi stabilit, imediat după conectare, prin programare. După aceste operații, conținutul registrului SP urmărește

adresa corespunzătoare vârfului stivei. Această urmărire se realizează prin incrementare/decrementare cu o unitate a conținutului registrului SP la fiecare operație de scriere a unui cuvânt în stivă, respectiv prin decrementare/incrementare cu o unitate la fiecare operație de citire (extragere) din stivă. O altă variantă de realizare a stivei, utilizată în SPN care nu conțin registru SP, constă în implementarea prin program a logicii de funcționare corespunzătoare registrului SP. O astfel de stivă se numește stivă soft. Se precizează că memoria stivă poate fi utilizată și ca memorie de uz general, cu acces rapid, de tip LIFO și adresabilă cu registrul SP. Accesul stivei ca memorie de uz general se realizează cu instrucțiuni de scriere în stivă (*PUSH*) și cu instrucțiuni de citire din stivă (*POP*). În acest caz, gestiunea conținutului stivei pentru funcționarea corectă la reveniri din subrutine și la extrageri de date din stivă este sarcina programatorului.

Codul mașină corespunzător unei instrucțiuni de transfer al controlului (salt sau apel de subrutină) trebuie să precizeze, în afara codului operație, adresa la care se transferă controlul (adresa de salt, respectiv adresa de început a subrutinei). Această adresă poate fi precizată în variantă absolută sau relativă. În cazul transferului absolut al controlului, codul mașină al instrucțiunii conține cuvântul reprezentând adresa la care se realizează transferul controlului, cuvânt care se încarcă în registrul PC. Astfel, transferul controlului se realizează în orice zonă din memoria de program. În cazul transferului relativ al controlului, codul mașină al instrucțiunii cuprinde cuvântul care reprezintă un deplasament. Adresa la care se efectuează transferul controlului se obține prin adunarea deplasamentului la conținutul registrului PC existent după extragerea din memorie a cuvintelor reprezentând codul mașină al instrucțiunii de transfer al controlului corespunzătoare. Valoarea deplasamentului corespunde codului complementului lui doi. Se precizează că adresele de salt relativ (deplasamentul) din cadrul unui program nu se modifică la deplasarea programului dintr-o zonă în alta a memoriei de program.

Instrucțiunile de transfer al controlului pot fi în variantă condiționată, situație în care saltul, apelul subrutinei sau revenirea din subrutină, se poate efectua sau nu (se execută instrucțiunea următoare celei de transfer al controlului), funcție de starea unui bit indicator de condiție, de rezultatul comparării a două cuvinte, de conținutul zero sau diferit de zero al acumulatorului sau funcție de nivelul logic al unui bit din memoria de date.

## **1.4. TEHNICI DE ADRESARE A OPERANZILOR PENTRU EXECUȚIA INSTRUCȚIUNILOR**

Prin execuția instrucțiunilor, un SPN efectuează operații aritmetice, logice, de transfer date și de transfer al controlului. Operația executată de o instrucțiune este precizată în primul cuvânt din codul mașină corespunzător instrucțiunii. Din punctul

de vedere al operației, execuția instrucțiunii necesită precizarea operandilor numiți de tip sursă sau a adreselor acestora, precum și precizarea adresei de destinație a rezultatului operației. Aceste precizări sunt conținute în codul mașină al instrucțiunii. În cele ce urmează se utilizează termenul de operand al unei instrucțiuni, atât pentru un operand sursă, cât și pentru destinație. Operandii pentru instrucțiuni pot fi dați de registrele interne, memoria de date, memoria de program și porturile de I/E. De exemplu, conținutul unei locații de memorie poate fi operand sursă pentru o instrucțiune și o altă locație de memorie poate fi destinație a rezultatului operației.

Tehnicile de adresare a operandilor pentru execuția instrucțiunilor reprezintă modalitățile de indicare a operandilor, de formare și calcul al adreselor operandilor. O clasificare generală cuprinde următoarele tehnici de adresare a operandilor:

**1. Adresare inerentă.** Operandul este dat de un registru intern stabilit prin alocare specifică de registre pentru instrucțiuni. Alocarea specifică este dată prin structura SPN.

**2. Adresare imediată.** Operandul sursă este conținut în codul mașină al instrucțiunii care utilizează această tehnică de adresare. Rezultă că operandul sursă este conținut în memoria de program.

**3. Adresare directă.** Operandul este dat de un registru intern, o locație de memorie sau un port de I/E de la adresa conținută în codul mașină al instrucțiunii corespunzătoare.

**4. Adresare indexată.** Operandul este dat de o locație de memorie sau un port de I/E de la adresa dată de conținutul unui registru intern. Există variante de adresare indexată în care adresa se obține prin adunarea la conținutul unui registru intern a conținutului altui registru intern și/sau a unui deplasament conținut în codul mașină al instrucțiunii corespunzătoare.

**4. Adresare indexată-indirectă.** Tehnica de adresare indexată-indirectă este utilizată într-o instrucțiune pentru accesul unui operand sursă sau destinație corespunzător unei locații de memorie a cărei adresă numită adresă indicator se obține din memorie prin adresare indexată.

## 1.5. FUNCȚIONAREA UNUI SPN CU ÎNTRERUPERI

Tehnica întreruperilor este utilizată pentru sincronizarea și corelarea activității SPN cu evenimente rezultate din funcționarea acestuia și a echipamentelor de I/E. De exemplu, tehnica întreruperilor este utilizată pentru sincronizarea și realizarea transferului de date între un SPN și echipamente de I/E de viteză redusă, cu acces periodic sau aleator.

Conform tehnicii întreruperilor, SPN își desfășoară activitatea de bază în cadrul unui program, numit program principal, a cărui execuție poate fi întreruptă de o cerere de întrerupere efectuată de către un echipament de I/E sau rezultată din funcționarea SPN. Recepția și acceptarea de către SPN a unei cereri de întrerupere



determină, după terminarea instrucțiunii în curs de execuție, apelul unei subrutine de întrerupere specifică sursei a cărei cerere de întrerupere a fost acceptată. Astfel, în subrutina de întrerupere se realizează un transfer de date sau o funcție, în corelație cu sursa a cărei cerere de întrerupere a fost acceptată. După execuția subrutinei de întrerupere, SPN revine la execuția programului principal începând cu instrucțiunea următoare celei după care s-a acceptat întreruperea. Pentru execuția corectă a programului principal se impune ca la începutul și la sfârșitul subrutinei de întrerupere să se salveze, respectiv să se restabilească starea programului întrerupt. De exemplu, operațiile de salvare și restabilire vizează conținuturile registrelor interne cu care operează și programul principal și subrutina de întrerupere. Salvarea stării programului principal se poate face, de exemplu, în memoria stivă cu instrucțiuni *PUSH*. În acest caz, restabilirea stării programului principal se face din memoria stivă cu instrucțiuni *POP*.

Sursele de cereri de întrerupere pot fi nemascabile sau mascabile. O sursă se numește nemascabilă dacă nu există posibilitatea de blocare prin programare a acceptării de către SPN a cererilor de întrerupere corespunzătoare sursei. În cazul surselor mascabile există posibilitatea de validare sau invalidare prin programare a acceptării de către SPN a cererilor de întrerupere corespunzătoare. Validarea sau invalidarea se poate realiza global, pentru toate sursele de cereri de întrerupere mascabile sau individual, pentru fiecare sursă. În general, un SPN conține un bit (bistabil) care poate fi stabilit prin programare la un nivel logic pentru invalidarea tuturor surselor mascabile sau la nivelul logic complementar, pentru care se permite validarea/invalidarea individuală a surselor. Pentru această din urmă operație, SPN conține un registru de validare (mascare) a surselor de cereri de întrerupere. Fiecărei surse mascabile îi corespunde un bit al acestui registru, care poate fi stabilit prin programare pentru validarea/invalidarea sursei corespunzătoare.

Din cele prezentate mai sus a rezultat că un SPN poate primi cereri de întrerupere de la mai multe surse, fiecărei surse corespunzându-i o subrutină de întrerupere specifică. Deoarece, la un moment dat, un SPN poate deservi o singură cerere de întrerupere, prin rularea subrutinei de întrerupere corespunzătoare, rezultă necesitatea stabilirii unei ordini de prioritate a surselor de cereri de întrerupere, ordine bazată pe criterii funcționale. Ordinea de prioritate a surselor de cereri de întrerupere într-un SPN se stabilește atât prin structura sistemului, cât și prin programare.

Funcționarea unui SPN la acceptarea unei cereri de întrerupere începe, ca și în cazul instrucțiunilor de apel de subrutine, cu salvarea în memoria stivă a adresei de revenire din subrutina de întrerupere. Această adresă de revenire coincide cu conținutul registrului numărător de adrese PC, existent după execuția ultimei instrucțiuni din programul principal, înainte de apelul subrutinei de întrerupere. După salvarea adresei de revenire, se efectuează apelul subrutinei de întrerupere, prin încărcarea conținutului registrului PC cu adresa de început a subrutinei de întrerupere, în corelație cu sursa a cărei cerere de întrerupere se servește. În funcție de tipul SPN, există două variante de principiu pentru efectuarea apelului subrutinei de întrerupere.

Conform primei variante, într-o zonă a memoriei SPN se construiește, în prealabil, o tabelă cu adresele de început ale subrutinelor de întrerupere corespunzătoare diferitelor surse. Poziția tabelii în memorie, precum și pozițiile în tabelă corespunzătoare diferitelor surse pot fi stabilite prin structura SPN și/sau prin programare, funcție de tipul SPN. Astfel, apelul subrutinei de întrerupere corespunzătoare unei surse se realizează prin încărcarea registrului PC cu adresa de început a subrutinei de întrerupere extrasă din tabelă, de la locațiile de memorie corespunzătoare sursei.

Conform variantei a doua, registrul PC se încarcă cu o adresă corespunzătoare memoriei de program a cărei valoare este în corelație cu sursa de cerere de întrerupere și este stabilită prin structura SPN și/sau prin programare. Pentru diferite surse de cereri de întrerupere, adresele cu care se încarcă registrul PC sunt decalate cu 2, 3 sau 8 poziții, funcție de tipul SPN. Acest decalaj este egal cu sau mai mare decât numărul de cuvinte din codul mașină al unei instrucțiuni de salt corespunzătoare sistemului. Astfel, pentru fiecare sursă de cerere de întrerupere, există locațiile de memorie necesare scrierii prealabile a unei instrucțiuni de salt la adresa de început a subrutinei de întrerupere.

După execuția unei subrutine de întrerupere, SPN revine la execuția programului întrerupt, printr-o procedură similară cu procedura de revenire din subrutine apelate cu instrucțiuni *CALL*. Astfel, revenirea se realizează prin încărcarea registrului PC cu adresa de revenire din memoria stivă, ca urmare a execuției de către SPN a unei instrucțiuni *RETI* de revenire din subrutina de întrerupere, care este ultima instrucțiune a subrutinei. Această instrucțiune specifică pentru funcționarea cu întreruperi realizează și informarea logicii de control întreruperi a SPN cu privire la terminarea execuției subrutinei de întrerupere. Informarea este necesară pentru acceptarea, pe baza sistemului de priorități, a unei eventuale cereri de întrerupere de la altă sursă de nivel de prioritate egal cu sau mai scăzut decât sursa tocmai deservită. Se precizează că execuția unei subrutine de întrerupere poate fi la rândul ei întreruptă de o cerere de întrerupere de la o sursă de nivel de prioritate mai ridicat.

## 1.6. MEMORIA UNUI SPN

În acest paragraf se prezintă probleme generale privind organizarea și controlul memoriei unui sistem de prelucrare numerică în general și în special pentru sisteme care lucrează în regim multiutilizator și multisarcină.

Se precizează că prin sarcină se înțelege o secvență de instrucțiuni logic independentă dintr-un program care poate fi executată separat de alte părți ale aceluiași program. Regimul de funcționare multisarcină al unui sistem constă în execuția întrețesută (prin multiplexare în timp) a două sau mai multe sarcini corespunzătoare unui program, cu consecința reducerii duratei de execuție a programului. Realizarea unui sistem de prelucrare numerică care să lucreze în regim multiutilizator și/sau

multisarcină necesită gestiunea alocării în timp a componentelor sistemului pentru diferiți utilizatori și sarcini. Această gestiune se realizează printr-un grup de programe numit sistem de operare.

Referitor la memoria unui sistem de prelucrare numerică, este necesară rezolvarea următoarelor probleme:

- mărirea capacității de memorie;
- creșterea vitezei de acces;
- alocarea de blocuri de memorie (pagini, segmente) diferite pentru utilizatori, sarcini, sistem de operare, programe, date;
- protecția informațiilor din blocurile de memorie.

Componentele din structura sistemului utilizate pentru rezolvarea acestor probleme sunt reunite sub denumirea de unitate de gestiune a memoriei și funcționează sub controlul sistemului de operare.

În cele ce urmează se prezintă, la nivel de principiu, metode pentru rezolvarea problemelor enunțate mai sus.

Un bloc de memorie reprezintă o mulțime de locații de memorie de la adrese succesive corespunzătoare unui spațiu de adresare. Gestiunea memoriei unui sistem pe bază de blocuri de memorie se poate realiza prin tehnica paginării sau segmentării. Rezultă blocuri de memorie care se numesc pagini, respectiv segmente.

Conform tehnicii paginării, memoria sistemului este împărțită în pagini succesive de capacități egale. Altfel spus, spațiul de adresare corespunzător memoriei este împărțit în subspații adiacente de aceeași dimensiune. Rezultă că spațiul de adresare corespunzător unei pagini este precizat printr-o adresă care indică pagina de memorie, adresă numită de bază sau adresă de pagină. Adresa unei locații de memorie dintr-o pagină este dată de o adresă numită relativă. Astfel, în tehnica paginării, secvența de biți  $A.n-1 \div A.0$  de adresare a locațiilor de memorie se împarte în două secvențe, una corespunzătoare biților mai semnificativi  $A.n-1 \div A.p$ , care reprezintă adresa de bază  $a_b$ , și una corespunzătoare biților mai puțin semnificativi  $A.p-1 \div A.0$ , care reprezintă adresa relativă  $a_r$ . Rezultă că dimensiunea spațiului de adresare corespunzător memoriei este de  $2^n$  cuvinte, numărul de pagini de memorie este  $2^{n-p}$  și dimensiunea spațiului de adresare corespunzător unei pagini este de  $2^p$  cuvinte. Adresa fizică  $a_f$  a unei locații de memorie se calculează în funcție de adresa de bază  $a_b$  și de adresa relativă  $a_r$  cu relația

$$a_f = a_b 2^p + a_r. \quad (1.1)$$

Conform tehnicii segmentării, memoria sistemului este împărțită în segmente de capacități diferite, până la 64 Kcuvinte. De asemenea, segmentele de memorie și, implicit, spațiile de adresare corespunzătoare segmentelor pot fi adiacente, disjuncte, suprapuse parțial sau suprapuse total. Spațiul de adresare corespunzător unui segment este precizat prin adresa segmentului, numită adresă de bază și prin dimensiunea segmentului. Poziția unei locații de memorie într-un segment este dată de o adresă,

numită relativă, a cărei valoare limită corespunde cu dimensiunea segmentului. Adresa fizică a unei locații de memorie se obține din adresa de bază și adresa relativă.

Memoria unui sistem de prelucrare numerică adresabilă direct prin magistrala de adrese a acestuia se numește memorie principală. Capacitatea maximă a memoriei principale este de  $2^n$  cuvinte, unde  $n$  reprezintă numărul de linii ale magistralei de adrese a SPN (de exemplu:  $n=20$  la microprocesorul 8086 și  $n=24$  la microprocesorul 80286). Memoriei principale îi corespunde un spațiu de adresare, numit real sau fizic, de  $2^n$  cuvinte. Adresa unei locații de memorie, încărcată pe magistrala de adrese a SPN, se numește adresă fizică.

Mărirea capacității de memorie a unui SPN la valori mai mari decât  $2^n$  cuvinte (1 Moctet pentru microprocesorul 8086 și 16 Mocteți pentru microprocesorul 80286) se poate obține prin includerea în sistem a unor echipamente de intrare/ieșire de tipul unități de memorie cu disc rigid și/sau flexibil. Aceste unități de memorie formează memoria auxiliară a sistemului. Accesul la memoria auxiliară se realizează prin transferul unor blocuri de memorie (pagini, segmente) din memoria auxiliară în memoria principală și invers. De exemplu, accesul la un segment care în momentul respectiv nu se găsește în memoria principală necesită eliberarea unui spațiu de adresare din memoria principală, prin transfer de segmente din memoria principală în memoria auxiliară și transferul segmentului de accesat din memoria auxiliară în memoria principală. Aceste transferuri se realizează de către unitatea de gestiune a memoriei sub controlul sistemului de operare în funcție de activitatea SPN și de alocarea blocurilor de memorie pentru utilizatori, sarcini, sistem de operare, programe și date. Astfel se realizează accesul memoriei auxiliare a SPN în mod transparent pentru utilizator.

Ansamblul format din memoria principală și memoria auxiliară cu acces transparent la memoria auxiliară se numește memoria virtuală a SPN. Rezultă că memoria virtuală a unui sistem poate avea o capacitate mult mai mare decât memoria principală. Capacitatea memoriei virtuale a unui SPN este limitată de caracteristicile unităților de memorie auxiliară, ale unității de gestiune a memoriei și ale sistemului de operare. Memoriei virtuale îi corespunde un spațiu de adresare virtual. De exemplu, în cazul unui SPN cu microprocesor 80286, spațiul de adresare virtual este de 1 Goctet pentru o sarcină.

Protecția informației din blocurile de memorie se realizează de către sistemul de operare pe baza unor niveluri de privilegiu prin care se reglementează posibilitatea sau imposibilitatea de acces a utilizatorilor, sarcinilor sau sistemului de operare la diferite blocuri de memorie.