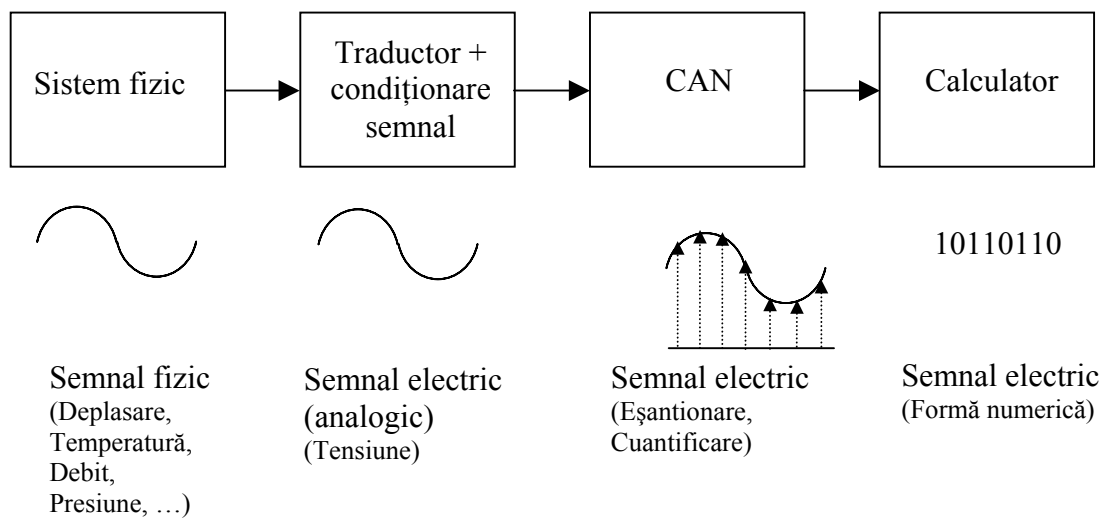


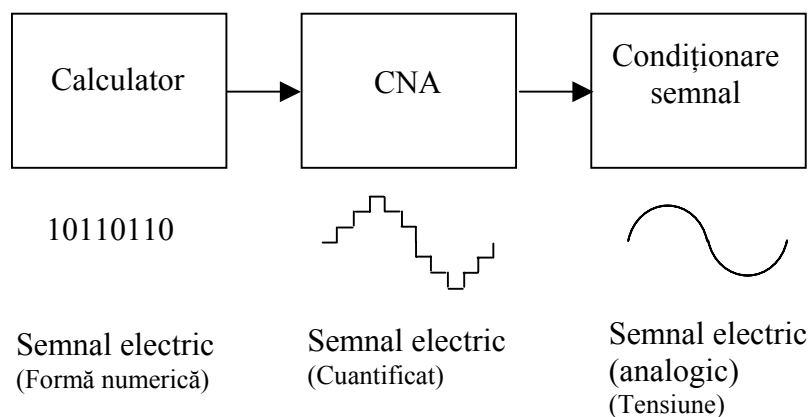
### CAP. 3. CONVERSII NUMERIC-ANALOGICE ȘI ANALOG-NUMERICE

Informațiile provenite din lumea fizică reală trebuie convertite într-o formă compatibilă pentru prelucrarea lor în calculator. De asemenea, semnalele provenite de la calculator (comenzi în cazul sistemelor de conducere) trebuie convertite într-o formă corespunzătoare echipamentelor cuplate la calculator.

În Fig. 3.1 este prezentată principal schema de intrare a unor semnale fizice în calculator. Se observă că după blocul corespunzător senzorilor și circuitelor de condiționare a semnalelor urmează un bloc de conversie analog-numerică, care are ca element principal convertorul analog-numeric (CAN). Fig. 3.2 ilustrează conversia numeric-analogică, cel mai important bloc din schemă fiind convertorul numeric-analogic (CNA).



**Fig. 3.1.** Conversia analog-numerică



**Fig. 3.2.** Conversia numeric-analogică

În paragrafele următoare vor fi descrise reprezentările mărimilor numerice în cod binar și vor fi analizate schemele de conversie numeric-analogică și analog-numerică.

### 3.1. CODURI NUMERICE BINARE

Spre deosebire de reprezentarea obișnuită a numerelor în sistem zecimal, se știe că în interiorul calculatoarelor se utilizează reprezentarea binară, care folosește numai valori de 0 și de 1, numiți digiți binari sau biți. Baza sistemului este 2, iar ponderile coeficienților depind de puterile lui 2. De exemplu, numărul 23 în baza zece se scrie 10111 în baza 2:

$$(23)_{10} = 2 \times 10^1 + 3 \times 10^0; \quad (10111)_2 \rightarrow 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 23$$

În continuare vom prezenta câteva din cele mai utilizate coduri binare, coduri care pot fi clasificate în două categorii: coduri binare unipolare și coduri binare bipolare (necesare deoarece semnalul fizic are în multe situații amplitudine bipolară: pozitivă și negativă).

#### 3.1.1. Coduri binare unipolare

Cel mai cunoscut cod binar unipolar este **codul binar natural**. Bitul cel mai din dreapta este cel mai puțin semnificativ bit (Least Significant Bit – LSB) iar bitul cel mai din stânga este cel mai semnificativ bit (Most Significant Bit – MSB). În codul binar natural ponderea fiecărui bit depinde de poziția acestuia în cadrul șirului. Exemplul de reprezentare anterior pentru numărul 23 (în baza 10) este un cod binar natural: 10111. Trebuie să observăm că și numărul binar 00010111 reprezintă tot numărul 23.

Dacă șirul are  $n$  biți, bitul MSB are o pondere de  $2^{n-1}$ , iar numărul maxim care poate fi reprezentat în cod binar natural este  $2^n - 1$ . Pentru a codifica  $2^n$  nivele sunt necesari minimum  $n$  biți. Un digit zecimal (o cifră zecimală) necesită cel puțin 4 biți pentru codificare binară.

Dacă vom considera numărul binar ca fiind o fracție dintr-un întreg sau dintr-un domeniu (situație întâlnită la CAN-uri), pentru un număr de  $n$  biți bitul MSB va avea o pondere de  $2^{-1}$  iar bitul LSB de  $2^{-n}$ , ceea ce înseamnă că la stânga numărului binar avem un punct binar (similar cu punctul zecimal). De exemplu, în cazul numărului binar 10111:

$$1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} + 1 \times 2^{-5} = \frac{1}{2^1} + \frac{1}{2^3} + \frac{1}{2^4} + \frac{1}{2^5} = \frac{23}{32}$$

Ponderea bitului LSB de  $2^{-n}$  este cea care dă rezoluția reprezentării pe  $n$  biți, rezoluție care trebuie înțeleasă în sensul definiției numărului maxim de valori distincte ale mărimii de intrare într-un CAN. Rezoluția determină în cazul CAN tensiunea minimă ce poate fi reprezentată atunci când cuantificăm un semnal continuu (analogic):

$$U_{\min} = D \times r$$

unde  $D$  este domeniul maxim și  $r$  este rezoluția. De exemplu, dacă avem un semnal analogic de până la 10 V și o codificare pe 5 biți, rezoluția va fi  $r = 2^{-5} = \frac{1}{32}$ , iar

$$U_{\min} = 10 \cdot \frac{1}{32} = 0.3125 \text{ V}.$$

Se poate folosi și o reprezentare procentuală a rezoluției:  $r_{\%} = \frac{1}{32} \cdot 100\% = 3.125\%$ .

În tabelul 3.1 sunt date rezoluțiile pentru câteva valori ale lui  $n$ .

Se observă că pentru îmbunătățirea rezoluției este necesară creșterea numărului de biți utilizat pentru codificare.

**Codul zecimal codificat binar** este un alt cod unipolar, care folosește pentru codificarea fiecărei cifre zecimale 4 biți, cu ponderile în varianta 8 – 4 – 2 – 1. Pentru a codifica un număr zecimal cu codul zecimal codificat binar, fiecare cifră a numărului zecimal

trebuie codificată separat în echivalentul ei pe 4 biți. De exemplu, numărul zecimal 8391 este codificat astfel:

1000 0011 1001 0001

**Tabelul 3.1.**

Numărul de biți $n$	$2^n$	$r$	$r\%$
1	2	0.5	50
2	4	0.25	25
3	8	0.125	12.5
4	16	0.0625	6.25
5	32	0.0312	3.12
6	64	0.0156	1.56
7	128	0.0078	0.78
8	256	0.0039	0.39
9	512	0.0019	0.19
10	1024	0.0009	0.09

### 3.1.2. Coduri binare bipolare

Aceste coduri binare au posibilitatea de a codifica și semnul (polaritatea) mărimii analogice. Semnul este codificat cu ajutorul unui bit plasat pe poziția cea mai din stânga a șirului. Folosirea acestui bit determină dublarea domeniului analogic, dar rezoluția se dublează și ea.

**Codul complement față de doi** este unul dintre cele mai utilizate coduri bipolare. Numerele pozitive sunt reprezentate cu un bit 0 pentru semn, iar cele negative sunt obținute prin complementarea fiecărui bit al numărului pozitiv și prin adăugarea unui 1. Are avantajul unei reprezentări unice pentru zero.

**Codul complement față de unu** este obținut prin complementarea fiecărui bit din numărul pozitiv.

**Codul semn – amplitudine** utilizează același cod pentru numerele pozitive și pentru cele negative, cu excepția bitului MSB care este 0 pentru cele pozitive și 1 pentru cele negative. Este utilizat pentru semnale care au valori în jurul lui zero, deoarece are tranziții mici în jurul lui zero.

**Codul binar offset** este similar cu codul complement față de doi, diferența fiind la MSB (1 pentru numere pozitive, 0 pentru negative).

Codurile bipolare sunt prezentate comparativ în tabelul 3.2 (pentru 4 biți). Pentru a converti un număr dintr-un cod în altul se pot utiliza algoritmi de conversie.

**Tabelul 3.2.** Coduri binare bipolare

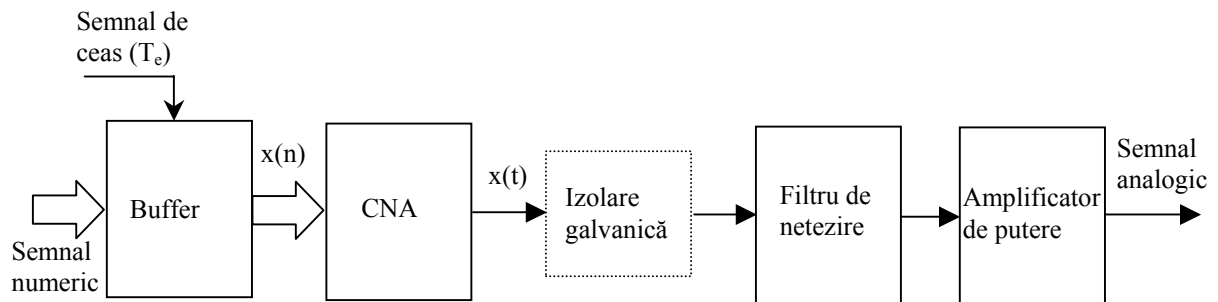
Număr zecimal	Fracția zecimală	Codul complement față de doi	Codul complement față de unu	Codul semn-amplitudine	Codul binar offset
+7	+7/8	0111	0111	0111	1111
+6	+6/8	0110	0110	0110	1110
+5	+5/8	0101	0101	0101	1101
+4	+4/8	0100	0100	0100	1100
+3	+3/8	0011	0011	0011	1011
+2	+2/8	0010	0010	0010	1010
+1	+1/8	0001	0001	0001	1001
0	0+	0000	0000	0000	1000

0	0-	(0000)	1111	1000	(1000)
-1	-1/8	1111	1110	1001	0111
-2	-2/8	1110	1101	1010	0110
-3	-3/8	1101	1100	1011	01101
-4	-4/8	1100	1011	1100	0100
-5	-5/8	1011	1010	1101	0011
-6	-6/8	1010	1001	1110	0010
-7	-7/8	1001	1000	1111	0001
-8	-8/8	(1000)			(0000)

## 3.2. CONVERTOARE NUMERIC-ANALOGICE (CNA)

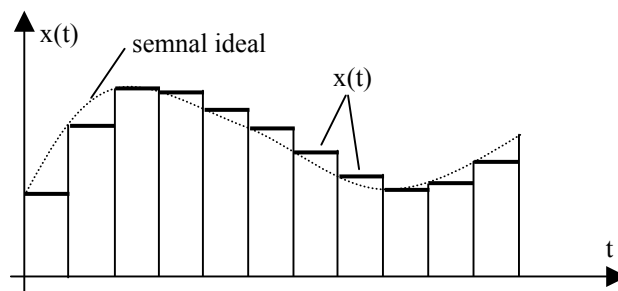
### 3.2.1. Reconstituirea semnalelor

Reconstituirea (reconstrucția) fizică a semnalelor este operația utilizată atunci când se dorește ca un semnal eșantionat și eventual prelucrat numeric de către un sistem de calcul să fie reconstituit din eșantioanele sale și furnizat în formă analogică la ieșirea sistemului. Într-un sistem de achiziție și conducere, reconstituirea și generarea semnalelor analogice se realizează prin intermediul *circuitelor de ieșiri analogice*. Reconstituirea fizică se realizează prin intermediul unui grup de dispozitive, dintre care rolul esențial îl are *convertorul numeric-analogic*. În Fig. 3.3 este prezentată organizarea generală a unui circuit de ieșire analogică.



**Fig. 3.3.** Schema bloc a reconstituirii fizice a semnalelor (organizarea ieșirilor analogice)

Convertorul numeric-analogic realizează o aproximare a operației de extrapolare de ordin zero. CNA-ul acceptă la intrarea sa semnalul în formă numerică, registrul tampon (bufferul) fiind utilizat pentru păstrarea datelor (numerele binare) pentru un interval de timp egal cu perioada de eșantionare ( $T_e$ ). CNA-ul convertește numărul binar într-o tensiune aproximativ egală cu valoarea actuală a semnalului numeric și o menține constantă pentru  $T_e$  secunde. Atunci când următorul număr apare la ieșirea bufferului, CNA-ul va schimba tensiunea de ieșire conform noii valori de la intrare. Principial, forma semnalului la ieșirea CNA este similară ieșirii extrapolatorului de ordin zero (Fig. 3.4).



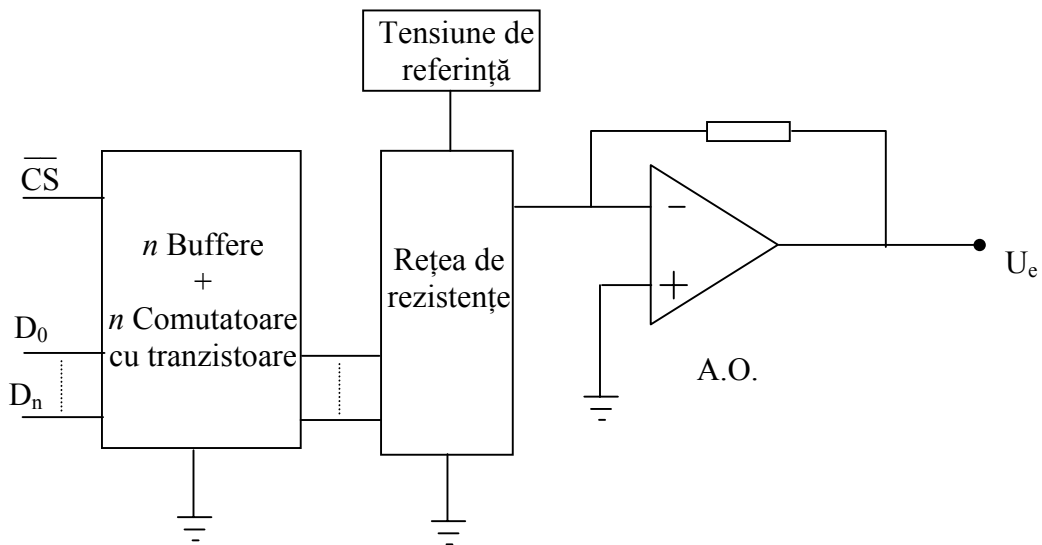
**Fig. 3.4.** Forma semnalului de ieșire din CNA (răspunsul unui extrapolator de ordin zero)

La ieșirea CNA (Fig. 3.3) se utilizează un filtru de tip trece jos care are rolul de a netezi semnalul provenit de la convertor, apoi un amplificator de putere care are rolul adaptării de nivel și de putere cu circuitele spre care este transmis semnalul analogic. În unele cazuri, la ieșirea CNA se dispune un circuit de izolare galvanică.

### 3.2.2. Tipuri de CNA

Convertoarele numeric-analogice sunt dispozitive utilizate la conversia semnalelor din forma numerică, provenite de regulă de la un sistem de calcul, în formă analogică. Semnalele numerice de intrare în CNA sunt cuvinte cu lungimi de regulă între 8 și 16 biți. Semnalele analogice de la ieșirea CNA sunt fie tensiuni continue (în gama 0 – 10 V, -10 – +10 V) fie curenți continui (de ordinul miliamperilor). CNA-urile funcționează pe baza principiului superpoziției efectelor: curentul printr-o rezistență de sarcină este dat de suma curenților care străbat o rețea de rezistențe conectate la rezistența de sarcină. La rândul lor, acești curenți sunt determinați de valorile binare 1 sau 0 ale cifrelor cuvântului de intrare în convertor, precum și de rangul acestor cifre. Rezistențele corespunzătoare cifrelor binare sunt introduse în circuit atunci când cifra binară asociată este 1 sau sunt deconectate în caz contrar, prin intermediul unor comutatoare electronice. Cele mai obișnuite tipuri de rețele sunt cele cu rezistențe ponderate și cu rezistențe  $R - 2R$ . Comutatoarele electronice se pot realiza cu tranzistoare bipolare sau cu tranzistoare cu efect de câmp.

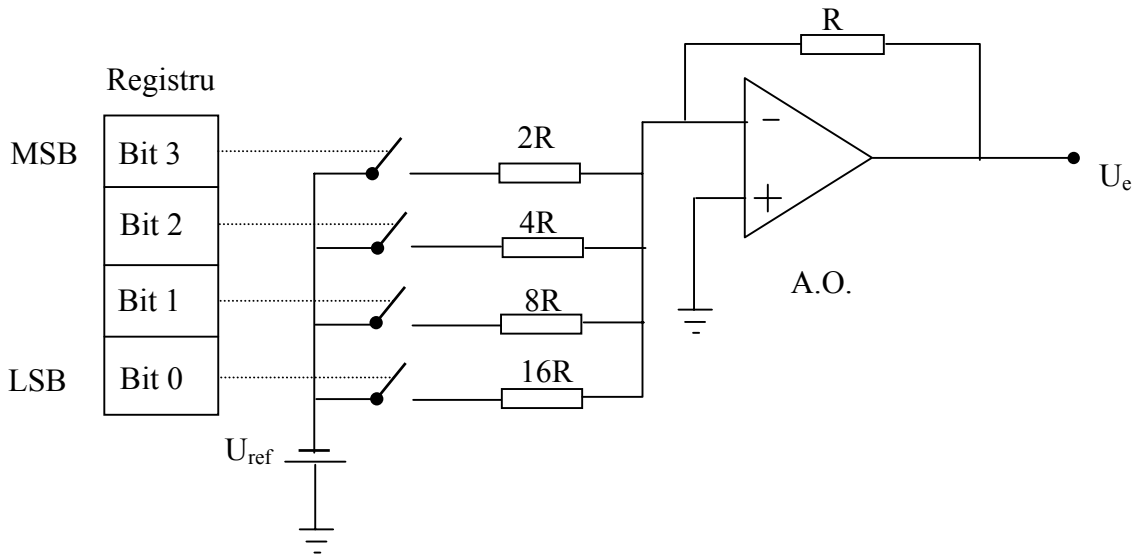
În Fig. 3.5 este prezentată schema bloc de principiu a unui CNA pe  $n$  biți.



**Fig. 3.5.** Schema bloc a unui CAN

Convertorul numeric-analogic are  $n$  buffere care au rolul de a memora valoarea actuală a intrării binare care trebuie convertită într-un semnal analogic. Ieșirile bufferelor controlează câte un comutator (cheie) care este asociat unei anumite rezistențe din rețea. O tensiune internă de referință de mare precizie este conectată la rețeaua de rezistențe pentru stabilirea domeniului tensiunii de ieșire. Amplificatorul operațional furnizează tensiunea de ieșire  $U_e$  care va depinde de acțiunea comutatoarelor care pot conecta sau deconecta rezistențe din circuit.

Pentru înțelegerea principiului de proiectare a unui astfel de convertor, în Fig. 3.6 este prezentat un CNA pe 4 biți. Se observă că registrul care conține biții de intrare controlează cele 4 comutatoare, astfel încât sunt posibile 16 configurații în funcție de poziția comutatoarelor. Amplificatorul operațional este conectat ca un circuit de însumare.



**Fig. 3.6.** Structura unui CNA pe 4 biți cu rezistențe ponderate

Pentru o anumită configurație a comutatoarelor, tensiunea de ieșire este determinată de produsul dintre tensiunea de referință și raportul dintre rezistența de pe reacție și rezistența introdusă în circuit de comutatoare. De exemplu, dacă doar MSB este 1 și ceilalți biți sunt 0 vom avea:

$$U_e = \frac{R}{2R} U_{ref} = \frac{1}{2} U_{ref}$$

Dacă bitul 2 este 1 și ceilalți sunt 0 tensiunea de ieșire va fi:

$$U_e = \frac{R}{4R} U_{ref} = \frac{1}{4} U_{ref}$$

Dacă avem MSB egal cu 1, bitul 2 și bitul 1 sunt 0 iar LSB este 1 atunci tensiunea de ieșire va fi o sumă a efectelor combinate ale comutatoarelor:

$$U_e = \frac{R}{2R} U_{ref} + \frac{R}{16R} U_{ref} = \frac{9}{16} U_{ref}$$

Cu această structură de CNA se pot obține 16 nivele de tensiune la ieșire. Pentru asigurarea unei dependențe liniare intrare-ieșire este necesar ca rezistențele să fie precise și stabile.

O altă problemă este legată de implementarea practică a structurii pe un singur cip. Cazul prezentat corespunde alegerii unor rezistențe în gama  $R - 16R$ . Valoarea unei rezistențe implementate pe un circuit integrat pornește de la valoarea realistă de  $5 \text{ k}\Omega$ . Prin urmare rezistența de  $16R$  va fi de  $80 \text{ k}\Omega$ . În cazul unui CNA pe 8 biți sunt necesare rezistențe între  $5 \text{ k}\Omega$  și  $1.28 \text{ M}\Omega$ , ceea ce este deja o problemă, deoarece spațiul pe un cip este limitat și nu se pot implementa rezistențe de valori mari. Pentru surmontarea acestei probleme se utilizează structuri de CNA bazate pe rețele de rezistențe  $R - 2R$ , care se pretează foarte bine tehnologiei de fabricare a circuitelor integrate, deoarece se repetă aceleași operații de implantare a unor rezistențe de valori mici și egale.

Un CNA pe 4 biți cu rețea  $R - 2R$ , echivalent cu cel din Fig.3.6, este prezentat în Fig. 3.7. Structura rețelei de rezistențe din Fig. 3.7 este astfel proiectată încât oricare dintre comutatoare determină același efect în tensiunea de ieșire ca și comutatorul similar din structura din Fig. 3.6.

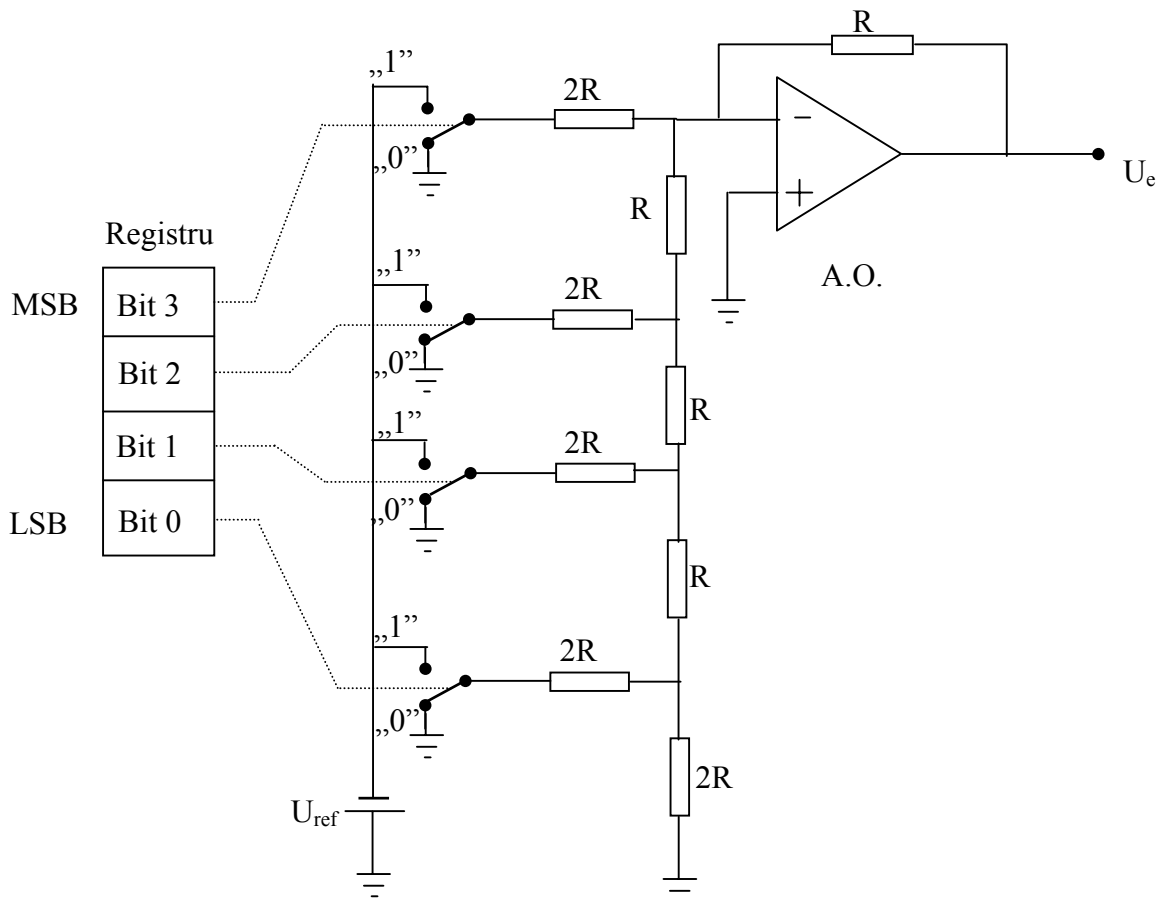


Fig. 3.7. CNA pe 4 biți cu rețea de rezistențe  $R - 2R$

În proiectarea și implementarea CNA-urilor cu circuite integrate trebuie acordată o atenție deosebită aspectelor privind legarea la masă. Această problemă este valabilă de fapt pentru toate sistemele care conțin componente analogice și numerice. Pentru reducerea zgomotelor trebuie ca traseul de masă pentru componentele analogice să fie separat în sistem de traseul de masă pentru cele numerice, urmând ca cele două trasee de masă să fie conectate într-un singur punct.

Convertoarele prezentate furnizează un semnal de ieșire al cărui semn este același cu cel al tensiunii de referință. Pentru utilizările bipolare, bitul de semn este tratat separat, fiind folosit de exemplu un inversor comandat de bitul de semn, plasat la ieșirea convertorului.

### 3.3. CONVERTOARE ANALOG-NUMERICE (CAN)

#### 3.3.1. Eșantionarea și cuantificarea semnalelor

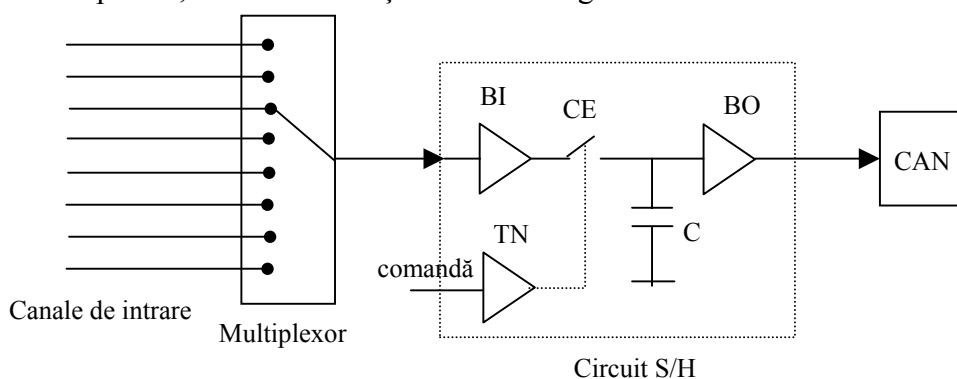
Un *convertor analog-numeric* acceptă la intrare un semnal continuu în timp  $x(t)$  sub forma unei tensiuni electrice și furnizează la ieșire o secvență de numere binare  $x(n)$ , care aproximează eșantioanele corespunzătoare  $x(nT_e)$ . Pentru majoritatea aplicațiilor, tensiunea electrică de intrare nu este furnizată direct CAN-ului, ci printr-un dispozitiv numit element de eșantionare și reținere (memorare) – *sample-hold* (S/H). Elementul S/H este un circuit analogic care măsoară semnalul de intrare în ritmul unui tact (multiplu de  $T_e$ ) și menține acest semnal fixat pentru un interval de timp suficient de mare pentru ca operațiunea de conversie analog-numerică să se poată realiza. Deoarece operațiunea de conversie analog-numerică este

relativ lentă, este important ca o eventuală variație a tensiunii de intrare să nu perturbe procesul de conversie, și prin urmare circuitul S/H previne aceste perturbații prin menținerea constantă a tensiunii de intrare pe durata desfășurării unei conversii.

Circuitele S/H sunt alcătuite de regulă dintr-un buffer urmat de un comutator electronic conectat la un condensator. Operațiunea de eșantionare și reținere (*sample and hold*) constă în următoarele etape principale:

1. La un moment de eșantionare, comutatorul electronic conectează bufferul și condensatorul la semnalul de intrare;
2. Condensatorul se încarcă cu tensiunea de intrare;
3. Sarcina condensatorului este menținută atâta timp cât CAN-ul convertește semnalul;
4. Dacă sunt mai multe canale conectate (multiplexate) la un CAN, etapele anterioare sunt repetate pentru fiecare canal de intrare;
5. Întregul proces este repetat la următorul moment de eșantionare.

Un circuit de intrare analogică utilizat pentru eșantionarea semnalelor conține de regulă un multiplexor, un circuit S/H și un CAN – Fig. 3.8.



**Fig. 3.8.**

Circuitul S/H conține bufferele de intrare și de ieșire de tip amplificator BI și BO, translatorul de nivel TN care asigură execuția corectă a comenzii de eșantionare/memorare, comutatorul (cheia) electronic CE și condensatorul de memorare C. Atunci când CE este închis circuitul este în așa-numitul regim de eșantionare sau de urmărire, iar când CE este deschis – regimul de memorare.

*Observație:* Din punct de vedere al modului de eșantionare, hardware-ul de achiziție poate realiza o eșantionare secvențială a canalelor de intrare sau o eșantionare simultană (*simultaneous sample and hold* – SS/H).

O dată realizată eșantionarea – adică preluarea eșantioanelor din semnalul de intrare, semnalul eșantionat trebuie convertit din tensiune (așa cum se află la ieșirea circuitului S/H) într-un număr binar pe care sistemul de calcul îl poate citi și prelucra. Conversia semnalului eșantionat, caracterizat de valori ale amplitudinii infinite precise, în număr binar se numește *cuantificare*. În timpul cuantificării, CAN-ul folosește un număr finit de valori pentru reprezentarea semnalului analogic. Numărul de valori este determinat de numărul de biți utilizați pentru conversie (CAN-urile moderne utilizează de regulă 12 – 16 biți).

### **3.3.2. Indicatori de performanță ai CAN**

Convertoarele analog-numerice sunt dispozitive utilizate la conversia semnalelor din formă analogică în formă numerică, în scopul prelucrării ulterioare prin mijloace numerice. Semnalele analogice de intrare în CAN sunt în mod obișnuit tensiuni continue în gama 0 – 10 V sau -10 – +10 V. Semnalele de ieșire din CAN sunt cuvinte care au de regulă un număr de biți între 6 și 22.



Cele mai des utilizate tipuri de CAN funcționează pe baza principiului compensării automate. Tensiunii de la intrare  $i$  se asociază o valoare numerică, care este după aceea convertită din nou în formă analogică. Valoarea numerică este ajustată în mod automat, astfel încât să compenseze eroarea dintre tensiunea de intrare în CAN și corespondentul analogic al mărimii numerice de la ieșire. Există mai multe tipuri de CAN, în funcție de modul de generare a valorii numerice. Printre tipurile mai des folosite se numără: CAN cu comparare multiplă, cu aproximări succesive, cu integrare (cu pantă simplă sau cu pantă dublă), cu conversie intermediară în frecvență, CAN cu filtrare digitală etc.

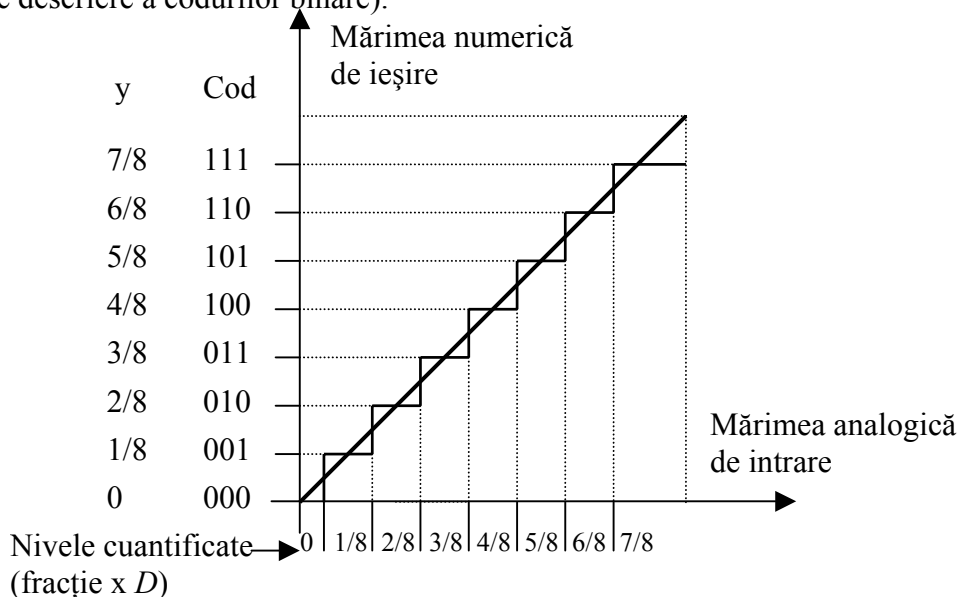
Principalele performanțe ale CAN sunt liniaritatea, precizia, rezoluția, timpul și rata de conversie, domeniul (gama) tensiunii de intrare.

Conversia analog-numerică constă în convertirea unui semnal de intrare continuu într-o fracție  $y$  prin compararea cu un semnal de referință. Mărimea de ieșire numerică din CAN este o reprezentare codificată a acestei fracții  $y$ . Dacă ieșirea CAN este pe  $n$  biți, atunci numărul nivelelor de valori discrete de ieșire este de  $2^n$ . Prin urmare, pentru a respecta o corespondență, domeniul analogic de intrare trebuie cuantificat în același număr de nivele. Fiecare astfel de nivel sau cuantă este o valoare analogică pentru care două coduri adiacente de ieșire diferă, și mai este numită dimensiune a bitului LSB:

$$LSB = D / 2^n$$

În relația de mai sus notația LSB este ușor improprie și reprezintă valoarea analogică a celui mai puțin semnificativ bit, iar  $D$  este domeniul mărimii de intrare (full scale). Toate valorile analogice de intrare din interiorul unui nivel de dimensiune  $D / 2^n$  sunt reprezentate de același cod numeric de ieșire. Deoarece semnalul de intrare poate să difere de valoarea medie a unui astfel de subinterval analogic cu cel mult  $\pm LSB / 2$  și să fie totuși reprezentat de același cod de ieșire, rezultă că există o incertitudine sau eroare de cuantificare de  $\pm LSB / 2$  în orice CAN. Efectul acestei erori poate fi redus doar prin creșterea numărului de biți ai ieșirii CAN.

Aspectele prezentate mai sus pot fi analizate cu ajutorul caracteristicii statice ideale a unui CAN pe 3 biți dată în Fig. 3.9. În acest caz, dimensiunea unui nivel de cuantificare din mărimea de intrare este de  $D / 2^3 = D / 8$  și prin urmare domeniul de intrare este împărțit în 8 subintervale distincte de la 0 la  $(7/8) \cdot D$ . Trebuie observat faptul că ieșirea maximă, adică numărul binar 111, corespunde nu domeniului maxim de intrare ci lui  $(7/8) \cdot D$  (a se vedea paragraful de descriere a codurilor binare).



**Fig. 3.9.** Caracteristica statică a unui CAN pe trei biți

CAN-urile au caracteristici statice reale, care diferă de caracteristica ideală prezentată anterior. Pe aceste caracteristici reale pot fi evidențiate tipurile de erori care pot apare la un CAN. Una dintre erori este *eroarea de offset* (sau eroarea de zero) (Fig. 3.10) care poate fi definită ca valoarea analogică egală cu intervalul de la origine la intersecția caracteristicii reale cu abscisa. O altă eroare este *eroarea de câștig* sau de pantă (Fig. 3.11) care apare datorită unei abateri de pantă a caracteristicii. Aceste două erori sunt în general mici și pot fi eliminate prin tehnici de ajustare. Ceva mai dificil de eliminat sunt *erorile de liniaritate*. Erorile de liniaritate pot fi de două tipuri: eroare totală de liniaritate (deviația maximă a caracteristicii reale de la caracteristica ideală, atunci când erorile de offset și de câștig sunt nule) - Fig. 3.12, sau eroare diferențială (deviația unui nivel de la valoarea analogică ideală) – Fig. 3.13. În cazul în care o neliniaritate de tip diferențial este mai mare de 1 LSB, atunci ieșirea are coduri lipsă.

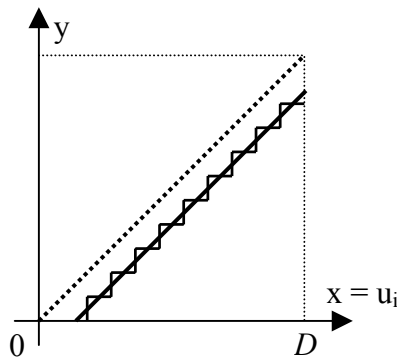


Fig. 3.10.

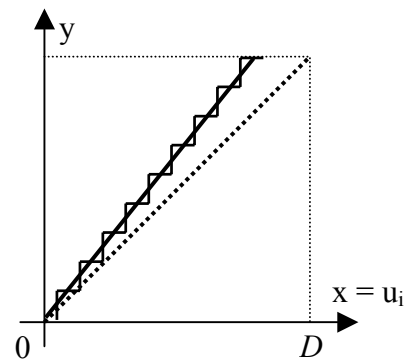


Fig. 3.11.

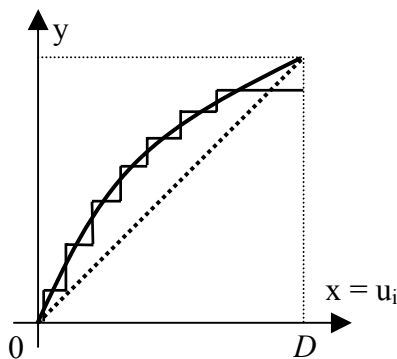


Fig. 3.12.

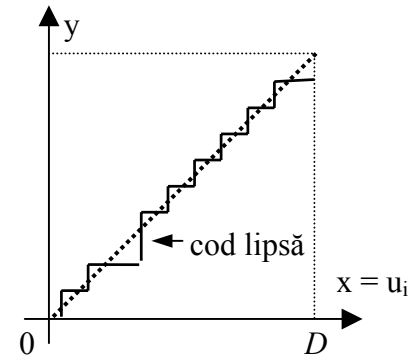


Fig. 3.13.

În afara erorilor de conversie, un alt indicator important al performanțelor CAN este *rezoluția*. Rezoluția poate fi definită ca cea mai mică schimbare a intrării analogice care determină schimbarea codului de ieșire al CAN cu un nivel. Rezoluția este definită de regulă pe caracteristica ideală și prin urmare definește mai degrabă posibilitățile CAN decât performanța reală. Rezoluția se poate defini procentual, în milivolți pentru un domeniu de intrare dat sau pur și simplu ca numărul de biți ai CAN. De exemplu, dacă avem un CAN pe 12 biți, rezoluția acestuia se spune că este de 12 biți, iar rezoluția procentuală  $r_{\%} = \frac{1}{2^{12}} \cdot 100 = 0.0244\%$ . Dacă avem un domeniu de intrare de 10 V, se poate defini rezoluția în milivolți  $r_{mV} = 10 / 2^{12} = 2.44 \text{ mV}$ , aceasta fiind de fapt tensiunea minimă de intrare ce poate fi reprezentată (a se vedea și paragraful de coduri binare).

Un alt indicator de performanță al CAN este *precizia*. Trebuie să remarcăm aici că în specificațiile tehnice și în literatura de specialitate în limba engleză se utilizează uneori termenul de *precision* pentru rezoluție și termenul *accuracy* pentru precizie. Precizia se definește ca fiind diferența dintre tensiunea de intrare reală și echivalentul analogic al codului

binar de ieșire din CAN. Atunci când este exprimată în volți se numește precizie absolută, dar de multe ori se exprimă relativ la semnalul analogic, se exprimă în LSB și se numește precizie relativă. Precizia este dată de suma maximă a tuturor erorilor de conversie, inclusiv a erorii (incertitudinii) de cuantificare.

Pentru regimul dinamic al CAN se definește  *timpul de conversie  $t_c$*  ca fiind intervalul de timp dintre startarea operației de conversie și momentul în care convertorul furnizează mărimea de ieșire finală. Inversa timpului de conversie este  *rata de conversie* care se măsoară în conversii/secundă. Utilizarea circuitelor de tip Sample/Hold între intrarea analogică și CAN conduce la apariția unui timp de întârziere suplimentar numit  *timp de deschidere (aperture time)*, de care trebuie să se țină seama în aprecierea performanțelor.

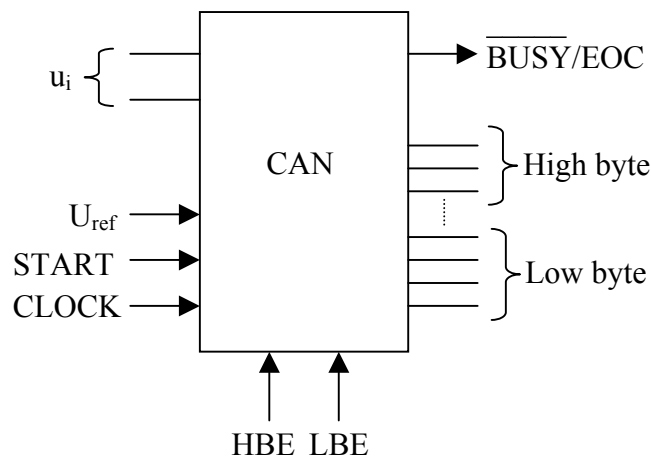
### 3.3.3. Intrări și ieșiri ale CAN

Pentru analiza tipurilor de semnale de intrare și de ieșire ale unui CAN se poate observa Fig. 3.14, care prezintă intrările și ieșirile unui CAN monolitic tipic.

**Semnalul de intrare analogic.** Majoritatea CAN sunt proiectate pentru semnale de intrare de tip diferențial sau de tip single-ended (asimetrice) unipolare. Cele mai des întâlnite domenii sunt între 0 – 10 V și 0 – 5 V. Dacă semnalul de intrare real nu acoperă întregul domeniu de intrare, o parte din codurile de ieșire ale CAN nu sunt folosite. Problema se poate corecta prin alegerea CAN-ului cu domeniul potrivit și prin scalarea semnalului de intrare utilizând amplificatoare operaționale.

Dacă semnalul analogic este bipolar, se poate utiliza tot un CAN unipolar, dar se scalează mai întâi semnalul de intrare prin adăugarea unui offset. În cazul în care se dorește o ieșire bipolară (în cod binar bipolar) se utilizează CAN bipolare.

**Semnalul de referință analogic.** CAN-urile necesită un semnal de referință analogic pentru convertirea semnalului analogic de intrare într-o fracție. Orice eroare în semnalul de referință determină o eroare de câștig a caracteristicii statice a CAN.

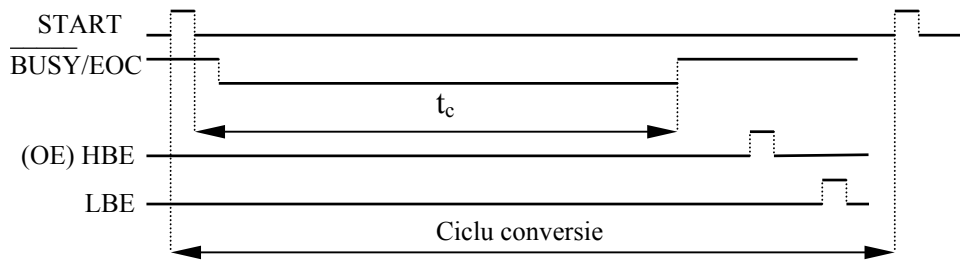


**Fig. 3.14.** Intrări și ieșiri tipice ale CAN

Prin urmare este necesar un semnal de referință  $U_{ref}$  stabil și precis. Sursele de semnal de referință sunt de regulă cuplate cu un buffer (situat uneori chiar pe cipul monolitic).

**Semnalul de ieșire.** Ieșirea numerică a unui CAN este dată de numărul de biți (deci de rezoluție) și de tipul de codificare utilizată. Cele mai obișnuite CAN sunt cele cu 8 și cu 12 biți. Codul folosit la ieșirea CAN unipolare este de obicei de tip binar natural, sau de tip zecimal codificat binar în cazul unor aparate de măsurat numerice.

**Semnalele de control.** Toate CAN au nevoie de un ceas și de semnale de control corespunzătoare. Modul de acțiune al semnalelor de control poate fi observat prin analiza unui ciclu de conversie al CAN, prezentat în Fig. 3.15.



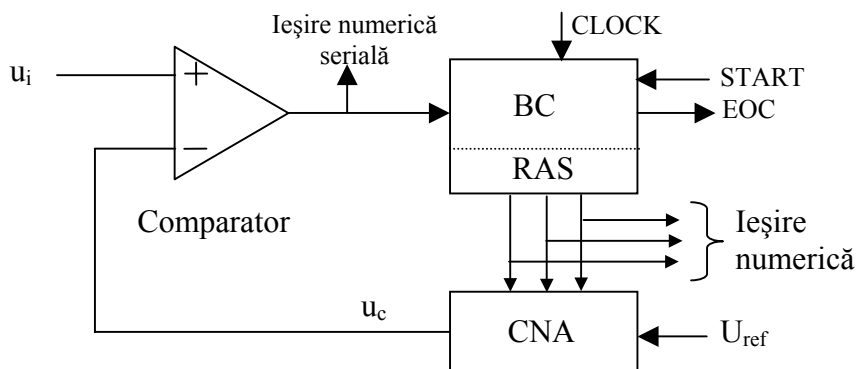
**Fig. 3.15.** Forme tipice ale semnalelor de control

Dispozitivul extern cu care este interfațat CAN (microprocesorul) inițiază conversia printr-un semnal *START*, după care CAN comută pe nivel jos linia *BUSY* sau *EOC* (End of Conversion) ceea ce înseamnă că procesul de conversie este în desfășurare. La sfârșitul unei conversii curente, CAN comută *EOC* pe nivel înalt, tranziție care generează o întrerupere în microprocesor pentru a semnala sfârșitul conversiei. Microprocesorul (dispozitivul extern) trimite un Output Enable (*OE*) către CAN pentru ca acesta să furnizeze cuvântul de ieșire. Pentru CAN de rezoluție mai mare de 8 biți, dar cuplate la o magistrală de date de 8 biți, semnalul de Output Enable constă în două semnale: High Bit Enable *HBE* și Low Bit Enable *LBE* pentru dispunerea cuvântului de ieșire pe magistrala de ieșire de 8 biți în două transferuri secvențiale.

### 3.3.4. Tehnici de conversie și tipuri de CAN

În funcție de principiul de conversie utilizat, CAN se pot clasifica în: CAN cu comparație (cum ar fi CAN cu aproximări succesive, CAN de tip paralel sau *Flash*), CAN cu integrare (cu pantă simplă, cu pantă dublă, cu integrare și conversie intermediară în frecvență), CAN cu filtrare digitală. În continuare vor fi prezentate principalul câteva din aceste tipuri de CAN.

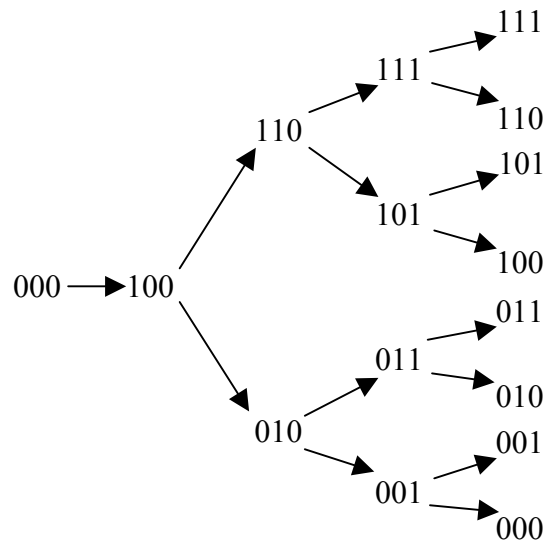
**CAN cu aproximări succesive.** Este unul din cele mai utilizate CAN pentru rezoluții medii și mari. Acest CAN se bazează pe aproximarea semnalului de intrare analogic cu un cod binar, convertirea acestui cod binar cu un CNA într-un semnal analogic care este comparat cu semnalul de intrare și apoi revizuirea succesivă a aproximării pentru fiecare bit din cod până când se obține cea mai bună aproximare. La fiecare pas din acest proces, valoarea binară curentă a aproximării este stocată într-un registru numit registru de aproximări succesive RAS. Fig. 3.16 prezintă structura bloc a unui CAN cu aproximări succesive pe trei biți.



**Fig. 3.16.** Structura unui CAN cu aproximări succesive

În Fig. 3.16, BC este un bloc de comandă, care controlează conversia și mărimea de intrare în CNA. Schema de conversie se bazează pe principiul de înjumătățire a intervalului și poate fi prin adăugare sau scădere.

Un ciclu de conversie începe de la valoarea 000 a mărimii de intrare în CNA, apoi bitul MSB este făcut 1. Urmează un algoritm de testare care va verifica efectul modificării câte unui bit. Prin urmare, la primul pas convertorul numeric-analogic va converti valoarea 100 (care corespunde unei estimări inițiale de jumătate din domeniul de intrare) într-o tensiune  $u_c$  care este comparată cu tensiunea de intrare. În funcție de rezultatul comparației, bitul MSB este lăsat 1 sau este făcut 0. La pasul al doilea (la următorul tact sau perioadă de ceas), al doilea bit este făcut 1, urmează conversia în  $u_c$  și compararea cu tensiunea de intrare. Se decide dacă cel de-al doilea bit este 1 sau 0 și urmează al treilea pas. În cazul exemplului nostru este vorba de LSB care este făcut 1, se realizează conversia și comparația și se stabilește valoarea finală a LSB și prin urmare a codului de ieșire care va aproxima cel mai bine semnalul de intrare. Schema logică a acestui algoritm (cu adăugare) cu toate opțiunile posibile este prezentată în Fig. 3.17.



**Fig. 3.17.** Schema logică a algoritmului de testare

Deoarece deciziile pentru fiecare bit sunt luate serial, poate fi furnizată o ieșire numerică serială la ieșirea din comparator.

Ca performanțe, aceste CAN au timpul de conversie independent de mărimea de intrare (este o sumă dintre timpul de inițializare și  $n \cdot T_{bit-test}$ ) și este de ordinul 100 nsec – 5 μsec pentru CAN pe 8 biți și mai mare pentru CAN pe mai mulți biți. Rezoluția este de 6 – 16 biți.

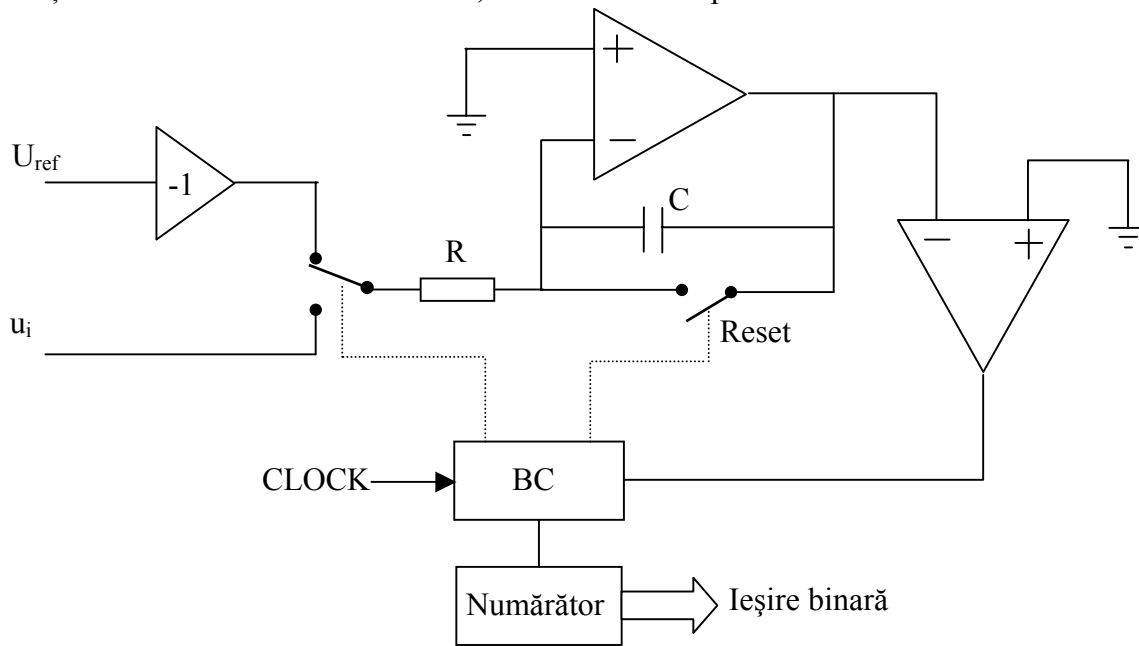
**CAN cu integrare cu pantă dublă.** Tehnica utilizată în cadrul acestor CAN este ilustrată de schema bloc din Fig. 3.18 și de diagrama temporală din Fig. 3.19. Tensiunea de intrare este integrată pe un interval fix de timp  $T_1$  care în general corespunde numărului maxim furnizat de numărătorul din cadrul CAN. La sfârșitul acestui interval, numărătorul este resetat și intrarea integratorului este comutată pe tensiunea de referință negativă. Ieșirea integratorului va descrește liniar până la zero, moment în care numărătorul este oprit și integratorul este resetat. Sarcina de pe condensatorul integratorului acumulată în primul interval  $T_1$  trebuie să fie egală cu pierderea de sarcină din al doilea interval  $T_2$  și prin urmare vom avea:

$$T_1 \cdot u_{i(mediu)} = T_2 \cdot U_{ref} \Rightarrow \frac{T_2}{T_1} = \frac{u_{i(mediu)}}{U_{ref}} = y$$

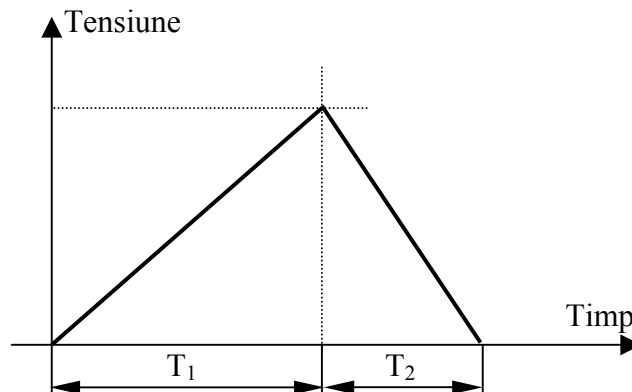
Fracția  $y$  este raportul celor două intervale de timp și este de asemenea ieșirea binară a numărătorului în raport cu numărul maxim furnizat de acest numărător. Prin urmare numărul

(ieșirea număratorului) de la sfârșitul celui de-al doilea interval  $T_2$  este chiar cuvântul binar de ieșire din CAN. Schema prezentată poate fi adaptată la diferite codificări binare.

Tehnica de tip pantă dublă oferă mai multe avantaje, cum ar fi rejecția foarte bună a zgomotelor și faptul că variațiile în frecvența ceasului nu afectează rezoluția, care este determinată doar de performanțele componentelor analogice utilizate și nu de neliniaritățile diferențiale ale convertorului. Se poate obține ușor o rezoluție bună ajustând dimensiunea număratorului și frecvența de ceas. Aceste tipuri de CAN au o rezoluție foarte bună (12 - 18 biți). Dezavantajul principal este viteza redusă (timp de conversie mare). Se utilizează în aplicații cum ar fi multimetre numerice, la măsurarea temperaturii etc.



**Fig. 3.18.** Structura unui CAN cu pantă dublă

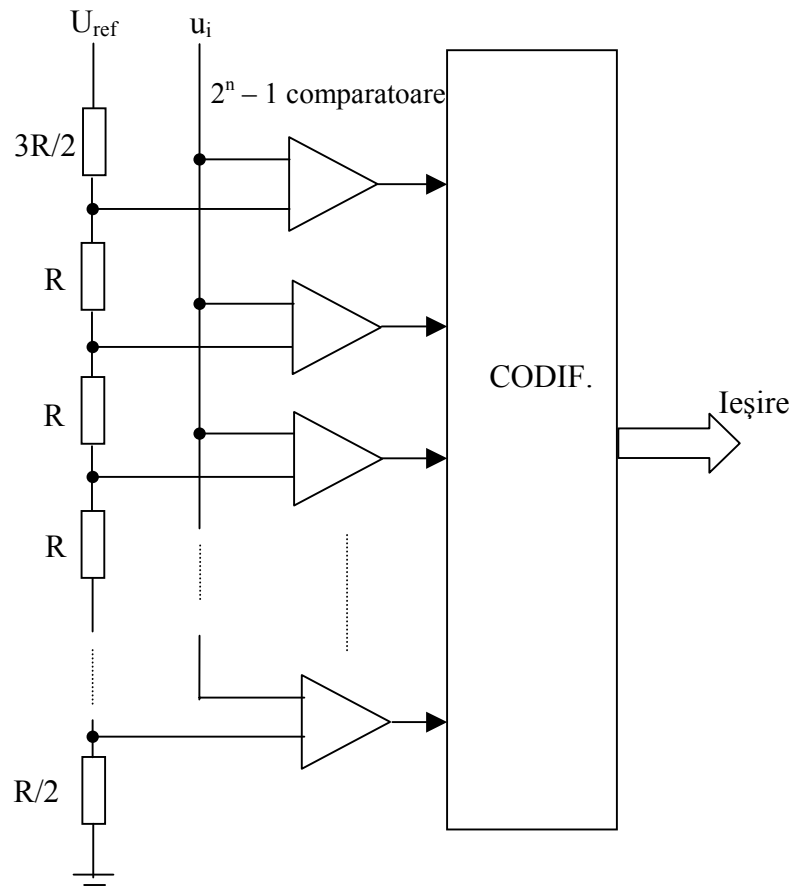


**Fig. 3.19.** Diagrama temporală

**CAN de tip paralel (Flash).** Conversia de tip paralel sau Flash este cea mai rapidă tehnică de conversie. Din analiza caracteristicii statice a unui CAN se poate observa că mărimea de ieșire poate fi determinată dacă se prelucrează ieșirile unor comparatoare care compară poziția mărimii de intrare față de valorile de frontieră ale subintervalelor în care este împărțit domeniul de intrare. Sunt necesare  $2^n - 1$  comparatoare, ceea ce limitează rezoluția acestor CAN.

Schema de principiu este prezentată în Fig. 3.20. Semnalul de intrare este comparat simultan cu toate valorile medii ale subintervalelor în care este împărțit domeniul de intrare,

prin utilizarea unor comparatoare deplasate ca domeniu unul de altul cu 1 LSB (dimensiunea analogică a subintervalului). Această deplasare dintre comparatoare este realizată prin intermediul unei tensiuni de referință și a unei rețele de rezistențe de precizie. Toate ieșirile comparatoarelor peste semnalul de intrare vor fi făcute *off*, iar cele sub semnalul de intrare *on*. Deoarece toate comparatoarele își schimbă starea simultan, procesul de conversie este realizat într-un singur pas. Un codificator convertește ieșirea comparatoarelor într-o descriere binară pe  $n$  biți. Viteza acestor CAN este foarte mare, timpul de conversie ajungând la ordinul sutimilor de microsecundă. Dezavantajul este legat de rezoluție care este de maxim 10 biți. Aplicațiile acestor CAN sunt în domeniul video, radar, osciloscopia numerică.



**Fig. 3.20.** Schema principală a unui CAN paralel

#### *Selectarea unui CAN*

Pentru a selecta un CAN conform aplicației date trebuie cunoscute mai întâi performanțele care se cer, cum ar fi rezoluția, timpul de conversie, erori permise etc. Acestea sunt determinate de specificațiile de proiectare ale sistemului respectiv, cum ar fi:

- numărul canalelor analogice
- rata minimă de conversie
- localizarea traductoarelor
- precizia de conversie
- condițiile de interfață (serie/paralel) și protocoale
- mediul de lucru (interesează zgomotele electrice și domeniul de temperatură al mediului)
- prețul sistemului