



(12) CERERE DE BREVET DE INVENȚIE

(21) Nr. cerere: a 2010 00377

(22) Data de depozit: 29.06.2010

(41) Data publicării cererii:  
30.11.2011 BOPI nr. 11/2011

(71) Solicitant:  
• COJOC WISERNIG EDUARD,  
STR. MOROIENI NR.28, SINAIA, PH, RO;  
• TOMA ANDREI, STR. SF.MARIA NR.2,  
BL.9A, SC.1, ET.2, AP.29, SECTOR 1,  
BUCUREȘTI, B, RO

(72) Inventatori:  
• COJOC-WISERNIG EDUARD,  
STR.MOROIENI NR.28, SINAIA, PH, RO;  
• TOMA ANDREI, STR. SF.MARIA NR.2,  
BL.9A, SC.1, ET.2, AP.29, SECTOR 1,  
BUCUREȘTI, B, RO

(54) ANSAMBLU DE CONTROL AL UNUI DISPOZITIV DIGITAL  
PRIN COMANDĂ ÎN LIMBAJ, PSEUDO- NATURAL FLEXIBIL

(57) Rezumat:

Prezenta invenție se referă la un ansamblu de control al unui dispozitiv digital prin comandă vocală. Ansamblul de control (100), conform invenției, permite interacțiunea dintre un utilizator (101) și un dispozitiv (160), prin intermediul unei platforme (300) de control vocal și al unei platforme (400) de control hardware, folosind, pentru comunicare om- mașină, limbajul Speak (500), iar pentru comunicare mașină-mașină, limbajul Squeak (600).

Revendicări: 9  
Figuri: 6

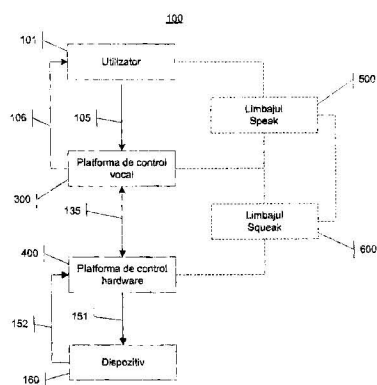
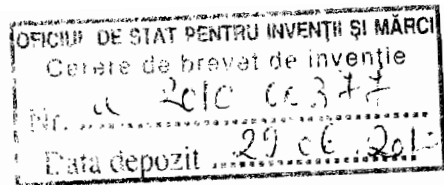


Fig. 1





# Ansamblu de control al unui dispozitiv digital prin comanda vocala in limbaj pseudo-natural flexibil

---

## Domeniul tehnic

Inventia de fata se incadreaza in domeniul metodelor de control al dispozitivelor digitale si in special in subdomeniile recunoasterii vocale, interpretarii sintactico-semantic (a unui limbaj pseudo-natural), comunicarii intre mai multe dispozitive prin utilizarea unui cod masina virtual universal.

## Problema tehnica

Problema pe care o abordeaza acest ansamblu de control este cea a controlului unui dispozitiv - controlabil electronic. Aceasta problema poate fi incadrata in mai mult sub-probleme astfel:

1. Comunicare om-masina. Problema comunicarii om-masina este abordata prin diverse interfete: mecanice, electrice, electronice, grafice, vocale etc.
2. Recunoastere vocala. In cazul interfetelor vocale, apare problema recunoasterii vocale adica a transformarii unui sunet (i.e. comanda vocala inregistrata) in text adica reprezentarea „scrisa” a sunetului. Aceasta problema
3. Comunicare masina-masina. Dispozitivul controlat poate avea capacitatea de a sustine un motor de recunoastere vocala, insa in majoritatea cazurilor nu o are. Din acest motiv apare necesitatea comunicarii masina-masina adica intre dispozitivul care indeplineste rolul de comunicare cu omul si dispozitivul tinta care se doreste controlat.

## Solutia tehnica

Solutia pe care o propune aceasta inventie este un ansamblu de control (100) format dintr-o platforma de control vocal (300) care are sarcina de a comunica cu utilizatorul si o platforma de control hardware (400) care face managementul functionalitatilor dispozitivului.

Sistemul de functionare al inventiei este modular, fiind impartit in module si sub-module, atat fizice cat si conceptuale. Pe platforma de control vocal utilizatorul foloseste limbajul Speak (500) pentru a formula enunturi; aceste enunturi sunt captate (205), transformate in text (305) si analizate (318), iar pe baza acestei interactiuni unitatea centrala de procesare (320) interactioneaza cu platforma de control hardware (400) prin intermediul modulului de comunicatie (329), a limbajului Squeak (600) si a modelului matematic si functional aferent.

Platforma de control hardware (400) are autonomie functionala dar este subordonata comenzilor platformei de control vocal (300), iar comunicarea intre acestea se realizeaza prin intermediul modulului de comunicatie (442). Unitatea centrala de procesare (450) face managementul resurselor hardware ale dispozitivului prin intermediul actuatorilor de stare (455).

## Avantajele inventiei

**Omogenitate tehnologica.** Solutii distincte pentru unele domenii ale inventiei exista precum recunoastere vocala, comunicare om-masina sau comunicare masina-masina, insa principalul avantaj este oferirea unei solutii omogene care sa incorporeze toate acestea. Acesta este avantajul cu impactul cel mai mare asupra stadiului actual al tehnicii.

**Interactiune optima.** O prima urmare a aceste implementarii omogene este asigurarea unei optimizari eficiente a interactiunii dintre module, limbajele Speak si Squeak fiind bazate pe acelasi model matematic generativ.

**Limbaje simetrice.** Limbajele Speak si Squeak au forme diferite insa sunt bazate pe acelasi model matematic. Astfel cele doua limbaje sunt reductibile unul la celalalt. Echivalenta acestora permite compilarea din Speak in Squeak (si invers, decompilarea) cu eforturi minime de calcul.

Limbajul Speak (500) – limbaj de control vocal in forma pseudo-naturala flexibila. Avantajul limbajului Speak este utilizarea unui mediu de comunicare om-masina bazat pe urmatoarele caracteristici:

*- forma pseudo-naturala pastreaza asemanarea cu limbajul natural pentru o usoara utilizare de catre utilizatori comuni*

*- structura limbajului este determinista si independenta de context minimizand efortul de analiza*

*- structura enunturilor este dinamica, orientata pe comanda, astfel reducand substantial efortul de analiza in comparatie cu metodele de analiza a limbajului natural*

*- limbajul permite atat utilizarea comenzilor atomice (i.e. control asupra la starilor) si a structurilor de control (e.g. daca, cat timp etc.), cat si definirea de cuvinte noi in dictionar – comenzi compuse (e.g. „defineste aprinde lumina ca seteaza iesirea 1 cu 1”), dupa bunul plac al utilizatorului.*

Limbajul Squeak (600) – limbaj de comunicare (i.e. control) masina-masina. Avantajul limbajului Squeak este utilizarea unui mediu de comunicare intre dispozitive bazat pe urmatoarele caracteristici:

*- orice dispozitiv este redus la un model matematic bazat pe stari si comenzi*

*- structura limbajului si implementarii Squeak permite comunicarea intre doua dispozitive fundamental diferite fara drivere, numai prin corelarea de stari interne. Toate comenzile Squeak functioneaza identic pe orice dispozitiv.*

**Autonomie usor de controlat.** Folosind limbajul Squeak, oricarui dispozitiv i se pot programa comenzi automate si reactii la diversi factori (observabili prin senzori). Pe langa capacitatea de autonomie, trebuie remarcata si usurinta cu care se poate reprograma, folosind limbajul Speak pentru a trimite o comanda Squeak.

## Descrierea succinta a desenelor

Inventia de fata este ilustrata prin exemple si nu limitativ in figurile urmatoare:

FIG. 1. este o schema generala a functionalitatilor inventiei

FIG. 2. este o schema de interactiune a componentelor electronice

FIG. 3. este o diagrama conceptuala a platformei de control vocal

FIG. 4. este o diagrama conceptuala a platformei de control hardware

FIG. 5. este o colectie de modele structurale de reprezentare in limbajul Speak

FIG. 6. este o colectie de modele structurale de reprezentare in limbajul Squeak

### Descrierea detaliata a deseneilor

In schema generala (100) din FIG. 1 este ilustrata interactiunea modulelor, pornind de la utilizatorul (101) care comunica platformei de control vocal (300) o comanda (105) dupa structura limbajul Speak (500). Platforma (300) analizeaza aceasta comanda si comunica cu platforma de control hardware (400) prin intermediul unei implementari (135) a limbajului Squeak (600). Platforma (400) comunica cu dispozitivul (160) prin comenzi (151) si prin raspunsuri (152).

Schema de interactiune a componentelor electronice din FIG. 2 prezinta fluxul informatiei in interiorul sistemului electronic folosit pentru implementarea inventiei. Elementele de captare de sunet (205) inregistreaza o comanda vocala de la utilizator (101); convertorul analog-numeric (211) prelucreaza comanda vocala si o transforma in format digital, pentru a fi prelucrata de encoder/decoder (215). Elementele de redare (206) sunt folosite pentru a reda informatiile destinate utilizatorului, preluand sunetul de la convertorul numeric-analog (212).

In functie de implementare, se poate folosi un chip de recunoastere vocala (217). In cazul in care nu se foloseste un asemenea chip, microprocesorul (220) preia aceasta functie.

Microprocesorul (220) este unitatea centrala de procesare (320) si (in functie de implementare) indeplineste rolul de sistem de operare a placii de control vocal sau ruleaza aplicatia de management al controlului vocal in cazul in care este executata intr-un sistem de operare generic.

Microprocesorul (220) foloseste transceiverul (230) pentru a comunica cu microcontrollerul placii de control hardware (250) prin intermediul transceiverului atasat (241).

Principalul rol al microcontrollerul (250) este de a face managementul resurselor hardware ale sistemului (240) si de a comunica cu placa de control vocal (210). Pentru a realiza managementul resurselor, sunt utilizate elemente de comanda (251) si senzori (252).

In FIG. 3 gasim diagrama conceptuala a platformei de control vocal. Componentele care au contact direct cu utilizatorul (101) sunt motorul de recunoastere vocala (305) si motorul de sintetizare vocala (306). Motorul de recunoastere vocala (305) preia sunetul primit de la utilizator (101) si prin diverse metode care variaza in functie de implementare, transforma sunetul in format text. Acest text este analizat sintactic si semantic de motorul de analiza sintactica si semantica (318). Aceasta analiza se face prin raportarea la modelul limbajului Speak (500).

Unitatea centrala de procesare (320) primeste textul analizat sub forma unei comenzi. Aceasta comanda este interpretata, schimbând stari interne sau externe. In urma schimbarilor facute, daca este cazul, motorul de sincronizare de stari (324) comunica cu platforma de control hardware si transmite comenzi prin modulul de comunicatie (329) compiland informatia dupa o implementare Squeak (326, 600).

Motorul de sincronizare de stari (324) lucreaza dupa o metoda de corelare de stari (325) adaptata la modelul limbajului de comunicatie Squeak (600).

FIG. 4 reprezinta diagrama conceptuala a platformei de control hardware (400). Similar platformei de control vocala (300), modulul de comunicatie (442), compilatorul Squeak (445) si executorul/interpretorul Squeak (446) realizeaza transmisia de date intre cele doua platforme (300,400).

Motorul de sincronizare de stari (449) implementeaza metoda de corelare de stari (325), bazata pe limbajul Squeak (600) pentru a accepta/executa comenzi si trimite notificari de stare prin intermediul modulului de comunicatie (442).

Unitatea centrala de procesare (450) realizeaza managementul tuturor resurselor platformei de control hardware, ruland un sistem de operare dedicat (nativ Squeak) sau un sistem de operare pe care ruleaza o masina virtuala Squeak.

Unitatea centrala de procesare (450) implementeaza o versiune a limbajului Squeak (600); o componenta necesara a acestui limbaj este stiva de subrutine stocate (451). In aceasta stiva se stocheaza si apeleaza subrutinele folosite de unitatea centrala de procesare (450).

Actuatorii de stare (455) mediaza comunicarea cu elementele de comanda (251) si senzorii (252).

In FIG. 5 avem o colectie de modele structurale de reprezentare a limbajului Speak. Unitatea structurala si functionala a limbajului este comanda. Orice act de vorbire nativ (se pot utiliza alte acte de vorbire, dar prin reductie la comanda) este o comanda (510). Orice comanda are un cuvânt cheie sau o sintagma (501) care precede comanda (e.g. seteaza in „seteaza iesirea 0 cu 1”), urmat de parametrii (502) si particule optionale de comanda (503).

In exemplul „seteaza iesirea 1 cu 0” avem comanda *seteaza... cu...*, cuvântul cheie fiind *seteaza* iar particula optionala *cu*. Parametrii sunt *iesirea, 1 si 0*.

Limbajul Speak are doua niveluri: nivelul atomic si nivelul complex. La nivelul atomic, se pot formula propozitii utilizand numai comenzi de control de stare (e.g. „seteaza variabila 1 cu 12”) si structuri de control (e.g. „daca variabila 1 este 12 atunci seteaza iesirea 1 cu 0”).

La nivelul complex se poate extinde dictionarul de termeni prin definire de cuvinte noi (e.g. „defineste aprinde lumina ca seteaza iesirea 1 cu 1”). Odata definiti termenii noi, se pot folosi in acelasi mod ca si termenii nativi (e.g. „Daca intrarea 1 = 0 atunci aprinde lumina altfel ctinge lumina”).

Colectia de modele structurale din FIG. 6 reprezinta reprezentarile structurilor fundamentale ale limbajului Squeak. Limbajul Squeak este un limbaj cod masina virtual universal care permite

comunicarea între două dispozitive fără a utiliza drivere. Folosind numai un acord de sincronizare a stărilor interne, oricare două dispozitive dotate cu Squeak pot comunica între ele.

O instrucțiune sau comandă are structura ilustrată (610), fiind constituită dintr-o comandă (601) și parametrii comenzii (602). De exemplu, pentru a seta portul 0 cu valoarea 1, se folosește comanda 01 00 01.

Modelul Squeak (600) este proiectat pentru a permite implementarea și controlul facil al unui comportament autonom al dispozitivului.

Stiva de subrutine (630) are rol de stocare și apel al subrutinelor utilizate.

De exemplu, putem programa un ansamblu format dintr-un bec și un senzor să se aprindă numai când senzorul detectează un om. Pentru aceasta trebuie să stocăm o subrutină care să aprindă becul (01 01 01, la adresa 0B) și o subrutină care să stingă becul (01 01 00 la adresa 0A). Apoi stocăm structura de control într-o subrutină: 04 8A 00 FF 0A 0B la adresa 0C (dacă portul 8A este 255 atunci execută 0A, altfel 0B). Pentru a realiza autonomia aparatului punem într-o buclă infinită execuția subrutinelor precedente (stocată la 0C).

## Termeni utilizati

Nr.	Descriere
100	Ansamblu de control
101	Utilizator
105	Comanda vocala
106	Raspuns audio
135	Metoda de transmisie
151	Interactiune cu dispozitivul – comanda
152	Interactiune cu dispozitivul – raspuns
160	Dispozitivul controlat
200	Sistem de dispozitive electronice de control
205	Elemente de captare de sunet
206	Elemente de redare
210	Placa de control vocal
211	Convertor analog-numeric
212	Convertor numeric-analog
215	Encoder/Decoder
217	Chip de recunoastere vocala
220	Microprocesor
230	Transceiver
235	Mediu de transmisie
240	Placa de control hardware
241	Transceiver
250	Microcontroller
251	Elemente de comanda
252	Senzori
300	Programul de control
305	Motor de recunoastere vocala
306	Motor de sintetizare vocala
318	Motor de analiza sintactica si semantica
320	Unitate centrala de procesare
324	Motor sincronizare stari
325	Metoda corelare stari
326	Compiler Squeak
327	Executor/interpretor squeak
329	Modul de comunicatie
400	Sistem de operare minimal
442	Modul de comunicatie
445	Compiler Squeak
446	Executor Squeak
449	Motor sincronizare stari
450	Unitate centrala de procesare
451	Stiva de subrutine
455	Actuatori de stare
500	Limbajul Speak
501	Comanda
502	Parametru
503	Particula optionala de comanda
510	Structure unei comenzi in Speak
600	Limbajul Squeak
601	Comanda

- 
- 602** Parametru
  - 610** Structura unei comenzi in Squeak
  - 620** Reprezentarea porturilor de intrare/iesire
  - 630** Reprezentarea stivei de subrutine
-



## Revendicari

1. Ansamblu de control al unui dispozitiv digital prin comanda vocală în limbaj pseudo-natural flexibil (100), **caracterizat prin aceea ca** ansablul (100) cuprinde sistemul de dispozitive electronice (200) si o metoda de control al unui dispozitiv prin comanda vocala (100), o metoda de comunicare om-masina in limbaj pseudo-natural (300) - limbajul Speak (500) si o metoda de comunicare masina-masina printr-un limbaj masina virtual (400) – limbajul Squeak (600).
2. Conform revendicarii 1, sistem de dispozitive electronice (200), **caracterizat prin aceea ca** este consuit din placa de control vocal (210) si placa de control hardware (240).
3. Conform revendicarii 2, placa de control vocal (210) , **caracterizata prin aceea ca** are urmatoarele elemente si functionalitati:
  - element de captare de sunet (205) care este fie un microfon, fie o instalatie capabila sa inregistreze sunete de la utilizator (101),
  - elemente de redare (206) care sunt fie audio, video sau tactile,
  - convertor analog-numeric (211) care preia sunetul inregistrat (205) si il reprezinta in forma digitala,
  - encoder/decoder (215) care codifica/decodifica sunetul digital primit/trimis in/din formatul utilizat,
  - chip de recunoastere vocala (217) este o componenta optionala care poate in deplini sarcina de motor de recunoastere vocala (305),
  - microprocesor (220) care are rolul de unitate de procesare centrala a sarcinilor placii (210), in cazul in care nu se foloseste un chip de recunoastere vocala (217), microprocesorul (220) preia sarcina de motor de recunoastere vocala (305),
  - transceiver (230) o componenta electronica folosita pentru a comunica cu placa de control hardware (240) prin intermediul transceiverului corespondent (241).
4. Conform revendicarii 2, placa de control hardware (240), **caracterizata prin aceea ca** are urmatoarele elemente si functionalitati:
  - transceiver (241) care are aceeasi functie ca (230),
  - microcontroller (250) care are rolul de unitate de procesare centrala a sarcinilor placii (240),
  - elemente de comanda (251) care primesc sarcini de la microcontroller (250) si comanda functionalitatile dispozitivului controlat,
  - senzori (252) care trimit date despre stari externe microcontrollerului (250)
5. Conform revendicarii 1, sistemul de dispozitive electronice (200), **caracterizat prin aceea ca** poate fi constituit dintr-o singura placa electronica in loc de doua (revendicarea 2); pentru aceasta microprocesorul (220) poate prelua sarcinile microcontrollerului (250) astei componente (230) (235) si (241) nemaifiind necesare.
6. Conform revendicarii 1, metoda de control a unui dispozitiv prin comanda vocala (100), **caracterizat prin aceea ca** un utilizator (101) foloseste limbajul Speak (500) pentru a formula enunturi destinate platformei de control vocal (300): enunturile sunt interpretate ca niste sarcini pe care le indeplineste fie intern - platforma de control vocal (300), fie extern - platforma de control hardware (400); pentru sarcinile externe platforma de control vocal (300) si platforma de control hardware (400) comunica prin intermediul limbajului Squeak (600);

limbajul Squeak (600) este un limbaj cod masina virtual dezvoltat pe un model matematic minimal sub care pot fi reprezentate functiile fundamentale ale oricaror doua masini, necesitand astfel numai o diagrama de reprezentare a starilor interne platforma de control hardware (400) interactioneaza bidirectional cu dispozitivul (151,152)

7. Conform revendicarii 1, metoda de comunicare om-masina in limbaj pseudo-natural (300), **caracterizata prin aceea ca are urmatoarele componente structurale care functioneaza astfel:**
- motor de recunoastere vocala (305) care preia sunetul de la elementele de captare sunet (205) si transmite o reprezentare a sunetului in format text motorului de analiza sintactica si semantica (318);
  - motorul de analiza sintactica si semantica (318) face o analiza intai sintactica apoi semantica a textului primit de la motorul de recunoastere vocala (305) pe baza specificatiei limbajului Speak (500) si trimite rezultatele analizei unitatii centrale de procesare (320);
  - unitatea centrala de procesare (320) este nucleul structural si functional al intregului subsistem (300) si executa operatiile de procesare
  - motorul de sincronizare de stari (324) realizeaza sincronizarea reprezentarii starilor interne cu reprezentarile starilor externe prin metoda de corelare a starilor (325);
  - metoda de corelare a starilor (325) este o metoda de corelare a reprezentarilor starilor interne/externe si a comenzilor a doua dispozitive digitale, prin intermediul limbajului Squeak (600) si a paradigmei aferente de reprezentare;
  - compilatorul (326) si executorul/interpretorul Squeak (327) sunt componente conceptuale care mediaza interactiunea dintre motorul de sincronizare stari (324) si modulul de comunicare (329);
  - modulul de comunicare (329) face managementul schimbului de informatii dintre placa de control vocal (210) si placa de control hardware (240)
8. Conform revendicarii 1, metoda de comunicare masina catre masina printr-un limbaj masina virtual, **caracterizata prin aceea ca are urmatoarele componente structurale care functioneaza astfel:**
- modulul de comunicare (442) face managementul schimbului de informatii cu alte dispozitive;
  - compilatorul (445) si executorul/interpretorul Squeak (446) sunt componente conceptuale care mediaza interactiunea dintre motorul de sincronizare stari (449) si modulul de comunicare (442);
  - motorul de sincronizare de stari (449) realizeaza sincronizarea reprezentarii starilor interne cu reprezentarile starilor externe prin metoda de corelare a starilor (325);
  - metoda de corelare a starilor (325) este o metoda de corelare a reprezentarilor starilor interne/externe si a comenzilor a doua dispozitive digitale, prin intermediul limbajului Squeak (600) si a paradigmei aferente de reprezentare;
  - unitatea centrala de procesare (450) este nucleul structural si functional al intregului subsistem (400) si realizeaza managementul resurselor de calcul; aceasta (450) interactioneaza cu stiva de subrutine stocate (451);
  - stiva de subrutine stocate (451) este o componenta de stocare necesara implementarii unei masini virtuale in cadrul unitatii centrale de procesare (450) pentru a rula limbajul Squeak (600)
  - actuatorii de stare (455) sunt componente conceptuale care fac managementul elementelor de comanda (251) si a senzorilor (252).

9. Conform revendicarii 6, metoda de transformare, corelare a starilor si echivalenta a limbajelor Speak (500) si Squeak (600), **caracterizata prin aceea ca** enunturile comunicarii om-masina exprimate in Speak sunt echivalate cu comenzi Squeak, in cod masina virtual universal:

Speak utilizeaza ca unitate structurala si functionala comanda; structura oricarei comenzi este ilustrata in sectiunea (510);

suporta extinderea dictionarului limbajului prin definirea de cuvinte noi;

exista doua tipuri de comenzi, cele atomice, care au corespondenta directa in Squeak (600) si cele complexe (care sunt definite drept cuvinte noi) – acestea sunt descompuse recursiv pentru a extrage comenzile Squeak

datorita modelului matematic al limbajului Squeak (600), doua dispozitive care ruleaza acest limbaj pot comunica fara a necesita drivere, numai prin corelare de stari si transmisie de comenzi

structura comenzilor interne in Squeak este ilustrata in sectiunea (610)

Squeak utilizeaza o stiva de subrutine conform diagramei (630)

Squeak utilizeaza o reprezentare interna a porturilor de intrare/iesire (620)

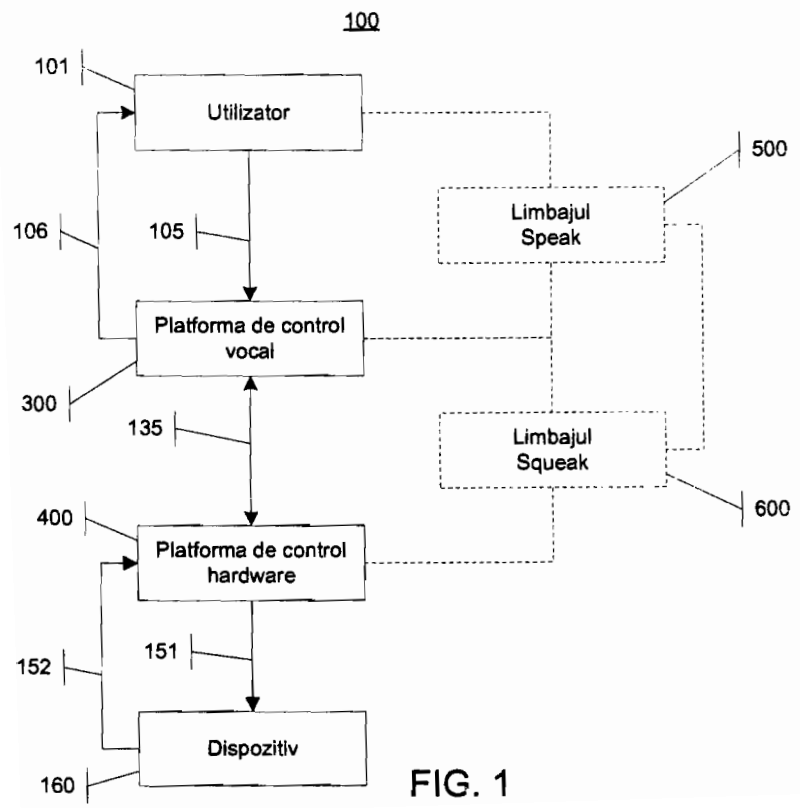


FIG. 1

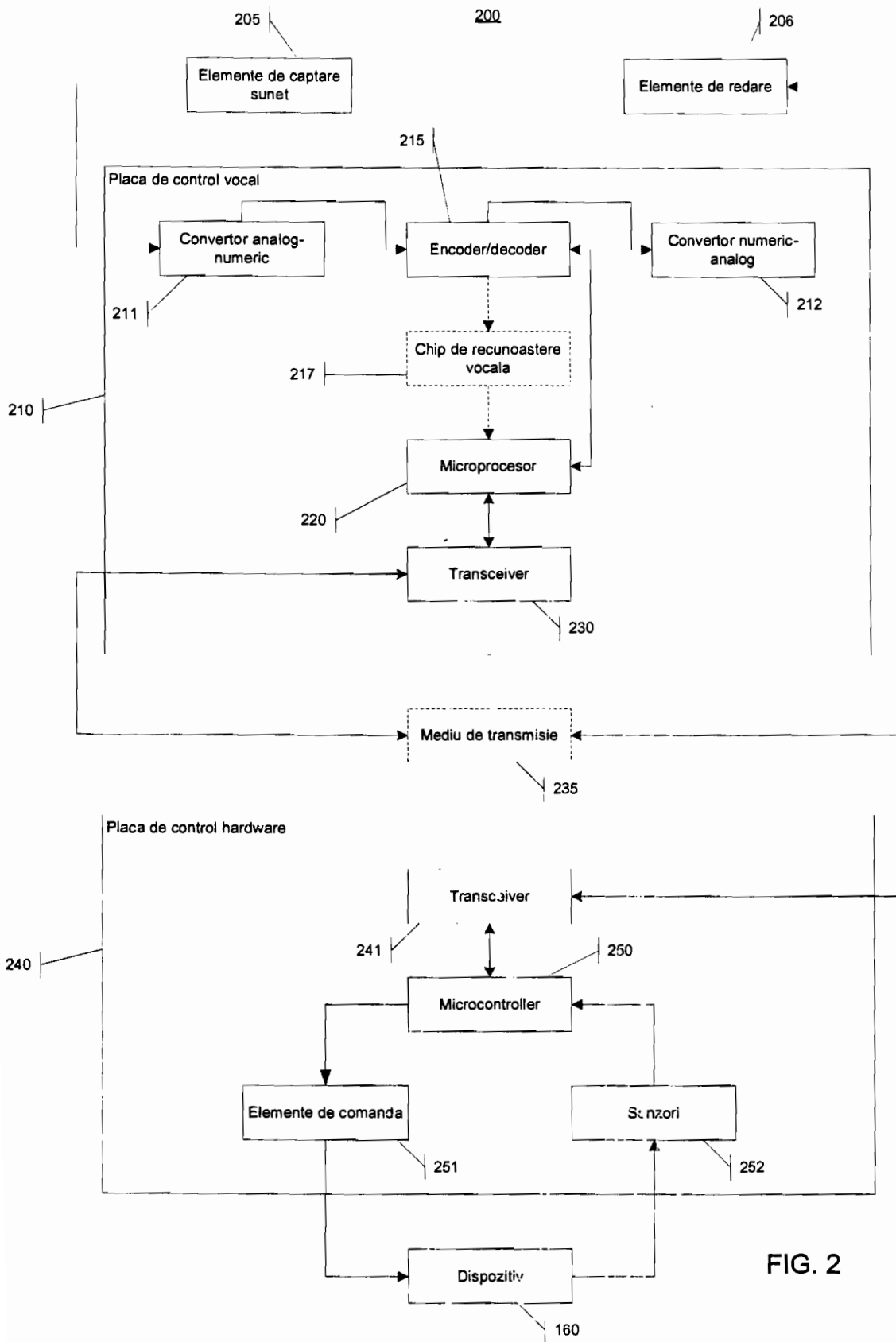


FIG. 2

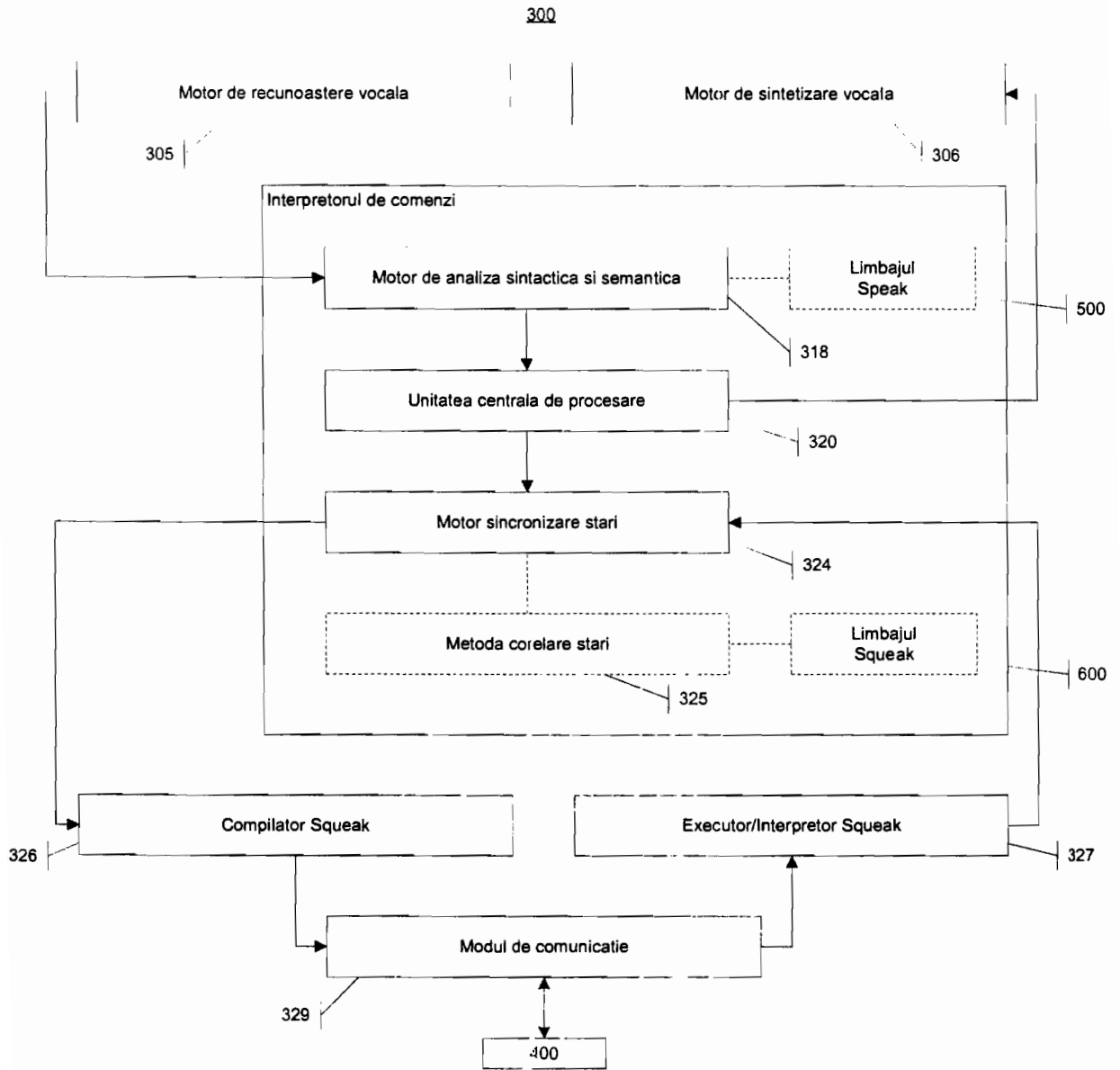
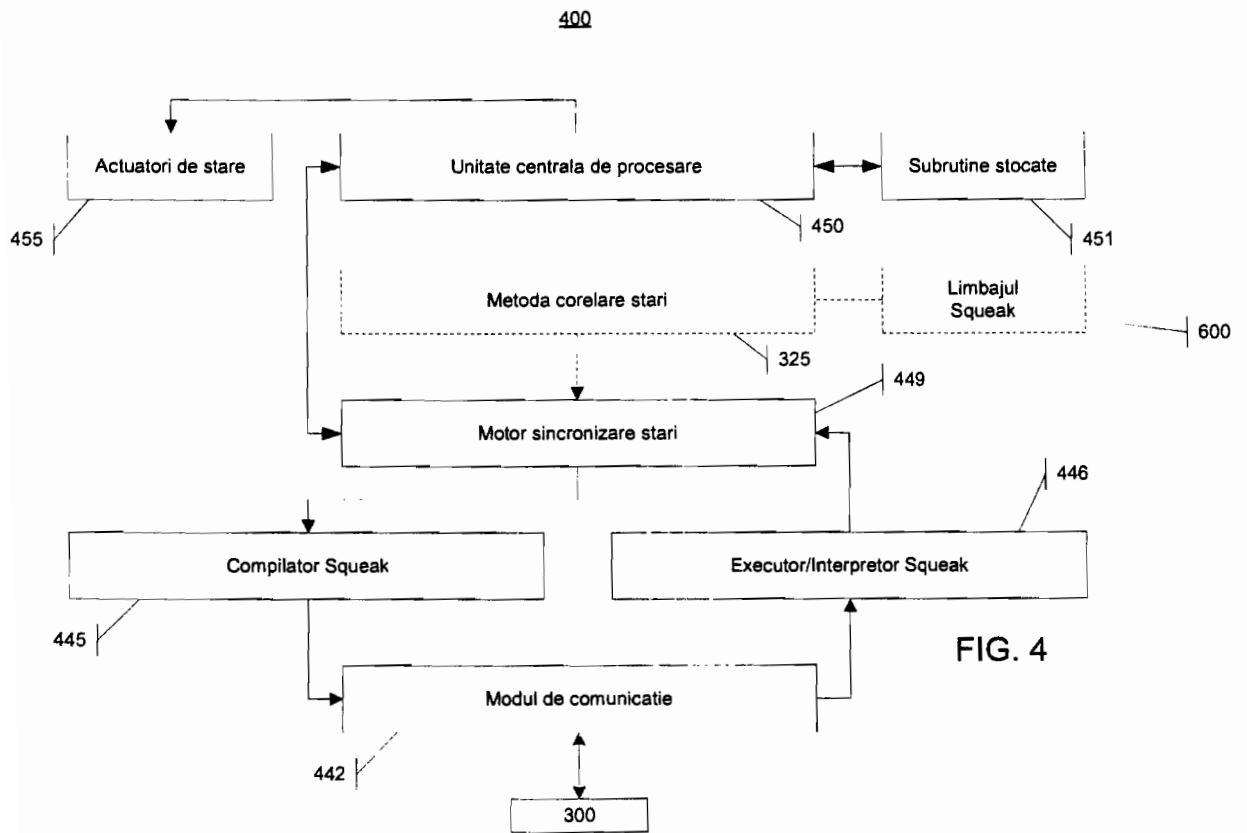


FIG. 3



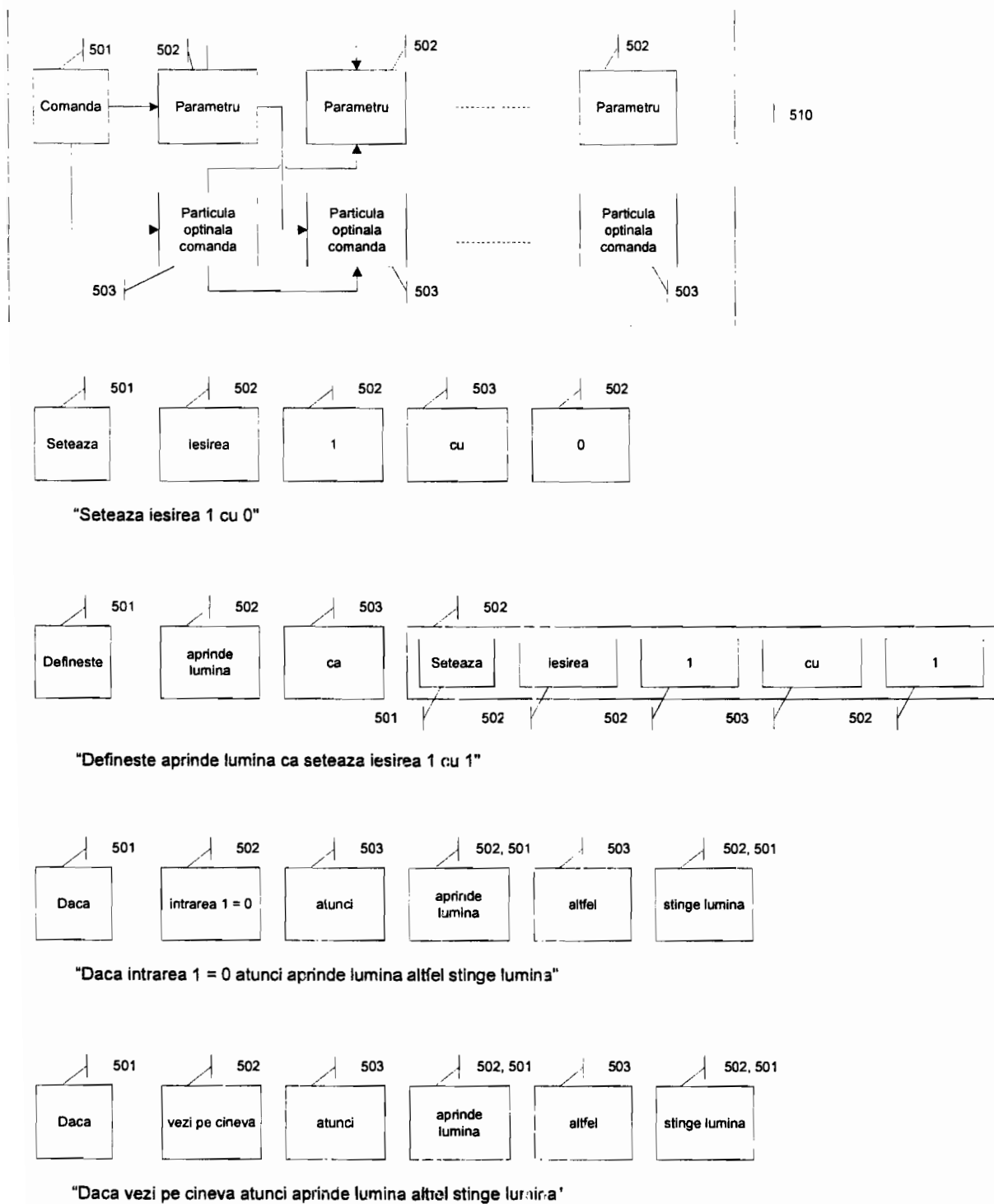
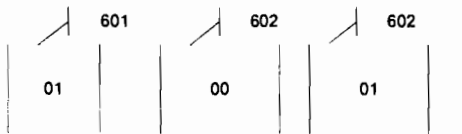
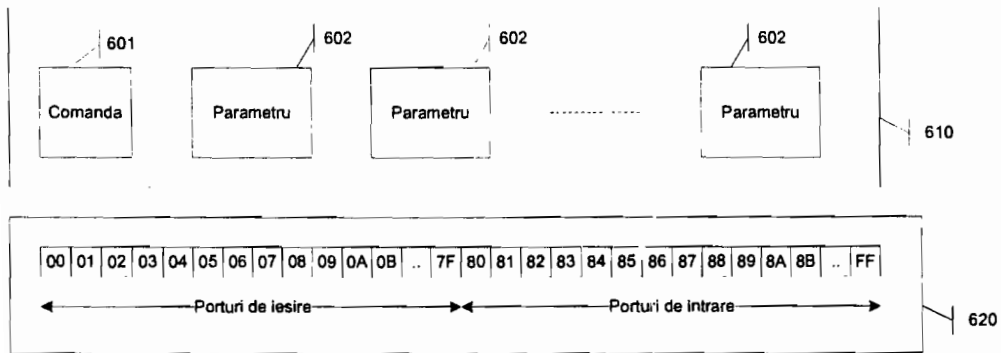
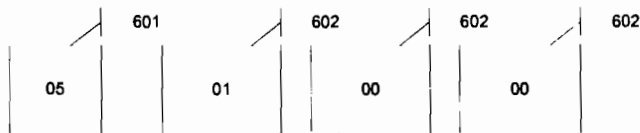


FIG. 5





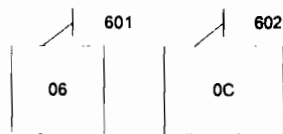
01 00 01 = "Seteaza (01) portul 0 (00) cu 1 (01)"



05 01 00 00 = "Adauga subrutina (05) seteaza (01) portul 0 (00) cu 0 (00)"



04 8A 00 FF 0A 0B = "Daca (04) portul (8A) = (00) 255 (FF) atunci executa subrutina (0A), altfel subrutina (0B)"



06 0C = "Executa subrutina stocata la 0C"

0D	06 0C
0C	04 8A 00 FF 0A 0B
0B	01 01 01
0A	01 01 00
..	..
01	01 00 01
00	01 00 00

630

FIG. 6