



F100093676B



SUOMI-FINLAND  
(FI)

Patentti- ja rekisterihallitus  
Patent- och registerstyrelsen

(B) (11) KUULUTUSJULKAISU  
UTLAGGNINGSSKRIFT

93676

(15) Patentti myönnetty  
Patent meddelat 10 05 1995

(51) Kv.1k.6 - Int.cl.6

G 08B 5/22, H 03M 13/22

(21) Patenttihakemus - Patentansökning	931648
(22) Hakemispäivä - Ansökningsdag	13.04.93
(24) Alkupäivä - Löpdag	13.04.93
(41) Tullut julkiseksi - Blivit offentlig	14.10.94
(44) Nähtävaksipanon ja kuul.julkaisun pvm. - Ansökan utlagd och utl.skriften publicerad	31.01.95

(71) Hakija - Sökande

1. Nokia Mobile Phones Ltd., PL 86, 24101 Salo, (FI)

(72) Keksijä - Uppfinnare

1. Kuusinen, Tero, Menninkäisenkatu 10 C 20, 20540 Turku, (FI)

(74) Asiamies - Ombud: Berggren Oy Ab

(54) Keksinnön nimitys - Uppfinningens benämning

KytKentä deinterliivauksen toteuttamiseksi hakulaitejärjestelmässä  
Koppling för förverkligande av deinterleaving i ett sökaranläggningssystem

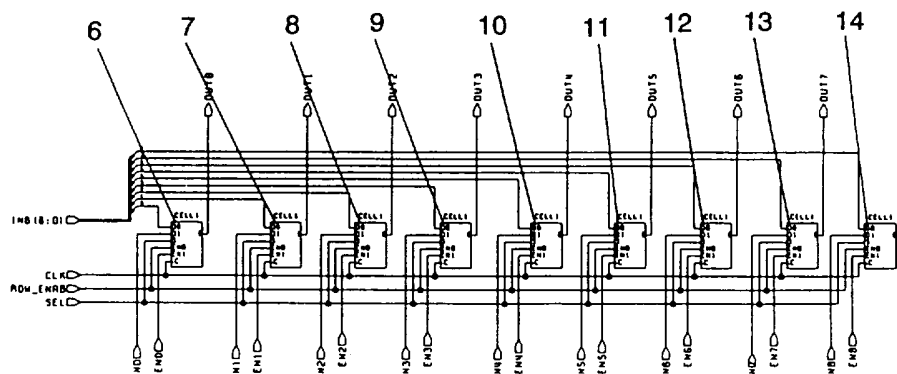
(56) Viitejulkaisut - Anförda publikationer

US A 5063533 (G 06F 12/00)

(57) Tiivistelmä - Sammandrag

Keksinnön kohteena on hakulaitejärjestelmässä käytettävä kytkentä sekä menetelmä deinterliivauksen toteuttamiseksi. RAM-muistin muistielementit on järjestetty matriisimuotoon siten, että kussakin rivissä olevien elementtien määrä vastaa koodisanan bittimäärää ja rivien lukumäärä vastaa koodisanojen määrää. Ennen muistia suoritetaan sarja rinnan-muunnos ja kussakin muistielementissä on multipleksattu sisääntulo siten, että muistiin voidaan kirjoittaa riveittäin vaakasuoraan ja muistista lukea pystysuoraan. Jos esimerkiksi muistissa on 30 x 9 elementtiä, tarvitaan keksinnön mukaisesti kirjoittamiseen 30 osoitetta ja lukemiseen 9 osoitetta eli yhteensä 39 osoitetta aiemman 270 osoitteen asemesta.

Uppfinningsföremålet är en koppling samt ett förfarande avsett att användas för förverkligande av deinterleaving i ett sökaranläggningssystem. Minneselementen i ett RAM-minne är anordnade i matrisform så att antalet element i var och en rad motsvarar kodordets bitmängd och antalet rader motsvarar mängden kodord. Före minnet utförs en serie-parallell-transformation och vart och ett minneselement uppvisar en multiplexerad ingång så, att minnet kan inskrivas radvis vågrätt samt avläsas lodrätt. Ifall minnet exempelvis har 30 x 9 element, krävs enligt uppfinningen för inskrivande 30 adresser och för avläsning 9 adresser dvs. sammanlagt 39 adresser i stället för 270 såsom tidigare.



KytKentä deinterliivauksen toteuttamiseksi hakulaitejärjestelmässä -

Koppling för förverkligande av deinterleaving i ett sökaranläggningssystem

Keksinnön kohteena on kytkentä sekä menetelmä deinterliivauksen toteuttamiseksi hakulaitejärjestelmässä.

Deinterliivauksella tarkoitetaan sitä, että vastaanotto-päässä puretaan lähettimen tekemä interliivaus eli bittien järjestäminen taulukkomuotoon ja niiden lähettäminen uudessa järjestyksessä eteenpäin. Interliivauksella saavutettava etu on siinä, että siirtokanavalla mahdolliset purskevirheet voidaan levittää useaan eri koodisanaan ja näin virheenkorjauksen kykyä purskevirheiden korjaukseen lisätään.

Esimerkin mukaisessa sovelluksessa standardin mukainen interliivaus on 270 bittiä syvä. Jokaisessa koodisanassa on 30 bittiä ja 9 koodisanan bitit järjestellään uudelleen eli interliivataan lähetystä varten siten, että koodisanan 0 eniten merkitsevä bitti MSB (MSB, most significant bit) lähetään ensin, sitten koodisanan 1 MSB jatkaen aina koodisanan 7 vähiten merkitsevään bittiin LSB (LSB, least significant bit) ja viimein lähetetään koodisanan 8 LSB. Seuraavassa taulukossa on esitetty standardin mukaisen interliivauksen toteutusperiaate.

8 <sub>0</sub>	7 <sub>0</sub>	6 <sub>0</sub>	5 <sub>0</sub>	4 <sub>0</sub>	3 <sub>0</sub>	2 <sub>0</sub>	1 <sub>0</sub>	0 <sub>0</sub>
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
8 <sub>28</sub>	7 <sub>28</sub>	6 <sub>28</sub>	5 <sub>28</sub>	4 <sub>28</sub>	3 <sub>28</sub>	2 <sub>28</sub>	1 <sub>28</sub>	0 <sub>28</sub>
8 <sub>29</sub>	7 <sub>29</sub>	6 <sub>29</sub>	5 <sub>29</sub>	4 <sub>29</sub>	3 <sub>29</sub>	2 <sub>29</sub>	1 <sub>29</sub>	0 <sub>29</sub>

Bittitaulukko täytetään yhdellä koodisanalla kerrallaan pystyriveittäin ja puretaan vaakariveittäin, jolloin interliivaus saadaan toteutettua. Taulukossa esim.  $4_{28}$  tarkoittaa 4. koodisanan 28. bittiä.

Interliivaus puretaan lähes poikkeuksetta RAM-muistin (RAM, random access memory) avulla ja sen pitää myös olla useimmiten sarjaväylällä varustettu, koska bitit kirjoitetaan sisään toiseen suuntaan ja luetaan ulos toiseen suuntaan RAM-muistin riveihin nähden.

Jos käytetään esimerkiksi 8 bitin sanoja RAM-muistin täyttämiseen, niin uloslukeminen on vaikea toteuttaa. Kun joudutaan käyttämään sarjamuotoista RAM-muistia, muistin osoitteita tarvitaan yhtä monta kuin on muistielementtejä ja tämä tässä sovelluksessa on  $30 * 9 = 270$  kappaletta. Tunnetun tekniikan mukainen ratkaisu onkin liian monimutkainen ja vie aivan liikaa RAM-muistin osoitetilaa.

Esillä olevan keksinnön tarkoituksena on aikaansaada sellainen ratkaisu, jonka avulla edellä esitetyt puutteet ja ongelmat voitaisiin ratkaista. Tämän saavuttamiseksi on keksinnön mukaiselle kytkennälle tunnusomaista se, mikä on esitetty patenttivaatimuksessa 1. Keksinnön mukaisen menetelmän tunnusmerkit selviävät vastaavasti vaatimuksesta 5.

..

Keksintöä selostetaan seuraavassa yksityiskohtaisesti viitaten oheisiin kuviin, joista:

- kuva 1 esittää keksinnön mukaisessa kytkennässä käytettävän muistisolun rakennetta,
- kuva 2 esittää keksinnön mukaisessa kytkennässä käytettävän muistisolusta koostuvan muistielementin rakennetta,
- kuva 3 esittää keksinnön mukaisen ratkaisun muistiin kirjoittamisen toteutusta,
- kuva 4 esittää keksinnön mukaisen ratkaisun muistista lukemisen toteutusta.

Keksinnön mukaista ratkaisua kuvataan seuraavassa viitaten kuviin 1 - 4, jotka esittävät keksinnön mukaisen ratkaisun toteutusta.

Kuvassa 1 on esitetty keksinnön mukaisessa kytkennässä käytettävän muistielementin rakenne. Muistielementtinä on D-kiikku 1, jossa on multipleksattu sisääntulo. Sisääntulossa on JA-elimet 2 ja 3 sekä TAI-elimet 4 ja 5.

Interliivauksen purkuun käytetään yhtä tai kahta RAM-muistia. Tämä multipleksaus mahdollistaa sen, että kirjoittaminen ja lukeminen voidaan suorittaa eri sisääntulojen kautta ja siis eri suuntiin, kirjoittaminen vaakasuoraan ja lukeminen pystysuoraan.

Kuvassa 2 on esitetty keksinnön mukaisessa kytkennässä käytettävän muistielementeistä koostuvan muistin rakenne. Muistin muistielementit 6 - 14 on järjestetty siten, että niitä on yhdessä rivissä 9 kpl. Näitä rivejä laitetaan päällekkäin 30 kpl, jolloin saadaan 30 riviä x 9 saraketta käsittävä RAM-muisti. Tähän muistiin kirjoitetaan rivi kerrallaan eli 9 bittiä kerrallaan, joten muistin ulkopuolella on tehtävä 9 bitin sarja rinnan-muunnos, joka on helppo toteuttaa yksinkertaisen siirtorekisterin avulla. Muistiin kirjoittaminen aloitetaan alimmasta rivistä, jonka osoite on 01 (HEX).

..

Kuvassa 3 on esitetty keksinnön mukaisen ratkaisun muistiin kirjoittamisen toteutus. Muistiin kirjoittaminen toteutetaan keksinnön mukaisessa ratkaisussa seuraavasti. Ensin valitaan multipleksauksen valintalinja SEL ei-aktiiviseksi "0". Tämän jälkeen valitaan osoitteella rivi, johon kirjoitetaan. Tämä osoite asettaa yhden rivin ROW\_ENAB-linjan aktiiviseksi, mikä sallii kellon menon kiikulle. Kun sarja rinnan-muunnoksessa on vastaanotettu ensimmäiset 9 bittiä, niin nämä 9 bittiä kellotetaan osoitteen mukaiseen riviin. Muistielementit 15 - 24 on

..

sijoitettu siten, että jokaisen rivin i:s muistielementti 15 - 24 eli muistielementin 15 - 24 kirjoituksen sisääntulo on johdotettu sarja rinnan-muunnoksen i:nteen bittiin. Tämän jälkeen vaihdetaan osoitetta, jolloin tätä uutta osoitetta vastaavan rivin linja ROW\_ENAB tulee aktiiviseksi ja edellisen rivin linja ROW\_ENAB muuttuu ei-aktiiviseksi. Taas odotetaan 9 bitin saapuminen, jonka jälkeen tämä yhdeksän bitin sana kellotetaan sisään muistielementteihin. Tätä proseduuria jatketaan kunnes kaikki 30 riviä on kirjoitettu täyteen.

Kuvassa 4 on esitetty keksinnön mukaisen ratkaisun muistista lukemisen toteutus. Kytkentä käsittää RAM-piirit 25 -27, JA-elimet 28 - 35, 37, TAI-elimet 36 ja 39, invertterin 38 ja kiikun 40. Ensin valitaan multipleksauksen valintalinja SEL aktiiviseksi "1". Tämän jälkeen valitaan osoitteella se sarake (30 bittiä), joka aiotaan lukea muistista. Tämä osoite asettaa yhden sarakkeen EN0 .. EN8 signaalin aktiiviseksi, mikä sallii kellon menon tämän sarakkeen kiikulle.

Muistielementit on kytketty pystysuoraan sarjaan siten, että ulos lukemisessa 30 kellojaksolla saadaan koko 30 bitin sana ulos RAMista MSB edellä. Kun osoitteet on valittu, kellotetaan CLK-signaalin avulla koko 30 bitin sarake ulos. Kun koko sana on luettu, valitaan uusi osoite, joka aktivoi uuden EN0 .. EN7 linjan ja tätä osoitetta vastaava sana voidaan lukea ulos.

Deinterliivaukseen voidaan käyttää joko yhtä tai kahta edellä kuvattua RAMia. Jos käytetään kahta RAMia, niin kun toista täytetään eli siihen kirjoitetaan, niin samanaikaisesti toista puretaan eli luetaan. Jos käytetään vain yhtä RAMia, niin muistia täytetään normaalisti, kunnes viimeinenkin rivi on kirjoitettu. Tämän jälkeen muisti puretaan sinä aikana, kun 9 seuraavaa bittiä vastaanotetaan. Tämä

tosin asettaa deinterliivauksen jälkeisille lohkoille varsin kovat nopeusvaatimukset.

Tällä tavalla johdotetulle RAMille osoitteiden generointi on varsin helppoa, koska rivit ja sarakkeet täytetään/puretaan järjestyksessä. Ennen kuin täyttö/purku alkaa, nolataan osoitelaskurit. Haluttu seuraava osoite saadaan aikaan, kun osoitelaskureita kellotetaan yksi askel eteenpäin. Näin muistin hallinta on todella helppoa, ja se on hoidettavissa kahden osoitelaskurin avulla.

Tässä selityksessä on käsitelty ERMES-järjestelmään sopivaa deinterliivauksen syvyyttä (30\*9 bittiä), mutta samaa metodologia voidaan käyttää minkä tahansa deinterliivauksen syvyyden toteuttamiseen. Keksinnön mukainen ratkaisu soveltuu erittäin hyvin käytettäväksi ERMES-hakulaitejärjestelmässä.

Keksinnön avulla deinterliivaus voidaan purkaa hyvin pienellä osoitteenkoodauksella. Perusmuistielementeissä on multipleksattu sisääntulo. Kun nämä sisääntulot johdetaan keksinnön mukaisella tavalla, voidaan deinterliivaus toteuttaa yksinkertaisesti.

Keksinnön mukaisessa ratkaisussa tarvitaan ainoastaan 30 osoitetta kirjoittamiseen sekä 9 osoitetta uloslukemiseen eli yhteensä  $30 + 9 = 39$  osoitetta. Tämä mahdollistaa sen, että käytetään vain yhtä RAMia deinterliivauksen tekemiseen, koska 30 bitin uloslukemiseen tarvitaan vain yksi osoite perinteisen ratkaisun 30 osoitteen sijasta.

Keksinnöllä on merkitystä lähinnä silloin, kun interliivauksessa tai deinterliivauksessa tarvitaan niin suurta nopeutta, että laitteessa oleva prosessori ei ehdi suorittaa operaatiota ohjelmallisesti. Keksinnön mukainen ratkaisu nopeuttaa operaatiota huomattavasti.

Patenttivaatimukset

1. Kytkentä deinterliivauksen toteuttamiseksi hakulaitejärjestelmässä, joka kytkentä käsittää ainakin yhden RAM-muistin, jonka muistielementit (6-14; 15-24;) on järjestetty matriisimuotoon siten, että yhdessä rivissä on koodisanan bittimäärää vastaava määrä muistielementtejä ja päällekkäisten rivien lukumäärä vastaa käytettyjen koodisanojen määrää, **tunnettu** siitä, että kussakin muistielementissä (6-14; 15-24) on multipleksattu sisääntulo, joka mahdollistaa muistiin kirjoittamisen vaarasuoraan rivi kerrallaan sekä muistista lukemisen pystysuoraan eli sarakke kerrallaan.
2. Patenttivaatimuksen 1 mukainen kytkentä, **tunnettu** siitä, että ennen RAM-muistia on sarja rinnan-muunnoksen suorittava lohko, esim. siirtorekisteri.
3. Patenttivaatimuksen 1 mukainen kytkentä, **tunnettu** siitä, että muistielementti koostuu D-kiikusta (1), jossa on multipleksattu sisääntulo, JA-elimistä (2, 3) sekä TAI-elimistä (4, 5).
4. Patenttivaatimuksen 1 mukainen kytkentä, **tunnettu** siitä, että muisti koostuu 30 rivistä muistielementtejä (6-14; 15-24), joista kussakin rivissä on 9 muistielementtiä.
5. Menetelmä deinterliivauksen toteuttamiseksi hakulaitejärjestelmässä, jossa käytetään RAM-muistia, jonka muistielementit (6-14; 15-24) on järjestetty matriisimuotoon siten, että yhdessä rivissä on koodisanan bittimäärää vastaava määrä muistielementtejä ja päällekkäisten rivien määrä vastaa käytettyjen koodisanojen määrää, **tunnettu** siitä, että ennen muistia suoritetaan sarja rinnan-muunnos, että kussakin muistielementissä käytetään multipleksattua sisääntuloa, jolloin muistiin kirjoittaminen suori-

tetaan riveittäin vaakasuoraan ja muistista lukeminen sarakkeittain pystysuoraan.

6. Patenttivaatimuksen 5 mukainen menetelmä, **tunnettu** siitä, että muistiin kirjoittamisen toteutuksessa
- valitaan multipleksauksen valintalinja (SEL) ei-aktiiviseksi "0",
  - valitaan osoitteella rivi, johon kirjoitetaan,
  - osoite asettaa yhden rivin linjan ROW\_ENAB aktiiviseksi, mikä sallii kellon menon kiikulle,
  - kellotetaan ensimmäiset sarja rinnan-muunnoksessa vastaanotetut koodisanan bitit osoitteen mukaiseen riviin,
  - vaihdetaan osoitetta, jolloin tätä uutta osoitetta vastaavan rivin linja ROW\_ENAB tulee aktiiviseksi ja edellisen rivin linja ROW\_ENAB muuttuu ei-aktiiviseksi, ja
  - jatketaan kunnes kaikki rivit on kirjoitettu täyteen.

7. Patenttivaatimuksen 5 mukainen menetelmä, **tunnettu** siitä, että muistista lukemisen toteutuksessa
- valitaan multipleksauksen valintalinja (SEL) aktiiviseksi "1",
  - valitaan osoitteella se sarake, joka aiotaan lukea muistista,
  - osoite asettaa yhden sarakkeen (EN0 .. EN8) signaalin aktiiviseksi, mikä sallii kellon menon tämän sarakkeen kiikulle,
  - kellotetaan kellosignaalin (CLK) avulla koko sarake ulos,
  - valitaan edelleen uusi osoite, joka aktivoi uuden sarakkeen (EN0 .. EN7) linjan, ja luetaan tätä osoitetta vastaava sana.

8. Patenttivaatimuksen 5 mukainen menetelmä, **tunnettu** siitä, että käytetään kahta RAM-muistia siten, että kun toiseen kirjoitetaan niin samanaikaisesti toisesta luetaan.

9. Patenttivaatimuksen 5 mukainen menetelmä, **tunnettu** siitä, että käytetään yhtä RAM-muistia siten, että muistia täytetään normaalisti, kunnes viimeinenkin rivi on kirjoitettu, minkä jälkeen muisti puretaan sinä aikana, kun seuraavaa riviä vastaava bittimäärä vastaanotetaan.

10. Patenttivaatimuksen 8 tai 9 mukainen menetelmä, **tunnettu** siitä, että osoitteiden generointi toteutetaan siten, että ennen kuin täyttö/purku alkaa, nollataan osoitelaskurit, minkä jälkeen haluttu seuraava osoite saadaan aikaan, kun osoitelaskureita kellotetaan yksi askel eteenpäin.

#### Patentkrav

1. Koppling för förverkligande av deinterleaving i ett sökaranläggningssystem, vilken koppling omfattar åtminstone ett RAM-minne, vars minneselement (6-14; 15-24) anordnats i matrisform sålunda, att en rad omfattar en minneselementmängd motsvarande kodordets bitmängd och antalet på varandra belägna rader motsvarar mängden av utnyttjade kodord, **kännetecknad** av att vart och ett minneselement (6-14; 15-24) uppvisar en multiplexerad ingång, vilken möjliggör en inskrivning i minnet av en vågrät rad åt gången samt läsning från minnet vertikalt dvs en kolumn åt gången.

2. Koppling enligt patentkrav 1, **kännetecknad** av att före RAM-minnet befinner sig ett block, exempelvis ett shiftregister som utför en serie-parallelltransformation.

3. Koppling enligt patentkrav 1, **kännetecknad** av att minneselementet utgörs av en D-vippa (1), vilken har en multiplexerad ingång, ett OCH-element (2, 3) samt ett ELLER-element (4, 5).

4. Koppling enligt patentkrav 1, **kännetecknad** av att minnet består av 30 rader minneselement (6-14; 15-24), där varje rad uppvisar 9 minneselement.

5. Förfarande för förverkligande av deinterleaving i ett sökaranläggningssystem, där man använder ett RAM-minne, vars minneselement (6-14; 15-24) anordnats i matrisform sålunda, att en rad uppvisar ett antal minneselement som motsvarar kodordets bitmängd och antalet på varandra belägna rader motsvarar antalet utnyttjade kodord, **kännetecknat** av att före minnet utförs en serie-parallelltransformation, att i vart och ett minneselement utnyttjas en multiplexerad ingång, varvid inskrivning i minnet utförs radvis vågrät och läsning från minnet kolumnvis vertikalt.

6. Förfarande enligt patentkrav 5, **kännetecknat** av att i förverkligandet av inskrivningen i minnet

- multiplexeringens väljarlinje (SEL) väljs icke-aktiv "0",
- medelst en adress utväljs den rad på vilken inskrivningen utförs,
- adressen inställer linjen för en rad till ROW\_ENAB aktiv, vilket tillåter klocksignalen till vippan,
- de första i serie-parallelltransformationen mottagna bitarna hos kodordet inklockas på rätt rad enligt adressen,
- adressen växlas, varvid den mot den nya adressen svarande radlinjen ROW\_ENAB blir aktiv och den föregående radlinjen ROW\_ENAB blir icke-aktiv, och
- fortskrides sålunda tills samtliga rader fullskrivits.

7. Förfarande enligt patentkrav 5, **kännetecknat** av att läsning från minnet förverkligas sålunda, att

- multiplexeringens vallinje (SEL) väljs aktiv "1",
- medelst en adress väljs den kolumn, vilken avses att utläses från minnet,
- adressen inställer signalen för en kolumn (EN0 .. EN8) aktiv, vilket tillåter klocksignalen till vippan för denna kolumn,
- hela kolumnen utklockas medelst klocksignalen (CLK),

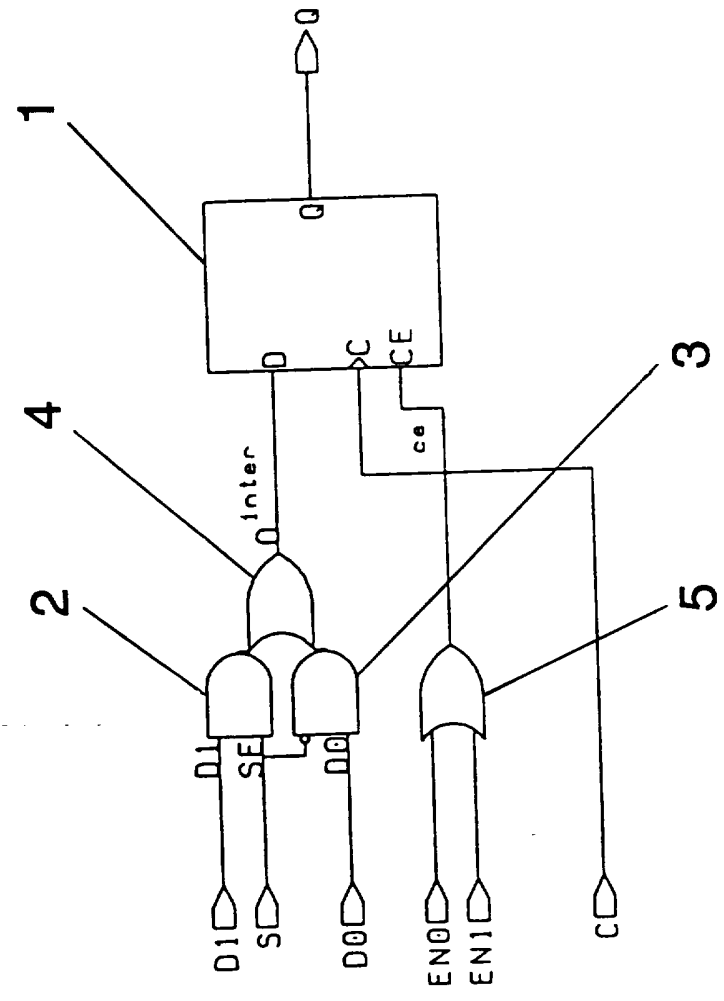
:

- vidare väljs en ny adress, vilken aktiverar linjen för en ny kolumn (EN0 .. EN7), samt utläses det ord som motsvarar denna adress.

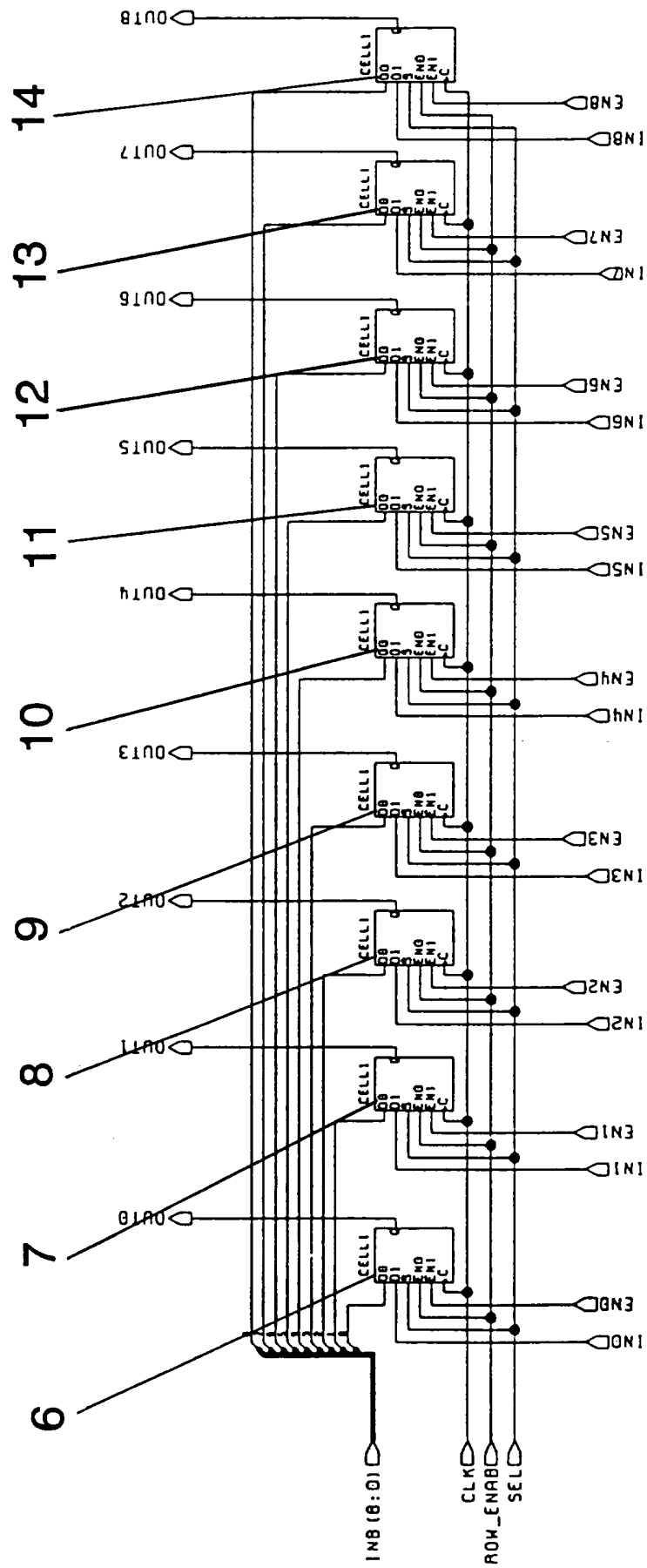
8. Förfarande enligt patentkrav 5, **kännetecknat** av att man utnyttjar tvenne RAM-minnen sålunda, att under inkrivningen i det ena samtidigt utförs avläsning från det andra.

9. Förfarande enligt patentkrav 5, **kännetecknat** av att man utnyttjar ett RAM-minne sålunda, att minnet fylls normalt, tills även den sista raden inskrivits, varefter minnet avläses under den tid då man mottar den mot den följande raden svarande bitmängden.

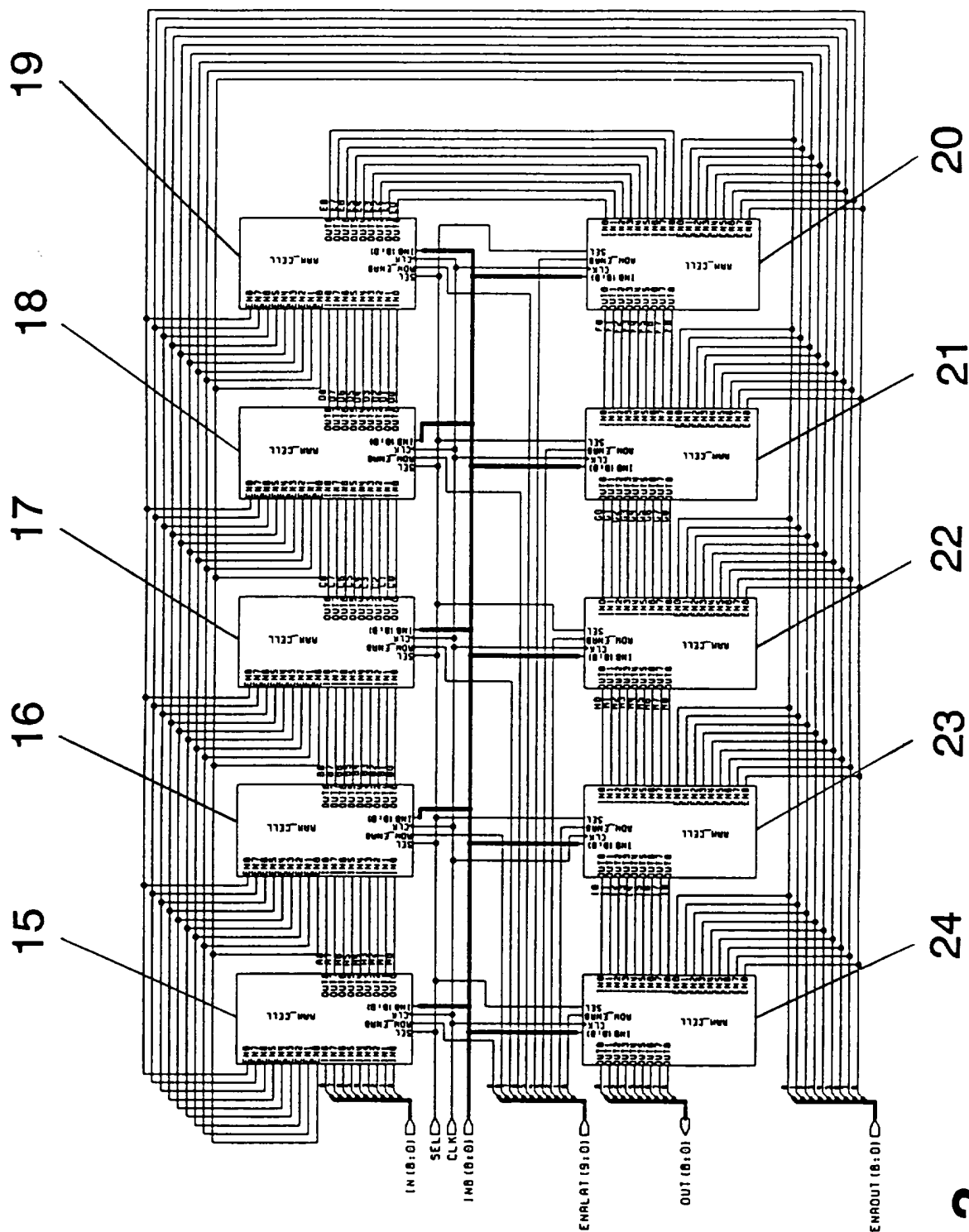
10. Förfarande enligt patentkrav 8 eller 9, **kännetecknat** av att genereringen av adresserna utförs sålunda, att innan påfyllningen/utläsningen utförs, adressräknarna nollas, varefter den önskade följande adressen erhålls då adressräknarna klockas ett steg framåt.



Kuva 1



Kuva 2



**Kuva 3**

