

(19)



SUOMI - FINLAND

(FI)

PATENTTI- JA REKISTERIHALLITUS

PATENT- OCH REGISTERSTYRELSEN

FINNISH PATENT AND REGISTRATION OFFICE

(10) **FI 752448 A7**

(12) **JULKISEKSI TULLUT PATENTTIHAKEMUS
PATENTANSÖKAN SOM BLIVIT OFFENTLIG
PATENT APPLICATION MADE AVAILABLE TO THE
PUBLIC**

(21) Patentihakemus - Patentansökan - Patent application **752448**

(51) Kansainvälinen patenttiluokitus - Internationell patentklassifikation -
International patent classification
B66B

(22) Tekemispäivä - Ingivningsdag - Filing date **29.08.1975**

(23) Saapumispäivä - Ankomstdag - Reception date **29.08.1975**

(41) Tullut julkiseksi - Blivit offentlig - Available to the public **05.03.1976**

(43) Julkaisupäivä - Publiceringsdag - Publication date **12.06.2019**

(32) (33) (31) Etuoikeus - Prioritet - Priority

04.09.1974 US 503201

(71) Hakija - Sökande - Applicant

1 •Westinghouse Electric Corporation, Westinghouse Building, Gateway Center, Pittsburgh, PA 15222,
AMERIKAN YHDYSVALLAT, (US)

(72) Keksijä - Uppfinnare - Inventor

1 •Winkler, Charles L., USA, AMERIKAN YHDYSVALLAT, (US)

(74) Asiamies - Ombud - Agent

Berggren Oy Ab, Antinkatu 3 C, 00100 Helsinki

(54) Keksinnön nimitys - Uppfinningens benämning - Title of the invention

Hissijärjestelmä

Hissystem

WESTINGHOUSE ELECTRIC CORPORATION, Westinghouse Building, Gateway Center, Pittsburgh, Pennsylvania 15222, USA

C-1401

Hissijärjestelmä - Hissystem

Esillä oleva keksintö kohdistuu patenttivaatimuksen 1 johdannon mukaiseen hissijärjestelmään ja erityisesti uuteen ja parannettuun järjestelmän valvontalaitteeseen ja strategioihin useiden hissikorien ohjaamiseksi vastaamaan hissipalvelua vaativiin kutsuihin.

Aikaisemmat yritykset hissiteollisuudessa jakaa hissipalvelukutsut joukolle hissikoreja muodostivat palveluvyöhykkeitä, jotka ulottuivat korista seuraavaan varattuun koriin, joka toimi ensin mainitun edellä. Painonapit sijoitettiin jokaiselle kerrosta-alle jokaista koria varten, jolloin korin palveluvyöhyke valaisi kerrospainonapit, jotka sijaitsivat palveluvyöhykkeen kerroksissa osoittaakseen mahdollisille matkustajille, mitä nappia painaa saadakseen nopeimman palvelun.

Eräs aikaisempi yritys kvoottityyppisessä ohjauksessa salli hissikorin pysähtyä vain niillä kerrostaokutsuilla, jotka oli rekisteröity ennen kuin se jätti pääteaseman. Myöhemmin kvoottiohjaukset tunkeutuivat palveluvyöhykkeisiin. Kvoottimekanismi voitiin asettaa estämään koria hyväksymästä useampia vastaamattomia kutsuja palveluvyöhykkeestä minään ajankohtana kuin mitä sen kvootti oli taikka se voi-

tiin asettaa rajoittamaan korin edestakaisella matkalla tekemien pysähdysten määrää estämällä koria hyväksymästä muita kutsuja sen jälkeen kun se oli hyväksynyt osamääränsä eli kvoottinsa.

Alaspäinhuippuvyöhykkeitys jakaa rakennuksen vyöhykkeisiin alaspäinhuippuliikennetilassa estääkseen rakennuksen alempien kerrosten kohtuuttoman palvelun. On tunnettua kääntää tiettyt ylöspäin kulkevat korit kahdesta vyöhykkeestä alemman korkeimman kutsun kohdalla, kun alemman vyöhykkeen liikenteen taso saavuttaa ennaltamäärätyn suuruuden. Toisena keinona oli kääntää korit vyöhykkeen ylimmän alaspäin-kutsun kohdalla, kun liikenteen taso saavutti ennaltamäärätyn suuruuden eikä vyöhykkeessä ollut koreja.

Tunnettuja ovat myös ajastetut toimitusjärjestelmät, jotka automaattisesti kytkevät useiden erilaisten liikennemuotojen välillä perustuen havaittuihin liikenneolosuhteisiin, mukaanluettuna vyöhykkeistys ylöspäinhuippu- ja alaspäinhuippuolosuhteissa. Erona edelliselle tunnetaan vyöhykeistetty kysyntäjärjestelmä, jossa hissikorit toimivat riippuvaisesti kysynnästä ajastetun toimitusjärjestyksen sijasta.

US-patentin 3 614 997 mukainen järjestelmä määrittää liikennetason kunkin korin palveluvyöhykkeessä ja liikennetasot summataan ja jaetaan korien luvulla keskiarvon saamiseksi. Tätä keskiarvoista liikennetasoa vyöhykkeissä käytetään ohjaamaan korien välimatkajärjestystä ja työkuormaa aiheuttamalla se, että kori, joka kulkee sellaisen korin edellä, jolla on pienempi kuin keskimääräinen työkuorma, ohittaa kerrostaokutsuja ja myös määräämällä, milloin kori on käytettävissä erityiseen suoraan palveluun prioriteettikerrostaokutsuille.

US-patentin 3 729 066 mukaisessa järjestelmässä kehitetään palveluvyöhykkeitä hissikorien välillä ja vaihdetaan palveluvyöhykkeitä palvelun tasapainoittamiseksi synnyttämällä korien imaginääriasemia ja tarkastelemalla palveluvyöhykkeitä näistä imaginäärisistä asemista.

Aikaisemmissa järjestelmäohjauslaitteissa käytettiin releitä, kun taas myöhemmissä ohjauslaitteissa käytetään puolijohdeteknologiaa. Vaihto puolijohde-järjestelmäohjauslaitteisiin jatkoi suurten, monimutkaisten releitä käyttävien ohjauslaitteiden filosofiaa käsittelemällä suurten määrän tietoja, jotka liittyivät useihin hissikoreihin suuressa rakennuksessa, mikä vaatii joko monimutkaista kovatavaralogiikkajär-

jestelmää taikka tehokasta, yleistarkoitusminutietokonetta. Vaikka nämä keinot ovat hyvin sopivia suuria, hyvin nopeiden hissien muodostamia joukkoja varten, ne ovat liian kalliita pienempiä, keskinopeuksisten hissien muodostamia joukkoja varten.

Mikroprosessorit, esimerkiksi Intel'in MCS-4 ja MCS-8, Rockwell'in PPS, Signetic'in PIP, National'in GPC/P ja AMI'n 7300 tarjoavat houkuttelevan kustannuspaketin sekä joustavuutta johtuen LSI-virtapiiristöstä ja ohjelmoitavuudesta. Keskusprosessoriyksikkönä (CPU) on tavallisesti yksi siru, tyypillinen ohjelmapaketti on varastoitu oheislaitteina oleviin pysyväismuisteihin (ROM). Tiedot varastoidaan poimintamuisteihin (RAM).

Vaikka mikroprosessori tarjoaa ohjelmointijoustavuuden kohtuullisin kustannuksin, se myös tuo mukanaan tiettyjä rajoituksia johtuen sen suhteellisen rajoitetusta nopeudesta ja muistikapasiteetista.

Esillä olevan keksinnön pääkohtana on saada aikaan uusi ja parannettu yleispätevä järjestelmä hissikoriryhmän ohjaamiseksi, joka järjestelmään soveltuu käytettäväksi mikroprosessorilla ilman erityistä prosessorin ohjelmointia jokaista installaatiota varten, käyttäen täysin hyödyksi mikroprosessorin kykyjä samalla kun toimitaan sen muisti- ja nopeusrajoitusten alaisena, nopean, tehokkaan hissipalvelun saavuttamiseksi kysymyksessä olevan rakennuksen kerroksille.

Tätä päämäärää silmälläpitäen keksintö käsittää yleispätevän hissijärjestelmän rakennusta varten, jossa on useita kerroksia, johon järjestelmään kuuluu joukko hissikoreja, joka on yleispätevä rakennuksille, jossa on useita kerroksia, johon järjestelmään kuuluu joukko hissikoreja, joista kukin on asennettu liikkuvaksi kerrokseen nähden, ylöspäin- ja alaspäin-kerrostaokutsujen rekisteröintivälineet hissipalvelukutsujen rekisteröimiseksi ylöspäin- ja alaspäin-suuntiin ainakin tietyistä kerroksista, korinohjausvälineet, jotka toimittavat valtuutussignaaleja jokaista hissikoria varten, sekä valvontaohjauslaite, joka on riippuvainen mainituista ylöspäin- ja alaspäin-kerrostaokutsujen rekisteröintivälineistä, ja keksinnön tunnusmerkit selviävät oheisesta patenttivaatimuksesta 1.

Erään suoritustavan mukaan valvontajärjestelmään sisältyy ensimmäiset välineet, jotka riippuvaisina aktivointisignaaleista ryhmitävät saman hissikoriyhdistelmän palvelemaan kerrokset sekä näiden palvelusuunnat ryhmäksi, toiset välineet, jotka määräävät kullekin ryhmälle ainakin yhden ennalta määrätyn rajoituksen siitä, miten ryhmään kuuluvan hissikorin on palveltava kutsua ryhmän kerroksesta, ja kolmannet välineet, jotka tehokkaasti osoittavat hissipalvelua vaativat kutsut mainitun hissikorijoukon hisseille toimittamalla signaaleja hissikorijoukon korinohjausvälineille mainittujen toisten välineiden mukaisesti. Lisäksi voi järjestelmään sisältyä keskiarvovälineet, jotka muodostavat ensimmäisen keskiarvon riippuvaisena rekisteröityjen ylöspäin- ja alaspäin-kerrostasokutsujen lukumäärästä rakennuksessa ja palveluksessa olevaa hissikoria kohden, jolloin mainittu ensimmäinen keskiarvo on rajoituksena mainituille toisille välineille määrättäessä minkä hissikorin on palveltava kerrostasoa.

Keksintö käy paremmin ilmi seuraavasta, oheisiin piirustuksiin liittyvästä keksinnön suoritus-esimerkin selityksestä.

Kuvio 1 on osittain kaaviollinen ja osittain lohkokaaavioesitys hissi-järjestelmästä, johon sisältyy järjestelmän valvontalaitteet, joissa voidaan käyttää esillä olevan keksinnön keksinnöllistä ajatusta.

Kuvio 2 on ajastuskaavio, joka kuvaa ajastussignaaleja, jotka synnytetään aikavälien yhtä täydellistä jaksoa varten.

Kuvio 3 on ajastuskaavio, joka kuvaa ajastussignaaleja, jotka liittyvät yhteen pyyhkäisyajaväliin.

Kuvio 4 on kaaviokuva järjestelmäprosessorista, johon sisältyy keskeinen prosessiyksikkö ja oheislaitteina ROMit ja RAMit, joita voidaan käyttää kuviossa 1 lohkomuodossa esitettyä järjestelmäprosessoria varten.

Kuvio 5 on kartta, joka kuvaa kuudentoista 20 bitin rekisterin muotoa, jotka on aikaansaatava kuviossa 4 esitetyillä RAMEilla.

Kuvio 6 on kaaviokuva liitäntäpiiristä (interface), jota voidaan käyttää kuviossa 1 esitetyn järjestelmäprosessorin liitännänä.

Kuvio 7 on kaavio, joka kuvaa sarjasignaalien muotoa hissikoreista järjestelmäprosessoriin sellaisena kuin ne esiintyvät kuviossa 6 esitetyn järjestelmäprosessorin liitäntäpiirin ulostulossa.

Kuviot 8A ja 8B ovat kaaviokuvia liitäntäpiireistä, joita voidaan käyttää kunkin kuviossa 1 lohkomuodossa esitetyn hissikorin liitäntäpiirinä. Kuvio 8C on kaavio, joka kuvaa sarjasignaalien muotoa järjestelmän valvontalaitteesta kuhunkin hissikoriin.

Kuvio 9 on vuokaavio, joka kuvaa ryhmävalvontastrategiaa useiden hissikorien ohjaamiseksi.

Kuviot 10-23 ovat yksityiskohtaisia vuokaavioita alaohjelmista, joita voidaan käyttää suorittamaan eri toimintoja, jotka lohkomuodossa on esitetty kuvion 9 vuokaaviossa.

Kuvio 24 on graafinen esitys, joka kuvaa aikavälien osoittamista koreille erityisessä esimerkissä.

Kuvio 25 on ajastusdiagramma, joka kuvaa estosignaaleja, jotka on kehitetty järjestelmän valvontalaitteessa kuviossa 24 esitettyä erityistä esimerkkiä varten.

Lyhyesti sanottuna esillä oleva hakemus on kohdistettu parannettuun hissijärjestelmään, jossa on joukko hissikoreja, joita ohjataan uuden ja parannetun universaalisen strategian mukaisesti, jonka puitteisiin sopivat kaikki mahdolliset rakennusmuodot, ja jossa mikä tahansa hissikori voi palvella mitä tahansa kerrosyhdistelmää. Yksilölliset korin ohjauslaitteet tuottavat täydellisen informaation kullakin hetkellä vallitsevasta rakennuksen muodosta ja täten valvontalaitteet voidaan universaalisesti soveltaa mihin tahansa järjestelmään ilman merkittävää modifiointia. Lisäksi uusi ja parannettu toimintastrategia jakaa työkuorman tasaisesti kaikkien hissikorien kesken suhteellisen yksinkertaisen ohjelman mukaisesti, joka toimii tehokkaasti mikroprosessorin nopeus- ja muistirajoitusten rajoissa.

Tarkemmin sanottuna uuteen ja parannettuun hissijärjestelmään ja järjestelmän valvontalaitteisiin sisältyy välineet peräkkäisten samanlaisten ajanjaksojen jakamiseksi yhtä suuriin pyyhkäisyajaväleihin, jolloin kullekin rakennuksen palveltavalle kerrokselle on osoitettu eri aikaväli. Osoitukset on tehty kerroksille palvelusuuntien perusteella niistä ja täten vaikka kullekin kerrokselle on osoitettu vain yksi aikaväli, jos sitä voidaan palvella sekä ylöspäin- että alaspäin-

suunnista, sillä sanotaan olevan kaksi aikaväliä, yksi ylöspäinsuuntaan ja yksi alaspäinsuuntaan, ja kumpikin aikaväli voi olla osoitettu eri korille. Täten kun tietty aikaväli on järjestelmän valvontalaitteella osoitettu korille, kori pystyy vastaamaan kerrostasokutsuun kerroksesta, joka liittyy tähän aikaväliin, siihen palvelusuuntaan tästä kerroksesta, joka liittyy tähän aikaväliin. Aikavälien osoitukset eri hissikoreille tehdään jättämällä synnyttämättä estosignaali osoitettua aikaväliä varten. Järjestelmän valvontalaitteen epäonnistuminen estosignaalien tuottamisessa ei tee hissikoreja kykenemättömiksi, koska kukin kori vastaa kerrostasokutsuihin sen oman ohjausstrategian mukaisesti ja täten jatkaa toimintaansa yksilöllisen korinohjauslaitteen alaisena ilman ryhmäohjausta, kunnes järjestelmän valvontalaitteet jälleen toimivat tuottaen estosignaaleja.

Pysyväismuistit kussakin korinohjauslaitteessa on asetettu tuottamaan järjestelmän valvontalaitteelle binäärisignaaleja, jotka osoittavat, mitä kerroksia ja palvelusuuntia näistä kukin hissikori on valtuutettu palvelemaan kerrostasokutsuilla. Järjestelmän valvontalaitte käyttää näitä signaaleja jakaakseen kerrokset ja palvelusuunnat näistä, so. aikavälit, ryhmiin siten, että kutakin ryhmää palvelee sama hissikori-yhdistelmä. Neljän hissikorin joukossa ovat mahdollisia 16 erilaista ryhmää, so. ne aikavälit, joita palvelee yksi kori, ne, joita palvelee jokin kahden tai kolmen korin yhdistelmä ja ne, joita palvelevat kaikki neljä koria ja ne, joita mikään kori ei palvele. Aikavälit, joita mikään kori ei palvele, ovat pätemätön ryhmä. Jos kaikki korit on valtuutettu kaikkiin kerroksiin ja kaikkiin palvelusuuntiin, on olemassa vain yksi ryhmä. Sen jälkeen kun ryhmät on määritetty, ne ovat häiritsemättömiä siihen saakka, kunnes kori tuottaa signaalin, joka osoittaa, että se on poistumassa palveluksesta tai tulossa palvelukseen, jolloin ryhmät määritetään uudelleen.

Järjestelmän valvontalaitteeseen sisältyy varastointivälineet niiden binäärisignaalien varastoimiseksi, joiden mukaisesti korit on valtuutettu palvelemaan kerroksia ja palvelusuuntia näistä. Binäärisignaalit muodostavat binäärisanan kullekin kerrokselle ja palvelusuunnille tästä. Binäärisanoja käytetään ryhmänumeroina ja myös osoitteina näitä ryhmänumeroja koskevan informaation varastoimiseksi osoitteella varustettaviin muistivälineisiin.

Joka kerran kun ryhmät määritetään uudelleen, järjestelmän valvontalaite määrittää aikavälien keskimääräisen luvun A_{SB} rakennuksessa palveluksessa olevaa hissikoria kohden. Keskusprosessori määrää myös aikavälien keskimääräisen luvun A_{SI} kussakin ryhmässä ryhmää palvelemaan pystyvää, palveluksessa olevaa koria kohden. Nämä keskiarvot varastoidaan, kunnes ryhmät määritetään uudelleen.

Järjestelmän valvontalaite lukee jaksottaisesti ja suurella nopeudella kerrostaokutsu- ja korintilainformaation ja tekee aikavälien osoitukset koreille.

Jos ryhmää palvelee vain yksi palveluksessa oleva kori, ryhmän aikaväli automaattisesti osoitetaan tälle korille, aikavälien ja kerrostaokutsujen luku näissä aikaväleissä taulukoidaan. Kukin aikaväli on osoitettu vain yhdelle korille ja osoitetut aikavälit merkitään osoittamaan, että ne on osoitettu.

Tarkoituksella osoittaa aikavälit, jotka liittyvät kerrokseen, joita palvelee useammalla kuin yhdellä hissikorilla, järjestelmän valvontalaite määrää keskiarvon A_{CB} rekisteröidyistä alaspäin- ja ylöspäin-kerrostaokutsuista rakennuksessa palveluksessa olevaa koria kohden ja se määrää kerrostaokutsujen lukumäärän keskiarvon palveluksessa olevaa koria kohden kussakin ryhmässä (A_{CI}) ja osoittaa pyyhkäisy suunnan kullekin hissikorille.

Valmistauduttaessa päivittämään aikaväliosoitusta, aikaisempi osoitus taulukko kullekin korille tyhjätkään kaikista aikaväliosoituksista paitsi niistä aikaväleistä, joita palvelee vain yhdellä korilla ja niistä aikaväleistä, joissa on rekisteröityjä kerrostaokutsuja. Osoittamattomat aikavälit osoitetaan sitten kolmessa osoitusajossa kullekin ryhmälle, ryhmät käsitellään ryhmiin kuuluvien korien lukumäärän nousevassa järjestyksessä.

Ensimmäisessä osoitusajossa ylöspäin- ja alaspäin-aikavälit, jotka liittyvät kerrokseen, jolle on parkeerattu palveluksessa oleva vapaa kori, osoitetaan tälle korille. Jos pääkerrosominaisuus ja/tai kokouskerrosominaisuus on aktiivinen, korille, joka on merkitty seuraavaksi koriksi poistumaan pääkerroksesta, annetaan pääkerroksen ylöspäin-aikaväliosoitus ja korille, jolla on osoitus kokouskerroksesta, annetaan ylöspäin- ja alaspäin-aikavälit, jotka liittyvät kokouskerrokseen. Kukin

aikaväli tutkitaan myös sen selvittämiseksi, onko korilla korikutsu, joka liittyy tähän aikaväliin. Jos näin on, korille, joka on asetettu kulkemaan ylöspäin, osoitetaan tämän kerroksen ylöspäin-aikaväli ellei tätä jo ole osoitettu eikä se ole päätekerros tälle korille. Jos se on päätekerros, sille osoitetaan tämän kerroksen alaspäin-aikaväli. Jos kori on asetettu kulkemaan alaspäin, sille osoitetaan tämän kerroksen alaspäin-aikaväli ellei tätä jo ole osoitettu eikä se ole päätekerros tälle korille. Jos se on päätekerros, se osoitetaan ylöspäin-aikaväliin tälle korille. Osoitukset tehdään keskiarvojen A_{SB} , A_{SI} , A_{CB} ja A_{CI} määäämissä rajoissa.

Toisessa osoitusajossa osoitetaan koreille ne ryhmän aikavälit, joita ei jo ole osoitettu. Kun korille osoitettujen aikavälien luku nousee arvoon A_{SI} aikaväliä ryhmässä, aikavälien osoittaminen tälle korille lopetetaan. Jos aikavälien kokonaismäärä, joka korille on osoitettu, nousee arvoon A_{SB} aikaväliä ennen kuin se saavuttaa arvon A_{SI} , aikavälien osoittaminen tälle korille lopetetaan tässä kohdassa. Jos aikaväliin liittyvien rekisteröityjen ylöspäin- ja alaspäin-kerrosta-sokutsujen lukumäärä, jotka on osoitettu korille ryhmästä, saavuttaa arvon A_{CI} ryhmälle taikka jo rekisteröityjen kerrosta-sokutsujen kokonaismäärä, jotka on osoitettu korille, saavuttaa arvon A_{CB} , kumpi tahansa edellä mainituista sattuu aikaisemmin, tälle korille osoitetaan aikavälejä, joihin ei liity kerrosta-sokutsuja, kunnes on saavutettu A_{SI} tai A_{SB} siitä riippuen, kumpi tapahtuu ensin.

Kolmas osoitusajo suoritetaan myös tarkoituksella osoittaa toisen osoitusajon jälkeen mahdollisesti jäljellä olevat osoittamattomat aikavälit, mutta soveltaen ainoastaan rajoitusta A_{CB} . Muut keski-arvot voidaan jättää huomioonottamatta tässä kolmannessa ajossa.

Edullisen suoritusmuodon kuvaus

Kuvio 1

Kuviossa 1 on esitetty hissijärjestelmä 10, jossa voidaan käyttää hyväksi esillä olevaa keksintöä. Hissijärjestelmään 10 sisältyy ryhmä hissikoreja, joista neljän korin ohjauslaitteet 14, 16, 18 ja 20 on esitetty esimerkkinä. Kuviossa on esitetty vain yksi kori 12, johon liittyy korin ohjauslaite 14; tarkoituksena on ollut yksinkertaistaa piirustusta, koska muut korit ovat samanlaisia. Jokaiseen korin ohjauslaitteeseen sisältyy korikutsun ohjaustoiminto, kerrostason valintatoiminto ja liitännäistoiminto ohjauslaitteen liittämiseksi järjestelmän valvontalaitteisiin 22. Järjestelmän valvontalaitteet 22 valvovat hissijärjestelmän toimintastrategiaa, kun hissikorit suorittavat tehtävänsä vastaamalla kerrostasoilta esitettyihin kutsuihin.

Tarkemmin sanottuna sisältyy korin ohjauslaitteisiin 14 korikutsun ohjaus 24, kerrostason valitsin 26 ja liitännäspiiri 28. Korin ohjauslaitteisiin 16 sisältyy korikutsun ohjaus 30, kerrostason valitsin 32 ja liitännäspiiri 34. Korin ohjauslaitteisiin 18 sisältyy korikutsun ohjaus 36, kerrostason valitsin 38 ja liitännäspiiri 40. Korin ohjauslaitteisiin 20 sisältyy korikutsun ohjaus 42, kerrostason valitsin 44 ja liitännäspiiri 46. Koska koriryhmän jokainen kori ja näiden ohjauslaitteet ovat samanlaisia sekä rakenteeltaan että toiminnaltaan, selitetään yksityiskohtaisesti vain korin 12 ohjauslaitteet.

Kori 12 on asennettu kuiluun 48 liikkumaan suhteessa rakennukseen 50, jossa on useita kerroksia tai porrastasoja, joista vain muutamia on esitetty piirustuksen yksinkertaistamiseksi. Kori 12 on ripustettu köyteen 52, joka on pujotettu yli kuljetuspyörän 54, joka on asennettu sopivan käyttömoottorin 56 akselille. Käyttömoottoria 56 ohjataan käyttöohjauslaitteella 57. Vastapaino 58 on liitetty köyden 52 toiseen päähän.

Korikutsut, rekisteröityinä koriin asennetulla painonappiryhmällä 60, tallennetaan ja sarjataan korikutsuohjauslaitteessa 24 ja tuloksena oleva sarjattu korikutsuinformaatio suunnataan kerrostasovalitsimeen 26.

Eteiskutsut (kerrostasokutsut), jotka rekisteröidään kerrostasoille asennetuilla painonapeilla, kuten painonapilla 62, joka sijaitsee pohjatasolla ja ylös- ja alas-painonapeilla 66, jotka sijaitsevat

välikerroksissa sekä alas-painonapilla 64, joka sijaitsee ylimmässä kerroksessa, tallennetaan ja sarjataan kerrostatokutsu-ohjauslaitteessa 68. Tuloksena oleva sarjamuotoinen kerrostatokutsuinformaatio suunnataan kaikkien hissikorien kerrosvalitsimiin sekä valvontajärjestelmän ohjauslaitteeseen 22.

Kerrosvalitsin 26 valvoo koria 12 ja kutsuja, joilla korilta vaaditaan palveluksia ja tuottaa signaaleja käytön ohjauslaitteelle 57. Kerrosvalitsin 26 tuottaa signaaleja myös sellaisten lisälaitteiden ohjaamiseksi, kuin ovikäyttölaitteet ja kerrostatolamput ja se ohjaa korikutsu- ja kerrostatokutsu-ohjauslaitteiden nollaamista, kun korikutsu taikka kerrostatokutsu on tullut palvelluksi.

Esillä oleva keksintö koskee uutta ja parannettua ryhmäohjausta useiden hissikorien ohjaamiseksi, kun nämä kulkevat tehtävissään vastaten hissipalvelua vaativiin kutsuihin ja mitä tahansa sopivaa kerrosvalitsinta voidaan käyttää. Esimerkin vuoksi oletetaan, että käytetään kerrosvalitsinta, joka on esitetty US-patentissa n:o 3 750 850. Tämä patentti kuvaa kerrosvalitsinta yhden hissikorin käyttämiseksi ottamatta huomioon korin toimintaa koriryhmässä. US-patentti n:o 3 804 209, käsittää modifikaatiota patentin n:o 3 750 850 mukaisesta kerrosvalitsimesta sovitettuna ohjattavaksi ohjelmoitavalla järjestelmäprosessorilla. Jotta vältettäisiin toistoja ja rajoitettaisiin esillä olevan hakemuksen monimutkaisuutta, nämä patentit, jotka ovat samojen hakijoiden kuin esillä oleva hakemuskinn, täten sisällytetään tähän hakemukseen viitteinä.

Valvontajärjestelmälaitteisiin 22 sisältyy prosessitoiminto 70 ja liitäntätoiminto 72. Prosessitoiminto 70 vastaanottaa korin tilasignaaleit kustakin korin ohjauslaitteesta liitäntätoiminnon 72 kautta sekä ylös- ja alas-kerrostatokutsut ja muodostaa määräyssanat kutakin korin ohjauslaitetta varten, mikä saa hissikorit palvelemaan palvelukutsuja ennaltamäärätyn strategian mukaisesti. Korin tilasignaaleit muodostavat informaation prosessitoimintoa 70 varten suhteessa siihen, mitä kukin kori voi tehdä eri kerrosten palvelemiseksi ja prosessitoiminto 70 tekee määräykset, jotka perustuvat tälle informaatiolle.

Pääkerros-ominaisuus ja kokouskerros-ominaisuus, jotka yleisesti on osoitettu viitenumeroilla 74 ja 76, voidaan aktivoida aikaansaamaan erityisiä valinnaisia strategioita, kuten jäljempänä selitetään.

Valvontalaitteet 22 muodostavat ajastussignaalin CLOCK järjestelmän ajastusfunktion 78 synkronoimiseksi. Järjestelmän ajastustoiminto 78 muodostaa ajastussignaalit datavirtauksen ohjaamiseksi hissijärjestelmän eri toimintojen välillä.

Kuvio 2

Kuvio 2 esittää tiettyjä ajastussignaaleja, jotka aikaansaadaan ajastustoiminnolla 78, kuvion 2 ajastussignaalien vastatessa kokonaista pyyhkäisyjaksoa. Hissijärjestelmä 10 on pohjaltaan sarjamultipleksijärjestelmä ja tästä syystä täytyy synnyttää täsmällinen ajastus tietojen esittämiseksi oikeassa ajallisessa suhteessa toisiinsa. Kullekin rakennuksen palveltavalle kerrokselle on osoitettu oma aikavälinsä kussakin aikajaksossa ja täten jakson aikavälien lukumäärän määrää asiantomaisen rakennuksen kerrosten lukumäärä. Kullakin kerroksella on erillinen aikaväli, joka liittyy tähän kerrokseen, mutta ei ole välttämätöntä, että kullekin aikavälille on osoitettu kerrostaso. Pyyhkäisyajavälejä synnytetään 16, 32, 64 tai 128 aikavälin jaksoina, niin että jakso valitaan siten, että käytettävissä on vähintään niin monta pyyhkäisyajaväliä kuin talossa on kerrostasoja. Esimerkin vuoksi oletetaan, että tässä kuvatussa rakennuksessa on 16 kerrostasoa, niin että 16 aikaväliä käsittävä jakso on riittävä.

Kuudentoista pyyhkäisyajavälin jakso synnytetään binäärilaskimella, jossa on ulostulot SOS, S1S, S2S ja S3S, kuten on esitetty kuviossa 2. Välin 00 binääriosoitte on 0000, riippuvaisesti S3S:stä, S2S:stä, S1S:stä ja SOS:stä ja pyyhkäisyajavälin 01 binääriosoitte on 0001 jne.

Pyyhkäisyvälajakso on jaettu kahteen samanlaiseen osaan, so. 8 pyyhkäisyajaväliä kummassakin, jako on suoritettu ajastussignaaleilla SECO ja SECl. Signaali SECO on tosi pyyhkäisyjaoson ensimmäiselle puolikkaalle kun taas signaali SECl on tosi pyyhkäisyjakson viimeiselle puolikkaalle. Ajastussignaalit DECO - DEC7 ovat kukin tosia eri pyyhkäisyajaväleille pyyhkäisyjakson kunkin ensimmäisen puolikkaan aikana tosien pyyhkäisyajavälien ollessa erotettuja toisistaan seitsemällä pyyhkäisyajavälillä. Täten mikä tahansa 16 pyyhkäisyajavälistä voidaan valita loogisesti yhdistämällä yksi signaaleista DECO - DEC7 yhden signaalin SECO tai SECl kanssa. Ajastussignaali MXCT esimerkiksi on tosi ainoastaan viimeisen pyyhkäisyajavälin, so. pyyhkäisyajavälin 15 aikana kunkin jakson kuluessa ja se tuotetaan signaalien DEC7 ja SECl loogisella yhdistelmällä.

Kuvio 3

Kuvio 3 kuvaa ajastussignaaleja, jotka siis on aikaansaatu ajastustoiminnolla 78, kuvion 3 ajastussignaalien ollessa riippuvaisuussuhteessa niihin signaaleihin, jotka liittyvät pyyhkäisyajaväliin. Kuvion 3 esittämät ajastussignaalit synnytetään jokaisen pyyhkäisyajavälin aikana paitsi signaaleja S100 ja S300, jotka esiintyvät vain pyyhkäisyajavälin 00 kuluessa.

Peruskelloa CL käytetään signaalin K08 johtamiseen, joka jakaa pyyhkäisyajavälin 8 samanlaiseen osaan ja signaalia K08 siirretään eteenpäin 90° signaalin K08S tuottamiseksi. Signaali K02 jakaa pyyhkäisyajavälin kahteen samanlaiseen osaan ja signaalia K02 siirretään eteenpäin 90° tuottamaan signaali K02S. Pulssisignaalit (strobe signals, aikavalintasignaalit) STA, STB, STC ja STD ovat kukin tosia pyyhkäisyajavälin eri neljänneksellä, so. toisella, neljännellä, ensimmäisellä ja kolmannella neljänneksellä. Signaalit S100 ja S300 esiintyvät vain pyyhkäisyajavälin 00 ensimmäisen ja vast. kolmannen neljänneksen aikana.

Kuvattaessa kuvion 1 mukaista hissijärjestelmää 10 yksityiskohtaisemmin on apua siitä, että esitetään eri signaalit ja niiden toiminnot, joihin jäljempänä viitataan, sekä symbolit, joita käytetään ohjelman identifioijina ja ohjelmanmuuttujina vuokaavioissa.

<u>SYMBOLI</u>	<u>TOIMINTA</u>
ACCU	Akkurekisteri CPU:ssa
A_{CB}	Keskimääräinen kutsujen lukumäärä rakennuksessa käytössä olevaa hissikoria kohden
A_{CI}	Keskimääräinen kutsujen lukumäärä ryhmässä käytössä olevaa hissikoria kohden
A_{SB}	Keskimääräinen pyyhkäisyajavälien lukumäärä rakennuksessa käytössä olevaa koria kohden
A_{SI}	Keskimääräinen pyyhkäisyvälien lukumäärä ryhmässä käytössä olevaa koria kohden, jotka ovat kykeneviä palvelemaan ryhmää
AVAS	Kori on käytettävissä kerrosvalitsimen mukaan
AVP0-AVP3	Edenneen korin kerrosasema binäärisenä
BYPS	Tosi, kun kori on ohittamassa kerrostasokutsuja
CALL	Tosi, kun korilla on korikutsu tai kerrostasokutsu osoitetussa pyyhkäisyajavälissä

<u>SYMBOLI</u>	<u>TOIMINTA</u>
<u>CLOCK</u>	Ajastussignaali, jonka on pannut alkuun järjestelmä-proessori
CM-RAM 1	Käskyohjausjohto CPU:sta 4:ään RAM:iin
CM-RAM 2	Käskyohjausjohto CPU:sta 4:ään RAM:iin
CM-ROM	Käskyohjausjohto CPU:sta 16:een ROM:iin
<u>COMO-COM3</u>	Sarjaohjaussignaali järjestelmäproessoriliitännästä 4 hissikoriin
CONV	Tosi, kun korilla on kokouskerrososoitus
CY	Muistinumero-linkki fli-flop CPU:ssa
<u>DATO-DAT3</u>	Sarjasignaali 4 hissikorista järjestelmäproessoriliitännään
DNAC	Todellinen niiden hissien lukumäärä, jotka on saatettu liikkumaan alaspäin
DNDES	Haluttu niiden korien lukumäärä, jotka on asetettu liikkumaan alaspäin
<u>DNIN</u>	Kerrostaso-alaskutsuesto, tosi, kun kori on estetty vastaamasta kerrostaso-alaskutsuun asianomaisella tasolla
<u>DOPN</u>	Käsky järjestelmäproessorista korin ovien avaamiseksi
D0-D3	4-bitin databussi järjestelmäproessorissa
<u>D89T</u>	Tosi, kun moottorigeneraattori on katkaistuna
FEN	Kerrostaso kykenevä - tosi kerrostasoille, joilla kori pystyy vastaamaan kerrostasokutsuihin ainakin toises-sa palvelusuunnassa
HRT	Puolet edestakaisesta kierroksesta
IDLE	Tosi, kun kori on palveluksessa, NOT NEXT, ja käytettävissä kerrosvalitsimen mukaan
INSC	Tosi, kun kori on palveluksessa järjestelmäproessorilla
INSV	Tosi, kun kori on palveluksessa järjestelmäproessorilla eikä ole ohittamassa kerrostasokutsuja
<u>INO-INT5</u>	16 sisäänmenoa järjestelmäproessoriin
<u>MDCL</u>	Ovisignaali, joka on tosi, kun ovet ovat suljettuina
MT00	Muistiurasignaali, joka on tosi kerroksille, joilla kori pystyy vastaamaan kerrostasokutsuihin ylöspäin
MT01	Muistiurasignaali, joka on tosi kerroksille, joilla kori pystyy vastaamaan kerrostasokutsuihin alaspäin

<u>SYMBOLI</u>	<u>TOIMINTA</u>
<u>MXCT</u>	Ajastussignaali, joka on tosi pyyhkäisyjakson viimeisen pyyhkäisyajakavälin aikana
N_{HCl}	Kerrostasokutsujen lukumäärä, jotka on osoitettu koriin yhden korin ryhmästä
N_{HCT}	Kerrostasokutsujen kokonaismäärä, jotka on osoitettu koriin tähän asti
N_{CI}	Kerrostasokutsujen lukumäärä, jotka on osoitettu koriin tähän asti, kysymyksessä olevassa ryhmässä
N_{CP}	Laskin, joka pannaan aloittamaan laskenta riippuvaisesti korin asemasta
N_{DIST}	Kelpaavien pyyhkäisyajakavälien lukumäärä korista tähän asti määräysrutiinissa (käytetään sen määrittämiseksi, milloin on kohdattu puolen edestakaisen matkan rajoitus)
<u>NEXT</u>	Signaali järjestelmäprosessorista, joka on tosi, kun kori on osoitettu seuraavaksi koriksi poistumaan pääkerroksesta
N_{POS}	Pyyhkäisyajakavälinumero, joka vastaa korin asemaa
N_{RCC}	Rekisteröityjen kerrostasokutsujen lukumäärä, jotka on osoitettu korille ryhmässä, jota palvelee useampi kuin yksi kori
N_{SC}	Hissijoukossa palveluksessa olevien korien lukumäärä
N_{SCI}	Niiden korien lukumäärä, jotka pystyvät palvelemaan ryhmää
N_{SCF}	Niiden korien lukumäärä, jotka pystyvät palvelemaan kokouskerrosta
N_{SI}	Pyyhkäisyajakavälien lukumäärä, joka on osoitettu korille tähän asti, kysymyksessä olevassa sarjassa
N_{SMF}	Niiden korien lukumäärä, jotka voivat palvella pääkerrosta
N_{SS}	Pyyhkäisyajakavälien kokonaismäärä, jotka on osoitettu korille tähän asti
<u>OUT0-OUT4</u>	Sarjasignaalit järjestelmäprosessorista järjestelmäprosessoriliitintään
<u>PFONFL</u>	Signaali, joka on tosi, kun kokouskerrosominaisuus on aktivoitu
<u>PCFLO-PCFL3</u>	Kokouskerroksen binääriosoitte
<u>PKFL</u>	Parkkeeraussignaali järjestelmäprosessorista
<u>PMNFL</u>	Signaali, joka on tosi, kun pääkerros-ohjelmakohta on aktivoitu

<u>SYMBOLI</u>	<u>TOIMINTA</u>
<u>PMNFO-PMNFL3</u>	Pääkerroksen binääriosoitte
Q_{MNF}	Niiden korien suhteellinen määrä, jotka pidetään pääkerroksessa
RAM	Poimintamuisti
RES	Palautussignaali, jota käytetään käynnistämään valvontajärjestelmä-ohjauslaitteet
ROM	Pysyväismuisti
<u>SDT</u>	Käsky järjestelmäprosessorista kerrosta-sovalitsimen asettamiseksi alaspäin-matkaan
<u>SUT</u>	Käsky järjestelmäprosessorista kerrosta-sovalitsimen asettamiseksi ylöspäin-matkaan
SYNC	Synkronointisignaali, jonka järjestelmäprosessori synnyttää käskyjakson käynnistyksessä
UPAC	Niiden korien todellinen lukumäärä, jotka on asetettu ylöspäin liikkeeseen
UPDES	Niiden korien haluttu lukumäärä, jotka on asetettu ylöspäin liikkeeseen
<u>UPIN</u>	Ylöspäin kutsun estosignaali järjestelmäprosessorista
UPSCAN	Pyyhkäisysuunta pyyhkäisyaikavälien osoittamiseksi korille, 1 = ylös, 0 = alas
WT50	Osoittaa korin kuormaa, 1 = suurempi kuin 50 %, 0 = pienempi kuin 50 %
1Z	Sarjamuotoiset kerrosta-sokutsut ylöspäin
2Z	Sarjamuotoiset kerrosta-sokutsut alaspäin
3Z	Sarjamuotoiset korikutsut
Ø1	Kahden toisensa ei-limittävän kellon vaihe 1 järjestelmäprosessorissa
Ø2	Kahden toisensa ei-limittävän kellon vaihe 2 järjestelmäprosessorissa

Kuvio 4

Kuvio 4 on kaaviokuva järjestelmäprosessorista 70, jota voidaan käyttää kuviossa 1 lohkomuodossa esitetyn valvontajärjestelmälaitteen 22 prosessitoimintoa 70 varten. Järjestelmäprosessorina 70 voidaan käyttää sopivaa mikroprosessoria (pienoisprosessoria), kuten esimerkiksi jotakin edellä mainituista mikroprosessoreista. Esimerkkinä

selitetään seuraavassa Intel Corporation MSC-4 micro computer set.

MCS-4 mikroprosessoriin sisältyy 4-bitin rinnakkaisohjauslaitteet ja aritmeettinen yksikkö 80 (Intelin 4004), jota seuraavassa nimitetään CPU 80, ohjausmuisti 82, johon sisältyy joukko ohjelmoitavia pysyväismuisteja (ROMS), kuten ROM 1 - ROM N (intel'in 4001), datamuisti 86, johon sisältyy joukko poimintamuisteja (RAMS), kuten RAM 1 - RAM N (intel'in 4002), kellot 88, jotka synnyttävät perusjärjestelmäajastuksen (750 kHz) kahden ei-limittyvän kellovaiheen Ø1 ja Ø2 muodossa, manuaalinen palautus 92 ja kello 94, joka tuottaa ajastussignaaleja $\overline{\text{CLOCK}}$ ulkopuolisia laitteita varten ja joka on riippuvainen CPU 80 tuottamasta ajastuksesta.

CPU 80 kommunikoi ohjausmuistin 82 ja datamuistin 86 kanssa neljän databussin D0, D1, D2 ja D3 kautta ja hissijärjestelmän periferiaosan kanssa ohjaus- ja datamuisteissa 82 ja 86 olevien tulo- ja lähtöveräjäien kautta. CPU 80 sisältää ohjausjohdon jokaista neljän RAMin sarjaa kohden, kuten ohjausjohdot CM-RAM 1 ja CM-RAM 2, ohjausjohdon CM-ROM, jota käytetään ohjaamaan ROM-joukkoa, jossa voi olla aina 16 ROMia. CPU 80 on yhdistetty kelloihin 88 ja 90 ja reagoi näille, so. jokaisella 8 kelloperiodilla antaa ulos synkronointisignaalin SYNC. Signaali SYNC lähetetään ohjaus- ja datamuisteihin 82 ja 86 ja kelloon 94 osoittamaan 10,8 mikrosekunnin käskyjakson käynnistystä.

CPU 80 on yhdistetty manuaaliseen palautukseen 92 ja siinä on testinasta, joka on kytketty vastaanottamaan signaali $\overline{\text{MXCT}}$. Signaali $\overline{\text{MXCT}}$ synnytetään kuvion 6 esittämässä laitteistossa ja, kuten on esitetty kuvion 2 ajastusdiagrammassa, se on tosi kunkin pyyhkäisyjakson viimeisen pyyhkäisyajavälän aikana.

Kukin ROMeista on kytketty databussiin D0, D1, D2 ja D3, kellovaiheisiin Ø1 ja Ø2, ROM-ohjausjohtoon CM-ROM, synkronointijohtoon SYNC ja palautukseen 92. Kussakin ROMissa 1, 2, 3 ja 4 on 4 sisääntuloa tuloinformaation vastaanottamiseksi hissijärjestelmästä, näitä 16 sisäänmenoa on merkitty IN0 - IN15.

Kukin RAMEista on kytketty databusseihin D0, D1, D2 ja D3, kellovaiheisiin Ø1 ja Ø2, yhteen RAM-ohjausjohtoon CM-RAM 1 tai CM-RAM 2, synkronointijohtoon SYNC ja palautukseen 92. RAMEissa 1 ja 3 on kummassakin ulostulot informaation lähettämiseksi hissijärjestelmään,

näitä ulostuloja on merkittu $\overline{OUT0}$ - $\overline{OUT4}$.

Palautus 92 aktivoidaan manuaalisesti hissijärjestelmän käynnistytksen aikana. Matala palautussignaali (tyhjäyssignaali, nollaussignaali) tyhjä muistit ja rekisterit CPU 80:ssa, se asettaa databussin nolnaan, se tyhjä staattiset flip-flopit ohjausmuistissa 82 sekä estää datan ulostulon ja se tyhjä datamuistin 86.

Kelloon 94 voi sisältyä JK flip-flop 96 ja NPN transistori 98. Flip-flopin 96 J ja K sisäänmenot on kytketty yksisuuntaiseen syöttöjännitteeseen navassa 99 ja sen kellosisäänmeno C on kytketty synkronointijohtoon SYNC. Sen Q ulostulo on kytketty vastuksen 100 kautta transistorin 98 kantaan. Transistorin 98 kanta on myös kytketty maahan vastuksen 102 kautta, sen emitteri on kytketty maahan ja sen kollektori on kytketty ulostulonapaan $\overline{CLOCK}$. Signaali SYNC on matala 10,8 mikrosekunnin käskyjakson viimeisen alijakson (1,35 mikrosekuntia) aikana ja flip-flop 96 vaihtaa ulostulotilansa SYNC-signaalin positiiviseksi menevällä muutoksella. Täten, signaali $\overline{CLOCK}$ on sakaraalto, kunkin puolijakson ollessa täydellinen käskyjakso (10,8 mikrosekuntia).

CPU 80 sisältää osoiterekisterin, indeksirekisterin, 4-bittisummaimen ja käskyrekisterin. Indeksirekisterinä on 16 x 14 bitin poimintamuisti. 16 neljän bitin asemaa, joita on merkitty viitemerkeillä R0 - R15, voidaan suoraan osoitteistaa laskentaa ja ohjausta varten ja ne voidaan myös osoitteistaa 8 parina muistiasemia, joita on merkitty viitemerkeillä P0 - P7, RAMien tai ROMien osoitteistamiseksi, tai datan varastoimiseksi ROMeihin.

Kukin ohjausmuistin 82 ROMeista varastoi 256 x 8 sanaa ohjelma- tai datataulukkoista ja se on varustettu 4 I/O nastalla ja ohjauslaitteilla sisääntulo- ja ulosmeno-operaatioiden suorittamiseksi. CPU 80 lähettää osoitteen ohjausmuistiin yhdessä ROM-numeron kanssa, kolmen ensimmäisen käskyalajakson aikana ja valittu ROM lähettää käskyn CPU 80:een seuraavien kahden käskyalajakson aikana. Käsky suoritetaan, so. data operoidaan CPU 80:ssa taikka data tai osoite lähetetään CPU 80:een tai CPU 80:sta käskyjakson kolmen viimeisen alijakson aikana. Kun I/O käsky vastaanotetaan ohjausmuistista 82, data siirretään CPU 80 akkuun tai akusta 4 datajohdolla, jotka on kytketty ohjausmuistiin 82.

Kukin datamuistin 86 RAMEista varastoi 320 bittiä, jotka on järjestetty 4 rekisteriin, joissa kussakin on kaksikymmentä 4 bitin merkkiä, joista 16 voidaan varustaa osoitteilla yhdellä käskyllä ja joista 4 voidaan varustaa osoitteilla toisella käskyllä. Kunkin rekisterin 16 bittiä muodostavat päämuistin samalla kun 4 bittijä muodostavat tilamerkkimuistin. Yhden RAMin, rekisterin ja merkin osoite varastoidaan kahteen indeksirekisteriin CPU 80:ssa ja siirretään valittuun RAMiin käskyjakson kahden alijakson aikana, kun RAM käsky suoritetaan. Kun RAM ulostulokäsky on vastaanotettu CPU 80:llä, siirretään CPU 80:n akun sisältö neljään RAM ulostulojohtoon.

Kuvio 5

Kuviossa 5 on esitetty RAM kartta, joka kaaviollisesti esittää 16 rekisteriä, rekistereitä 0 - 15, datamuistissa 86. Alimmat neljä riviä muodostavat rekisterin tilamerkkimuistit samalla kun ylemmät 16 riviä, jotka on numeroitu 00 - 15, muodostavat rekistereiden päämuistin. Rekistereiden erityistoiminnot selitetään seuraavassa, kun viitataan niihin varastoituihin signaaleihin ja dataan.

Kuviot 6 ja 7

Kuvio 6 on kaaviokuva prosessiliitännästä 72, jota voidaan käyttää kuviossa 1 lohkomuodossa esitettyyn toimintoon 72. Kukin neljästä hissikorista lähettää tilasignaalin liitännän 72 kautta valvontaohjauslaitteiden 22 järjestelmäprosessoriin 70. Kunkin korin tilasignaaliit sarjataan multipleksoreilla, kuten jäljempänä selitetään kuvioon 8B liittyen, näitä sarjasignaaleja hissikoreista 0, 1, 2 ja 3 on merkitty symboleilla $\overline{DAT0}$, $\overline{DAT1}$, $\overline{DAT2}$ ja $\overline{DAT3}$.

Kerrostasokutsut ylös ja alas sarjataan (muunnetaan sarjamuotoon) kukin kerrostasokutsujen ohjauslaitteessa 68, joka on kuvattu kuviossa 1, sarjamuodossa olevia kerrostasokutsuja ylös ja alas on merkitty symboleilla $\overline{1Z}$ ja $\overline{2Z}$. Sarjasignaaliit $\overline{DAT0}$, $\overline{DAT1}$, $\overline{DAT2}$, $\overline{DAT3}$, $\overline{1Z}$ ja $\overline{2Z}$ johdetaan liitännään 72. Kerrostasokutsut ylöspäin $\overline{1Z}$ ja kerrostasokutsut alaspäin $\overline{2Z}$ yhdistetään liitännässä 72 tilasignaaleihin $\overline{DAT0}$ ja $\overline{DAT1}$. Tämä toteutetaan jakamalla kukin pyyhkäisyajaväleistä 00 - 15 kuviossa 2 neljään osaan käyttämällä aikavalintapulsseja (strokes) STC, STA, STD ja STB, jotka on esitetty kuviossa 3.

Kuvio 7 kuvaa signaalien muotoja liitännästä 72 järjestelmäprosessoriin kuvaten kaaviollisesti, miten kukin pyyhkäisyajaväli on jaettu neljään neljänneksen aikavalintapulsseilla. Tilasignaaliit koreista

esiintyvät pyyhkäisyaikavälin ensimmäisenä neljänneksenä, joka on valittu pulssilla STC. Kerrostasokutsut ylöspäin esiintyvät pyyhkäisyaikavälien aikavalintapulssilla STA määrättyssä toisessa neljänneksessä aikasignaalia $\overline{INO}$ korista 0. Kerrostasokutsut alaspäin esiintyvät sarjasignaalin $\overline{INL}$ korista 1 pyyhkäisyaikavälien toisessa, STA:lla määrättyssä neljänneksessä. Pyyhkäisyaikavälien toista neljänneistä korien 2 ja 3 sarjasignaaleiden suhteen ei käytetä. Kerrostasokutsut sarjasignaaleissa $\overline{IZ}$ ja $\overline{ZZ}$ esiintyvät pyyhkäisyaikavälissä, joka liittyy kerrokseen, joista kutsut on rekisteröity.

Kunkin pyyhkäisyaikavälin kolmas neljännes, joka on määrätty aikavalintasisäänmenosignaalin STD, sisältää kerrostason kykeneväksi tekevän signaalin $\overline{FEN}$. Jos kori pystyy palvelemaan kerrosta, signaali $\overline{FEN}$ on tosi tähän kerrokseen liittyvän pyyhkäisyaikavälin aikana.

Kunkin pyyhkäisyaikavälin neljäs neljännes, joka on määrätty aikavalintasisäänmenosignaalin STB, sisältää korikutsut $\overline{ZZ}$. Jos korissa on korikutsu tiettyyn kerrokseen, se osoitetaan sen pyyhkäisyaikavälin neljännessä neljänneksessä, joka liittyy tähän kerrokseen.

Viittaamme jälleen kuvioon 6, jossa sarjasignaalit, jotka esiintyvät signaalissa $\overline{IZ}$, synkronoidaan pulssilla STA kaksoissisäänmeno-NAND-veräjässä 110, pulssin STA ollessa yhdistettynä yhteen sisäänmenoon NAND-veräjässä ja signaalin $\overline{IZ}$ ollessa kytkettynä toiseen sisäänmenoon invertterin tai NOT-veräjän 112 kautta. Sarjakutsut ylös, $\overline{IZ}$, viedään sarjasignaaliin $\overline{DATO}$ korista 0 kaksoissisäänmeno-NAND-veräjässä 114, NAND-veräjän 110 ulostulon ollessa kytkettynä yhteen sisäänmenoon NAND-veräjässä 114 ja signaalin $\overline{DATO}$ ollessa kytkettynä toiseen sisäänmenoon. NAND-veräjän 114 ulostulo on kytketty ulostulonapaan $\overline{INO}$ kääntävän puskurin 116 kautta. Puskuriin 116 voi sisältyä NPN-transistori 118 ja vastukset 120 ja 122. NAND-veräjän 114 ulostulo on kytketty transistorin 118 kantaan vastuksen 120 kautta ja kanta on kytketty maahan vastuksen 122 kautta. Kollektorielektrodi on kytketty ulostulonapaan $\overline{INO}$ ja emitterielektrodi on kytketty maahan.

Signaali $\overline{DATO}$ on korkea kunkin pyyhkäisyaikavälin toisen neljänneksen aikana tehden NAND-veräjän 114 kykeneväksi. Sarjamuodossa oleva ylös-kutsu (UP) pyyhkäisyaikavälille, joka kutsu esiintyy signaalissa $\overline{IZ}$, saattaa NAND-veräjän 110 ulostulon matalaksi STA:n aikana ja NAND-veräjän 114 ulostulo tulee korkeaksi sellaisen aikavälin kunkin

toisen neljänneksen aikana, jossa on ylöspäinkutsu. NAND-veräjän 114 korkea ulostulo kytkee transistorin 118 johtavaan tilaan ja ulostulonapa $\overline{IN0}$ tulee kytketyksi maahan. Jos pyyhkäisyajavälin kuluessa ei ole kerrostasokutsua ylöspäin, on NAND-veräjän 110 ulostulo korkea ja NAND-veräjän 114 ulostulo matala. Täten transistori 118 on ei-johdavassa tilassa ja ulostulonapa $\overline{IN0}$ on +15 voltissa, kuten on osoitettu kuviossa 4. Ensimmäisen, kolmannen ja neljännen jakson aikana pulssisignaali STA on matala ja NAND-veräjän 110 ulostulo on korkea, tehden NAND-veräjän 114 kykeneväksi läpäisemään tosisignaalit pyyhkäisyajavälin näiden kolmen neljänneksen aikana, jotka signaalit esiintyvät signaalissa $\overline{DAT0}$.

Samalla tavoin kerrostaso-alaskutsut, jotka esiintyvät sarjasignaalisissa $\overline{2Z}$, sijoitetaan niiden pyyhkäisyajavälien toiseen neljännekseen, joihin ne liittyvät, käyttäen kaksoissisäänmeno-NAND-veräjiä 124 ja 126, invertterinä 128 ja puskuria 130. Pulssi STA on kytketty NAND-veräjän 124 yhteen sisäänmenoon ja signaali $\overline{2Z}$ on invertterin 128 kautta kytketty toiseen sisäänmenoon. NAND-veräjän 124 ulostulo on kytketty yhteen sisäänmenoon NAND-veräjässä 126 ja sarjasignaali $\overline{DAT1}$ korista 1 on kytketty toiseen sisäänmenoon. NAND-veräjän 126 ulostulo on kytketty ulostulonapaan $\overline{IN1}$ puskurin 130 kautta. Puskuri 130 on samanlainen kuin puskuri 116, kuten muutkin ulostulopuskurit kuviossa 6 ja tästä syystä ne on esitetty lohkomuodossa.

Sarjasignaali $\overline{DAT2}$ korista 2 on kytketty ulostulonapaan $\overline{IN2}$ invertterin 132 kautta ja ulostulopuskurin 134 kautta ja sarjasignaali $\overline{DAT3}$ on kytketty ulostulonapaan $\overline{IN3}$ invertterin 136 ja ulostulopuskurin 138 kautta.

Hissijärjestelmää 10 voidaan käyttää joko siten, että yksi kerros on osoitettu pääkerrokseksi tai ilman, että mitään kerrosta on merkitty pääkerrokseksi, pääkerrosominaisuutta on kuviossa 1 esitetty loholla 74. Edelleen, kun järjestelmää käytetään pääkerrosmuodossa, mikä kerros tahansa rakennuksessa voidaan valita pääkerrokseksi pääkerros-binäärinumeron avulla. Jos hissijärjestelmää käytetään pääkerrosmuodossa, valitaan ennalta määrätty suhteellinen osuus, joka osoittaa niiden korien haluttua määrää, jotka pidetään pääkerroksessa ja tätä suhteellista osuutta voidaan automaattisesti modifioida vallitsevilla liikenneolosuhteilla. Esimerkiksi neljän korin järjestelmässä pääkerroksen suhteellinen osuus voidaan valita yhdeksi, joka modifioidaan kahdeksi huippuliikenne-ylöspäin tilassa ja nolaksi huippuliikenne-alaspäin-tilassa.

Huippuliikenne-ylöspäin-tila voidaan todeta sillä, että kori poistuu pääkerroksesta ylöspäin ennaltamäärätyllä kuormalla ja jos järjestelmä ei ole huippuliikenteessä alaspäin, tämä ilmiö käynnistää ajastimen saattamaan järjestelmän huippuliikenteeseen ylöspäin ennalta määrättyksi ajaksi. Kukin seuraava kori, joka poistuu pääkerroksesta asetettuna liikkumaan ylöspäin, on asetettu ohittamaan kerroskutsut, palauttaa ajastimen tämän maksimilaskentaan pidentämään aikaa, jonka järjestelmä on huippuliikenteessä ylöspäin.

Tila huippuliikenne alaspäin voidaan todeta sillä, että pääkerroksen yläpuolella oleva kori synnyttää ohitussignaalin alaspäinsuunnassa. Tämä tapahtuma myös käynnistää huippuajastimen, sijoittaen järjestelmän huippuliikenteeseen alaspäin ennaltamäärätyksi ajaksi, ohittaen ylöspäin huipun, jos järjestelmä sattuisi olemaan myös huippuliikenne-ylöspäin-tilassa.

Pääkerrosominaisuus valitaan kytkimellä (ei esitetty), joka on yhdistetty sisäänmenonapaan $\overline{PMNFL}$, joka on esitetty kuviossa 6. Kytkin antaa suhteellisen suuren jännitteen sisäänmenonapaan $\overline{PMNFL}$, kun ei haluta pääkerrosominaisuutta ja pienen jännitteen tai maapotentiaalisignaalin, kun mainittua ominaisuutta halutaan. Sisäänmenonapa $\overline{PMNFL}$ on kytketty korkean tason sisäänmenoliitäntään 140. Liitäntään 140 voi sisältyä operaatiovahvistin 142, vastukset 144, 146 ja 148, kondensaattori 150 ja diodi 152. Vastus 144 on kytketty vahvistimen 142 ulostulosta sen ei-kääntävään sisäänmenoon. Sen kääntävä sisäänmeno on kytketty positiiviseen yksisuuntaiseen jännitelähteeseen, esimerkiksi 12 voltin jännitteeseen vastuksen 146 kautta. Sen ei-kääntävä sisäänmeno on kytketty sisäänmenonapaan $\overline{PMNFL}$ vastuksen 148 kautta, maahan kondensaattorin 150 kautta ja maahan diodin 152 kautta. Diodi 152 on siten napaistettu, että se johtaa virtaa maasta ei-kääntävään napaan. Kun jännite navassa $\overline{PMNFL}$ on korkea, mikä osoittaa, että pääkerrosominaisuutta ei haluta, ylittää sisäänmenonavan $\overline{PMNFL}$ jännite sen jännitteen, joka syötetään kääntävään sisäänmenoon ja operaatiovahvistimen 142 ulostulo on positiivinen, so. loogisella tasolla yksi, joka käännetään invertterillä 154 loogiseksi tasoksi nolla ja johdetaan ulostulopuskuriin 156. Puskuri 156 kääntää loogisen nollan loogiseksi ykköseksi ja johtaa loogisen ykkösen ulostulonapaan $\overline{IN5}$. Kun signaali $\overline{PMNFL}$ on tosi (matala), ylittää kääntävään sisäänmenoon johdettu jännite ei-kääntävään sisäänmenoon johdetun jännitteen ja operaatiovahvistimen 142 ulostulo menee loogiselle nol-

latasolle. Inverttteri (kääntäjä) 154 kääntää tämän signaalin loogiseksi ykköseksi ja puskuri 156 kääntää tämän loogiseksi nollaksi, joka on tositaso ulostulonavalle $\overline{IN5}$.

Sen kerroksen binääriosoitte, joka on valittu pääkerrokseksi, johdetaan sisäänmenonapoihin $\overline{PMNFL0}$, $\overline{PMNFL1}$, $\overline{PMNFL2}$ ja $\overline{PMNFL3}$. Näihin sisäänmenonapoihin johdetut signaalit johdetaan ulostulonapoihin $\overline{IN8}$, $\overline{IN9}$, $\overline{IN10}$ ja $\overline{IN11}$, jokainen korkean tason liitännän, inverttlerin ja ulostulopuskurin kautta, jotka yleisesti on osoitettu viitenumeroilla 158, 160 ja 162. Korkean tason sisäänmenoliitäntä on yleisesti osoitettu viitenumerolla 158, kaikki muut korkean tason sisäänmenoliitännät, jotka on esitetty kuviossa 6, ovat samanlaisia kuin liitäntä 140.

Hissijärjestelmää 10 voidaan käyttää joko siten, että jokin kerros on määrätty kokouskerrokseksi tai ilman tätä ominaisuutta, sen mukaan kuin halutaan, kokouskerrosominaisuutta on osoitettu viitenumerolla 76 kuviossa 1. Kokouskerrokseksi voidaan määrätä mikä tahansa kerros pääkerroksen yläpuolella minä ajankohtana tahansa binäärinumerolla. Kun tämä ominaisuus on voimassa, käynnistettynä esimerkiksi manuaalisella kytkimellä, eikä mitään koria ole osoitetussa kerroksessa, vale tai virhekuutsu osoitetaan jokaiselle korille, kunnes kori pysähtyy tähän kerrokseen.

Kokouskerrosominaisuus valitaan kytkimellä, joka on kytketty sisäänmenonapaan $\overline{PCONFL}$ kuviossa 6. Samalla tavoin kuin signaali, jolla valitaan pääkerrosominaisuus, signaali $\overline{PCONFL}$ johdetaan korkean tason sisäänmenoliitännäänsä 164, jonka ulostulo käännetään invertterillä 166 ja johdetaan ulostulopuskuriin 168. Ulostulopuskuri 168 on kytketty ulostulonapaan $\overline{IN6}$.

Kokouskerrokseksi valitun kerroksen binääriosoitte on kytketty sisäänmenonapoihin $\overline{PCFL0}$, $\overline{PCFL1}$, $\overline{PCFL2}$ ja $\overline{PCFL3}$. Signaalit, jotka on johdettu näihin sisäänmenonapoihin, johdetaan ulostulonapoihin $\overline{IN12}$, $\overline{IN13}$, $\overline{IN14}$ ja $\overline{IN15}$, kukin korkean tason liitännän, inverttlerin ja ulostulopuskurin kautta, joita on yleisesti osoitettu viitenumeroilla 170, 172 ja 174.

Signaali $\overline{MXCT}$ CPU 80 varten, josta edellä mainittiin, tuotetaan kytkemällä ajastussignaali DEC7 ja SECl kahteen sisäänmenoon kaksois-sisäänmeno-NAND-veräjässä 176. Molemmat signaalit DEC7 ja SECl ovat

korkeita viimeisen pyyhkäisyavälin aikana, kuten on esitetty kuviossa 2, aiheuttaen NAND-veräjän 176 ulostulon tulon matalaksi tänä aikana. NAND-veräjän 176 matala ulostulo käännetään loogiseksi ykköseksi invertterillä 178 ja puskuri 180 kääntää tämän loogiseksi nol-laksi. Puskurin 180 ulostulo on kytketty ulostulonapaan $\overline{MXCT}$.

Kuviossa 4 esitetyt ulostulonavat $\overline{OUT0}$, $\overline{OUT1}$, $\overline{OUT2}$ ja $\overline{OUT3}$ datamuis-tista 86 tällä välin tuottavat sarjamuotoisia datasanoja hissikoreil-le 0, 1, 2 ja 3. Nämä datasanat sisältävät kiellot ja käskyt, jotka saavat hissikorit vastaamaan hissipalvelua vaativiin kutsuihin järjes-telmäprosessorin 70 operaatiostrategian mukaisesti. Nämä ulostulona-vat yhdessä ulostulonavan $\overline{OUT4}$ kanssa on kytketty kuvioissa 1 ja 6 esitettyyn prosessoriliitintään 72. Lisää ulostulonapoja datamuistis-ta 86 olisi muodostettava hissijärjestelmille, joissa on enemmän kuin 4 koria.

Navat $\overline{OUT0}$, $\overline{OUT1}$, $\overline{OUT2}$ ja $\overline{OUT3}$ on kytketty vastaavasti ulostulonapoi-hin $\overline{COM0}$, $\overline{COM1}$, $\overline{COM2}$ ja $\overline{COM3}$ kukin invertterin ja kääntävän ulostulo-puskurin kautta, joita on yleisesti osoitettu viitenumeroilla 182 ja 184.

Ulostulonapaa $\overline{OUT4}$ käytetään käynnistämään ulkopuolinen ajastin 190 sisäänmenonavan $\overline{IN4}$ tullessa matalaksi, kun ajastimen aika päättyy. Ajastimeen 190 sisältyy laskin 192, esim. RCA:n CD4024 binäärilaskin, kaksoissisäänmenoinen NAND-veräjä 194 ja invertterit 196, 198 ja 200. Napa $\overline{OUT4}$ on kytketty laskimen 192 palautussisäänmenoon RES invertte-rin 196 kautta. Sisäänmenonapa CL on kytketty NAND-veräjän 194 yhteen sisäänmenoon ja NAND-veräjän 194 ulostulo on kytketty laskimen 192 kellosisäänmenoon CLOCK invertterin 200 kautta. Laskimen 192 ulostulo Q on kytketty ulostulonapaan $\overline{IN4}$ invertterin 198, invertterin 202 ja kääntävän puskurin 204 kautta. Laskimen 192 ulostulo Q on myös kytket-ty NAND-veräjän 194 jäljellä olevaan sisäänmenoon invertterin 198 kautta. Sisäänmenonapa CL on kytketty vastaanottamaan ajastussignaa-li CL ajastustoiminnosta 78. Signaalin CL taajuus ja laskimen ulos-tulo on valittu siten, että ulostulo tulee korkeaksi halutun aikavä-lin lopussa.

Kun järjestelmän ohjauslaite haluaa käynnistää jonkin ajastuksen, se muodostaa matalan signaalin ulostulonapaan $\overline{OUT4}$, mikä palauttaa ajas-timen nolnaan ja invertterin 198 korkea ulostulo avaa NAND-veräjän

194. Täten NAND-veräjä 194 johtaa kellopulsseja laskimen 192 kellosisäänmenoon invertterin 200 kautta. Laskin 192 etenee yhden laskennan kunkin sisääntulopulssin negatiiviseksi menevällä muutoksella. Kun laskin 192 on aktiivinen, on laskimen 192 valittu ulostulonapa matala ja napa $\overline{IN4}$ on korkea. Kun laskin 190 saavuttaa halutun laskuaseman, tulee laskimen ulostulo korkeaksi ja napa $\overline{IN4}$ tulee matalaksi. Kun tämä valittu laskuasema on saavutettu, tulee invertterin 198 ulostulo matalaksi, mikä blokeeraa NAND-veräjän 194 läpäisemästä enää pulsseja CL-sisäänmenonavasta laskimen 192 kellosisäänmenoon, kunnes järjestelmän ohjauslaitteet palauttavat (nollaavat) laskimen tekemällä navat $\overline{OUT4}$ matalaksi.

Kuvio 8

Kaikki kuviossa 1 esitetyt korikohtaiset liitännät 28, 34, 40 ja 46 ovat rakenteeltaan samanlaiset ja tästä syystä selitetään vain yksi korikohtainen liitäntä 28 korille 0. Liitännän 28 kuvauksen helpottamiseksi se on jaettu liitäntätoimintoon, joka on esitetty kuviossa 8A ja joka käsittelee informaatiovuota järjestelmän valvontalaitteista 22 kerrostasovalitsimeen 26 ja kuviossa 8B esitettyyn liitäntätoimintoon, joka käsittelee informaatiovuota kerrostasovalitsimesta 26 järjestelmän valvontalaitteisiin.

Liitäntätoiminto järjestelmän valvontalaitteista kunkin korin kerrostasovalitsimeen on kriittisesti tärkeä, kun käytetään mikroprosessoria, koska kerrostasovalitsin toimii synkronisella eli jatkuvalla tavalla, so. tarvitsee jatkuvia ohjaussignaaleja järjestelmän valvontalaitteista, samalla kun mikroprosessori toimii asynkronisella eli panoksittaisella tavalla rajoitetuin muistikapasiteetin ja toimintanopeuksin. Mikroprosessori valmistaa datasanat kullekin hissikorille, lähettää ne eri korinohjauslaitteisiin ja sitten menee suorittamaan muita tehtäviä, kuten lukemaan tilasignaaleja eri koreista ja valmistelemaan uusia käskysanoja koreille perustuen viimeisimpään koreista vastaanotettuun informaatioon.

Kerrostasovalitsin 26 toimii sarjamuotoisesti, synkronoituna pyyhkäisyajaväleillä, jotka on tuotettu ajastussignaaleilla S0S - S3S, ja käskyjen järjestelmän valvontalaitteista 22 täytyy esiintyä oikeina aikaväleinä joka kerran kun jatkuvasti laskeva laskin S0S - S3S laskee läpi pyyhkäisyjakson pyyhkäisyajavälien. Epäonnistuminen tuottaa käsky- tai kieltosignaali järjestelmän valvontalaitteista 22 pyyhkäi-

syaikavälin aikana aikaansaa sen, että järjestelmän valvontalaitteet menettävät ohjausmahdollisuutensa ja kukin kori automaattisesti toimii sen yksilöllisen korinohjauslaitteen strategian mukaisesti. Korinohjausstrategiana on vastata kaikkiin kutsuihin, jotka ovat peräisin edestäpäin sen kulkusuunnassa ja kun edessäpäin ei ole enää kutsuja, se vastaa kaikkiin kutsuihin vastakkaisessa suunnassa, kunnes tässä suunnassa ei enää ole kutsuja. Kerrostasokutsu joutilaan korin ylä- tai alapuolella panee korin liikkeeseen oikeaan suuntaan kutsuun vastaamiseksi.

Kuviossa 8A esitetty liitäntä 28 ratkaisee liitäntäpulman järjestelmän valvontalaitteiden 22 ja korikutsuohjauslaitteiden välillä varastoimalla sarjamuotoisen datasanan, joka on vastaanotettu järjestelmän valvontalaitteista ja toistuvasti ja sarjamuotoisesti lukemalla varastoidun datasanan siihen liittyviin korinohjausvälineisiin. Sarjamuotoinen datasana varastoidaan jatkuvasti pyyhkäistyyn sarjasaantimuistiin, joka lukee ja uudelleenkierrättää dataa, kunnes uusi datasana on vastaanotettu. Kun uusi datasana on vastaanotettu, sarjasaantimuistin toimintamuoto muutetaan uudelleenkierrättämisluodosta päästämään uusi datasana sisään muistiin. Tämä toteutetaan ilman erillistä lukujohtoa järjestelmän valvontalaitteista ja keskeyttämättä sarjaajoitettua käskyvuotoa sarjasaantimuistista kerrostasovalitsimeen. Kun uusi datasana on täydellisesti sarjasaantimuistissa, muistin toimintamuoto automaattisesti palaa takaisin uudelleenkierrätysmuotoon tämän uuden sanan säilyttämiseksi, kunnes seuraava datasana on vastaanotettu.

Sarjasaantimuistilla voi olla siirtorekisterin 210 muoto, esim. RCA:n CD4031, jossa on datasisäänmeno D, kellosisäänmeno CL, toimintatapas sisäänmeno MODE, uudelleenkierrätys sisäänmeno REC ja ulostulo Q. Sarjakäskysana $\overline{COMO}$, joka määrätyn aikavälein lähetetään järjestelmän ohjauslaitteesta 22, johdetaan sisäänmenonapaa $\overline{COMO}$, ja sisäänmenonapa $\overline{COMO}$ on kytketty siirtorekisterin 210 datasisäänmenoon D. Ajastussignaali $\overline{KO8}$ on kytketty siirtorekisterin 210 kellosisäänmenoon CL invertterin 212 kautta. Kuten kuviossa 3 on esitetty, signaalilla $\overline{KO8}$, käännettynä, on positiiviseksi menevä muutos pyyhkäisy-aikavälin kunkin neljänneksen alussa. Kun siirtorekisterin 210 sisäänmeno MODE on matala, tulee looginen taso datasisäänmenossa D siirretyksi siirtorekisterin ensimmäiseen portaaseen jokaisella signaalin $\overline{KO8}$ positiiviseksi tulevalle muutoksella. Täten siirtorekisteri 210

on ajastettu neljä kertaa kunkin pyyhkäisyaikavälin kuluessa mikä tekee mahdolliseksi 4 bitin informaation sisältymisen jokaiseen pyyhkäisyaikaväliin, jotka ovat samanlaisia kuin kuviossa 7 esitetty sarjasisäänmenosignaali kustakin korista järjestelmän valvontalaitteisiin. Kuvio 8C kuvaa sarjasignaalien muotoa järjestelmän valvontalaitteista 22 kuhunkin hissikoriin. Käskeysanat, jotka eivät liity kerrokseen, sisältyvät kunkin pyyhkäisyaikavälin ensimmäiseen neljännekseen. Esimerkiksi käskeysignaali $\overline{SUT}$, joka vaatii, että korin kerrosvalitsin on asetettava ylöspäin ajoasentoon, voidaan lähettää pyyhkäisyaikavälinä 00; käskey $\overline{SDT}$, joka vaatii, että korin kerrosvalitsin asetetaan alaspäin ajoasentoon, voidaan lähettää pyyhkäisyaikavälinä 01; käskey $\overline{DOPN}$, joka vaatii koria avaamaan ovensa, voidaan lähettää pyyhkäisyaikavälinä 02; ja käskey $\overline{NEXT}$, joka ilmoittaa korille, että se on seuraava kori, joka poistuu pääkerroksesta, voidaan lähettää pyyhkäisyaikavälinä 03.

Kerrokseen liittyvät signaalit $\overline{PKFL}$, $\overline{UPIN}$ ja $\overline{DNIN}$, voidaan lähettää minkä aikavälin kuluessa tahansa, joka liittyy kerrokseen, jota hissi-kori pystyy palvelemaan. Tosi signaali $\overline{PKFL}$ lähetetään koriin, kun järjestelmän ohjaus 22 antaa korille käskeyn parkeerata tiettyyn kerrokseen, signaalin esiintyessä sen pyyhkäisyaikavälin toisessa neljänneksessä, joka liittyy kerrokseen, johon korin on parkeerattava. Tosi signaali $\overline{UPIN}$ lähetetään korin kerrosvalitsimeen niitä kerroksia varten, joita kori pystyy palvelemaan suunnassa ylöspäin, mutta jotka järjestelmäohjaus haluaa blokeerata kerrostasokutsuilta ylöspäin, jotka on rekisteröity siitä jäämään huomiotta korilta. Samalla tavoin tosi signaali $\overline{DNIN}$ lähetetään koriin niitä kerroksia varten, joita kori pystyy palvelemaan alaspäin, mutta jotka järjestelmäohjaus haluaa blokeerata kerrostasokutsuilta alaspäin, jotka on rekisteröity siitä jäämään huomiotta korilta. Täten esimerkiksi alaspäin kutsun kerroksesta 6 osoittamiseksi korille 0 järjestelmän valvontalaite 22 lähettää tosi $\overline{DNIN}$ signaalit koreihin 1, 2 ja 3 sen pyyhkäisyaikavälin 05 neljännessä neljänneksessä joka liittyy kerrosasemaan 6.

Kun siirtorekisterin 210 sisäänmeno MODE on korkea, uudelleenkiertosisäänmeno REC on kykenevä ja ulostulo Q ajastetaan takaisin siirtorekisteriin 210.

Järjestelmän ohjauslaite 22 voisi suoraan ohjata siirtorekisterin 210 MODE-sisäänmenoa, mutta tämä vaatisi toisen johdon järjestelmän valvontalaitteesta 22 kuhunkin koriin sekä lisäyksen strategiaohjelmaan,

mikä lisääisi muistin tarvetta, jolla on rajoitettu kapasiteetti. Edelleen, tämä järjestelmäprosessori 22 on suhteellisen hidas ja vaatimus sarja-datasanan lähettämisestä yhdessä erillisen signaalin kanssa, jonka täytyy täsmällisesti ladata sana dynaamiseen muistiin keskeyttämättä muistin sarjaulostuloa ja menettämättä yhtään lähetettyä bittiä, voi olla liian ankara.

Kuvion 8A järjestely ratkaisee pulman datasanan täsmälleen lataamisesta järjestelmän ohjauslaitteesta 22 siirtorekisteriin 210 käyttäen vain yhtä johtoa järjestelmän valvontalaitteesta 22 kuhunkin koriin ja ilman vaaraa databittien menettämisestä käyttäen datasanan muotoa ohjaamaan toimintanavan ohjausfunktiota.

Tarkemmin sanottuna toimintatavan ohjausfunktio toteutetaan ensimmäisellä ja toisella flip-flopilla 214 ja 216, kuten esimerkiksi RCA:n CD4027, kaksoissisäänmenoisella NAND-veräjällä 218 ja inverttereillä 220, 222 ja 224. Sisäänmenonapa $\overline{COMO}$ on kytketty invertterin 220 kautta ensimmäisen J-K-flip-flopin 214 asettelusisäänmenoon. Flip-flopin 214 J, C ja K sisäänmenot on kytketty yksisuuntaiseen potentiaalilähteeseen navassa 226. Kuviossa 3 esitetty ajastussignaali $\overline{S300}$, joka on tosi vain osalla pyyhkäisyajavälin 00 kolmatta neljänestä, on invertterin 222 kautta kytketty NAND-veräjän 218 sisäänmenoon. NAND-veräjän 218 ulostulo on kytketty invertterin 224 kautta flip-flopin 214 palautussisäänmenoon.

Toisen J-K-flip-flopin 216 asettelusisäänmeno on kytketty maahan. Flip-flopin 216 J ja K sisäänmenot on navassa 228 kytketty yksisuuntaiseen potentiaalilähteeseen. Kellosisäänmeno on kytketty vastaanottamaan ajastussignaali $\overline{S100}$, joka on tosi ainoastaan osalla pyyhkäisyajavälin 00 ensimmäistä neljänestä. Flip-flopin 216 palautussisäänmeno on kytketty flip-flopin 214 Q-ulostuloon. Flip-flopin 216 Q-ulostulo on kytketty NAND-veräjän jäljellä olevaan sisäänmenoon ja myös siirtorekisterin 210 MODE-sisäänmenoon.

Toimintatapaohjauksen toiminnassa oletetaan, että flip-flopin 216 Q-ulostulo on loogisella ykköstasolla, mikä sijoittaa rekisterin 210 uudelleenkiertotoimintatapaan. Flip-flopin 216 korkea Q-ulostulo tekee NAND-veräjän 218 kykeneväksi ja ajastussignaali $\overline{S300}$ palauttaa flip-flopin 214. Täten flip-flopin 214 Q-ulostulo tulee pidetyksi korkeana, mikä vuorostaan varmistaa sen, että flip-flopin 216 Q-ulostulo jää

korkeaksi pitämään siirtorekisteriä 210 uudelleenkiertotoimintatavassa.

Kun järjestelmän valvontalaite 22 haluaa lähettää datasanan koreihin, se toteaa ajastussignaalin $\overline{MXCT}$ (katso kuvio 2) negatiiviseksi käyvän muutoksen ja lähettää edeltävän nollan jokaiseen koriin signaalin $\overline{MXCT}$ aikana, mitä seuraa datasana. Edeltävä nolla datavirtauksessa käännetään loogiseksi ykköseksi invertterillä 220, mikä asettelee flip-flopin 214 tuottamaan matalan Q-ulostulon. Flip-flopilla 216 on nyt matala signaali sen palautussisäänmenossa, mikä avaa tämän flip-flopin lukituksen. Ajastussignaali $\overline{S100}$, joka esiintyy pyyhkäisyajavälin 00 ensimmäisen neljänneksen keskiosassa, liipaisee flip-flopin 216 sen vastakkaiseen tilaan ja täten sen Q-ulostulo tulee matalaksi tehden siirtorekisterin 210 datasisäänmenon D kykeneväksi ja blokeeraten NAND-veräjän 218 päästämästä ajastussignaalia $\overline{S300}$. Kello K08 täten ajastaa neljä databittiä kuhunkin 16 pyyhkäisyajavälin 64 portaisessa siirtorekisterissä 210. Seuraavan ajastussignaalin $\overline{S100}$ positiiviseksi menevällä muutoksella flip-flop 216 tulee liipaistuksi sen vastakkaiseen tilaan, mikä tekee sen Q-ulostulon korkeaksi. Tämä korkea ulostulo tekee kykeneväksi siirtorekisterin 210 uudelleenkiertotoimintatavan ennen pyyhkäisyajavälin 00 ensimmäisen neljänneksen päättymistä uuden datasanan kierrättämiseksi kunnes seuraava datasana vastaanotetaan. Flip-flopin 216 korkea Q-ulostulo myös tekee NAND-veräjän 218 kykeneväksi, niin että ajastussignaali $\overline{S300}$ palauttaa flip-flopin 214 johtaen loogisen ykkösen flip-flopin 216 palautussisäänmenoon estämään seuraavia ajastussignaaleja $\overline{S100}$ liipaisemasta flip-floppia 216 kunnes seuraava käskydatasana vastaanotetaan järjestelmän valvontalaitteesta 22.

Tarkoituksella poistaa kukin kori järjestelmäprosessorin viimeisen käskysanan rajattomasta ohjauksesta, jos järjestelmäprosessori epäonnistuu muiden signaalien tuottamisessa ja jokaisen korin poistamiseksi viallisen järjestelmäprosessorin ohjauksesta, joka ei tuota käskysanoja ennaltamäärättynä aikavälinä, käskysanoja tarkkaillaan ajastusvälineillä 230. Ajastusvälineisiin 230 voi kuulua multivibraattori 232, esim. PCA:n CD4047A, joka on siten kytketty, että sen Q-ulostulo pysyy korkeana niin kauan kun sisääntulopulssijakso on lyhyempi kuin ajastusjakso, jonka määräävät ajastusvälineiden 234 RC-komponentit.

Sisääntulopulssi, joka liipaisee multivibraattorin 232 ja uudelleen-

liipaisee sen pitääkseen CL-ulostulon korkeana, on sarjasignaalin $\overline{\text{COMO}}$ edessä oleva nolla. Sisääntulonapa $\overline{\text{COMO}}$ on kytketty invertterin 236 ja vaihtovirtakytkineliimen 238 kautta multivibraattorin 232 liipaisu- ja uudelleenliipaisusisäänmenoihin TRIG ja RETRIG. Vaihtovirtakytkineliin estää sen, että multivibraattorin 232 sisääntulo juuttuisi korkeaan tilaan. Multivibraattorin 232 korkea Q-ulostulo mahdollistaa käskysignaalien lähettämisen järjestelmän valvontalaitteesta 22 hissi-koreihin. Jos multivibraattori ajastaa loppuun ennen seuraavan käskysanan vastaanottamista, tulee Q-ulostulo matalaksi ja estää kaikki signaalit järjestelmän valvontalaitteesta 22 koreihin ja silloin korit toimivat riippumattomasti niiden yksilöllisen korinohjausstrategian mukaan, jolloin jokainen kori pystyy vastaamaan kaikkiin kerrostasokutsuihin.

Sarjakäskysana järjestelmän valvontalaitteesta 22, joka esiintyy siirtorekisterin 210 Q-ulostulossa, on kytketty yhteen sisäänmenoon kolmisisäänmenoisessa NAND-veräjässä 240, invertterin 244 kautta yhteen sisäänmenoon kolmisisäänmenoisessa NAND-veräjässä 242 ja D-sisäänmenoihin D-tyyppisissä flip-flopeissa 246, 248, 250, 252 ja 254, esimerkiksi RCA:n tyyppiä CD4013. Flip-flopin 246 Q-ulostulo on kytketty yhteen sisäänmenoon kolmisisäänmenoisessa NAND-veräjässä 256. Flip-flopien 248, 250, 252 ja 254 Q-ulostulot on kytketty sisäänmenoihin kaksisisäänmenoisissa NAND-veräjissä 258, 260, 262 ja 264. Multivibraattorin 232 Q-ulostulo on kytketty yhteen sisäänmenoon kussakin seuraavassa NAND-veräjässä 240, 242, 256, 258, 260, 262 ja 264, tehden nämä veräjät kykeneviksi niin kauan kun järjestelmän valvontalaite 22 toimii ajastetulla tavalla.

Aikavalintapulssi STA, joka on esitetty kunkin pyyhkäisyaikavälin toisen neljänneksen aikana, on kytketty NAND-veräjän 242 jäljellä olevaan sisäänmenoon. NAND-veräjän 242 ulostulo on kytketty ulostulonapaan $\overline{\text{PKFL}}$, joka tuottaa tosisignaalin $\overline{\text{PKFL}}$ niiden pyyhkäisyaikavälien toisessa neljänneksessä, jotka sisältävät tosi parkeerauskäskyn järjestelmän valvontalaitteesta 22.

Osoitukset järjestelmälaitteesta 22 ovat matala/tosia sarjasignaaleissa $\overline{\text{COMO}}$ ja koska kielto-signaalit $\overline{\text{UPIN}}$ ja $\overline{\text{DNIN}}$ ovat matalia, kun koria on estetty vastaamaan kerrostasokutsuun asianomaisessa kerroksessa, järjestelmälaite 22 tekee kerrosten määräykset korille korkeilla signaaleilla $\overline{\text{UPIN}}$ ja $\overline{\text{DNIN}}$. Täten matalan tosi määräyssignaali-

lin $\overline{COMO}$:ssa täytyy muodostaa korkea $\overline{UPIN}$ tai $\overline{DNIN}$ signaali.

Tarkemmin sanottuna pulssi STB, joka esiintyy kunkin pyyhkäisyaikavälin viimeisen neljänneksen aikana, on kytketty NAND-veräjän 240 jäljellä olevaan sisäänmenoon. NAND-veräjän 240 ulostulo on kytketty ulostulonapaan $\overline{DNIN}$ muodostaen korkean alas-estosignaalin $\overline{DNIN}$ niiden pyyhkäisyaikavälien viimeisessä neljänneksessä, jotka sisältävät tosi (matalan) määräyssignaalin järjestelmän valvontalaitteesta 22. Kerroksilla, joita ei ole osoitettu korille, on korkea signaali niihin liittyvien pyyhkäisyaikavälien viimeisessä neljänneksessä, mikä muodostaa tosi-estosignaalit $\overline{DNIN}$ näille kerroksille.

Ajastussignaali $\overline{K08}$ ja pulssi STD on kytketty kaksisisäänmenoisen NAND-veräjän 266 kahteen sisäänmenoon ja NAND-veräjän 266 ulostulo on kytketty flip-flopin 246 kellosisäänmenoon C. NAND-veräjän 266 ulostulo on matala kunkin pyyhkäisyaikavälin kolmannen neljänneksen ensimmäisellä osalla ajastaen dataa, joka esiintyy D-sisäänmenossa positiiviseksi menevän muutoksen aikana kellopulssissa, Q-ulostuloon. Jos ylöspäin palvelusuunta pyyhkäisyaikaväliin liittyvästä kerroksesta ei ole osoitettu korille liittyen signaaliin $\overline{COMO}$, on flip-flopin 246 D sisäänmeno korkea pyyhkäisyaikavälin kolmannen neljänneksen aikana ja flip-flopin 246 Q-ulostulo tulee korkeaksi. Jos ylöspäin palvelusuunta kerroksesta, joka liittyy pyyhkäisyaikaväliin, on osoitettu tälle korille, on flip-flopin 246 D-sisäänmeno matala kolmannen neljänneksen aikana ja flip-flopin 246 Q-ulostulo on matala. Flip-floppia 246 käytetään varastoimaan kielto tai määräys, niin että se voidaan esittää kerrosvalitsimelle 26 samanaikaisesti alaspäin kiellon tai määräyksen kanssa kullakin pyyhkäisyaikavälillä. Koska alaspäin määräys aikavalittiin pulssilla STB, NAND-veräjä 256, joka muodostaa ylöspäin määräyksen (osoituksen), on myös aikavalittu pulssilla STB. Jos pyyhkäisyaikaväliin liittyvää kerrosta ei ole osoitettu tälle korille, on flip-flopin 246 Q-ulostulo korkea, kun aikavälin neljäs neljännes alkaa, tehdäkseen NAND-veräjän 256 ulostulon matalaksi ja muodostaakseen tosi ylöspäin estosignaalin $\overline{UPIN}$ asianomaisen aikavälin neljänteen neljänneksen. Jos asianomaiseen aikaväliin liittyvä kerros on osoitettu tälle korille, flip-flopin 246 Q-ulostulo on matala asianomaisen aikavälin neljännen neljänneksen alussa ja NAND-veräjän 256 ulostulo tulee korkeaksi tehden korin kykeneväksi palvelemaan kerrostasokutsua ylöspäin tässä kerroksessa.

Jäljellä olevat käskyt $\overline{SUT}$, $\overline{SDT}$, $\overline{DOPN}$ ja $\overline{NEXT}$ järjestelmän valvontalaitteesta 22 eivät ole kerrokseen liittyviä ja ne sijoitetaan pyyhkäisyajavälien 00, 01, 02 ja 03 ensimmäiseen neljännekseen. Lisäksi, kun yksi näistä käskyistä on vastaanotettu, sen täytyy pysyä siihen saakka, kunnes käsky jälleen vastaanotetaan täyttä jaksoa myöhemmin eikä vain juuri sen pyyhkäisyajavälin ajan, jona käsky lähetettiin.

Ensimmäinen käsky $\overline{SUT}$, joka vaatii, että kerrostasovalitsin 26 on asetettava ylöspäin liikettä varten, poimitaan aikavälin 00 ensimmäisestä neljänneksestä flip-flopilla 248 ja ajastusvälineillä, joihin sisältyy kolmisisäänmenoinen NAND-veräjä 270, invertteri 272 ja kaksisisäänmenoinen NAND-veräjä 274. Ajastussignaali K08S ja SECO ja pulssi STC johdetaan NAND-veräjän 270 kolmeen sisäänmenoon. Nämä signaalit ovat kaikki korkeita aikavälien 00-07 keskiosien aikana tehden NAND-veräjän 270 ulostulon matalaksi tänä aikana, mikä matala signaali muutetaan korkeaksi signaaliksi invertterillä 272. Aikaväli 00 poimitaan aikaväleista 00-07 ajastussignaali DECO, joka on korkea vain kunkin pyyhkäisyjakson aikavälien 00 ja 08 aikana. Invertterin 272 ulostulo ja ajastussignaali DECO johdetaan NAND-veräjän 274 kahteen sisäänmenoon ja tämän veräjän ulostulo on kytketty flip-flopin 248 kellosisäänmenoon C. Jos järjestelmän valvontalaite 22 on vaati- massa, että kerrosvalitsin on asetettava ylöspäinliikkeeseen, on aikavälin 00 ensimmäinen neljännes matala, ja täten flip-flopin 248 D-sisäänmeno on matala tänä aikana. Tämä matala signaali ajastetaan ulostuloon Q. Täten ulostulo $\overline{Q}$, joka on kytketty yhteen sisäänmenoon NAND-veräjässä 258, tulee korkeaksi aiheuttaen NAND-veräjän 258 ulostulon tulon matalaksi, mikä aikaansaa tosi signaalin $\overline{SUT}$. Koska flip-flopia 248 ei ajasteta jälleen kunnes esiintyy seuraava aikaväli 00, pysyy ulostulonavassa $\overline{SUT}$ esiintyvä signaali, kunnes jälleen tutkitaan aikavälin 00 ensimmäisen neljänneksen looginen taso.

Järjestelmäohjauksesta 22 lähtevä käsky $\overline{SDT}$, joka vaatii, että kerrosvalitsin 26 on asetettava alaspäinliikenteeseen, poimitaan aikavälin 01 ensimmäisestä neljänneksestä kaksisisäänmeno- isella NAND-veräjällä 276, jonka yksi sisäänmeno on kytketty invertterin 272 ulostuloon ja yksi sisäänmeno ajastussignaaliin DECl. NAND-veräjän 276 ulostulo on kytketty kellosisäänmenoon C flip-flopissa 250, jonka $\overline{Q}$ -ulostulo on kytketty sisäänmenoon NAND-veräjässä 260, jonka ulostulo muodostaa käskyn $\overline{SDT}$.

Järjestelmäohjauksesta 22 lähtevä käsky $\overline{DOPN}$, joka vaatii, että korin ovet avataan, poimitaan aikavälin 02 ensimmäisestä neljänneksestä kaksisisäänmenoisella käskyveräjällä 278, jonka yksi sisäänmeno on kytketty invertterin 272 ulostuloon ja yksi sisäänmeno ajastussignaaliin DEC2. NAND-veräjän 278 ulostulo on kytketty kellosisäänmenoon C flip-flopissa 252, jonka $\overline{Q}$ -ulostulo on kytketty sisäänmenoon NAND-veräjässä 262, jonka ulostulo muodostaa käskyn $\overline{DOPN}$.

Järjestelmäohjauksesta lähtevä käsky $\overline{NEXT}$, joka osoittaa korin seuraavaksi koriksi poistumaan pääkerroksesta, poimitaan aikavälin 03 ensimmäisestä neljänneksestä kaksisisäänmenoisella NAND-veräjällä 280, jonka yksi sisäänmeno on kytketty invertteriin 272 ja toinen sisäänmeno ajastussignaaliin DEC3. NAND-veräjän 280 ulostulo on kytketty kellosisäänmenoon C flip-flopissa 254, jonka ulostulo $\overline{Q}$ on kytketty sisäänmenoon NAND-veräjässä 264, jonka ulostulo muodostaa signaalin $\overline{NEXT}$.

Se osa liitäntää 28, joka liittyy informaatiovuohon kerrosvalitsimesta 26 järjestelmäohjaukseen 22, on esitetty kuviossa 8B. Kerrosvalitsin 26 muodostaa tilasignaaleja AVP0-AVP3, INSC, BYPS, UPTR, $\overline{AVAS}$, WT50, $\overline{D89T}$, $\overline{MDCL}$ ja $\overline{CALL}$, jotka, kun ne vastaanotetaan järjestelmäohjauksessa, varastoidaan kuviossa 5 esitettyyn RAM0:llaan. Ylös ja alas valtuuttavat kerrossignaalit $\overline{MT00}$ ja $\overline{MT01}$ ja korikutsut $\overline{3Z}$ varastoidaan RAM2:een ja RAM3:een, jotka on esitetty kuviossa 5. Signaalit AVP0-AVP3 muodostavat sen kerroksen binääriosoitteen, jolla paikallaan oleva kori seisoo ja kun kori liikkuu, se muodostaa binääriosoitteen lähimmästä kerroksesta, jolla kori voisi suorittaa normaalin pysähtymisen. Signaali INSC on tosi, kun kori on palveluksessa järjestelmäohjauksella 22. Signaali BYPS on tosi, kun kori on aseteltu ohittamaan kerrostaokutsut. Esimerkiksi kun alaspäin kulkeva kori tulee kuormatuksi, se ohittaa kerrostaokutsut tiellään pääkerrokseen. Myös kun kori pääkerroksessa tulee kuormatuksi, se ohittaa kerrostaokutsut ylöspäin. Molemmissa tilanteissa kori antaa ulos tosi BYPS-signaalin. Signaali UPTR on korkea eli tosi, kun kori on aseteltu liikkeeseen ylöspäin ja matala, kun kori on aseteltu liikkeeseen alaspäin. Signaali $\overline{AVAS}$ on tosi, kun kori on palveluksessa, se on vastannut kaikkiin kutsuihinsa ja seisoo kerroksessa ovet suljettuina. Täten seuraavaa ($\overline{NEXT}$) koria, joka normaalisti seisoo pääkerroksessa ovet avoinna ja kerrostaolamppu sytytettynä, ei pidetä AVAS-korina. Signaali WT50 on tosi, kun hissikorin kuorma ylittää

50 % sille lasketusta kuormasta. Signaali $\overline{D89T}$ on tosi, kun moottorigeneraattoriyksikkö, joka tuottaa sähkötehon hissin käyttömoottorille, on katkaistuna. Signaali $\overline{MDCL}$ on tosi, kun korin ovet ovat kiinni ja signaali $\overline{CALL}$ on tosi, kun hissikori on rekisteröinyt korikutsun.

Järjestelmäohjaus 22 voidaan soveltaa mihin tahansa rakennukseen ilman että on tarpeen erityisesti räätälintyöllä sovittaa järjestelmäohjausta rakennuksen muotoon taikka alunperin suunnitella sitä tietäen, mitkä korit pystyvät palvelemaan eri kerroksia. Kaikki tämä informaatio annetaan järjestelmäohjaukselle 22 signaalien muodossa koriohjauksesta 14. Tämä on tärkeä piirre, joka merkittävästi lisää järjestelmäohjauksen 22 universaalisuutta. Signaalit $\overline{MT00}$ ja $\overline{MT01}$ ovat sarjasignaaleja, jotka voidaan muodostaa pysyväismuistiuralla koriohjauksessa 14 ja ne ovat tosia aikaväleissä, jotka liittyvät kerroksiin, joilla kori pystyy palvelemaan kutsuja ylöspäin ja vastaavasti alaspäin. Signaali $\overline{3Z}$ on sarjamuotoinen, kerrokseen liittyvä signaali, joka on tosi niiden aikavälien aikana, jotka liittyvät kerroksiin, joita varten korilla on rekisteröity korikutsu.

Multipleksoreita 290 ja 292, esimerkiksi RCA:n CD4051A ja nelikytkintä 294 esimerkiksi RCA:n CD4016D veräjänä käytettynä, käytetään muodostamaan sarjasignaali $\overline{DAT0}$. Multipleksoreita 290 ja 292 käytetään sijoittamaan ei-sarjasignaalit aikaväleihin kun taas neli-bilateraalikytkin 294 multipleksoi multipleksoreiden 290 ja 292 sarjamuotoiset ulostulot ja sarjamuotoisen kerrostasovaltuutussignaalin $\overline{FEN}$ ja korikutsusignaalin $\overline{3Z}$ kanssa.

Multipleksori 290 on kykenevä 16 aikavälijakson ensimmäiselle 8 aikavälille kytkemällä ajastussignaalin $\overline{SECO}$ estosisäänmenoon invertterin 296 kautta ja multipleksori 292 on kykenevä pyyhkäisyjakson viimeisille 8 aikavälille kytkemällä ajastussignaalin $\overline{SEC1}$ multipleksorin 292 estosisäänmenoon invertterin 298 kautta. Multipleksoreiden 290 ja 292 datasisäänmenot $D0-D7$ on kytketty vastaanottamaan multipleksoitavat signaalit ja niiden ohjaussisäänmenot $C1, C2$ ja $C3$ on kytketty ajastussignaaleihin $S0S, S1S$ ja $S2S$. Kun ohjaussisäänmenoihin johdettu binääriosoitte muuttuu, on eri datasisäänmeno kytketty ulostuloon OUT . Kuten kuviossa 8B on esitetty, edenneen korin asemasignaalit $\overline{AVP0-AVP3}$, palveluksessaolosignaali $\overline{INSC}$ ja ohitussignaali $\overline{BYP5}$ on kytketty datasisäänmenoihin multipleksorissa 290 ja liikesuuntasignaali $\overline{UPTR}$, käytettävissäolosignaali $\overline{AVAS}$, korikuormasignaali $\overline{WT50}$, moottorigeneraattorin katkaistunaolosignaali $\overline{D89T}$, ovisignaali $\overline{MDCL}$

ja korikutsusignaali $\overline{CALL}$ on kytketty datasisäänmenoihin multipleksorissa 292. Multipleksorin 290 ulostulo on kytketty neli-bilateraalikytkimen 294 kytkimeen A liittyvään sisäänmenoon ja multipleksorin 292 ulostulo on kytketty neli-bilateraalikytkimen 294 kytkimeen B liittyvään sisäänmenoon. Kytkimien A ja B ohjaussisäänmenot on kytketty pulssiin STC tilasignaalien sijoittamiseksi asianomaisten pyyhkäisyaikavälien ensimmäiseen neljännekseen, kuten on osoitettu kuviossa 7.

Kerrostasovaltuutussignaalit $\overline{MT00}$ ja $\overline{MT01}$ on yhdistetty muodostamaan pääkerrosvaltuutussignaali $\overline{FEN}$, joka myös ottaa huomioon, onko kori palveluksessa (INSC) järjestelmäohjauksella 22 ja onko vaiko ei kori ohittamassa kerrostasokutsuja (EYPS). Nämä signaalit yhdistetään NAND-veräjillä 296, 298 ja 300 ja inverttereillä 302 ja 304. Signaalit $\overline{MT00}$ ja $\overline{MT01}$ on kytketty NAND-veräjän 296 sisäänmenoihin, joka veräjä on kaksisisäänmenoinen veräjä ja jonka ulostulo on kytketty sisäänmenoon NAND-veräjässä 298, joka myös on kaksisisäänmenoinen NAND-veräjä. NAND-veräjän 298 ulostulo, joka muodostaa signaalin $\overline{FEN}$, on kytketty neli-bilateraalikytkimen 294 sisäänmenoon kytkintä C varten. Signaali INSC on kytketty sisäänmenoon NAND-veräjässä 300, joka on kaksisisäänmenoinen NAND-veräjä ja signaali BYPS on kytketty invertterin 302 kautta NAND-veräjän 300 toiseen sisäänmenoon. NAND-veräjän 300 ulostulo on kytketty invertterin 304 kautta NAND-veräjän 298 toiseen sisäänmenoon. NAND-veräjän 298 ulostulo on matala, jos kori on palveluksessaoleva kori, joka ei ole ohittamassa kerrostasokutsuja ja on valtuutettu palvelemaan kerrostasokutsuja ainakin yhdessä palvelusuunnassa kerroksesta, joka liittyy kysymyksessä olevaan aikaväliin. Pulssi STD on kytketty kytkimen C ohjaussisäänmenoon aiheuttaen pääkerrosvaltuutussignaalin $\overline{FEN}$ sijoittamisen kunkin aikavälin kolmanteen neljännekseen.

Sarjamuotoiset korikutsut $\overline{3Z}$ on kytketty neli-bilateraalikytkimen 294 kytkimen D datasisäänmenoon invertterin 306 kautta ja kytkimen D ohjaussisäänmeno on kytketty pulssiin STB, joka sijoittaa korikutsut aikavälien neljänteen neljännekseen.

Neli-bilateraalikytkimen 294 kytkimien A, B, C ja D ulostulot on kytketty yhteen ja yhteinen ulostulo on kytketty vastuksen 309 kautta tasapotentiaaalilähteen napaan 308. Yhteinen ulostulo on kytketty myös NPN-transistorin 310 kantaan ei-kääntävän puskurin 312 (esimer-

kiksi RCA:n CD4050AD) ja vastuksen 314 kautta. Transistorin 310 kanta on myös kytketty maahan vastuksen 316 kautta. Transistorin 310 emitteri on kytketty maahan ja sen kollektori on kytketty ulostulonapaan DAT0, joka tuottaa sarjamuotoisen datasiignaalin prosessoriliitäntää 72 varten.

Kuvio 9

Kuvio 9 on lohkokaavio, joka yleispiirteisesti esittää uutta ja parannettua ryhmävalvontastrategiaa hissikorijoukon ohjaamiseksi vastaamaan hissipalvelukutsuihin esillä olevan keksinnön mukaisesti. Kuviossa 9 esitetty järjestelmä kuvaa ohjelmaa keksinnön mukaisen strategian toteuttamiseksi, kukin kuviossa esitetyistä lohkoista on täysin esitetty kuvioden 11-23 mukaisissa vuokaavioissa. Kuvioden 11-23 vuokaaviot ovat ohjelmoitsijan vuokaavioita, jotka yhdessä muiden kuvioden, selityksen ja käyttäjän mikroprosessorikäsi­kirjan kanssa antavat riittävästi yksityiskohtia normaalin taitavuuden omaavalle ohjelmoitsijalle tarpeellisten käskyjen kirjoittamiselle mikroprosessorin ohjelmoimiseksi. Kuvion 9 lohkoihin sisältyy myös LCD identifikaatio­numero, joka viittaa niihin alaohjelmiin, jotka on kuvattu kuvioden 11-23 vuokaavioissa.

Yleensä on uusi ja parannettu ryhmävalvontastrategia luonteeltaan universaalinen, mahdollistaen sen soveltamisen mihin tahansa rakennukseen ilman merkittävää modifikaatiota. Järjestelmäprosessori on täysin riippuvainen informaatiosta eri korinohjaajista sen suhteen, mitä kukin kori pystyy tekemään. Järjestelmäprosessori käyttää tätä informaatiota rakennuksen sen muodon asettelemiseksi, joka kulloinkin on olemassa, so. mitkä korit ovat palveluksessa ja mitä kerroksia ja palvelusuuntia näistä nämä palveluksessa olevat korit on valtuutettu palvelemaan. Sen jälkeen järjestelmäprosessori soveltaa universaali­strategiaansa tähän muotoon.

Universaalistrategia pyrkii jakamaan kaikkien palveluksessa olevien korien kesken tasaisesti kulloisenkin työkuorman sekä sen työkuorman, joka voi muodostua osoitusten välillä. Tämän todellisen ja mahdollisen työkuorman jakaminen perustuu tiettyihin dynaamisiin keskimääriin, jotka on laskettu juuri ennen osoitusten tekemistä.

Osoitukset ovat primäärisesti kerrostasonappi-orientoituja paremmin kuin kerrostasokutsuorientoituja, ainakin siihen saakka kunnes kerrostasokutsut on osoitettu korille, koska kerrostasopainonappien osoi-

tukset vastaavat yhtä sovellettavaa dynaamista keskiarvoa. Jokainen kerrostasokutsunappi on tehokkaasti osoitettu pyyhkäisyajaväliin ja nämä aikavälit on osoitettu koreille mainitun universaalistrategian mukaisesti. Hissijärjestelmä on sarjamuotoinen, aikamultipleksoitu järjestely, jossa aikavälit kerroksia varten otetaan vuorotellen.

Aikavälien osoitusta eri koreille ei tehdä viereisten kerrosten jäykän ryhmän perusteella, mikä normaalisti liittyy vyöhykeajatteluun, sitä ei tehdä viereisten kerrosten joustavan ryhmän perusteella, mikä normaalisti liittyy leijuvavyöhykeajatteluun korien toiminnan kesken eikä se ole sattumanvaraista toimintaa. Aikavälien osoittaminen on rakennettu ennaltamäärätylle prioriteettirakenteelle, johon kuuluu:

- (1) tiettyjen aikavälien osoitusten tyhjäminen ennen kutakin osoitusprosessia;
- (2) aikavälien osoittaminen yleisessä järjestyksessä, joka perustuu koreihin, jotka palvelevat samaa kerroskombinaatiota, jokaista tällaista joukkoa on nimitetty ryhmäksi (sarjaksi);
- (3) ryhmien aikavälien osoitus useissa osoitusten läpiajoissa, muuttaen sovellettuja rajoituksia ja valvoen dynaamisia keskimääriä kussakin läpiajossa, rajoituksin ja dynaamisin keskimäärin, jotka sisältyvät niihin, jotka ovat ryhmäorientoituja sekä rakennusorientoituja;
- (4) aikavälien osoitus koreille, jotka on valtuutettu kuhunkin ryhmään, dynaamisen koriprioriteettijärjestyksen mukaan, joka on laskettu ennen kutakin osoitusprosessia todellisen työkuorman mukaan sekä ottaen huomioon sellaiset seikat kuin onko korilla vai eikö sillä ole seuraava-(NEXT)osoitus, ja onko koriin liittyvä moottorigeneraattoriyksikkö katkaistuna ennaltamäärätyksi epäaktiivisuusajaksi;
- (5) aikavälien osoittaminen koreille, aloittaen koreista ennaltamäärättyyn suuntaan, jolloin varatun korin ennaltamäärätynä suuntana on sen liikesuunta ja joutilaan korin ennaltamäärätyn suunnan perustuessa kulloinkin esiintyviin liikenneolosuhteisiin ja varattujen korien osoitussuuntiin;
- (6) aikavälien osoittaminen varattuihin (toimessa oleviin) koreihin sillä rajoituksella, että asianomaiset kerrokset ovat ennaltamäärätyn kulkumatkan päässä korista, mikä on eri asia kuin fysikaalinen väli; ja
- (7) aikavälien osoittaminen palveluksessa oleville joutilaille koreille ilman kulkumatkarajoituksia (6).

osoitusprosessin kuvaus koskee aikavälien osoittamista koreille. Kukin aikaväli liittyy eri kerrostasokutsupainonappiin, ja kerrostasokutsunapit liittyvät niihin suuntiin kerroksesta, johon kerroksesta halutaan matkustaa. Täten aikavälien osoituksen koreihin voidaan ajatella olevan osoitus koreille kerrostasoille ja palvelusuuntiin näistä taikka lyhyesti, osoitus palvelusuunnista kerrostasoilta koreille. On huomattava, että sanonnalla palvelusuunta, kun sitä sovelletaan kerrostasoihin osoitusprosessissa, tarkoitetaan suuntaa kerroksesta, johon tämän kerroksen liikenne pyrkii matkustamaan eikä se liity palvelusuuntien asetteluun eri hissikoreille.

Kuviossa 1 esitetyn hissijärjestelmän 10 käynnistystä on osoitettu viitenumerolla 320. Vaihe 322 lukee sisäänmenosignaaleit $\overline{IN0} - \overline{IN3}$, jotka on johdettu ohjausmuistin 82 (kuvio 4) sisääntuloon eri koreista ja varastoi signaalit datamuistiin 86. Vaihe 324 laskee niiden hissikorien lukumäärän, jotka ovat palveluksessa järjestelmäohjauksella 22 (N_{SC}) ja vaihe 326 toteaa, onko vähintään kaksi koria järjestelmäohjauksen 22 ohjauksessa. Ellei ole, ei ole mitään tarvetta ryhmäohjaukseen ja ohjelma tekee silmukan takaisin vaiheeseen 322. Ohjelma pysyy tässä silmukassa kunnes vähintään kaksi koria on palveluksessa järjestelmäohjauksella 22. Ilman ryhmävalvontaohjausta korit pystyvät vastaamaan kaikkiin kerrostasokutsuihin ja ne vastaavat hissipalvelukutsuihin sen strategian mukaisesti, joka on rakennettu niiden yksilöllisiin koriohjauslaitteisiin, kuten jäljempänä selitetään.

Jos vaihe 326 havaitsee, että vähintään kaksi koria on palveluksessa järjestelmäohjauksella 22, ohjelma etenee vaiheeseen 328, joka muodostaa alas- ja ylös-kutsupeitteet. Alas- ja ylöskutsupeitteet varastoidaan datamuistiin 86 RAM9 ja vast. RAM10 päämuisteihin. Kun puhutaan RAMEista 0-15, on hyödyllistä tarkistaa RAMin numero kuvion 5 kartasta. RAMit 9 ja 10 olennaisesti näyttävät kerrostasovaltuutussignaalit MT01 ja MT00 osoittaen kullekin korille kerroksen ja suunnat näistä, jotka voidaan palvella korilla. Täten jos RAMin 10 binäärisana, joka vastaa kerrostasoa 15, on esimerkiksi 0111, se osoittaa, että vain korit 0, 1 ja 2 pystyvät palvelemaan kerrostasokutsuja ylöspäin kerroksesta 15. On huomattava, että tämä järjestely säilyttää ohjelman universaalisuuden, tehden sen sovellettavaksi mihin tahansa rakennusmuotoon, koska ohjelma saa koreista informaation rakennuksen muodosta ja sitten varastoi rakennuksen muodon kunnes muutos tapahtuu.

Vaihe 330 laskee aikavälit kussakin ryhmässä sekä aikavälien kokonaismäärän rakennuksessa ja varastoi nämä summat tulevaa käyttöä varten. Kullekin kerrostasokutsunapille on osoitettu aikaväli. Täten rakennuksessa, jossa on 16 kerrostasoa, ensimmäisellä ja kuudennella toista kerrostasolla on kummallakin yksi aikaväli ja välissä olevilla 14 kerroksella tai tasolla kullakin on 2 aikaväliä, mikä tekee yhteensä 30 aikaväliä. Ryhmä tarkoittaa niiden kerrosten joukkoa, joita palvelee samalla koriyhdistelmällä. Esimerkiksi 4 korilla voi olla 16 eri ryhmää, joista ryhmä 0000 on epäkelpo ryhmä. Jos kaikki korit palvelevat kaikkia kerroksia, on olemassa vain yksi kelvollinen ryhmä. Keskimääräisessä rakennuksen muodossa on tavallisesti vain harvoja ryhmiä, mutta ohjelma käsittelee maksimaalisen mahdollisen ryhmien lukumäärän.

Vaihe 332 määrää keskimääräisen lukumäärän aikavälejä ryhmää kohden A_{SI} , jakamalla aikavälit kussakin ryhmässä, jotka on määrätty vaiheessa 330, niiden palveluksessa olevien korien lukumäärällä, jotka pysyvät palvelemaan ryhmää (N_{SCI}). Vaihe 332 määrää myös A_{SB} :n, keskimääräisen aikavälien lukumäärän rakennuksessa palveluksessa olevaa hissikoria kohden jakamalla aikavälien kokonaismäärän rakennuksessa luvulla N_{SC} , palveluksessa olevien korien luvulla.

Vaiheet 334 ja 336 sitten toistavat vaiheet 322 ja 336, lukemalla ohjausmuistin 82 ROM 1:n sisääntulon ja laskemalla palveluksessa olevat korit. Vaihe 338 määrää, onko tapahtunut muutosta rakennuksen muodossa sisääntulon viimeisen lukemisen jälkeen. Esimerkiksi vaihe 338 määrää, onko palveluksessa olevien korien lukumäärä muuttunut. Jos on tapahtunut muutos, ohjelma palaa vaiheeseen 322, kun kerrosvaltuuspeitteet ja aikaisemmin muodostetut aikavälikeskimäärät eivät enää pidä paikkaansa ja täten ne on saatettava ajan tasalle käyttäen viimeisintä rakennuksen muotoa.

Jos vaihe 338 havaitsee, että ei ole tapahtunut mitään muutosta, joka tekee N_{SC} , A_{SB} tai A_{SI} kelvottomaksi jollekin ryhmälle, ohjelma etenee vaiheeseen 340. Vaihe 340 laskee kerrostasokutsujen lukumäärän ryhmää kohden sekä kerrostasokutsujen kokonaismäärän rakennuksessa ja varastoi nämä summat myöhempää käyttöä varten.

Vaihe 342 määrää keskimääräisen lukumäärän rekisteröityjä kerrostasokutsuja ryhmää kohden, A_{CI} , jakamalla kunkin ryhmän kerrostasokutsujen lukumäärän ryhmää palvelevien, palveluksessa olevien korien lu-

vulla. Keskimääräinen rekisteröityjen kerrostasokutsujen lukumäärä koria kohden rakennuksessa, A_{CB} , määrätään jakamalla kerrostasokutsujen kokonaismäärä rakennuksessa luvulla N_{SC} , palveluksessa olevien hissikorien luvulla.

Vaihe 344 tarkistaa liikenteen erityisolosuhteet, kuten esim. sellaiset, jotka aloittavat huippuliikenneominaisuudet ylöspäin tai alaspäin. Jos havaitaan olotila, joka aloittaa huippuliikennetilän, vaihe 344 ottaa käyttöön strategian, joka liittyy havaittuun huippuun.

Vaihe 346 tarkistaa, onko olemassa erityisiä kerrostaso-ominaisuuksia, kuten pääkerros- tai kokouskerros-ominaisuuksia. Jos on olemassa vaatimus yhdestä tai useammasta erikoiskerros-ominaisuudesta, vaihe 346 ottaa käyttöön strategian, joka liittyy valittuun erikoiskerros-ominaisuuteen.

Vaihe 348 tyhjä ylös- ja alas-osoitustaulut, jotka on varastoitu RAMEihin 6 ja 7, kaikista aikaväliosoituksista, paitsi niistä, joille aikaisemmin osoitettiin aikavälit, joihin liittyy rekisteröity kerrostasokutsu ja niistä, jotka kuuluvat yhden korin ryhmään.

Vaihe 350 poistaa liiat aikaväliosoitukset. Esimerkiksi jos kutsujen lukumäärä yhden korin ryhmästä osoitettuna korille on sama tai ylittää kerrostasokutsujen keskimäärän rakennuksessa koria kohden A_{CB} , kaikki muut osoitukset tälle korille tyhjäetään. Jos kutsut, jotka on osoitettu korille yhden korin ryhmästä eivät ylitä A_{CB} , mutta kaikki kutsut, jotka on osoitettu korille ovat sama tai suurempi kuin A_{CB} , vaihe 350 laskee aikavälit, jotka on osoitettu korille, jolla on rekisteröity kerrostasokutsu, käynnistyen aikavälissä, joka liittyy korin asemaan ja edeten korin liikesuuntaan ja kun saavutetaan rakennuksen kutsukeskimäärä A_{CB} koria kohden, tyhjäetään kaikki muut aikavälit, jotka on osoitettu tälle korille.

Vaihe 352 osoittaa suunnan palveluksessa olevasta joutilaasta korista, johon aikaväliosoitukset on tehtävä korille. Jos kori on työssä, pyyhkäisysuunta aikavälien osoittamiseksi korille on korin liikennesuunta. Työssä olevien korien osoitetut pyyhkäisysuunnat otetaan huomioon yhdessä olemassa olevien liikenneolosuhteiden kanssa määrättäessä pyyhkäisysuuntaa, joka on osoitettava palveluksessa olevalle joutilaalle korille. Tietyissä tapauksissa, jotka selitetään jäljempänä,

on myös sopivaa käyttää viimeistä palveluksessa olevan joutilaan korin liikesuuntaa.

Vaihe 354 osoittaa järjestyksen, jossa koreja käsitellään osoitettaessa niille aikavälejä, kori, jolla on pienin yhdistetty määrä kerrostasokutsuja ja korikutsuja käsitellään ensin jne.

Vaihe 356 osoittaa kunkin koriryhmän aikavälit vaiheen 354 määräämässä korijärjestyksessä. Ryhmiä käsitellään siinä järjestyksessä kuin korien lukumäärä nousee ryhmää kohden. Aikavälien osoittaminen kuhunkin ryhmään kuuluville koreille tehdään useissa läpiajoissa, esimerkiksi kolmessa. Ensimmäinen osoitusläpiajo on erityinen osoitusläpiajo, joka ottaa huomioon ennalta identifioidut tilanteet ja etuoikeudet. Esimerkiksi niihin kerroksiin liittyvät aikavälit, joihin koreissa on osoitettu korikutsu, osoitetaan asianomaisille koreille; ylös- ja alas-aikavälit, jotka liittyvät kerrokseen, jossa seisoo palveluksessa oleva joutilas kori, osoitetaan tälle korille; jos on olemassa kori, jolla on NEXT-osoitus, tälle korille osoitetaan aikaväli, joka liittyy pääkerroksen ylöspäin palvelusuuntaan; ja jos on olemassa kori, jolla on kokouskerrososoitus CONV, tälle korille osoitetaan molemmat aikavälit, jotka liittyvät kokouskerrokseen. Toinen ajo on yleinen osoitus, joka osoittaa aikavälit niiden ryhmien koreille, jotka ovat ennaltamäärättyjen dynaamisten rajoittavien keskimäärä- ja etäisyysrajoitusten alaisina. Kolmatta ajoa voidaan käyttää yrittämään osoittaa ei-osoitetut aikavälit, joita voi olla jäljellä kahden ensimmäisen ajon jälkeen. Kolmas ajo poistaa tiettyjä rajoituksia, joita käytetään toisen ajon aikana.

Vaihe 358 lukee RAMit 4, 5, 6 ja 7 datamuistin 86 ulostuloon, missä informaatio näistä RAMEista esiintyy sarjamuotoisina ulostulosignaaleina $\overline{OUT0}$, $\overline{OUT1}$, $\overline{OUT2}$ ja $\overline{OUT3}$ koreille 0, 1, 2 ja 3.

Sen jälkeen kun on ulosannettu osoitukset koreille, ohjelma palaa vaiheeseen 334, joka jo on edellä kuvattu.

Kuvio 10

Kuvio 10 on vuopiirros alaohjelmasta LCD2, jota voidaan käyttää lukemaan sarjamuotoisia sisäänmenosignaaleja $\overline{IN0}$ - $\overline{IN3}$ koreista, jotka signaalit esiintyvät ROM1:n sisään tulossa, ja varastoimaan nämä signaalit RAMEihin 0, 1, 2 ja 3. Kuten on esitetty kuviossa 5, tilasignaali

lit kustakin korista, jotka esiintyvät aikavälin ensimmäisessä neljänneksessä, varastoidaan RAM0:aan, kerrostasokutsut ylös ja alas, $\overline{1Z}$ ja $\overline{2Z}$, jotka esiintyvät signaalien $\overline{IN0}$ ja $\overline{IN1}$ toisessa neljänneksessä, varastoidaan RAM1:een. Kerrosvaltuutussignaalit $\overline{FEN}$, jotka esiintyvät kolmannessa neljänneksessä, varastoidaan RAM2:een ja korikutsut $\overline{3Z}$ varastoidaan RAM3:een. Kuten jäljempänä selitetään, kerrosvaltuutussignaalit $\overline{FEN}$ varastoidaan vain hetkellisesti RAM2:een ja siirretään myöhemmin toiseen RAM-muistipaikkaan, kun muodostetaan ylös ja alas kutsupeitteet.

Tarkemmin sanottuna alaohjelma LCD2 alkaa kohdasta 360 ja vaihe 362 tyhjää akun ja CPU80:n muistilinkin CY (kuvio 4), kun kaikki sisään-tulosiirrot on tehty akun lävitse. Kuten edellä selitettiin, kuviossa 2 graafisesti esitettyä ja kuvion 6 mukaisella kovalla tavalla kehitettyä signaalia $\overline{MXCT}$ käytetään CPU 80:llä määrittämään pyyhkäisyjakson alkua. Signaali $\overline{MXCT}$ on matala viimeisen aikavälin aikana ja CPU 80 synkronoi itsensä pyyhkäisyjakson kanssa signaalin $\overline{MXCT}$ negatiiviseksi tulevalla muutoksella. Vaihe 364 on silmukoitu takaisin itseensä, kun $\overline{MXCT}$ on nolla, koska se on kadottanut negatiiviseksi menevän muutoksen. Kun $\overline{MXCT}$ on looginen ykkönen, ohjelma etenee vaiheeseen 366, joka määrää, onko $\overline{MXCT}$ looginen ykkönen. Niin kauan kun $\overline{MXCT}$ pysyy loogisena ykkösenä, vaihe 366 toistuu. Niin pian kun $\overline{MXCT}$ tulee loogiseksi nollaksi, ohjelma etenee vaiheeseen 368. Vaihe 368 lukee ROM1 sisäänmenon RAMEihin 0, 1, 2 ja 3, aikavälin kerrallaan. Kunkin aikavälin jälkeen vaihe 370 tarkistaa onko pyyhkäisyjakso päättynyt ja ellei ole, vaihe 368 toistetaan seuraavan aikavälin lukemiseksi ja varastoimiseksi. Kun kaikki aikavälit on luettu ja varastoitu, vaihe 370 etenee alaohjelman poistumiskohtaan 372.

Kuvio 11

Kuvio 11 on vuokaavio alaohjelmasta LCD1, jota voidaan käyttää laskemaan korit, jotka ovat palveluksessa järjestelmän ohjauslaitteella 22 ja täten sitä voidaan käyttää suorittamaan toimintoja, joita kuviossa 9 on osoitettu lohkoilla 324 ja 336. Koska huomioon otetaan kaikki hissikorit, vaihe 382 valmistaa korisilmukan aloittamalla korinumeroinnin tai laskennan korista 0. Vaihe 382 myös tyhjää binäärilaskimen, joka sisältää palveluksessa olevien korien lukumäärän N_{SC} .

Vaiheet 383 ja 386 lukevat 4-bittiset sanat INSC ja BYPS, jotka sijaitsevat RAM 0:ssa (kuvio 5) ja sanat varastoidaan ohimenevästi ase-

maan, jossa bittejä voidaan tutkia. Vaihe 388 tutkii BYPS-sanan bitin, joka liittyy koriin 0, ja jos se on looginen yksi, kori on ohittamassa kerrostasokutsuja eikä sitä lasketa palveluksessa olevaksi koriksi. Sitten ohjelma etenee vaiheeseen 396, joka lisää korinumeroa, niin että kori 1 voidaan tutkia. Jos kori ei ole ohittamassa, vaihe 388 etenee vaiheeseen 390, joka tarkastaa sanan INSC koriin 0 liittyvän bitin sen määrittämiseksi, onko kori palveluksessa järjestelmäprosessorilla korin 0 ohjauslaitteen mukaan. Jos tämä bitti on looginen nolla, korien ei lasketa ja ohjelma etenee vaiheeseen 396. Jos INSC-bitti on looginen ykkönen, ohjelma etenee vaiheeseen 392 mikä lisää N_{SC} , palveluksessa olevien korien binäärilaskentaa järjestelmän ohjauslaitteen näkökulmasta katsoen. Vaihe 394 valtuuttaa sanan INSV bitin korille 0. Sana INSV on 4 bitin sana, yksi bitti kullekin korille, joka osoittaa onko kukin kori vai eikö ole palveluksessa järjestelmän ohjauslaitteen 22 mukaan.

Vaihe 393 etenee vaiheeseen 396, joka lisää korinumeroa ja vaihe 398 tarkastaa ovatko kaikki korit tulleet huomioon otetuiksi. Elleivät ole, vaihe 398 palaa vaiheeseen 388 tarkastaakseen BYPS ja INSC bititälle korille. Kun kaikki korit on otettu huomioon, on tullut muodostetuksi uusi 4-bittinen sana INSV ja vaihe 400 lataa tämän sanan RAM 0:n tilamerkkimuistiin. Vaihe 402 lataa N_{SC} laskennan RAM 0:n tilamerkkimuistiin ja ohjelma poistuu poistumiskohdastaan 404.

Kuvio 12

Kuvio 12 on vuokaavio alaohjelmasta LCD9, jota voidaan käyttää muodostamaan ylös- ja alaskutsupeitteet, jotka on spesifioitu kuvion 9 lohossa 328. Alaohjelma LCD9 alkaa kohdassa 410 ja vaihe 412 tyhjä ylös ja alas kutsupeitteet RAMEista 10 ja 9, se aloittaa kerrosten laskun aikaväliin 00, se aloittaa korien laskun korista 0 ja se asettelee testilipun nolnaan kullekin korille.

Vaihe 414 lukee 4-bitin kerrosvaltuutussanan RAM 2:n päämuistin aikavälistä 00 ja kirjoittaa tämän sanan RAMien 9, 10 ja 11 päämuistin aikaväliin 00. RAM 11 on uusi sijaintipaikka kerrosvaltuutussignaalille, kun kaikki kerrokset ovat tulleet huomioonotetuiksi, mikä jättää RAM 2:n käytettäväksi muiden signaalien varastoimiseen ja RAMit 9 ja 10, kun kaikki kerrokset on käsitelty, osoittavat kerroksen, josta kukin kori voi palvella alaspäin ja vastaavasti ylöspäin kutsuja.

RAM 9:n alaskutsupeitteet ovat samanlaisia kuin kerrosvaltuutus RAM-kartta ja RAM 11:ssa, paitsi että pyyhittää pois kerrosvaltuutusbitti alinta kerrosta varten, jota kori on valtuutettu palvelemaan. RAM 10:n ylöskutsupeitteet ovat samanlaiset kuin kerrosvaltuutus RAM-kartta RAM 11:ssa, paitsi että pyyhittää pois kerrosvaltuutusbitti ylintä kerrosta varten, jota kori on valtuutettu palvelemaan. Kuvion 12 ohjelma suorittaa näiden bittien poistamisen ylös- ja alas-kutsupeitteiden muodostamiseksi.

Esimerkin vuoksi oletetaan, että rakennuksessa on 16 kerrosta ja kaikki korit on valtuutettu palvelemaan kaikkia kerroksia ja täten bittien RAM 9 päämuistissa aikavälissä 00 tulee kaikkien olla looginen nolla, samalla kun RAM 9 päämuistin muut bitit ovat looginen ykkönen ja RAM 10 päämuistin aikavälin 15 kaikkien bittien tulee olla nolla, samalla kun RAM 10 päämuistin muut bitit ovat looginen ykkönen.

Tarkemmin sanottuna, sen jälkeen kun kerrosvaltuutussana aikaväliä 00 varten on kirjoitettu RAMEihin 9, 10 ja 11, vaihe 416 tarkistaa tämän sanan samalla kun se on CPU 80:n akussa. Jos tämä aikaväli ei ole osoitettu kerrokselle, kuten silloin, kun on olemassa useampia aikavälejä kuin kerrostasoja, sanan kaikki bitit ovat nollia ja vaihe 416 etenee vaiheeseen 436, joka lisää (increments) kerroslaskentaa. Esimerkissä ensimmäisen sanan kaikki bitit ovat ykkösiä ja koska sana ei ole pelkkiä nollia, ohjelma etenee vaiheeseen 418, joka siirtää akkua oikealle sijoittaakseen koriin 0 liittyvän bitin akun muistinumeroon CY. Sitten vaihe 420 tarkistaa muistinumeron ja jos tämä on nolla, mikä osoittaa, että tämä kori ei ole valtuutettu tätä kerrosta varten, ohjelma etenee vaiheeseen 428, joka lisää korilaskentaa. Esimerkissä kaikki korit ovat valtuutettuja kaikkiin kerroksiin, niin että muistinumero on ykkönen ja ohjelma etenee vaiheeseen 422, joka kuormaa tämän kerroksen osoitteen ylös-pyyhkimisrekisteriin tätä koria varten, so. indeksirekisteriin CPU 80:ssa. Joka kerran kun tämä kori havaitaan valtuutetuksi ylempää kerrosta varten, tämän ylemmän kerroksen osoite kirjoitetaan alemman kerroksen osoitteen päälle. Sen vuoksi, kun kaikki kerrokset on käsitelty, osoite ylös-poistamisrekisterissä on ylimmän kerroksen osoite, jota kori on valtuutettu palvelemaan ja bitti tätä kerrosta, tätä koria varten pyyhittää RAM 10:ssä, ylöspäinkutsupeitteessä.

Sitten vaihe 424 tarkistaa testilipun tätä koria varten. Jos se on

nolla, se osoittaa, että alaspäin peitebitti alimmalle kerrokselle, jota tämä kori voi palvella, ei vielä ole tullut pyyhityksi ja vaihe 426 pyyhkii muistinumeron tämän bitin pyyhkimiseksi ja testilippu tälle korille asetellaan 1:ksi, osoittaakseen seuraavalle kerralle kun vaihe 424 tavataan, että on hypättävä yli vaiheen 426. Vaihe 428 lisää korinumeroa ja vaihe 430 tarkistaa, ovatko kaikki korit tulleet huomioonotetuksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 418 tarkistaakseen kerrosvaltuutussanan bitin seuraavalle korille.

Kun kaikki korit on käsitelty tämän kerroksen kerrosvaltuutussanan suhteen, vaihe 432 siirtää akkua oikealle palauttaakseen kerrosvaltuutussanan alkuperäiseen tilaansa ja vaihe 434 lataa tämän sanan asianomaiseen aikaväliin RAM 9:ssä, alaspäinkutsupeitteessä. Koska alimman kerroksen bitit, jota korit ovat valtuutetut palvelemaan, on eliminoitu sanasta vaiheessa 426, oikea alaspäinkutsupeite synnytetään yksinkertaisesti kirjoittamalla sana, joka pidettiin akussa saman aikavälin sanan päällä, alaspäinkutsupeitteeseen, RAM 9:ään.

Vaihe 436 lisää kerroslaskua, vaihe 438 tarkistaa, ovatko kaikki kerrokset tulleet käsitellyiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 414, joka lukee tälle kerrokselle kuuluvan kerrosvaltuutussanan RAM 2:sta RAMEihin 9, 10 ja 11. Vaiheet suoritetaan kuten edellä paitsi että nyt testilippu on ykkönen kaikille koreille, hypäten yli vaiheen 426, koska enempää bittejä ei ole poistettava sanasta ennen lataamista RAMiin 9.

Kun vaihe 438 havaitsee, että kaikki kerrokset on käsitelty, ylöspyyhkimisrekisteri jokaiselle korille sisältää sen ylimmän kerroksen osoitteen, jota kukin kori on valtuutettu palvelemaan ja vaihe 440 aloittaa korilaskennan ja lataa tämän ylöspyyhkimisosoitteen korille 0 akkuun. Vaihe 442 käyttäen tätä osoitetta pyyhkii tämän korin ja kerroksen bitin RAM 10:ssä, ylöspäinkutsupeitteessä. Vaihe 444 lisää korilaskentaa ja vaihe 446 tarkistaa, onko kaikki korit käsitelty. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 440. Kun kaikki korit on käsitelty, ylöspäinpeite on täydellinen ja koska alaspäinpeite oli täydellinen, kun vaihe 438 eteni vaiheeseen 440, ohjelma poistuu poistumiskohdassa 448.

Kuvio 13

Kuvio 13 esittää vuodiagrammaa alaohjelmasta LCD10, jota voidaan käyttää laskemaan aikavälien kokonaismäärä rakennuksessa sekä aika-

väliden lukumäärä kussakin ryhmässä, joka vastaa lohko toimintoa 330 kuviossa 9. Alaohjelma LCD-10 tulee mukaan tulokohdassa 450 ja vaihe 452 kuormaa RAM 10 osoitteen, ylöspäinkutsu peitteeseen, akkuun, ja asettaa lipun ykköseksi. Vaihe 454 tyhjä sanan A_{SB} , aikaväliden keskimääräisen lukumäärän rakennuksessa palveluksessa olevaa koria kohden, joka sana sijoitetaan tilamerkkimuistiin RAMissa 8 ja se myös tyhjä RAMin 8 päämuistiin, mihin on varastoitu ryhmien A_{SI} sanat. Sana A_{SI} on ryhmän keskimääräinen lukumäärä aikavälejä ryhmää palvelemaan valtuutettua koria kohden.

RAMissa 8 rivit viittaavat ryhmänumeroihin eikä aikaväleihin tai kerrostasoihin. Valvontalaitteiden universaalisuutta on parannettu antamalla jokaiselle kuudelletoista mahdolliselle ryhmälle, laskien mukaan pätemättömän ryhmän, jossa mitkään korit eivät palvele aikaväliä, eri binäärinumero 0000-1111. Ryhmää koskeva informaatio varastoidaan RAMin muistiin ryhmän binäärinumeron mukaisesti. Ryhmää 0001 koskeva informaatio, esimerkiksi, varastoidaan riville 1, ja ryhmää 1111 koskeva informaatio varastoidaan riviin 15. Peitesanaa kerrokselle, sekä ylöspäin- että alaspäinpeitteissä, käytetään ryhmänumerona. Sen vuoksi ei ole tarpeen CPU 80:n määrätä, kuinka monta ryhmää on olemassa taikka mitä ne ovat. Esimerkiksi jos ylöspäin- ja alaspäin-peitesana kerrokselle on 1111, joka osoittaa, että kaikki korit on valtuutettu palvelemaan kerrosta ja tähän aikaväliin liittyvää suuntaa siitä, tämä aikaväli kuuluu ryhmään 1111 ja tätä ryhmää koskeva informaatio varastoidaan RAMin päämuistiin riviin 15. Jos peitesana on 1100, joka osoittaa, että vain korit 2 ja 3 ovat valtuutetut palvelemaan tähän aikaväliin liittyvää kerrosta ja suuntaa siitä, tämä aikaväli kuuluisi ryhmään 1100, joka varastoitaisiin riviin 12. Jos nämä ovat ainoat kelvolliset ryhmät, vain rivejä 12 ja 15 käytetään varastoimaan ryhmiä koskevaa informaatiota ja kaikki muut rivit sisältävät nollia.

Tarkemmin sanottuna vaihe 454 etenee vaiheeseen 456, joka aloittaa kerroslaskennan ja vaihe 458 lukee ylöspäinpeitesanan aikaväliä 00 varten. Vaihe 460 tarkistaa ja määrää, onko aikaväli liittynyt kerrokseen. Jos peitesana on nolla, se ei ole liittynyt kerrokseen ja ohjelma etenee vaiheeseen 468, joka lisää kerroslaskentaa. Jos sana ei ole nolla, vaihe 462 lisää kokonaisaikavälisummaa, joka on varastoitu työmuistiin (scratch pad memory), kuten esimerkiksi jonkin RAMin 12, 13, 14 tai 15 päämuistiin.

Vaihe 464 lataa sen ryhmän osoitteen, johon tämä aikaväli kuuluu, joka, kuten edellä selitettiin, on sama kuin kysymyksessä oleva peitesana, ja vaihe 466 lisää aikavälikokonaissummaa tälle ryhmälle. Täten jos peitesana oli 1111, osoite 1111, joka on RAMin päämuistin rivi 15, tulee lisätyksi yhdellä.

Vaihe 468 lisää kerroslaskentaa ja vaihe 470 määrittää, ovatko kaikki aikavälit tulleet käsitellyiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 458 lukeakseen peitesanan seuraavalle aikavälille. Kun vaihe 470 havaitsee, että kaikki aikavälit ovat tulleet käsitellyiksi, vaihe 472 lataa RAMin 9 osoitteen, alaspäinkutsupeitteen, akkuun ja vaihe 474 tarkistaa lipun. Jos lippu on ykkönen, se osoittaa, että alaspäinkutsupeitteitä ei vielä ole käsitelty, vaihe 476 asettaa lipun nolllaksi ja ohjelma palaa vaiheeseen 456 käsitelläkseen alaspäinkutsupeitteet. Kun vaihe 474 havaitsee lipun nolllaksi, sekä ylöspäinkutsuettua alaspäinkutsupeitteet on käsitelty ja ohjelma poistuu poistumiskohdassa 478.

Kuvio 14

Kuvio 14 on vuokaavio alaohjelmasta LCD11, jota voidaan käyttää kuvion 9 molempiin lohko-toimintoihin 332 ja 342. Toiminto 332 määrää keskimäärät A_{SB} ja A_{SI} ja toiminto 342 määrää keskimäärät A_{CB} ja A_{CI} . Jos kaikkia koreja ei ole valtuutettu samoille kerroksille ja samoihin palvelusuuntiin, on olemassa enemmän kuin yksi ryhmä ja kullakin ryhmällä on omat keskimääränsä A_{SI} ja A_{CI} . Rakennuskeskimäärillä A_{SB} ja S_{CB} ei ole mitään suhdetta ryhmäkeskimääriin. Jos kaikki korit on valtuutettu kaikkiin kerroksiin, on olemassa vain yksi ryhmä. Tässä tapauksessa tämän yhden ryhmän keskimäärä A_{SI} on sama kuin rakennuskeskimäärä A_{SB} ja tämän ryhmän keskimäärä A_{CI} on sama kuin keskimäärä A_{CB} . Kuviota 14 selitettäessä oletetaan, että se on toiminto 332. Toiminnon 342 selityksen saamiseksi on vain tarpeen korvata "kerrostaokutsut" "aikaväleillä" RAM 2 RAMilla 8, A_{CB} A_{SB} :llä ja A_{CI} A_{SI} :llä.

Alaohjelma LCD11 alkaa kohdassa 490 ja vaiheessa 492 sana N_{SC} , järjestelmän valvontalaitteen 22 mukaan palveluksessa olevien korien lukumäärä, joka on varastoitu RAMin 0 tilamerkkimuistiin, ladataan akkuun ja ryhmälaskenta aloitetaan, niin että ryhmät voidaan tutkia ryhmänumer järjestyksessä.

Vaihe 494 lataa rakennuksen aikavälien kokonaismäärän, joka oli va-

rastoitu väliaikaiseen kohtaan kuvion 13 vaiheessa 462. Vaihe 494 jakaa rakennuksen aikavälien kokonaismäärän luvulla N_{SC} ja varastoi tuloksen, binäärisanan A_{SB} , RAMin 8 tilamerkkimuistiin.

Vaihe 500 lataa ensimmäisen ryhmän osoitteen ja aikavälien kokonaismäärän tässä ryhmässä. Tämän ryhmäosoitteen aikavälien kokonaismäärä määrättiin kuvion 13 vaiheessa 466. Vaihe 502 määrää, onko todellista ryhmää tarkistamalla, onko ryhmän aikavälinumero nolla. Jos se on nolla, ohjelma etenee vaiheeseen 510, joka lisää ryhmänumeroa. Jos aikavälien kokonaismäärä ei ole nolla, vaihe 504 määrää niiden palveluksessa olevien korien luvun, jotka ovat valtuutetut palvelemaan ryhmää N_{SCI} , mikä määrätään laskemalla ykköset ryhmänumerossa, ja vaihe 505 jakaa aikavälien kokonaismäärän tämän ryhmän N_{SCI} :llä. Osmäärä on tämän ryhmän A_{SI} , so. aikavälien keskimääräinen lukumäärä palveluksessa olevaa koria kohden ja vaihe 508 varastoi tämän luvun, binääriluvun, RAMin 8 päämuistin riviin, joka vastaa tämän ryhmän osoitetta.

Vaihe 510 lisää ryhmänumeroa ja vaihe 512 määrittää, ovatko kaikki ryhmät tulleet huomioonotetuiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 500. Jos kaikki ryhmät on käsitelty, vaihe 512 etenee poistumiskohtaan 514.

Kuvio 15

Kuvio 15 on vuokaavio alaohjelmasta LCD4, jota voidaan käyttää kuvion 9 lohko toimintona 340 laskemaan kerrostasokutsujen kokonaismäärää sekä kerrostasokutsujen lukumäärää kussakin ryhmässä.

Alaohjelma LCD4 alkaa kohdassa 520 ja vaihe 522 lataa RAMien 1, 9 ja 10 osoitteet, jotka sisältävät ylös- ja alas-kerrostasokutsut, alaspäinkutsupeitesanat ja ylöspäinkutsupeitesanat. Vaihe 524 tyhjä A_{CB} :n, kerrostasokutsujen keskimäärän rakennuksessa palveluksessa olevaa koria kohden, joka oli varastoitu RAMin 2 tilaominaisuusmuistiin ja se tyhjä A_{CI} :n jokaiselle ryhmälle, keskimääräisen kerrostasokutsujen määrän ryhmässä palveluksessa olevia koreja kohden, jotka on valtuutettu palvelemaan ryhmää ja joka keskimäärä oli varastoitu RAMin 2 päämuistiin. Vaihe 526 aloittaa kerroslaskennan ja vaihe 528 lukee kutsusanan RAMin 2 rivistä 00. Vaihe 530 tarkistaa tämän kutsusanan ensimmäisen bitin ylöspäinkutsua varten. Jos ensimmäinen bitti on nolla, ohjelma etenee vaiheeseen 540 tarkistaakseen alaspäinkutsua varten. Jos ensimmäinen bitti on ykkönen, vaihe 532 tarkistaa

RAMiin 10 varastoidun ylöspäinkutsupeitesanan tälle aikavälille. Jos peitesana on nolla, mitään koria ei ole valtuutettu tälle aikavälille ja vaiheen 530 havaitsema "ykkönen" oli kelvoton. Tämän vuoksi ohjelma etenee vaiheeseen 540. Jos vaihe 532 havaitsee, että peitesana ei ole kokonaan nolliä, vaihe 534 lisää rakennuksen kerroskutsujen kokonaislukua, joka on varastoitu väliaikaiseen kohtaan ja vaihe 536 lataa tämän kutsun ryhmäosoitteen. Ryhmäosoite on ylöspäinkutsupeitesana, joka juuri tarkistettiin vaiheessa 532 ja vaihe 538 lisää kerroskutsujen kokonaismäärää tälle ryhmälle, joka kokonaislukumäärä varastoidaan väliaikaiseen kohtaan.

Vaihe 538 etenee vaiheeseen 540, joka tarkistaa kutsusanan toisen bitin RAMista 1. Jos tämä bitti on nolla, ohjelma etenee vaiheeseen 550, joka lisää kerroslaskentaa. Jos toinen bitti on ykkönen, ohjelma tarkistaa alaskutsupeitesanan RAMista 9 tälle aikavälille. Jos peitesana on nolla, vaiheella 540 havaittu kutsu on kelvoton ja ohjelma etenee vaiheeseen 550. Jos peitesana ei ole nolla, vaihe 544 lisää kerroskutsujen kokonaislukua rakennuksessa. Vaihe 546 lataa kutsun ryhmäosoitteen, so. alaskutsupeitesanan tälle aikavälille, ja vaihe 548 lisää kerroskutsujen kokonaislukua tälle ryhmälle.

Vaihe 550 lisää kerroslaskentaa ja vaihe 552 tarkistaa, ovatko kaikki kerrokset (aikavälit) tulleet huomioon otetuiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 528. Jos kaikki kerrokset ovat tulleet huomioon otetuiksi, ohjelma poistuu poistumiskohdasta 554.

Informaatio, joka on tarpeen kuvion 9 toimintoa 342 varten, on nyt käytettävissä ja alaohjelma LCD11 valmistaa keskimäärät A_{CI} kullekin ryhmälle ja A_{CB} :n rakennukselle samalla tavoin kuin edellä selitettiin keskimäärien A_{SI} ja A_{SB} (kuvio 14) valmistukseen nähden.

Kuvio 16

Kuvio 16 on vuokaavio alaohjelmasta LCD12, jota voidaan käyttää kuvion 9 lohko toimintoon 344, joka liittyy liikenteen erityispiirteisiin. Alaohjelma LCD12 toteaa ennaltamäärätyt liikenneolosuhteet ja näihin reagoiden toimii ennaltamäärättyyn suuntaan. Esimerkiksi huippuliikennetilä alaspäinsuunnassa voidaan ilmaista pääkerroksen yläpuolella olevalla korilla, joka on asetettu liikenteeseen alaspäin ohittaen kerrostasokutsut. Tämä voidaan todeta tarkistamalla 4-bittinen sana BYPS, joka on varastoitu RAMin 0 riviin 07. Huippuliikennetilä

ylöspäin suunnassa voidaan havaita kuormatusta korista, joka poistuu pääkerroksesta. Se voidaan myös todeta pääkerroksessa olevasta korista, joka on asetettu liikenteeseen ylöspäin ja ohittamaan kerrostasokutsut. Myös tässä tapauksessa voidaan tarkistaa 4-bittinen sana BYPS.

Jos huippuliikennetapaus sekä ylöspäin että alaspäin sattuvat samanaikaisesti, alaspäin huippuliikenteellä on etusija.

Ennaltamäärätty toimintatapa, jonka alaohjelma LCD12 ottaa riippuvaisena huippuliikennetilasta, määrää niiden korien suhteellisen osuuden, jotka pidetään pääkerroksessa, Q_{MFL} ja aktivoi huippuajastimen. Huippuajastin ylläpitää huippuun liittyvän strategian ennaltamäärätyn ajan kunkin tapauksen esiintymisen jälkeen, jota käytetään osoittamaan huipun esiintymistä.

Alaohjelma LCD12 alkaa kohdasta 560 ja vaihe 562 tarkistaa CPU 80 sisääntulosignaalin $\overline{IN5}$ määrittääkseen, onko pääkerrosominaisuus tosi, mitä osoittaa tosisignaali $\overline{PMNFL}$ (kuvio 6), jota voidaan ohjata manuaalisella kytkimellä. Jos pääkerrosominaisuutta ei ole aktivoitu, ohjelma etenee vaiheeseen 592. Jos pääkerrosominaisuus on aktiivinen, vaihe 564 tarkistaa RAMiin 0 varastoidun BYPS-sanatodetukseen, onko mikään kori ohittamassa kerrostasokutsuja. Kuten edellä mainittiin, tätä testiä voidaan käyttää osoittamaan huiput molemmissa liikennesuunnissa. Jos sana BYPS on nolla, ohjelma etenee vaiheeseen 592. Jos sanan BYPS kaikki bitit eivät ole nollija, vaihe 566 aloittaa korilaskennan ja vaihe 568 tarkistaa RAMiin 0 varastoidun sanan INSC ensimmäisen bitin, joka liittyy koriin 0. Jos tämä bitti on nolla, mikä osoittaa, että tämä kori ei ole palveluksessa järjestelmän ohjauslaitteella 22, ohjelma etenee vaiheeseen 588. Jos kori on palveluksessa, vaihe 570 tarkistaa RAMiin 0 varastoidun sanan BYPS bitin, joka liittyy koriin 0. Jos tämä bitti on nolla, kori ei ole ohittamassa kerrostasokutsuja ja ohjelma etenee vaiheeseen 588. Jos BYPS-bitti on ykkönen, kori on ohittamassa ja vaihe 572 määrittää, liittyykö ohittaminen ylöspäin vai alaspäin liikenteeseen tarkistamalla, onko kori pääkerroksessa. Jos kori on pääkerroksessa, vaihe 574 tarkistaa RAMiin 0 varastoidun sanan UPTR bitin todetukseen, onko kori asetettu liikenteeseen ylöspäin. Ellei, ohjelma etenee vaiheeseen 588. Jos kori on asetettu liikenteeseen ylöspäin, vaihe 576 sijoittaa huippubitin muistin RAM 0 tilamerkkimuistiin, se sijoittaa huipun identifioimisbitin samaan RAMiin osoittamaan huippua, se asettaa

niiden korien suhteellisen osuuden, jotka pidetään pääkerroksessa (Q_{MNF}) johonkin ennaltamäärättyyn lukuun, kuten 2:ksi 4 korin joukosta ja se sijoittaa lipun osoittamaan, että huippubitti on asetettu. Vaihe 578 asettelee huippuajastimen, joka pitää järjestelmän huippuliikennetilassa ylöspäin ennaltamäärätyn ajan.

Jos vaihe 572 havaitsee, että ohittava kori ei ollut pääkerroksessa, vaihe 580 tarkistaa, onko kori pääkerroksen yläpuolella. Ellei, ohjelma etenee vaiheeseen 588. Jos kori on pääkerroksen yläpuolella, sen liikennesuunta tarkistetaan vaiheessa 582 tarkistamalla RAMiin 0 varastoidun sanan UPTR tähän koriin liittyvän bitin. Jos kori on asetettu liikenteeseen ylöspäin, tämä bitti on ykkönen ja ohjelma etenee vaiheeseen 588. Jos kori on asetettu liikenteeseen alaspäin, UPTR-bitti on nolla ja vaihe 584 sijoittaa bitit RAMin 0 tilamerkkimuistiin osoittamaan alaspäinhuippua, suhteellinen pääkerrososuus Q_{MNF} asetellaan joksikin ennaltamäärätyksi numeroksi, esimerkiksi nollaksi 4 korin joukossa ja se tyhjä lipun osoittaakseen, että alaspäinhuippubitti on aseteltu.

Jos on asetettu joko ylöspäin tai alaspäin huippubitti, ohjelma saavuttaa vaiheen 578, joka asettelee huippuajastimen ja vaihe 586 tarkistaa lipun nähdäkseen, onko järjestelmä aseteltu ylöspäin tai alaspäin huippuliikenteeseen. Jos lippu on nolla, mikä osoittaa alaspäinhuippua, mitään muita koreja ei tarvitse tarkistaa, koska alaspäinhuipulla on etusija ylöspäinhuippuun nähden. Jos lippu on ykkönen, osoittaen ylöspäinhuippua, muut korit täytyy tarkistaa sen määrittämiseksi, laukaiseeko jokin alaspäinhuippuominaisuuden, koska alaspäinhuipulla on etusija. Jos vaihe 586 havaitsee, että lippu on aseteltu, vaihe 588 lisää korilaskentaa ja sanat BYPS, INSC ja UPTR siirretään näiden sanojen niiden bittien katsomiseksi, jotka liittyvät uuteen koriin. Vaihe 590 tarkistaa, ovatko kaikki korit tulleet huomioonotetuiksi ja ellei, ohjelma tekee silmukan takaisin vaiheeseen 568.

Kun vaihe 590 havaitsee, että kaikki korit on käsitelty taikka niin pian kun alaspäinhuippu on aktivoitu taikka jos PMNFL tai sana BYPS oli nolla, saavutetaan vaihe 592, joka tarkistaa huippuajastimen. Jos huippuajastin on aktiivinen, ohjelma poistuu kohdassa 596. Jos huippuajastin on ajastanut loppuun, vaihe 594 palauttaa huippubitin RAMin 0 tilamerkkimuistissa ja asettelee suhteellisen pääkerrososuuden Q_{MNF} johonkin ennaltamäärättyyn lukuun, esimerkiksi 1 neljän hissin joukolle ja sitten ohjelma poistuu kohdassa 596.

Kuvio 17

Kuvio 17 on vuokaavio alaohjelmasta LCD13, jota voidaan käyttää suorittamaan kuvion 9 lohkotoiminto 346, joka liittyy erityisiin kerrosominaisuuksiin. Kuten edellä kuvioon 6 liittyen selitettiin, esillä olevaan keksintöön sisältyy pääkerrosominaisuus ja kokouskerrosominaisuus, mutta muita kerrosominaisuuksia, kuten ravintolakerros ja tämänkaltaisia voidaan lisätä tavalla, joka edellä selitettiin kuvion 6 liittyen ja joka selitetään kuvioon 17 liittyen. Palautettakoon mieleen, että pääkerrosominaisuus aktivoidaan kytkimellä, joka ajaa kuvion 6 navan $\overline{PMNFL}$ todeksi. Pääkerrokseksi voidaan valita rakennuksen mikä kerros tahansa ja se voidaan haluttaessa muuttaa. Pääkerrokseksi valitun kerroksen binääriosoitte johdetaan napoihin $\overline{PMNFL0}$ - $\overline{PMNFL3}$ esimerkiksi useilla kytkimillä ja täten pääkerroksen aseman lataamiseksi on vain tarpeen johtaa uuden kerroksen asianomainen binääriosoitte näihin napoihin.

Samalla tavoin napa $\overline{PCONFL}$ kuviossa 6 aktivoi kokouskerrosominaisuuden ja navat $\overline{PCFL0}$ - $\overline{PCFL3}$ valitsevat kerroksen osoitteen, jonka kerroksena jälleen voi olla rakennuksen mikä kerros tahansa.

Pääkerrosominaisuus, aktivoituna, pyrkii pitämään alaohjelmassa LCD12 (kuvio 16) Q_{MNF} :llä asetellun suhteellisen osuuden koreista pääkerroksessa esittäen valekutsuja pääkerrokseen ja se aikaansaa NEXT-kori-ominaisuuden, jolla kori osoitetaan seuraavaksi koriksi jättämään pääkerroksen, joka kori odottaa pääkerroksessa, edullisesti ovet avoinna ja kerroskutsu-ylöspäin-lamppu sytytettynä, kunnes korikutsu on rekisteröity koriin. Seuraavaa koria (NEXT) käsitellään eri tavoin, kun osoitetaan aikavälejä korille, kuten jäljempänä selitetään liittyen alaohjelmaan, joka osoittaa aikavälit.

Kun kokouskerros-ominaisuus on aktivoitu eikä mitään koria ole valittu kokouskerroksessa, valekutsuja käytetään tuomaan kori kerrokseen. Kori, joka on parkeerattu kokouskerrokseen, on parkeerattuna ovet avoinna, kunnes on rekisteröity kutsu kokouskerroksessa.

Alaohjelma LCD13 alkaa kohdassa 600 ja vaihe 602 aloittaa tyhjäämällä kaikki valekutsut ($\overline{PKFL}$) asettamalla sanan FLOOR CPU 80 indeksirekisteriin osoitteella $\overline{PMNFL0}$ - $\overline{PMNFL3}$ osoitettuun kerrokseen, asettamalla pääkerroslipun ykköseksi, joka osoittaa, että käsitellään pääkerrosominaisuutta ja asettamalla väliaikaisen sanan ASGN 4-bittiseen sanaan NEXT, joka on varastoitu RAMin 4 päämuistiin.

Vaihe 604 tarkistaa PMNFLR todetakseen, onko pääkerrosominaisuus aktivoitu, ellei, PMNFLR on nolla ja ohjelma etenee vaiheeseen 610, joka tyhjästä sanat NEXT, DOPN ja SUT, jotka on varastoitu RAMiin 4, koska nämä osoitukset CPU 80:llä tehdään vain kun pääkerrosominaisuus on aktiivinen.

Jos PMNFLR on ykkönen, vaihe 606, ohjelmataarkistuksena, määrää, onko sana N_{SMF} , joka sisältää korien luvun, jotka on valtuutettu palvelemaan valittua pääkerrosta, yhtä suuri kuin nolla. Jos pääkerrososoite valitsee aikavälin, jota varten mikään kori ei ole valtuutettu, pääkerrosominaisuus on epäkelpo ja ohjelma etenee vaiheeseen 610. Jos sana N_{SMF} ei ole nolla, pätevä aikaväli on valittu ja vaihe 608 tarkistaa pääkerroksen suhteellisen osuuden Q_{MNF} , joka on asetettu LCD12:een (kuvio 16). Jos Q_{MNF} on nolla, ohjelma etenee vaiheeseen 610 tyhjästäkseen sanan NEXT, DOPN ja SUT. Jos suhteellinen pääkerrososuus ei ole nolla, ohjelma etenee vaiheeseen 612. Tässä silmukassa vaiheen 612 lävitse sana ASGN on sana NEXT, joka on asetettu vaiheessa 602, siten vaihe 612 tarkistaa sanan ASGN todetakseen, onko olemassa koria, joka on osoitettu seuraavaksi koriksi jättämään pääkerroksen. Jos sana ASGN on nolla, ei ole olemassa koria, joka on merkitty seuraavaksi (NEXT) koriksi jättämään pääkerroksen ja ohjelma etenee vaiheeseen 630. Jos on olemassa seuraava kori, vaihe 614 identifioi seuraavan korin. Vaihe 616 tarkistaa, onko kori kerroksessa, joka ohjelman tässä silmukassa viittaa pääkerrokseen, koska pääkerroslippu on ykkönen. Jos kori ei ole pääkerroksessa, vaihe 618 osoittaa valekutsun PKFL tälle korille pääkerrokseen.

Jos kori on tässä kerroksessa, vaihe 620 tarkistaa pääkerroslipun. Tässä silmukassa vaiheen 620 kautta pääkerroslippu on ykkönen ja ohjelma etenee vaiheeseen 624. Vaihe 624 tarkistaa kutsun testaamalla RAMissa 0 sanan $\overline{CALL}$ sen bitin, joka liittyy koriin, joka on identifioitu seuraavana. Jos tämä $\overline{CALL}$ bitti on 0, osoittaen, että seuraavalla korilla on kutsu, vaihe 626 tyhjästä osoitussanat NEXT, DOPN ja SUT salliakseen korin palvella kutsua. Jos sanan $\overline{CALL}$ tämä bitti on ykkönen, osoittaen ei-kutsua, vaihe 628 asettelee ovi avoinna bitin DOPN korille ja myös asettelee ylöspäinliikenne-bitin SUT korille.

Sen jälkeen kun seuraava (NEXT) kori pääkerroksessa vastaanottaa ovi- ja liikesuuntaosoituksensa vaiheessa 628, ohjelma etenee vaiheeseen 668, joka tarkistaa pääkerroslipun. Jos tämä on ykkönen, se osoittaa,

että kokouskerrosominaisuutta ei ole tarkistettu ja vaihe 670 asettaa sanan FLOOR kokouskerroksen osoitteeseen, joka on valittu $\overline{PCFL0}$ - $\overline{PCFL3}$:lla, se asettaa pääkerroslipun nolaksi ja se asettaa sanan ASGN sanaksi CONV, joka sana varastoidaan RAMiin 4.

Vaihe 672 tarkistaa PCONFL todetakseen, onko kokouskerrosominaisuus aktivoitu. Ellei, vaihe 676 tyhjä RAMiin 4 varastoidun sanan CONV ja ohjelma poistuu kohdassa 678. Jos kokouskerrosominaisuus on aktiivinen, vaihe 674 määrää, onko niiden korien luku $N_{SCF} = 0$, jotka ovat valtuutetut palvelemaan kokouskerrosta. Jos näin on, kokouskerrososoi- te on valinnut epäkelvon aikavälin ja vaihe 676 tyhjä kokouskerros- sanan CONV. Jos N_{SCF} on nolla, ohjelma suorittaa silmukan takaisin vaiheeseen 612.

Vaihe 612 tarkistaa, onko osoitussana ASGN suurempi kuin nolla, joka, tässä silmukassa, tarkistaa sanan CONV todetakseen, onko jollekin ko- rille annettu kokouskerrososoitus. Jos korille on annettu kokousker- rososoitus, vaihe 614 identifioi korin ja vaihe 616 tarkistaa, onko kori kokouskerroksessa. Ellei se ole kokouskerroksessa, vaihe 618 antaa valeparkeerauskutsun $\overline{PKFL}$ kokouskerrokseen tälle korille. Jos se on kerroksessa, vaihe 620 tarkistaa pääkerroslipun ja tässä silmu- kassa se on nolla, suunnaten ohjelman vaiheeseen 621, joka tarkistaa, onko tällä korilla kokouskerroksessa kutsu. Jos sillä ei ole kutsua, bitti $\overline{CALL}$ on ykkönen ja ohjelma etenee vaiheeseen 668, joka havait- see, että pääkerroslippu on nolla ja ohjelma poistuu kohdassa 678.

Jos sanan ASGM havaittiin olevan nolla kun tarkistettiin joko silmuk- kaa pääkerrosominaisuudelle tai silmukkaa kokouskerrosominaisuudelle, tämä merkitsee, että läsnä oleva ominaisuus, jota tarkistetaan silmu- kalla, on aktivoitu, mutta millään korilla ei tällä hetkellä ole osoi- tusta, so. NEXT osoitusta pääkerrossilmukalle tai CONV osoitusta ko- kouskerrossilmukalle. Tässä tapauksessa ohjelma etenee vaiheeseen 630, joka aloittaa ohjelman sen osan, joka paikantaa sopivan korin tällaista osoitusta varten. Ohjelma myös etenee vaiheeseen 630 vai- heesta 626 pääkerrossilmukan aikana, kun NEXT-korilla pääkerroksessa on kutsu ja täten täytyy löytää toinen kori NEXT-osoitusta varten. Samalla tavoin ohjelma etenee vaiheeseen 630 vaiheesta 622, kun ko- kouskerrossilmukassa korilla kokouskerroksessa, jolla korilla on CONV- osoitus, on korikutsu, koska tämän korin täytyy poistua kokouskerrok- sesta ja toinen kori on löydettävä CONV-osoitusta varten.

Vaihe 630 tarkistaa 4-bitin sanan AVAS katsoakseen, onko eri korien kerrosvalitsimien mukaan koreja joutilaina tai käytettävissä. Tämä sana varastoidaan RAMiin 0. Jos on olemassa käytettävissä oleva kori, sana AVAS ei ole nolla, ja ohjelma etenee vaiheeseen 632, joka käynnistää prosessin kysymyksessä olevaa kerrosta lähinnä olevan AVAS-korin löytämiseksi. Vaihe 632 aloittaa korilaskennan ja asettaa muutuvan DIST numeroon, joka on suurempi kuin pisin matkaväli rakennuksessa. Esimerkiksi kerrosluvun ollessa 16, DIST asetellaan lukuun 16.

Vaihe 634 tarkistaa RAMin 0 AVAS bitin ensimmäiselle korille korisilmukassa. Jos tämä kori ei ole AVAS, ohjelma etenee vaiheeseen 636, joka lisää korinumeroa ja jos korisilmukka on loppuun käsitelty, koestettuna vaiheella 648, ohjelma suorittaa silmukan takaisin vaiheeseen 634.

Jos vaihe 634 havaitsee, että kori on AVAS, vaihe 636 määrää onko kori valtuutettu (kelpuutettu) palvelemaan tätä kerrosta, tarkistamalla kerrosvaltuutuksen RAMissa 11. Jos kori ei ole valtuutettu tälle kerrokselle, ohjelma etenee vaiheeseen 646. Jos kori on valtuutettu, vaihe 638 tarkistaa tähän koriin liittyvän NEXT-sanan bitin todetakseen, onko korille annettu NEXT-osoitus. Jos se ei ole NEXT, vaihe 640 määrittää etäisyyden korista kysymyksessä olevaan kerrokseen käyttäen kerrosnumeroiden erotuksen itseisarvoa. Vaihe 642 tarkistaa onko tämä etäisyys pienempi kuin DIST ja koska tämä on ensimmäinen löydetty AVAS kori, se on lähempänä kuin DIST, koska DIST sijoitettiin vapaasti suurempaan lukuun kuin pisin liikeväli. Vaihe 644 lataa korinumeron väliaikaiseen kohtaan ja muuttaa sanan DIST välimatkaksi tästä korista kysymyksessä olevaan kerrokseen.

Vaihe 646 lisää korinumeroa ja 648 määrittää, ovatko kaikki korit tulleet käsitellyiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 634. Kun kaikki korit ovat tulleet käsitellyiksi, väliaikaiseen kohtaan varastoitu korinnumero on lähin kysymyksessä olevaa kerrosta oleva AVAS kori ja vaihe 650 muodostaa osoitussanan NEXT tai CONV siitä riippuen missä silmukassa ohjelma on sekä lähettäen valekutsun $\overline{PKFL}$ korille kysymyksessä olevaan kerrokseen.

Vaihe 652 tarkistaa pääkerroslipun ja jos se on ykkönen, sana NEXT ladataan RAMiin 4 ja jos se on nolla, vaihe 656 lataa sanan CONV RAMiin 4. Ohjelma etenee vaiheeseen 668, joka tarkistaa päälipun.

Jos se on ykkönen, kokouskerrosominaisuutta ei ole tarkistettu ja ohjelma etenee edellä aikaisemmin kuvattuun vaiheeseen 670. Jos se on nolla, ohjelma poistuu kohdassa 678.

Jos vaihe 630 ei löytänyt yhtään käytettävissä olevaa koria, ohjelma etenee vaiheeseen 658. Vaihe 658 tarkistaa huippuliikennebitin RAMin 0 tilamerkkimuistissa. Jos se on ykkönen, on olemassa huippuliikennetilaa ylöspäin tai alaspäin ja jos se on nolla, ei ole huippuliikennetilaa. Jos vaihe 658 havaitsee, että ei ole huippuliikennetilaa, vaihe 658 etenee vaiheeseen 662, joka antaa kaikille koreille valekutsun pää- tai kokouskerrokseen, riippuen siitä mitä ominaisuutta käsitellään. Vaihe 662 etenee vaiheeseen 668.

Jos on olemassa huippu, vaihe 658 etenee vaiheeseen 664, joka tarkistaa huippuliikenteen tyyppin. Jos se on alaspäinhuippu, mitään valekutsuja pääkerrokseen ei osoiteta, koska alaspäinhuipussa NEXT kori lähetetään välittömästi ja sen vuoksi ei ole tarpeen antaa NEXT-osoitusta. Ei myöskään millekään varattuna olevalle korille anneta kokouskerrososoitusta alaspäinhuipun aikana. Jos järjestelmä on huippuliikennetilassa ylöspäin, vaihe 666 tarkistaa pääkerroslipun. Jos se on nolla, ohjelma etenee vaiheeseen 668, koska mitään kokouskerrososoitusta ei tehdä varatuille koreille ylöspäinhuipun aikana. Jos pääkerroslippu on ykkönen, vaihe 662 antaa kaikille koreille valekutsun PKFL pääkerrokseen.

Kuvio 18

Kuvio 18 on vuokaavio alaohjelmasta LCD5, jota voidaan käyttää suorittamaan kuvion 9 lohko toiminto 348, joka tyhjä ylös- ja alasosoitustaulut, jotka on varastoitu RAMien 6 vast. 7 päämuisteihin, kaikista aikaväleistä ennaltamäärätyin poikkeuksin. Esimerkiksi LCD5:11ä säilytetään aikavälit, joita vain yksi kori on valtuutettu palvelemaan. Myös säilytetään aikavälit, joilla on rekisteröity kerrostasokutsu.

Alaohjelma LCD5 alkaa kohdassa 680 ja vaihe 682 aloittaa kerroslaskennan, se tyhjä N_{HCl} muuttujan niiden kerrostasokutsujen määrän laskemiseksi, jotka on osoitettu korille yhden korin ryhmässä, so. ryhmässä, jolle on valtuutettu vain yksi kori, se tyhjä N_{SS} muuttujan niiden aikavälien kokonaismäärän laskemiseksi, jotka on osoitettu korille tähän asti, se tyhjä korikohtaiset rekisterit, RAMit 12, 13, 14 ja 15, se asettaa ylöspäin lipun ykköseksi ja se lataa ylöspäin

kutsupeiteosoitteen (RAM 10) ja ylöspäin osoitusosoitteen (RAM 6).

Vaihe 684 tarkistaa ylöspäinkutsupeitesanan RAMin 10 aikavälistä 00 todetakseen, onko se nolla. Ellei, mikään kori ei ole valtuutettu tälle aikavälille, vaihe 696 tyhjä mahdollisen osoituksen (RAM 6) aikavälille ja ohjelma etenee vaiheeseen 702, joka lisää kerroslas-kentaa.

Jos ylöspäinkutsupeitesana ei ole nolla, se merkitsee, että kysymyksessä on kelvollinen aikaväli ja vaihe 686 määrittää peitesanasta, onko vain yksi kori valtuutettu palvelemaan aikaväliä. Jos näin on, vaihe 704 identifioi korin ja vaihe 706 tarkistaa RAMin 1 todetakseen, onko olemassa tähän aikaväliin liittyvää kerrostaokutsua. Jos on olemassa kerrostaokutsu, vaihe 708 lisää muuttujaa N_{HCl} tälle korille laskeakseen niiden kerrostaokutsujen määrän, jotka on osoitettu tälle korille 1 korin ryhmästä ja vaihe 710 lisää muuttujaa N_{SS} tälle korille laskeakseen niiden aikavälien kokonaismäärän, jotka tähän asti on osoitettu tälle korille. Jos ei ole olemassa kerrostaokutsua, vaihe 706 etenee suoraan vaiheeseen 710.

Koska mitkään muut korit eivät palvele tätä aikaväliä, korille voidaan välittömästi antaa aikaväliosoitus ja vaihe 702 lataa ylöspäinkutsupeitesanan RAMiin 6, koska ylöspäinkutsupeitesana yhden korin ryhmälle on sama kuin osoitussana. Sen jälkeen vaihe 712 etenee vaiheeseen 702.

Jos vaihe 686 havaitsee, että aikaväliä palvelee useampia kuin 1 kori, vaihe 688 tarkistaa RAMin 1 todetakseen, onko tähän aikaväliin liittyneenä kerrostaokutsu ylöspäin (1Z). Ellei, vaihe 696 tyhjä aikaväliosoituksen RAMissa 6 ja etenee vaiheeseen 702. Jos kerrostaokutsu on olemassa, vaihe 690 tarkistaa todetakseen, onko kutsuun liittyvä aikaväli aikaisemmin saanut osoitusta. Ellei, ohjelma etenee vaiheeseen 702. Jos aikavälille oli aikaisemmin annettu osoitus, vaihe 692 tarkistaa osoituksen (RAM 6) ylöspäinkutsupeitteellä (RAM 10) ja vaihe 694 määrittää, onko osoitus kelvollinen, so. onko kerrostaokutsu osoitettu korille, joka on valtuutettu kutsun aikaväliin. Jos osoitus ei ole kelvollinen, vaihe 696 tyhjä osoituksen. Jos osoitus on kelvollinen, vaihe 698 lisää muuttujaa N_{RCC} korille, so. niiden rekisteröityjen kerrostaokutsujen lukua, jotka on osoitettu korille ryhmästä, joka on valtuutettu useammalle kuin yhdelle korille. Muuttujat N_{RCC} koreille 0, 1, 2 ja 3 varastoidaan ohjelman LCD5 aikana

RAMien 12, 13, 14 ja 15 tilamerkkimuisteihin.

Vaihe 700 lataa osoituksen yhden RAMin 12, 13, 14 tai 15 päämuistiin siitä riippuen, mille korille aikaväli on osoitettu. Ainoat osoitukset, jotka esiintyvät korikohtaisissa rekistereissä (RAMit 12, 13, 14 ja 15), kun LCD5 on loppuun suoritettu, ovat niille ryhmille, jotka on valtuutettu useammalle kuin yhdelle korille. Aikavälit yhden korin ryhmille on suoraan osoitettu vaiheessa 712.

Vaihe 702 lisää kerroslaskentaa, vaihe 714 tarkistaa todetakseen, onko kerrossilmukka tullut loppuun suoritetuksi, suorittaen silmukan takaisin vaiheeseen 684, ellei näin ole tapahtunut ja edeten vaiheeseen 716, kun näin on tapahtunut.

Vaihe 716 tarkistaa ylöslipun. Jos se yhä on ykkönen, vaihe 718 asettaa ylöslipun nolllaksi ja lataa alaspäinkutsupeitteen osoitteen (RAM 9) ja alaspäinosoitusosoitteen (RAM 7) ja palaa vaiheeseen 684 käsittelemään alaspäin-aikaväliosoituksia.

Sen jälkeen kun alaspäin-aikaväliosoitukset on käsitelty, vaihe 716 toteaa, että ylöslippu on nolla ja ohjelma poistuu kohdassa 720.

Kuvio 19

Kuvio 19 on vuokaavio alaohjelmasta LCD6, jota voidaan käyttää suorittamaan kuvion 9 lohkon 350 toiminto, joka poistaa liiat aikaväliosoitukset koreista, jos näitä on, käyttäen opastuksena keskimääräistä kutsujen lukua koria kohden rakennuksessa A_{CB} .

Alaohjelma LCD6 tulee mukaan kohdassa 730 ja vaihe 732 aloittaa korilaskennan. Vaihe 734 tarkistaa sanan NEXT bitin, joka liittyy tähän koriin ja on varastoitu RAMiin 4 ja jos tällä korilla on NEXT osoitus, vaihe 738 tyhjä ne osoitukset tälle korille, jotka oli sijoitettu tähän koriin liittyvään korikohtaiseen rekisteriin, so. yhteen RAMEista 12 - 15. Palautettakoon muistiin, että LCD5 (kuvio 18) vain sijoitti korikohtaisiin rekistereihin osoitukset ryhmille, joita palveli useampia kuin yksi kori. Täten kerrossoituksia korikohtaisessa rekisterissä palvelevat muut korit ja ne osoitetaan uudelleen ohjelmassa LCD14, jos osoitus on poistettu ohjelmassa LCD5. Yhden korin ryhmien osoitukset sijoitettiin suoraan RAMEihin 6 ja 7 eikä niitä siten häiritä vaiheella 738. Vaihe 740 lisää (jatkaa) korilaskentaa.

Jos vaihe 734 toteaa, että kori ei ole NEXT, vaihe 736 määrittää, onko tälle korille yhden korin ryhmästä osoitettujen kerrostasokutsujen lukumäärä, yhteenlaskettuna N_{HCl} :ssä korille ohjelmassa LCD5, yhtä suuri tai suurempi kuin A_{CB} . Jos näin on, tällä korilla on kaikki, mitä se voi käsitellä kerroksista, joita palvellaan vain tällä korilla ja vaihe 738 poistaa mahdolliset aikaväliosoitukset tälle korille, jotka ovat korikohtaisissa rekistereissä.

Jos vaihe 736 toteaa, että kerrostasokutsujen lukumäärä, jotka on osoitettu korille yhden korin ryhmästä, N_{HCl} , ei ole sama tai suurempi kuin keskimäärä A_{CB} , vaihe 742 summaa kerrostasokutsut, jotka on osoitettu korille, lisäämällä N_{HCl} lukuun N_{RCC} . Luku N_{RCC} , joka kehitettiin kuviossa 18 esitettyssä vaiheessa 698, on kerrostasokutsujen lukumäärä, jotka on osoitettu korille ryhmistä, joita palvellaan useamalla kuin yhdellä korilla. Jos tämä summa ei ole sama tai ylitä rakennuksen kutsukeskimäärää koria kohden, A_{CB} , vaihe 744 asettaa muuttujan N_{HCT} korille lukujen N_{RCC} ja N_{HCl} summaksi ja ohjelma etenee vaiheeseen 740.

Jos lukujen N_{HCl} ja N_{RCC} summa on sama tai enemmän kuin A_{CB} , ohjelma starttaa korin aikavälistä ja edeten korista valittuun pyyhkäisysuuntaan, tarkistettuna bitillä sanassa UPSCAN, se laskee aikavälit, jotka on osoitettu korille. Kaikkiin aikaväleihin, jotka on osoitettu koreille korikohtaisissa rekistereissä, liittyy kerrostasokutsu. Täten kun on saavutettu laskentatila yhtä kuin A_{CB} , kaikki muut aikavälit, jotka kohdataan osoitettuna tälle korille, poistetaan korikohtaisista rekistereistä.

Pyyhkäisyjakson eri osille, jotka tutkivat aikavälejä, käynnistyen korista, annetaan pyyhkäisynumerot seuraavan koodin mukaisesti:

- Pyyhkäisy 1: Pyyhkäisy, joka alkaa korin asemasta ja etenee pyyhkäisyjakson yhteen päähän.
- Pyyhkäisy 2: Pyyhkäisy, joka vaihtaa suuntaa pyyhkäisyn 1 lopussa ja etenee pyyhkäisyjakson toiseen päähän.
- Pyyhkäisy 3: Pyyhkäisy, joka vaihtaa suuntaa pyyhkäisyn 2 lopussa ja etenee takaisin korin pyyhkäisyväliin.

Palatkaamme kuvioon 19. Kun summa N_{HCl} plus N_{RCC} on sama tai suurempi kuin A_{CB} , ohjelma etenee vaiheeseen 750, joka aloittaa pyyhkäisy-

numeron pyyhkäisylle 1. Vaihe 752 aloittaa aikaväliaseman ja laskee kerroslaskentaa korin kerrossaseman määräämiseksi. Vaihe 754 tarkistaa, onko kori päätekerroksessa. Jos näin on, tapahtuu vain kaksi pyyhkäisyä kolmen sijasta ja ohjelma etenee vaiheeseen 770 lisätäkseen pyyhkäisynumeroa. Jos kori ei ole päätekerroksessa, vaihe 756 määrää ensimmäisen huomioonotettavan aikavälin osoitteen (korin kerrosta so minus yksi) ja vaihe 758 toteaa, onko se osoitettu tarkasteltavalle korille. Ellei, vaihe 766 lisää kerroslaskentaa. Jos on, vaihe 760 määrittää, onko N_{HCl} , tälle korille osoitettujen kutsujen lukumäärä 1 korin ryhmästä, yhtä suuri tai suurempi kuin A_{CB} . Ellei, vaihe 762 lisää N_{HCl} ja vaihe 766 lisää kerroslaskentaa. Jos N_{HCl} on yhtä suuri tai suurempi kuin A_{CB} , vaihe 764 tyhjäää tämän aikavälin osoituksen tälle korille korikohtaisesta rekisteristä ja vaihe 766 lisää kerroslaskentaa.

Vaihe 768 tarkistaa, ovatko kaikki aikavälit nykyisessä pyyhkäisy-suunnassa tulleet tukityiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 756. Jos esillä oleva pyyhkäisy on loppuun suoritettu, vaihe 770 lisää pyyhkäisynumeroa ja vaihtaa pyyhkäisy-suuntaa. Vaihe 772 tarkistaa, onko pyyhkäisy-silmukka tullut loppuun suoritetuksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 752. Jos kaikki pyyhkäisyt on käsitelty, vaihe 772 etenee vaiheeseen 740, joka lisää korinumeron. Vaihe 746 tarkistaa, ovatko kaikki korit tulleet huomioonotetuiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 734. Jos kaikki korit ovat tulleet huomioonotetuiksi, ohjelma poistuu kohdassa 748.

Kuvio 20

Kuvio 20 on vuokaavio alaohjelmasta LCD7, jota voidaan käyttää suorittamaan lohkon 352 toiminto kuviossa 9, joka toiminto osoittaa pyyhkäisy-suunnat palveluksessa oleville joutilaille koreille, käytettäväksi, kun osoitetaan aikavälejä koreille toiminnossa 356 kuviossa 9, mikä yksityiskohtaisesti on esitetty ohjelmalla LCD14 kuviossa 22.

Kun liikevälirajoitus korista osoitettuun kerrokseen palvelusuuntaan on annettu kaikille palveluksessa oleville koreille olkootpa ne varattuja tai joutilaita, on tärkeää valita alkuperäinen osoitussuunta palveluksessa olevasta joutilaasta korista, joka ottaa huomioon varattujen korien liikesuunnan sekä kulloinkin esiintyvät liikenneolosuhteet. Jos kulkumatkarajoitus annetaan vain varatuille koreille, vähennetään sen seikan tärkeyttä, että valitaan osoitussuunta dyna-

misesti. Viimeksimainitussa tapauksessa voidaan käyttää palveluksessa olevan joutilaan korin viimeistä liikesuuntaa. Esimerkin vuoksi oletetaan, että käytetään ohjelmaa LCD7.

Aikavälit osoitetaan koreille LCD14:ssa käyttäen samaa pyyhkäisysilmukkaa kuin selitettiin kuvioon 19 liittyen. Varatut korit, so. korit, joilla on korikutsu edessä, valekutsu edessä taikka osoitettu kerrostasokutsu edessä, osoitetaan samaan pyyhkäisysuuntaan kuin on niiden liikesuunta. Palveluksessa oleva kori, jossa ei ole korikutsuja, valekutsuja taikka osoitettuja kutsuja, osoitetaan pyyhkäisysuuntaan, joka parhaiten tyydyttää seuraavan jaon, olettaen 4 korin joukon.

- (1) Huippuliikennetila ylöspäin: vain yksi kori palvelemaan alaspäinliikennettä
- (2) Huippuliikennetila alaspäin: vain yksi kori palvelemaan ylöspäin liikennettä
- (3) Normaali liikenne (ei huippua): puolet koreista kumpaankin palvelusuuntaan

Alaohjelma LCD7 alkaa kohdassa 780 ja vaihe 782 aloittaa korilaskennan ja asettaa RAMin 0 tilamerkkimuistiin varastoidun 4 bitin sanan UPSCAN sanaksi UPTR. Sana UPTR varastoidaan RAMin 0 päämuistiin. Vaihe 784 tarkistaa, onko palveluksessa olevia joutilaita koreja tarkistamalla sanan AVAS, joka on varastoitu RAMin 0 päämuistiin. Jos sana AVAS on nolla, ei ole AVAS koreja ja ohjelma poistuu kohdassa 828. On huomattava, että sanaa "käytettävissä", jota normaalisti käytetään tarkoittamaan "käytettävissä osoitusta varten" ei voida soveltaa prosessoritasolla, koska kaikille palveluksessa oleville koreille on annettu kerrososoitukset. Jos kerrososoituksilla ei ole kerrostasokutsua ja korilla ei ole korikutsuja eikä parkeerauskutsua, kori on joutilas eli epäaktiivinen, mutta se ei ole "käytettävissä".

Jos sana AVAS ei ole nolla, on olemassa ainakin yksi käytettävissä oleva kori kerrostasovalitsimelle ja vaihe 786 tekee oman päätelmänsä siitä, onko kori palveluksessa ja todella epäaktiivinen tai joutilas muodostaen sanan IDLE tähän koriin liittyvistä INSC, NEXT ja AVAS biteistä. Vaihe 788 tarkistaa, onko tämä sana IDLE nolla. Jos näin on, tämä osoittaa, että kori on palveluksessa, sillä ei ole NEXT osoitusta ja se on käytettävissä tämän korin kerrosvalitsimen mukaan. Jos

se ei ole nolla, vaihe 790 laskee korin varatukseksi ja etenee vaiheeseen 792.

Jos sana IDLE on nolla, ohjelma etenee suoraan vaiheeseen 792 vaiheesta 788. Vaihe 792 lataa sanan IDLE sen korikohtaisen rekisterin päämuistiin, joka liittyy koriin, so. RAMiin 12 korille 0 ja korilaskenta jatkuu. Vaihe 794 määrittää, ovatko kaikki korit tulleet huomioonotetuiksi ja ellei, ohjelma tekee silmukan takaisin vaiheeseen 786. Jos kaikki korit ovat tulleet käsitellyiksi, vaihe 796 tuottaa pyyhkäisysuuntien mielivaltaisen jakaantumien sijoittamalla muuttujan UPDES palveluksessa oleviin koreihin, joiden lukumäärä on N_{SC} miinus 1 ja muuttuja DNDES asetetaan yhteen. Kun alaohjelma edelleen etenee, muuttujat UPDES ja DNDES sisältävät halutun määrän koreja, jotka on asetettava ylös- ja alas-pyyhkäisysuuntiin.

Vaihe 798 tarkistaa huippuliikennebitin RAMin 0 tilamerkkimuistissa ja jos se ei ole asetettu, vaihe 800 lataa $1/2 N_{SC}$ UPDES:iin ja $1/2 N_{SC}$ DNDES:iin. Jos huippuliikennebitti on asetettu, vaihe 802 tarkistaa bitin tilamerkkimuistissa RAMissa 0, mikä identifioi, onko järjestelmä huippuliikenteessä ylöspäin tai alaspäin. Jos järjestelmä on huippuliikenteessä ylöspäin, mitään muuta ei tehdä UPDES:ille ja DNDES:ille, koska näiden muuttujien mielivaltainen asettaminen vaiheessa 796 asettaa ne huippuliikenteeseen ylöspäin. Jos järjestelmä on huippuliikenteessä alaspäin, vaihe 804 vaihtaa UPDES:in ja DNDES:in asettaen UPDES:in yhdeksi ja DNDES:in $N_{SC}-1$.

Sitten ohjelma etenee vaiheeseen 806, joka aloittaa korilaskennan ja vaihe 810 lataa tämän korin UPTR bitin akkuun. Vaihe 812 tarkistaa sanan IDLE, joka on varastoitu tämän korin korikohtaiseen rekisteriin, todetakseen, onko se käytettävissä järjestelmän ohjauslaitteen määrittelmän mukaan. Jos se ei ole käytettävissä, ohjelma etenee vaiheeseen 822. Jos se on käytettävissä, vaihe 814 määrittää, onko niiden korien todellinen määrä, jotka on asetettu alaspäin liikenteeseen, DNAC, yhtä suuri tai suurempi kuin haluttu alaspäin liikenteeseen asetettujen korien määrä DNDES. Jos vastaus on ei, vaihe 816 osoittaa korin alaspäin, vaihe 822 asettaa tähän koriin liittyvän sanan UPSCAN nol-laksi osoittamaan, että koriosoituspyyhkäisy on alaspäin suuntaan ja korilaskentaa jatketaan.

Jos niiden korien todellinen määrä, jotka on asetettu alaspäinpyyh-

käisyyn, on yhtä suuri tai suurempi kuin haluttu määrä, vaihe 818 määrittää niiden korien todellisen määrän UPAC, jotka on asetettu liikenteeseen ylöspäin, todetakseen, onko tämä yhtä suuri tai suurempi kuin haluttu määrä UPDES. Jos vastaus on ei, vaihe 820 osoittaa korin ylöspäin suuntaan ja vaihe 822 asettaa tähän koriin liittyvän bitin sanassa UPSCAN ykköseksi osoittamaan, että se on osoitettu ylöspäin suuntaan ja jatkaa korilaskentaa.

Jos vaihe 818 havaitsee, että UPAC on yhtä suuri tai suurempi kuin UPDES, ohjelma etenee vaiheeseen 822, UPSCAN bitti on häiritsemätön ja korilaskenta jatkuu.

Vaihe 824 tarkistaa, ovatko kaikki korit tulleet huomioonotetuiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 810. Jos kaikki korit on käsitelty, vaihe 826 lataa sanan UPSCAN RAMin 0 tilamerkkimuistiin ja ohjelma poistuu kohdassa 828.

Kuvio 21

Kuvio 21 on vuokaavio alaohjelmasta LCD8, jota voidaan käyttää toimintoon 354 kuviossa 9, joka toiminto osoittaa järjestyksen, jossa korit käsitellään, kun aikavälit osoitetaan niille kuvion 9 vaiheessa 356.

Alaohjelma LCD8 alkaa kohdassa 830 ja vaihe 832 tyhjä RAMien 4, 5, 6 ja 7 tilamerkkimuistit niihin varastoiduista korikutsulaskennoista. Vaihe 834 tarkistaa kaikki kutsut koreille ensimmäisessä aikavälissä käyttäen RAMin 3 päämuistin ensimmäistä 4 bitin sanaa, johon muistiin korikutsut 3Z on varastoitu. Jos korikutsu todetaan korille, se lisätään korin korikutsusummaan. Vaihe 836 jatkaa kerroslaskentaa ja vaihe 838 tarkistaa, ovatko kaikki kerrokset tulleet käsitellyiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 834. Jos ne kaikki ovat tulleet käsitellyiksi, vaihe 840 lisää niiden korikutsujen määrän, joka kullakin korilla on, korille osoitettujen kerrokskutsujen määrään ja summat varastoidaan väliaikaiseen kohtaan.

Vaihe 842 sitten aloittaa korilaskennan ja vaihe 844 määrittää, onko korilla NEXT osoitus tutkimalla RAMin 4 päämuistin NEXT sanan bitin, joka liittyy tähän koriin. Jos kori on NEXT, vaihe 846 lisää tähän koriin liittyvään kori- ja kerrostasokutsujen kokonaismäärään mielivaltaisen määrän kutsuja, jonka mielivaltaisen luvun suuruus on riittävä varmistamaan sen, että NEXT korilla on suurempi luku kuin mitä

millään muulla korilla on mahdollista olla.

Vaihe 848 tarkistaa, onko moottorigeneraattoriyksikkö, joka liittyy hissikorin käyttömoottoriin, katkaistuna. Tämä toteutetaan tarkistamalla RAMin 0 päämuistiin varastoidun sanan $\overline{D89T}$ bitti. Jos $\overline{D89T}$ bitti on nolla, mikä osoittaa, että moottorigeneraattori on katkaistuna, vaihe 850 lisää ylimääräisiä kutsuja kori- ja kerrostasokutsusummaan tälle korille, ylimääräisten kutsujen määrä valitaan siten, että korilla on suurin numero, jos ei ole mitään koria, jolla on NEXT osoitus ja toiseksi suurin numero siinä tapauksessa, että on olemassa kori, jolla on NEXT osoitus.

Vaihe 852 lisää korilaskentaa ja vaihe 854 tarkistaa, ovatko kaikki korit tulleet käsitellyiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 844. Jos kaikki korit ovat tulleet käsitellyiksi, ohjelma etenee vaiheeseen 856.

Vaiheet 856-876 järjestävät korit ohjelman LCD8 aikaisemmassa osassa juuri valmistettujen numeroiden suuruuden mukaisesti siten, että järjestyksessä ensimmäisellä korilla on pienin lukumäärä kutsuja jne. Mitä tahansa lajittelu- tai järjestämistekniikka voidaan käyttää. Kuviossa 21 kuvattu tekniikka alkaa korien ollessa ennaltamäärätyssä järjestyksessä, esimerkiksi järjestyksessä 0, 1, 2 ja 3, käyttäen korien numeroita ja vertaa koreja pari kerrallaan, vaihtaen korien paikat aina kun oikealla puolella olevaan koriin liittyvien kutsujen määrä on pienempi.

Neljän korin joukossa koreilla on olemassa neljä asemaa ja näille neljälle asemalle annetaan numerot 1, 2, 3 ja 4 aloittaen vasemmanpuoleisesta asemasta ja on huomattava, että sijaintinumero ei liity korinumeroon. Käyttäen sijaintinumeroa olisi neljän korin joukon vertailujono taulukossa I esitetty:

TAULUKKO I

Vaiheet	Vertailu	
	Sijainti	Sijainti
1	1	2
2	1	3
3	1	4
4	2	3
5	2	4
6	3	4

Taulukon I tekniikka otetaan käyttöön aloittamalla vaiheella 856. Vaihe 856 lataa niiden korien laskennat, jotka sijaitsevat ensimmäisessä ja toisessa asemassa aloittaen taulukon vaiheen 1. Vaihe 858 tarkistaa kutsulaskentojen merkittävimmät bitit ja vaihe 860 tarkistaa, ovatko ne yhtä suuret. Ellei, enempi vertailu ei ole tarpeen ja ohjelma etenee vaiheeseen 864, joka kysyy, onko ensimmäinen kutsulaskenta yhtä suuri tai pienempi kuin toinen kutsulaskenta. Jos vaihe 860 havaitsee, että merkittävimmät bitit ovat yhtä suuret, vaihe 862 vertaa alempia bittejä ja sitten etenee vaiheeseen 864.

Jos vaihe 864 havaitsee, että ensimmäinen laskenta (loppusumma) ei ole pienempi tai yhtä suuri kuin toinen laskenta, vaihe 866 vaihtaa korinumerot ja niiden korilaskennat, siirtäen toisessa asemassa olevan korin luvun ensimmäiseen asemaan ja ensimmäisessä asemassa olevan korin luvun toiseen asemaan. Jos ensimmäinen korilaskenta on yhtä suuri tai pienempi kuin toinen, koriluvut ovat oikeassa järjestyksessä sikäli kuin on kysymys tästä parista ja vaihe 864 etenee vaiheeseen 868, johon vaihe 866 etenee vaihdettuaan korinumerot.

Vaihe 868 lisää toisen aseman asemanumeron, joka on vaihe 2 taulukossa I, verratakseen asemassa 1 olevan korin kutsulaskentaa asemassa 3 olevan korin kutsulaskentaan. Vaihe 870 tarkistaa onko ensimmäisessä asemassa oleva kori tullut verratuksi kaikkiin muihin koreihin ja ellei, ohjelma tekee silmukan takaisin vaiheeseen 858. Täten ohjelma tekee silmukan takaisin suorittaakseen taulukon I vaiheet 2 ja 3, ja sitten vaihe 870 havaitsee, että silmukka on loppuun suoritettu ja ohjelma etenee vaiheeseen 872.

Vaihe 872 lisää ensimmäisen aseman asemanumeroa, so. vaihtaa 1:n 2:een ja myös lataa tämän numeron (2) toiseen asemaan. Sitten vaihe 874 lisää toisen aseman numeroa aikaansaadakseen numeron 3. Täten vaiheen 874 jälkeen asemissa 2 ja 3 olevien korien kutsulaskennat (loppusummat) ovat tulleet verratuiksi, joka on taulukon vaihe 4.

Vaihe 876 tarkistaa onko vertailun tämä toinen vaihe tullut loppuun suoritetuksi ja koska näin ei ole tapahtunut, ohjelma tekee silmukan takaisin vaiheeseen 858 tehdäkseen taulukon I vaiheen 4 vertailun. Saavutettaessa vaihe 868 toisen aseman laskentaa lisätään asemissa 2 ja 4 olevien korien vertaamiseksi, joka on vaihe 5 taulukossa I ja ohjelma tekee silmukan takaisin vaiheesta 870 vaiheeseen 858 tämän vertailun tekemiseksi.

Vaihe 870 toteaisi, että vertailun toinen vaihe on tullut loppuun suoritetuksi, vaihe 872 lisääisi ensimmäisen aseman asemanumeroa siirtääkseen sen eteenpäin kolmeksi ja numero 3 ladattaisiin toiseen asemaan. Vaihe 874 lisää toisen aseman numeroa tehden sen 4:ksi ja täten korit asemassa 3 ja 4 ovat valmiit vertailtavaksi, joka on vaihe 6 taulukossa I. Ohjelma tekee silmukan takaisin vaiheeseen 858 tämän vertailun tekemiseksi ja etenee vaiheiden 870 ja 876 "kyllä" haarojen lävitse, koska kolmannessa vaiheessa on vain yksi vertailu.

Vaihe 878 lataa järjestetyt korinumerot RAMien 4, 5, 6 ja 7 tilamerkki-muisteihin.

Taulukko II sisältää esimerkin edellä selitetystä lajittelutekniikasta, kun korilla 0 on kutsumäärä 4, korilla 1 kutsumäärä 9, korilla 2 kutsumäärä 7 ja korilla 3 kutsumäärä 3.

TAULUKKO II

<u>Asemat</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>
Korien lähtöjärjestys (kori n:o)	0	1	2	3
Vaihe 1 (1-2)	0	1	2	3
Vaihe 2 (1-3)	0	1	2	3
Vaihe 3 (1-4)	3	1	2	0
Vaihe 4 (2-3)	3	2	1	0
Vaihe 5 (2-4)	3	0	1	2
Vaihe 6 (3-4)	3	0	2	1

Kuvio 22

Kuvio 22 on vuokaavio alaohjelmasta LCD14, jota voidaan käyttää toimintoon 356 kuviossa 9, joka toiminto osoittaa aikavälit koreille. Aikavälit osoitetaan kolmessa ohiajossa kullekin ryhmälle, kunkin ajon käsittäessä kaikki ryhmät ennen seuraavan ajon alkua. Ryhmät käsitellään ryhmän korilukumäärän nousevassa järjestyksessä ja kussakin ryhmässä pyyhkäistävien (skannattavien) korien järjestys määrätään ohjelmassa LCD8 (kuvio 21).

Alaohjelma LCD14 alkaa kohdassa 890 ja vaihe 892 lataa korikutsut RAMista 3 korikohtaisten rekistereiden (RAMit 12 - 15) päämuisteihin. Vaihe 893 tarkistaa onko A_{CB} , keskimääräinen kerrostasokutsujen määrä rakennuksessa palveluksessa olevaa koria kohden, yhtä suuri tai suurempi kuin ennaltamäärätty minimiluku. Tämän luvun suuruus määrää,

milloin joutilaat (IDLE) korit otetaan palvelukseen, kun liikenne alkaa kasvaa rakennuksessa. Jos halutaan, että kahden kerrostasokutsun on käynnistettävä kaksi koria, minimiluku voi olla aseteltu nol-laksi. Minimiluvun asettaminen kahdeksi vaatii 3 kerrostasokutsua samalle korille ennenkuin toinen kori käynnistetään jne.

Jos A_{CB} ei ole yhtä suuri tai suurempi kuin minimiluku, vaihe 894 asettaa sen samaksi kuin tämä minimiluku ja ohjelma etenee vaiheeseen 895. Jos A_{CB} on yhtä suuri tai suurempi kuin minimi, vaihe 893 ete-nee vaiheeseen 895.

Vaihe 895 aloittaa ajolaskennan osoittamisen aloittaen osoituksesta ajolle 1. Vaihe 896 aloittaa ryhmälaskennan, niin että ryhmät otetaan ryhmien korilukujen nousevassa järjestyksessä. Kuten edellä mainittiin, ryhmänumerot ovat binäärilukuja, jotka on tuotettu ylös- ja alaspeit-teissä, RAMEissa 10 vast. 9, loogisilla ykkösillä kussakin rivissä, joka liittyy kerrostasoon, jota kukin kori on valtuutettu palvelemaan. Jos kori ei ole valtuutettu, sen bittiasema kerrokselle on looginen nolla. Vaihe 898 kutsuu ensimmäistä käsiteltävää ryhmää noutoinstruk-tiolla, joka pääsee ohjausmuistin 82 hakutaulukkoon (kuvio 4). Binää-rilaskin, joka on asetettu laskemaan 4:stä 15:een, kutsuu aina 12 ryhmää, tämän laskimen siirtäessä kutsun aina seuraavaan ryhmään. Alaohjelma LCD5 (kuvio 18) on jo tehnyt osoitukset 1 korin ryhmille mainitun ohjelman vaiheessa 712, joka vähentää LCD14:ssa käsiteltä-vien ryhmien maksimimäärän 16:sta 12:een.

Vaihe 900 tarkistaa, onko kutsuttu ryhmä pätevä ryhmä, koska kaikki mahdolliset useampikoriset ryhmänumerot tutkitaan. Tämä toteutetaan tarkistamalla onko A_{SI} , keskimääräinen aikavälien määrä ryhmässä pal-veluksessa olevaa koria kohden ryhmän kohdalla nolla. Jos näin on, se on epäpätevä ryhmä ja ohjelma etenee vaiheeseen 978 ryhmälaskennan jatkamiseksi. Jos se on kelvollinen ryhmä, A_{SI} ei ole nolla ja vaihe 902 lataa tämän ryhmän peitteen korikohtaisten rekistereiden (RAMit 12 - 15) päämuistiin. Ryhmän peite paljastaa ryhmän kerrokset, so. looginen ykkönen sijaitsee ryhmän jokaisessa kerroksessa, joka vastaa kutakin koria, joka voi palvella ryhmää ja kaikki muut bittiasemat ovat arvossa looginen nolla.

Vaihe 904 aloittaa korilaskennan ja lataa 4 bittiset sanat INSV ja UPSCAN, jotka on varastoitu RAMiin 0, väliaikaiseen paikkaan. Vaihe

906 tarkistaa INSV bitin ensimmäiselle kysymyksessä olevalle korille ja jos kori ei ole palveluksessa, ohjelma etenee vaiheeseen 974, joka jatkaa korilaskentaa. Jos kori on palveluksessa, vaihe 908 tarkistaa, onko kori valtuutettu tälle ryhmälle. Ellei, peite korikohtaisessa rekisterissä on nolla tälle korille ja ohjelma etenee vaiheeseen 974.

Jos kori on ryhmässä, ohjelma alkaa ensimmäisen osoitusajon vaiheella 910. Vaihe 910 tarkistaa, onko tälle korille annettu NEXT osoitus. Jos näin on, vaihe 914 antaa tälle korille ylöspäin-aikaväli-osoituksen, ja jos kerrostasovalitsimien mukaan on olemassa käytettävissä olevia koreja, tarkistettuna vaiheessa 916, laskematta koreja, joilla on NEXT tai CONV osoitukset, NEXT korille ei anneta muita osoituksia ja ohjelma etenee vaiheeseen 974. Jos sana AVAS on nolla, mikä osoittaa, että kerrostasovalitsimien mukaan ei ole käytettävissä olevia koreja, NEXT korille voidaan antaa lisäosoituksia, ja ohjelma etenee vaiheeseen 918.

Jos kori ei ollut NEXT, vaihe 912 määrittää, onko tämä ensimmäinen osoitusajo. Jos näin on, korin AVAS bitti tarkistetaan vaiheessa 918 sen toteamiseksi, onko kori käytettävissä sen kerrosvalitsimen mukaan. Jos se on käytettävissä, vaihe 920 osoittaa tälle korille ylös- ja alas-aikavälit, jotka liittyvät kerrokseen, jossa kori sijaitsee ja ohjelma etenee vaiheeseen 922. Jos kori ei ole käytettävissä, ohjelma etenee suoraan vaiheeseen 922.

Vaihe 922 määrittää, onko korille annettu kokouskerrososoitus tarkistamalla asianomaisen bitin sanassa CONV. Jos tämä bitti on ykkönen, vaihe 924 osoittaa tälle korille ylös- ja alas-aikavälit, jotka liittyvät kokouskerrokseen. Jos CONV bitti ei ole ykkönen, ohjelma etenee vaiheeseen 926, joka aloittaa aikavälilaskennan ja tyhjää muuttujat N_{DIST} , N_{SI} ja N_{CI} . Aikavälilaskenta kolmeen pyyhkäisyyn 1, 2 ja 3 nähden selitettiin edellä ohjelmaan LCD6 liittyen (kuvio 19). Muuttujaa N_{DIST} käytetään laskemaan pätevät aikavälit, joihin laskenta- ja osoitusjono on edennyt korista tähän asti osoitusrutiinissa. Muuttujaa N_{SI} käytetään laskemaan korille tähän asti osoitettujen aikavälien määrää kysymyksessä olevassa ryhmässä.

Vaihe 928 määrittää parametrit pyyhkäisylle, so. luvun, joka on vähennettävä ylöspäin tai alaspäin kulkevan korin kerrostasosta, niin että aikaväliosoite voidaan määrätä ja vaihe 930 vähentää parametrin

pyyhkäisystä aikaväliosoitteen määrittämiseksi. Kolme aikaväliosoitetta ylöspäin liikkuvalla korille, jotka käynnistävät pyyhkäisyt korin edellä tapahtuvaa pyyhkäisyä varten, pyyhkäisylle korin liikesuunnalle vastakkaiseen suuntaan ja pyyhkäisylle korin jäljessä, ovat N_{CP-1} , $\overline{N_{CP-1}}$ ja vast. $N_{CP} - \overline{N_{POS+1}}$, missä N_{CP} on sillä tavoin aloitettu laskenta, että loppusumma on 15, kun laskin on lisännyt yhdellä kutakin kerrosta kohden korista ja N_{POS} on aikavälinumero, joka vastaa korin sijaintia. Kolme aikaväliosoitetta alaspäin liikkuvalla korille, jotka käynnistävät pyyhkäisyt pyyhkäisylle korin edellä, pyyhkäisylle korin liikesuunnalle vastakkaiseen suuntaan ja pyyhkäisylle korin jäljessä, ovat $\overline{N_{CP-1}}$, N_{CP-1} ja $\overline{N_{CP-(N_{POS+1})}}$.

Ohjelma osoittaa aikavälit AVAS koreille ilman rajoitusta matkaetäisyyteen korista kerrokseen, joka liittyy osoitettuun aikaväliin. Ohjelma rajoittaa kuitenkin aikavälien osoituksen varattuihin koreihin perustuen aikavälin etäisyyteen korista kerrokseen ja palvelusuuntaan käyttäen korin nykyistä matkasuuntaa eikä korin fysikaalista etäisyyttä aikaväliin liittyvästä kerroksesta. Esimerkiksi 16 kerroksisessa rakennuksessa ylöspäin liikkuva kori kolmannessa kerroksessa on ekvivalenttinen 27 kerroksen kanssa alaspäin kutsusta toisessa kerroksessa samalla kun fysikaalinen etäisyys on 1 kerros. Esimerkkinä mainittakoon, että etäisyysrajoitus, jota käytetään osoitettaessa aikavälejä, on puolet korin edestakaisesta matkasta. Tämä lasketaan mukavasti vähentämällä alin kerros, jota kori on valtuutettu palvelemaan, ylimmästä.

Vaihe 932 jatkaa N_{DIST} laskentaa ja vaihe 934 määrittää, onko aikaväli valtuutettu tarkistamalla ryhmäpeitteen. Vaihe 936 tarkistaa korin AVAS bitin RAMissa 0. Jos kori on käytettävissä, AVAS bitti on ykkösen ja korille ei aseteta puolen edestakaisen matkan rajoitusta. Jos kori ei ole käytettävissä, vaihe 938 määrittää, onko N_{DIST} vähemmän tai yhtä suuri kuin puolet korin edestakaisesta matkasta. Kuten edellä mainittiin, puolet korin edestakaisesta matkasta määritetään vähentämällä alin kerrostaso, jota kori on valtuutettu palvelemaan, ylimmästä kerrostasosta, jota kori on valtuutettu palvelemaan. Jos rakennuksessa on 16 kerrostasoa ja kori on valtuutettu kaikkiin kerroksiin, puoli edestakaistamatkaa olisi 15 kerrosta. Jos vaihe 938 havaitsee, että N_{DIST} on suurempi kuin puolet edestakaisesta matkasta, ohjelma etenee vaiheeseen 974. Jos N_{DIST} on yhtä suuri tai pienempi kuin puolet edestakaisesta matkasta, vaihe 940 tarkistaa, onko aikaväli jo

tullut osoitetuksi. Jos näin on, ohjelma etenee vaiheeseen 966, joka jatkaa aikavälilaskentaa. Jos aikaväliä ei ole osoitettu, vaihe 942 määrittää, onko tämä ensimmäinen ajo. Jos näin on, vaihe 944 tarkistaa, onko korilla rekisteröityä korikutsua. Ellei, ohjelma etenee vaiheeseen 966 jatkaakseen aikavälilaskentaa. Jos osoitusrutiini on ensimmäisessä ajossa ja korilla on korikutsu taikka jos osoitusrutiini ei ole ensimmäisessä ajossa, ohjelma etenee vaiheeseen 946, joka tarkistaa, onko aikavälille rekisteröityä kerrostasokutsua. Jos näin on, vaihe 948 määrittää, onko N_{HCT} , tälle korille tähän asti osoitettujen kerrostasokutsujen kokonaismäärä plus yksi vähemmän tai yhtä suuri kuin A_{CB} , kerrostasokutsujen keskimäärä koria kohden rakennuksessa. Jos N_{HCT} plus yksi on suurempi kuin A_{CB} , ohjelma etenee vaiheeseen 966. Jos N_{HCT} plus yksi on yhtä suuri tai pienempi kuin A_{CB} , vaihe 950 tarkistaa, onko pyyhkäisy kolmannessa ajossa. Ellei, vaihe 952 tarkistaa, onko N_{CI} plus yksi vähemmän tai yhtä suuri kuin A_{CI} , missä N_{CI} on kerrostasokutsujen lukumäärä, jotka on osoitettu korille tähän asti kysymyksessä olevassa ryhmässä ja A_{CI} on kutsujen keskimäärä palveluksessa olevaa koria kohden kysymyksessä olevassa ryhmässä. Jos N_{CI} plus 1 on suurempi kuin A_{CI} , ohjelma etenee vaiheeseen 966. Jos N_{CI} plus yksi on yhtä suuri tai pienempi kuin A_{CI} , ohjelma etenee vaiheeseen 954. Jos vaihe 950 toteaa, että osoitus on kolmannessa ajossa, vaiheen 952 rajoitus hypätään ylitse ja ohjelma menee suoraan vaiheeseen 954. Vaihe 954 lisää lukuja N_{CI} ja N_{HCT} ja etenee vaiheeseen 962. Vaihe 962 lisää muuttujia N_{SI} ja N_{SS} ja vaihe 964 osoittaa aikavälin korille.

Jos vaihe 946 toteaa, että aikavälissä ei ole kerrostasokutsua, ohjelma etenee vaiheeseen 956. Vaihe 956 tarkistaa, onko osoitus kolmannessa ajossa. Ellei, ohjelma etenee vaiheeseen 958, joka määrittää, onko N_{SI} plus yksi yhtä suuri tai pienempi kuin A_{SI} . Muuttuja N_{SI} on aikavälien luku, jotka on tähän asti osoitettu korille kysymyksessä olevasta ryhmästä ja A_{SI} on keskimääräinen aikavälien lukumäärä palveluksessa olevaa koria kohden kysymyksessä olevassa ryhmässä. Jos N_{SI} plus yksi on suurempi kuin A_{SI} , ohjelma etenee vaiheeseen 966. Jos N_{SI} plus 1 on yhtä suuri tai pienempi kuin A_{SI} , vaihe 960 tarkistaa, onko N_{SS} plus 1 pienempi tai yhtä suuri kuin A_{SB} . Muuttuja N_{SS} on sama kuin aikavälien kokonaismäärä, jotka on osoitettu korille tähän asti ja A_{SB} on aikavälien keskimäärä palveluksessa olevaa koria kohden rakennuksessa. Jos N_{SS} plus 1 on suurempi kuin A_{SB} , ohjelma etenee vaiheeseen 966. Jos se on yhtä suuri tai pienempi kuin A_{SB} , ohjelma etenee vaiheeseen 962, joka lisää N_{SI} :tä

ja N_{SS} :ää ja vaihe 964 osoittaa aikavälin korille. Jos vaihe 956 toteaa, että osoitus on kolmannessa ajossa, vaiheiden 958 ja 960 rajoitukset hypätään ylitse ja ohjelma etenee suoraan vaiheeseen 962.

Ohjelma etenee vaiheeseen 966, joka lisää aikavälilaskentaa. Vaihe 968 tarkistaa, onko pyyhkäisylukumäärä tullut loppuun suoritetuksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 930. Jos kaikki pyyhkäisylukuun liittyvät aikavälit ovat tulleet loppuun käsiteltyiksi, vaihe 970 lisää pyyhkäisylaskentaa ja pyyhkäisysuuntaa vaihdetaan. Vaihe 972 tarkistaa, ovatko pyyhkäisylaskennan kaikki kolme vaihetta (1, 2 ja 3) tulleet loppuun suoritetuiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 928. Jos pyyhkäisylaskenta on tullut loppuun suoritetuksi, ohjelma etenee vaiheeseen 974, joka lisää korilaskentaa ja siirtää UPSCAN ja INSV sanat paljastaakseen bitit, jotka liittyvät seuraavaan käsiteltävään koriin. Vaihe 976 määrittää, onko korilaskenta tullut loppuun suoritetuksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 906. Jos se on tullut loppuun suoritetuksi, ohjelma etenee vaiheeseen 978, joka lisää ryhmälaskentaa kut-suakseen seuraavan ryhmän. Vaihe 980 tarkistaa, ovatko kaikki ryhmät tulleet huomioon otetuiksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 898. Jos kaikki ryhmät on käsitelty, ohjelma etenee vaiheeseen 982, joka lisää osoitusajolaskentaa. Vaihe 984 tarkistaa, onko ajosilmukka tullut loppuun käsiteltyksi. Ellei, ohjelma tekee silmukan takaisin vaiheeseen 895. Jos ajosilmukka on tullut loppuun suoritetuksi, ohjelma poistuu kohdassa 986. Kolme osoitusajoa voidaan yhteenvetona kuvata seuraavasti:

Ensimmäinen ajo

NEXT korille annetaan pääkerros-ylöspäin-osoitus (vaihe 914). AVAS ja CONV koreille osoitetaan ylöspäin ja alaspäin aikavälit, jotka liittyvät kerrokseen, jossa AVAS kori sijaitsee ja vast. kokouskerrokseen (vaiheet 920 ja 924). Jos korilla on korikutsu kerrokseen, joka liittyy kysymyksessä olevaan aikaväliin, aikaväli osoitetaan korille, ennalta määrättyin rajoituksin. Vaihe 938 tuo mukaan puolen edestakaisen matkan rajoituksen varatuille koreille ja vaihe 946 valitsee muut sovellettavat rajoitukset riippuen siitä, onko kysymyksessä olevaan aikaväliin liittyneenä kerrostaokutsu vaiko ei. Ellei sillä ole kerrostaokutsua, keskimääriä A_{SI} (vaihe 958) ja A_{SB} (vaihe 960) käytetään rajoituksina. Jos sillä on kerrostaokutsu, keskimääriä A_{CB} (vaihe 948) ja A_{CI} (vaihe 952) käytetään rajoituksina. Jos koril-

la ei ole korikutsua kysymyksessä olevalla aikavälillä, aikaväliä ei osoiteta tässä ajossa.

Toinen ajo

NEXT korille annetaan pääkerros-ylöspäin-osoitus (vaihe 914). Tämä vaihe toistetaan vaikkakin se sisällytettiin ensimmäiseen ajoon jotta vaihe 916 voitaisiin tarkistaa jokaisessa kolmessa ajossa, koska on toivottavaa poistaa NEXT kori osoitusrutiinista niin pian kuin järjestelmässä on käytettävissä oleva kori.

Vaiheet 918, 920, 922 ja 924, jotka koskevat AVAS ja CONV koreja, jätetään pois toisesta ajosta, koska ne toteutettiin ensimmäisessä ajossa.

Toinen ajo hyppää yli myös vaiheen 944, joka oli aktiivinen ensimmäisessä ajossa, koska toinen ajo käsittelee osoittamattomia aikavälejä siitä riippumatta, onko korilla korikutsua aikavälin kerrokseen. Puolen edestakaisen matkan rajoitusta varatuille koreille ja keskimääriä A_{SI} , A_{SB} , A_{CB} ja A_{CI} sovelletaan siten kuin selitettiin ensimmäiseen ajoon nähden.

Kolmas ajo

NEXT korilla annetaan jälleen pääkerros-ylöspäin-osoitus syistä, jotka osoitettiin toisen ajon suhteen. Myös samalla tavoin kuin toisessa ajossa hypätään yli vaiheet 918, 920, 922, 924 ja 944.

Kolmannessa ajossa osoittamattomat (vapaat) ja tyhjät (ei kerrostasokutsua) aikavälit osoitetaan koreille, joita koskee vain puolen edestakaisen matkan rajoitus varatuille koreille, koska A_{SI} ja A_{SB} rajoitukset, jotka ovat aktiivisia vaiheissa 958 ja 960 hypätään ylitse.

Jos aikaväli on osoittamaton mutta siinä on kerrostasokutsu, kolmatta ajoa koskee vain puolen edestakaisen matkan rajoitus varatuille koreille ja A_{CB} rajoitus, koska A_{CI} rajoitus, joka on aktiivinen vaiheissa 952, hypätään ylitse.

Täten jos on olemassa palveluksessa olevia joutilaita koreja, kaikki aikavälit, jotka liittyvät kerroksiin, saavat osoituksen. Jos ei ole olemassa palveluksessa olevia joutilaita koreja, on mahdollista, että tietyllä ajolla läpi ohjelman tämä tai useampia aikavälejä, jotka liittyvät kerroksiin, jäävät ilman osoitusta johtuen matkaetä-

syysrajoituksesta aikavälejä osoitettaessa. Nämä aikavälit osoitetaan niin pian kun jokin kori liikkuu asemaan, joka tyydyttää ohjelman vaatimukset aikavälien osoittamiseksi. Koska mikään kori ei sijaitse sopivasti nopeasti vastatakseen kutsuun, joka liittyy osoittamattomaan aikaväliin, ei hyödyttäisi osoittaa aikaväliä tai -välejä kunnes on määrätty, mille korille olisi osoitettava aikavälit ohjelman strategian mukaan.

Kuvio 23

Kuvio 23 on vuokaavio alaohjelmasta LCD3, jota voidaan käyttää toimintoon 358 kuviossa 9, joka toiminto lataa RAMEihin 4, 5, 6 ja 7 (kuvio 5) varastoidun informaation RAMin 1 ulostuloon (kuvio 4) lähettääkseen aikaväliosoitukset ja käskyt koreille. Alaohjelma LCD3 alkaa kohdassa 990, vaihe 992 aloittaa RAM muistiosoitteen ja RAM ulostulo-osoitteen ja vaihe 994 aloittaa kerroslaskennan. Vaiheet 996 ja 998 synkronoivat pyyhkäisyjakson alun kanssa, käyttäen signaalia $\overline{MXCT}$, kuten edellä selitettiin vaiheisiin 364 ja 366 nähden kuviossa 10. Vaihe 1000 lukee RAMit 4, 5, 6 ja 7 RAMin 1 ulostuloon, yhden kerroksen kerrallaan, vaiheiden 1002 ja 1004 palauttaessa ohjelman vaiheeseen 1000 lukemaan seuraavaa kerrosta koskevan informaation. Kun kaikki kerrokset ovat tulleet loppuun käsitellyiksi, vaihe 1004 etenee poistumiskohtaan 1006.

Kuviot 24 ja 25

Kuviot 24 ja 25 ovat kaavioita, joita käytetään kuvaamaan keksinnön mukaista strategiaa tietyn esimerkin suhteen. Kuten kuviossa 24 on esitetty, rakennuksessa on 16 kerrosta, joita palvelee neljä koria 0, 1, 2 ja 3. Rakennuksessa on pääkerros 1, kaksi kellarikerrosta B1 ja B2 ja kaksi huippulisäkerrosta TE1 ja TE2. Kori 0 on valtuutettu molempiin kellarikerrokseen B1 ja B2 ja kerrokseen 1-12. Korit 2 ja 3 on valtuutettu kerrokseen 1-12 ja molempiin huippulisäkerrokseen TE1 ja TE2. Kori 1 on valtuutettu kellarikerrokseen B1 ja kerrokseen 1-12. Pätevät ryhmät määrätään alaspäin ja ylöspäin kutsupeitteillä, RAMEilla 9 ja 10. On olemassa kaksi aikaväliä yhden korin ryhmässä 0001. On olemassa kaksi aikaväliä kahden korin ryhmässä 0011. On olemassa neljä aikaväliä kahden korin ryhmässä 1100 ja on olemassa 22 aikaväliä 4 korin ryhmässä 1111. On olemassa 2 aikaväliä pätemättömyydessä ryhmässä 0000. Kaikki muut ryhmät ovat tyhjiä. Taulukko III esittää ryhmät ja kuhunkin ryhmään liittyvien aikavälien lukumäärät ja esittää myös A_{SI} ja A_{CI} kullekin ryhmälle. Keskimäärä A_{CI} on laskettu käyttäen taulukossa olevia kerrostaokutsujen määriä.

TAULUKKO III

Ryhmä	Kerrostasokutsuja	Aikavälejä	A_{SI}	A_{CI}
0001	1	2	2	1
0011	0	2	1	0
1100	3	4	2	2
1111	4	22	6	1
0000	X	2(PÄTEMÄTÖN)	X	X
32 YHTEENSÄ				

Korit ovat ympyröillä osoitetuissa asemissa, korilla 2 on NEXT osoitus pääkerroksessa. Korikutsuja on osoitettu merkillä CC. Kerrostaokutsuja on osoitettu kiilakuviolla sarakeotsikon kerrostaokutsut (Hall Calls) alla. Toiminto 332 kuviossa 9, jota yksityiskohtaisesti on kuvattu ohjelmalla LCD11 kuviossa 14, määrittää keskimäärän A_{SB} rakennukselle ja keskimäärät A_{SI} ryhmille. Keskimäärä A_{SB} on 8, so. 30 pätevää aikaväliä jaettuna neljällä palveluksessa olevalla korilla. Keskimäärät A_{SI} määrätään jakamalla aikavälien lukumäärä ryhmässä ryhmälle valtuutettujen palveluksessa olevien korien lukumäärällä. Ne on luetteloitu taulukossa III ja varastoitu oikeaan ryhmä-asemaan RAMissa 8 kuviossa 24. On huomattava, että kun osamäärä on murtoluku, käytetään lähinnä suurempaa kokonaislukua.

Toiminto 342 kuviossa 9, yksityiskohdittain LCD11 kuviossa 14, määrää keskimäärän A_{CB} rakennukselle ja keskimäärät A_{CI} ryhmille. Keskimäärä A_{CB} on 2, 8 kerrostaokutsua jaettuna 4 palveluksessa olevalla korilla. Keskimäärät A_{CI} määrätään jakamalla kerrostaokutsujen määrä ryhmässä niiden palveluksessa olevien korien luvulla, jotka on valtuutettu palvelemaan ryhmää. Myös ne on luetteloitu taulukossa III ja ne on varastoitu oikeaan ryhmä-asemaan RAMissa 2 kuviossa 24.

Taulukko IV auttaa muistamaan keskimäärät ja rajoitukset, joita sovelletaan kolmeen osoitusajoon.

TAULUKKO IV

Osoitusajo	A_{SB}	A_{SI}	A_{CB}	A_{CI}	Puolen edestakaisen matkan rajoitus
1	kyllä	kyllä	kyllä	kyllä	kyllä
2	kyllä	kyllä	kyllä	kyllä	kyllä
3	ei	ei	kyllä	ei	kyllä

Ensimmäisessä osoitusajossa korille 2, jolla on NEXT osoitus, annetaan pääkerros-ylöspäin-osoitus, jota on osoitettu merkillä X kuvion 24 ylöspäin-osoitustaulukossa. Oletetaan, että ei ole olemassa AVAS koreja, niin että NEXT kori otetaan huomioon seuraavia osoituksia varten, mutta se on viimeinen prioriteettijärjestyksessä. Oletetaan myös, että korien prioriteettijärjestys, jonka määrää LCD8 kuviossa 21, on 1, 3, 0, 2. LCD14 (kuvio 22) vaihe 944 valitsee erilleen ne aikavälit, joille koreilla on rekisteröidyt korikutsut. Korilla A on korikutsu yhdeksänteen kerrokseen ja koska se on asetettu liikkeeseen ylöspäin, sille osoitetaan aikaväli 10-UP, joka liittyy yhdeksänteen kerrokseen. Ydeksännellä kerroksella on rekisteröity kerrostasokutsu ylöspäin, niin että tämä osoitus automaattisesti ottaa korin, jolla on tämä samanaikainen kutsu.

Korilla 3 on korikutsu kahdenteentoista kerrokseen ja koska se on asetettu liikkeeseen ylöspäin, korille 3 osoitetaan aikaväli 13-UP, joka liittyy kahdenteentoista kerrokseen.

Korilla 0 on korikutsu pääkerrokseen ja koska se on asetettu liikkeeseen alaspäin ja on valtuutettu kulkemaan pääkerroksen alapuolelle, sille osoitetaan aikaväli 02-DN, joka liittyy pääkerrokseen. Pääkerros-alas on osa ryhmää 0011, jossa A_{SI} on 1. Sen vuoksi tämä osoitus korille 0 täyttää A_{SI} vaatimuksen ryhmälle 0011 korille 0.

Korilla 2 ei ole korikutsuja ja se ei vastaanota muita osoituksia ensimmäisen ajon aikana. Täten ensimmäinen ajo on loppuun suoritettu osoittaen pääkerros-ylöspäin aikavälin NEXT korille ja aikavälit, jotka liittyvät rekisteröityihin korikutsuihin ja niiden korien liikesuuntiin, joilla on korikutsuja. On muistettava, että LCD14 osoittaa aikavälit vain niille ryhmille, jotka ovat valtuutetut useammalle kuin yhdelle korille, koska LCD5, kuvio 18 on jo osoittanut yhden korin ryhmiin liittyvät korit, so. aikavälit 00-UP ja 01-DN oli jo aikaisemmin osoitettu korille 0.

Toisessa ajossa on korien prioriteettijärjestys yhä 1, 3, 0, 2. Koska ei ole AVAS koreja, pyyhkäisy-suunta aikavälien osoittamiseksi koreista on sama kuin korin liikesuunta.

Ajo 2 ottaa ensin ryhmän 0011. Aikaväli 02-DN on jo osoitettu korille 0, siten aikaväli 01-UP osoitetaan korille 1. Tämä täydentää kahden aikavälin osoittamisen ryhmästä 0011.

Sitten otetaan ryhmä 1100 ja kori 3, jolle osoitettiin aikaväli 13-UP ensimmäisessä ajossa, saa nyt osoituksen aikavälistä 14-UP, joka täyttää ryhmäkeskimäärän A_{SI} on 2 ja korille 2, toiselle korille, joka on valtuutettu tälle kahden korin ryhmälle, osoitetaan aikavälit 15-DN ja 14-DN. Aikavälin 14-UP osoittaminen korille 3 huolehtii kerrostasokutsusta ylöspäin TE1:ssä ja aikavälin 15-DN osoittaminen korille 2 huolehtii kerrostasokutsusta alaspäin TE2:ssa.

Nyt otetaan ryhmä 1111 ja korille 1 osoitetaan aikavälit 09-UP, 11-UP, 12-UP, 12-DN ja 11-DN. Aikaisemmalla osoituksella 10-UP korille 1 on kerrostasokutsu, joka täyttää ryhmän kutsukeskiarvon A_{CI} 1 ryhmälle 1111. Täten aikaväliä 13-DN ei osoiteta korille 1, koska sillä on rekisteröity kerrostasokutsu. Osoittaminen pysähtyy aikaväliin 11-DN, koska tämä täyttää ryhmäkeskimäärän A_{SI} 6.

Nyt osoitetaan kori 3, lähtien korista suuntaan ylöspäin. Aikaväli 12-UP oli aikaisemmin osoitettu korille 1. Täten ensimmäinen aikaväli, joka osoitetaan korille 3 tässä ryhmässä on 13-DN. Koska tässä aikavälissä on kerrostasokutsu, tämä täyttää keskiarvon A_{CI} on 1 ja keskimäärän A_{CB} on 2 ja vain aikavälit, joissa ei ole kutsuja, osoitetaan tälle korille toisen ajon loppuosan aikana. Aikavälit 12-DN ja 11-DN oli jo aikaisemmin osoitettu korille 1, niin että seuraava aikaväli, joka osoitetaan korille 3 on 10-DN. Aikavälit 09-DN ja 08-DN hypätään ylitse, koska niillä on kerrostasokutsut ja aikavälit 07-DN, 06-DN, 05-DN ja 04-DN osoitetaan korille 3. Tämä täyttää keskimääräehdon A_{SI} kuusi tälle ryhmälle ja keskimääräehdon A_{SB} kahdeksan. Kaikki nämä aikavälit sijaitsevat puolen edestakaisen matkan rajoituksen sisällä.

Nyt osoitetaan kori 0, lähtien korista suuntaan alaspäin. Aikaväli 04-DN on jo osoitettu korille 3, niin että aikaväli 03-DN on ensimmäinen osoitettava. Seuraava korille 0 osoitettu aikaväli on 03-UP. Koska aikaväliin 03-UP liittyy kerrostasokutsu, tämä täyttää ryhmäkutsukeskimäärän A_{CI} on yksi korille 0 ja keskimäärän A_{CB} on kaksi ja vain ilman kerrostasokutsuja olevat aikavälit osoitetaan nyt korille 0 tästä ryhmästä tässä ajossa. Täten aikavälit 04-UP, 05-UP ja 06-UP osoitetaan korille 0, mikä täyttää rakennuksen keskimäärän A_{SB} on kahdeksan ja ryhmäkeskimäärän A_{SI} on 6 ja osoitukset ovat puolen edestakaisen matkan rajoituksen sisällä.

Nyt osoitetaan korille 2 aikavälit ryhmästä 1111, lähtien korista ja edeten ylöspäin. Ensimmäinen vapaa (osoittamaton) aikaväli ylöspäin

tässä ryhmässä on 07-UP ja täten aikavälit 07-UP ja 08-UP osoitetaan korille 2. Seuraava vapaa aikaväli on 09-DN, mutta tämä on puolen edestakaisen matkan rajoituksen ulkopuolella eikä sitä osoiteta korille 2. Tämä saattaa loppuun toisen ajon ja kaikki muut aikavälit on osoitettu, paitsi 09-DN ja 08-DN ja molemmissa on rekisteröity kerrostasokutsu.

Kolmas ajo osoittaa vapaat aikavälit, poistaen kaikki toisessa ajossa käytetyt rajoitukset paitsi rakennuksen kutsukeskimäärän A_{CB} ja puolen edestakaisen matkan rajoitusta. Korit otetaan samassa järjestyksessä alkaen korilla 1. Korille 1 on osoitettu vain yksi kutsu, niin että sille osoitetaan aikaväli 09-DN. Tämä aikaväli on puolen edestakaisen matkan rajoituksen sisällä ja täyttää rakennuksen kutsukeskiarvon A_{CB} on kaksi tälle korille. Sen vuoksi tälle korille ei voida osoittaa aikaväliä 08-DN. Sitten käsitellään kori 3. Korille 3 on jo osoitettu kaksi kerrostasokutsua, täyttäen A_{CB} on 2, ja siten sille ei osoiteta aikaväliä 08-DN.

Kori 2 käsitellään viimeiseksi ja koska aikaväli 08-DN on sen puolen edestakaisen matkan rajoituksen ulkopuolella, sitä ei osoiteta korille 2. Täten aikaväliä 08-DN ei osoiteta ohjelman tässä ajossa.

Kuvio 25 on kaavio, joka kuvaa estosignaaleja, joita järjestelmän ohjauslaite 22 tuottaa kuvion 24 esittämässä esimerkissä.

Kun edellä esitetty selitys kuvaa keksinnön edullista suoritusmuotoa, on ymmärrettävä, että voidaan käyttää tiettyjä vaihtoehtoisia järjestyksiä ja että nämä ovat keksinnön puitteiden sisäpuolella. Esimerkiksi edullisessa suoritusmuodossa käytetään "sil-mukkapyyhkäisyä" osoitettaessa aikavälejä koreille, mihin sisältyy kolme osoitusajoa. Sil-mukkapyyhkäisy, joka alkaa korista liikesuuntaan ja palaa koriasemaan, on edullinen, koska se mahdollistaa samalla tavoin numeroitujen ryhmien ryhmittämisen siitä riippumatta, mihin palvelusuuntaan kerroksen binaarisana on liittynyt. Olisi kuitenkin myös sopivaa säilyttää palvelusuunnan erottaminen ja käyttää "ylös" ryhmiä ja "alas"-ryhmiä. Sil-mukkapyyhkäisyä ei käytettäisi tässä tapauksessa, koska ylös-ryhmät osoitettaisiin pyyhkäisemällä ylöspäin ja alasryhmät osoitettaisiin pyyhkäisemällä alaspäin.

Edelleen edullisessa suoritusmuodossa, yleinen osoittaminen osoittaa aikavälit valitulle korille kunnes tavataan yksi dynaamisista rajoitavista keskimääristä A_{SI} tai A_{SB} taikka liikematkarajoitus varatulle korille. Olisi myös sopivaa osoittaa aikaväli yhdelle korille kerrallaan, edeten korista koriin, kunnes kukin kori saavuttaa dynaamisen rajoittavan keskimäärän taikka liikevälimatkarakojituksen.

Patenttivaatimukset

1. Hissijärjestelmä, joka on yleisnätevä rakennuksille, jossa on useita kerroksia, johon järjestelmään kuuluu joukko hissikoreja (0,1,2,3, kuv. 1), joista kukin on asennettu liikkuvaksi kerrokseen nähden, ylöspäin- ja alaspäin-kerroistasokutsujen rekisteröintivälineet (62,64,66,kuv. 1) hissipalvelukutsujen rekisteröimiseksi ylöspäin- ja alaspäin-suuntiin ainakin tietyistä kerroksista, korinohjausvälineet (14,16,18,20), jotka toimittavat valtuutussignaaleja jokaista hissikoria varten, sekä valvontaohjauslaite (22), joka on riippuvainen mainituista ylöspäin- ja alaspäin-kerroistasokutsujen rekisteröintivälineistä, t u n n e t t u siitä,
- että kukin korinohjausväline (14,16,18,20) tuottaa ROM-muistin uralta valtuutussignaaleja, jotka valtuuttavat hissikorin palvelemaan kerrosvyöhykettä, joka voi myös sisältää ei-vierekkäisiä kerroksia, ja jotka osoittavat kerroistasoja sekä niistä lähteviä palvelusuuntia, joita kyseistä hissikoria ohjataan palvelemaan, kun kerros- tai hissikutsuja rekisteröidään tällaiseen palveluun,
 - että valvontaohjauslaite (22) on riippuvainen kunkin korinohjausvälineen tuottamasta valtuutussignaalista, jolloin valvontaohjauslaitteeseen kuuluvat muistielimet (RAM 2,3,10,11, kuv. 5) valtuutussignaalien varastoimiseksi siten, että saadaan rakennuksen kunkin hetken rakenneinformaatio siitä, mitkä korit ovat palveluksessa ja mitä kerroksia sekä palvelusuuntia niistä kyseiset hissikorit voivat palvella, ja
 - että valvontaohjauslaite sarjamuodossa kunkin kerroksen ylöspäin- ja alaspäin-palvelusuuntien perusteella, osoittaa aikaväleissä, joilla on kullakin sama kesto kutakin palvelusuuntaa varten, niiden kerroistasokutsujen käskyt, jotka on rekisteröity ylöspäin- ja alaspäin-kerroistasokutsujen rekisteröintivälineisiin (62,64,66) määrätyille koreille (0,1,2,3), muistielimiin varastoidun rakennusta koskevan informaation perusteella sekä sen perusteella, mikä on rekisteröityjen ylöspäin- ja alaspäin-kerroistasokutsujen keskilukumäärä sekä aikavälien lukumäärä rakennuksessa palveluksessa olevaa hissikoria kohti, sen määrittämiseksi, mitkä kerrokset ja palvelusuunnat esiintyvät rakenteessa, sekä mitä kerroistasoja sekä kerroistasojen palvelusuuntia kukin hissikori pystyy palvelemaan.

2. Patenttivaatimuksen 1 mukainen hissijärjestelmä, t u n - n e t t u siitä, että valvontaohjauslaitteeseen sisältyy ensimmäiset välineet /kuvio 5 (RAMit 2,9,10,11)7, jotka riippuvaisina aktivointisignaaleista ryhmittävät saman hissikoriyhdistelmän palvelemat kerrokset sekä näiden palvelusuunnat ryhmäksi, toiset välineet (kuvio 22), jotka määräävät kullekin ryhmälle ainakin yhden ennaltamäärätyn rajoituksen siitä, miten ryhmään kuuluvan hissikorin on palveltava kutsua ryhmän kerroksesta, ja kolmannet välineet /kuvio 5 (RAMit 6,7)7, jotka tehokkaasti osoittavat hissipalvelua vaativat kutsut mainitun hissikorijoukon hisseille toimittaen signaaleja /kuvio 8A (UPIN, DNIN)7 hissikorijoukon korinohjausvälineille mainittujen toisten välineiden mukaisesti.

3. Patenttivaatimuksen 2 mukainen hissijärjestelmä, t u n - n e t t u siitä, että siihen sisältyy keskiarvovälineet /kuviot 9, 14(342)7, jotka muodostavat ensimmäisen keskiarvon riippuvaisena rekisteröityjen ylöspäin- ja alaspäin-kerrostasokutsujen lukumäärästä rakennuksessa ja palveluksessa olevaa hissikoria kohden, jolloin mainittu ensimmäinen keskiarvo on rajoituksena mainituille toisille välineille määrättäessä minkä hissikorin on palveltava kerrostasoa.

4. Patenttivaatimuksen 2 tai 3 mukainen hissijärjestelmä, t u n n e t t u siitä, että muistivälineisiin sisältyy ensimmäinen ja toinen muistiosa, jotka reagoivat korinohjausvälineistä tuleviin signaaleihin, jolloin ensimmäinen muistiosa (RAM 9) varastoi binäärisignaalin kutakin kerrosta, kutakin koria, yhtä palvelusuuntaa varten ja toinen muistiosa (RAM 10) varastoi binäärisignaalin kutakin kerrosta, kutakin koria, toista palvelusuuntaa varten, sekä edelleen osoitteella varustettavat varastointiasemat ryhmiä koskevan informaation varastoimiseksi, jolloin binäärisana, joka on varastoitu muistivälineiden ensimmäiseen ja toiseen osaan, on osoite sille ryhmälle, johon tämä sana liittyy osoitteella varustettavassa varastointiasemassa.

5. Patenttivaatimuksen 3 mukainen hissijärjestelmä, t u n - n e t t u siitä, että keskiarvon muodostamisvälineet jaksottaisesti muodostavat toisen keskiarvon (A_{CB}) riippuen hissipalvelu-

kutsujen lukumäärästä ja ryhmän palvelusuunta-keskiarvon (A_{CI}) kullekin ryhmälle, jota palvelee useampi kuin yksi hissikori, riippuen kerrostasojen ja siitä lähtevien palvelusuuntien keskilukumäärästä ryhmässä hissikoria kohti, joka on valtuutettu palvelemaan ryhmää ja että valvontaohjauslaite lisäksi on riippuvainen ryhmän palvelusuunta-keskiarvosta jakaessaan kerroksia ja palvelusuuntia näistä kussakin ryhmässä niiden hissikorien kesken, jotka ovat valtuutetut palvelemaan ryhmää.

6. Patenttivaatimuksen 5 mukainen hissijärjestelmä, t u n - n e t t u siitä, että keskiarvon muodostamisvälineet jaksotaisesti muodostavat ryhmän kutsukeskiarvon (A_{SB}) ainakin eräille ryhmille riippuvaisesti ylöspäin- ja alaspäin-kutsujen keskimääräisestä lukumäärästä ryhmässä, ryhmää palvelevaa hissikoria kohti ja että valvontaohjauslaitteet lisäksi ovat riippuvaiset ensimmäisestä keskiarvosta sekä ryhmän kutsukeskiarvosta jakaessaan kerroksia ja palvelusuuntia näistä ryhmän hissikorien kesken.

7. Patenttivaatimuksen 6 mukainen hissijärjestelmä, t u n - n e t t u siitä, että valvontaohjauslaite jaksottaisesti tyhjä [kuvio 9 (348, 350); kuviot 18, 19] tietyt kerrostasopalvelusuunta-osoitukset hissikoreille ja uudelleen osoittaa (kuvio 22) nämä kerrostasot ja palvelusuunnat perustuen viimeiseen informaatioon, jonka keskiarvon muodostamiselimet ovat muodostaneet.

Patentkrav

1. Hissystem vilket är allmängiltigt för byggnader omfattande ett flertal våningar, vilket system innefattar ett antal hisskorgar (0,1,2,3, fig. 1), av vilka var och en är monterad att röra sig i relation till våningarna, registreringsorgan (62,64, 66, fig. 1) för våningsnivåanrop uppåt och nedåt i och för registrering av hissbetjäningsanrop i uppgående och nedgående riktning från åtminstone vissa av våningarna, korgstyrningsanordningar (14,16,18,20), vilka levererar fullmäktigande signaler för var och en hisskorg, samt en styrövervakningsanordning (22), vilken är beroende av nämnda registreringsorgan för uppåt och nedåt riktade våningsnivåanrop, k ä n n e t e c k n a t av
- att vart och ett korgstyrningsorgan (14,16,18,20) producerar fullmaktssignaler från ett ROM-minnesspår, vilka signaler befullmäktigar hisskorgen att betjäna en våningszon som även kan innefatta icke intill varandra belägna våningar, och vilka anger våningsnivåerna samt de från dessa utgående betjäningsriktningarna, vilka ifrågavarande hisskorg styrs att betjäna då vånings- eller hissanrop registreras för dylik betjäning,
 - att styrövervakningsanordningen (22) är beroende av den av vart och ett korgstyrningsorgan producerade fullmaktsignalen, varvid styrövervakningsanordningen omfattar minnesorgan (RAM 2,9,10,11, fig. 5) för lagring av fullmaktssignalerna sålunda, att det vid var och en tidpunkt erhålls en strukturinformation över byggnaden om vilka korgar som är i tjänst och vilka våningar samt betjäningsriktningar utgående från dessa kan betjänas genom ifrågavarande hisskorgar, och
 - att styrövervakningsanordningen i serieform på basen av de uppåt- och nedåtgående betjäningsriktningarna för var och en våning inom tidsintervall, vilka vart och ett omfattar samma längd för var och en betjäningsriktning, anvisar de våningsnivåbefallningar, vilka registrerats i registreringsorganen (62,64,66) för våningsnivåanrop uppåt och nedåt till bestämda korgar (0,1,2,3) på basen av de i minnesorganen lagrade informationerna samt på basen av vad som utgör medeltalet för de registrerade uppåt- och nedåtgående våningsnivåanropen samt antalet tidsintervall per hisskorg som är i tjänst i byggnaden, för bestämning av vilka våningar och betjäningsriktningar som

uppträder i byggnaden samt vilka våningsnivåer och betjäningsriktningar från våningsnivåerna var och en hisskorg förmår betjäna.

2. Hissystem enligt patentkravet 1, k ä n n e t e c k n a t av att styrövervakningsanordningen omfattar första organ /Figur 5 (RAM 2,9,10,11)7, vilka beroende av aktiveringssignaler grupperar samma hisskorgskombination att betjäna våningarna och dessas betjäningsriktningar i en grupp, andra organ (figur 22), vilka för var och en grupp bestämmer åtminstone en på förhand bestämd begränsning i det avseendet hur hisskorgen bör betjäna ett anrop från en våning inom gruppen, och tredje organ /Figur 5 (RAM 6,7)7, vilka effektivt visar de hissbetjäning anfordrande anropen till hissarna hos nämnda antal hisskorgar under levererande av signaler /Figur 8A (UPIN, DNIN)7 till korgstyrningsorganen för grupperna av hisskorgar i enlighet med nämnda andra organ.

3. Hissystem enligt patentkravet 2, k ä n n e t e c k n a t av att det innefattar medelvärdesorgan /Figurerna 9,14(342)7, vilka bildar ett första medelvärde i beroende av antalet registrerade uppåt- och nedåtgående våningsnivåanrop i byggnaden per i tjänst varande hisskorg, varvid nämnda första medelvärde utgör en begränsning för nämnda andra organ vid bestämning av vilken hisskorg bör betjäna våningsnivån.

4. Hissystem enligt patentkravet 2 eller 3, k ä n n e t e c k n a t av att minnesorganen omfattar en första och andra minnesdel, vilka reagerar på de från korgstyrningsorganen kommande signalerna, varvid den första minnesdelen (RAM 9) upplagrar en binärsignal för var och en våning, var och en korg, för en betjäningsriktning och den andra minnesdelen (RAM 10) upplagrar en binärsignal för var och en våning, var och en korg, för en andra betjäningsriktning, samt vidare adressförsedda upplagringsstationer för upplagring av information som berör gruppen, varvid ett binärord som upplagrats i minnesorganens första och andra del utgör en adress för den grupp som detta ord tillhör i den med adress försedda upplagringsstationen.

5. Hissystem enligt patentkravet 3, k ä n n e t e c k n a t av att organen för bildande av medelvärde periodvis bildar ett andra medeltal (A_{CB}) beroende på hissbetjäningsanropens antal och gruppens betjäningsriktningsmedeltal (A_{CI}) för var och en grupp som betjänas av flere än en hisskorg, beroende på våningsnivåernas och de från dessa utgående betjäningsriktningarnas medelvärdesantal i gruppen per hisskorg, vilken är befullmäktigad att betjäna gruppen, och att styrövervakningsanordningen därtill är beroende av gruppens betjäningsriktningsmedeltal vid fördelande av våningar och betjäningsriktningar utgående från dessa i var och en grupp mellan de hisskorgar som är befullmäktigade att betjäna gruppen.

6. Hissystem enligt patentkravet 5, k ä n n e t e c k n a t av att organen för bildande av medelvärde periodiskt bildar gruppens anropsmedelvärde (A_{SB}) för åtminstone vissa grupper beroende på det genomsnittliga antalet uppåt- och nedåtgående anrop i gruppen, per hisskorg som betjänar gruppen, och att styrövervakningsanordningen därtill är beroende av det första medelvärde samt gruppens anropsmedelvärde vid fördelande av våningar och betjäningsriktningar från dessa mellan gruppens hisskorgar.

7. Hissystem enligt patentkravet 6, k ä n n e t e c k n a t av att styrövervakningsanordningen periodiskt tömmer /figur 9 (348,350); figurerna 18,19 bestämde våningsnivå-betjäningsriktningsanvisningar till hisskorgarna och på nytt adresserar (figur 22) dessa våningsnivåer och betjäningsriktningar på basen av den senaste informationen som bildats av organen för bildande av medeltalet.

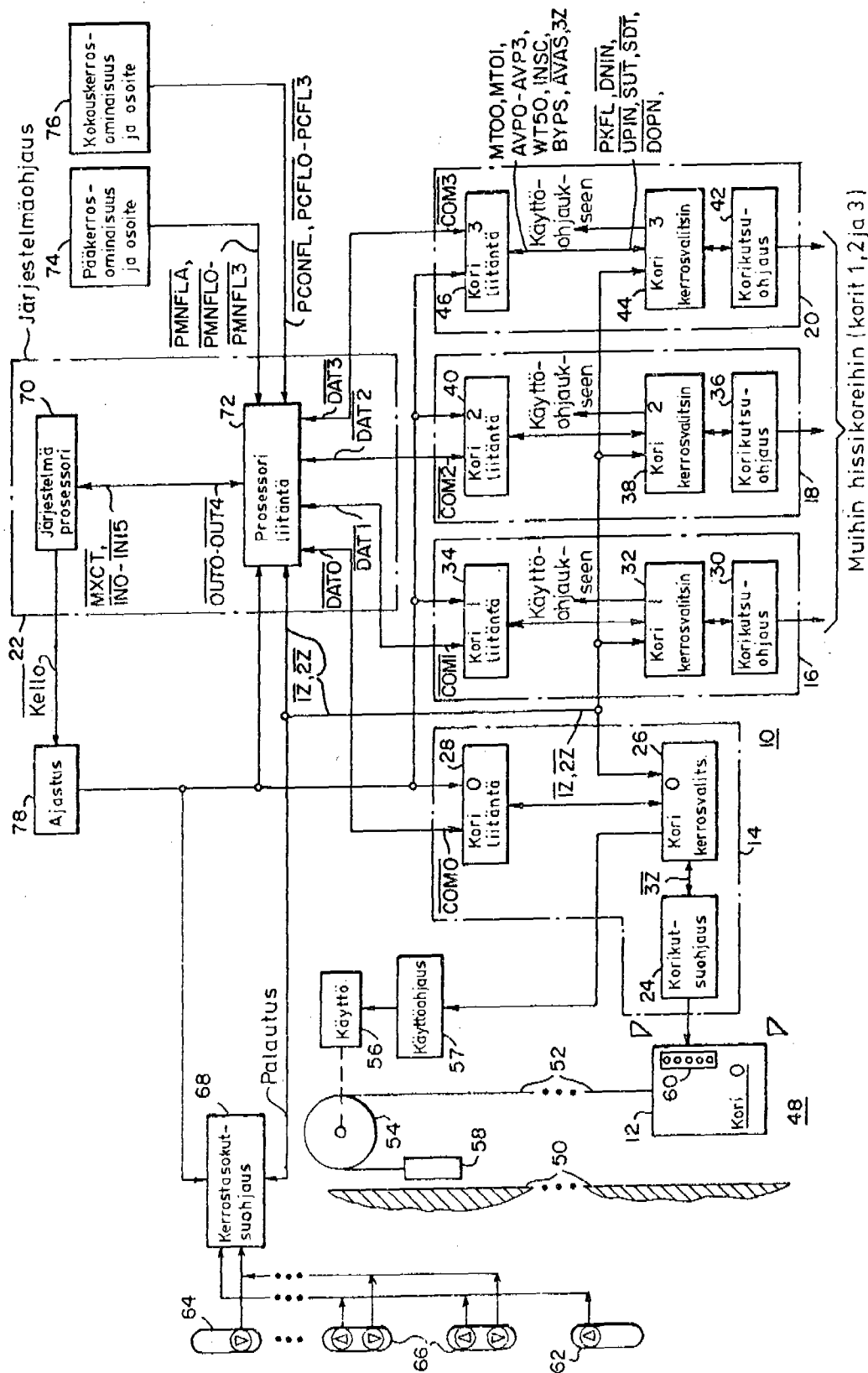


FIG.1

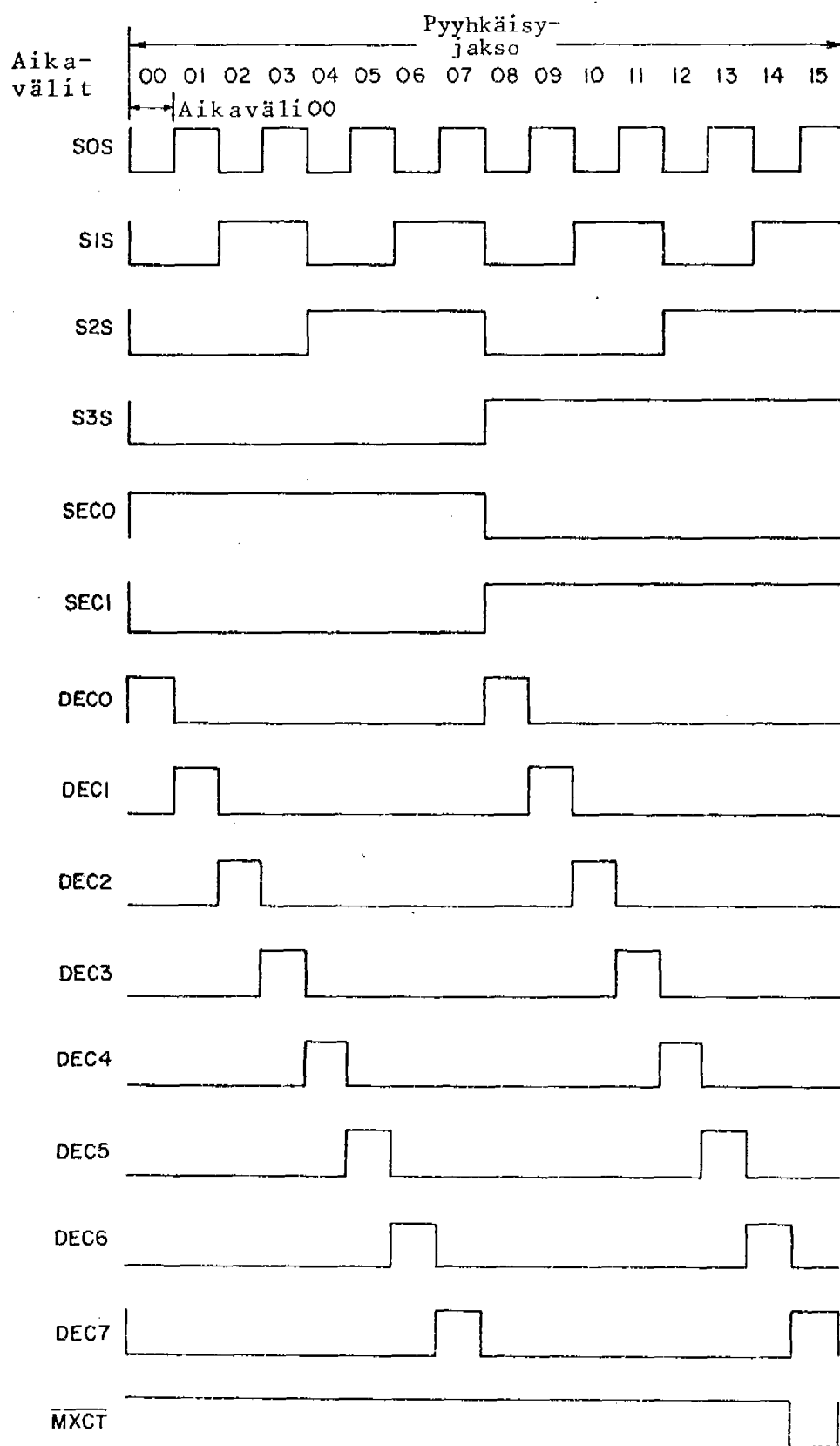


FIG. 2

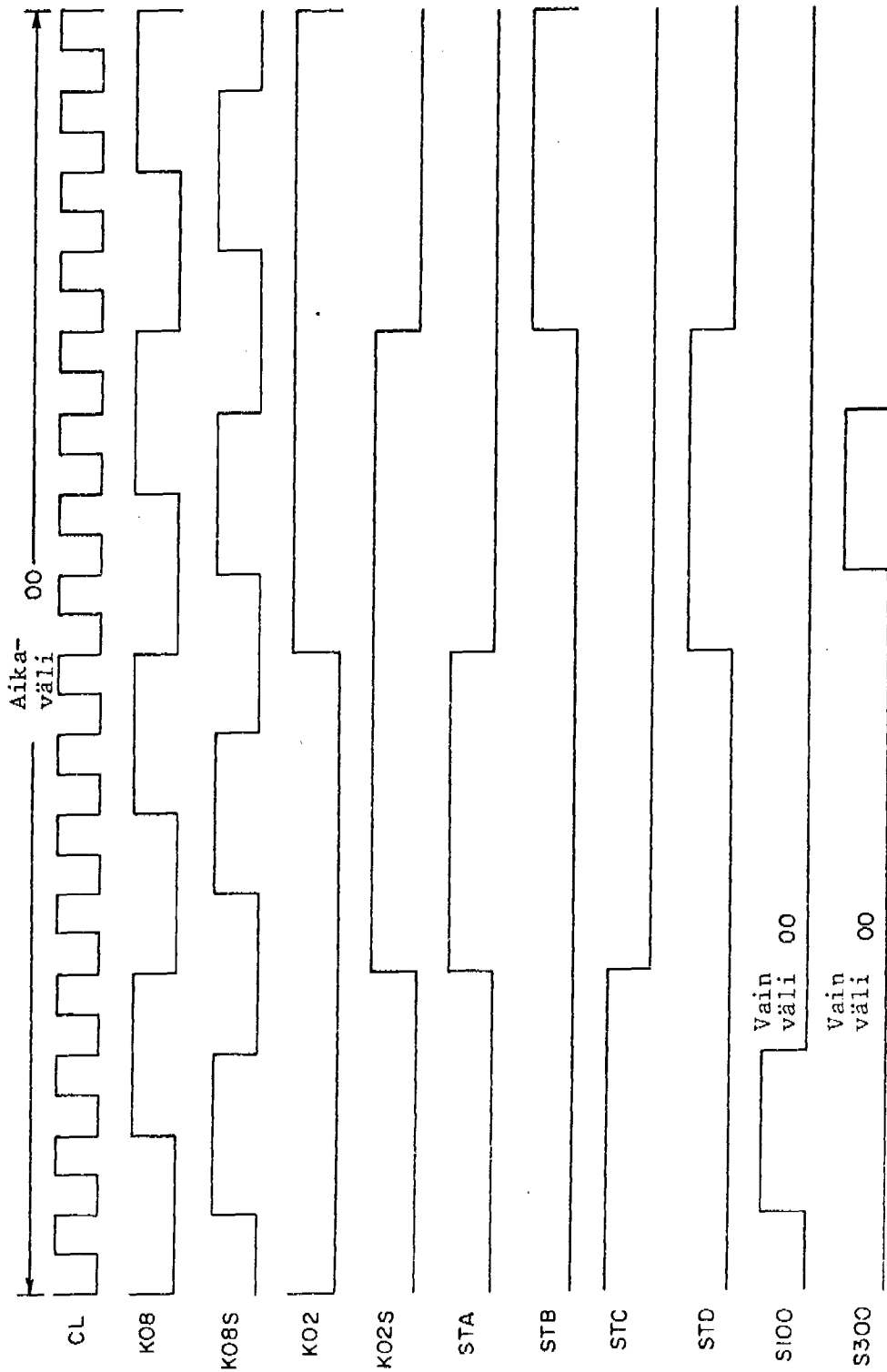


FIG. 3

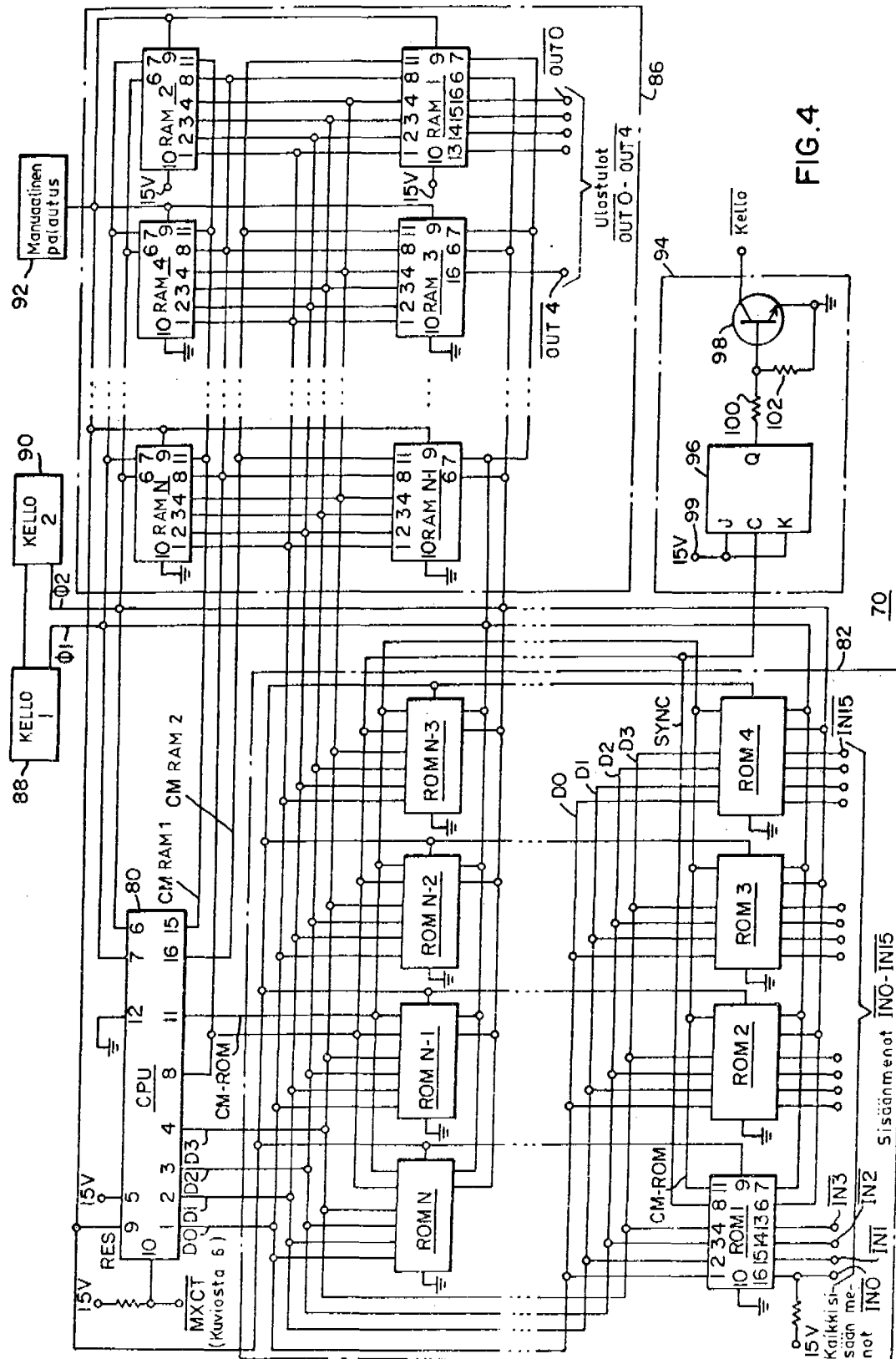


FIG. 4

Aikaväli tai rivi

FIG. 5

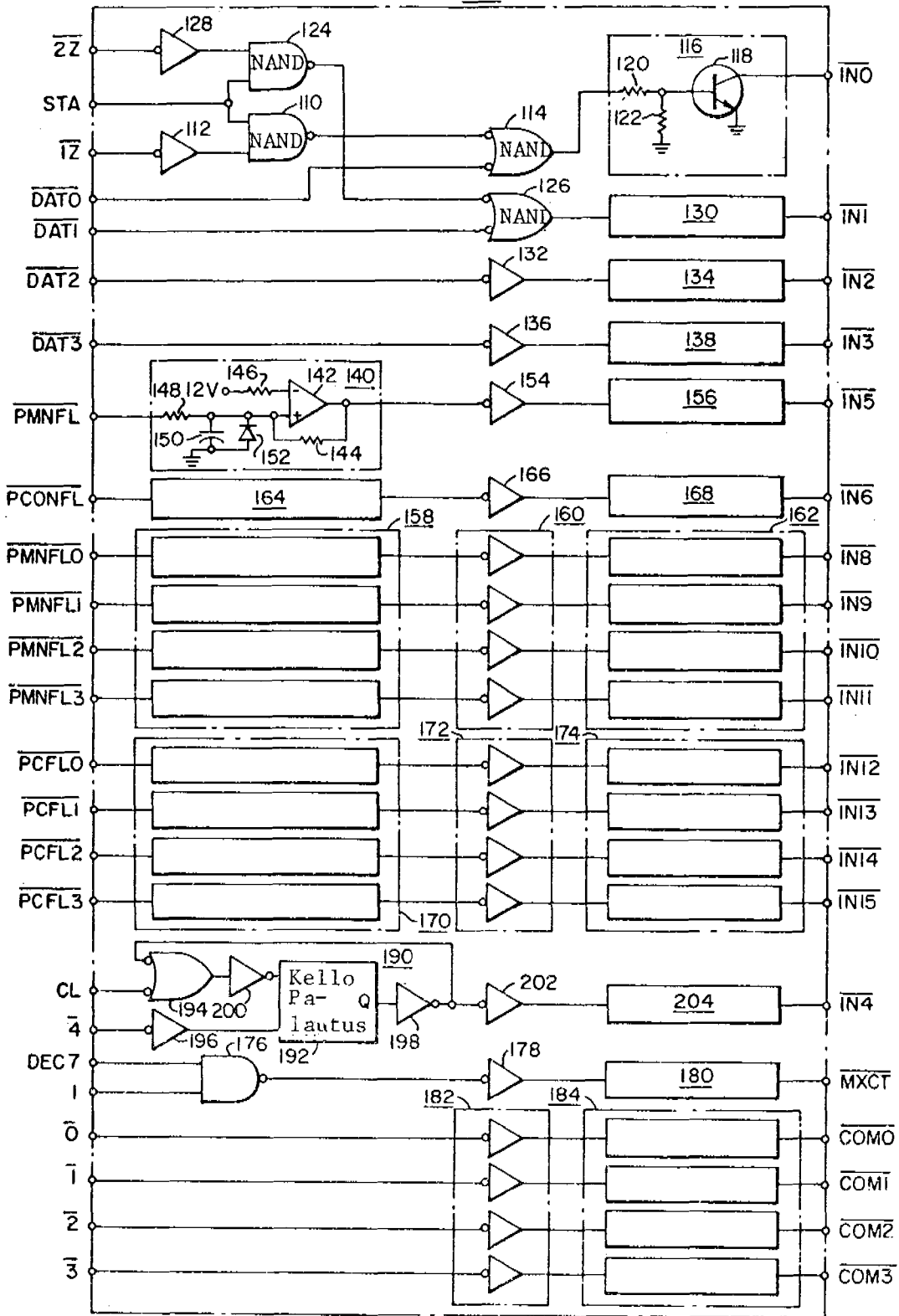


FIG. 6

SARJA - SISÄÄN MENOSIGNAALIT KÖREISTA

KERROSASEMA AIKA VÄLI	1		2		3		4		5		6	
	TILA	00	TILA	01	TILA	02	TILA	03	TILA	04	TILA	05
INO (KORI 0)	IZ	FEN	3Z	IZ	FEN	3Z	IZ	FEN	3Z	IZ	FEN	3Z
IN1 (KORI 1)	"	"	"	2Z	"	"	"	2Z	"	"	2Z	"
IN2 (KORI 2)	"	"	"	"	"	"	"	"	"	"	"	"
IN3 (KORI 3)	"	"	"	"	"	"	"	"	"	"	"	"
MERKKI	STC	STA	STD	STB								

FIG.7

SARJA SIGNAALIT CPU:STA KÖREILLE

KERROSTASO AIKA VÄLI COMIX)	1		2		3		4		5		6	
	TILA	00	TILA	01	TILA	02	TILA	03	TILA	04	TILA	05
MUUT KASKYT	MUUT KASKYT	PKFL	UPIN	DNIN	MUUT KASKYT	PKFL	UPIN	DNIN	MUUT KASKYT	PKFL	UPIN	DNIN
MERKKI	STC	STA	STD	STB								

FIG.8C

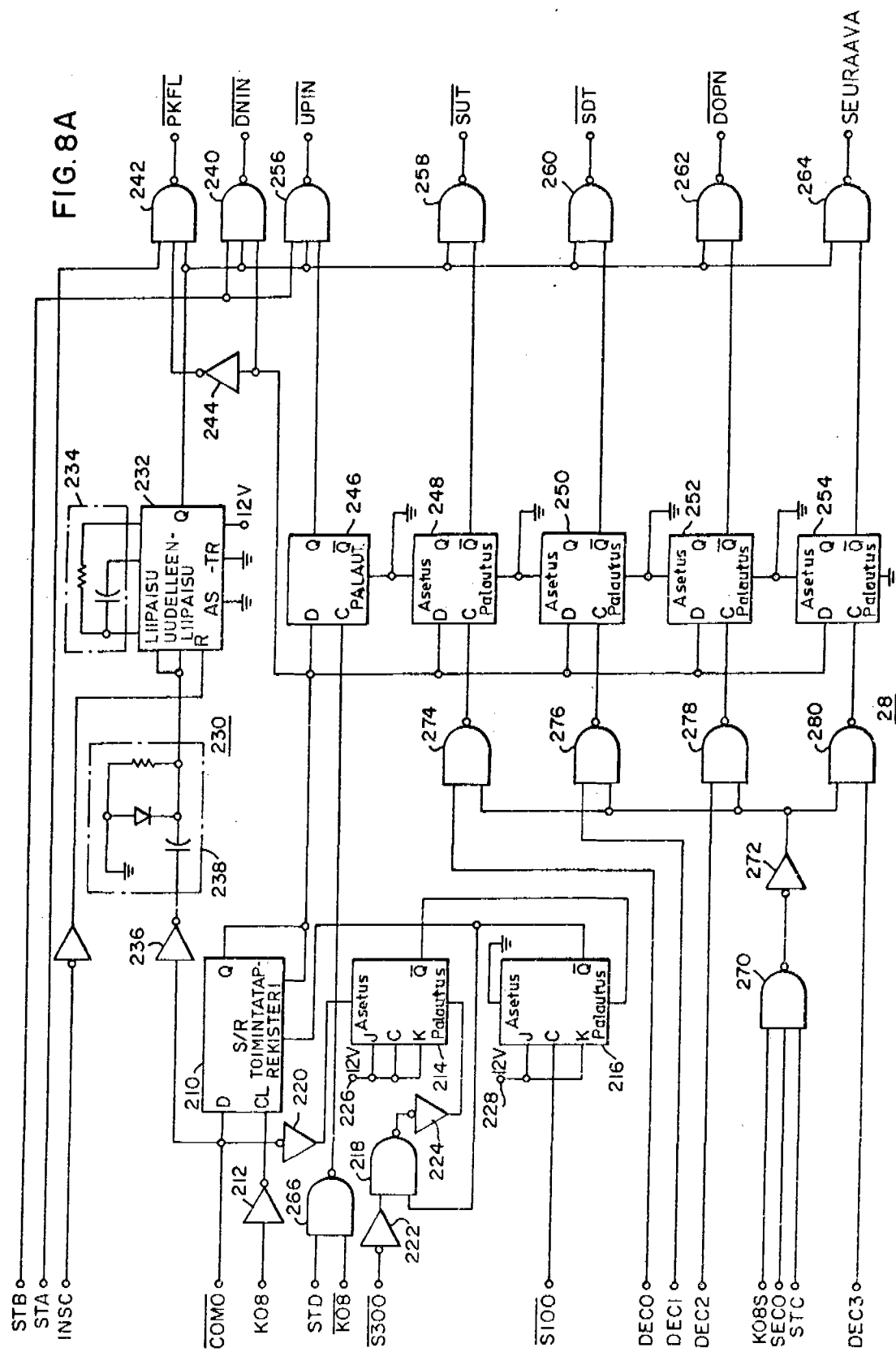
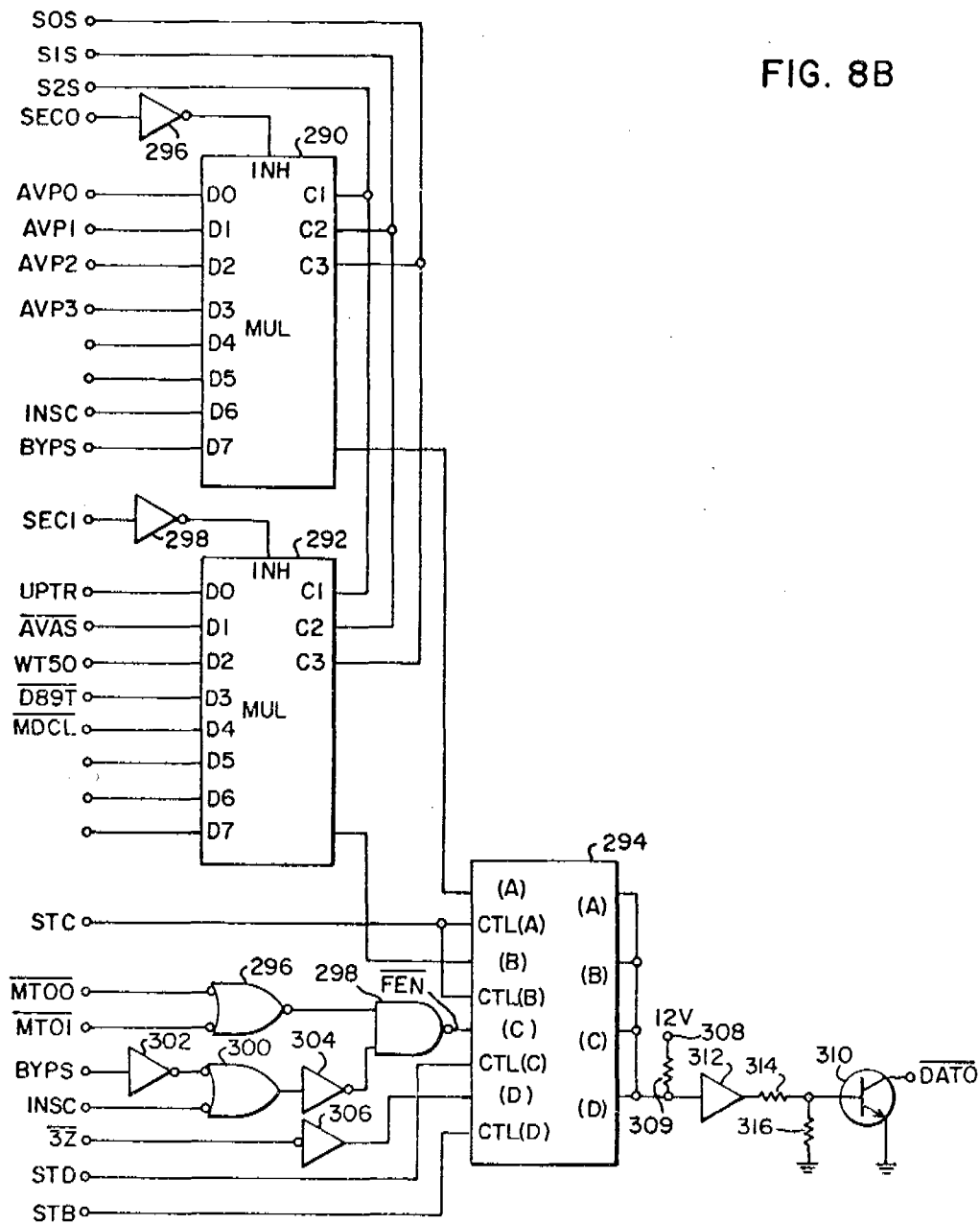


FIG. 8B



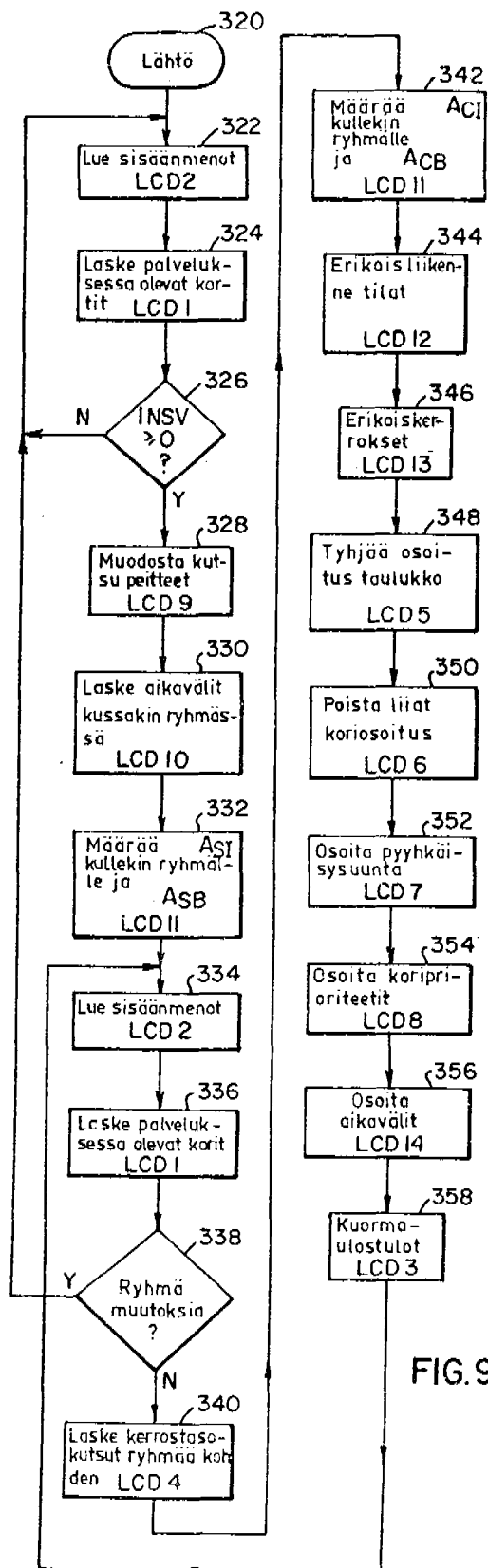


FIG.9

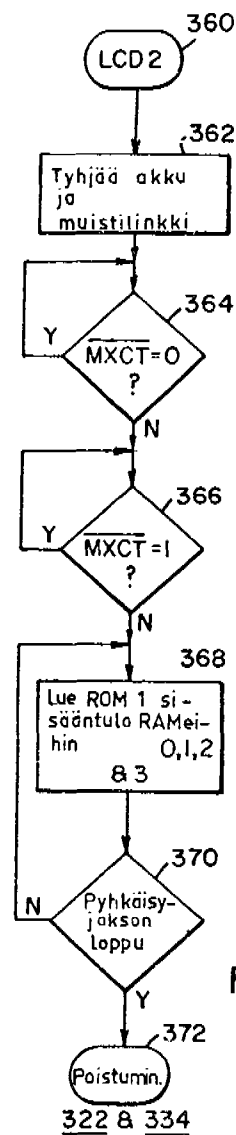
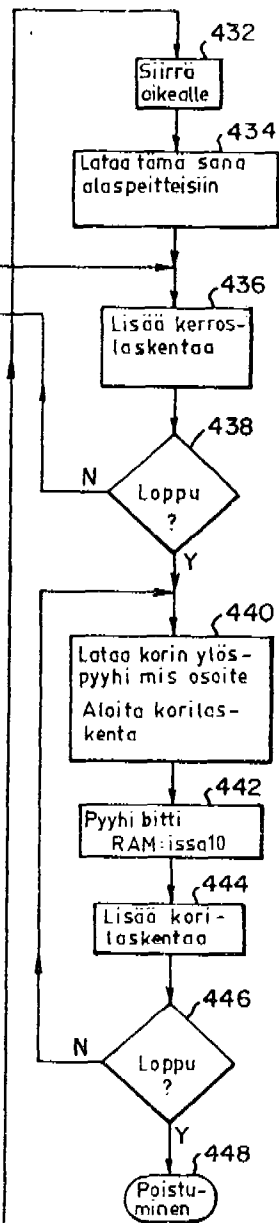
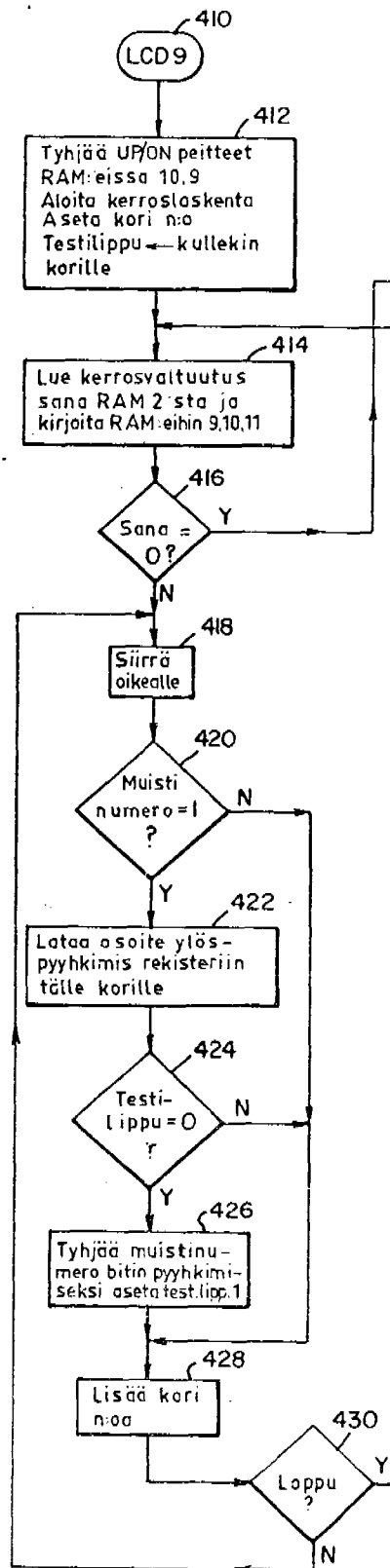
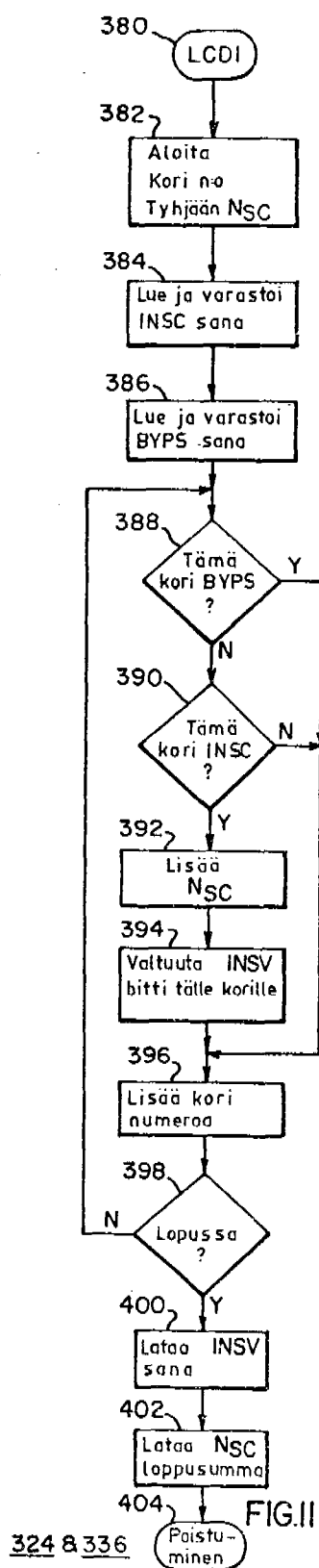


FIG.10



328
FIG.12

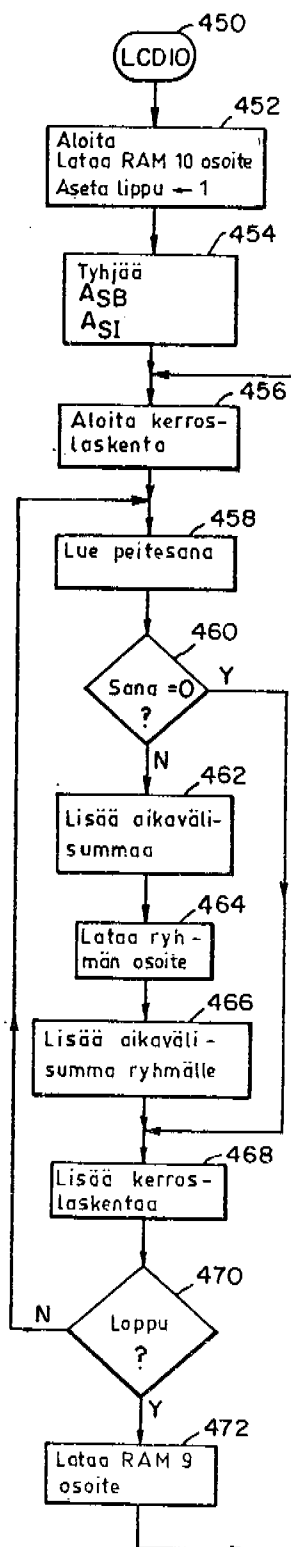


FIG.13

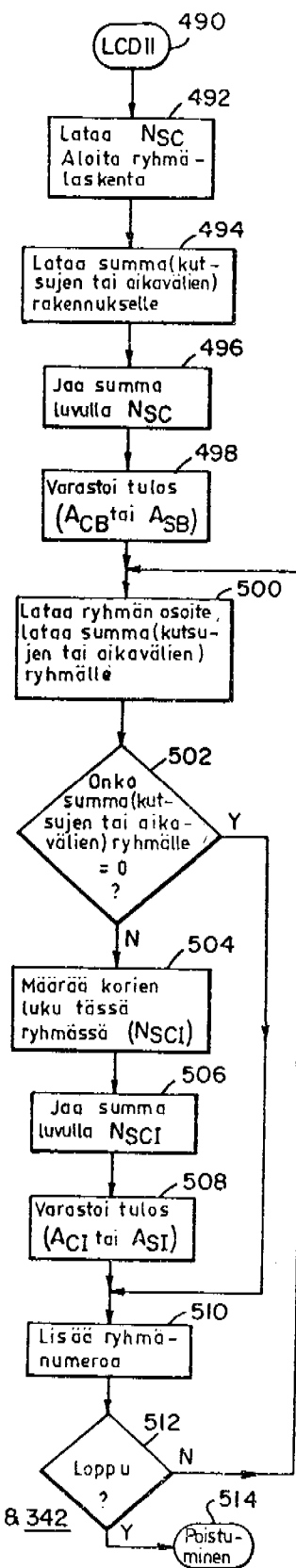


FIG.14

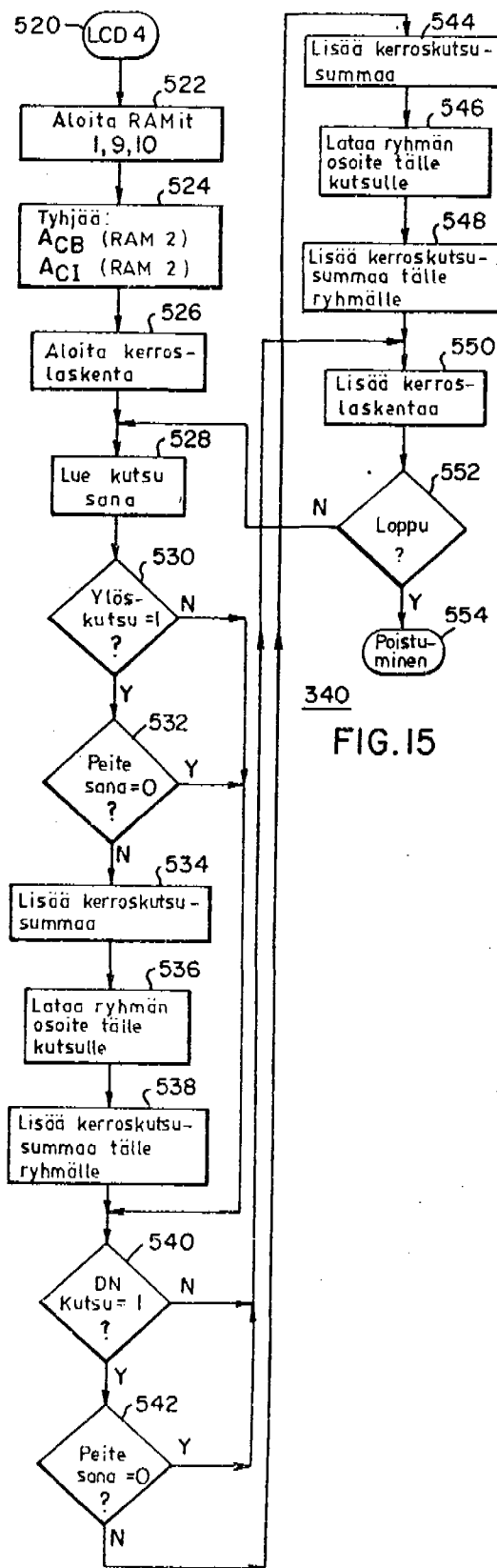


FIG. 15

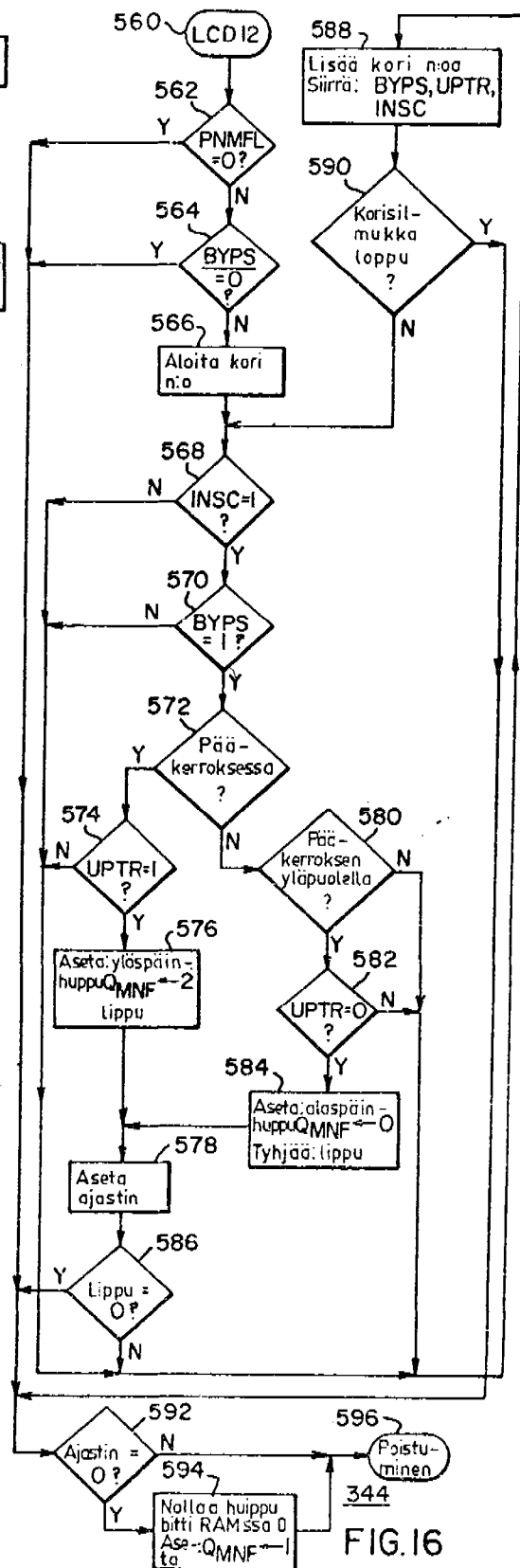
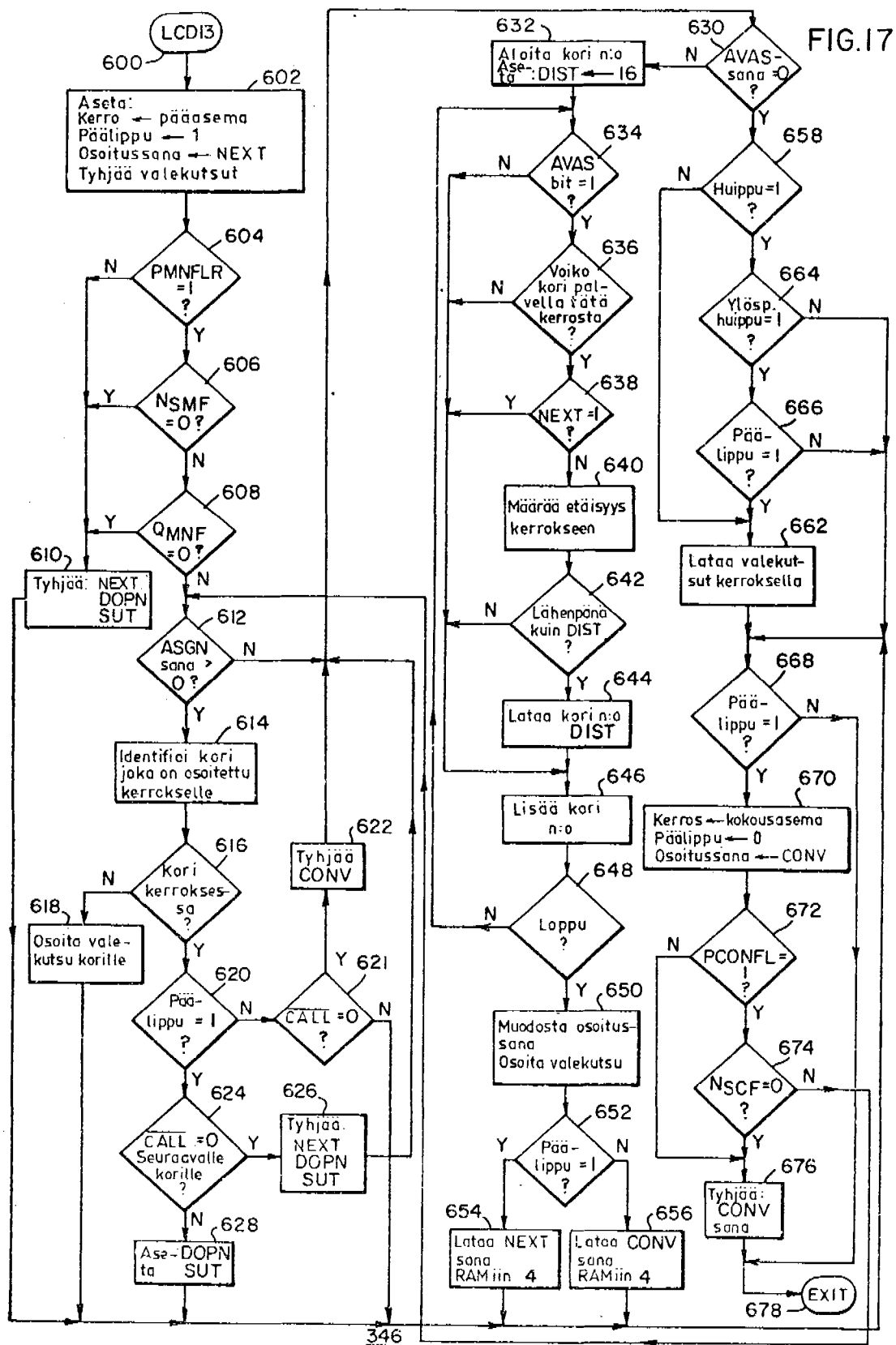


FIG. 16



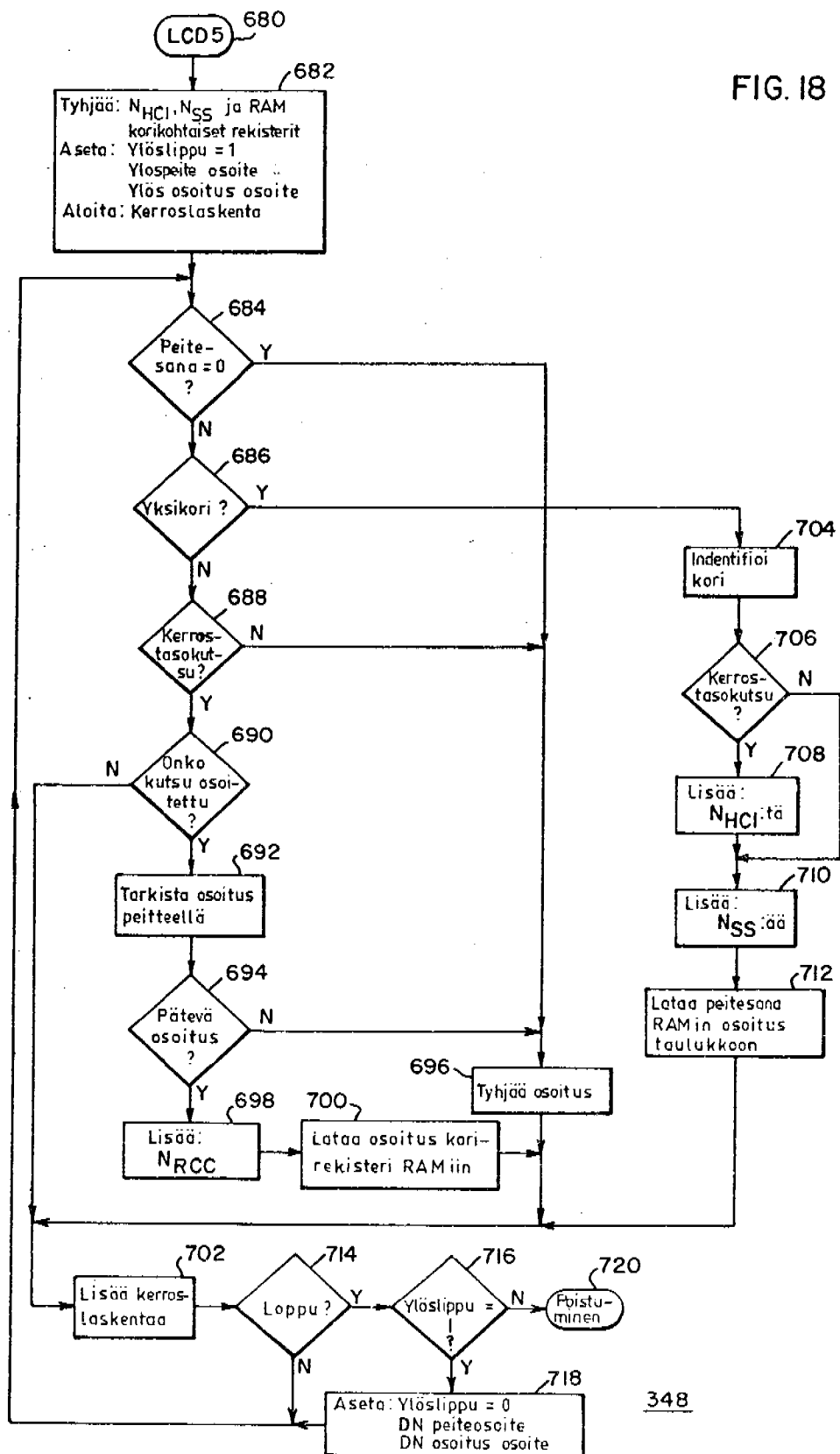
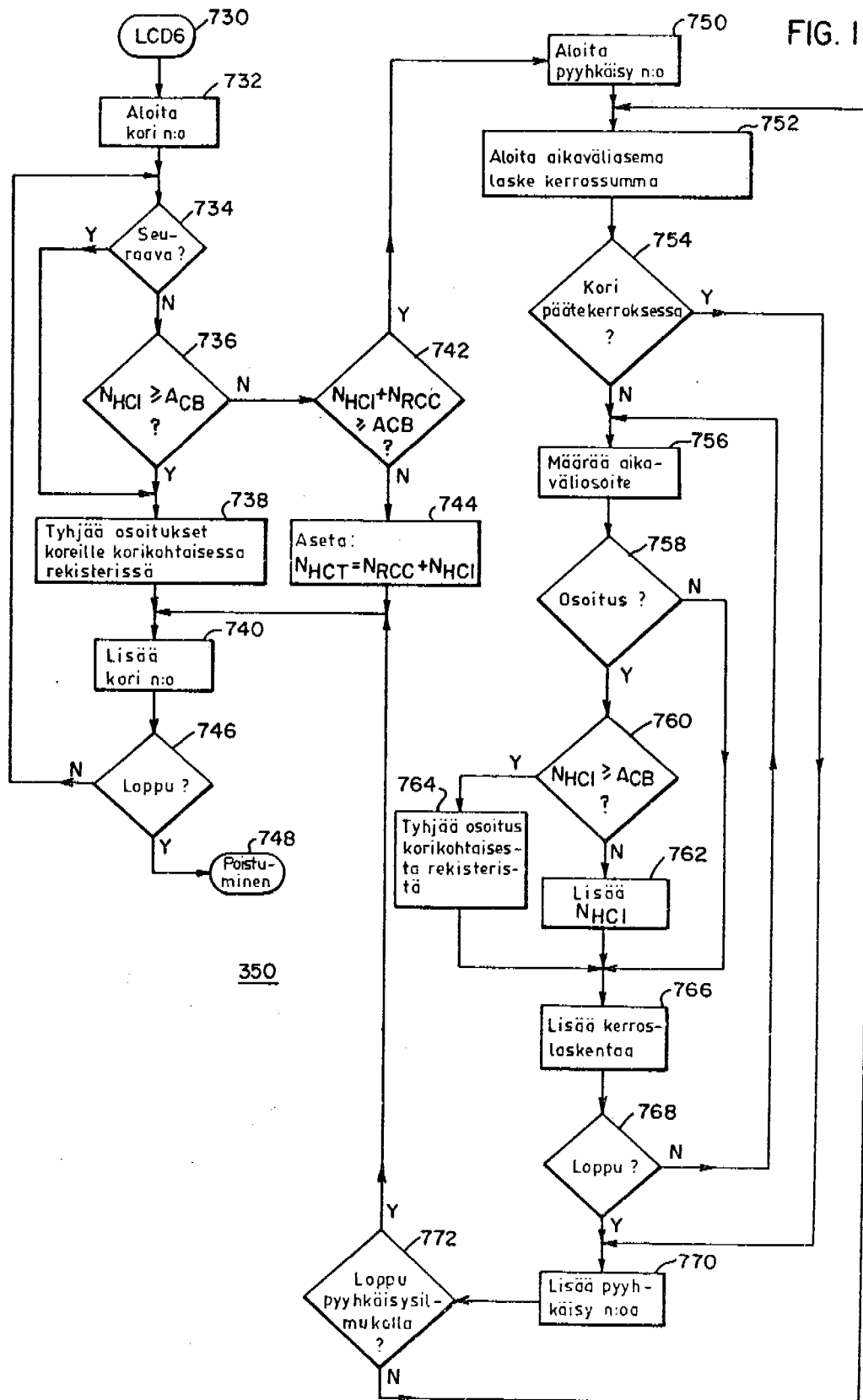
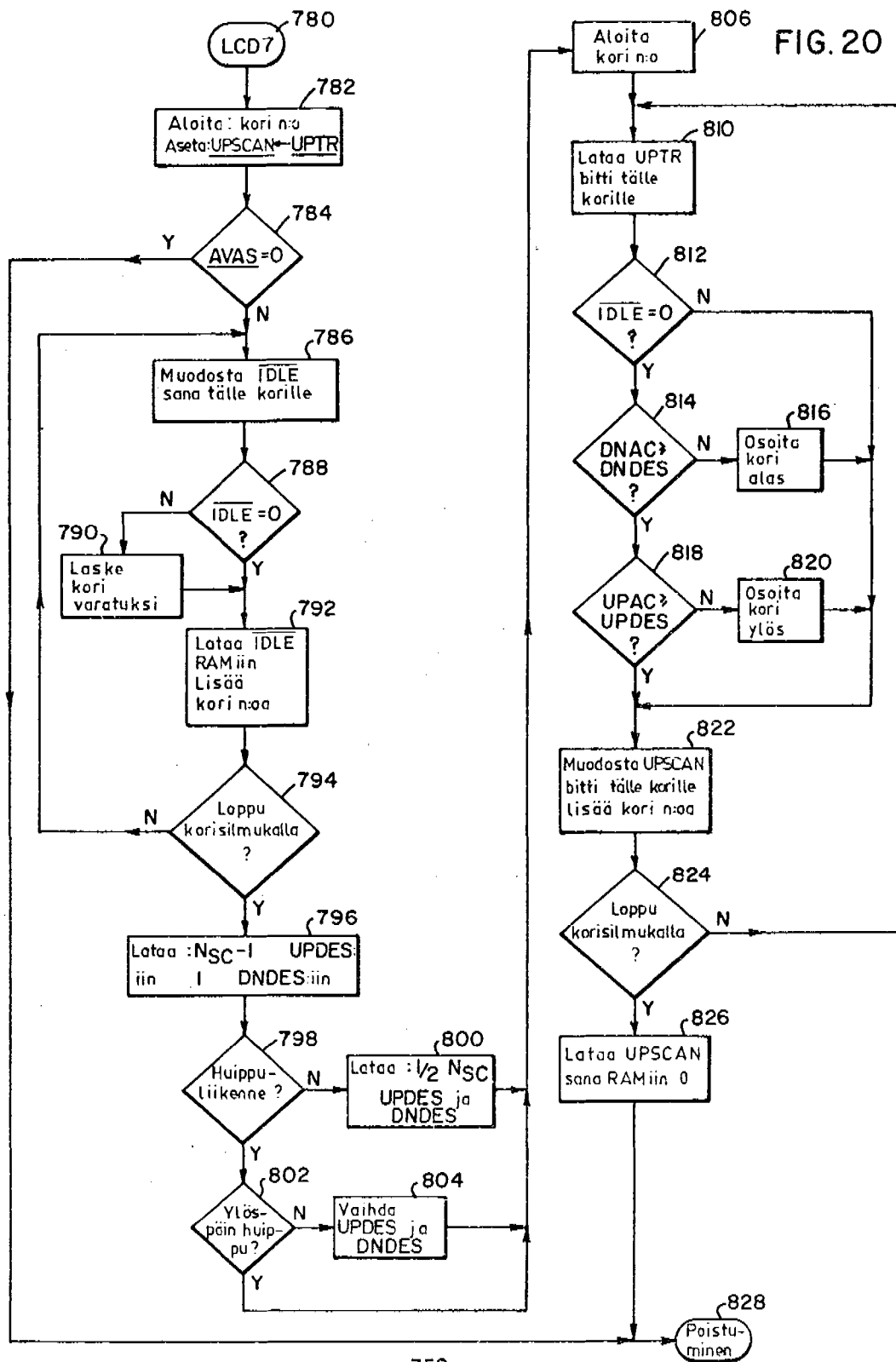
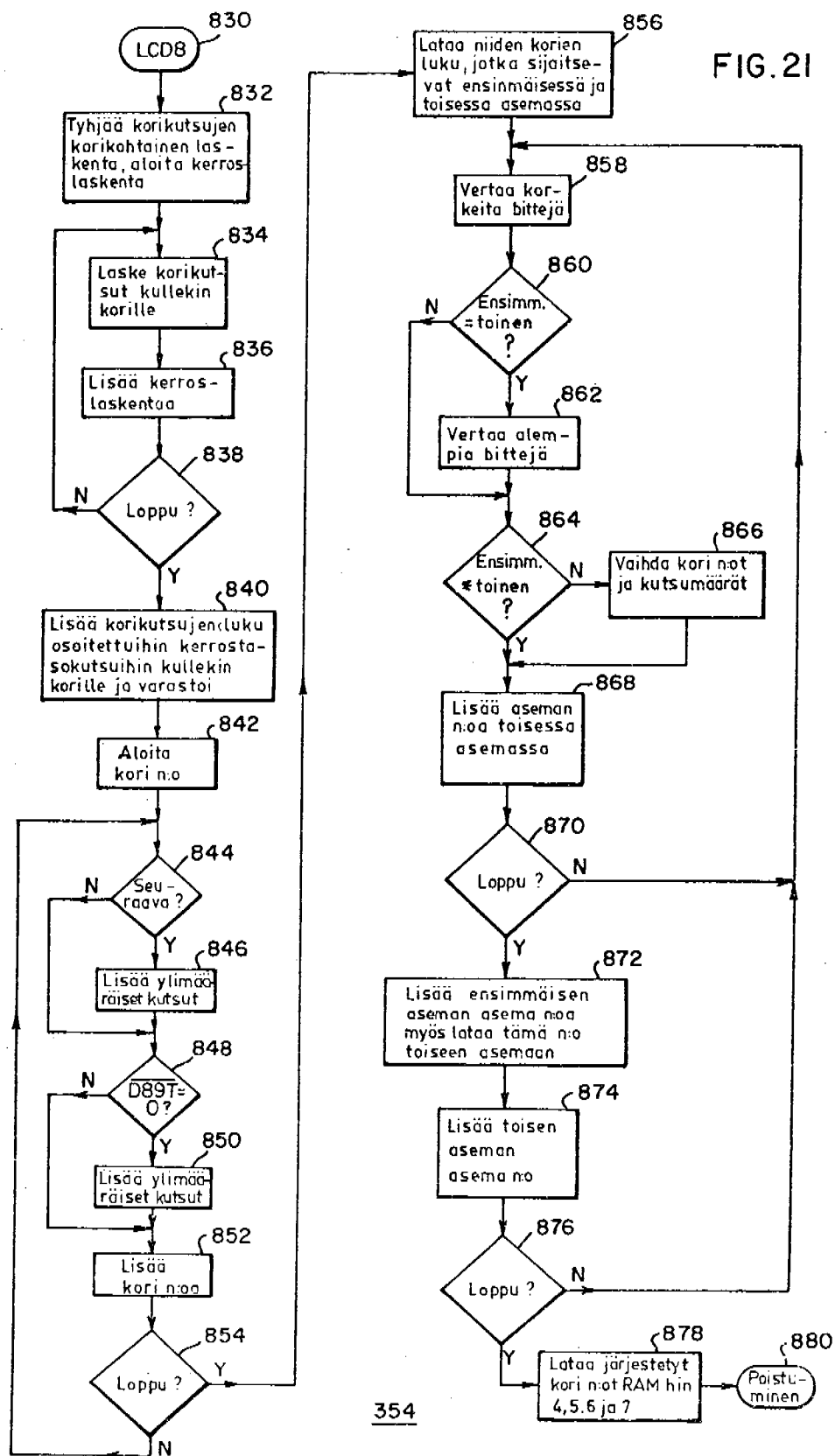


FIG. 19



350





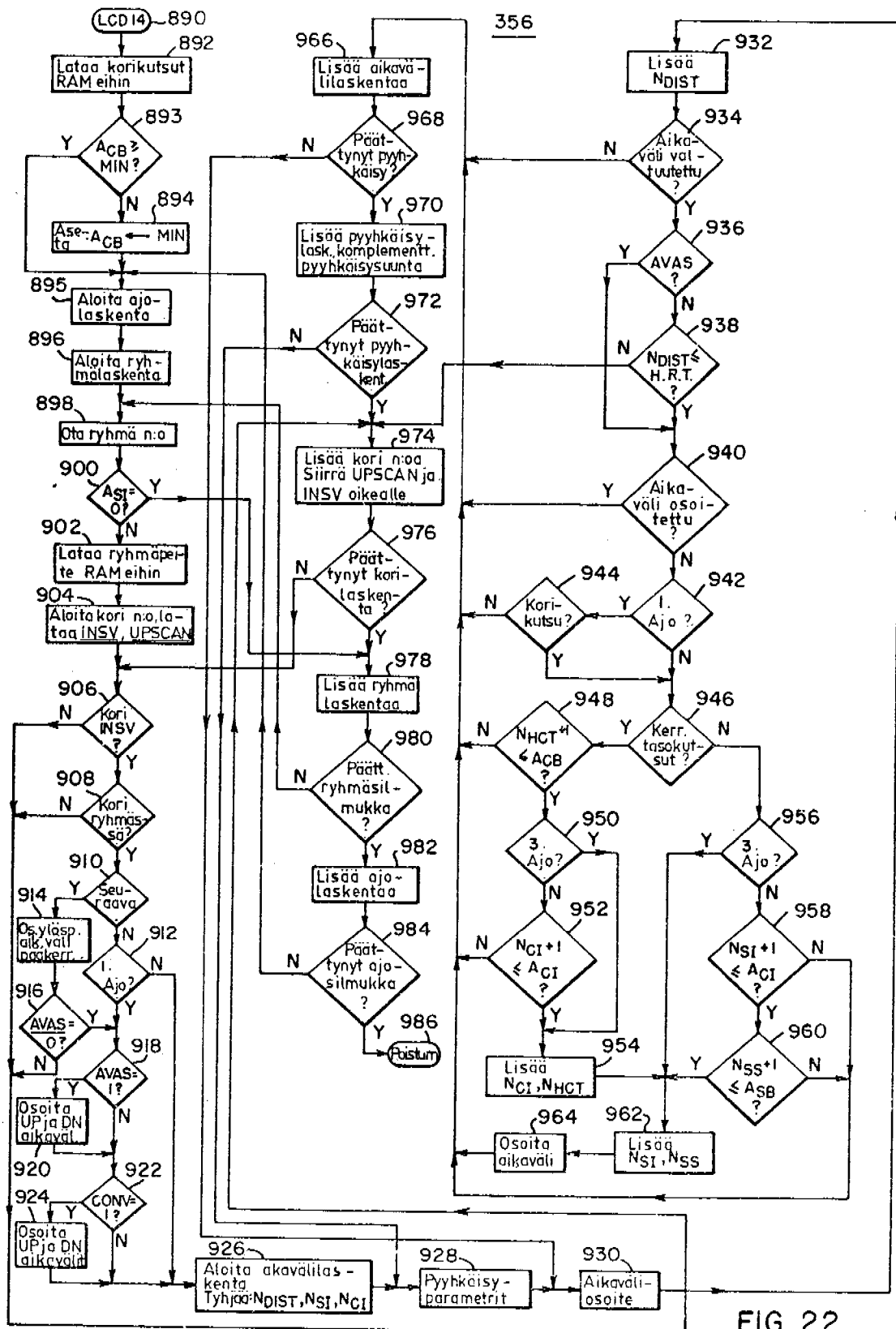


FIG. 22

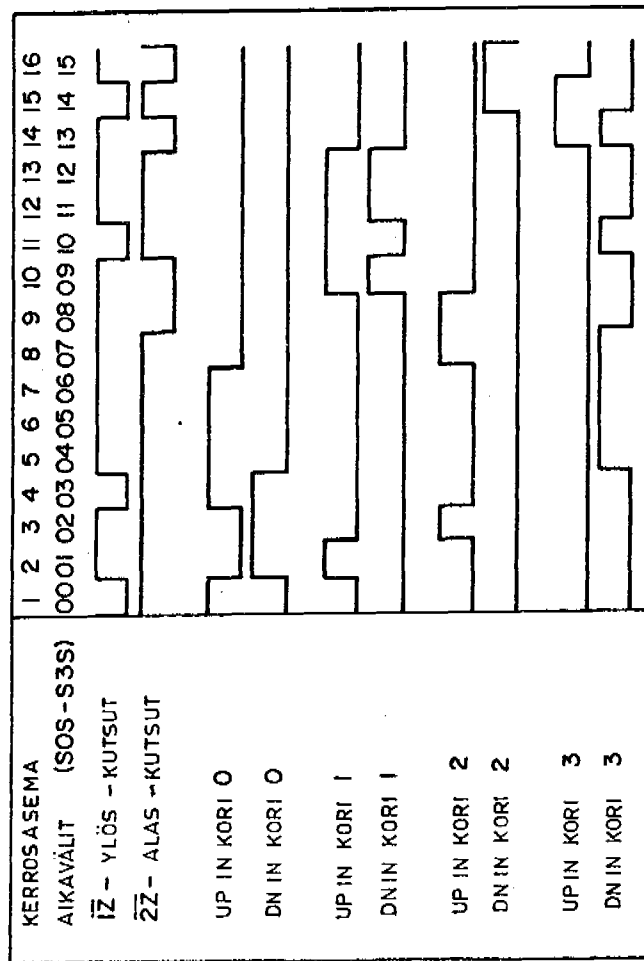
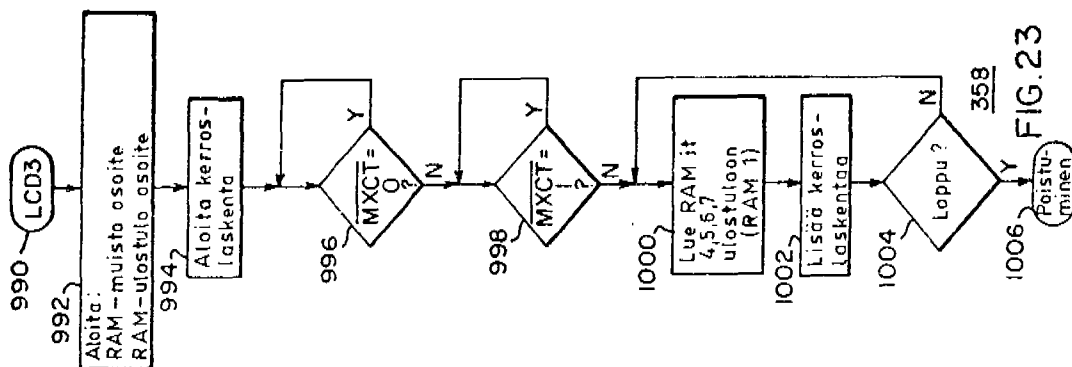


FIG. 25

AIKA VALIT	KERROS ASEMAT FUNKTIOT	KORIT				AIKA VALIT	KERRAS TASO	OSOITUS TAU- LUKOT				RYHMÄ OSOITTEET	ACI (RAM 2)	ASI (RAM 8)	ALAS-KUTS PEITTEET (RAM 9)				YLÖS-KUTS. PEITTEET (RAM 10)																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																										
		3 U D	2 U D	1 U D	0 U D			YLOS (RAM 6)	KORIT 3 2 1 0	ALAS (RAM 7)	KORIT 3 2 1 0				3	2	1	0	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																							
15	16	TE2																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											

FIG.24

Viitejulkaisuja - Anförda publikationer

Julkisia suomalaisia patenttihakemuksia: - Offentliga finska patentansökningar:

Hakemus-, kuulutus- ja patenttijulkaisuja: - Ansökningspublikationer,
utläggnings- och patentskrifter:

FI

CH

DE

2 319 440

DK

FR

GB

NO

SE

US

3561 571 , 3589 473 , 3828 892

Merkitse hakemusjulkaisu (esim. saksal. Offenlegungsschrift) numeron
eteen H ja vastaavasti kuulutus- ja patenttijulkaisu numeron eteen K ja P.