



H U 0 0 0 2 1 4 4 2 7 B

(19) Országkód:

**HU****MAGYAR  
KÖZTÁRSASÁG****MAGYAR  
SZABADALMI  
HIVATAL****SZABADALMI  
LEÍRÁS**

(11) Lajstromszám:

**214 427 B**

(21) A bejelentés ügyszáma: 3984/91

(22) A bejelentés napja: 1991. 12. 17.

(30) Elsőbbségi adatok:  
07/629,864 1990. 12. 18. US(51) Int. Cl.<sup>6</sup>**H 04 Q 3/545**

G 06 F 15/16

(40) A közzététel napja: 1992. 11. 30.

(45) A megadás meghirdetésének a dátuma a Szabadalmi  
Közlönyben: 1998. 03. 30.

(72) Feltalálók:

Kaiser, John Michael, Cedar Park, Texas (US)  
St. Clair, Joe Christopher, Round Rock, Texas  
(US)

(73) Szabadalmas:

International Business Machines Corp., Armonk,  
New York (US)

(74) Képviseelő:

S.B.G. & K. Budapesti Nemzetközi Szabadalmi  
Iroda, Budapest

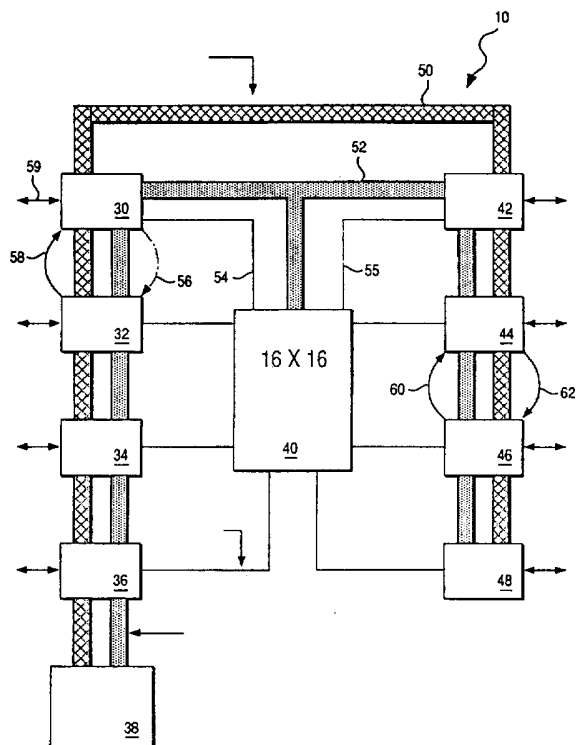
(54)

**Adatátviteli rendszer**

(57) KIVONAT

A találmány adatátviteli rendszerre vonatkozik.

Az adatátviteli rendszernek kapcsolómátrixa (40), vezérlő bemenetekkel/kimenetekkel ellátott N portja (30–36, 42–48) van, ahol a portok vezérlő bemeneteit/kimeneteit vezérlőbusz (52) köti össze egymással és a kapcsolómátrix vezérlőbemenetével, továbbá az i-ik ( $i=1, \dots, N$ ) port bemenete és kimenete a kapcsolómátrix (40) i-ik kimenetéhez és bemenetéhez kapcsolódik, amelyre a találmány szerint az jellemző, hogy kapcsolómátrix (40) összekötő/szétkapcsoló logikát tartalmaz, ahol a logika bemenete egyben a kapcsolómátrix (40) vezérlőbemenete, amely címtároló vezérlő bemenetére van kötve.



2. ábra

**HU 214 427 B**

A találmány adatátviteli rendszerre vonatkozik.

Az adatfeldolgozó rendszereknél gyakran szükség van az információ átvitelére az egyes perifériális egységek, terminálok vagy több számítógép között. Nagy teljesítményű adatfeldolgozó rendszerekben, a rendszer elemei közötti adatátvitel kritikusan befolyásolja az elérni kívánt nagy teljesítményt. A nagy teljesítményű rendszerek egy általánosan használt példáját képezik a telekommunikációs rendszerek.

Az US 4,256,926 sz. szabadalom például egy olyan mikrovezérelt telekommunikációs kapcsolórendszert ismertet, amelynek mikroprocesszorokkal megvalósított elosztott vezérlése van, ahol a mikroprocesszorok pufferral vannak ellátva a mikroprocesszorok közti adatátvitel biztosítására.

Az US 4,119,803 sz. szabadalom olyan telefonközponti kapcsolórendszert ismertet, amelyet két vezérlőegység vezérel egy központi mechanizmuson keresztül, és amelynek több regisztere van a forgalom vezérlésére, illetve párbeszéd egysége van a vezérlőjelek és adatjelek átvitelére a telefonközpont irányában.

Az US 4,412,282 sz. szabadalom egy mikroprocesszoros vezérlésű telefonkapcsoló áramkört ismertet, amelynél nagyobb megbízhatóságot biztosít egy adat- és címparitás áramkör.

Az US 4,580,011 sz. szabadalom olyan elosztott feldolgozású telefon kapcsolórendszert ismertet, amelynek fő vezérlő (master) mikroprocesszora van, amely több speciális feladatot ellátó (function) mikroprocesszort vezérel.

Egy telekommunikációs áramkör és egy általános adatfeldolgozó interkommunikációs áramkör közötti különbség abban van, hogy az adatfeldolgozó rendszerben az adatátviteli terminálok száma kisebb, a megkívánt adatátviteli sebesség viszont általában nagyobb. Mindemellett további követelmény a dinamikus összekapcsolhatóság. Ezért különböző megoldásokat dolgoztak ki az egyes adatfeldolgozó rendszerek közötti adatátvitelre. Erre található példa többek között az US 4,264,782 sz. szabadalomban, ahol egy saját (host) processzor kommunikál több terminállal egy adatátviteli hálózaton keresztül. Ebben a rendszerben az átvitel titkosítható, de továbbra is a saját CPU vezérli azt. Ha a saját (host) CPU a fő vezérlő (master), akkor a terminálok kiszolgálók (slave) lesznek és az adatátviteli hálózaton folyó adatátvitel teljes vezérlését a saját CPU veszi át. Ez azonban szűk keresztmetszetet jelenthet, ha az adatátviteli rendszer több önálló kommunikációs egységet tartalmaz.

Az US 4,551,831 sz. szabadalom egy másik példát ismertet az adatfeldolgozó rendszereknél alkalmazható adatátviteli hálózat kialakítására, amelynél egy multiplex kapcsoló vezérel több csatornát. A multiplex kapcsolót egy központi vezérlő (CPU) vezérli. Adatfeldolgozó egységek adatátvitelére további példát találunk az US 4,710,868 sz. szabadalmi leírásban, amelynél több munkaállomás csatlakozik egy központi memóriához. A munkaállomások egy kétfokozatú kapcsolón keresztül férnek hozzá a központi memóriához, amely keresztül kívül a valós és virtuális címzés átfordítását is elvégzi.

Sok nagy teljesítményű adatfeldolgozó adatátviteli rendszerrel szükség van arra, hogy több adatfeldolgozó egység egyszerre kommunikálhasson egymással. Ezt a lehetőséget egy keresztponthoz kapcsoló biztosítja. Keresztponthoz kapcsoló alkalmazását ismerteti például az US 4,539,564 sz. szabadalmi leírás. Ez a megoldás összehasonlítható egy egyszerű információs busszal, amely egy időben csak egyetlen kommunikációs csatornát biztosít. A keresztponthoz kapcsoló szokásos konfiguráció esetén lehetőséget biztosít bármely terminál számára, hogy bármely másik, a rendszerhez tartozó nem foglalt terminállal kapcsolatba lépjen és több terminálpár között egyidejű adatátvitelt bonyolítsa le. Hagyományosan ez úgy valósul meg, hogy egy terminál hozzáférést kér a keresztponthoz kapcsolóhoz egy központi kapcsolóvezérlőn keresztül, amely először meghatározza, hogy a vevő terminál képes-e üzenet vételére az adó termináltól. Egy ilyen adatátvitel lehetőségét megerősítő státuszjel vétele után a keresztponthoz kapcsoló utasítást kap arra, hogy hozzon létre kapcsolatot az adó és a vevő terminál között, annak érdekében, hogy a két terminál információt cserélhessen. Ha az adatátvitel véget ért, a küldő terminál jelzést küld a keresztponthoz kapcsoló központi vezérlőjének a vevő terminál lekapcsolása érdekében.

A találmány célja egy intelligens rendszer kialakítása egy adatfeldolgozó rendszerben az egyes elemek közötti kommunikáció vezérlésére az adatátvitel figyelésével annak meghatározására, hogy mikor kell változást létrehozni az adatátviteli rendszerben. Ezt a célt a kiválasztott két rendszerrel közötti adatátvitel figyelésével érjük el.

Az US 4,539,564 sz. szabadalmi leírás egy keresztponthoz kapcsoló típusú kapcsolónál a bemenetek és a kimenetek megfigyelését írja le. Jóllehet ez a megfigyelés csupán karbantartási célokat szolgál annak meghatározása érdekében, hogy mikor jelentkezik hiba és hogy mi a hibaállapotok forrása. Az IBM Technical Disclosure Bulletin, Vol. 31, No. 1, June, 1988. „Cross Point Switch Tracer” címen közölt megoldása szintén tartalmaz egy karbantartó áramkört, amely figyelési és időjellel egészíti ki a keresztponthoz kapcsolón keresztül folyó adatátvitelt. A fenti hivatkozások egyike sem ismertet olyan rendszert, amelyben az adatátvitel megfigyelése annak érdekében történne, hogy vezéreljék az adatátvitelt egy információs buszon.

A találmány kidolgozásánál olyan adatátviteli rendszerből indultunk ki, amelynek kapcsolómátrixa és vezérlő bemenetekkel/kimenetekkel ellátott N portja van, ahol a portok vezérlő bemeneteit/kimeneteit vezérlőbusz köti össze egymással és a kapcsolómátrix vezérlőbemenetével, továbbá az i-ik ( $i = 1, \dots, N$ ) port bemenete és kimenete a kapcsolómátrix i-ik kimenetéhez és bemenetéhez kapcsolódik. A találmány szerinti adatátviteli rendszerrel a kapcsolómátrix összekötő/szétkapcsoló logikát tartalmaz, ahol a logika bemenete egyben a kapcsolómátrix vezérlőbemenete, amely címtároló vezérlő bemenetére van kötve.

Egy előnyös kiviteli példánál a kapcsolómátrix N adatkiválasztó áramkört is tartalmaz, amelyek vezérlő bemenetei a címtároló kimeneteihez csatlakoznak.

Előnyös továbbá egy olyan kialakítás is, ahol a kapcsolómátrix N bemeneti pufferrel is rendelkezik, ahol az *i*-ik puffer bemenete a kapcsolómátrix *i*-ik bemenetéhez csatlakozik, és a puffer kimenete mindegyik adatkiválasztó áramkör egy-egy bemenetére rá van kötve.

Egy további előnyös kiviteli példánál a kapcsolómátrix N kimeneti pufferrel is rendelkezik, ahol az pufferek bemeneteikkel az adatkiválasztó áramkörök kimeneteihez csatlakoznak, és az *i*-ik kimeneti puffer kimenete a kapcsolómátrix *i*-ik kimenetét képezi.

Szintén előnyös egy olyan kialakítás, amelynél a kapcsolómátrix N regiszterrel is rendelkezik, ahol az *i*-ik regiszter bemenete a címtároló kimenetéhez, kimenete pedig az *i*-ik adatkiválasztó áramkör vezérlő bemenetéhez csatlakozik.

A találmány további céljait, változatait és előnyeit a találmány előnyös kiviteli alakjainak részletes leírásából ismerhetjük meg, ahol az

1. ábra egy blokkvázlat, ahol nyolc rendszer csatlakozik az egyik keresztponthoz és egy további rendszer egy második keresztponthoz, a
2. ábra az 1. ábra szerinti keresztponthoz belső felépítésének blokkvázlata, a
3. ábra a találmány szerinti kommunikációs hálózaton alkalmazott kapcsolómátrix port áramkörének blokkvázlata, a
4. ábra a keresztponthoz kapcsoló blokkvázlata, az
- 5a. ábra az A és B port, valamint a keresztponthoz kapcsoló közötti események idődiagramja egy kapcsolat létrehozása során, az
- 5b. ábra az A és B port szétválasztásának eseménydiagramja, a
6. ábra a port vezérlésének folyamatábrája egy kommunikációs keret vételkor, a
7. ábra a port vezérlésének folyamatábrája egy kiszolgálókérés vételkor az adatátviteli buszról, a
8. ábra az adatátvitel port által kezdeményezett befejezésének folyamatábrája és a
9. ábra a kapcsolómátrix vezérlésének folyamatábrája a portok közötti adatátvitel során.

Az 1. ábra egy olyan adatátviteli rendszer blokkvázlata, amely különböző 14, 16, 18, 20, 22, 24, 26 és 28 rendszereket tartalmaz, amelyek 10 keresztponthoz csatlakoznak. A rendszerek mindegyike, így például a 14 rendszer is 8 porton keresztül csatlakozik a 10 keresztponthoz. Megjegyezzük, hogy mindegyik rendszer, így például a 24 rendszer is további, pl. 12 keresztponthoz is csatlakozhat a redundancia növelése érdekében. A jelen előnyös kiviteli példánál a 14 és 24 rendszerek RISC System/6000 munkaállomások, amelyek soros száloptikai csatornán keresztül kapcsolódnak a 10 keresztponthoz. Ennél a kiviteli alaknál mindegyik RISC System/6000 tartalmazhat négy portot az egymás közötti soros kapcsolat létrehozásához. Az egymás közötti soros kapcsolat adatátvitelére például ESCON (Enterprise System Connection for the IBM 3090) Enterprise rendszer soros bemeneti/kimeneti csatorna protokoll szerinti lehet. Ennél a kiviteli példánál

világos, hogy ha az egyik rendszert egy másik rendszerhez kell kapcsolni, hogy adatokat szolgáltatson a második rendszer számára, akkor minden információ ezen a soros kapcsolatot biztosító száloptikai csatornán fog keresztüljutni. A forrás rendszer legfeljebb 32 bájtos keretinformációt küld ki a fogadó rendszerrel való kommunikáció megkezdéséhez. Az első keret elküldése és vétele után, amely létrehozza a kapcsolatot a 10 keresztponthoz, ez a kapcsolat fennmarad és így a forrás rendszer folyamatosan további információs kereteket tud küldeni a fogadó rendszernek egészen addig, amíg egy megszakító keret nem jelzi a fogadó rendszernek és a 10 keresztponthoz kapcsolóknak az adatátvitel végét, amire a 10 keresztponthoz kapcsoló szétválaszt. Az előnyös kiviteli példánál a 10 keresztponthoz kapcsoló egy  $N \times N$  porttal rendelkező  $N \times N$  típusú kapcsoló, amely egyidejű adatátvitelt biztosít az összekapcsolt portok és a portokhoz csatlakozó rendszerek között.

A 2. ábra a 10 keresztponthoz kapcsoló blokkvázlata. A rajzon látható előnyös kiviteli példánál a 10 keresztponthoz kapcsoló  $16 \times 16$ -os mátrixot alkot. A leírás egyszerűsége kedvéért a tizenhat portból csak nyolc látható. Mindegyik 30–36, illetve 42–48 port portot 50 kiválasztó buszhoz, 52 vezérlőbuszhoz és 54, illetve 55 adatátviteli vonalhoz kapcsolódik. A portok mindegyike egy ilyen adatátviteli vonalon keresztül csatlakozik a  $16 \times 16$ -os 40 kapcsolómátrixhoz. A 40 kapcsolómátrix szokásos katalógus elem lehet, mint például GIGABIT Logic 10G051, amely keresztponthoz kapcsolókat biztosít a portok között (a 600 logika és a 602 címtároló kivételével, 4. ábra).

Az előnyös kiviteli példánál mindegyik port optikai-elektromos átalakítást végez annak érdekében, hogy az információk villamos jelek formájában cserélődjenek ki a portok között a  $16 \times 16$ -os 40 kapcsolómátrixon keresztül. Először az egyik port, például a 30 port megpróbál kapcsolatot felvenni egy másik porttal, például a 32 porttal. Ennek során a 30 port először kiválasztást kér. Ez azt jelenti, hogy a 30 port engedélyt kér a portot kiválasztó 38 buszvezérlőtől a portot 50 kiválasztó buszon keresztül az 52 vezérlőbusz használatára. Az engedély megérkezésekor kapcsolatkérelmet küld az 52 vezérlőbuszon keresztül a 32 port felé. Ezután státusz adat vétel következik. A 2. ábrán olyan kiviteli példa látható, ahol 32 port próbál kapcsolatot teremteni a 30 porttal a 30 port kiküldésével, amit 58 nyíllal jelöltünk. A 30 port foglaltjelet küld (amit szaggatott 56 nyíllal jelöltünk) vissza a 32 portnak, elutasítva az adatátvitel-kérést. Ez alatt a portok közötti (port-to-port) kapcsolatfelvétel alatt a  $16 \times 16$ -os mátrix nem vesz részt az adatforgalomban. Ez azért lehetséges, hogy a kapcsoló mechanizmusának vezérlése el van osztva az egyes portok között. Más szavakkal, a portok közötti kapcsolatba csak azután lesz bevonva a 40 kapcsolómátrix, miután az adatátvitel lehetősége megerősítést nyert.

A 40 kapcsolómátrix az 52 vezérlőbuszra csatlakozik, amely engedélyezheti, hogy a 40 kapcsolómátrix válaszoljon a hozzá intézett utasításokra. Az előnyös kiviteli példánál csak diagnosztikai jellegű utasításokat kap a 40 kapcsolómátrix. Normál működés során a 40 kapcsolómátrix csupán az 52 vezérlőbuszt és a portok közötti

adatátvitelt figyeli, hogy meghatározza mikor kell az adatátvitelt folytatni és mikor kell azt befejezni. Amikor létrejön egy kapcsolat, az egyik, például az 54 adatátviteli vonal egy másik, például az 55 adatátviteli vonalhoz kapcsolódik és ez biztosítja az adatátvitelt a portok, mint például a 30 és 42 port között anélkül, hogy ehhez külön utasításra lenne szükség a portoktól vagy más vezérlőtől.

A szétkapcsolási műveletet a 40 kapcsolómátrix szintén a portok külön utasítása nélkül hajtja végre. A 40 kapcsolómátrix figyeli az 52 vezérlőbuszon az utasításokat, hogy meghatározza, hogy mikor kell a kapcsolatot megszakítani. Amikor egy megszakító üzenetet küld az egyik rendszer egy másik felé, az 52 vezérlőbuszt figyelő 40 kapcsolómátrix automatikusan észleli, hogy a kapcsolatot meg kell szakítani, és ezzel időt takarít meg, mert nem szükséges egy külön utasítás protokoll annak közlésére, hogy a 40 kapcsolómátrix szakítsa meg a kapcsolatot. Ez azért fontos, mert a megszakítás művelete magas prioritású, mivel az érintett portok csak a szétkapcsolás után építhetnek fel új kapcsolatokat.

A 3. ábra az egyes portok, mint például a 30 port belső felépítésének blokkvázlata. A mester vezérlő és a kapcsoló/foglalt 78 állapotlogika vezérli a portlogika működését. A 78 állapotlogika 82 megszakításvezérlőhöz csatlakozik, amely megszakítást kap és ad az 52 vezérlőbuszon hibaállapotok esetén. A 78 állapotlogika 88 szinkronlogikához is csatlakozik. A 88 szinkronlogika működése megegyezik az IBM Technical Disclosure Bulletin, Vol. 32, No. 6A, november, 1989 számában, a 21–22 oldalakon „Method for Validating Dynamic Data Paths in a Data Switching Unit” című tanulmányban ismertetett megoldásával. Amikor egy rendszertől egy keret érkezik, az először az 59B buszra kerül, ahol karakterenként eltárolásra kerül a 102 vevőregiszterben. Ennek a regiszternek a tartalma azután áttöltődik egy kapcsolat/szinkronizálás 104 pufferbe, ahol 90 írásvezérlő logika vagy 92 olvasásvezérlő logika a 78 állapotlogikával együtt meghatározza, hogy a 104 puffer FIFO tárként vagy pedig befogadó tárként működik. A 90 írásvezérlő logika határozza meg, hogy a 104 pufferben hova kell beírni az adatokat. A 92 olvasásvezérlő logika határozza meg, hogy a 104 pufferből honnan kell kiolvasni a következő karaktert. A dekódoló és hibafelismerő 106 logika is a 78 állapotlogikához kapcsolódik az esetleges hibaállapotok észlelésére. Ha a keretet egy másik porthoz kell továbbítani, kapcsolatkerő üzenet kerül az 52 vezérlőbuszra. Amint azt már korábban tárgyaltuk, az egyik port kiválasztó kérést küld a portkiválasztó 38 buszvezérlőnek a portot 50 kiválasztó buszon és a 100 busz interfészen keresztül. Engedélyezés esetén a port 78 állapotlogika kapcsolatkerést küld és kiértékeli a csatlakoztatni kívánt porttól az 52 vezérlőbuszon érkező státusz információt. Ha a csatlakoztatni kívánt port nem foglalt, akkor a kapcsolatot automatikusan létrehozza a 40 kapcsolómátrix, és a kapcsolat/szinkronizálás 104 puffer adata a 108 regiszteren keresztül a 40 kapcsolómátrix 54B adatátviteli vonalára kerül. A 94 vevőmultiplexer meghatározza, hogy az 59B buszról vagy 88 szinkronlogikából kell-e adatot betölteni a 108 regiszterbe. Hasonló módon, az 54A adatátviteli

vonalon a 40 kapcsolómátrixtól vett adat áthalad a 80 regiszteren, a 76 merge-logikán, amely kiküszöböli a blokk-kód hibákat, a 72 átviteli multiplexeren a 70 átviteli regiszterig, amely az adatokat az 59A buszra juttatja.

5 Megjegyezzük, hogy az átviteli oldalon mind a 74 foglalt/reject (elutasít) logika, mind pedig a dekódoló és hibafelismerő 84 logika a hibás állapotok kiszűrésére van beiktatva. A 74 foglalt/reject logika határozza meg, hogy foglaltjelzés érkezett-e az 52 vezérlőbuszon és ha igen, foglalt keretjelt állít elő az 59A buszon. A 86 keretpuffer előzőleg meghatározott, bizonyos hibaállapotokat jelző keretek átvitelét szolgálja.

A 4. ábrán a 40 kapcsolómátrix belső felépítésének blokkvázlata látható. Az ábrán bemutatott módon az 52 vezérlőbuszra összekapcsoló/szétválasztó 600 logika és 602 címtároló kapcsolódik, amelyek egymással is össze vannak kötve. A 602 címtárolóhoz 614 és 624 regiszteren keresztül 608 és 620 adatkiválasztó áramkör csatlakozik. A 608 és 620 adatkiválasztó áramkörök 606 és 618 belső buszon, valamint 604 és 616 bemeneti puffereken keresztül a bemenő adatátviteli 54A vonalhoz, 610 és 622 belső buszon, valamint 612 és 626 kimeneti puffereken keresztül pedig a kimenő adatátviteli 54B vonalhoz csatlakoznak. A 608 és 620 adatkiválasztó áramkörök a párhuzamosított belső buszon keresztül egymással is össze vannak kötve.

Az 5A ábra egy esemény diagram, amely az A és B port közötti kapcsolatot írja le. Az ábra szerint a keretinformáció először a buszon, pl. 59B buszon jelenik meg, amit az egyik port vesz a 120 esemény során. A 122 esemény során a portlogika megvizsgálja a keretet, létrehozza a kapcsolatot, és a 124 esemény során kiválasztja az 52 vezérlőbuszt. A portkiválasztó 38 buszvezérlő a 126 esemény során veszi a buszhasználat-kérést és a 128 esemény során engedélyezi azt. Ekkor az A port logika a 130 esemény alatt kapcsolatkerést küld, amely tartalmazza az 52 vezérlőbuszra kapcsolandó port címét a 132 esemény során. A 4. ábrán látható mátrix 600 logika észleli ezt a kérést a 134 esemény során és a 602 címtároló (latch) eltárolja a port címét. Ezt az eseményt a B port a 136 esemény alatt látja, mire 142 esemény során választ küld, amit a mátrix 600 logika a 140 esemény során észlel az 52 vezérlőbuszon, amint azt a 138 esemény szemlélteti. Ezt a választ az A port a 144 esemény során olvassa ki. A jelen példában egy sikeres kapcsolatfelépítés látható, így a mátrix 600 logika a 602 címtárolóból 614 és 624 regiszterbe tölti, miáltal engedélyezi a 608 és 620 adatkiválasztó áramköröket és összeköti egymással a 606 és 622 belső buszokat. Az A port logika ezután szinkronizáló (handshake) jeleket küld B port felé a mátrix buszon keresztül, például az 54A és 54B adatátviteli vonalon. Először a kimenő szinkronjeleket (handshake out) adják ki a portok a 152 és 154 események alatt, majd pedig a 156 és 158 esemény során a bejövő szinkronjeleket (handshake in) adják ki a másik port felé. Vegyük észre, hogy a mátrix logika automatikusan kapcsolta össze az A és B portot a 40 kapcsolómátrixon keresztül. Végül a 160 esemény alatt a mátrix buszra jutó keretinformáció a 162 esemény során a mátrix bemeneti vonalon a B portra kerül a 166

esemény során, ahol a port logika megvizsgálja a keretet (164 esemény). Ez a keret azután a 168 esemény során a kimenő vonalon az összekapcsolt készülékre jut.

Az 5B ábrán egy másik eseménydiagram látható, amely a szétkapcsolási műveletet ábrázolja. Ennél a példánál az A port szétkapcsoló keretet kap a hozzákapcsolt készüléktől a 200 esemény során, amely a mátrix kimeneti buszra kerül a 202 esemény alatt. Ezt a B port a 208 esemény során veszi, majd pedig a logika a 206 esemény során ellenőrzi a keretet és az összekapcsolt készülékre adja a 204 esemény során. A B port logikája azután a 212 esemény alkalmával lekérdezi a vezérlőbuszt, amit a portkiválasztó 38 buszvezérlő a 210 esemény során vesz és 214 esemény alatt engedélyezi a hozzáférést. A B port logikája ezután a 222 esemény során szétkapcsoló utasítást ad ki, ami a vezérlőbuszon a 218 esemény alatt, a mátrix logikánál 220 esemény alatt, az A portnál pedig 216 esemény alatt látható. Ezt követően az A és B port szinkronizáló jelet küld a vezérlőbuszra a 232 és 224 esemény alatt, a mátrix bemeneti vezetékeken a 228 és 236 események során, valamint a mátrix kimeneti vezetékeken a 226 és 234 események során. A folyamat lényege, hogy a 40 kapcsolómátrix automatikusan szétkapcsolja az A és B portot a 230 esemény során, miután ellenőrizte a vezérlőbuszt és sikeresen kiadott megszakító utasítást látott.

Ezek után a szakértő számára érthető, hogy a buszt ellenőrizve és ott összekapcsoló és szétkapcsoló utasításokat észlelve, nincs szükség további busz ciklusokra a kapcsoló vezérléséhez, bár a kapcsoló úgy működik, hogy önálló kapcsolatot tart fenn a portok között.

A 6. ábrán látható folyamatábra a 78 állapotlogika működését írja le, amikor az egy információs keretet vesz. A 300 blokkban egy információs keret vétele történt meg a port vonali oldalához csatlakozó készüléktől. A logika a 302 lépésben először azt ellenőrzi, hogy a keret egy létező kapcsolathoz érkezett-e. Amennyiben az előző keret már létrehozott egy kapcsolatot és ez a keret egy keretsorozat egyik eleme, akkor a 304 lépésben a keret átjut a létező kapcsolaton a mátrix buszon és a kapcsolómátrixon. A vezérlő logika ezután visszatér a 300 lépéshez és várakozik a következő keretre. Ha azonban a 302 lépés során a logika úgy találja, hogy előzőleg nem jött létre egy kapcsolat, akkor a vezérlő logika megvizsgálja, hogy a 104 puffer megtelt-e. Ha igen, akkor a keretet figyelmen kívül hagyja a 308 lépésben és a vezérlő logika visszatér az újabb keretet váró állapotba. Ha a keret 104 puffer nincs tele, akkor a keret a pufferbe kerül a 310 lépésben és a vezérlő logika vezérlőbusz használatot kér a 312 lépésben. A 314 lépésben a logika az engedély megérkezését várja. Ennek megérkezése esetén 318 lépésben kapcsolatkeres üzenetet küld. A 320 lépésben a vezérlő logika kiolvassa a lekérdezett port választát. A választ a 324 lépésben vizsgálja meg és a 326 lépésben ellenőrzi a foglaltságot és foglaltjelet ad ki a 332 lépésben, vagy pedig a 328 lépésben a hibás állapot detektálása esetén hibás működés üzenetet küld a 334 lépésben. Visszatérve a 324 lépéshez, ha a válasz sikeres, a port kapcsolatba lépett jelzést kap a 330 lépésben

és szinkronjel küldés kezdődik a 336 lépésben a mátrixon keresztül. Ha a 338 lépésben megtörtént a bejövő szinkronjelek vétele, a 340 lépésben sor kerül azok vizsgálatára. Ha a szinkronjel nincs rendben, hibajelzést adunk ki a 342 lépésben, majd pedig a 316 lépésben töröljük a 104 puffert. A 340 lépéshez visszatérve, ha a szinkronjelek vétele rendben megtörtént, elküldjük a keretet a 344 lépésben a 40 kapcsolómátrixhoz, és ezzel eljutunk a 8. ábrán is látható A csomóponthoz, amit a későbbiekben tárgyalunk.

A 7. ábrán a buszról érkező kérést követően a logika működését szemléltető folyamatábra látható, ami a 400 lépésben történik. Ekkor a logika megvizsgálja, hogy a kapcsolat létrejött-e már vagy sem a 402 lépésben. Ha igen, akkor a port a 404 lépésben foglaltjelzéssel válaszol. Ha nem, akkor a 406 lépésben olyan választ ad, hogy képes a kapcsolat létrehozására. A 408 lépésben a port eltárol egy jelzést, hogy csatlakoztatva van, és a 410 lépésben szinkronjelet állít elő. A válaszként érkező szinkronjelet 412 lépésben vesszük és a 414 lépésben megvizsgáljuk, hogy az rendben van-e. Ha nem, akkor a 416 lépésben hibajelzést küldünk és a port a 418 lépésben szétkapcsolt jelzéssel látja el magát, majd pedig visszatér a 400 lépéshez. Ha azonban a 414 lépésben a szinkron válaszjel rendben van, akkor a vezérlő logika továbblép az A csomóponthoz.

A 8. ábrán látható A csomópont a 6. és 7. ábra logikai lépéseinek folytatásaként a 420 lépéshez vezet, amelynek során a keretek átadása történik a mátrixtól a vonal irányában. Szükség esetén a kereteket a vonal irányából is lehet továbbítani a mátrix felé. A 424 lépésben a port logika megvizsgálja, hogy érkezett-e megszakító keret a kapcsolómátrixtól. Ha nem, akkor a port logika 426 lépésben megvizsgálja, hogy érkezett-e megszakító keret a vezérlőbuszról. Ha nem, akkor a port logika visszatér a 420 lépéshez és folytatja a keretek átadását. Visszatérve a 424 lépéshez, ha megszakító keret érkezett a kapcsolómátrixon keresztül, akkor a 428 lépésben megszakítás üzenetet ad ki a vezérlőbuszra. A port ezután szétkapcsolt jelölést kap a 430 lépésben. Hasonlóképpen, ha a 426 lépésben megszakító utasítás érkezik a vezérlőbuszon, a port szétkapcsolt jelölést kap a 430 lépésben.

A 9. ábra a 40 kapcsolómátrix vezérlő logikájának működését szemléltető folyamatábra. Megjegyezzük, hogy a 40 kapcsolómátrix egy kiszolgáló eszköz, amely ellenőrzi a vezérlőbuszt és ennek megfelelően vezérli a kapcsolatok felépítését. Az 500 lépésben a kapcsoló vezérlő logikája meghatározza, hogy van-e kiadott parancs a vezérlőbuszon. Ha nincs, akkor folytatja a várakozást. Ha talál parancsot, akkor a port címetek eltárolja az 502 lépésben. Az 504 lépés során megvizsgálja az utasítást, hogy az egy kapcsolat felépítésére vonatkozik-e. Ha igen, akkor az 506 lépésben figyelmeztet és ellenőrzi a port választát. Ha az 508 lépésben ellenőrzött válasz helyes, akkor az 514 lépésben felépíti a kapcsolatot. Hasonló módon, az 510 lépésben megvizsgálja az utasítást, hogy az bontásra vonatkozik-e. Ha igen, akkor a port kapcsolatot megszakítja az 512 lépésben.

Szakember számára nyilvánvaló, hogy a kapcsoló-

mátrix ellenőrző logikája a készülékek összekapcsolásán és szétkapcsolásán kívül más funkciók vezérlésére is alkalmazható. Például, a kapcsolómátrix ellenőrző logikája meghatározhatja a két port kapcsolatára vonatkozó információk vizsgálatával, hogy egy adott esemény bekövetkezett-e. Így például felügyelheti az önálló készülékek működését és megakadályozhatja két egymást követő kapcsolat létrehozását ugyanahhoz a porthoz, vagy a kapcsolatban nem lévő port szétkapcsolását.

A találmány kiviteli példák alapján történő ismertetése nem jelenti azt, hogy a találmány az ismertetett kiviteli példákra korlátozódna. A bemutatott kiviteli alak számos módosítása és így további kiviteli alakok is nyilvánvalóak a szakember számára a találmány leírása alapján. Ezért, szándékunk szerint, az igénypontok határozzák meg a tényleges oldalmi kört, ami kiterjed az ilyen jellegű módosításokra és további kiviteli alakokra is.

#### SZABADALMI IGÉNYPONTOK

1. Adatátviteli rendszer, amelynek kapcsolómátrixa (40), vezérlő bemenetekkel/kimenetekkel ellátott N portja (30–36, 42–48) van, ahol a portok vezérlő bemeneteit/kimeneteit vezérlőbusz (52) köti össze egymással és a kapcsolómátrix vezérlőbemenetével, továbbá az i-ik ( $i=1, \dots, N$ ) port bemenete és kimenete a kapcsolómátrix (40) i-ik kimenetéhez és bemenetéhez kapcsolódik, *az-*

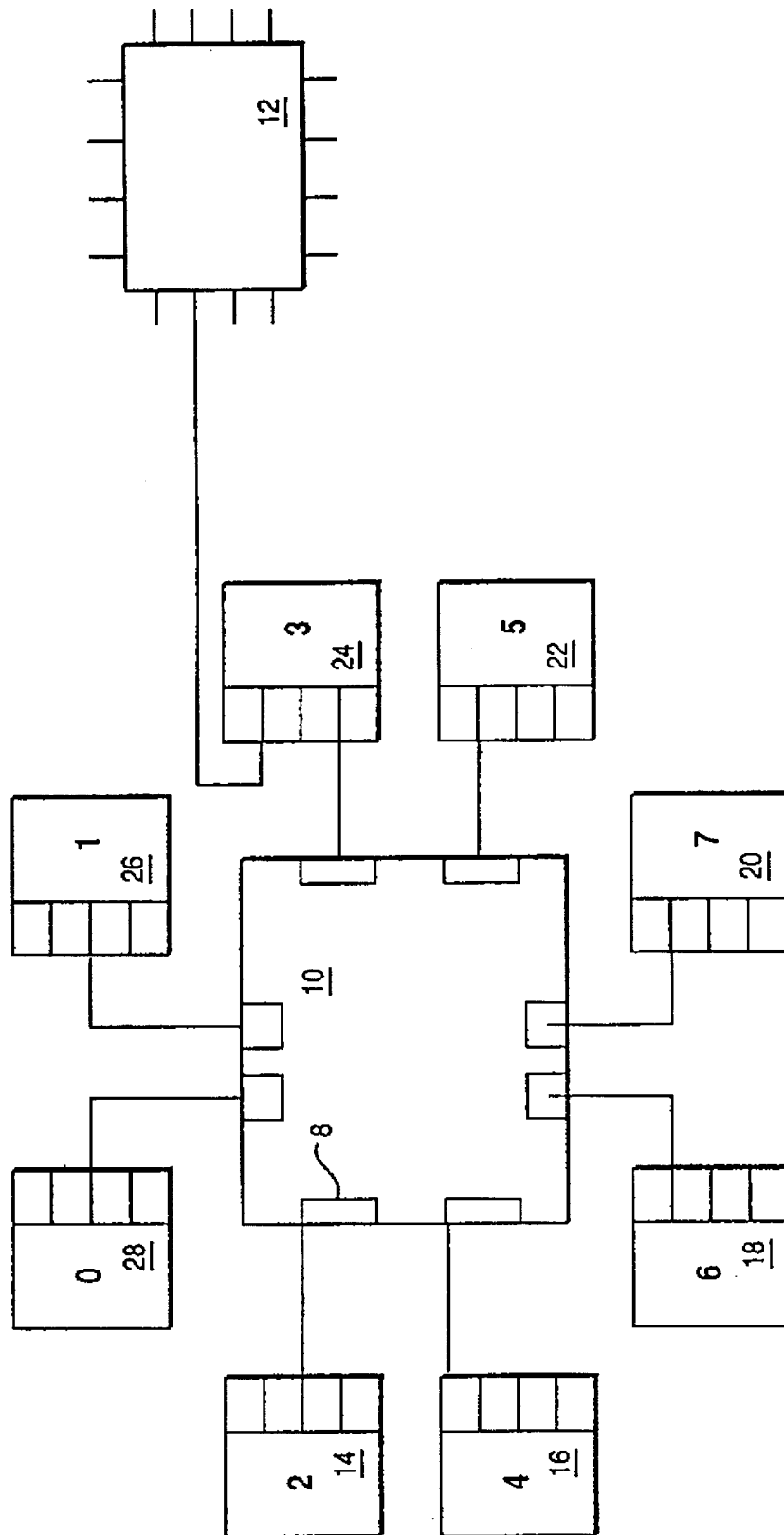
*zal jellemezve*, hogy a kapcsolómátrix (40) összekötő/szétkapcsoló logikát (600) tartalmaz, ahol a logika (600) bemenete egyben a kapcsolómátrix (40) vezérlőbemenete, amely címtároló (602) vezérlő bemenetére van kötve.

2. Az 1. igénypont szerinti rendszer, *azzal jellemezve*, hogy a kapcsolómátrix (40) N adatkiválasztó áramkört (608, 620) is tartalmaz, amelyek vezérlő bemenetei a címtároló (602) kimeneteihez csatlakoznak.

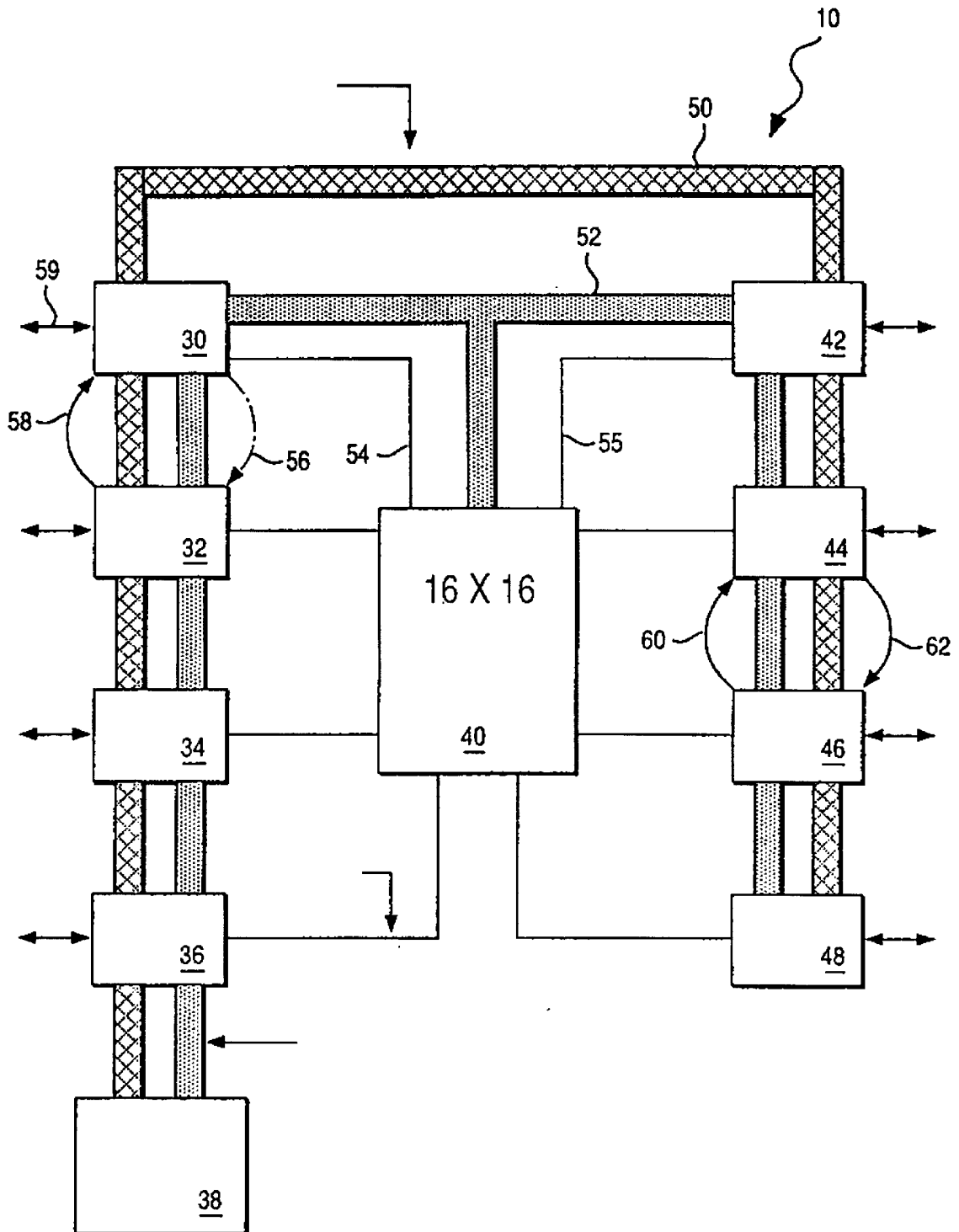
3. A 2. igénypont szerinti rendszer, *azzal jellemezve*, hogy a kapcsolómátrix (40) N bemeneti pufferrel (604, 616) is rendelkezik, ahol az i-ik puffer bemenete a kapcsolómátrix i-ik bemenetéhez csatlakozik, és a puffer kimenete (606, 608) mindegyik adatkiválasztó áramkör (608, 620) egy-egy bemenetére rá van kötve.

4. A 3. igénypont szerinti rendszer, *azzal jellemezve*, hogy a kapcsolómátrix (40) N kimeneti pufferrel (612, 626) is rendelkezik, ahol a pufferek bemeneteikkel az adatkiválasztó áramkörök (608, 620) kimeneteihez csatlakoznak, és az i-ik kimeneti puffer kimenete a kapcsolómátrix i-ik kimenetét képezi.

5. A 2. vagy 4. igénypont szerinti rendszer, *azzal jellemezve*, hogy a kapcsolómátrix (40) N regiszterrel (614, 624) is rendelkezik, ahol az i-ik regiszter bemenete a címtároló (602) kimenetéhez, kimenete pedig az i-ik adatkiválasztó áramkör (608, 620) vezérlő bemenetéhez csatlakozik.

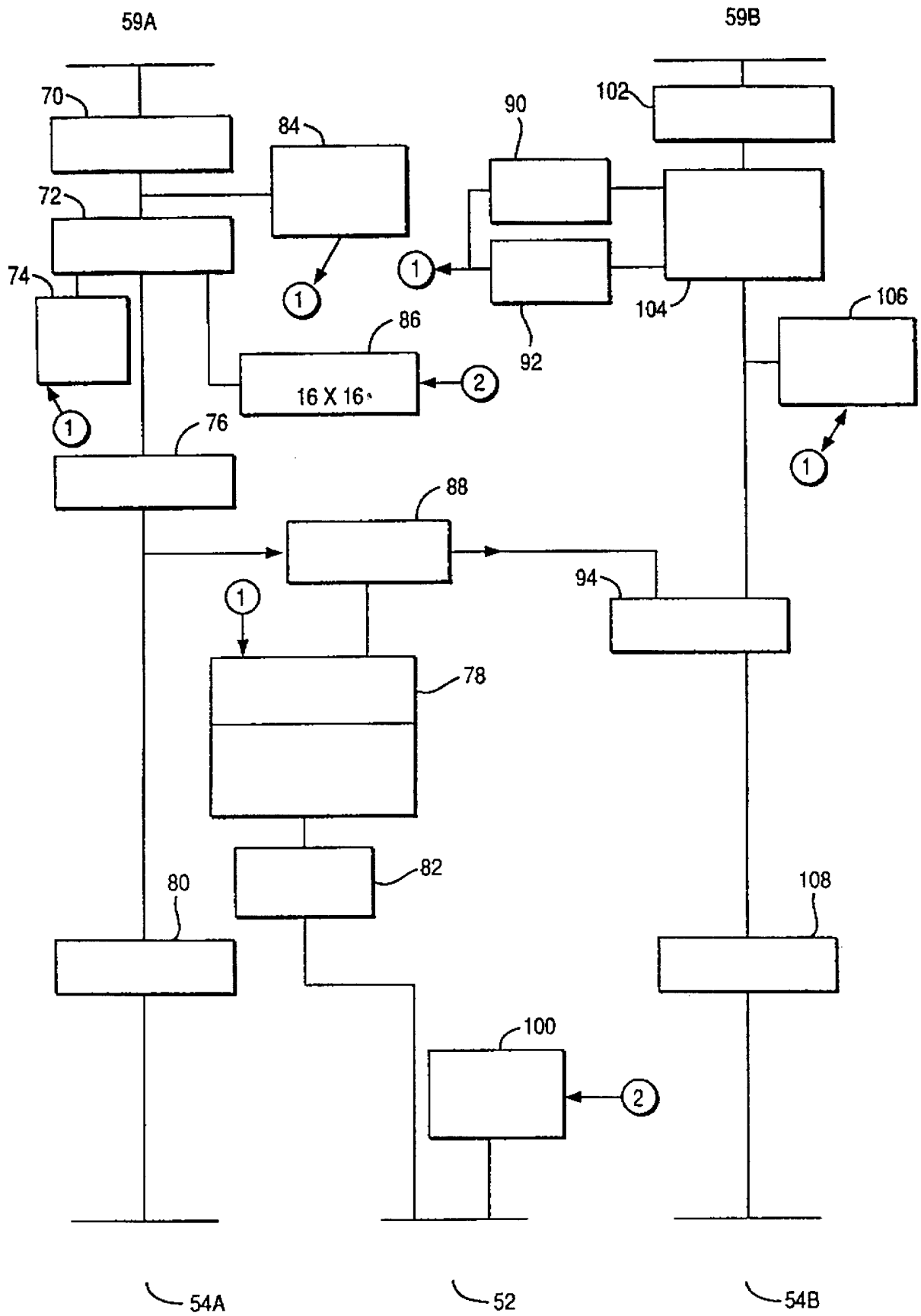


1. ábra

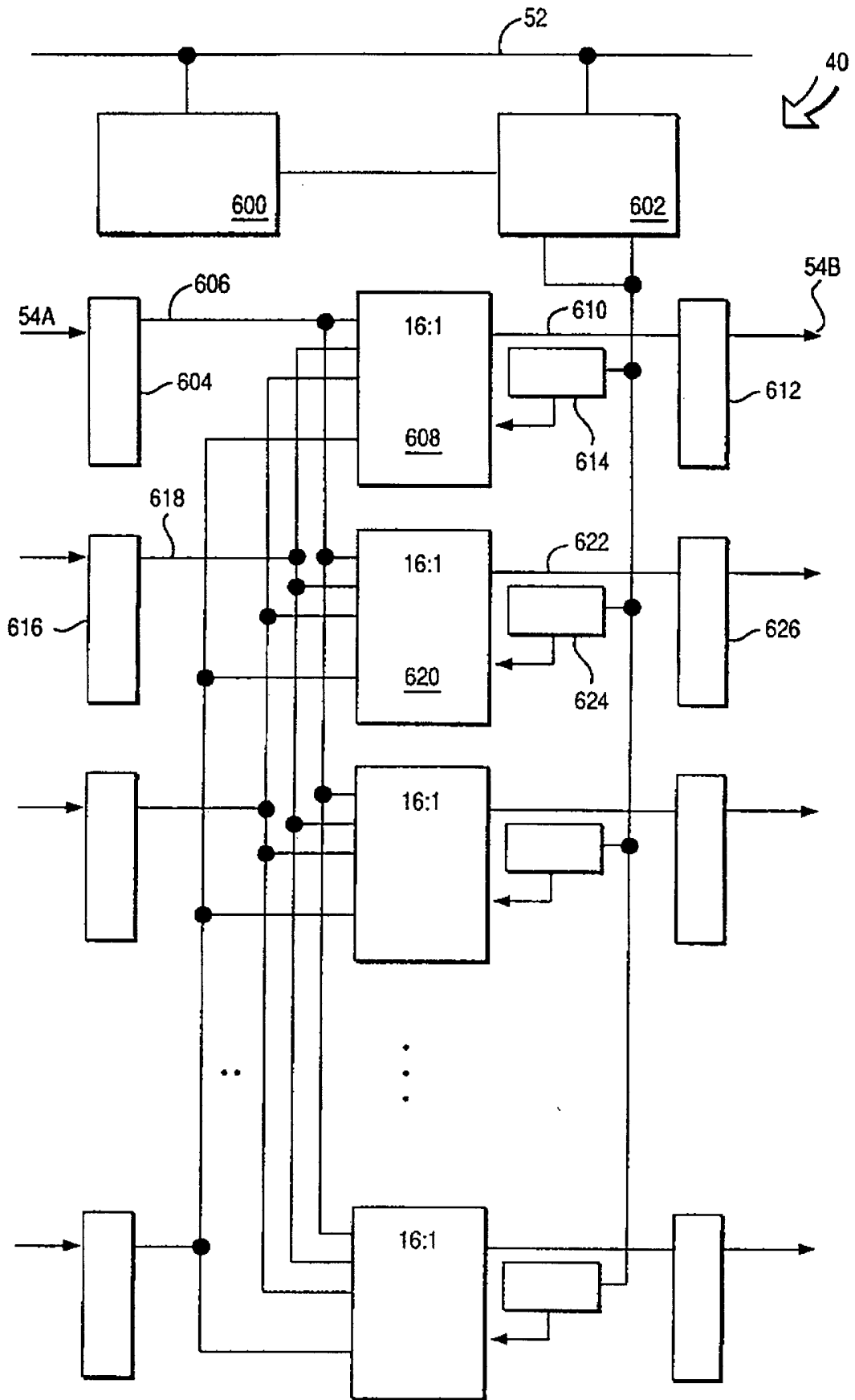


2. ábra

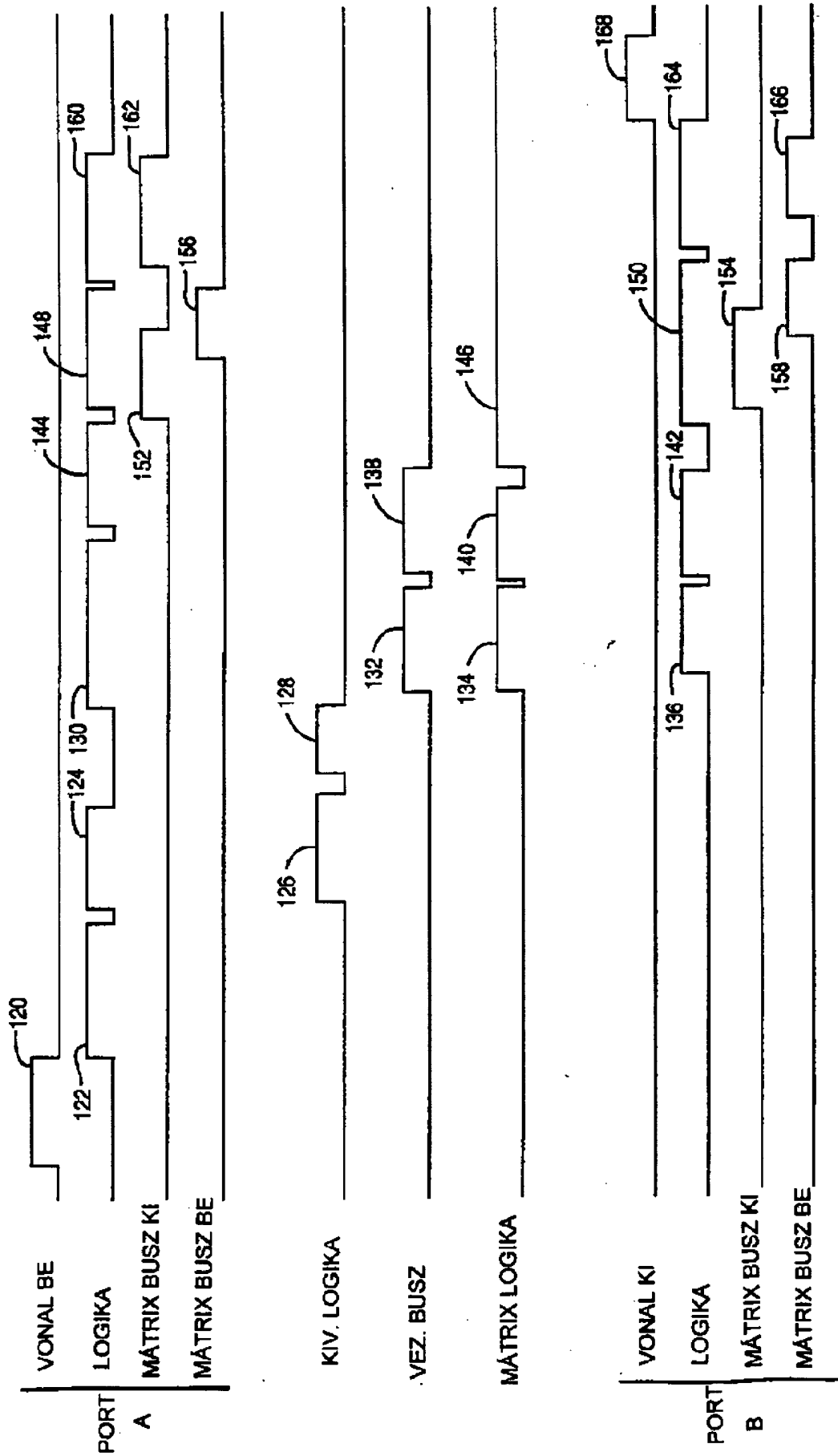




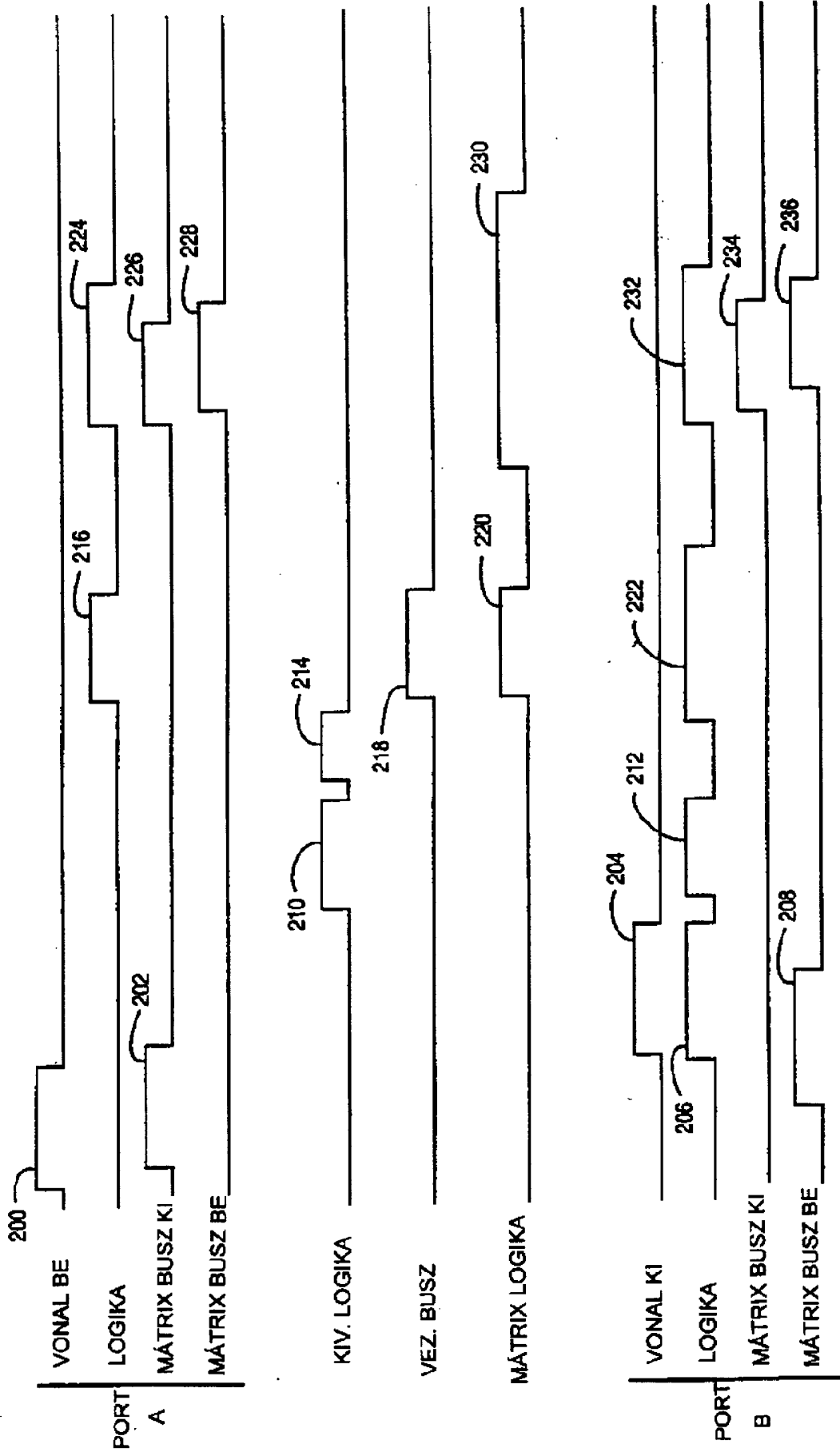
3. ábra



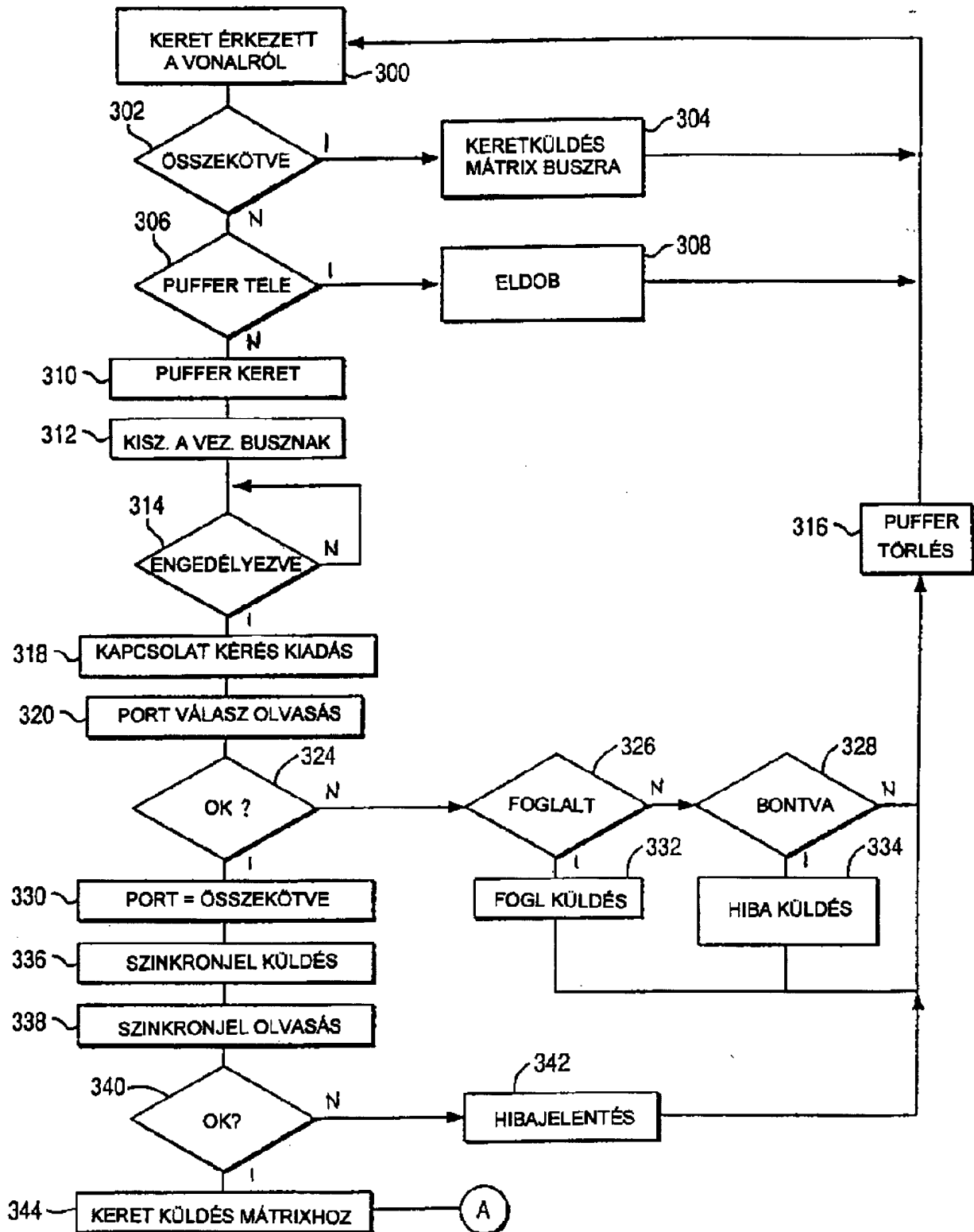
4. ábra



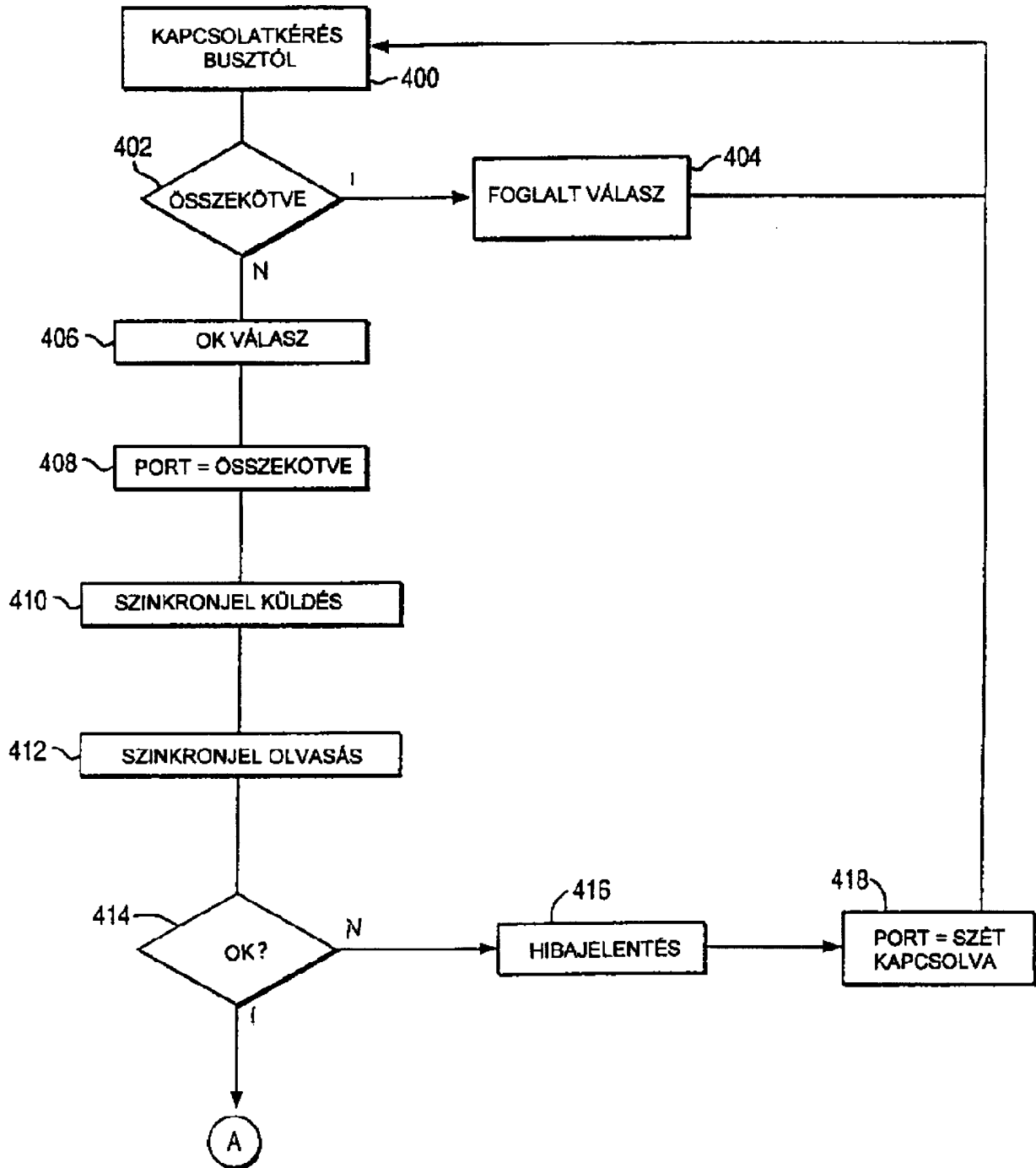
5a. ábra



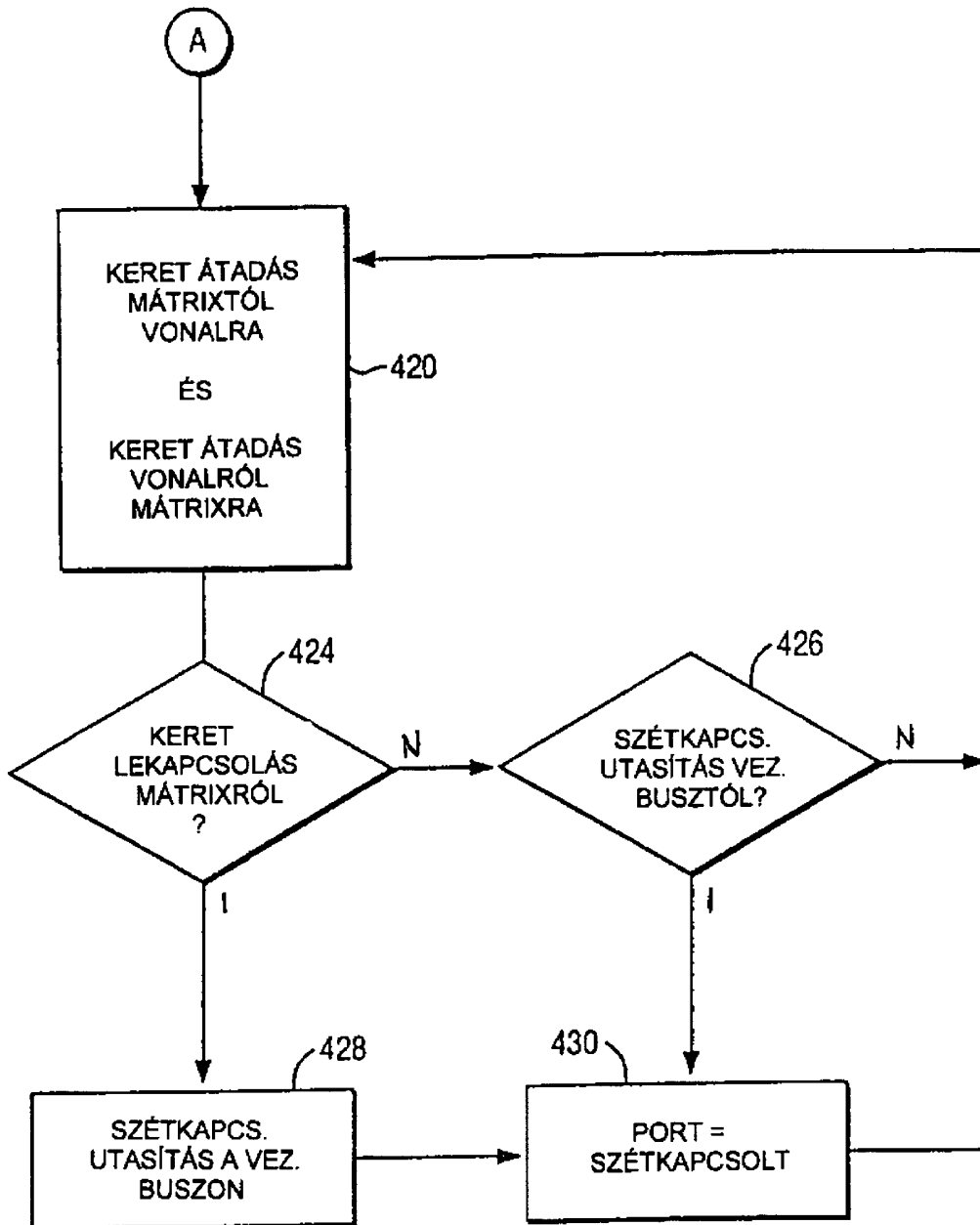
5b. ábra



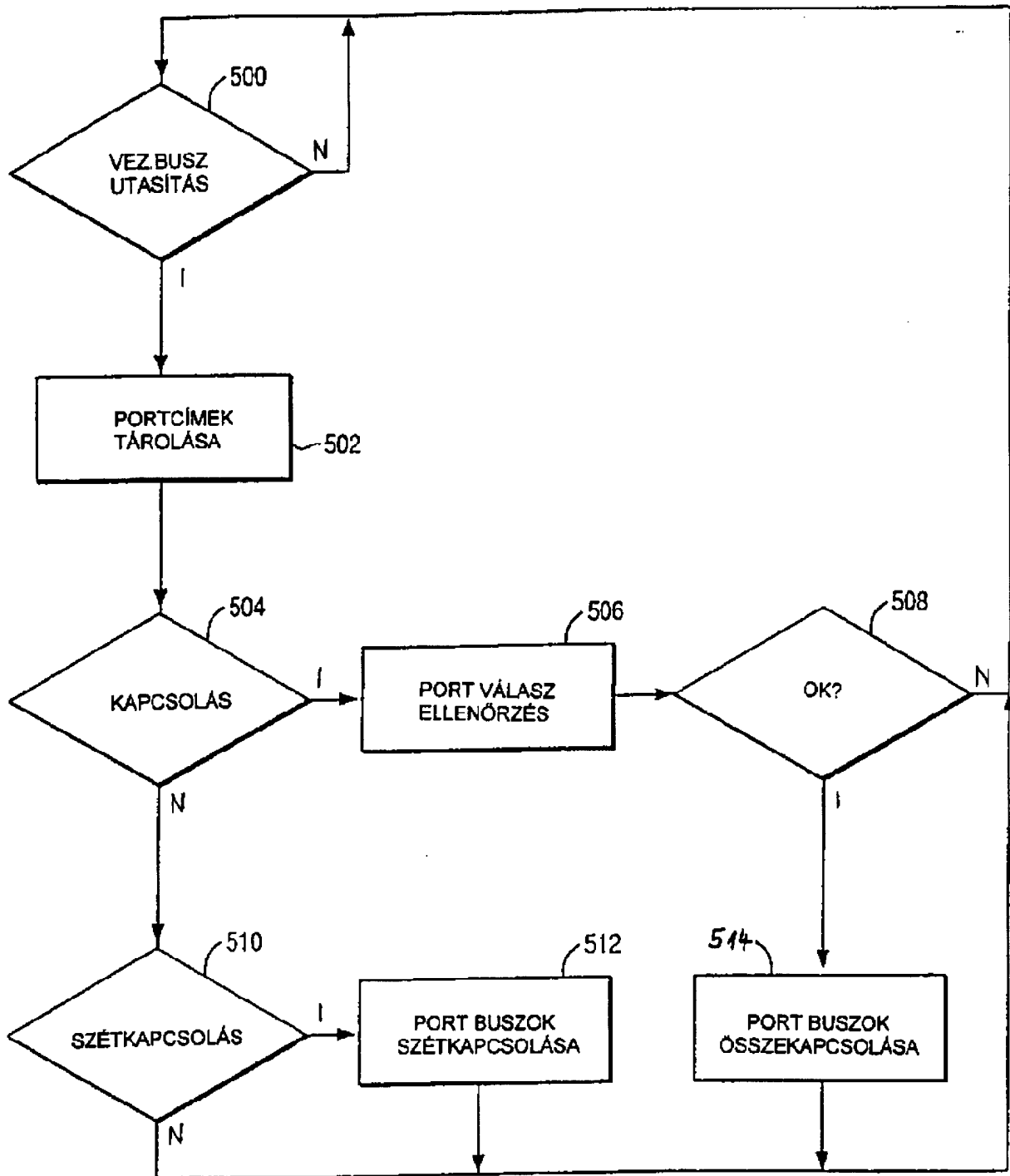
6. ábra



7. ábra



8. ábra



9. ábra