

Eszközmeghajtó és eljárás modemmel folytatott kommunikációra és modem vezérlésére, valamint vevő és dekódoló berendezés**Kivonat**

A találmány tárgya egyrészt egy eszközmeghajtó modemmel folytatott kommunikációra és modem vezérlésére, amely a modemtől üzeneteket vevő és tároló puffertárat, ⁽⁵⁰³⁾ vezérlő paramétereket tároló vezérlőtárat, ⁽⁵⁰²⁾ valamint az eszközmeghajtó vezérlését valamint az üzenetforgalom vezérlését végző logikai fokozatot tartalmaz. A logikai fokozat (501) a vezérlő ^{tárában} ~~memóriában~~ (502) tárolt minták és a puffertárban (503) tárolt üzenetek közötti egyezést kereső, és az eszközmeghajtóhoz csatlakoztatott eszközközkezelőhöz elküldendő jelet előállító összehasonlító fokozatot ~~(520)~~ tartalmaz.

A találmány tárgyát képezi továbbá egy eljárás modemmel folytatott kommunikációra és modem vezérlésére, amelynek során puffertárat ⁽⁵⁰³⁾ és vezérlőtárat ⁽⁵⁰²⁾ tartalmazó eszközmeghajtót használunk úgy, hogy a modemtől üzeneteket veszünk és azokat eltároljuk a puffertárban (503), a vezérlőtárban (502) vezérlő paramétereket tárolunk el, és a vezérlő ^{tárában} ~~memóriában~~ (502) eltárolt mintákat összehasonlítjuk a puffertárban (503) eltárolt üzenetekkel, és mintazonosságot keresünk, hogy az eszközmeghajtóhoz csatlakoztatott eszközközkezelőnek elküldendő jelet állítsunk elő.

A találmány tárgyát képezi továbbá egy olyan vevő és dekódoló berendezés, amely a javasolt eszközmeghajtóval van ellátva.

(3. ábra)

VÁRHEGYI C.

Eszközmeghajtó és eljárás modemmel folytatott kommunikációra és modem vezérlésére, valamint vevő és dekódoló berendezés

A találmány tárgya egyrészt eszközmeghajtó modemmel folytatott kommunikációra és modem vezérlésére, amely a modemtől üzeneteket vevő és tároló puffertárat, vezérlő paramétereket tároló vezérlő tárat, valamint az eszközmeghajtó vezérlését valamint az üzenetforgalom vezérlését végző logikai fokozatot tartalmaz. A találmány tárgya másrészt a javasolt eszközmeghajtót magában foglaló vevő és dekódoló berendezés, valamint harmadrészt egy modemmel kommunikációt folytató és modemet vezérlő eljárás, amelynek során puffertárat és vezérlőtárat tartalmazó eszközmeghajtót használunk.

Jelen találmányunk különösen előnyösen alkalmazható olyan műsorszóró digitális televízió rendszerben, amelyben a vett jeleket egy vevőberendezésen át előbb egy vevő és dekódoló berendezéshez, majd abból egy televízió készülékhez továbbítjuk.

A "vevő és dekódoló berendezés" kifejezés alatt leírásunkban egyrészt olyan vevőberendezést értünk, amellyel kódolt vagy kódolatlan jeleket, például televízió és/vagy rádió jeleket vehetünk, amelyeket hagyományos műsorszórással vagy más módon juttatnak el a végfelhasználókhoz. Ez a kifejezés a vett jelek dekódolására szolgáló dekódoló berendezést is magában foglalja, ismert módon az ilyen vevő és dekódoló berendezések tartalmazhatnak a vevővel egybeépített, a vett jelek dekódolására szolgáló dekódert, amelyet akár egy "Set-Top-Box"-ban, vagyis a többi ismert vételi modultól különállóan, de azok közelében helyeznek el. Egy ilyen dekóder egy tőle fizikailag különálló vevővel együtt működhet, de tartalmazhat más, járulékos funkciókat is, például az Internet használatát jelentő és lehetővé tevő Web böngészőt, videomagnetofont, vagy televízió készüléket.

A digitális jelátviteli rendszerek fejlődése lehetővé tette ezen rendszerek más célokra történő felhasználását is. Egy ilyen lehetséges felhasználás interaktív kapcsolatot tesz lehetővé a végfelhasználóval.

Leírásunkban a "digitális jelátviteli rendszer" kifejezés alatt minden olyan jel átvitelére alkalmas rendszert értünk, amellyel lehetséges például elsődlegesen audiovizuális vagy multimédiás digitális adatok átvitele vagy műsorszórása. Jóllehet, mint jeleztük, találmányunk elsősorban műsorszóró digitális televízió rendszerekben használható, a találmány hasonlóképpen eredményesen alkalmazható vezetékes telekommunikációs hálózatokban, multimédiás Internet alkalmazás céljára, zártláncú televízió rendszerekben, és így tovább. A "digi-

tális televízió rendszer" kifejezés például bármilyen műholdas, földi műsorszóró, kábel vagy egyéb rendszert is magában foglalhat.

Jelen találmányunk különösen előnyösen alkalmazható olyan műsorszóró digitális televízió rendszerben, amelyben a vett jeleket egy vevőberendezésen át előbb egy vevő és dekódoló berendezéshez, majd abból egy televízió készülékhez továbbítjuk. A "vevő és dekódoló berendezés" kifejezés alatt leírásunkban egyrészt értünk olyan vevőberendezést, amellyel kódolt vagy kódolatlan jeleket, például televízió és/vagy rádió jeleket vehetünk, amelyeket hagyományos műsorszórással vagy más módon juttatnak el a végfelhasználókhoz. Ez a kifejezés a vett jelek dekódolására szolgáló dekódoló berendezést is magában foglal, ismert módon az ilyen vevő és dekódoló berendezések tartalmazhatnak a vevővel egybeépített, a vett jelek dekódolására szolgáló dekódert, amelyet akár egy "Set-Top-Box"-ban, vagyis a többi ismert vételi modultól különállóan, de azok közelében helyeznek el. Egy ilyen dekóder egy tőle fizikailag különálló vevővel együtt működhet.

A vevő és dekódoló berendezés egy tömörített MPEG típusú jelet a televízió készülék által értett és feldolgozott televízió jellé dekódol. A vevő és dekódoló berendezést általában kézi távvezérlővel vezéreljük, amely egy megfelelő interfészen keresztül kommunikál a vevő és dekódoló berendezéssel.

A fent vázolt interaktivitás megvalósításának az egyik lehetséges módja, hogy egy megfelelő alkalmazást futtatunk abban a vevő és dekódoló berendezésben, amelyen keresztül a televíziójelet vesszük. Célszerű, ha különböző alkalmazások számára lehetővé tesszük, hogy különböző fizikai eszközökkel kommunikáljanak lehetőleg teljesen transzparens módon. A WO 98/43172 és a WO 98/43433 számú szabadalmi leírások olyan rendszereket mutatnak be, amelyekben egy vagy több alkalmazást le lehet tölteni egy vevő és dekódoló berendezéssel, amelyekben egy vagy több alkalmazást le lehet tölteni egy vevő és dekódoló berendezéssel és az alkalmazások a vevő és dekódoló berendezés egyes fizikai eszközeivel, például párhuzamos interfészével, soros interfészével, kártyaolvasóival kommunikálnak az egyes eszközökhöz készített eszközmeghajtók, valamint úgynevezett általános eszközközkezelő segítségével.

Születtek javaslatok arra, hogy a vevő és dekódoló berendezést alkalmassá kell tenni arra, hogy különböző jelcsatornákkal tudjanak kapcsolódni és kommunikálni, mint amilyen a modem, a soros csatorna, párhuzamos csatorna, MPEG csatorna (tömörített és kódolt videójelű csatorna), kártyaolvasók, és így tovább. A vevő és dekódoló berendezés egy olyan

virtuális gépet tartalmaz, amely főalgoritmust foglal magában. A virtuális gép olyan eszközkezelőhöz kapcsolódik, amely a különböző csatornák fizikai interfészeihez van eszközök és eszközmeghajtók révén csatlakoztatva.

Az "MPEG" rövidítés egy adatátviteli szabványt jelent, amelyet a Nemzetközi Szabvány Szervezet (ISO, International Standard Organization) "Motion Pictures Expert Group" elnevezésű munkacsoportja fejlesztett ki, és az ő nevéhez fűződik elsősorban de, nem kizárólagosan az MPEG-2 jelzésű szabvány kidolgozása is, amelyet kimondottan a digitális televízió alkalmazások céljára hoztak létre, és az "ISO 13818-1, ISO 13818-2, ISO 13818-3 és ISO 13818-4" jelzésű dokumentumokban rögzítettek. Találmányunk jelen leírásában az MPEG-2 elnevezés alatt az említett szabvány összes változatait, módosításait vagy továbbfejlesztéseit is értjük, melyek alkalmasak a digitális adatátvitel területén valamilyen felhasználásra.

Mint fent már jeleztük, a csatornák egyike előnyösen modem. Találmányunk fő célja éppen ezért egy ilyen modem vezérlésének, vezérlési lehetőségeinek a bővítése és javítása. Ezt egy modem vezérlésére alkalmas, javított tulajdonságokkal rendelkező eszközmeghajtó révén érjük el. Céljainkhoz az eszközök és eszközmeghajtók közötti bármilyen megkülönböztetés nem lényeges, és az "eszközmeghajtó" kifejezést, amelyet gyakrabban, néha eszköz helyett is használunk, úgy kell érteni, mint a hardver és egy alkalmazás közötti interfész bármely megnyilvánulási formáját, amíg ennek az ellenkezőjét külön és nyomatékosan nem jelezzük.

A kitűzött feladatot egyrészt eszközmeghajtóval oldottuk meg, modemmel folytatott kommunikációra és modem vezérlésére, amely a modemtől üzeneteket vevő és tároló puffertárat, vezérlő paramétereket tároló vezérlő tárat, valamint az eszközmeghajtó vezérlését valamint az üzenetforgalom vezérlését végző logikai fokozatot tartalmaz. Ezt a találmány értelmében úgy fejlesztettük tovább, hogy a logikai fokozat a vezérlő memóriában tárolt minták és a puffertárban tárolt üzenetek közötti egyezést kereső, és az eszközmeghajtóhoz csatlakoztatott eszközkezelőhöz elküldendő jelet előállító összehasonlító fokozatot tartalmaz.

Egy eszközkezelő általában több eszközmeghajtó vezérlésére szolgál, és az eszközmeghajtók és egy vagy több alkalmazás között továbbít üzeneteket, de jelen leírásunkban az "eszközkezelő" kifejezést arra használjuk, hogy bármely, az eszközmeghajtó vezérlésére alkalmas megoldást jelöljünk vele, így ez maga egy vezérlő alkalmazás is lehet. Találmá-

nyunk további lényeges jellemzői az alábbi részletesebb leírásból és az igénypontokból könnyen megérthetők.

A berendezés legelőnyösebben eszközmeghajtóként valósítható meg egy vevő és dekódoló berendezésben, például olyan digitális jelátviteli rendszer számára, amelyet a WO 98/43430, WO 98/43425, WO 98/43426, WO 98/43162, WO 98/43432, WO 98/43431, WO 98/43248, WO 98/43165, WO 98/43415, WO 98/43172, WO 98/43433, és a WO 98/43427 számú szabadalmi leírásokból ismerhetünk meg. Ilyen megvalósításban az eszközmeghajtó egy alkalmazás vezérlete alatt képes működni az eszközközkezelő segítségével, amely így kényelmes és rugalmas felügyeletet biztosít az eszközmeghajtó fölött. Az alkalmazás előnyösen interpretált nyelven fut, az eszközmeghajtó pedig előnyösen kompilált.

A találmány szerinti eszközmeghajtó egy előnyös kiviteli alakja értelmében az összehasonlító fokozat az eltárolt mintákat az üzenet előre meghatározott végleges hosszával összehasonlító fokozat.

A találmány szerinti eszközmeghajtó egy további előnyös kiviteli alakja értelmében a minták különböző típusú minták, valamint azonos típusú több mintából állhatnak.

Ugyancsak előnyös a találmány értelmében, ha a minta típusok esemény, pozitív nyugtázás, és negatív nyugtázás típusok, és az előállított jel az egyező minta típusát jelző jel.

Előnyös továbbá, ha a jelet a modemen keresztül kapott üzenethez hozzáadva az eszközközkezelőnek elküldő eszközmeghajtó.

Fentiekén túlmenően előnyös, ha a parancsok az eszközmeghajtó paramétereit beállító hívások formájában, továbbá az eszközmeghajtóhoz vezérlőjeleket és adatokat küldő I/O műveletek formájában, valamint minták észlelését, vagy üzenet vételét, vagy adatátviteli problémákat az eszközközkezelőnek jelző események formájában kerülnek továbbításra az eszközmeghajtó és az eszközközkezelő között.

Ugyancsak előnyös a találmány értelmében, ha a hívás parancs egyik formája az összehasonlító fokozat által keresendő egy vagy több minta meghatározásához legalább egy minta beállítás parancsot tartalmaz.

Előnyösen a keresendő mintákat meghatározó három minta beállítási parancs altípust tartalmaz, egy minta beállítás parancsot minden egyes minta típushoz.

A találmány szerinti eszközmeghajtó egy további előnyös kiviteli alakja értelmében a legalább egy mintabeállítás parancs több beazonosítandó mintát meghatározó parancs, ahol minden egyes minta a megfelelő minta detektálása esetén előállítandó hozzárendelt esemény jelzést tartalmaz.

Előnyös továbbá, ha a keresendő minták a vezérlőtár alterületein vannak eltárolva.

Fentiekén túlmenően előnyös, ha a keresendő minták a vezérlőtár egyetlen folyamatos területén vannak eltárolva.

Ugyancsak előnyös a találmány értelmében, ha többszörös egyezés esetén csupán az utolsó egyezést veszi figyelembe.

Előnyösen többszörös egyezés esetén a különböző egyezés típusok között előre meghatározott elsőbbségi sorrendnek megfelelően jelzi az egyezés típusát.

Az eszközmeghajtót megvalósíthatjuk hardveresen, például egy erre a célra kifejlesztett integrált áramkörrel, ami igen nagy sebességű működést eredményez. Előnyösebb azonban, ha előnyösen az egy vagy több alkalmazást futtató feldolgozó eszközökkel végrehajtható módon, szoftveresen van megvalósítva, ez nagyobb rugalmasságot tesz lehetővé, kevesebb alkatrész szükséges a megvalósításához és az eszközmeghajtó is lényegesen könnyebben frissíthető.

A kitűzött feladatot ezen túlmenően olyan eljárással oldottuk meg modemmel folytatott kommunikációra és modem vezérlésére, amelynek során puffertárat és vezérlőtárat tartalmazó eszközmeghajtót használunk. Ezt a találmány értelmében úgy fejlesztettük tovább, hogy a modemtől üzeneteket veszünk és azokat eltároljuk a puffertárban, a vezérlőtárban vezérlő paramétereket tárolunk el, és a vezérlő memóriában eltárolt mintákat összehasonlítjuk a puffertárban eltárolt üzenetekkel, és mintaazonosságot keresünk, hogy az eszközmeghajtóhoz csatlakoztatott eszközkezelőnek elküldendő jelet állítsunk elő.

A találmány szerinti eljárás egy előnyös foganatosítási módja értelmében az eltárolt mintákat az üzenet előre meghatározott végleges hosszával hasonlítjuk össze.

A találmány szerinti eljárás egy további előnyös foganatosítási módja értelmében a minták különböző típusúak, és lehetnek azonos típusú minták is.

Ugyancsak előnyös a találmány értelmében, ha a minta típusok esemény, pozitív nyugtázás, és negatív nyugtázás típust tartalmaznak, és az előállított jellel az egyező minták típusát jelezzük.

Előnyösen az eszközmeghajtóval a jelet úgy küldjük el az eszközközkezelőnek, hogy azt a modemén keresztül kapott üzenethez hozzáadjuk.

Előnyös továbbá a találmány értelmében, ha a parancsokat az eszközmeghajtó és az eszközközkezelő között hívások formájában továbbítjuk, amelyekkel paramétereket állítunk be az eszközmeghajtóba, továbbá I/O műveletek formájában továbbítjuk, amelyekkel vezérlőjeleket és adatokat küldünk az eszközmeghajtókhoz, valamint események formájában továbbítjuk, amelyekkel mintadetektálást, vagy üzenetvételt, vagy adatátviteli problémákat jelzünk az eszközközkezelőnek.

Fentiekén túlmenően előnyös, ha a hívás parancs egyik formája legalább egy mintabeállítás parancsot tartalmaz, amellyel a vezérlőtárban keresendő egy vagy több mintát határozunk meg.

Ugyancsak előnyös a találmány értelmében, ha keresendő mintákat meghatározó három minta beállítási parancs altípust tartalmaz, egy minta beállítás parancsot minden egyes minta típushoz.

A találmány szerinti eljárás egy további előnyös foganatosítási módja értelmében a legalább egy mintabeállítás parancs több beazonosítandó mintát meghatározó parancs, ahol minden egyes minta a megfelelő minta detektálása esetén előállítandó hozzárendelt esemény jelzést tartalmaz.

Előnyös továbbá, ha a keresendő mintákat a vezérlőtár alterületein tároljuk el.

Fentiekén túlmenően előnyös, ha a keresendő mintákat a vezérlőtár egyetlen folyamatos területén tároljuk el.

Ugyancsak előnyös a találmány értelmében, ha többszörös egyezés esetén csupán az utolsó egyezést vesszük figyelembe.

Előnyös végül, ha többszörös egyezés esetén a különböző egyezés típusok között előre meghatározott elsőbbségi sorrendnek megfelelően jelezzük az egyezés típusát.

A találmányt az alábbiakban a csatolt rajz segítségével ismertetjük részletesebben, amelyen az

1. ábra egy vevő és dekódoló berendezés interfészeinek vázlatos rajza, a
2. ábrán a vevő és dekódoló berendezés működési tömbvázlata látható, a
3. ábra a találmány szerinti eszközmeghajtó általános logikai szervezését mutatja, a
4. ábrán a találmány szerinti eszközmeghajtó mintalekezelő szempontjainak logikai szervezését tüntettük fel részletesebben, és az
5. ábra az üzenet struktúra és tárolás lehetőségét mutatja.

A számítógép rendszer megértésének elősegítésére előbb röviden vázoljuk a digitális műholdas 2020 vevő és dekódoló berendezés főbb jellemzőit.

Áttérve az 1. ábra részletesebb bemutatására, azon egy digitális interaktív televízió rendszerben használatos 2020 vevő és dekódoló berendezés vagy Set-Top-Box látható, amelyben a találmány szerinti eszközmeghajtókat telepítettük. Egy megfelelő digitális interaktív televízió rendszer jellemző részletei megtalálhatók a WO 98/43430, WO 98/43425, WO 98/43426, WO 98/43162, WO 98/43432, WO 98/43431, WO 98/43248, WO 98/43165, WO 98/43415, WO 98/43172, WO 98/43433, WO 98/43427 számú szabadalmi leírásokban, melyek kitanítását leírásunkban referenciaként vesszük. A könnyebb megértés érdekében az említett szabadalmi leírásokban szereplő részleteket jelen leírásunkban azonos néven és ugyanazzal a hivatkozási jellel jelöltük.

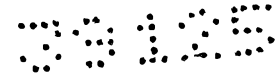
Mint az a többi szabadalmi leírásban részletesebben is megtalálható, a 2020 vevő és dekódoló berendezés több portot tartalmaz, nevezetesen az MPEG jelfolyam számára egy 4028 MPEG áram tunert, egy 4030 soros interfészt, egy 4032 párhuzamos interfészt és két 4036 kártyaolvasót, melyek közül az egyik a rendszer részét képező programozható csipkártyához tartozik, egy pedig közönséges bankkártyák olvasására szolgál (amelyet a fizetések lebonyolítására, a házibank szolgáltatásokhoz, és így tovább használunk). A 2020 vevő és dekódoló berendezésnek 4034 interfésze is van, egy 4002 modemes válaszcatornához, amellyel a televíziójelcset előállító szolgáltatóhoz lehet visszajelezni, úgy, hogy a felhasználó jelezheti a részére előnyös választásokat, kívánságait és így tovább a televízió műsorok gyártóinál. A 2020 vevő és dekódoló berendezésnek 4008 főalgoritmus, 4068 eszközekezelője valamint több 4062 eszköze és 4066 eszközmeghajtója van, amelyek egy vagy több 4056 alkalmazás futtatását segítik elő, illetve biztosítják (lásd a 2. ábrát).

Leírásunkban alkalmazás alatt egy számítógépi kód készletet értünk, amellyel előnyösen a 2020 vevő és dekódoló berendezés magas szintű funkcióit vezéreljük. Például, ha a felhasználó 2026 távirányítóját, pontosabban annak fókuszált sugarát ráirányítja a 2022 televízió készülék képernyőjén látható gomb objektumra, és megnyomja a 2026 érvényesítés gombját, azzal lefuttatja a képernyőn látható nyomógommbal társított utasítás szekvenciát.

Egy interaktív alkalmazás menüket kínál fel, a felhasználó kérelmére parancsokat hajt végre és az alkalmazás céljához kapcsolódó és illeszkedő adatokat szolgáltat. Az alkalmazások vagy úgynevezett rezidens alkalmazások, amelyek a 2020 vevő és dekódoló berendezés ROM-jában (vagy FLASH, vagy más nem-felejtő tárolójában) vannak eltárolva, vagy a 2020 vevő és dekódoló berendezés RAM-jába (vagy FLASH-ROM-jába) kerülnek kisugárzás és letöltés útján.

Az alábbiakban nem kizárólagosan, csupán példaképpen felsorolunk néhány lehetséges alkalmazást:

- Inicializáló alkalmazás. A 2020 vevő és dekódoló berendezés egy rezidens inicializáló alkalmazással van ellátva, amely a 2020 vevő és dekódoló berendezést az MPEG-2 környezetben azonnal működőképessé tevő modulok illeszthető, adaptálható gyűjteménye.
- Indító alkalmazás. Az indító alkalmazás teszi lehetővé bármely alkalmazás számára, legyen az letöltött vagy rezidens alkalmazás, hogy fusson a 2020 vevő és dekódoló berendezésen.
- Programkalauz. A programkalauz egy olyan interaktív alkalmazás, amely a programozásra nyújt teljes körű információt.
- Nézd-és-fizess alkalmazás. A nézd-és-fizess alkalmazás egy olyan interaktív szolgáltatás, amely a digitális televízió csomag minden egyes nézd-és-fizess csatornáján rendelkezésre áll egy feltételes hozzáférési rendszerrel együttműködésben.
- PC letöltés alkalmazás. Amennyiben kívánja a végfelhasználó ezzel a PC letöltés alkalmazással számítógép szoftvert tud letölteni.
- Magazin böngésző alkalmazás. A magazin böngésző alkalmazás ciklikusan ismétlődően képeket sugároz, amelyeket a végfelhasználó a képernyőn megjelenő gombokkal tud navigálni.



- Távvásárlás alkalmazás. A távvásárlás alkalmazás egy lehetséges megvalósításánál eladásra kínált árukról szóló ajánlatokat továbbítanak a 2020 vevő és dekódoló berendezéshez, mely ajánlatok megjelennek a végfelhasználó 2022 televízió készülékén. A 2026 távirányító használatával a végfelhasználó kiválaszthat egy megfelelő árucikket, amelyet meg kíván vásárolni.

Az alkalmazásokat a 2020 vevő és dekódoló berendezés memóriahelyein tároljuk és ezek erőforrás fájllokként és adatokként jelennek meg. Az erőforrás fájlok grafikus objektum leíró egység fájllokból, változó blokk egység fájllokból, utasítás sorrend fájllokból és alkalmazás fájllokból állnak, amint az az említett szabadalmi dokumentumokból részletesebben is megismerhető.

Egy MPEG adatáramban minden egyes modul MPEG táblázat csoportot tartalmaz. Minden egyes MPEG táblázat szekciókként formázható. Az MPEG adatáramban minden egyes szekciónak maximum 4 kB nagyságú "mérete" van. Ahhoz, hogy például a soros vagy párhuzamos porton keresztül adatátvitelt valósíthassunk meg, a modulokat hasonlóképpen táblázatokká és szekciókká kell felbontanunk, ahol a szekciók mérete a mindenkori adatátvitelhez használt vagy adatátvitelre szolgáló közegetől függ.

A modulokat az MPEG adatáramban jellemzően 188 bájtos adatcsomagok alakjában szállítjuk a megfelelő típusú adatáramban, például video adatáramban, audio adatáramban, vagy teletext adatáramban. Minden egyes csomagot egy 13 bites csomag azonosító (PID, Packet Identifier) előz meg, még hozzá minden egyes az MPEG adatáramban továbbított csomaghoz egy-egy csomag azonosító tartozik. A különböző adatáramok listáját egy programozható térkép táblázat (PMT, Program Map Table) tartalmazza, amely a hozzátartozó csomag azonosító szerint meghatározza az egyes adatáramok tartalmát is. Egy csomag azonosító felhívhatja egy adott eszköz figyelmét arra, hogy az adatáramban számára hasznos alkalmazás található, úgy, hogy a csomag azonosítót a program térkép táblázat felhasználásával azonosítjuk.

A 2020 vevő és dekódoló berendezés dekódere olyan memóriát tartalmaz, amely egy írható olvasható RAM részre, egy meghatározott módon frissíthető FLASH részre, valamint egy csak olvasható ROM részre van felosztva, de ez a fizikai szervezés különbözik a logikai szervezéstől. A memóriát továbbá a különböző interfészekkel társított memória kötetekre oszthatjuk. Ha úgy vesszük, a memóriát a hardver részeként tekinthetjük egy másik szem-

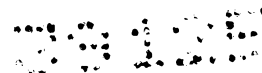
pontból viszont a memóriát a hardveren kívül eső, a rendszer egészét támogató vagy tartalmazó elemként is vehetjük.

A rendszert úgy is tekinthetjük, mint egy, a 4007 virtuális gép részét alkotó 4008 főalgoritmusra központosított elrendezést. A 4007 virtuális gép egyrészt alkalmazásokhoz kapcsolódik (ez az úgynevezett "magas szintű" oldal), másrészt különböző közbenső logikai egységekkel, amelyek még részletesebben is bemutatunk, 4061 hardverhez kapcsolódik (ez pedig az úgynevezett "alacsony szintű" oldal), amely a 2020 vevő és dekódoló berendezéshez tartozik. A 2020 vevő és dekódoló berendezés 4061 hardverjét úgy tekinthetjük, mint amely tartalmazza a fent felsorolt és vázolt különböző portokat (azaz a 2026 távirányítóhoz tartozó 2030 interfészt, a 4028 MPEG áram tunert, a 4030 soros interfészt, a 4032 párhuzamos interfészt, a 4036 kártyaolvasók interfészeit valamint a 4002 modemes vonal 4034 interfészét).

A 2. ábrán megfigyelhető, hogy a 4007 virtuális géphez 4057 alkalmazások kapcsolódnak, a legáltalánosabban használt 4057 alkalmazások némelyike többé-kevésbé permanensen eltárolható a rendszerben - ezeket 4057 alkalmazásokként jelöltük -, míg a többi 4056 alkalmazást, például az MPEG adatáramból vagy más portokról, interfészekről szükség szerint tölthetjük le a rendszerre.

A 4007 virtuális gép a 4008 főalgoritmuson kívül néhány rezidens alkalmazás- és adatszerver 4006 könyvtár funkciót is tartalmaz, melyek 4058 eszközkészletet foglalnak magukban. Az alkalmazás- és adatszerver 4006 könyvtár különböző funkciókat tartalmaz, amelyeket C nyelven írtak és a 4008 főalgoritmus használja őket. Ezek a funkciók magukban foglalják az adatokkal való különböző manipulációkat, például az adatok tömörítését, kibontását, az adatstruktúrák összehasonlítását, vonal vagy sor rajzolást, és így tovább. A 4006 alkalmazás- és adatszerver információt is tartalmaz a 2020 vevő és dekódoló berendezés 4061 hardveréről, például a szoftver és hardver verziószámokat, a rendelkezésre álló RAM területet, valamint egy olyan funkciót, amelyet akkor használunk, amikor egy új 4062 eszközt töltünk le. A funkciókat letölthetjük a könyvtárba, és a FLASH vagy a RAM memóriában tárolhatjuk el.

A 4008 főalgoritmus 4068 eszközközkezelőhöz kapcsolódik, és a 4068 eszközközkezelő 4062 eszközök egy olyan készlethez van csatlakoztatva, amelyek 4060 eszközmeghajtói a portokhoz vagy interfészekhez kapcsolódnak. Nagy vonalakban egy 4060 eszközmeghajtót logikai interfészként határozhatunk meg, és ennek megfelelően két különböző 4060 eszköz-



meghajtó ugyanahhoz a fizikai porthoz is csatlakozhat. Egy 4062 eszköz általános esetben egynél több 4060 eszközmeghajtóhoz kapcsolódik; ha egy 4062 eszköz csupán egyetlen 4060 eszközmeghajtóhoz van csatlakoztatva, akkor a 4062 eszköz általában úgy van kialakítva, hogy tartalmazza a kommunikációhoz szükséges összes funkciót és képességet, úgy, hogy emiatt feleslegessé válik egy vagy több külön további 4060 eszközmeghajtó használata. Néhány 4062 eszköz egymás közötti kommunikációt is folytathat.

Az alábbiakban arra mutatunk példát, hogy a 4062 eszközöktől háromféle kommunikáció zajlik a 4008 főalgoritmus irányában: egyrészt változók révén, másrészt pufferek felhasználásával, harmadrészt olyan "események" révén, amelyek egy esemény várakozási sor készletbe kerültek továbbításra.

A 2020 vevő és dekódoló berendezés minden egyes funkcióját egy-egy 4062 eszközként jelöltük. A 4062 eszközök vagy 4064 helyi eszközök, vagy 4066 távoli eszközök lehetnek. A 4064 helyi eszközökbe tartoznak a programozható csipkártyák, a SCART csatlakozó jelek, modemek, soros és párhuzamos interfészek, MPEG video és audio lejátszó és egy MPEG szekció és táblázat kibontó. A 4066 távoli eszközök, amelyek valamilyen távolabbi helyen kerülnek végrehajtásra, abban különböznek a 4064 helyi eszközöktől, hogy a rendszer felügyelőnek vagy tervezőnek inkább egy portot vagy eljárást kell definiálni ahelyett, hogy egy 4062 eszközt definiálna, és a 4060 eszközmeghajtót a 2020 vevő és dekódoló berendezés gyártója tervezi és alakítja ki.

Ha egy új 4062 eszközt hozunk létre, úgy tudjuk installálni a már meglévő 2020 vevő és dekódoló berendezésekbe, hogy a hozzátartozó 4056 alkalmazást a műsorszóró központból letöltjük. Ezt a letöltést a 2020 vevő és dekódoló berendezésben egy olyan 4056 alkalmazás hajtja végre, amely leellenőrzi a hardver és szoftver verziószámokat, és ha azok megfelelőek, az új 4062 eszközt jelképező szoftver modult letölti, majd a 4006 könyvtár egy eljárását meghívja, hogy az telepítse az új eszközkódot a főmverren (= mikroprogramozott illetve ROM-ban tárolt program) belül a FLASH memóriába). Ez úgy tud a 2020 vevő és dekódoló berendezésen belül új funkciókat rugalmasan és biztosan telepíteni, hogy a szoftver többi részét nem befolyásolja semmilyen módon.

A 4068 eszközközkezelő egy közös szoftver interfész a 4056 alkalmazás valamint a 2020 vevő és dekódoló berendezés speciális funkciói között. A 4068 eszközközkezelő felügyeli a hozzáféréseket a 4062 eszközökhöz, rögzíti és jelenti valamilyen nem várt esemény bekövetkeztét vagy vételét, és gondoskodik az osztott memória kezeléséről.

A 4008 főalgoritmus a mikroprocesszor valamint olyan közös alkalmazás programozó felület (API, Application Programming Interface) vezérlése alatt működik, melyek minden egyes 2020 vevő és dekódoló berendezésben megtalálhatók, így az összes 2020 vevő és dekódoló berendezés egy adott 4056 alkalmazás felől nézve azonos, megfelel egymásnak.

A 4008 főalgoritmus futtatja a 4056 alkalmazásokat a 2020 vevő és dekódoló berendezésben, végrehajtja az interaktív 4056 alkalmazásokat és fogadja a 2020 vevő és dekódoló berendezésen kívülről érkező eseményeket, gondoskodik a grafikák és szövegek megjelenítéséről, meghívja a szolgáltatásokhoz tartozó eszközöket, és meghatározott számítások elvégzéséhez használja a 4008 főalgoritmushoz kapcsolt 4006 könyvtár funkcióit.

Maga a 4008 főalgoritmus az egyes 2020 vevő és dekódoló berendezésekben telepített végrehajtható kód, amely alkalmazások interpretálása és futtatása céljából egy program értelmezőt (interpretert) tartalmaz. A 4008 főalgoritmus számos operációs rendszerhez adaptálható, beleértve az egyfeladatos operációs rendszereket, mint amilyen az elterjedt MS-DOS. A 4008 főalgoritmus folyamat sorrendvezérlő eszközökön alapul (amelyek különböző eseményeket, például egy kulcsfolyamatot használnak fel meghatározott hatások elérésére), és egy saját ütemezőt tartalmaz, hogy kezelni tudja a különböző hardver interfészekről származó esemény várakozási sorokat. Ugyancsak ő gondoskodik a grafikák és szövegek megjelenítéséről is. Egy ilyen folyamat sorrendvezérlő egység egy művelet csoport készletet tartalmaz. Minden egyes esemény a művelet sorrendvezérlő egységet arra készíti, hogy aktuális művelet csoportjából egy másik művelet csoportba lépjen át, az esemény jellegétől függően, és hogy az új művelet csoport műveleteit hajtsa végre.

A 4008 főalgoritmus kódbetöltőt tartalmaz ahhoz, hogy a 4056 alkalmazásokat be tudja tölteni és le tudja tölteni a 2020 vevő és dekódoló berendezés 2028 memóriájába. Annak érdekében, hogy optimális felhasználást biztosítsunk, csupán a mindig szükséges kódot töltjük a RAM vagy FLASH memóriába. A letöltött adatot egy ellenőrző mechanizmussal megvizsgáljuk, hogy meggátoljuk a bennlévő 4056 alkalmazás nem kívánt módosítását, vagy akár valamilyen jogosítatlan alkalmazás lefuttatását. A 4008 főalgoritmus ezen kívül egy dekompresszort is tartalmaz, hiszen mivel az alkalmazási kód (egy közbenső kód forma) tárolótér megtakarítása, az MPEG-2 szállító adatáramból történő gyors letöltés vagy egy beépített vételi/dekódoló üzemmód érdekében tömörített, ezért mindenképpen ki kell bontanunk (dekomprimálnunk kell), mielőtt betölthetjük a RAM memóriába. A 4008 főalgoritmus ezért olyan értelmezőt tartalmaz, amely az alkalmazási kódot interpretálva elvég-

zi a különböző változók értékeinek felfrissítését, meghatározza az állapot változásokat, és hibaellenőrzést végez.

Mielőtt bármely 4062 eszköz szolgáltatását igénybe tudnánk venni, egy programot (például egy alkalmazás utasítás szekvenciát) "kliens"-ként kell deklarálnunk, amivel egy logikai hozzáférési utat biztosítunk a 4062 eszközhöz vagy a 4068 eszközkezelőhöz. A 4068 eszközkezelő az újonnan létrehozott kliensnek egy olyan kliens számot ad, amelyet a jövőben a 4062 eszközhöz történő összes hozzáférés során következetesen használunk. Egy 4062 eszköz több különböző klienssel rendelkezhet, és az egyes 4062 eszközökhöz tartozó kliensek számát a 4062 eszközök mindenkori típusa fogja meghatározni. Egy klienst a 4062 eszközben egy "eszköz_nyitva csatorna" elnevezésű eljárással vezethetünk be, amellyel egy kliens számot rendelünk a klienshez, és egy klienst a 4068 eszközkezelő klienslistájáról egy "eszköz_zárva csatorna" elnevezésű eljárással távolíthatunk el.

A 4062 eszközhöz a 4068 eszközkezelő által lehetővé tett hozzáférés lehet szinkron vagy aszinkron hozzáférés. Szinkron hozzáférés esetén "ESZKÖZ: HÍVÁS" eljárást használunk, ami azt jelenti, hogy olyan adatokhoz férhetünk hozzá, amelyek vagy közvetlenül rendelkezésre állnak, vagy olyan funkciókat hívhatunk meg, amelyeknél nincs szükség valamilyen kívánt válasz megvárására. Aszinkron hozzáférés esetén egy "ESZKÖZ: I/O" elnevezésű eljárást használunk, amely azt jelenti, hogy olyan adatokhoz kívánunk hozzáférni, amelyek esetében válaszra kell várnunk, például egy tuner frekvencia sávját letapogatva egy jelet szeretnénk megkeresni, vagy az MPEG adatáramból egy táblázatot szeretnénk megkapni. Ha a kívánt eredmény rendelkezésre áll, a 4008 főalgoritmus várakozó sorába egy esemény íródik be, jelezve ezzel a válasz megérkezését. Egy további eljárás, amelynek a neve "ESZKÖZ: ESEMÉNY", nem várt események lekezelésére biztosít megfelelő megoldást.

Mint azt fent jeleztük, a 4008 főalgoritmus fő hurka nagy számú művelet sorbarendező egységhez csatlakozik, és ha a fő hurok megfelelő eseményt észlel, a vezérlést ideiglenesen a művelet sorbarendező egységek egyikének adja át.

Látható tehát, hogy az ismertetett alapvető 2020 vevő és dekódoló berendezés platformunk igen nagy rugalmasságot visz be abba a módszerbe, ahogy lehetővé tesszük egy alkalmazás számára, hogy különböző eszközökkel kommunikáljon.

A fent bemutatott különböző funkciók megvalósításának pontos részletei, valamint a funkciók hardver és szoftver részek közötti megosztása a konkrét megvalósító mindenkori

döntésétől függ, így itt részletesebben erre nem térünk ki. Megjegyezzük azonban, hogy a szükséges műveletek végrehajtására alkalmas célorientált integrált áramkörök kereskedelmi forgalomban is kaphatók vagy ilyenek könnyen tervezhetők és gyárthatók, és ezeket használhatjuk hardver gyorsító alapjaként vagy még előnyösebben módosíthatjuk, hogy egy célorientált hardver gyorsítót kapjunk, megvalósítsuk a szükséges műveleteket, és ezzel csökkentjük a szoftver futtatásához szükséges feldolgozási erőforrás igényt. Az igényelt műveleteket azonban szoftveresen is megvalósíthatjuk, ha elegendő erőforrás és feldolgozó teljesítmény áll rendelkezésünkre. A 4060 eszközmeghajtót úgy tekinthetjük, mint amely olyan funkcionális egységekből áll össze, amelyekhez egy 4056 alkalmazás (leírásunkban az alkalmazás kifejezést parancsként is használjuk), külön-külön hozzá tud férni. Minden egyes parancs egy 4056 alkalmazással egy 4062 eszközön keresztül kerül kapcsolatba, amely a 4068 eszközközkezelő vezérlete és felügyelete alatt fut, a fent már felsorolt három szabványos eljárás valamelyikével, amelyek más 4062 eszközöknél is használatosak. A 4056 alkalmazás és a 4060 eszközmeghajtó között az információt paraméter táblázatok útján tudjuk továbbítani. A könnyebb érthetőség kedvéért a már jelzett, de csupán felsorolt három alapvető műveletet előbb röviden összefoglaljuk:

1. ESZKÖZ: HÍVÁS. Ezt a parancsot arra használhatjuk egy 4056 alkalmazással, hogy szinkron utasításokat hajtsunk végre, vagy adatátvitelt valósítsunk meg. A 4056 alkalmazás végrehajtása felfüggesztésre kerül mindaddig, amíg a vezérlés visszatér, ha a 4060 eszközmeghajtó tevékenységét befejezte; ez olyan 4056 alkalmazásokat engedélyez, amelyeket igen szorosan meghatározott sorrendben kell végrehajtanunk, hogy megbízhatóan vezérelhessük őket.
2. ESZKÖZ: I/O. Ez a parancs aszinkron műveleteket tesz lehetővé. Ez azt jelenti, hogy egy 4056 alkalmazás kérelmet bocsáthat ki egy adatátvitel vagy a 4060 eszközmeghajtó által végrehajtandó meghatározott funkció érdekében, és a 4056 alkalmazás végrehajtása rögtön folytatódhat, miközben az adatátvitelt vagy a funkciót a 4060 eszközmeghajtó elvégzi.
3. ESZKÖZ: ESEMÉNY. Ez az esemény elfogó funkció lehetővé teszi, hogy a 4060 eszközmeghajtó bizonyos eseményeket jelezzon egy 4056 alkalmazásnak, és különösen, hogy az esemény függvényében a 4056 alkalmazás meghatározott műveletet hajtson végre, függetlenül attól a kódtól, amelyet a 4056 alkalmazás éppen végrehajtott amikor az esemény jelzése bekövetkezett; tulajdonképpen a 4056 alkalmazás ilyen esetben meg-

szakad. A különböző eseményekhez különböző prioritásokat rendelhetünk hozzá, és ilyen eseményeket használhatunk arra, hogy az interfésznél fellépő különböző eseményeket, például egy busz visszaállítást megbízhatóan jelezzünk.

A bemutatott eszközmeghajtót a 3. ábrán olyan 500 blokként jelöltük (szaggatott vonallal), amely 501 logikai fokozatot, 502 vezérlőtárat és 503 puffertárat tartalmaz. Ez a két tár logikailag eltér egymástól, de ugyanannak a fizikai memóriának a részét képezi (pontosabban az 503 puffertár a RAM részét képezi, míg az 502 vezérlőtár vagy a RAM vagy a FLASH memória része lehet). Az eszközmeghajtó a 4068 eszközközkezelőhöz és 550 modem interfészhez kapcsolódik a látható módon. Minden egyes parancs az 501 logikai fokozat részét fogja képezni és akár az 502 vezérlőtár, akár az 503 puffertár részeiben működhet.

Általánosabb megfogalmazással a jelek modulálását és demodulálását jelentő, szűken vett modem funkciókat egy olyan modem egység valósítja meg, amely kialakítható az 550 modem interfészben, vagy ahhoz csatlakoztatható. Az 550 modem interfészen keresztül áramló információ vezérlését és kezelését, a jelzett, szorosan vett modem funkcióktól különállóan, az 500 eszközmeghajtó végzi.

Mint leírtuk, az eszközök illetve eszközmeghajtók, és a 4068 eszközközkezelő közötti kommunikációt három eljárás típussal valósíthatjuk meg: hívásokkal, I/O műveletekkel és eseményekkel. Az 500 eszközmeghajtó által használt eljárásokhoz hívásokat használunk arra, hogy különböző paramétereket küldjünk az alkalmazástól a 4068 eszközközkezelőn át az eszközmeghajtóhoz; I/O műveleteket használunk arra, hogy parancsokat küldjünk a 4068 eszközközkezelőn át az eszközmeghajtóhoz, és eseményeket használunk fel arra, hogy az eszközmeghajtótól a 4068 eszközközkezelőn át üzeneteket juttassunk el egy alkalmazáshoz.

Az alábbiakban a találmány szerinti egy lehetséges eszközben kialakított parancsokat mutatjuk be. Minden egyes parancsot úgy tud egy 4056 alkalmazás elérni, hogy a parancs azonosítóját paraméterként megadja a fenti három szabványos eljárás valamelyikének a segítségével. Az ismertetésre kerülő parancsoknál nincs mindegyikre feltétlenül szükség, és az egyes parancsok funkciói is módosíthatók. Jóllehet az egyes parancsokat egymástól függetlenül hajthatjuk végre vagy módosíthatjuk, de a részletekből kiderül, hogy bizonyos együttes előnyök a bemutatásra kerülő parancsokkal elért kombinált funkcionalitásból fakadnak.

Az egyes parancsokat a parancsok által elért jellemzők és funkciók alapján mutatjuk be, ahogy azok egy 4056 alkalmazás számára láthatóak, egyéb opcionális és előnyös jellemzőkkel együtt. Az adott információval és a kapott specifikációkkal szakember számára a vázolt jellemzők megvalósítására szakmai rutin feladatot jelent, míg a pontos részleteket a mindenkori megvalósítóra célszerű ráhagyni. Például minden egyes parancsot szoftveresen is megvalósíthatunk, előnyösen C programozási nyelven megírva, és ezt kombinálhatjuk, hogy a 4056 alkalmazás futtatásához használt processzoron fusson; természetesen a 4060 eszközmeghajtó egy különálló processzoron is futtatható, és a parancsok némelyike, vagy akár mindegyike erre a célra kifejlesztett hardverrel is megvalósítható. Hívásokat vagy I/O műveleteket használva a 4060 eszközmeghajtó információt jelezhet vagy paramétereket adhat vissza egy 4056 alkalmazásnak, úgy, hogy egy memóriában tárolt paraméter táblázatban értékeket állít be, melyek címe az eszköz számára ismert.

Részletesebben az alkalmazott eljárások vagy parancsok a következők (elaterjedtsége következtében az itt felsorolt paramétereket angol nevükön használjuk, mert ez álláspontunk szerint egyértelműbb, mintha csak magyar fordításban tüntetnénk fel őket):

Eszköz: Hívás

Set (Beállít): az 503 puffertár méretét, az időzítők késleltetési idejét állítja be.

Comm set (kommunikáció beállítása): beállítja az adatméretet (5-8 bit), paritást, az adatátviteli sebességet.

Char set (karakterkészlet beállítás): beállítja a szünet karaktert és a szünet idejét.

Control set (vezérlés beállítása): engedélyezi, illetve letiltja a távoli berendezéshez menő jelet.

Pattern set (minta beállítás): beállítja a keresendő mintákat.

Eszköz: I/O

Parancs: interpretálás nélkül küld parancsot az eszközmeghajtóhoz.

Küldés: adatot küld az eszközmeghajtóhoz.

Akció: egy parancsot interpretálva küld az eszközmeghajtóhoz.

Bontás: leválaszt az eszközmeghajtóról.

Eszköz: Esemény

Vétel: jelzi egy üzenet vételét.

Ki: adatátviteli problémát jelez.

Egy eljárás ezen kívül hibakódot is tartalmaz a visszajelzéshez, és (kivéve az esemény: ki műveletet) egy hívásriportot, amellyel az eszközmeghajtótól vissza tud jelezni a 4068 eszközezőnek. Hasonlóképpen egy hívás vagy I/O eljárás is olyan memória címet tartalmaz, ahol megtalálható az elküldött információ (paraméterek és hasonló).

A legtöbb parancs funkciója széles körben ismert és szokásos. A bemutatott rendszer azonban tartalmaz egy speciális parancsot is, ez pedig a minta beállítás parancs. Ez a parancs egy vagy több olyan mintát definiál, amelyre az 501 logikai fokozat keresést folytat. A parancs egy memória címet tartalmaz, és a memória a jelzett címtől kezdődően egy mintasorozatot tartalmaz. (Pontosabban a memóriában az első hely a minták számát tartalmazza, ezt követik az egyes minták, és minden egyes önálló minta a memóriában tartalmaz egy élőfejet, amely megadja az adott minta hosszát).

Általánosságban három különböző minta típus állítható be, mégpedig események, pozitív nyugtázások és negatív nyugtázások előállításához. Természetesen az is lehetséges lenne, hogy mindhárom minta típust ugyanabban a parancsban foglaljunk, ahol az egyes minták élőfejei azonosítanak a minták típusát. Célszerűbbnek tűnik azonban három külön alparancs forma használata a három mintatípushoz. Így a pozitív és a negatív nyugtázás mintákat, amelyeket egy parancsok a modem általi megbízható vételének a megállapításához alkalmazunk, célszerűen dedikált parancsok felhasználásával állítjuk be, az összes többi mintát arra használjuk, hogy egy eseményt indítsunk, ahol minden egyes mintához egy-egy számot rendelünk hozzá, hogy a különböző minták között megkülönböztetést tehesünk. Ezek a parancsok a 4. ábrán látható módon az 502 vezérlőtárban 510 memóriazónát hoznak létre, ennek három szegmense van, melyek az esemény, a pozitív nyugtázás és a negatív nyugtázás mintákat tartalmazzák. Az 501 logikai fokozat 511 összehasonlító fokozatot tartalmaz, amely az 510 memóriazónához kapcsolódik.

Az 510 memóriazóna a három minta beállítás parancsokkal specifikált memória területekből tevődhet össze. Ebben az esetben az 510 memóriazóna három szegmense fizikailag a memória különböző helyein található. Az 511 összehasonlító fokozat fogja eltárolni a három szegmens kezdőcímét, és a különböző mintákhoz hozzáférés biztosítására használja a minták számát (ahogy az az egyes szegmensek elején eltárolásra került), valamint az egyes minták hosszát, hogy meghatározza az egyes minták pontos elhelyezkedését, és szükség esetén ezekhez hozzá tudjon férni. Alternatív esetben az 511 összehasonlító fokozat inicia-

lizálhatja a memóriát úgy, hogy az 510 memóriazónát egy előre meghatározott helyen mintákat tartalmazó táblázatként állítja fel.

Ha üzenet érkezik, az az 503 puffertár pufférébe kerül. Az 503 puffertár 512 pufferekre van felosztva úgy, hogy szükség esetén az üzenet egynél több puffert is elfoglalhat; az üzenetek számára rendelkezésre álló hely logikailag minden egyes üzenet részére logikailag folyamatos. Ha mintaegyezést kell megállapítani, akkor az 511 összehasonlító fokozat az üzenet utolsó 32 karakterét végignézi, hogy talál-e egyezést az 510 memóriazónában tárolt valamely mintával.

Ha egyezést talál, akkor az egyezés típusát (vagyis azt, hogy esemény, pozitív nyugtázás, vagy negatív nyugtázás) az üzenet élőfejébe írja annak a puffertben történő eltárolásakor, és/vagy bizonyos megfelelő akciót hajt végre az eszközmeghajtó révén.

Megjegyezzük, hogy a pozitív és a negatív nyugtázás minta detektálást felhasználhatjuk arra, hogy detektáljuk a modemtől érkező várt válaszokat, például, ha parancsokat küldtünk, miközben a modem off-line állapotban volt. A modemtől érkező egyéb jelek egy külön minta listában lévő mintákkal egyezhetnek, és ezeket arra használjuk, hogy eseményeket indítsunk vagy kezdeményezzük; ennek érdekében a pozitív nyugtázás és a negatív nyugtázás minták beállítását előnyösen függetlenül hajtjuk végre a nem várt eseményeknek vagy a modemtől érkező egyéb üzeneteknek megfelelő minták listájának a beállításaitól.

Ezen a módon lehetőségünk nyílik arra, hogy egy alkalmazás különböző modemek bármelyikével kommunikációt folytathasson. Például számos modem általánosan valamilyen szabványnak megfelelő, például a Hayes szabványnak megfelelő kommunikációt folytat és protokollt használ az utasítás szekvenciákhoz és a válaszokhoz, de járulékos tulajdonságokkal is rendelkezhet, melyekhez külön sajátos válasz szekvenciát állít elő. Ilyen esetben a pozitív nyugtázás és a negatív nyugtázás mintákat kell szabványos mintaként beállítani, és a járulékos jellemzőknek megfelelő válasz szekvenciákat esemény jelekhez rendelhetjük hozzá.

Az 5. ábra az 503 puffertárban rezidens 520 üzenetet vázol. Ez az üzenet 521 élőfejből és 522 üzenettestből áll. Az üzenet ebben az esetben fizikailag két 512, 512' puffer között van megosztva. Az 520 összehasonlító fokozat minta egyezést keres (vagyis azt nézi, hogy talál-e egyezést az 510 memóriazónában tárolt mintákkal) az üzenet 522 üzenettestének utol-

só 32 karakterét tartalmazó 523 régióban. Az esetlegesen talált bármely egyezés természetét az 503 puffertárban beilleszti az üzenet 521 élőfejébe.

Az is lehetséges, hogy egynél több egyezés lép fel, ami két esetben lehetséges. Először, két ugyanolyan típusú minta léphet fel (az üzenet utolsó 32 karakterének különböző helyein). Ez a kétszeres egyezést figyelmen kívül hagyhatjuk, vagyis mintegy sima egyszeri egyezést vesszük. Másodsor, két különböző típusú egyezés léphet fel (szintén az üzenet utolsó 32 karakterének különböző helyein). A mindenkori körülményektől függően az 511 összehasonlító fokozatot beállíthatjuk úgy, hogy jelezze az egyezés típusát, vagy a különböző egyezés típusok közötti előre meghatározott prioritás sorrendnek megfelelően, vagy aszerint, hogy melyik minta helyezkedik el az üzenet végéhez közelebb.

Parancs üzemmódban, vagyis, amikor a modem paramétereit állítjuk be, ezt a technikát használhatjuk fel arra, hogy egy parancs üzenetet küldjünk az eszközmeghajtónak. Az üzenetet az eszközmeghajtó fogja értelmezni, és nem vezeti tovább interfészként mondjuk egy adatszerverhez. Minden egyes elküldött parancs esetében az eszközmeghajtó mintakeresést hajt végre a válaszul kapott üzenetben, és az eszközmeghajtónak ennek megfelelően fog választ küldeni.

Ebben az üzemmódban arra is lehetőség nyílik, hogy egy üzenetet kapjunk az interfészen keresztül anélkül, hogy előtte parancsot küldtünk volna. Ez például abban az esetben történhet, ha csengetés feszültséget detektálunk. Az eszközmeghajtónak ebben az esetben szintén el kell végeznie egy mintakeresést az esemény minta listában.

On-line üzemmódban a modem egy adatszerverhez csatlakozik, és ilyenkor üzeneteket fogadhatunk az adatszervertől. Az eszközmeghajtó ilyen esetben az adatszervertől kapott üzenet utolsó 32 karakterében hajt végre mintakeresést.

A bemutatott parancsok és események kizárólag a találmány lényegének a bemutatására szolgáltak, és a találmányt különböző módokon is megvalósíthatjuk, és különösen néhány összetevőt kombinálhatunk más, hasonló funkciót végrehajtó összetevővel, vagy a megvalósítás egyszerűsítése érdekében egyes összetevőket el is hagyhatunk. Az egyes funkciók hardveres vagy szoftveres megvalósítását is szabadon keverhetjük és kombinálhatjuk, akár összetevők között, akár egy összetevőn belül.

Nyilvánvaló, hogy a hardver, a számítógép szoftver valamint hasonló módon megvalósított funkciók villamos és hasonló jeleken alapulva zajlanak le. A szoftver megvalósításokat

ROM-ban is eltárolhatjuk, vagy FLASH memóriában, mely utóbbi esetben a programok módosíthatók, frissíthetők.

A találmány szerinti eljárást leírásunkban valamint a rajzon kizárólag a találmányi gondolat megértetése céljából és az ahhoz szükséges mértékben ismertettük, így attól a találmány igényelt oltalmi körén belül nyilvánvalóan széles körben el lehet térni. A leírásban, az igénypontokban és a rajzokon látható jellemzők mind külön-külön, mind megfelelő kombinációban megvalósíthatók.

Szabadalmi igénypontok

1. Eszközmeghajtó modemmel folytatott kommunikációra és modem vezérlésére, amely a modemtől üzeneteket vevő és tároló puffertárat, vezérlő paramétereket tároló vezérlő tárat, valamint az eszközmeghajtó vezérlését valamint az üzenetforgalom vezérlését végző logikai fokozatot tartalmaz, **azzal jellemezve**, hogy a logikai fokozat (501) a vezérlő memóriában (502) tárolt minták és a puffertárban (503) tárolt üzenetek közötti egyezést kereső, és az eszközmeghajtóhoz csatlakoztatott eszközközkezelőhöz elküldendő jelet előállító összehasonlító fokozatot (520) tartalmaz.
2. Az 1. igénypont szerinti eszközmeghajtó, **azzal jellemezve**, hogy az összehasonlító fokozat (520) az eltárolt mintákat az üzenet előre meghatározott végleges hosszával összehasonlító fokozat.
3. Az 1. vagy 2. igénypont szerinti eszközmeghajtó, **azzal jellemezve**, hogy a minták különböző típusú minták, valamint azonos típusú több mintából állhatnak.
4. A 3. igénypont szerinti eszközmeghajtó, **azzal jellemezve**, hogy a minta típusok esemény, pozitív nyugtázás, és negatív nyugtázás típusok, és az előállított jel az egyező minta típusát jelző jel.
5. A 4. igénypont szerinti eszközmeghajtó, **azzal jellemezve**, hogy a jelet a modemen keresztül kapott üzenethez hozzáadva az eszközközkezelőnek elküldő eszközmeghajtó.
6. A 4. vagy 5. igénypont szerinti eszközmeghajtó, **azzal jellemezve**, hogy a parancsok az eszközmeghajtó paramétereit beállító hívások formájában, továbbá az eszközmeghajtóhoz vezérlőjeleket és adatokat küldő I/O műveletek formájában, valamint minták észlelését,

vagy üzenet vételét, vagy adatátviteli problémákat az eszközközkezelőnek jelző események formájában kerülnek továbbításra az eszközmeghajtó és az eszközközkezelő között.

7. A 6. igénypont szerinti eszközmeghajtó, *azzal jellemezve*, hogy a hívás parancs egyik formája az összehasonlító fokozat (520) által keresendő egy vagy több minta meghatározásához legalább egy minta beállítás parancsot tartalmaz.

8. A 7. igénypont szerinti eszközmeghajtó, *azzal jellemezve*, hogy a keresendő mintákat meghatározó három minta beállítási parancs altípust tartalmaz, egy minta beállítás parancsot minden egyes minta típushoz.

9. A 7. vagy 8. igénypont szerinti eszközmeghajtó, *azzal jellemezve*, hogy a legalább egy mintabeállítás parancs több beazonosítandó mintát meghatározó parancs, ahol minden egyes minta a megfelelő minta detektálása esetén előállítandó hozzárendelt esemény jelzést tartalmaz.

10. A 4-9. igénypontok bármelyike szerinti eszközmeghajtó, *azzal jellemezve*, hogy a keresendő minták a vezérlőtár (502) alterületein vannak eltárolva.

11. A 4-9. igénypontok bármelyike szerinti eszközmeghajtó, *azzal jellemezve*, hogy a keresendő minták a vezérlőtár (502) egyetlen folyamatos területén vannak eltárolva.

12. Az 1-11. igénypontok bármelyike szerinti eszközmeghajtó, *azzal jellemezve*, hogy többszörös egyezés esetén csupán az utolsó egyezést veszi figyelembe.

13. Az 1-11. igénypontok bármelyike szerinti eszközmeghajtó, *azzal jellemezve*, hogy többszörös egyezés esetén a különböző egyezés típusok között előre meghatározott elsőbbségi sorrendnek megfelelően jelzi az egyezés típusát.

14. Vevő és dekódoló berendezés, *azzal jellemezve*, hogy az 1-13. igénypontok bármelyike szerinti eszközmeghajtót tartalmaz.

15. Eljárás modemmel folytatott kommunikációra és modem vezérlésére, amelynek során puffertárat és vezérlőtárat tartalmazó eszközmeghajtót használunk, *azzal jellemezve*, hogy a modemtől üzeneteket veszünk és azokat eltároljuk a puffertárban (503), a vezérlőtárban (502) vezérlő paramétereket tárolunk el, és a vezérlő memóriában (502) eltárolt mintákat összehasonlítjuk a puffertárban (503) eltárolt üzenetekkel, és mintaaazonosságot keresünk, hogy az eszközmeghajtóhoz csatlakoztatott eszközközkezelőnek elküldendő jelet állítsunk elő.

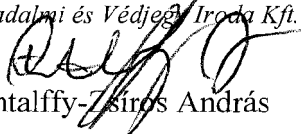
16. A 15. igénypont szerinti eljárás, **azzal jellemezve**, hogy az eltárolt mintákat az üzenet előre meghatározott végleges hosszával hasonlítjuk össze.
17. A 15. vagy 16. igénypont szerinti eljárás, **azzal jellemezve**, hogy a minták különböző típusúak, és lehetnek azonos típusú minták is.
18. A 17. igénypont szerinti eljárás, **azzal jellemezve**, hogy a minta típusok esemény, pozitív nyugtázás, és negatív nyugtázás típust tartalmaznak, és az előállított jellel az egyező minták típusát jelezzük.
19. A 18. igénypont szerinti eljárás, **azzal jellemezve**, hogy az eszközmeghajtóval a jelet úgy küldjük el az eszközközkezelőnek, hogy azt a modemén keresztül kapott üzenethez hozzáadjuk.
20. A 18. vagy 19. igénypont szerinti eljárás, **azzal jellemezve**, hogy a parancsokat az eszközmeghajtó és az eszközközkezelő (4068) között hívások formájában továbbítjuk, amelyekkel paramétereket állítunk be az eszközmeghajtóba, továbbá I/O műveletek formájában továbbítjuk, amelyekkel vezérlőjeleket és adatokat küldünk az eszközmeghajtókhoz, valamint események formájában továbbítjuk, amelyekkel mintadetektálást, vagy üzenetvételt, vagy adatátviteli problémákat jelzünk az eszközközkezelőnek.
21. A 20. igénypont szerinti eljárás, **azzal jellemezve**, hogy a hívás parancs egyik formája legalább egy mintabeállítás parancsot tartalmaz, amellyel a vezérlőtárban (502) keresendő egy vagy több mintát határozzunk meg.
22. A 21. igénypont szerinti eljárás, **azzal jellemezve**, hogy keresendő mintákat meghatározó három minta beállítási parancs altípust tartalmaz, egy minta beállítás parancsot minden egyes minta típushoz.
23. A 21. vagy 22. igénypont szerinti eljárás, **azzal jellemezve**, hogy a legalább egy mintabeállítás parancs több beazonosítandó mintát meghatározó parancs, ahol minden egyes minta a megfelelő minta detektálása esetén előállítandó hozzárendelt esemény jelzést tartalmaz.
24. A 21-23. igénypontok bármelyike szerinti eljárás, **azzal jellemezve**, hogy a keresendő mintákat a vezérlőtár (502) alterületein tároljuk el.
25. A 21-23. igénypontok bármelyike szerinti eljárás, **azzal jellemezve**, hogy a keresendő mintákat a vezérlőtár (502) egyetlen folyamatos területén tároljuk el.

26. A 15-25. igénypontok bármelyike szerinti eljárás, *azzal jellemezve*, hogy többszörös egyezés esetén csupán az utolsó egyezést vesszük figyelembe.

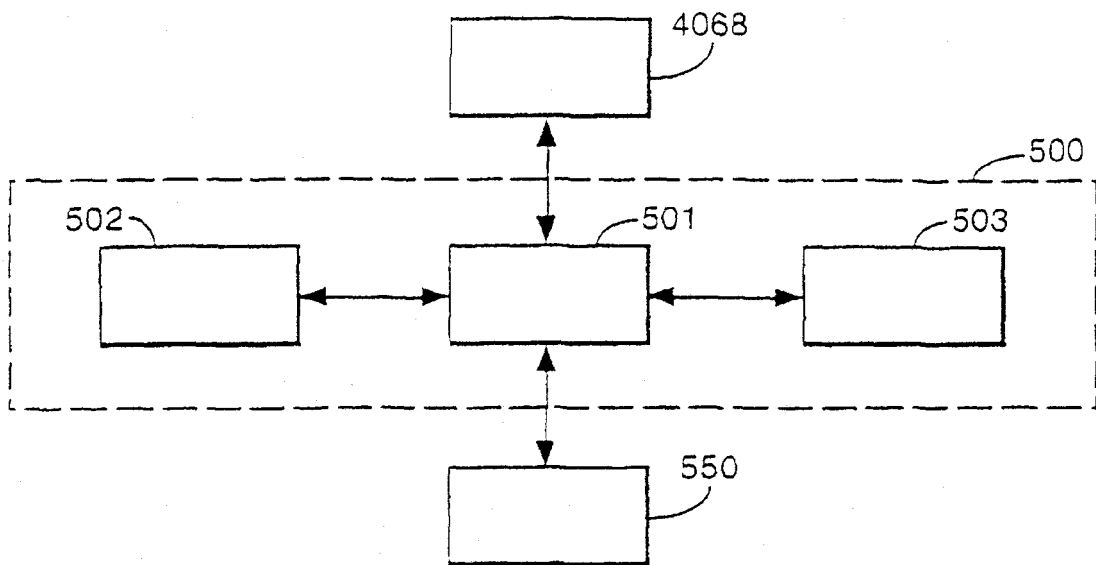
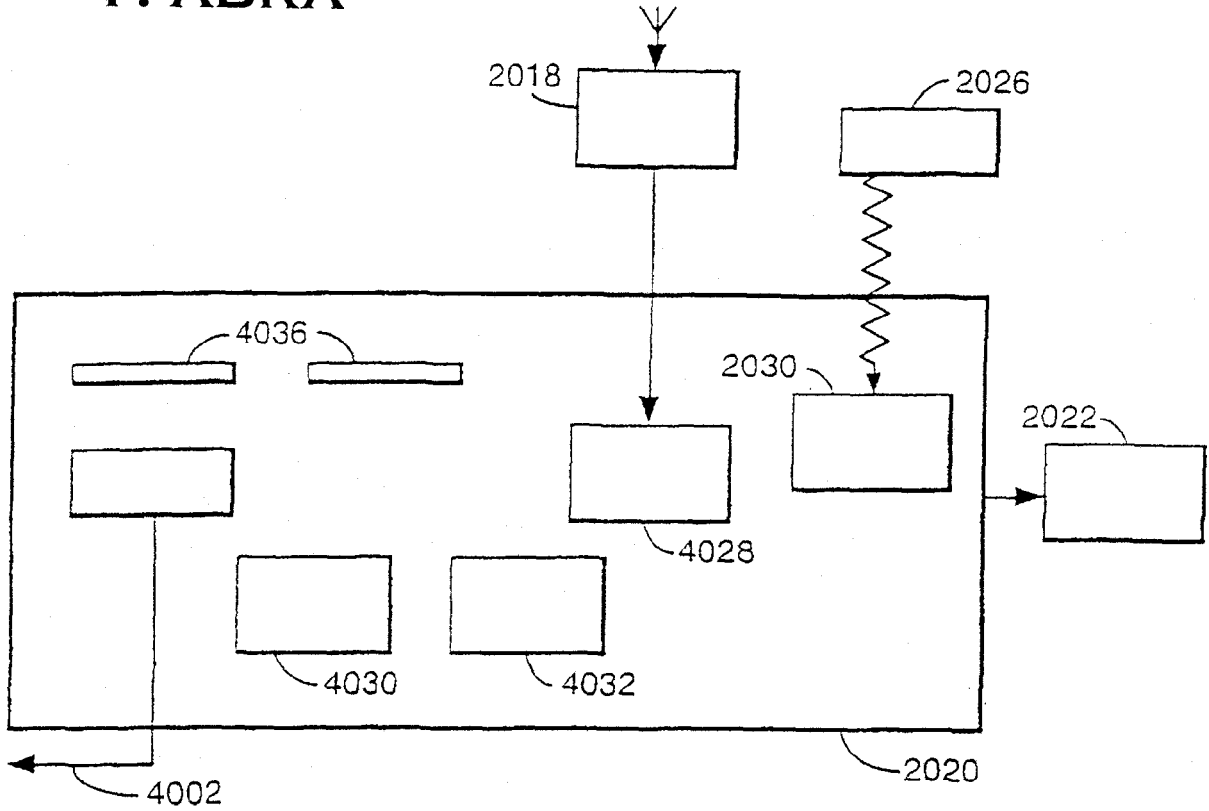
27. A 15-25. igénypontok bármelyike szerinti eljárás, *azzal jellemezve*, hogy többszörös egyezés esetén a különböző egyezés típusok között előre meghatározott elsőbbségi sorrendnek megfelelően jelezzük az egyezés típusát.

A meghatalmazott:

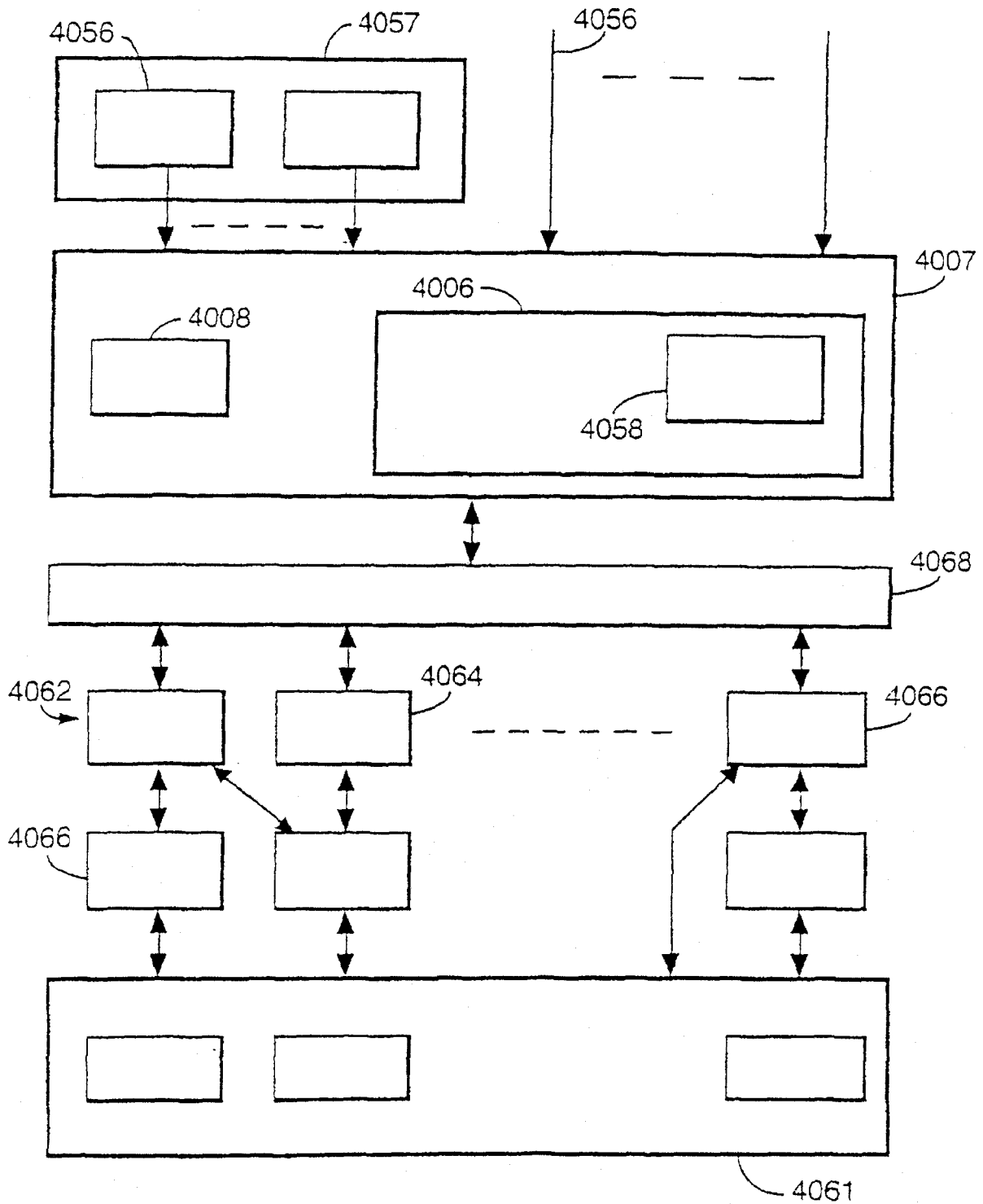
VÁRHEGYI C.

DANUBIA
Szabadalmi és Védjegyi Iroda Kft.

Antalffy-Zsírós András

1. ÁBRA

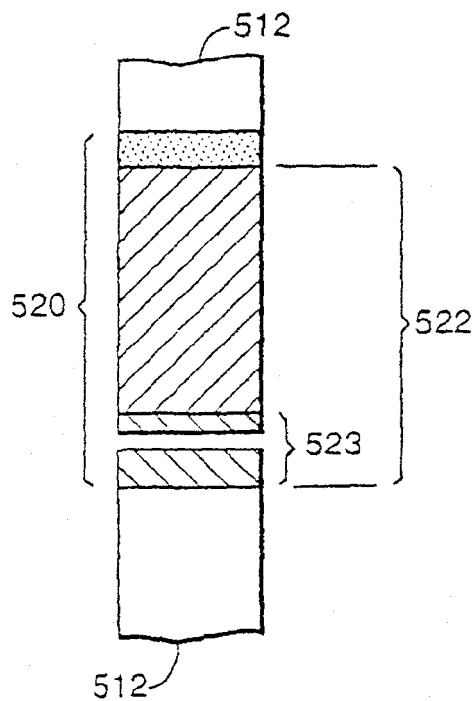
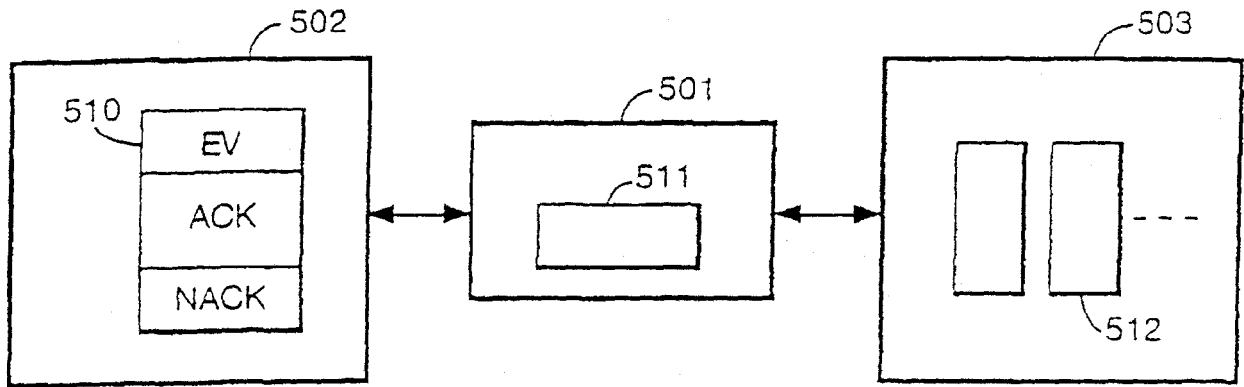


3. ÁBRA



2. ÁBRA

4. ÁBRA



5. ÁBRA