



Eljárás és berendezés kommunikációs csatorna létesítésére egyrészt egy program és egy számítógép rendszer egy összetevője, másrészt egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, továbbá eljárás adatkommunikációra egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, valamint vevő és dekódoló berendezés

Kivonat

A találmány tárgya egy eljárás valamint azt megvalósító berendezés kommunikációs csatorna létesítésére egyrészt egy program és egy számítógép rendszer egy összetevője között, amelynek során illetve amellyel a számítógép rendszerben lévő összetevőkkel társított logikai eszközöket tárolunk el, ahol minden egyes logikai eszköznek egy eszközazonosítót adunk; a programhoz egy program azonosítót társítunk; és a programból egy olyan jelet bocsátunk ki, amely tartalmazza a program azonosítót és egy logikai eszköz eszközazonosítóját, a program és az egyes logikai eszközök között létrehozott eszközkezelőnek, mely utóbbival egy kommunikációs csatornát létesítünk a program és a logikai eszköz között, a kapott program azonosítót és eszközazonosítót felhasználva, lehetővé téve ezzel a program és a logikai eszköz közötti ezt követő kommunikációt.

A találmány tárgya másrészt egy eljárás valamint azt megvalósító berendezés kommunikációs csatorna létesítésére egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, amelynek során illetve amellyel a vevő és dekódoló berendezés (2020) egyes összetevőivel társított logikai eszközöket tárolunk el, ahol mindegyik logikai eszközhöz egy-egy eszközazonosítót rendelünk hozzá; az alkalmazáshoz egy alkalmazás azonosítót rendelünk hozzá; és az alkalmazástól az alkalmazás azonosítót valamint egy logikai eszköz eszközazonosítóját tartalmazó jelet bocsátunk ki egy, az alkalmazás valamint az egyes logikai eszközök között létrehozott közös interfészhez, amellyel kommunikációs csatornát létesítünk az alkalmazás és a logikai eszköz között a közös interfészen keresztül.

A találmány tárgya harmadrészt egy eljárás adatkommunikációra egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, amelynek során az alkalmazás és az összetevő között a valamelyik fenti eljárással egy kommunikációs csatornát létesítünk; és ezt követően az alkalmazástól az alkalmazás azonosítót és a logikai eszköz eszközazonosítóját, valamint az összetevőnek a vele társított logikai eszköz révén végrehajtandó működésére vonatkozó parancsot tartalmazó jelet bocsátunk ki az eszközkezelőhöz.

A találmány tárgyát képezi továbbá egy vevő és dekódoló berendezés, amely a javasolt berendezések valamelyikét tartalmazza.

(4. ábra)

KÖZZÉTÉTELI PÉLDÁNY.



Eljárás és berendezés kommunikációs csatorna létesítésére ~~(egyrészt egy program és egy számítógép rendszer egy összetevője, másrészt egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között), továbbá eljárás adatkommunikációra (egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között),~~ valamint vevő és dekódoló berendezés

A találmány tárgya egy eljárás valamint egy berendezés kommunikációs csatorna létesítésére egyrészt egy program és egy számítógép rendszer egy összetevője között, másrészt egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, harmadrészt egy eljárás adatkommunikációra, egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, negyedrészről egy vevő és dekódoló berendezés.

A digitális jelátviteli rendszereket elsősorban televíziójelek sugárzása céljából, különösen, bár nem kizárólagosan a műholdas televíziós rendszerekben történő használatra fejlesztették ki, ám ez megnyitotta a lehetőséget a megoldás más célokra történő felhasználására is. Ezen lehetőségek egyike, hogy interaktív kapcsolatot tudunk létrehozni a végfelhasználókkal.

Az interaktivitás megvalósításának egyik lehetséges módja, hogy egy alkalmazást futtatunk abban a vevő és dekódoló berendezésben, amelyen keresztül a televíziójelet vesszük. Az alkalmazás programkódja, röviden kódja permanensen tárolható a vevő és dekódoló berendezésben, de ennek lehetősége eléggé korlátozott. Előnyösebb, ha a vevő és dekódoló berendezés arra alkalmasan van kiképezve, hogy egy szükséges vagy kívánt alkalmazáshoz a kódot saját maga le tudja tölteni. Ezen az úton nagyobb variációt biztosíthatunk, és az alkalmazásokat is akkor frissíthetjük fel, amikor arra szükség van anélkül, hogy ehhez a végfelhasználótól valamilyen tevékenységre, közreműködésre lenne szükség.

Az egy vevő és dekódoló berendezésben történő végrehajtásra tervezett alkalmazásokat több, különböző műsorsugárzó szolgáltató cég készíti. Egy vevő és dekódoló berendezés rendszerint több különböző interfésszel rendelkezik, mint amilyen a soros interfész és a párhuzamos interfész, amelyeken keresztül a vevő és dekódoló berendezést külső egységekkel köthetjük össze. Az ezekhez az interfészekhez való eszközmeghajtókat a vevő és dekódoló berendezések gyártói biztosítják. A számos alkalmazás forrás és a számos vevő és dekódoló berendezés gyártó és gyártmány következtében lényeges, hogy egy alkalmazás ugyanúgy viselkedjen bármely vevő és dekódoló berendezésben, valamint, hogy minden



egy vevő és dekódoló berendezés ugyanazt az alkalmazást ugyanolyan, helyes és eredményes módon hajtja végre.

A kitűzött feladat megoldása során olyan eljárást vettünk alapul kommunikációs csatorna létesítésére egy program és egy számítógép rendszer egy összetevője között, amelynek során a találmány szerint a számítógép rendszerben lévő összetevőkkel társított logikai eszközöket tárolunk el, ahol minden egyes logikai eszköznek egy eszközazonosítót adunk; a programhoz egy program azonosítót társítunk; és a programból egy olyan jelet bocsátunk ki, amely tartalmazza a program azonosítót és egy logikai eszköz eszközazonosítóját, a program és az egyes logikai eszközök között létrehozott eszközközvetítőnek, mely utóbbival egy kommunikációs csatornát létesítünk a program és a logikai eszköz között, a kapott program azonosítót és eszközazonosítót felhasználva, lehetővé téve ezzel a program és a logikai eszköz közötti ezt követő kommunikációt.

A találmány szerinti eljárás egy előnyös fogantatási módja értelmében a program egy alkalmazásnak legalább egy részét alkotja, és a számítógép rendszer egy vevő és dekódoló berendezésnek legalább részét képezi.

A kitűzött feladat megoldása során olyan továbbá olyan eljárást vettünk alapul egy kommunikációs csatorna létesítésére, egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, amelynek során a találmány értelmében a vevő és dekódoló berendezés egyes összetevőivel társított logikai eszközöket tárolunk el, ahol mindegyik logikai eszközhöz egy-egy eszközazonosítót rendelünk hozzá; az alkalmazáshoz egy alkalmazás azonosítót rendelünk hozzá; és az alkalmazástól az alkalmazás azonosítót valamint egy logikai eszköz eszközazonosítóját tartalmazó jelet bocsátunk ki egy, az alkalmazás valamint az egyes logikai eszközök között létrehozott közös interfészhez, amellyel kommunikációs csatornát létesítünk az alkalmazás és a logikai eszköz között a közös interfészen keresztül.

A találmány szerinti eljárás egy előnyös fogantatási módja értelmében a közös interfész a kapott alkalmazás azonosító és eszközazonosító felhasználásával az alkalmazás és a logikai eszköz között kommunikációs csatornát létesítő eszközközvetítőt tartalmaz.

A találmány szerinti eljárás egy további előnyös fogantatási módja értelmében az alkalmazás azonosítót az eszközközvetítővel rendeljük az alkalmazáshoz.



Ugyancsak előnyös a találmány értelmében, ha az alkalmazás által egy, a logikai eszközzel társított összetevő állapotváltozását jelző, a logikai eszköz által kibocsátott üzenet vételét engedélyező parancsot tartalmaz.

Előnyös továbbá, ha az üzenetet átmenetileg az alkalmazáshoz történő rákövetkező továbbítás céljára egy várakozási sorban tároljuk.

Fentiekén túlmenően előnyös, ha a vevő és dekódoló berendezés több várakozási sort tartalmaz, ahol minden egyes várakozási sorhoz az egyes üzeneteknek a várakozási sorokból az alkalmazáshoz történő továbbítási sorrendjét jelző egy-egy prioritási szintet rendelünk.

Ugyancsak előnyös a találmány értelmében, ha a jel az üzenetet átmenetileg tároló várakozási sor prioritási szintjét is tartalmazza.

A kitűzött feladat megoldása során olyan továbbá olyan eljárást vettünk alapul adatkommunikációra, egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, amelynek során az alkalmazás és az összetevő között az előzőekben említett találmány szerinti eljárással egy kommunikációs csatornát létesítünk; és ezt követően az alkalmazástól az alkalmazás azonosítót és a logikai eszköz eszközazonosítóját, valamint az összetevőnek a vele társított logikai eszköz révén végrehajtandó működésére vonatkozó parancsot tartalmazó jelet bocsátunk ki az eszközközlelőhöz.

A találmány szerinti eljárás egy előnyös fogatosítási módja értelmében a jel a logikai eszköz által az összetevőnek adandó adat címét, valamint az összetevőtől a logikai eszközhöz kibocsátandó adat címét tartalmazza.

A találmány szerinti eljárás egy további előnyös fogatosítási módja értelmében a jel annak a várakozási sornak a prioritási szintjét is tartalmazza, amelyben a logikai eszköz által kibocsátott üzenetet tárolunk el átmenetileg az alkalmazáshoz történő továbbítását megelőzően.

Ugyancsak előnyös a találmány értelmében, ha az alkalmazás és a logikai eszköz legalább egyikét a vevő és dekódoló berendezésnek a vevő és dekódoló berendezés egy összetevőjén keresztül adjuk be.

Előnyös továbbá, ha az összetevőként egy MPEG folyam tuner, egy soros interfész, egy párhuzamos interfész, egy modem vagy egy kártyaolvasó valamelyikét tartalmazza.



A kitűzött feladat megoldása során olyan berendezést vettünk alapul kommunikációs csatorna létesítésére egy program és egy számítógép rendszer egy összetevője között, amelynek a találmány értelmében a számítógép rendszerben lévő összetevőkkel társított logikai eszközöket eltároló és minden egyes logikai eszköznek egy eszközazonosítót adó eszköze; a programhoz egy program azonosítót társító eszköze; és a programból egy, a program azonosítót és egy logikai eszköz eszközazonosítóját tartalmazó jelet a program és az egyes logikai eszközök között létrehozott eszközkezelőnek kibocsátó eszköze van, mely utóbbival a kapott programazonosítót és eszközazonosítót felhasználva és ezzel a program és a logikai eszköz közötti ezt követő kommunikációt lehetővé téve egy kommunikációs csatornát a program és a logikai eszköz között létesítő eszköze van.

A találmány szerinti berendezés egy előnyös kiviteli alakja értelmében a program egy alkalmazásnak legalább egy részét alkotja, és a számítógép rendszer egy vevő és dekódoló berendezésnek legalább részét képezi.

A kitűzött feladat megoldása során továbbá olyan berendezést vettünk alapul egy kommunikációs csatorna létesítésére, egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, amelynek a találmány szerint a vevő és dekódoló berendezés egyes összetevőivel társított logikai eszközöket eltároló és mindegyik logikai eszközhöz egy-egy eszközazonosítót hozzárendelő eszköze; az alkalmazáshoz egy alkalmazás azonosítót hozzárendelő eszköze; és az alkalmazástól az alkalmazás azonosítót valamint egy logikai eszköz eszközazonosítóját tartalmazó jelet egy, az alkalmazás valamint az egyes logikai eszközök között létrehozott közös interfészhez kibocsátó és azzal az alkalmazás és a logikai eszköz között a közös interfészen keresztül kommunikációs csatornát létesítő eszköze van.

A találmány szerinti berendezés egy előnyös kiviteli alakja értelmében a közös interfész a kapott alkalmazás azonosító és eszközazonosító felhasználásával az alkalmazás és a logikai eszköz között kommunikációs csatornát létesítő eszközkezelőt tartalmaz.

A találmány szerinti berendezés egy további előnyös kiviteli alakja értelmében az eszközkezelő az alkalmazás azonosítót az alkalmazáshoz társító eszközkezelőként van kiképezve.

Ugyancsak előnyös a találmány értelmében, ha a logikai eszköz által az alkalmazáshoz történő továbbításra kibocsátott üzenetet átmenetileg tároló várakozási sort tartalmaz.



Előnyös továbbá, ha a vevő és dekódoló berendezés több várakozási sort tartalmaz, ahol minden egyes várakozási sorhoz az egyes üzeneteknek a várakozási sorokból az alkalmazáshoz történő továbbítási sorrendjét jelző egy-egy prioritási szint van hozzárendelve.

Fentiekén túlmenően előnyös, ha az összetevőhöz a logikai eszköz által beadandó adatokat tároló eszköze, valamint a komponens által a logikai eszközhöz kibocsátott adatokat tároló eszköze van.

A kitűzött feladat megoldása során olyan fentiekén túlmenően olyan vevő és dekódoló berendezést vettünk alapul, amely a találmány szerinti berendezést tartalmaz.

Előnyös végül a találmány szerinti vevő és dekódoló berendezés olyan kiviteli alakja, amelynek a találmány értelmében egy tömörített MPEG jelet fogadó eszköze, a vett jelet dekódoló és ezzel egy televízió jelet előállító eszköze valamint a televízió jelet egy televízió készülékhez továbbító eszköze van.

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

1. ábra a találmány szerinti digitális televízió rendszer egy lehetséges kiviteli alakjának tömbvázlatát mutatja, a
2. ábrán a digitális televízió rendszer interaktív rendszerének a vázlatos felépítése látható, a
3. ábra egy interaktív vevő és dekódoló berendezés memóriájába letöltött modulon belül a fájlok elrendezését mutatja, a
4. ábrán az interaktív vevő és dekódoló berendezés memóriájának memóriakötet elrendezése látható, a
5. ábra a vevő és dekódoló berendezés interfészeit mutatja vázlatosan, a
6. ábrán a vevő és dekódoló berendezésben működő szoftver fontosabb blokkjait tüntettük fel tömbvázlat szinten, és a
7. ábra egy eszközközvetítő, több kliens és több eszköz közötti kapcsolatra mutat példát.

Az 1. ábrán a találmány szerinti 1000 digitális televízió műsorszóró és vételi rendszer vázlatát tüntettük fel az áttekintés elősegítésére. Az 1000 digitális televízió műsorszóró hagyományosan felépített 2000 digitális televízió rendszert foglal magában, amely az ismert MPEG-2 tömörítő rendszert használja tömörített digitális jelek kibocsátásához. Egy kicsit

részletesebben, a 2002 MPEG-2 tömörítő, amely egy műsorszóró központban helyezkedik el, digitális adatáramot kap (amely rendszerint video jel adatáram). A 2002 MPEG-2 tömörítő 2006 vonalon keresztül 2004 multiplexer és bitsorkódolóhoz van csatlakoztatva. A 2004 multiplexer és bitsorkódoló számos további bemenőjelet kap, ezekből egy vagy több szállítási adatáramot állít össze, és a tömörített digitális jelet a műsorszóró központban lévő 2008 adóberendezéshez továbbítja egy további 2010 vonalon keresztül, amely természetesen számos módon megvalósítható, beleértve a szokásos távközlési vezetékcsatlakozásokat is. A 2008 adóberendezés egy 2012 földi állomástól műholdra irányuló adatátviteli kapcsolat segítségével elektromágneses jelet továbbít 2014 transzponderhez, amely azt elektronikusan feldolgozza, és 2016 műholdról földi állomás felé irányuló adatátviteli kapcsolaton át földi 2018 vevőhöz továbbítja, mely 2018 vevő általában egy végfelhasználó tulajdonában álló, vagy általa bérelt parabola antenna. A 2018 vevővel fogadott jelek szintén a végfelhasználó tulajdonában álló vagy általa bérelt integrált 2020 vevő és dekódoló berendezésbe kerülnek, amely a tömörített MPEG-2 jelet olyan televízió jellé alakítja vissza, amely minden további nélkül használható 2022 televízió készülékben.

A 2004 multiplexer és bitsorkódolóhoz és a 2020 vevő és dekódoló berendezéshez egy 3000 feltételes hozzáférési rendszer kapcsolódik, amely részben a műsorszóró központban, részben a 2020 vevő és dekódoló berendezésben van kialakítva. Ez a 3000 feltételes hozzáférési rendszer lehetővé teszi a felhasználó számára, hogy egy vagy több műsorszóró szolgáltatótól digitális televízió műsorokat fogadjon. A kereskedelmi ajánlatokra vonatkozó üzenetek dekódolására képes programozható csipkártya (ezek az üzenetek tulajdonképpen a műsorszóró szolgáltató által felkínált egy vagy több televízió programot jelentenek) helyezhető a 2020 vevő és dekódoló berendezésbe, és a kettő együttes használatával a végfelhasználó vagy előfizetési módon, vagy fizetős módon (pontosabban "fizess-és-nézd", pay per view módon) műsorok által megtestesülő eseményeket vásárolhat.

Az ugyancsak a 2004 multiplexer és bitsorkódolóhoz és a 2020 vevő és dekódoló berendezéshez kapcsolódó 4000 interaktív rendszer szintén egyrészt a műsorszóró központban, másrészt a 2020 vevő és dekódoló berendezésben van kialakítva, és lehetővé teszi a végfelhasználó számára, hogy számos alkalmazással egy modemes 4002 válaszcsatormán keresztül interaktív módon álljon kapcsolatban.

A 2. ábrán a találmány szerinti 1000 digitális televízió műsorszóró 4000 interaktív rendszerének általános felépítését vázoltuk.



A 4000 interaktív rendszer például lehetővé teszi egy végfelhasználó számára, hogy különböző árucikkeket vásároljon képernyőn megjelenő katalógusokból, helyi híreket és időjárás térképeket "fogyasszon" kívánsága szerint és 2022 televízió készüléke segítségével játszszon.

A 4000 interaktív rendszer négy fő részből áll:

- a) a műsorszórási központban vagy valahol máshol telepített 4004 alkotói eszköztárat, amellyel a műsorszórási szolgáltató alkalmazásokat állíthat elő, fejleszthet, tesztelhet és követhet nyomon;
- b) a műsorszórási központban elhelyezett 4006 alkalmazás- és adatszervert, amely a 4004 alkotói eszköztárral áll kapcsolatban és lehetővé teszi a műsorszórási szolgáltató számára, hogy alkalmazásokat és adatokat készítsen elő, jogosítsa fel és alakítsa, amelyeket azután a 2004 multiplexer és bitsorkódolóhoz továbbíthat, hogy beillessze az MPEG-2 szállító jeláramba (jellemzően annak privát szekciójába), amelyet a végfelhasználónak sugároz ki;
- c) egy 4008 végrehajtás alatt futó főalgoritmust magában foglaló virtuális gép, amely a végfelhasználó által bérelt, vagy a tulajdonát képező 2020 vevő és dekódoló berendezésben installált végrehajtható kód, és amely lehetővé teszi a végfelhasználó számára, hogy a 2020 vevő és dekódoló berendezés memóriájába végrehajtás céljából alkalmazásokat vegyen jogosultságokat szerezzen, dekomprimáljon és betöltsön. A 4008 végrehajtás alatt futó főalgoritmust magában foglaló virtuális gép rezidens általános célú alkalmazásokat is futtat, és független a hardvertől és az operációs rendszertől.
- d) A 2020 vevő és dekódoló berendezés és a 4006 alkalmazás- és adatszerver között elhelyezkedő 4002 modemes vonal, amely lehetővé teszi a 4006 alkalmazás- és adatszervert a jelek segítségével arra utasítani, hogy a végfelhasználói kérelemre az MPEG-2 jeláramba adatokat és alkalmazásokat illesszen.

A 4000 interaktív rendszer olyan "alkalmazások" felhasználásával működik, amelyek a 2020 vevő és dekódoló berendezés valamint az abban lévő különböző eszközök egyes funkcióit vezérlik. Az alkalmazások a 4008 végrehajtás alatt futó főalgoritmust magában foglaló virtuális gépben "erőforrás fájlként" vannak jelen. Egy "modul" nem más, mint ilyen erőforrás fájlból és adatokból álló készlet. Egy alkalmazáshoz adott esetben több modulra lehet szükség. A 2020 vevő és dekódoló berendezés "memória kapacitása" az említett modulok számára rendelkezésre álló tárolóhely. A modulok letöltéséhez egy úgyne-



vezett "interfészt" használunk. A modulokat a 2020 vevő és dekódoló berendezésbe az MPEG-2 jeláramból tölthetjük le.

Az előző bekezdésben csupán felsorolt elemeket az alábbiakban részletesebben is bemutatjuk.

Leírásunkban alkalmazás alatt egy számítógépi kód készletet értünk, amellyel előnyösen a 2020 vevő és dekódoló berendezés magas szintű funkcióit vezéreljük. Például, ha a végfelhasználó 2026 távirányítóját (mint az a 4. ábrán jobban megfigyelhető), pontosabban annak fókuszált sugarát ráirányítja a 2022 televízió készülék képernyőjén látható gomb objektumra, és megnyomja a 2026 távirányító érvényesítés gombját, azzal lefuttatja a képernyőn látható nyomógommbal társított utasítás szekvenciát.

Egy interaktív alkalmazás menüket kínál fel, a végfelhasználó 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 (ezt a kifejezést később még részletezzük). Az alkalmazás kulcsfontosságú jellemzőket nyújt, amelyet szükség esetén a műsorszóró szolgáltató módosítani tud. Ezen kívül interfészt képez a rezidens alkalmazások valamint a letöltött alkalmazások között.
- Indító alkalmazás. Az indító alkalmazás teszi lehetővé bármely alkalmazás számára, legyen az letöltött vagy rezidens alkalmazás, hogy fusson a 2020 vevő és dekódoló berendezésen. Ez az alkalmazás egy betöltő rutinként szolgál, amely egy szolgáltatás megérkezésekor kerül végrehajtásra, annak érdekében, hogy az alkalmazás elindulhasson. Az indító alkalmazás a 2020 vevő és dekódoló berendezés RAM-jába kerül letöltésre, így könnyen frissíthető. Az indító alkalmazás úgy konfigurálható, hogy bármely csatornán rendelkezésre álló interaktív alkalmazásokat tudunk kiválasztani és futtatni,

vagy közvetlenül a letöltést követően, vagy egy előzetes letöltés utáni későbbi időpontban. Ez utóbbi esetben az alkalmazás a 2020 vevő és dekódoló berendezés 2024 memóriájába töltődik, és az indító alkalmazás igény szerint fogja aktiválni.

- Programkalauz. A programkalauz egy olyan interaktív alkalmazás, amely a programozásra nyújt teljes körű információt. Például információt adhat mondjuk egy digitális televízió csomag egyes csatornáin egy egész hét során sugárzásra kerülő televíziós műsorokról. Ha a végfelhasználó a 2026 távirányítón megnyom egy megfelelő gombot, a 2022 televízió készüléken éppen látható esemény képernyőre rányíló további képernyőt kap, amely nem más, mint egy olyan böngésző, amely információt nyújt a digitális televízió csomag egyes csatornáinak éppen futó aktuális és rákövetkező műsor eseményeiről. Ha a végfelhasználó a 2026 távirányítón most egy másik gombot nyom meg, akkor egy olyan alkalmazást indít el, amely kiad neki egy információs listát egy teljes hét eseményeiről. A végfelhasználó ilyen módon egyszerű és testre szabott feltételekkel keresheti és válogathatja ki a számára érdekes eseményeket, de éppígy közvetlenül hozzáférhet egy kiválasztott csatornához is.
- 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 a 3000 feltételes hozzáférési rendszerrel együttműködésben. A végfelhasználó ezt az alkalmazást egy televízió műsorfűzet vagy csatorna böngésző felhasználásával tudja indítani. Ezen túlmenően az alkalmazás automatikusan megindul abban a pillanatban, amint a nézd-és-fizess csatornán a műsorszóró szolgáltató egy nézd-és-fizess eseményt megindít. A végfelhasználó ekkor ezt az aktuális eseményt, azaz műsort meg tudja venni, vagy a saját gyermek 3020 programozható csipkártyáján, vagy a 3022 kommunikációs szerveren keresztül (modem, telefon és DTMF kódok, vagy MINITEL vagy hasonló rendszer használatával, természetesen). Ez az alkalmazás vagy rezidensként a 2020 vevő és dekódoló berendezés ROM-jában található, vagy pedig annak RAM-jába letölthető.
- 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.
- Vetélkedő alkalmazás. A vetélkedő alkalmazás célszerűen szinkronizált a kisugárzott vetélkedő programokkal. Például a 2022 televízió készülék képernyőjén több, kiválasztható válaszú kérdés jeleníthető meg, és a végfelhasználó a megjelenített válaszlehetőségekből a 2026 távirányító segítségével tudja az általa megfelelőnek tartott választ kiválasztani. A vetélkedő alkalmazás információt adhat a felhasználónak arról, hogy az általa kiválasztott válasz helyes volt-e vagy sem, és nyilvántartja a végfelhasználó által elért pontszámot is.
- 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. A kiválasztott árucikkre vonatkozó megrendelés a 4002 modemes vonalon keresztül jut vissza a 4006 alkalmazás- és adatszerverhez vagy egy olyan külön árusító rendszerhez, amelynek telefonszáma letöltésre került a 2020 vevő és dekódoló berendezéshez és a megrendeléshez lehetőség szerint egy folyószámla terhelési megbízás is társul egy olyan hitelkártya tekintetében, amelyet a végfelhasználó a 2020 vevő és dekódoló berendezés valamelyik 4036 kártyaolvasójába a rendelés elküldése előtt behelyezett.
- Telebank alkalmazás. A Telebank alkalmazás egy lehetséges megvalósítása esetében a végfelhasználó behelyezi hitelkártyáját vagy bankkártyáját a 2020 vevő és dekódoló berendezés valamelyik 4036 kártyaolvasójába. A 2020 vevő és dekódoló berendezés feltárcsazza a végfelhasználó bankját, amihez a bankkártyán eltárolt vagy pedig magában a 2020 vevő és dekódoló berendezésben korábban eltárolt telefonszámot használ, majd a Telebank alkalmazás különböző lehetőségeket kínál fel a végfelhasználónak, amelyek közül a végfelhasználó a 2026 segítségével választhat, például a telefonvonalon keresztül letöltheti a számlájára vonatkozó státusz információt, számlák közötti átutalásokat bonyolíthat, csekkfüzetet rendelhet, és így tovább.

- Internet böngésző alkalmazás. Az Internet böngésző alkalmazás egy megvalósítási példájában a végfelhasználótól kiinduló utasításokat, mint például egy meghatározott URL-lel rendelkező Web-lap megtekintésére vonatkozó kérelmet a felhasználó a 2026 távirányító segítségével tud bevinni, és a kérelmét tartalmazó adat a 4002 modemes vonalon keresztül jut a 4006 alkalmazás- és adatszerverhez. A kért Web-lapot a műsor-szóró központból induló jelátvitelbe beépítik és azt a 2020 vevő és dekódoló berendezés a 2012 földi állomáson, a 2014 transzponderen és a 2016 műholdon keresztül megkapja, majd továbbítja a 2022 televízió készülékhez, amelynek képernyőjén a végfelhasználó a Web-lapot megtekintheti.

Az alkalmazásokat a 2020 vevő és dekódoló berendezés memóriahelyein tároljuk és ezek erőforrás fájlkként és adatokként jelennek meg. Az erőforrás fájlok grafikus objektum leíró egység fájlkból, változó blokk egység fájlkból, utasítás sorrend fájlkból, alkalmazás fájlkból és adat fájlkból állnak.

A 4014 grafikus objektum leíró egység fájl a különböző képernyőket írja le, vagyis tulajdonképpen az alkalmazás ember-gép közötti interfészét. A 4016 változó blokk egység fájl az alkalmazás által megszólitott és "kezelt" adat struktúrákat írja le, a 4018 utasítás szekvencia fájl az alkalmazások feldolgozási folyamatait írja le, míg a 4012 alkalmazás fájl az alkalmazásokhoz tartozó egyes belépési pontokat definiálják.

Az ily módon felépített alkalmazások adat fájlkat, mint amilyenek az ikon könyvtár fájl, képfájl, betűtípus fájl, színtáblázat fájl és tiszta szöveges (ASCII) fájl, képek használni. Egy interaktív alkalmazás bemenetek és/vagy kimenetek felhasználásával online adatokat is nyerhet.

A 4008 végrehajtás alatt futó főalgoritmust magában foglaló virtuális gép csupán azokat az erőforrás fájlkat hívja be a memóriába, amelyekre egy adott időpontban éppen szükség van. Ezeket az erőforrás fájlkat a 4008 végrehajtás alatt futó főalgoritmust magában foglaló virtuális gép a 4014 grafikus objektum leíró egység fájlkból, 4018 utasítás szekvencia fájlkból, 4012 alkalmazás fájlkból és 4016 változó blokk egység fájlkból olvassa ki, amelyek a memóriában vannak eltárolva, méghozzá egy a modulok betöltésére szolgáló eljárás meghívását követően, és az erőforrás fájl mindaddig ott maradnak, amíg egy, a modulok kitöltésére, eltávolítására szolgáló eljárást meg nem hívunk.

A 3. ábrán látható, hogy egy 4010 modul, mint amilyen a részletesebben is bemutatásra kerülő vásárlási modul, erőforrás fájlok és adatok készleteként értelmezendő, amely az alábbiakat tartalmazza:

- egy egyetlen 4012 alkalmazás fájlt,
- meghatározatlan számú 4014 grafikus objektum leíró egység fájlt,
- meghatározatlan számú 4016 változó blokk egység fájlt,
- meghatározatlan számú 4018 utasítás szekvencia fájlt, és
- ahol szükséges 4020 adat fájlokat, például ikon könyvtár fájlokat, képfájlokat, betű-típus fájlokat, szintábrázat fájlokat és ASCII szöveges fájlokat.

Egy MPEG adatfolyamban minden egyes modul MPEG táblázat csoportokat tartalmaz. Minden egyes MPEG táblázatot adott számú szekcióként formázhatunk. Az MPEG adatfolyamban minden egyes szekció legfeljebb 4 kB méretű. A soros vagy a párhuzamos interfészen történő adatátvitelhez például az egyes modulokat hasonlóképpen táblázatokká és szekciókká osztjuk fel, ahol az egyes szekció mérete a szállító közegtől függően változik.

A modulokat az MPEG adatfolyamban általában 188 bájt nagyságú adatsomagokként továbbítjuk a különböző típusú adatfolyamokban, például video adatfolyamban, audio adatfolyamban és teletext adatfolyamban. Minden egyes csomagot egy 13 bit nagyságú csomagazonosító (packet identifier, PID) előz meg, ahol az MPEG adatáramban továbbított minden egyes csomaghoz egy ilyen csomagazonosító tartozik. A különböző adatfolyamok listáját egy úgynevezett programtérkép táblázat (program map table, PMT) tartalmazza, amely egyben definiálja az egyes adatfolyamok tartalmát a hozzájuk tartozó csomagazonosító szerint. A programtérkép táblázat segítségével azonosított csomagazonosító például egy eszközt értesíthet az adatfolyamban egy vele összefüggésben álló alkalmazás jelenlétéről.

A 4010 modulokat használó koncepció együtt egy kód csupán kis részeinek a letöltését célzó koncepcióval lehetővé teszi a különböző alkalmazások egyszerű kifejlesztését vagy továbbfejlesztését. Ezeket a kis kódrészeket a 2020 vevő és dekódoló berendezés FLASH memóriájába akár rezidens szoftverként is le tudjuk tölteni, vagy kisugározzuk őket annak érdekében, hogy csak akkor töltődjenek le egy 2020 vevő és dekódoló berendezés RAM-jába, ha a végfelhasználónak, ezen keresztül a 2020 vevő és dekódoló berendezésnek valóban szüksége van rájuk.

Egy memória kötet nem más, mint egy meghatározott tárolótér a 4010 modulok számára. Ezek a tárolóterek a 2020 vevő és dekódoló berendezés 2024 memóriájában található. Az 5. ábrán látható például, hogy a 2024 memória jellemző módon egy 4022 RAM kötetre, 4024 FLASH kötetre és 4026 ROM kötetre van felosztva. A 2024 memóriát ezen kívül azokkal a különböző interfészekkel társított memória kötetekre oszthatjuk, amelyeken keresztül az egyes 4010 modulokat a 2020 vevő és dekódoló berendezésbe töltjük le, így például egy MPEG kötetet hozhatunk létre, hogy eltároljuk azokat a 4010 modulokat, amelyeket egy MPEG bitfolyamból töltöttünk le, vagy egy soros kötetet hozhatunk létre a soros interfészen keresztül vett 4010 modulok tárolására.

A 4022 RAM kötetet is továbboszthatjuk egy, a főmver részére fenntartott zónára, a 4008 motor számára szükséges munkaterületre, valamint a pufferek számára fenntartott területre. A FLASH memóriát és a többi nem felejtő memóriát vagy egy alkalmazáson keresztül vagy pedig egy eszközközlelőn át a 4008 motorral tudjuk elérni.

Minden egyes kötet tartalmaz egy 4010 modul listát, és minden egyes 4010 modul tartalmaz egy listát, amelyben 4012, 4014, 4016, 4018, 4020 fájlok vannak felsorolva. Lehetőség van arra, hogy két 4012, 4014, 4016, 4018, 4020 fájlnek ugyanaz legyen a neve, de ezeknek különböző 4010 modulokban kell elhelyezkedniük. Például egy alkalmazás egy adott változata általában a 4026 ROM kötetben van eltárolva, az alkalmazás későbbi változatai pedig 4024 FLASH kötetbe tölthetők le, hogy helyettesítsük velük az eredetileg a 4026 ROM kötetben eltárolt eredeti változatot. A 4012, 4014, 4016, 4018, 4020 fájlok tartalma például LZW formátumban tömörített lehet, azonban mivel a tömörített fájlok kifejtése mindenképpen bizonyos időt vesz igénybe, ha az idő lényeges tényező, akkor azt tömörítés nélkül is vehetjük.

A 2020 vevő és dekódoló berendezés fizikai interfészeit használjuk az adatok letöltéséhez. A 6. ábrán látható, hogy a 2020 vevő és dekódoló berendezés például hat letöltéshez használt közeget foglalhat magában, nevezetesen 4028 MPEG folyam tunert, 4030 soros interfészt, 4032 párhuzamos interfészt, 4034 modemet és két 4036 kártyaolvasót.

Tekintettel az alkalmazások különböző eredetére, valamint az egyes 2020 vevő és dekódoló berendezések számos gyártójára és típusára, lényeges, hogy egy adott alkalmazás ugyanolyan módon viselkedjen minden egyes 2020 vevő és dekódoló berendezésen futtatva és fordítva, minden egyes 2020 vevő és dekódoló berendezés minden egyes alkalmazást ugyanolyan megfelelő módon és eredményesen tudjon lefuttatni. A 7. ábrán megfigyelhe-

tő, hogy a 2020 vevő és dekódoló berendezés egy mikroprocesszor vezérlete alatt működő 4008 főalgoritmust, valamint egy közös 4054 alkalmazás programozó interfészt (Application Programming Interface, API) tartalmaz. Ezek minden egyes 2020 vevő és dekódoló berendezésben megtalálhatók, úgy hogy a 2020 vevő és dekódoló berendezések az alkalmazások felől nézve egymással csereszabatosak.

A 7. ábra 4056 alkalmazásokat futtató 2020 vevő és dekódoló berendezés felépítését mutatja vázlatosan. 4007 virtuális gép hajtja végre a 4056 alkalmazásokat, amelyek közvetlenül a 4007 virtuális géppel kapcsolatban álló 4056' alkalmazásokat, valamint például egy MPEG adatáramból a 2020 vevő és dekódoló berendezésbe letöltött 4056" alkalmazásokat foglalhatnak magukban. A 4008 főalgoritmus ezen kívül gondoskodik a grafikák és a szövegek megjelenítéséről, a különböző szolgáltatásokhoz szükséges eszközök hívásáról, veszi a különböző "eseményeket", és meghatározott számítási műveletek elvégzéséhez használja egy 4058 könyvtár funkcióit.

A 7. ábra vonatkozásában egy alkalmazást tekintve a 2020 vevő és dekódoló berendezés egy-egy funkcióját egy 4060 eszközként "látjuk". Létezhetnek ezért a 2020 vevő és dekódoló berendezésnek olyan funkciói is, amelyeket egy adott alkalmazás nem lát.

A 4007 virtuális géppel kommunikáló prezentációs funkció szolgál a szövegek és grafikák végfelhasználó számára történő bemutatására, valamint a végfelhasználói akcióknak a 4007 virtuális géphez történő továbbítására. A szöveg és a grafika a 2022 televízió készülék kijelzőjén jelenik meg, és a végfelhasználó a 4056 alkalmazással együttműködik a billentyű használatával, mely billentyű legáltalánosabb esetben a 2026 távirányítót jelenti.

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.

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 benne lévő 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 4007 virtuális gép ezért olyan értelmezőt tartalmaz, amely az alkalmazási kódot interpretálva elvégzi a különböző változók értékeinek felfrissítését, meghatározza az állapot változásokat, és hibaellenőrzést végez.

A 6. ábra vonatkozásában egy alkalmazást tekintve a 2020 vevő és dekódoló berendezés egy-egy funkcióját egy 4060 eszközként "látjuk". Létezhetnek ezért a 2020 vevő és dekódoló berendezésnek olyan funkciói is, amelyeket egy adott alkalmazás nem lát.

Egy 4060 eszköz olyan logikai egységet tartalmaz, amely megfelel a 4066 hardver 4062 komponensének vagy 4064 fizikai interfészének. Az ilyen 4060 eszközöket "alacsony szintű" 4068 eszközöknek nevezzük. Egy ilyen 4068 eszköz kimenete legalább egy 4070 eszközmeghajtóhoz kapcsolódhat, hogy a 4068 eszköz által kibocsátott logikai jeleket például egy 4064 fizikai interfész meghajtásához szükséges jelekké alakítsuk át. Alternatív módon a 4068 eszköz önmaga is meghajthatja a 2020 vevő és dekódoló berendezés egy 4062 komponensét vagy 4064 fizikai interfészét, ilyen esetben a 4068 eszköz kimenete közvetlenül csatlakozik a 4066 hardverhez.

Az alábbiakban néhány példát adunk az alacsony szintű 4068 eszközökre.

Egy LCARD eszköz lehetővé teszi egy program számára, hogy az egyik 4036 kártyaolvasóban lévő 3010 programozható csipkártyával kommunikáljon, egy LCARD eszköz pedig lehetővé teszi egy program számára, hogy a másik 4036 kártyaolvasóban lévő 3010 programozható csipkártyával kommunikáljon. Ezek az eszközök például lehetővé tehetik egy program számára, hogy kiolvassa a 3010 programozható csipkártya állapotát, elolvassa a

3010 programozható csipkártya eseménynaplóját, és bemenő üzenetet küldjön a 3010 programozható csipkártyára. Az eszközök ezen kívül egy programot értesítenek arról is, hogy egy 3010 programozható csipkártya lett valamelyik 4036 kártyaolvasóba behelyezve, vagy onnan el lett távolítva, továbbá jelzik azt is, ha a 3010 programozható csipkártyát alaphelyzetbe állítottuk vissza abban az esetben, ha ezt nem maga a program igényelte. Az LCARD és RCARD eszközök a kártya használatához szükséges protokollokhoz igazodnak, napjainkban általában az ISO 7816 jelzésű protokollt használják.

Egy SCTV eszköz lehetővé teszi egy program számára, hogy ellenőrizze és konfigurálja a 2022 televíziókészülékhez vezető SCART csatlakozót. Például ez az eszköz lehetővé teszi egy program számára, hogy információt kérjen be a SCART kimeneten jelen lévő hang jellemzőkről, elnémítást idézzen elő, vagy dinamikusan programozza az RGB szinteket.

Egy TUNER eszköz lehetővé teszi egy program számára, hogy az használja a 4028 MPEG folyam tunert. Például az eszköz lehetővé teszi a program számára, hogy frekvencia tapogató módon végighaladjon a sávon, vagy a legkisebb frekvenciától kezdve, vagy a 4028 MPEG folyam tuner éppen beállított frekvenciájától kezdve, kiolvassa a 4028 paramétereit és programozza azt.

Egy SERIAL eszköz lehetővé teszi egy program számára, hogy egy berendezéssel egy soros kapcsolaton keresztül folytasson kommunikációt, míg egy PARALLEL eszköz lehetővé teszi egy program számára, hogy egy berendezéssel párhuzamos kapcsolaton keresztül folytasson kommunikációt. Például ezek az eszközök lehetővé teszik egy program számára, hogy üzenetet küldjön a megfelelő kapcsolaton keresztül, és jelzik továbbá egy program számára, ha a megfelelő kapcsolaton keresztül kívülről egy üzenet érkezett.

Egy MODEM eszköz lehetővé teszi a 2020 vevő és dekódoló berendezés számára, hogy egy belső félduplex, V23 szabványt támogató modemen keresztül valamilyen adatszolgáltatóval kommunikáljon. Ez a MODEM eszköz gondoskodik a telefonszám tárcsázásáról, egy üzenet elküldéséről az adatszolgáltatóhoz, a modem leválasztásáról, jelzi egy üzenet vételét, jelzi a felismert hibákat, valamint jelzi a vivőjel eltűnését vagy megjelenését.

Egy távoli helyen végrehajtásra kerülő távoli eszköz a helyi eszközök bármelyike lehet, azzal az eltéréssel, hogy mindenképpen meg kell adnunk egy portot valamint egy protokollt.

Az alacsony szintű 4068 eszközökön túl a 2020 vevő és dekódoló berendezés "magas szintű" 4072 eszközöket is tartalmazhat, amelyek lényegében a 2020 vevő és dekódoló berendezés működését vezérlik.

Az alábbiakban a 4072 magas szintű eszközökre mutatunk néhány példát.

Egy MLOAD eszköz lehetővé teszi egy alkalmazás számára, hogy hardver és szoftver szűrési kritériumokban MPEG bitfolyamból egy MPEG szekciót egy teljes MPEG táblázatot vagy MPEG szekció csoportot betöltsön. Például az MLOAD eszköz lehetővé teszi egy program számára, hogy egy csoportból csak azokat a szekciókat töltsse le, amelyek egy adott időben egy alkalmazás futtatásához szükségesek.

Egy FDLOAD eszköz automatikusan kezeli a 4030 soros interfészen és/vagy a 4032 párhuzamos interfészen keresztül a fájlok letöltését. Az FDLOAD eszköz a fájlok letöltésének soros vagy párhuzamos interfészen keresztül történő megkezdését vagy befejezését is jelezni tudja, ha egy program erre a jelzésre igényt tart. Az FDLOAD eszköz a program számára olyan kérelmet bocsáthat ki, hogy a program bocsássa rendelkezésére a letöltendő fájl PID azonosítóját valamint a dekódolásához szükséges PID ECM kódot tartalmazó index táblázatot. A program betölti ezt az index táblázatot és elküldi az eszköznek, amely ezt követően kérelmezi a fájl letöltését a program felé. A program kivonja az index táblázatból a fájlhoz szükséges PID és PID ECM azonosítókat, és demultiplexelő kérelmet bocsát ki, hogy lehetővé váljék a letöltendő fájl teljes egészében történő letöltése. A program bármikor kibocsáthat egy "fdload_offline" amellyel az FDLOAD eszközt arra utasítja, hogy álljon le a 4030 soros interfész és/vagy a 4032 párhuzamos interfész kezelésével.

Az eszközöket is egy egyedi azonosítóval azonosítjuk, például "LCARD_DEVICE_ID" az LCARD eszközt azonosítja, az "RCARD_DEVICE_ID" az RCARD eszközt azonosítja.

Ha egy új 4062 eszközt hozunk létre, azt a már létező és működő 2020 vevő és dekódoló berendezésekben is telepíthetjük úgy, hogy a releváns 4056 alkalmazást letöltjük a műsor-szóró központból.

Ezt a letöltést a 2020 vevő és dekódoló berendezésben olyan 4056 alkalmazással hajtjuk végre, amely előbb leellenőrzi a hardver és szoftver verziót, és amennyiben ezeket megfelelőnek találja, az új 4062 eszközt képviselő szoftver modult letölti, majd a 4058 eszközkészlet egy műveletét felhívja, hogy telepítse az új eszköz kódot a főmemóriában, például a FLASH memóriában. Ezzel úgy tudunk új funkciókat a 2020 vevő és dekódoló berendezés-

sen belül rugalmasan és biztonságosan telepíteni, hogy a működtető szoftver többi részét nem érintjük és nem befolyásoljuk.

4074 eszközközkezelő egy közös interfészt képez a 4056 alkalmazás és a 2020 vevő és dekódoló berendezés meghatározott funkcióihoz társított 4060 eszközök között. Egy ilyen közös interfész felhasználásával a 4008 főalgoritmus "könyvtárában" tárolt és a 4060 eszközökhöz hozzáférést lehetővé tevő műveleteket valamint a 4060 eszközökkel társított "eseményeket" egy korlátozott műveletszám segítségével szabványosítani tudjuk. A 4074 eszközközkezelő felügyeli a hozzáférést a 4060 eszközökhöz, jelzi egy nem várt esemény vételét, valamint kezeli a megosztott memóriát.

18 olyan műveletet sorolhatunk fel, amelyek hozzáférést biztosítanak a 4074 eszközközkezelőhöz, magához egy 4060 eszközhöz, vagy a memória kezeléshez:

Device_Open_Channel ()	Megnyit egy csatornát a 4074 eszközközkezelőhöz
Device_Close_Channel ()	Bezár egy csatornát a 4074 eszközközkezelőhöz
Device_Open_Channel()	Megnyit egy csatornát a 4060 eszközhöz
Device_Close_Channel ()	Bezár egy csatornát a 4060 eszközhöz
Device_Event ()	Nem várt esemény kezelés
Device_Call ()	Szinkron hozzáférés egy 4060 eszközhöz
Device_Io ()	Aszinkron hozzáférés egy 4060 eszközhöz
Device_Info ()	Információ egy 4060 eszközhöz
Set_Buffer_outline ()	Memória partíciókat definiál a pufferek számára
Get_Buffer_outline ()	A jelenlévő puffertár particionálási területet olvassa
Pool_Info ()	Információt nyújt a jelenlegi puffertár particionálási területről
Device_Alloc_Buffer ()	Memóriát foglal le
Device_Free_Buffer ()	Memóriát szabadít fel
Device_Lock_Buffer ()	Memóriát zárol
Device_Info_Free_Buffer ()	A szabad pufferek számát adja
Device_Info_Alloc_Buffer ()	Listázza a lefoglalt puffereket
Get_Buffer_Size ()	Megadja a lefoglalt pufferek méretét

A 7. ábrára áttérve, 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 eszközközke-

zelőhöz. A 4068 eszközközkezelő az újonnan létrehozott kliensnek egy olyan kliensszámot 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 "Device_Open_Channel" elnevezésű művelettel vezethetünk be. Ezzel a művelettel a 4074 eszközközkezelő a 4076 kliensnek megadja a kliensszámát, vagyis egy olyan "client_id" kliensazonosítót ad ki, amely tartalmazza a lefoglalt kliensszámot magában foglaló változó címét (sohasem lehet két azonos számú 4076 kliens) valamint egy olyan "error_code" hibakódot ad ki, amely vagy egy "0" hibakódot tartalmaz, amely azt jelzi, hogy a művelet sikeresen befejeződött, vagy pedig egy "e_client_max" hibakódot tartalmaz, amely azt jelzi, hogy a 4074 eszközközkezelő által kezelhető számú 4076 kliens, jellemzően 256 darab 4076 kliens megvan, a létszám betelt.

Az első lefoglalásra kerülő kliensszám 0, mely érték egyesével nő, egészen 255-ig, a "Device_Open_Channel" műveletet használó minden egyes híváskor. Ez attól függetlenül következik be, hogy a kliens listáról eltávolítottunk-e esetleg egy 4076 klienst. Ha a 255-ös kliensszám lefoglalásra került, és ezután következik be egy "Device_Open_Channel" hívás, akkor vagy az "e_client_max" hibakódot adjuk ki, jelezve ezzel, hogy a kliens lista "betelt", vagy pedig a 4076 kliensnek a rendelkezésre álló legkisebb kliensszámot adjuk, vagyis a kliens listáról ezt megelőzően eltávolított valamely 4076 kliensnek a volt kliensszámát.

Egy 4076 kliens 4060 eszközzé nyilvánításához egy 4076 kliens a 4080 "Device_Open_Device ()" műveletet használja fel, és azzal továbbítja "client_id" kliensazonosítóját és "device_id" eszközazonosítóját. A 4074 eszközközkezelő "0" hibáüzenetet ad ki, ha a művelet sikeresen befejeződött, valamint "e_client_inconnu" hibáüzenetet bocsát ki, ha a 4076 kliens nem létezik, "e_periph_inconnu" hibáüzenetet, ha a 4060 eszköz nem létezik, és "e_client_max" vagy "w_deja_vu" hibáüzenetet, ha a 4076 klienst már korábban a 4060 eszközként deklaráltuk.

Egy 4076 kliens egy 4060 eszköz listáról a Device_Close_Device () művelettel távolíthatjuk el. Ehhez a 4074 eszközközkezelőnek megadjuk a "client_id" kliensazonosítót és a "device_id" eszközazonosítót, amely vagy "0" hibakódot bocsát ki, ha a műveletet eredm-

nyesen és sikeresen befejezte, vagy "w_client_inconnu" hibakódot, ha a 4076 kliens ismeretlen a 4074 eszközzel számára, vagy pedig a "w_periph_inconnu" hibakódot, ha a 4060 eszköz ismeretlen a 4074 eszközzel számára.

Egy 4076 klienst a 4074 eszközzel kliens listájáról a "Device_Close_Channel ()" művelettel tudunk eltávolítani. A "client_id" kliensazonosítót megadjuk ehhez a 4074 eszközzel, amely erre vagy "0" "error_code" hibakódot ad ki, ha a művelet eredményesen be tudta fejezni, vagy "w_client_inconnu" hibakódot, ha a 4076 kliens számára ismeretlen. Ez a művelet a 4076 klienst kitörli a 4074 eszközzel kliens listájáról valamint minden egyes 4060 eszköz olyan listájáról, amelyen kliensként szerepelt, például ha azt az alkalmazás nem tette meg, úgy a "Device_Close_Channel ()" művelettel meghívjuk a "Device_Call_Device ()" műveletet minden egyes olyan eszközhöz, amely vonatkozásában kliensként szerepel a 4076 kliens. Az így felszabadított kliensszámot a "Device_Open_Channel ()" művelet segítségével ezután egy újabb 4076 kliens részére le tudjuk foglalni.

A "Device_Info ()" információt nyújt egy 4076 kliens számára egy 4060 eszközről. Ehhez a "client_id" kliensazonosítót és a "device_id" eszközazonosítót megadjuk a 4074 eszközzel, amely erre válaszként a 4060 eszközre vonatkozó információt ad vissza, jellemzően a szóban forgó 4060 eszköz verzió számát, a 4060 eszközhöz társítható 4076 kliensek maximális számát, valamint a 4076 kliensek tényleges számát, akik éppen a 4060 eszközt használják, együtt adott esetben olyan hibakóddal, amellyel jelzi, hogy a művelet eredményesen befejeződött, vagy hogy a 4076 kliens vagy a 4060 eszköz számára (a 4074 eszközzel számára) ismeretlen.

Az "Device_Event" művelet az úgynevezett váratlan események lekezelésére szolgáló eszköz, amely akkor jelentkezik, ha valamilyen nem várt körülmény vagy más jelenség lépett fel. A "Device_Event" művelet lehetővé teszi egy 4076 kliens számára, hogy saját magát jelölje egy 4060 eszköz váratlan eseményéről szóló üzenet vevőjének.

Az alkalmazásoktól érkező üzeneteket, amelyeket "eseményeknek" nevezünk, a 4008 főalgoritmus adott esetben öt várakozási sora közül az egyikbe helyezzük. Minden egyes várakozási sor egy-egy prioritási szintnek felel meg (ahol 4 a legnagyobb prioritást, 0 pedig a legkisebb prioritást, elsőbbséget jelenti). Ha egy várakozási sorból egy alkalmazáshoz történő továbbítás céljára üzeneteket bontunk ki, akkor a 4008 főalgoritmus végignézi a legmagasabb prioritási szintű várakozási sort, hogy tartalmaz-e eseményt. Ezt az eseményt

eltávolítja a várakozási sorból, és aktiválja vele a hozzátartozó művelet sorbarendező egységet.

Minden úgynevezett "külső" eseményt, amelyet akár a billentyűzet segítségével juttatunk el a prezentációs funkcióhoz vagy egy interfészen keresztül fogadunk, egy esemény interfészen vezetünk keresztül, mielőtt a 4008 főalgoritmus elé kerül. Az úgynevezett "belső" események azonban, amelyeket általában belső eszközök generálnak, nem haladnak keresztül ezen az esemény interfészen, hanem közvetlenül jutnak a 4008 főalgoritmushoz. Minden egyes művelet esetében az utolsó hívó paraméter annak a változónak a címe lesz, amelyet a 4074 eszközközkezelő állít be, jelezve ezzel, hogy a parancsot hogyan kell a 4074 eszközközkezelőnek lekezelnie.

Egy nem várt eseménynek egy 4076 klienshez történő kibocsátásának az oka mindig a 4060 eszközön múlik. Például, a már említett LCARD DEVICE vonatkozásában a nem várt események magukban foglalják:

- egy csipkártya eltávolítását a kártyaolvasóból;
- egy csipkártya behelyezését a kártyaolvasóba; vagy
- a csipkártya kliens általi visszaállítását.

Mindegyik fenti eseményt egy hozzátartozó egyedi eseménykód ("ev_lcard_extract", "ev_lcard_insert" és "ev_card_reset") azonosít.

A nem várt események fogadásához a 4076 kliensnek önmagát kell minden fogadni kívánt esemény fogadójaként, címzettjeként deklarálnia a "device_event" művelet alkalmazásával. A 4074 eszközközkezelőnek az alábbi paramétereket kell megadni:

- "client_id" kliensazonosítót;
- "device_id" eszköazonosítót;
- "get_event" elnevezésű parancsot, amely engedélyezi egy nem várt esemény elküldését;
- az eseménykódot; és
- a 0 és 4 közötti esemény prioritást.

Egyes 4060 eszközök ugyanannak a nem várt eseménynek a vételét is deklarállhatják, ugyanazzal, vagy akár eltérő prioritással. Ebben az esetben az eseményt egy meghatározott sorrendben minden egyes 4076 kliens számára el kell küldeni (a sorrendet a prioritás és a kliensszám határozza meg).

Például 2 kliens saját magát deklarálja annak az `ev_card_extract` nevű eseménynek a vevőjeként, amely az LCARD eszköztől kettes prioritással érkezik. 3 kliens magát deklarálja ugyanannak az eseménynek a vevőjeként abban az esetben, ha az esemény kettes prioritású, míg 1 kliens a négyes prioritású azonos eseményt fogadja, 4 kliens pedig a hármas prioritású eseményt veszi. Ha a csipkártyát eltávolítjuk a kártyaolvasóból, az alábbi esemény sorozat zajlik le:

- az `ev_card_extract` esemény, amely az 1 kliensnek van címezve, bekerül a 4008 főalgoritmus négyes prioritású várakozási sorába;
- ugyanaz az esemény, amely a 4 kliensnek van címezve, bekerül a 4008 főalgoritmus hármas prioritású várakozási sorába;
- ugyanaz az esemény, amely a 2 kliensnek van címezve, bekerül a 4008 főalgoritmus kettes prioritású várakozási sorába; és
- ugyanaz az esemény, amely a 3 kliensnek van címezve, bekerül a 4008 főalgoritmus kettes prioritású várakozási sorába.

Egyes 4060 eszközök számára egy nem várt esemény során egy puffertárat is alkalmaznunk kell. Ilyen esetben a 4060 eszköz ezt a puffertárat a 4068 eszközkezelő segítségével foglalja le.

Az alkalmazás programozási nyelvben egy művelet eseményekhez társítható paraméterekhez biztosít hozzáférést:

- "event_code" (eseménykód);
- "client_id" (kliensazonosító);
- "evt_param1" (jellemzően 4 bájt nagyságú), és
- "evt_param2" (jellemzően 2 bájt nagyságú).

Az "evt_param" paraméterek mindenkori pontos jelentése a 4060 eszköztől valamint az esemény elküldésének az okától függ. Egy puffertárnak egy nem várt eseményhez történő társítása a 4060 eszköztől függ. A társítás létrejötte esetén az "evt_param1" paraméter a társított puffertár adatzóna címének felel meg. Ha nincs puffertár, az esemény ugyan kibocsátásra kerül, de ebben az esetben az "evt_param1" paraméter értéke "0" lesz.

Ha a 4060 eszköznek egy nem várt eseménnyel társított puffertára van, akkor egy 4076 kliensnek egy eszköz listáról vagy eszközkezelő listáról történő eltávolítása a puffertár fel-

szabadítását fogja okozni, kivéve, ha a szóban forgó puffertárat a "device_loc_buffer" művelettel az alkalmazás részére "zároltuk".

Az "evt_param2" paraméter azt jelzi, hogy a művelet eredményesen befejeződött és a 4060 eszköz részére használható eredményt kapott.

Egy 4076 kliens a nem várt események vételét a "get_event" parancs helyett a "ret_event" paranccsal is leállíthatja.

A 4074 eszközközkezelő a 4060 eszközökhöz két különböző módon, nevezetesen szinkron hozzáféréssel vagy aszinkron hozzáféréssel biztosíthat hozzáférést a 4076 kliensek részére.

A "device_call" művelet használatával végrehajtott szinkron hozzáférés egy eszközspezifikus funkcióhoz történő hozzáféréshez szolgál, amely nem foglalja magában az eszköztől érkező válasz 4008 főalgoritmus valamelyik várakozási sorába történő behelyezését, hanem a válasz azon nyomban rendelkezésre áll. Ezt a műveletet tipikusan a 2020 vevő és dekódoló berendezés hardver interfészeinek a konfigurálása során használjuk.

Ezt a műveletet használva a 4076 kliens az alábbi paramétereket továbbítja a 4060 eszközhöz a 4074 eszközközkezelőn keresztül:

- "client_id" kliensazonosító;
- "device_id" eszközazonosító;
- "call_cmde" paraméter, amely az adott 4060 eszköz által végrehajtandó művelet típusát tartalmazza. Például egy program számára a berendezéssel soros összeköttetésen keresztül történő kommunikációt lehetővé tevő SERIAL eszköz vonatkozásában a "serial_setup" parancs felépíti a soros összeköttetés konfigurációt;
- "em_adr" paraméter, amely a bevitt adatok memória helyének a címét tartalmazza, és
- "rec_adr" paraméter, amely annak a memória helynek a címét tartalmazza, amelybe a 4060 eszköznek a kimenő adatokat kell írnia.

A 4076 kliens által kért művelet befejezését követően a 4060 eszköz egy "call_report" adatot ad ki, amely tulajdonképpen az alkalmazás futtatásának a jelentését tartalmazza. A "call_report" paraméter általában egy 4076 kliens számára kibocsátható, előre meghatározott számú jelentések közül az egyik. Erre fel a 4074 eszközközkezelő egy "error_code" hibakódot bocsát ki a 4076 kliensnek, amely az alábbiak egyikét tartalmazhatja jellemzően:

- "e_client_inconnu";

- "e_periph_inconnu";
- "e_cmde_inconnu" paraméter, amely jelzi a 4076 kliensnek, hogy a szóban forgó parancs számára ismeretlen volt;
- "e_report_periph" paraméter, amely a 4060 eszköz eszközszerkezetes hibát jelent, és
- "0", amely azt jelzi, hogy a művelet sikeresen befejeződött.

A 4060 eszköz csupán akkor írja a kimenő adatokat a "rec_adr" paraméter által azonosított memóriahely címre, ha megkapja a "0" hibakódot.

A "Device_call" paraméter az összes többi alkalmazást leállítja mindaddig, amíg a jelenlegi alkalmazás futtatását a 4060 eszköz teljesen be nem fejezi.

A "Device_Io" művelet alkalmazásával végrehajtott aszinkron hozzáférés egy adott készülékre jellemző funkcióhoz történő hozzáférésre szolgál, viszont ellentétben a szinkron hozzáféréssel, ez magában foglalja a válaszra való várakozást is. Például, ismét a SERIAL eszköz vonatkozásában a "serial-send" parancs egy üzenetet küld a soros kapcsolaton keresztül. Az összes adat eredményes átvitelét követően (mielőtt még bármilyen előre megadott időtúllépés bekövetkezne) a 4060 eszköz jelentést küld a programnak, és ha ez a jelentés rendelkezésre áll, úgy a 4008 főalgoritmus várakozási sorába egy esemény kerül, jelezve a jelentés megérkezését.

A művelet alkalmazásával a 4076 kliens az alábbi paramétereket adja meg a 4076 eszköznek a 4074 eszközkezelőn keresztül:

- "client_id" kliensazonosító;
- "device_id" eszközazonosító;
- "io_cmde" paraméter, amely a 4060 eszköz által végrehajtandó művelet típusát tartalmazza;
- a művelet végével társított eseménykód;
- az eseménnyel társított esemény prioritás;
- "em_adr" paraméter; és
- "rec_adr" paraméter.

A 4076 kliens által kért művelet befejezését követően a 4060 eszköz egy "io_report" paramétert bocsát ki, amely a program futtatásának a jelentését tartalmazza.

Az "io_report" paraméter jellemző módon egy 4076 kliens számára kibocsátható, előre meghatározott számú jelentések közül egyet, valamint az eseménnyel társított két jelentés közül egyet tartalmaz (ahol az említett két jelentés egyike az "e_not_done" paraméter, amely jelzi, hogy a kért információ még nem áll rendelkezésre a 4076 kliens számára, míg a másik, "0" paraméter jelzi, hogy a kért információ már rendelkezésre áll).

Ez utóbbi "0" paramétert tartalmazó jelentés vételét követően a 4076 eszközközkezelő egy olyan "error_code" hibakódot bocsát ki a 4076 kliens számára, amely az alább felsorolt paraméterek valamelyikét tartalmazza:

- "e_client_inconnu";
- "e_periph_inconnu";
- "e_cmde_inconnu";
- "e_cmde_inconnue" paraméter, amely jelzi, hogy az eseménykód ismeretlen;
- "e_priorite_inconnue" paraméter, amely jelzi, hogy a prioritás ismeretlen;
- "e_report_periph";
- "0".

Kihangsúlyozzuk, hogy a találmányt csupán a megértéséhez szükséges mértékben, példák segítségével mutattuk be, és a találmánynak a szabadalmi igénypontokkal körülbástyázott oltalmi körén belül számos részletbeli módosítás tehető.

A leírásból, igénypontokból és rajzokból kivehető minden egyes jellemző akár külön, akár tetszőleges kombinációban létrehozható.

A találmány bemutatásánál leírt előnyös kiviteli alakoknál a találmány néhány, lényeges és nem annyira lényeges jellemzőjét számítógép program segítségével valósítottuk meg, azonban szakember számára nyilvánvaló és egyértelmű, hogy ezek a jellemzők kivétel nélkül hardveresen is megvalósíthatók. Ezen túlmenően nyilvánvaló, hogy a hardveresen, szoftveresen és hasonló módon megvalósított funkciókat villamos és elektronikus jelek felhasználásával hajtjuk végre.

Oltalmi igényünkkel kapcsolatosan hivatkozunk az alább felsorolt függő szabadalmi bejelentéseinkre is, melyek bejelentési dátuma megegyezik a jelen szabadalmi bejelentés bejelentési dátumával, és a bennük foglalt kitanításokat jelen leírásunkban is lényeges ismeretanyagnak tekintjük. (PCT/EP97/02106, "Jejelőállító és sugárzó eljárás és berendezés"; PCT/EP97/02107, "Adattárolásra szolgáló kártya (smartcard) kódolt sugárzott jeleket vevő

vevőberendezéshez, továbbá vevőberendezés"; PCT/EP97/02108, "Információ sugárzó és vevő rendszer, valamint ahhoz társítható hozzáférés-engedélyező rendszer"; PCT/EP97/02109, "Eljárás és rendszer fájl számítógépre történő letöltésére egy adóról dekóder vevőberendezés segítségével"; PCT/EP97/02110, "Eljárás és berendezés televízió műsorok és más adatok adására és vételére"; PCT/EP97/02111, "Eljárás adatok MPEG dekóder/vevőberendezésbe történő letöltésére, továbbá MPEG átviteli rendszer az eljárás megvalósítására"; PCT/EP97/02112, "Eljárás és rendszer számítógép memória egység szervezésére"; PCT/EP97/02113, "Eljárás televízió vagy rádió vezérlőrendszer fejlesztésére"; PCT/EP97/02114, "Eljárás és berendezés adatrészek kivonására egy továbbított adatáramból"; PCT/EP97/02116, "Vevő-dekódoló egység digitális sugárzású televízió rendszerhez"; PCT/EP97/02117, "Műsorsugárzó és vevő rendszer, továbbá vevő és dekódoló berendezés, és távirányító a berendezéshez"; PCT/EP98/01604, "Eljárás és berendezés adatok vevő-dekódoló berendezéshez továbbítására"; PCT/EP98/01605, "Eljárás és rendszer memória puffertár címkezelésére"; PCT/EP98/01606, "Eljárás feltételes hozzáférésű rendszerben a csalárd hozzáférések megakadályozására, továbbá adóberendezés valamint vevő és dekódoló berendezés az eljárás megvalósítására"; PCT/EP98/01607, "Kombinált számítógép-dekóder rendszer digitálisan sugárzott adatok vételére, továbbá számítógép, továbbá dekóder a rendszerhez".

Szabadalmi igénypontok

1. Eljárás kommunikációs csatorna létesítésére egy program és egy számítógép rendszer egy összetevője között, **azzal jellemezve**, hogy a számítógép rendszerben lévő összetevőkkel társított logikai eszközöket tárolunk el, ahol minden egyes logikai eszköznek egy eszközazonosítót adunk; a programhoz egy program azonosítót társítunk; és a programból egy olyan jelet bocsátunk ki, amely tartalmazza a program azonosítót és egy logikai eszköz eszközazonosítóját, a program és az egyes logikai eszközök között létrehozott eszközkezelőnek, mely utóbbival egy kommunikációs csatornát létesítünk a program és a logikai eszköz között, a kapott program azonosítót és eszközazonosítót felhasználva, lehetővé téve ezzel a program és a logikai eszköz közötti ezt követő kommunikációt.
2. Az 1. igénypont szerinti eljárás, **azzal jellemezve**, hogy a program egy alkalmazásnak legalább egy részét alkotja, és a számítógép rendszer egy vevő és dekódoló berendezésnek (2020) legalább részét képezi.
3. Eljárás egy kommunikációs csatorna létesítésére, egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, **azzal jellemezve**, hogy a vevő és dekódoló berendezés (2020) egyes összetevőivel társított logikai eszközöket tárolunk el, ahol mindegyik logikai eszközhöz egy-egy eszközazonosítót rendelünk hozzá; az alkalmazáshoz egy alkalmazás azonosítót rendelünk hozzá; és az alkalmazástól az alkalmazás azonosítót valamint egy logikai eszköz eszközazonosítóját tartalmazó jelet bocsátunk ki egy, az alkalmazás valamint az egyes logikai eszközök között létrehozott közös interfészhez, amellyel kommunikációs csatornát létesítünk az alkalmazás és a logikai eszköz között a közös interfészen keresztül.
4. A 3. igénypont szerinti eljárás, **azzal jellemezve**, hogy a közös interfész a kapott alkalmazás azonosító és eszközazonosító felhasználásával az alkalmazás és a logikai eszköz között kommunikációs csatornát létesítő eszközkezelőt tartalmaz.
5. A 2. vagy 4. igénypont szerinti eljárás, **azzal jellemezve**, hogy az alkalmazás azonosítót az eszközkezelővel rendeljük az alkalmazáshoz.
6. A 2-5. igénypontok bármelyike szerinti eljárás, **azzal jellemezve**, hogy az alkalmazás által egy, a logikai eszközzel társított összetevő állapotváltozását jelző, a logikai eszköz által kibocsátott üzenet vételét engedélyező parancsot tartalmaz.

7. A 6. igénypont szerinti eljárás, **azzal jellemezve**, hogy az üzenetet átmenetileg az alkalmazáshoz történő rákövetkező továbbítás céljára egy várakozási sorban tároljuk.
8. A 7. igénypont szerinti eljárás, **azzal jellemezve**, hogy a vevő és dekódoló berendezés (2020) több várakozási sort tartalmaz, ahol minden egyes várakozási sorhoz az egyes üzeneteknek a várakozási sorokból az alkalmazáshoz történő továbbítási sorrendjét jelző egy-egy prioritási szintet rendelünk.
9. A 8. igénypont szerinti eljárás, **azzal jellemezve**, hogy a jel az üzenetet átmenetileg tároló várakozási sor prioritási szintjét is tartalmazza.
10. Eljárás adatkommunikációra, egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, **azzal jellemezve**, hogy az alkalmazás és az összetevő között a 2, 4-9. igénypontok bármelyike szerinti eljárással egy kommunikációs csatornát létesítünk; és ezt követően az alkalmazástól az alkalmazás azonosítót és a logikai eszköz eszközazonosítóját, valamint az összetevőnek a vele társított logikai eszköz révén végrehajtandó működésére vonatkozó parancsot tartalmazó jelet bocsátunk ki az eszközzel.
11. A 10. igénypont szerinti eljárás, **azzal jellemezve**, hogy a jel a logikai eszköz által az összetevőnek adandó adat címét, valamint az összetevőtől a logikai eszközhöz kibocsátandó adat címét tartalmazza.
12. A 8. vagy 11. igénypont szerinti eljárás, **azzal jellemezve**, hogy a jel annak a várakozási sornak a prioritási szintjét is tartalmazza, amelyben a logikai eszköz által kibocsátott üzenetet tárolunk el átmenetileg az alkalmazáshoz történő továbbítását megelőzően.
13. A 2-12. igénypontok bármelyike szerinti eljárás, **azzal jellemezve**, hogy az alkalmazás és a logikai eszköz legalább egyikét a vevő és dekódoló berendezésnek (2020) a vevő és dekódoló berendezés (2020) egy összetevőjén keresztül adjuk be.
14. Az 1-13. igénypontok bármelyike szerinti eljárás, **azzal jellemezve**, hogy az összetevőként egy MPEG folyam tuner, egy soros interfész (4030), egy párhuzamos interfész (4032), egy modem (4034) vagy egy kártyaolvasó (4036) valamelyikét tartalmazza.
15. Eljárás kommunikációs csatorna létesítésére, egy program és egy számítógép rendszer egy összetevője között, vagy egy alkalmazás és egy vevő és dekódoló berendezés (2020) egy összetevője között, vagy egy eljárás adatkommunikációra egy alkalmazás és egy vevő és dekódoló berendezés (2020) egy összetevője között lényegében a leírás és a rajz alapján.

16. Berendezés kommunikációs csatorna létesítésére egy program és egy számítógép rendszer egy összetevője között, **azzal jellemezve**, hogy a számítógép rendszerben lévő összetevőkkel társított logikai eszközöket eltároló és minden egyes logikai eszköznek egy eszközazonosítót adó eszköze; a programhoz egy program azonosítót társító eszköze; és a programból egy, a program azonosítót és egy logikai eszköz eszközazonosítóját tartalmazó jelet a program és az egyes logikai eszközök között létrehozott eszközkezelőnek kibocsátó eszköze van, mely utóbbival a kapott programazonosítót és eszközazonosítót felhasználva és ezzel a program és a logikai eszköz közötti ezt követő kommunikációt lehetővé téve egy kommunikációs csatornát a program és a logikai eszköz között létesítő eszköze van.

17. A 16. igénypont szerinti berendezés, **azzal jellemezve**, hogy a program egy alkalmazásnak legalább egy részét alkotja, és a számítógép rendszer egy vevő és dekódoló berendezésnek (2020) legalább részét képezi.

18. Berendezés egy kommunikációs csatorna létesítésére, egy alkalmazás és egy vevő és dekódoló berendezés egy összetevője között, **azzal jellemezve**, hogy a vevő és dekódoló berendezés (2020) egyes összetevőivel társított logikai eszközöket eltároló és mindegyik logikai eszközhöz egy-egy eszközazonosítót hozzárendelő eszköze; az alkalmazáshoz egy alkalmazás azonosítót hozzárendelő eszköze; és az alkalmazástól az alkalmazás azonosítót valamint egy logikai eszköz eszközazonosítóját tartalmazó jelet egy, az alkalmazás valamint az egyes logikai eszközök között létrehozott közös interfészhez kibocsátó és azzal az alkalmazás és a logikai eszköz között a közös interfészen keresztül kommunikációs csatornát létesítő eszköze van.

19. A 18. igénypont szerinti berendezés, **azzal jellemezve**, hogy a közös interfész a kapott alkalmazás azonosító és eszközazonosító felhasználásával az alkalmazás és a logikai eszköz között kommunikációs csatornát létesítő eszközkezelőt tartalmaz.

20. A 17. vagy 19. igénypont szerinti berendezés, **azzal jellemezve**, hogy az eszközkezelő az alkalmazás azonosítót az alkalmazáshoz társító eszközkezelőként van kiképezve.

21. A 17., 19. vagy 20. igénypontok bármelyike szerinti berendezés, **azzal jellemezve**, hogy a logikai eszköz által az alkalmazáshoz történő továbbításra kibocsátott üzenetet átmenetileg tároló várakozási sort tartalmaz.

22. A 21. igénypont szerinti berendezés, **azzal jellemezve**, hogy a vevő és dekódoló berendezés (2020) több várakozási sort tartalmaz, ahol minden egyes várakozási sorhoz az egyes

üzeneteknek a várakozási sorokból az alkalmazáshoz történő továbbítási sorrendjét jelző egy-egy prioritási szint van hozzárendelve.

23. A 16-22. igénypontok bármelyike szerinti berendezés, *azzal jellemezve*, hogy az összetevőhöz a logikai eszköz által beadandó adatokat tároló eszköze, valamint a komponens által a logikai eszközhöz kibocsátott adatokat tároló eszköze van.

24. Vevő és dekódoló berendezés, *azzal jellemezve*, hogy a 16-23. igénypontok bármelyike szerinti berendezést tartalmaz.

25. A 24. igénypont szerinti vevő és dekódoló berendezés, *azzal jellemezve*, hogy egy tömörített MPEG jelet fogadó eszköze, a vett jelet dekódoló és ezzel egy televízió jelet előállító eszköze valamint a televízió jelet egy televízió készülékhez továbbító eszköze van.

26. Berendezés, vagy vevő és dekódoló berendezés, lényegében a leírás és a rajz alapján.

A meghatalmazott:

DANUBIA

Szabadalmi és Védjegy Iroda Kft.

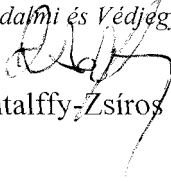

Antalfy Zsírós András

Fig. 1.

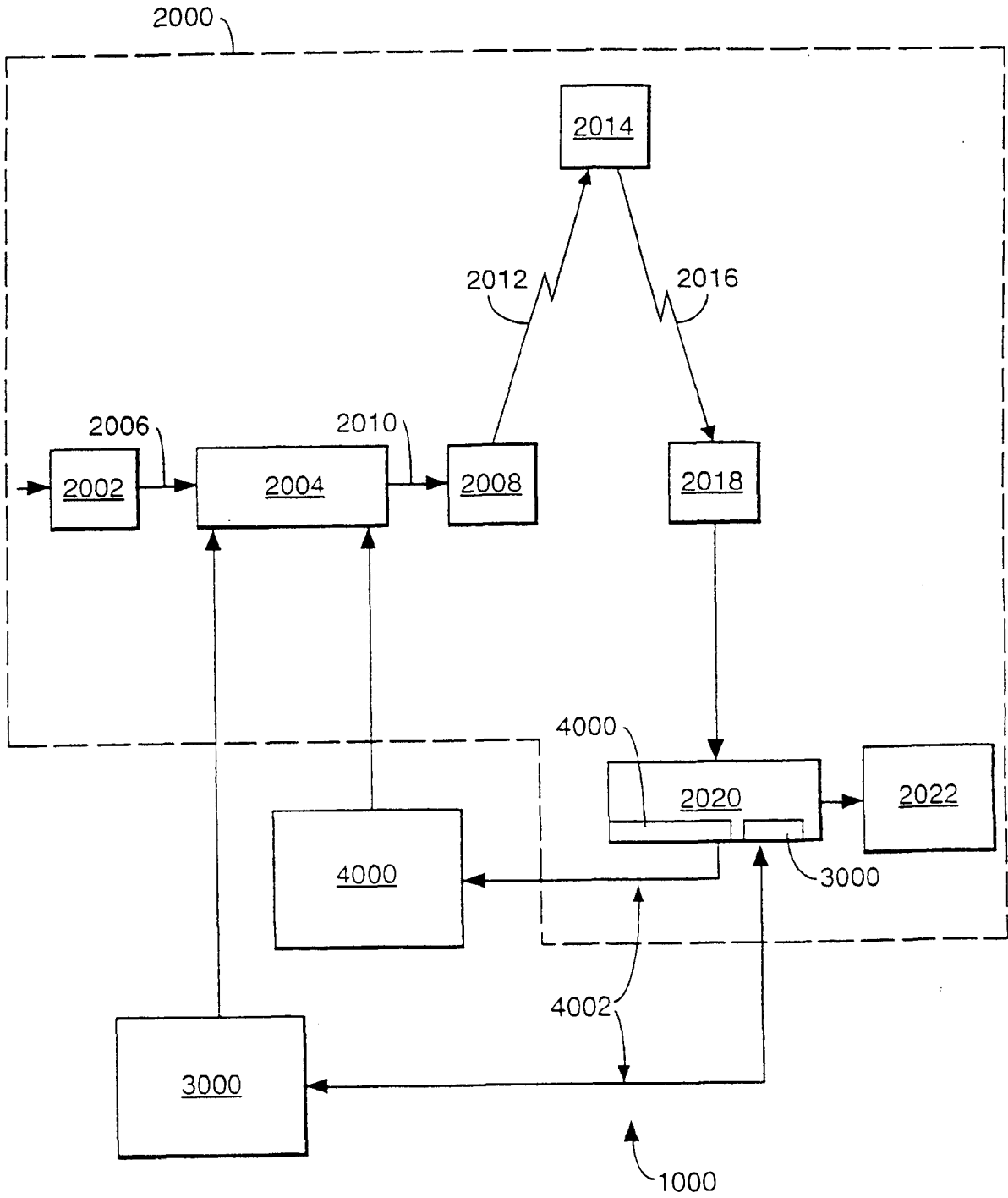
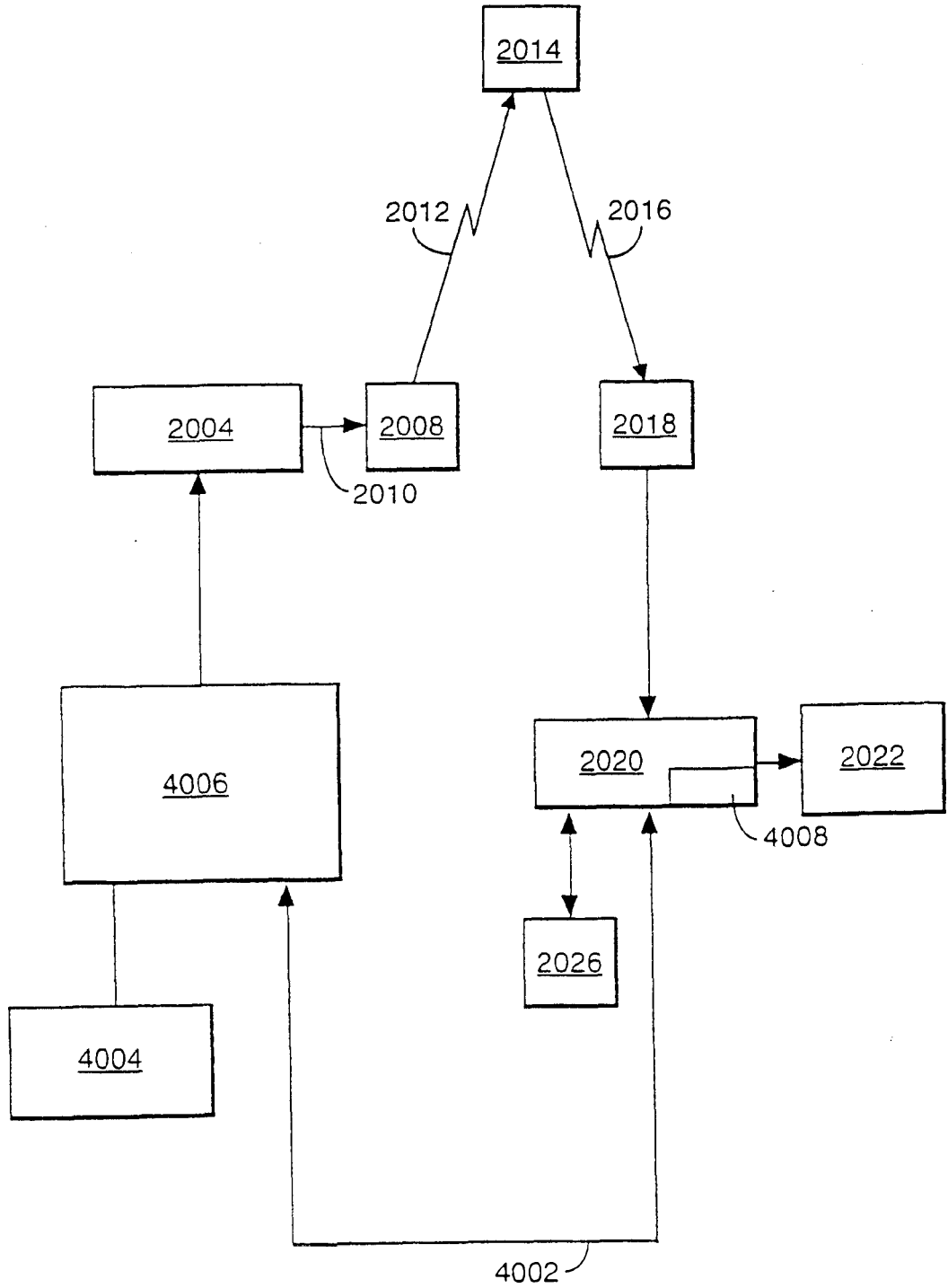


Fig.2.



1004141

Fig.3.

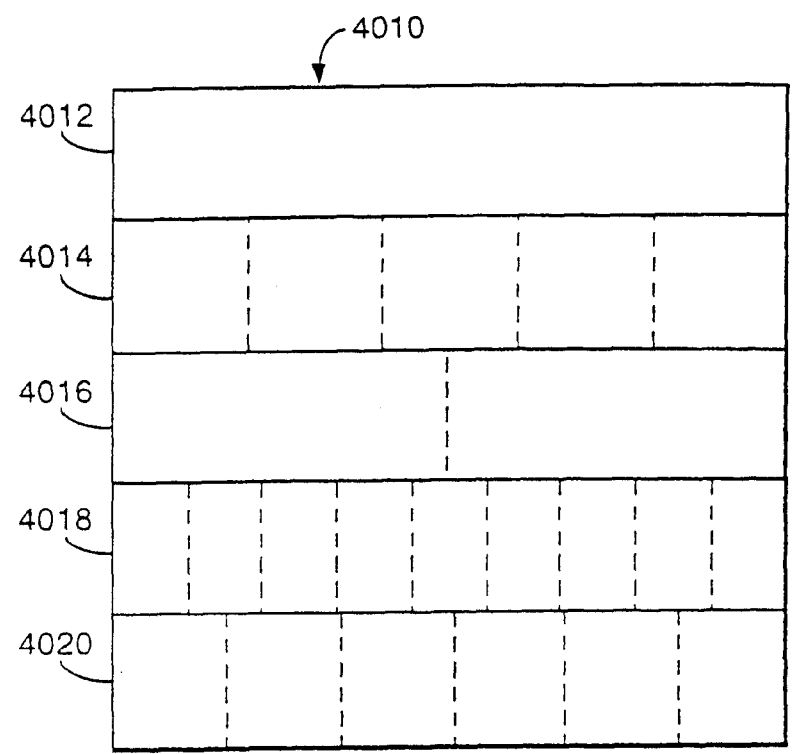
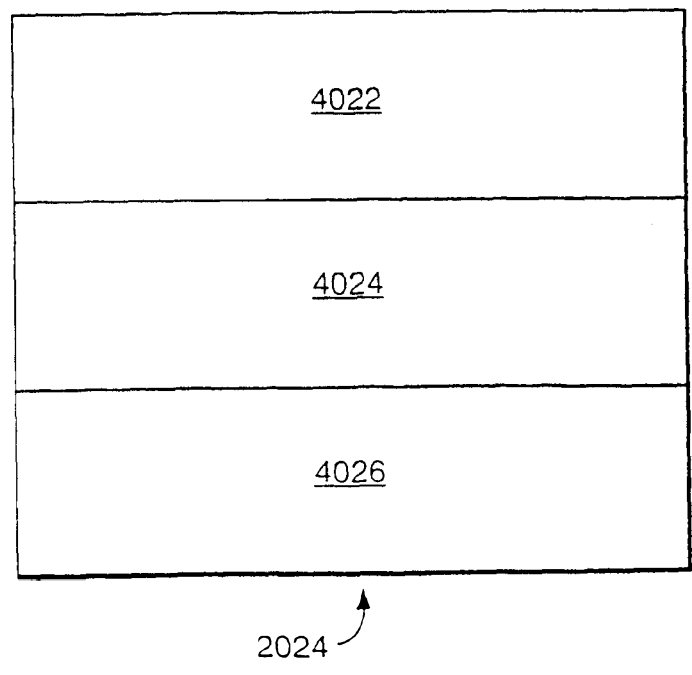


Fig.4.



P 0004149

1944

Fig.5.

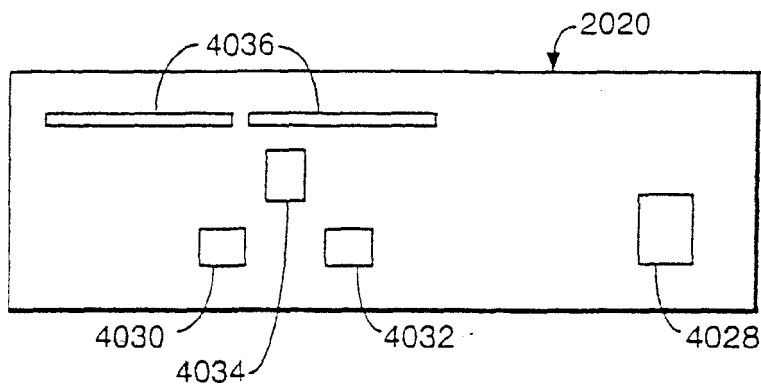
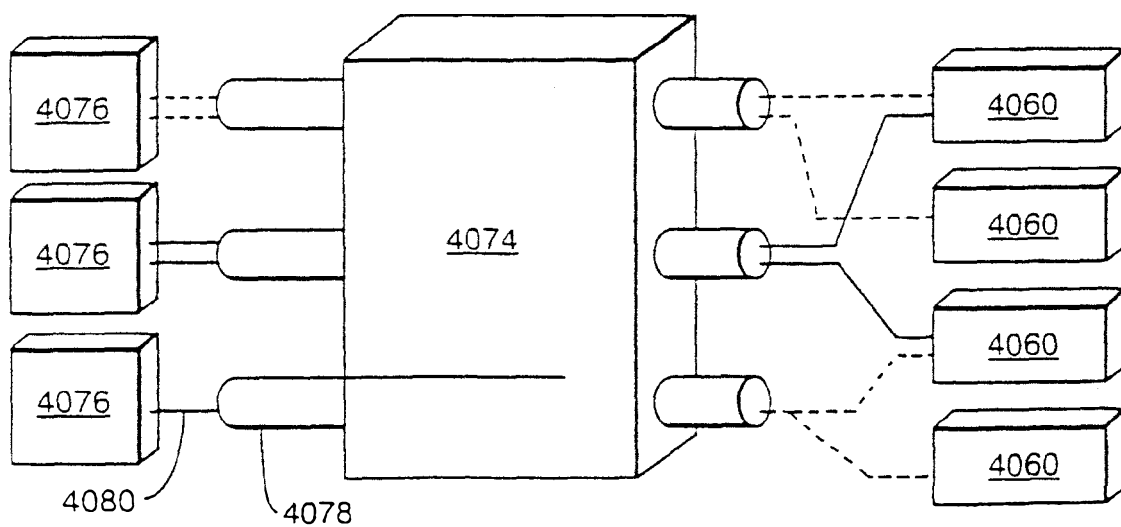


Fig.7.



10004141

Fig.6.

