

## Linux és RTAI Automaták építéséhez

Ez a könnyen telepíthető, web-alapú vezérlőrendszer szabványos telefon vezetékeket használ és bármely távirányítóval rendelkező egységet képes kezelni.

**C**ikkünkben egy olyan, az épületekben található különféle légkondicionáló berendezések központosított működtetését végző vezérlőrendszer fejlesztéséről és tervezéséről írunk, amelyet *Linux* illetve valós idejű *Linux* alkalmazás csatlófelület (*RTAI*) segítségével hoztunk létre. Az épületben elszórtan elhelyezkedő légkondicionálók saját infravörös távirányítóval rendelkeznek. Célunk az volt, hogy a légkondicionálók vezérlését központosított, számítógéppel vezérelt megoldással váltsuk fel, amely képes ki- és bekapcsolni a készülékeket, valamint a kívánt hőmérsékletet és fordulatszámot beállítani. A projekt ötlete akkor merült fel, amikor a helyi egyetemnek, szerény keretből megvalósítható, központosított és rugalmas a légkondicionáló vezérlési megoldásra volt szüksége. Hasonló célokra léteznek üzleti alkatrészek és programok, de általában túlságosan költségesek és gyártófüggők.

Projektünk alkatrész megvalósítása egy *Linux*ot futtató központi vezérlő-számítógépből és a hozzá csatlakoztatott *RS-485*-ös mikrovezérlő hálózattól áll. A mikrovezérlők a közeli felszereléseket irányító infravörös jelekkel parancsokat tudnak küldeni a megfelelő légkondicionálóknak. A rendszer programterve két valós idejű feladatot (a fő vezérlő feladatot és a *RS-485* hálózat vezérlő feladatot) valamint további, nem valós idejű feladatokat (a webkiszolgálót és az adatbázist) tartalmaz. A felhasználói felületért a webkiszolgáló felel így az egyetem számítógép-hálózatán bármely böngészőről elérhető. Adataink tárolására a *PostgreSQL* adatbázist választottuk. Megvalósításunk olyan alacsony költségű megoldás, amely igény szerint rugalmasan bővíthető. Ezen kívül gyártófüggetlen és bármely légkondicionálóval működik, amely képes infravörös távirányítót kezelni. Valamennyi légkondicionáló önállóan működik, saját hőmérséklet-vezérlő rendszerével. Az eszközök működésének ellenőrzésére a hálózat valamennyi mikrovezérlőjét hőérzékelővel láttuk el amely regisztrálja az aktuális teremhőmérsékletet és jelenti a központi számítógépnek.

### Felhasználói felület

A teljes felhasználói felület weblap alapú. A kezdőlapra az összes légkondicionáló állapotának összefoglalóját soroljuk fel. A megjelenített adatok között találjuk a légkondicionálót azonosító szöveget, az épületben elfoglalt helyét, a jelenlegi teremhőmérsékletet valamint, hogy van-e az eszköznek



1. ábra A vezérlőfelületről a hitelesített felhasználó beindíthatja a légkondicionálót, új hőmérsékletet állíthat be vagy kikapcsolhatja azt

valamilyen előreprogramozott működési módszere, amely alapján éppen dolgozik. Erről a lapról valamennyi légkondicionáló külön kezelőfelületére mutat egy-egy hivatkozás. Mielőtt erre a lapra juthatnánk, a rendszer felhasználónevet és jelszót kér. Ezen a szinten lehetőségünk nyílik utasításokat adni a rendszernek, közvetlenül vezérelni a légkondicionálót illetve megváltoztatni vagy létrehozni az automatikus működés valamilyen programját (1. ábra).

A rendszer legérdekesebb része a programozott működés. Például a rendszer képes az órák kezdete előtt minden nap automatikusan bekapcsolni a légkondicionálókat, minden szobában előre megadott hőmérsékletértékekkel. Aztán este, amikor már nincs élet az épületben vagy az adott teremben, a rendszer lekapcsolja a légkondicionálókat.

### Alkatrész felépítés

Az alkatrészekészlet a központi számítógépből és a légkondicionáló gépeket irányító mikrovezérlőkből áll. Valamennyi mikrovezérlőt *RS-485* kéteres hálózatra csatlakoztattuk. Megvalósításunkban az *Intel* származék *ATMEL AT89C2051*-es jelzésű vezérlőt alkalmaztuk. Hús lábú DIP csomagban kapható, 128 byte adat memóriával és 2KB ROM kód helytel, egy aszinkron soros kapuval és 14 független digitális I/O kapuval rendelkezik.

A mikrovezérlő lapját és elemeit a 2. ábrán mutatjuk be. Az infravörös jeleket a mikrovezérlőben található program hozza létre az egyik digitális kimeneti kapu segítségével. A soros kapu kapcsolja a mikrovezérlőt az RS-485 hálózathoz. Valamennyi mikrovezérlő lapot felszereltük a Dallas Semiconductor DS1620 típusú hőmérőjével. A mikrovezérlő 3 eres, digitális szinkron soros felületen kommunikál a hőérzékelővel. A mikrovezérlőben nincs alkatrész szintű támogatás a szinkron soros felület kezeléséhez, ezért a vezérlést programozottan oldottuk meg a normál I/O kapukon keresztül.

Alkalmazásunkban azért esett az RS-485 hálózatra a választás, mert könnyen telepíthető, olcsón megvalósítható és könnyen összekapcsolható vele hasznos mennyiségű csomópont. A csomópontokat mindössze egyetlen 3-as kategóriájú telefon osztályú kábel köti össze. Az alkatrész meghajtó korlátai miatt a lehetséges csomópontok száma 32, de ezt a számot könnyedén megnövelhetjük hálózati ismétlők segítségével. A vezérlő-számítógép és az első ismétlő, vagy két ismétlő közötti maximális kábelhossz 1200 méter. A fizikai kábel elérést mester-szolga elven vezéreljük. A Linuxot futtató számítógép a mester, amely előre meghatározott sűrűséggel kérdezi le a csomópontokat. A mester minden lekérdezőparancsot küld a csomópontnak; a lekérdezett csomópont adattal válaszol a mesternek avagy üres csomagot küld, jelezve, hogy a csomópont élő. Az ilyesfajta elérési megoldás hátrányára akkor derül fény, amikor a mester áll le. Ilyenkor ugyanis a teljes hálózat is csődöt mond.

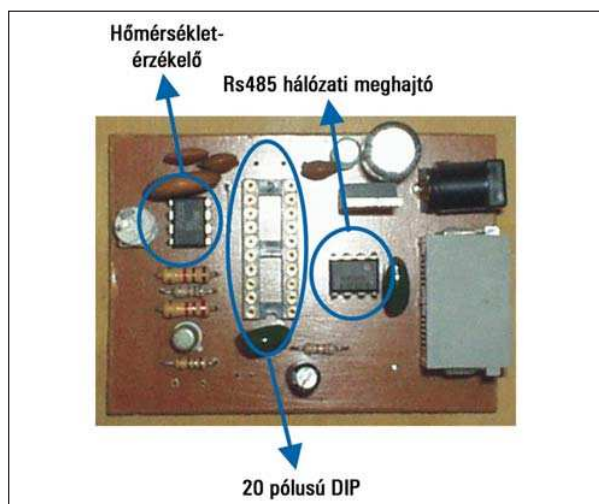
A mikrovezérlők korlátozott számát figyelembe véve a 9. bit protokoll segítségével állapíthatjuk meg, hogy a hálózaton átküldött csomag az adott vezérlőhöz tartozik-e. A hálózaton átküldött minden egyes bájtához tartozik egy 9. bit is. Ez a kiegészítő bit kizárólag a csomag cél cím esetében lesz magas rétekkü. A mikrovezérlő UART (universal asynchronous receiver and transmitter) alapértelmezés szerint csak akkor vált ki megszakítást, ha a fogadott érték 9. bitje magas. A megszakítás szolgáltatás rutinja ezután a fogadott bájtot összehasonlítja a csomópont címével. Amennyiben megegyeznek, a megszakítás átprogramozza az UART-ot, hogy a 9. bit állapotától függetlenül, valamennyi bájtot fogadjon a csomag végéig. Amennyiben a cél címe nem egyezik meg a csomópont címével a megszakítás rutin visszatér.

A központi számítógép UART-ja, lévén PC alkatrész, nem támogatja közvetlenül a 9. bit protokollt. E hátrányt a meghajtó a paritás bit segítségével szimulálva próbálja meg kiküszöbölni. A bájt átvitele előtt a meghajtó úgy állítja be a paritást, hogy a cím bájtokban esetében egy legyen a többi bájt esetében pedig nulla.

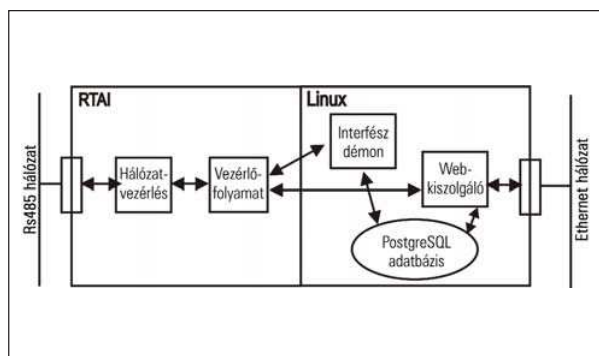
A 3. ábra a rendszer feladatait és a köztük élő kommunikációs vonalakat mutatja be.

### Valós idejű feladatok

A fő vezérlő feladat, a hálózati elérést vezérlő feladat és az RS-485 hálózat fizikai rétegének program meghajtója valós időben futó feladatok. Az RTAI-hoz kifejlesztettünk egy RS-485 meghajtóját. A meghajtó az alkalmazásban felhasznált és korábban bemutatott 9. bit protokoll kivételével hasonló a többi soros meghajtóhoz.



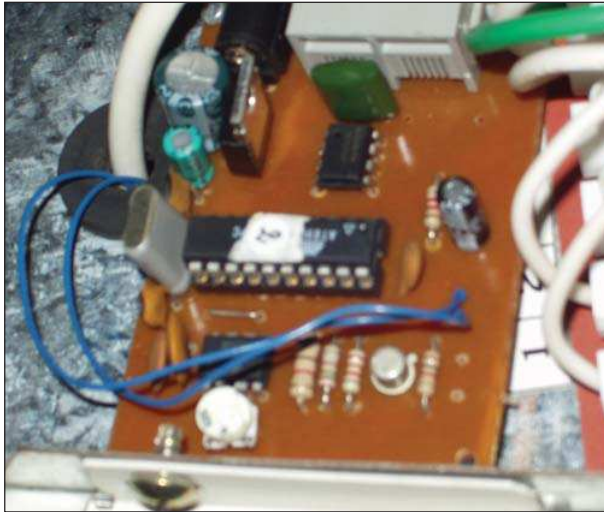
2. ábra A mikrovezérlő lapja a mikrovezérlő eltávolítása után



3. ábra A rendszerfeladatok közötti kommunikáció

A másik valós idejű feladat a hálózati elérést vezérlő feladat, mely meghatározott időnként csomagokat küld az egyes hálózati csomópontoknak. A csomag lehet az infravörös jelet bekapcsoló parancs vagy lekérdezés amivel kideríthetjük, hogy a csomópont aktív-e, valamint az aktuális teremhőmérsékletet lekérdező parancs. Az első két csomagtípus esetében a csomópont visszaigazolás, az utolsó esetében a teremhőmérséklettel válaszol. A fő vezérlő folyamat valamennyi csomópont aktuális állapotát ismeri, és értesíti a felhasználói felületet ha valamelyik csomópont meghibásodott.

A fővezérlő feladat, az adatbázisban tárolt, előre programozott információk alapján működteti az épület légkondicionáló berendezéseit. A feladat képes parancsokat fogadni a felhasználói felületről, amelyek két RT-FIFO segítségével felülbírálnak az előreprogramozott beállításokat. Az RT-FIFO folyamatközi kommunikációs rutinok feladata a valós idejű és normál Linuxos feladatok párbeszédének lehetővé tétele. A PostgreSQL adatbázissal való kommunikációhoz Linux demont kellett kifejleszteni. A démon két másik RT-FIFO csatornán keresztül beszélget a fővezérlő folyamattal. A démon másik fontos feladata a rendszer időt és dátumot elküldeni a fővezérlőnek, ugyanis ennek kiolvasására nem létezik megoldás RTAI alatt.



4. ábra A teljes mikrovezérlő lap a telepítés után

A kifejlesztett rendszer parancsokat küld a légkondicionálókknak, így nincs szükség helyi távirányítókra. A légkondicionálók hőszabályzó rendszerébe nem avatkoztunk bele és nem változtattuk meg a belső kapcsolási sémákat sem. Minden légkondicionáló saját beépített hőszabályzó rendszerét használja, a mikrovezérlőkbe épített hőérzékelők pedig ellenőrzik, hogy az eszköz helyesen működik-e. A 4. ábra a telepített mikrovezérlőt mutatja be.

### Linux feladatok

A *Linux* feladatok célja a felhasználó felület megjelenítése a fő adatforrásként használt *PostgreSQL* adatbázis futtatásával a webkiszolgálón keresztül. Mint korábban leírtuk, a másik *Linux* oldali feladat a démon, amelyet a *RTAI* fővezérlő feladat használ a rendszer idő/dátum és az adatbázis eléréséhez.

A felhasználói felület egyszerű. Az első lapon az egyes légkondicionálók aktuális állapotát láthatjuk. Ezt a lapot minden felhasználó elérheti. A program megváltoztatásához vagy egy adott légkondicionáló utasításához a rendszer felhasználói nevet és jelszót kér. Az adatbázisból lekért információ dinamikusan megjelenítéséhez a *PHP* nyelvet használtuk. A rendszer a *PostgreSQL* adatbázisban tárolja az általános információkat (*BTU*, elhelyezés, márka, mikrovezérlő hálózati azonosító) programozott műveleteket és az egyes kondicionálók működtetéséhez használt infra parancsokat.

### IRC parancsfelület

A rendszer fontos eleme az a modul, amely a légkondicionáló távirányítójának jeleit olvassa be és a megfelelő eszközhöz rendelve az adatbázisban tárolja az kapott azokat, hogy később a hálózatra kötött mikrovezérlőkkel a jeleket újra létre lehessen hozni. Ezt a modult csak olyankor használjuk amikor egy új márkájú és/vagy vagy eltérő parancskészletű légkondicionálót veszünk fel a rendszerbe.

A modul két részből áll: az első az infra jeleket beolvasó valós idejű feladat. A hálózati forrásokban felsorolt *LIRC Projekt* valamint a *Ripoll and Acosta* tanulmány részletes információkat tartalmaz az infravörös távirányítókról. Ugyanitt

normál *Linux* és *RT-Linux* rendszerekhez készült példaalkalmazásokat is találunk. A modul másik feladata a *Linuxon* futó felhasználói felület. A két feladat *RT-FIFO* kapcsolaton keresztül kommunikál.

A mikrovezérlőkben használható kis mennyiségű RAM és a hosszú infravörös jelhossz miatt, a program fontos funkciója, hogy segít a felhasználónak megszerezni a különféle infra távirányítók gombokhoz vagy gombkombinációkhoz rendelt ismétlődő parancsokat. Ezeket a mintákat aztán a mikrovezérlő belső programjába (*firmware*) beégetve újra elő tudjuk állítani az eszközt vezérlő jelet. Például amennyiben tíz különféle minta létezik, a megfelelő mikrovezérlőhöz a hálózaton keresztül küldött parancs a következő lesz: ismételd meg az egyes számú jelet tízszer, aztán a kettest háromszor és így tovább, míg a teljes parancsot meg nem kapjuk. A technika előnye hogy kevesebb erőforrást használ a jel ismételt előállításához. Hátránya, hogy valahányszor új légkondicionálót vásárolunk, a mikrovezérlő programját meg kell változtatni, hogy az új eszköz parancskészletét kezelje.

### Költségek

Jelen pillanatban kilenc légkondicionálót kezelünk a leírt rendszerrel; valamennyi ugyanabban az épületben található. Hamarosan további tizenötöt veszünk fel. Az egyes mikrovezérlők alkatrész-költsége 60 dollár, a központi vezérlő számítógép költsége 500 dollár. További költségek az *RS-485* hálózat kiépítése és természetesen a rendszer kifejlesztésének és megvalósításának költsége.

### Összefoglalás

A megvalósított rendszer megfelel az aktuális felhasználó igényeinek. A projekt sikere miatt az egyetem által vásárolt valamennyi további légkondicionálónak együtt kell működnie a rendszerrel. Ennek egyetlen követelménye, hogy valamennyi új légkondicionálónak rendelkeznie kell infravörös távirányítóval.

Az *RTAI*-nak hála a rendszer fővezérlő feladata független a *Linux* alatt futó felhasználói felület folyamataitól. Még abban a valószínűtlen esetben is, ha a *Linux* leállna, a rendszer tovább folytatná az előre beprogramozott műveleteket. A jövőben a rendszer könnyedén kiterjeszhető az épület világításának, riasztóinak, korlátozott elérhetőségű területeinek és egyéb rendszereinek vezérlésére.

*Linux Journal* 2004. november, 127. szám



**Andres Benitez** (adorego@conacyt.org.py) Villamosmérnöki diplomáján dolgozik az Asuncioni Katolikus Egyetemen, Paraguaiban. A cikkben bemutatott megoldás e diplomamunka végső feladata.



**Vicente Gonzalez** (vgonzale@uca.edu.py) építőmérnök. Automatizálásból szerzett MSc diplomát a Madridi Polytechnic Egyetemen. Jelenleg az Asuncioni Katolikus University Egyetem elektronikai és számítástudományi tanszékén dolgozik tanársegédként.