

R15-R40 kapcsolat megteremtése CTCA-n keresztül,

avagy

nosztalgikus visszaemlékezés egy projectre

1. Nemzetközi háttér

A 80-as évek elején a Nyugat részéről még keményen működött az embargó politika. Ami ebből leginkább érintett minket, a gép-gép kapcsolatot megteremtő hardware és software eszközöktől teljes mértékben el voltunk vágva. Például, a nagyszámítógép gyártó cégek úgy küldték az operációs rendszereket a megrendelt gépekhez, hogy nem tartalmazták az intelligens kapcsolat kialakításához szükséges eljárásokat. Ezt a hátteret ismerni kell ahhoz, hogy értékelni tudjuk a project jelentőségét.

2. A project lényege

A HRL kapta meg azt a feladatot, hogy kifejlesszen egy olyan hardware eszközt, amelynek funkciója két számítógép összekapcsolása, valamint a kapcsolat kiépítése után a gépek közötti adatátvitel biztosítása, mindkét irányban. Természetesen maga után vonva az érintett operációsrendszerek továbbfejlesztését is, olyan mértékben, hogy az új eszköz software támogatása is rendelkezésre álljon.

3. Mi az a CTCA

A CTCA, másképp: Channel-To-Channel Adapter, volt annak a kifejlesztett hardware eszköznek a neve, amely specifikációja szerint, alkalmas volt arra, hogy két gép között, akár nagymennyiségű adat átvitelét is végrehajtsa. Ezt manapság egy FTP kapcsolatnak is nevezhetnénk, de akkor még ez a protokoll nem is létezett, illetve nem volt róla tudomásunk.

Működése három fázisra bontható:

- a) inicializáló szakasz: érzékeli és fogadóképesé teszi azt az oldalt ahonnan felszólítást kapott, valamint a másik oldal értesítése, hogy aktivitás történt,
- b) adó-vevő szakasz: az adóbuffer továbbítása a vevő oldalnak,
- c) lezáró szakasz: a kapcsolat leépítésének végrehajtása.

4. Hol volt a helyem ebben a fejlesztésben

Mivel a HRL-ben software fejlesztőként dolgoztam, a munkába akkor kapcsolódtam be, amikor már a CTCA specifikációja elkészült. Feladatomban a következő volt: az R15 oldali operációs rendszerbe létrehozni az új berendezés teljes támogatását, beleértve az ASSEMBLER programnyelvi support kialakítását is. Az ASSEMBLER support első alkalmazása maga a tesztprogram létrehozása volt. Tehát, felülről lefelé, az alkalmazói szinttől a hardware-ig, egyszerre tesztelődött, kicsit nagyképűen, az egész project. (Önbizalomban sohasem volt hiány!) A munkában kollégáim segítségét, támogatását is élveztem. Ezt most is köszönöm nekik. A feladat tehát a következő volt:

- DOS/VS operációs rendszer kibővítése egy új I/O egységgel,
- az új ASSEMBLER makrók megírása, illetve
- a meglévő makrók módosítása a CTCA igényeinek megfelelően, és
- egy olyan tesztprogram megírása, amely jelzéseivel a hardware fejlesztőket is segíti, valamint a főnököket is meggyőzi arról, hogy az eszköz hibátlanul működik.

Elvileg a feladatot „megkönnyítette” az, hogy az akkor üzemelő operációs rendszer, beleértve az ASSEMBLER-t is, forráskódban elérhető volt (LEGÁLISAN!). Azért írom, hogy elvileg, mert ellenkező esetben a feladat nem lett volna végrehajtható, legalább is belátható időn belül nem. Másrészt a feladat olyan nehéznek, bonyolultnak tűnt, hogy bénítóan hatott rám.

5. A fejlesztés nehézségei

A legfőbb nehézséget az okozta, hogy a csatornára csatlakozott adapter I/O műveletei adott időn belül elinduljanak és befejeződjenek. Az IBM 370-es architektúrában, így az R15-re is vonatkozóan is, a szelektor csatornának szabályai vannak, amit egy I/O berendezésnek ismernie kell és be kell tartania. Az időkorlátok átlépése „unit check” státuszt eredményez, amit hibaként értelmezhetünk. A szelektor csatorna működési elvét nem szeretném most kifejteni, mert túl messzire vezetne, de legfőképp sok időbe kerülne. A csatorna követelmények kielégítése miatt kellett az alábbi szabályok szerint működnie az adapternek:

- az „adó” oldal kezdi a kommunikációt,
- a „vevő” oldal értelmezi a hívást és kiadja a „read” parancsot,
- az „adó” értesítést kap az aktív „read”-ről, és azonnal kiadja a „write” parancsot,
- az „adó” és „vevő” oldal is megkapja a „channel end + device end” státuszt, amely a művelet hibátlan befejezését jelzi.

Nagyon röviden ez a működési elve a CTCA-nak. Felmerül a kérdés, hogy miért a „read”-hez kell a „write” parancsot szinkronizálni, és miért nem fordítva? Nos, itt jön be az időzítés problematikája. A kísérletek azt igazolták, hogy egyéb feltételek kielégítését is figyelembe véve, ez a legbiztonságosabb, legjobb megoldás.

Visszatérve a béműveltséghez, a fejlesztés akkor indult be igazán, amikor kiderült, hogy az adapter, tulajdonképpen az operátori konzol és egy mágnesszalag egység valamilyen összegyűrt egysége. Hogy mennyiben igen és mennyiben nem, ebbe most ne menjünk bele, de az igaz, hogy megvezetett abban, hogy mely rutinok, eljárások tanulmányozása segít a feladat megoldásához. Az 1. sz. melléklet részletesen bemutatja az adó-vevő rutin működési elvét. A 2. sz. melléklet az inicializáló szakasz működését vázolja fel. A 3. sz. melléklet azt illusztrálja, hogy az operációs rendszerben történő módosításoknak egy része ilyen pár soros változtatás volt, ami jelentéktelennek tűnik, de komoly kutatás előzte meg, hogy ott és éppen azt kell arra helyre beilleszteni. Szóval nem volt egyszerű feladat.

6. Végeredmény

Végül következzen a konklúzió. Az átadás napjára minden összeállt és teljes sikerrel átadtuk a berendezést. Egy mágnesszalag teljes állományát beolvastuk és felírtuk a másik oldalon lévő üres szalagra. Az input és output mágnesszalag teljes tartalmi azonosságot mutatott. Ha most hozzátehetem saját véleményemet, azóta is büszke vagyok arra, hogy ebben a fejlesztésben részt vehettem, sőt arra is büszke vagyok, hogy egy olyan csapattal dolgozhattam együtt, akiket én a hazai számítástechnika színe-javának tartok, azóta is.

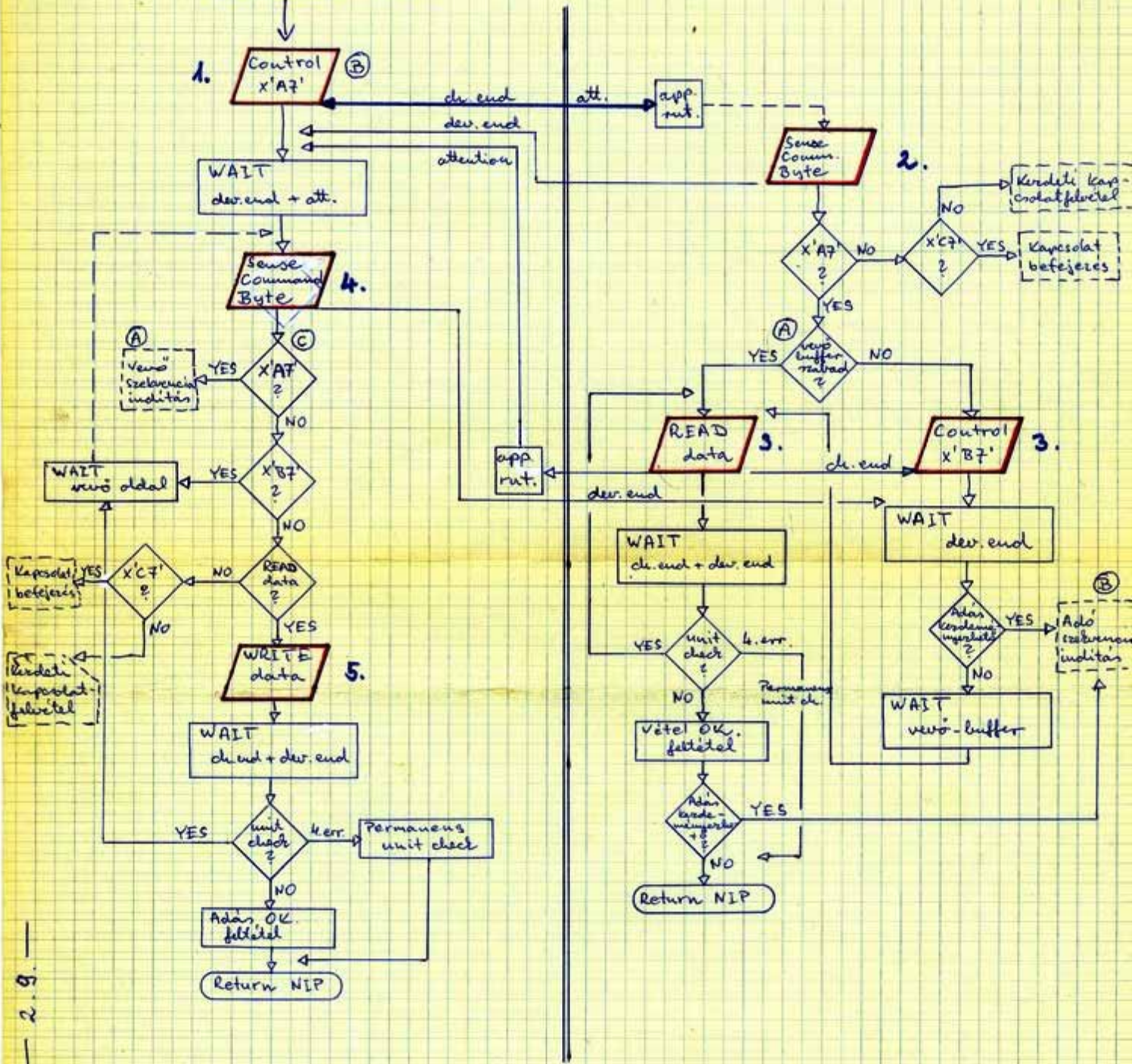
Köszönöm, hogy lehetőséget kaptam, hogy mindezeket elmondhassam.

Vác, 2010, október

Pellei Jenő, HRL

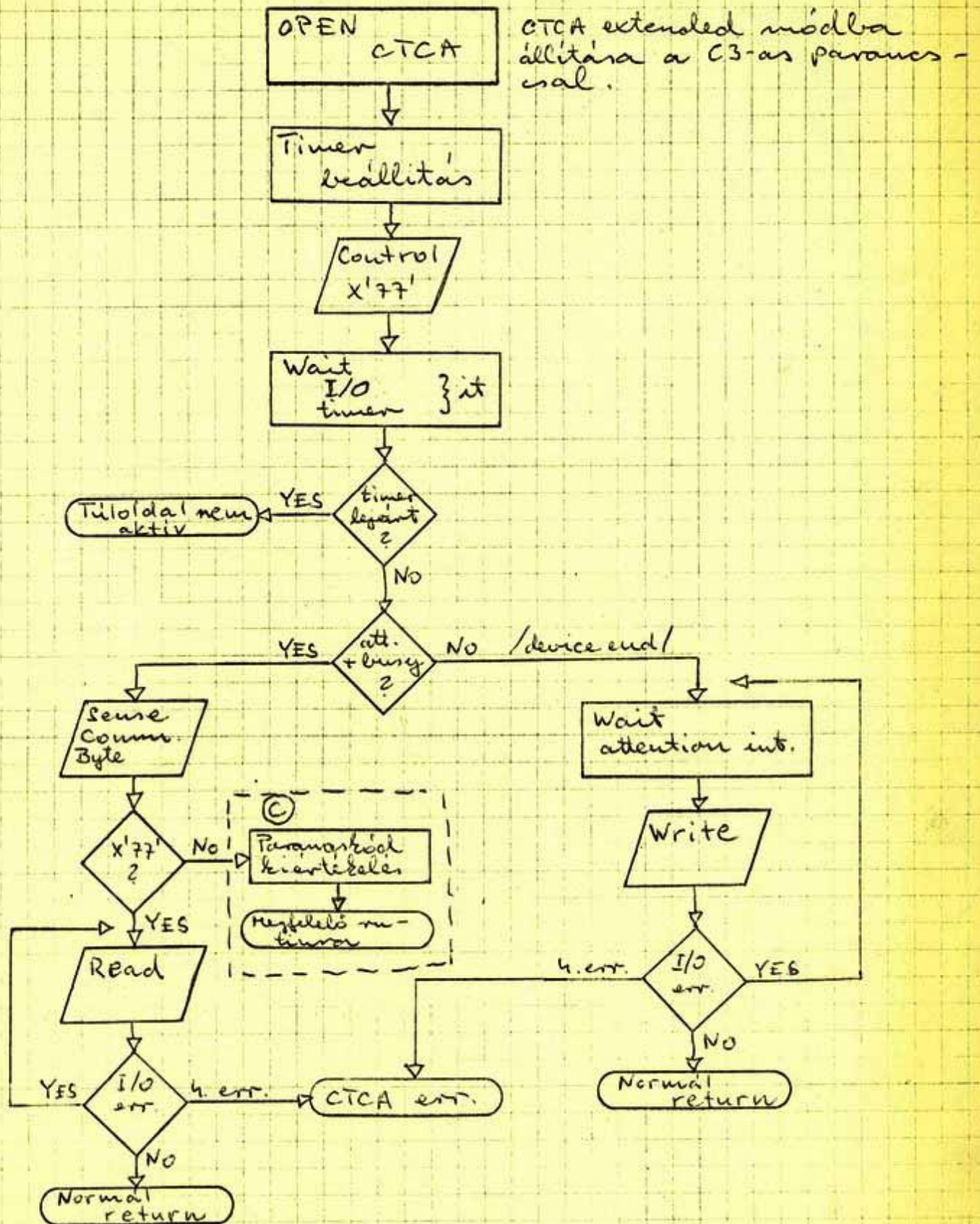
1. sz. melléklet

Adásra felhívás



2. sz. melléklet

Szimmetrikus modul, az att+busy státusz ha-
tárolóra meg az egyik ill. másik oldalalt.



3. sz. melléklet

- Attention + busy status továbbítása a felhasználónak

