

A BepiColombo űrszonda adatgyűjtő és vezérlő rendszere

SULYÁN JÁNOS, SZALAI LAJOS

KFKI Részecske és Magfizikai Kutatóintézet, {sulyan, szlajos}@rmki.kfki.hu

LIPUSZ CSABA, SZALAI SÁNDOR

SGF Kft., lipusz.csaba@freemail.hu, szalai@sgf.hu

Kulcsszavak: űrszonda, adatgyűjtő, beágyazott processzor, buszrendszer, SpaceWire

A BepiColombo űrszondában való magyar részvétel kapcsán röviden ismertetjük a Merkúrhoz induló programot. Ezen a szondán kerül alkalmazásra egy új adatgyűjtő és vezérlő buszrendszer; a SpaceWire. Ennek az űreszközök fedélzeti adatátvitelére kidolgozott szabványnak az előnyeit ismerteti és rövid összefoglalóját tartalmazza a cikk.

1. Bevezetés

A BepiColombo az Európai Űrügynökség (ESA) első űrprogramja a Merkúr bolygó kutatására, a programot dr. Giuseppe (Bepi) Colombo olasz asztrofizikusról nevezték el, aki a gravitációs hintamanőver elvét dolgozta ki, azaz az űrszondák gyorsítását a bolygók melletti elhaladás révén. A program keretében két űrszonda készül: Mercury Planetary Orbiter (MPO) és Mercury Magnetospheric Orbiter (MMO). Az MMO alapvetően a japán űrkutatók fejlesztése, míg az MPO az európai kutatók együttműködésében készül [1,2].

A KFKI Részecske és Magfizikai Kutatóintézet valamint az SGF Kft. a SERENA (Search for Exospheric Refilling and Emitted Neutral Abundances) rendszer PICAM (Planetary Ion Camera) érzékelőjének a fejlesztésében vesz részt. A SERENA rendszer négy érzékelőt tartalmaz, melyek a Merkúr exoszférájának folyamatait fogják vizsgálni. Az SGF Kft. feladata a PICAM műszer földi ellenőrző berendezésének fejlesztése, melynek az űrszonda parancsait illetve a PICAM mérési adatait kell fogadni és megjeleníteni.

A bolygóközi űrszondák a különböző tudományos kísérletek műszerei számára a biztosítják a megfelelő működtetési feltételeket, azaz energiaellátást, viszonylag elviselhető működési hőmérsékleti ($-60^{\circ}\text{C} \div +100^{\circ}\text{C}$) tartományt, vezérelhetőségét (földi parancsok továbbítását), valamint a mérési adatok begyűjtését és a Földre való továbbítását. Az utóbbi két feladat szokásos vezérlő és adatgyűjtő rendszernek felel meg. A bolygóközi űrszondák egyik legszigorúbb követelménye a súlyminimalizálása, ezért a különböző szondákon a súlyminimalizálás elsődleges céljából – és figyelembe véve a Földdel való rádiókapcsolat átbocsátó képességét –, különböző egyedi adatgyűjtő rendszerek kerültek alkalmazásra. Ez ellentmond az egységes szabvány alkalmazásának, amely kizárja a tudományos műszerek újrafelhasználhatóságát. A múltban is történtek törekvések szabványok alkalmazására, így például a NASA-nál több szondán is alkalmazta a Mil-1553-as repülőipari busz szabványt. Az ESA a jövő szondáin alkalmazásra kerülő buszrendszert, a SpaceWire-t választotta a BepiColombo szondára.

2. A BepiColombo űrszonda

Az MMO és az MPO orbiterek egy egységként startolnak 2013-ban, majd hatévi utazás és több hintamanőver után 2019-ben válnak szét és állnak pályára a Merkúr körül. Az út során a szonda ionhajtóművet fog használni és egyszer a Hold, kétszer pedig a Vénusz mellett is elhalad. A szonda csak az út legvégén válik szét, ekkor elhagyják az ionhajtóművet, majd a két egység különböző pályára áll a bolygó körül.

Az MMO-nek elnyúlt poláris pályát választottak, ami a Merkúr mágneses terének feltérképezéséhez és dinamikájának megfigyeléséhez alkalmas, az MPO viszont alacsonyabb körpályára áll, mivel ennek feladata a felszín és a magas légkör vizsgálata. Az MMO a SERENA rendszeren kívül kamerákat, infravörös-, ultrabolya- és röntgen-spektrométereket, lézeres magasságmérőt, rádiókísérletet és gyorsulásmérőt visz a fedélzetén. A műszerek évente több mint másfél terabit mérési adatot gyűjtenek, amit 50 kbit/s-os adatátviteli sebességgel sugároznak majd a Föld felé.

Az adatgyűjtő és a kommunikációs egységekről általában nem sok szó esik az űreszközökről szóló hírekben, sőt a szondáról szóló publikációkban sem, mégis szinte a legfontosabb eszközök, hisz ezek hibája a teljes küldetést tönkretelheti. Az adatgyűjtő rendszer feladata nemcsak a kísérletek adatainak összegyűjtése, tárolása és továbbítása, hanem az egész rendszer működésének összehangolása, ellenőrzése. Mivel minden szonda egyedi, speciális feladatra szánt adatgyűjtő rendszert kell fejleszteni hozzá. Ennek a rendszernek minden kísérlettel kapcsolatot kell tartania, így nagyon fontos, hogy milyen interfészen keresztül kommunikálnak. A gyakran különböző fejlesztések költségesebbé és nehezebben kezelhetővé tették a nagyméretű, általában különböző nemzetek által fejlesztett modulokból álló űrrendszereket.

A szondán három típusú adatgyűjtő és vezérlő rendszer lesz alkalmazva: SpaceWire interfész a viszonylag nagy adatforgalomra, a Mil-1553 zajérzékenyen csak a rádiós kísérlet vezérlésére, harmadikként pedig a csak monitorozást szolgáló rendszer, amely a műszerek álla-

potát jellemző hőmérsékleti adatokat lassú mintavételezéssel gyűjti. A továbbiakban a SpaceWire alkalmazását részletezzük.

3. A SpaceWire, mint megoldás a problémákra

A SpaceWire az ESA által kifejlesztett kommunikációs szabvány, melyet űreszközök fedélzetén kívánnak használni [3]. Ezért az ESA létrehozta az ECSS-E-50-12A szabványt, amely a korábbi lassú (1 Mbit/s), nagyobb fogyasztású, korlátozottabban használható (maximum 32 végpont) Mil-1553-as szabványt hivatott felváltani [4].

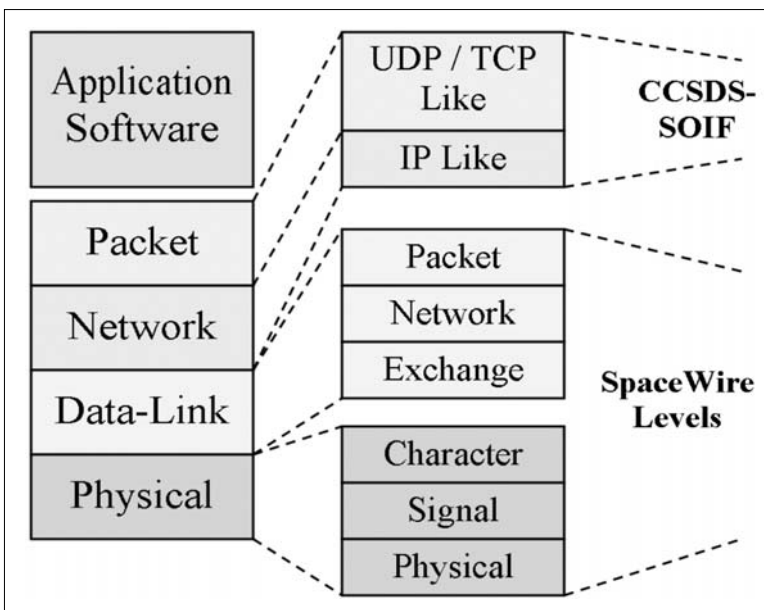
A SpaceWire-nek sokféle feladatra kell megfelelnie. Egyre fontosabb szempont a nagy adatátviteli sebesség és a kis energiateljesítmény, hiszen napjainkban egyre messzebb küldenek űrszondákat, melyek kamerái igen nagy adatmennyiséget produkálnak. Fontos a bővíthetőség, a megbízhatóság és a kompatibilitás. A SpaceWire kis méretű hardvert igényel, valamint egyszerű kábelezést és könnyű szoftveres kezelést tesz lehetővé.

3.1. A SpaceWire tulajdonságai

A SpaceWire szabvány egy olyan hálózati interfészt definiál, mely az OSI modell alsó két rétegét, a fizikai (physical layer) és az adatkapcsolati (data-link layer) réteget fed le (1. ábra).

A felsőbb rétegeken módosított UDP és TCP protokollok működhetnek IP alapú hálózati rendszer felett. Ennek neve SpaceWire Networking Protocol és egye-

1. ábra
OSI modell és a SpaceWire rétegek

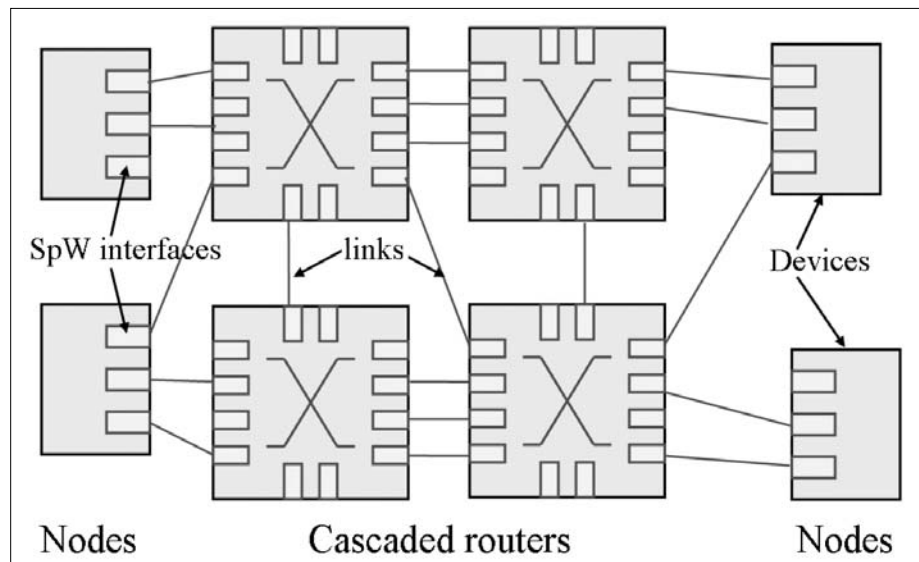


lőre fejlesztés alatt áll, de mindez nem hátráltatja a szabvány alkalmazását, hiszen már több űrszonda is ezzel készült. Sikerét az is jelzi, hogy a NASA is használja egyes misszióiban.

További lehetőségeket nyújthat a TopNet (Technology for On-board Processing in a Network with Extended Throughput) rendszer, amely elképzelés szerint a SpaceWire hálózattal készülő eszközöket egy kapugépen keresztül közvetlenül el lehet érni az Internettel, így a fejlesztést távolról is segíthetik, illetve így is lehet modulokat csatlakoztatni a szondához tesztelési célokkal.

Egy SpaceWire hálózat csatolóelemekből (linkekből), végpontokból (node-okból) és útválasztókból (routerekből) áll. Az egyes node-ok pont-pont kapcsolatban állnak egymással. Kommunikációjuk kétirányú, full-duplex, 2-200 Mbit/s adatátviteli sebességű linkeken történik, a hálózatot a routerek kapcsolómátrixként kötik össze (2. ábra).

2. ábra
A SpaceWire hálózat felépítése



Mint az 1. ábrán is látható, a SpaceWire további hat rétegre bontható.

A *fizikai szint* (physical level) a kábeleket, csatlakozókat, a *jelszint* (signal level) a vezetékeken futó elektromos jeleket és ezeknek logikai megfelelőit, a *karakterszint* (character level) pedig a kommunikáció elemeit írja le. Az *adatkapcsolati szint* (exchange level) a link működését, a *hálózati szint* (network level) a hálózati topológiát, végül a *csomagszint* (packet level) az egyes csomagok strukturáját definiálja.

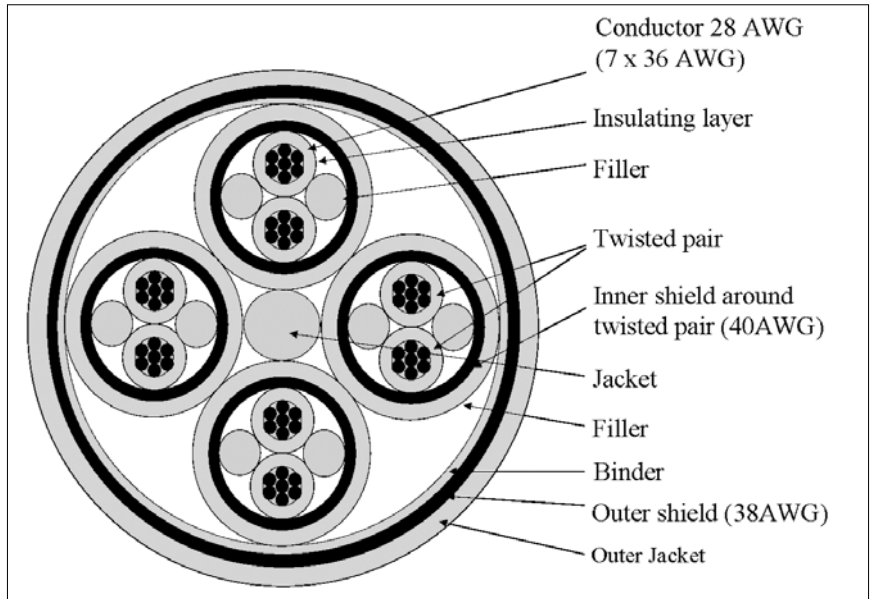
3.2. Fizikai szint (Physical Level)

A jó zajvédetség miatt árnyékolt, sodort érpárokat használnak. Egy kábelben összesen nyolc ér fut párosával sodorva, árnyékolva, majd ezt a négy érpárt újabb árnyékolás védi (3. ábra).

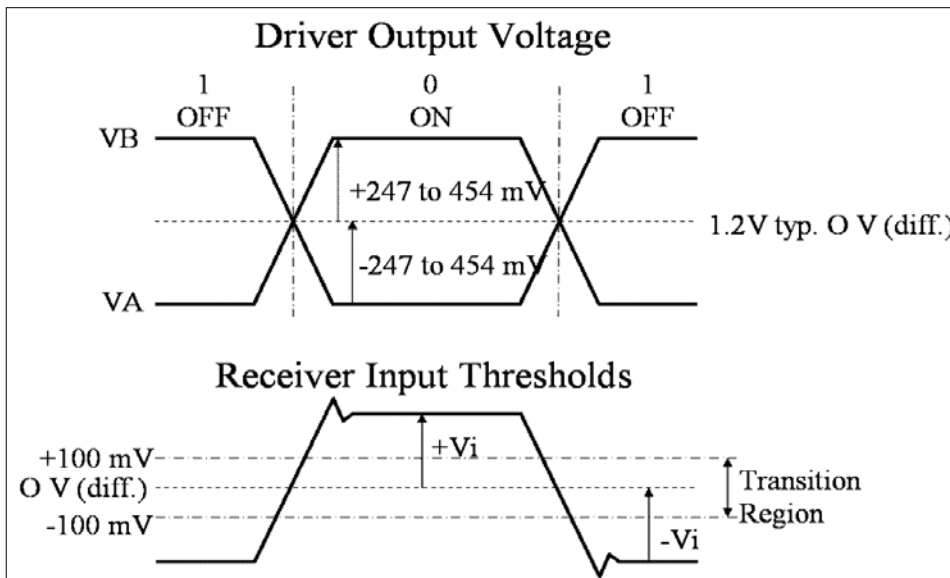
A kábelvégeken 9 tűs mikro-miniatúr D csatlakozót használnak. A kábelek hossza 10 méter lehet, ezen felül már nem biztosítható a maximális adatátviteli sebesség (400 Mbit/s), de a sebesség csökkentésével a kábelhossz akár 100 méterig is növelhető.

3.3. Jelszint (Signal Level)

A jó átvitel és a kis EM sugárzás érdekében a jelkódolást is megfelelően kellett kialakítani, ezért esett a választás az LVDS-re (Low Voltage Differential Signalling – kisfeszültségű differenciális jelkódolás). Az LVDS egy jelet két vezetéken visz át, melyek ellenütemben dolgoznak (4. ábra).



3. ábra
A SpaceWire kábel



4. ábra
LVDS jelszintek

5. ábra
Data-strobe kódolás

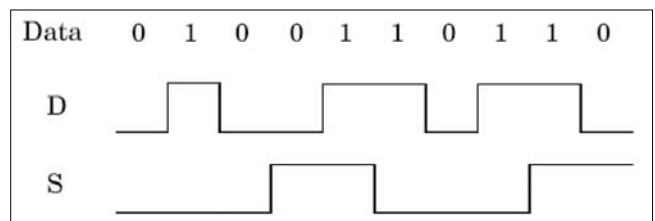
Jelenként 3,5 mA áramfelvétellel számolhatunk, az az kábelenként 14 mA-t vesznek fel a meghajtó áramkörök, így a teljesítményfelvétel körülbelül 50 mW lesz, ami jóval kedvezőbb a korábbi ECL és PECL meghajtó áramkörök 120 mW körüli fogyasztásánál.

Az adatokat soros módon Data-Strobe (adat-enge-délyező) kódolással továbbítják. Ez a kódolás két jelvezetéken (data és strobe) történik úgy, hogy az adatvonalon az adat legalacsonyabb bitjével kezdve sorosan kiküldésre kerül és a strobe vonalon akkor változik a jel, ha az adatvonal nem változott (5. ábra).

Így egy egész bitidőnyi csúszás (skew) engedhető meg a két jel között, szemben az órajeles megoldás fél bitidős skew toleranciájával. A vételi órajelet a D és S jelekből XOR logikai művelet segítségével kaphatjuk. Ekkor az órajel lefutó és felfutó élénél is egy-egy bejövő adattal kell számolni.

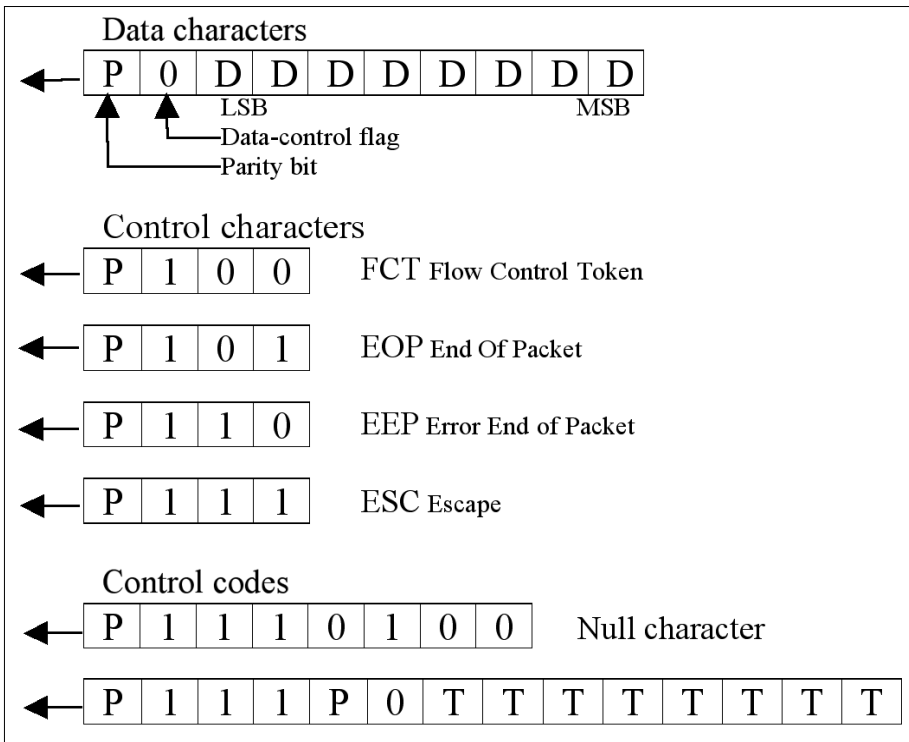
3.4. Karakter szint (Character Level)

A SpaceWire kétféle karaktertípust definiál, az adat- és a vezérlő karaktereket. Az adatkarakterek tíz, a ve-



zérlő karakterek négybitesek, paritásbittel védettek. Összesen négy vezérlő karakter létezik, ezek az adatfolyam irányítására valók. Szükséges még két vezérlőszó is, melyek az előbbi karakterekből épülnek fel. A Null-karakter a link fenntartásához kell, a Time-Code karakter pedig az óraszinkronizálásnál használható. A karakterek felépítése a 6. ábrán látható.

Érdekes a paritás generálása is, ugyanis a páratlan paritás nem az aktuális karakter bitjeit fedi, hanem az adatkarakter utolsó 8, vagy az előző vezérlő karakter utolsó 2 bitjét, valamint az aktuális karakter Data-control flagét. A Data-control flag, vagyis adat-kontrol jel mutatja meg, hogy az aktuális karaktert adatként, vagy vezérlő jelként kell-e értelmezni.



Az inicializálás során a végpontok Null-karaktereket küldenek, majd FCT-vel (Flow Control Token – adatfolyam vezérlő jel) nyugtázzák, ha a másik végponttól Null-karaktert fogadtak. Az ekkor küldött FCT-k száma a vevőoldali puffer méretére utal, minden kiküldött FCT 8 byte puffer méretet jelent. A maximális pufferméret 56 byte, ami 7 FCT kiküldését jelenti. Később minden 8 fogadott adatkarakter után egy újabb FCT kerül kiküldésre, ha a vevő pufferében van 8 karakternyi szabad hely.

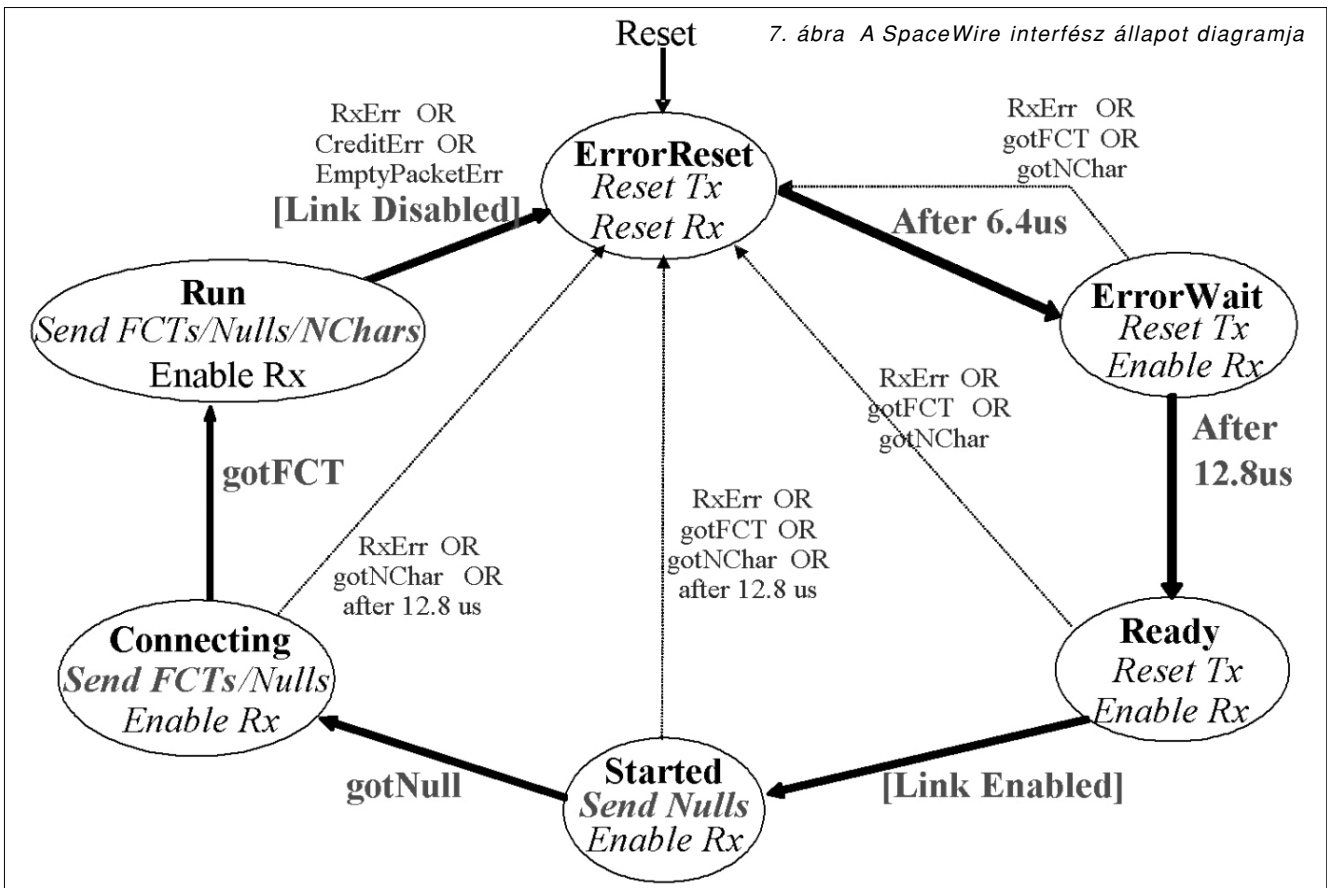
Ezzel a *Flow control*-al biztosítható, hogy a vevő mindig képes legyen az adatok fogadására, a végpontok ugyanis számolják a fogadott FCT-eket és a küldött adatokat, így rögtön kiderül, ha a túlóldali vevő puffere megtelt.

6. ábra SpaceWire karakterek

3.5. Adatkapcsolati szint (Exchange Level)

Ez a réteg építi fel és tartja fenn a kapcsolatot a két végpont között. Feladata a link inicializálása, az adatátvitel nyomon követése, a hibák felderítése és javítása.

Az adatátvitel során pillanatnyi zaj, vagy a data vagy strobe jelek váratlan elcsúszása (skew) miatt paritáshiba léphet fel. Az egybites paritás egy bitnyi hiba detektálására alkalmas, de ez elég is, mert az erősen védett linken igen valószínűtlen egy karakterben 2 bithiba előfordulása. Ennek felderítése viszont már a felsőbb pro-



tokollok feladata. További hibajelenség a Disconnect error (kapcsolatbomlási hiba), mely akkor lép fel, ha a bemeneti vonalak (data és strobe) egyikén sincs változás 850 ns-ig. Ez határozza meg a legkisebb adatátviteli sebességet is, ilyen hiba akkor következhet be, ha az egyik végpontban szoftverhiba lép fel vagy váratlan újraindítás történik. További hibák is felléphetnek, például egy csatlakozó rossz illeszkedése esetén, ha csak az egyik vezeték hibás, akkor sem épülhet fel a kapcsolat, vagy ha ez menet közben következik be, paritáshiba keletkezik. Egyetlen eset lehetséges, amikor a strobe vonal szakadása nem okoz hibát, ha egy adatkarakter küldése közben történik. Ekkor a vevő folyamatos „AA” adatot vesz.

Létrejöhetnek még a protokollt érintő hibák is, melyek kényes pillanatokban bekövetkező adatátviteli hibákra, vagy az egyes végpontok hibás működésére hívhatják fel a figyelmet.

A link működését egy állapotgép vezérli, melynek egyszerűsített diagramja az előző oldali 7. ábrán látható.

Egy kapcsolat inicializálása a következő képen történik: a link bekapcsolás vagy bármely hiba után Error-Reset állapotba kerül, majd idővel, ha a link engedélyezve van és nem keletkezik hiba, eljut a Started állapotig, ahol megkezdődik a Null karakterek küldését. Ha a másik oldal is eljutott ebbe az állapotba és fogadjuk az onnan jövő Null-t, akkor a Connecting fázis következik, mikor FCT karakterek segítségével megállapodnak a végpontok a puffer-méreteken és máris Run állapotba kerülnek, ami a kapcsolat további teljes időtartama alatt fennáll.

Bármely fellépő hiba a link lebomlásához, majd újrapcsolódáshoz vezet, ez a folyamat elvileg rejtett is maradhat a felsőbb rétegek előtt, feltéve, ha épp nem történik adatátvitel, ezért az inicializálás során fellépő hibák nem is kerülnek jelentésre. Ha viszont adatküldés során lép fel egy hiba, akkor az épp küldés alatt álló adatcsomag hátralévő részét el kell dobni, küldését folytatni nem szabad. Ekkor a vevőoldalon a csomag fogadása nem fejeződik be, hibajelöléssel ellátva jelenteni kell, hogy a packet sérült, hiányos. EEP karaktert akkor kell küldeni a másik node-nak, ha a packet adását valami miatt (például felhasználói megszakítás) leállítjuk. EEP karaktert fogadhatunk akkor is, ha a kapcsolat rou-

teren keresztül jött létre és a hiba a router és a távoli node között jött létre. Ekkor az EEP karaktert a router generálja.

3.6. Csomagszint (Packet Level)

A csomag egy olyan struktúra, mely címből, törzsből és csomag vége jelből áll:

`<address><cargo><EOP>`

A csomag a linken egy adat-karakterrel kezdődik, és EOP (End of Packet – csomag vége) vagy EEP (Error End of Packet – hibás csomagvég) kontrol-karakterrel végződik. A cím több részből is állhat, ezek egymás után vannak felsorolva. Az utolsó címet a törzs első byte-ja követi. A csomag törzsére nincs korlátozás, mérete tetszőleges lehet, a SpaceWire fölött futó protokoll határozza meg. A csomagot lezáró EOP vagy EEP utáni első adat-karakter már a következő packet elejét jelenti.

A BepiColombo esetében a cargo egy protokoll azonosítót, két kitöltő bájtot (zéro értékűt), valamint a egy telemetria (TM) vagy egy telecommand (TC) csomagot tartalmaz. A csomagok a Consultative Committee for Space Data System (CCSDS) által kidolgozott szabvány szerinti struktúrával rendelkeznek [5]. A TM csomag egy fejrészből és adatrészből áll, az adatrész maximum 4112 bájtot tartalmazhat. Ilyen csomagban érkeznek a tudományos adatok, és a rendszer állapotát leíró, úgynevezett housekeeping adatok. A telemetria csomagok adatmezőjének fejléce tartalmaz egy 48 bites időkódot, ami elengedhetetlen a különböző érzéklők által gyűjtött adatok együttes elemzéséhez.

A TM struktúráját a 8. ábra mutatja be. A TC csomag három részből áll, a fejrészt a maximum 246 bájtos hosszúságú adat mező követi, végül pedig az ellenőrző összeget tartalmazó adat következik (9. ábra). A telecommand csomag általában a földről felküldött parancsokat tartalmaz, de a fedélzeten lévő autonóm rendszerek vagy időzítők is generálhatnak ilyen packeteket.

3.7. Hálózati szint (Network Level)

A SpaceWire hálózat pont-pont jellegű, de az egyes node-ok nincsenek minden másikkal összekötve, hanem kaszkád router-hálózaton keresztül kapcsolódnak. Az egyes node-ok rendszerint több SpaceWire interfésszel rendelkeznek, melyek egy vagy több routerhez

8. ábra
A TM csomag
struktúrája

Packet Header (48 bits)			Packet Data Field (variable)	
Packet ID	Packet Seq. Contol	Packet Length	Data Field Header (time code 48b)	Source Data
16 bits	16 bits	16 bits	80 bits	variable <4112

9. ábra
A TC csomag
struktúrája

Packet Header (48 bits)			Packet Data Field (variable)	Packet Error
Packet ID	Packet Seq. Contol	Packet Length	Data Field Header	Error Code
16 bits	16 bits	16 bits	32 bits	variable <246
				16 bits

csatlakoznak (2. ábra). Egy űreszköz hálózatának felépítésénél fontos szempont a megbízhatóság, ezért minél nagyobb redundanciát érdemes beleépíteni. A routerek közti többszörös kapcsolat a hálózat áteresztőképességét is növeli, így a nagyobb adatmennyiséget produkáló kísérleteknek még nagyobb sávszélesség biztosítható.

Az egyes végpontok kétféle címezéssel érhetők el. A fizikai címezés (path addressing, hardware addressing) a hálózat tényleges felépítéséből adódik, ez a cím a routerek kimeneti portjai azonosítóinak sorozatából áll. Logikai címezés (logic addressing) esetén minden végpont egyedi azonosítót kap, ebben az esetben a cím csak ebből az azonosítóból áll és a routerek egy táblázat szerint (routing table) választják ki portjaikat. Nagyobb hálózatok esetében, vagy ha a fizikai felépítés ismeretlen, a logikai címezés használata az előnyös. Ekkor használható a regional addressing (régional címzés), mely voltaképp logikai címezés. Ebben az esetben a fizikai címezéshez hasonlóan a cím több részből áll és az egyes alhálózatokat (régiónkat) összekötő routerek kitörlnek a cím elején az őket azonosító logikai címet. Így elvileg tetszőlegesen sok végpontot tartalmazó hálózat jöhet létre. Előfordulhat azonban, hogy az egyik router adott portját már egy másik link lefoglalta, ezen segíthet logikai címezés esetén a Group adaptive routing. Ekkor két router közt nem a routing table szerinti kapcsolat jön létre, hanem egy szabad portot választanak ki.

Meg kell említeni, hogy elvileg egy aktív link folyamatosan lefoglalja a résztvevő node-okat, routereket. Ez viszont felesleges sávszélesség-csökkenést jelenthetne, hisz így a routerek közti kisebb számú csatlakozást helykitöltő null-karakterek forgalmazásával terhelnék. A wormhole routing eljárás lehetővé teszi, hogy egy ilyen link csak akkor legyen foglalt, amikor a végpontok közt tényleges kommunikáció van, azaz épp egy packet küldése folyik. Az egyes node-ok elől ez el van rejtve, ha a vonal épp foglalt, akkor a router puffert töltheti meg, majd szünetelteti az adást, míg FCT-t nem kap.

Űreszközökön általában megfelelő a fizikai címezés, hisz viszonylag kevés a végpont, és pontosan ismert a fizikai struktúra, de igen hasznos lehet, ha a fejlesztés során az egyes eszközök szélesebb, dinamikusan változó hálózatról is elérhetők.

4. Összefoglalás

A SpaceWire hálózat előnyei más rendszerekkel szemben onnan fakad, hogy kifejezetten űreszközök fedélzeti kommunikációjához fejlesztették ki, az ottani igényeknek megfelelően. Redundanciájának köszönhetően akkor is teljes működést tud biztosítani, ha egyes eszközök meghibásodnak, kicsi a teljesítményfelvétele, sávszélessége széles határok közt akár menet közben is változtatható, kis válaszidőt és késleltetést ad, full-duplex működést biztosít és modulárisan bővíthető. Az adaptive és a wormhole routingnak köszönhetően jóval

nagyobb lehet a teljes hálózat áteresztőképessége, mint egy ethernetes hálózaté. Hardverigénye kicsi, és egyszerű szoftverekkel lehet a meglévő protokollokhoz illeszteni.

További nagy előnye, hogy kompatibilissé tette az eszközöket, így nem kell ugyanazokat a funkciókat el látó modulokat, mint az adatgyűjtő rendszer és a háttértárak, minden űreszközhöz újra megtervezni, megépíteni, és ezzel jelentős költségcsökkenés érhető el.

Köszönetnyilvánítás

A BepiColombo misszióban való részvétel az ESA PECS irodájánál elnyert pályázatok, valamint az MTA támogatásával valósul meg. Az ennek során szerzett ismeretek alapján készülhetett el jelen cikkünk.

A szerzőkről

Sulyán János 1962-ben végzett a BME Villamosmérnöki Kar Híradástechnikai szakán – KFKI ösztöndíjasként. A KFKI-MSZKI-ban kommunikációs illesztők és lokális, laboratóriumi hálózatok fejlesztésével, valamint kisgép-nagygép összekapcsolására szolgáló illesztőegységek létrehozásával foglalkozott. 1990-től a KFKI RMKI-ban űrszondák tudományos berendezéseinek földi bevizsgálására szolgáló eszközök és fedélzeti elektronikák fejlesztésében vett részt. Nyugdíjasként az SGF Kft-nek fejleszt beágyazott rendszereket.

Szalai Lajos 2003-ban végzett a BME Villamosmérnöki Karán Beágyazott rendszerek szakirányon. 2004-től a KFKI Rézecske és Magfizikai Kutató Intézetben fejlesztőmérnök. Ezidáig többek között USB és SpaceWire interfészek fejlesztésével és kezelésével foglalkozott, jelenleg a fedélzeti tápegységek tervezése, illetve a BepiColombo-hoz fejlesztett SpaceWire interfész alsóbb rétegeinek megvalósítása volt a feladata.

Lipusz Csaba 1986-ban végzett fizikus szakon az ELTE TTK-n. A hardverközeli programozás és az informatikai rendszerek technikai architektúrájának fejlesztése olyan területek, melyeken pályafutása során több évet dolgozott. Jelenleg fedélzeti adatgyűjtő berendezések real-time Linux operációs rendszereivel és illesztő kártyák device drivereinek fejlesztésével foglalkozik.

Szalai Sándor 1962-ben villamosmérnöki, 1967 folyamatszabályozási szakmérnöki oklevelet szerzett a BME-n. Kezdetben a nukleáris elektronika területén, majd 1980-tól az űrkutatás területén fejlesztett vezérlő és adatgyűjtő rendszereket. Az első munkái hardver, majd később hardverközeli szoftver-fejlesztések, valamint rendszertervezés voltak. 1977-ben kandidátus, 1992-ben a műszaki tudomány doktora lett. 1977-től a KFKI RMKI-ban dolgozik, tudományos tanácsadó. 1996-tól az SGF Kft. ügyvezető igazgatója.

Irodalom

- [1] ESA Science & Technology – BepiColombo Homepage
<http://sci.esa.int/science-e/www/area/index.cfm?fareaid=30>
- [2] ESA Research and Scientific Support Department – BepiColombo Homepage
<http://www.rssd.esa.int/index.php?project=BEPICOLOMBO>
- [3] ESA SpaceWire Homepage
<http://spacewire.esa.int>
- [4] ECSS-E-50-12A Spacewire – Links, nodes, routers and networks, 2003.01.24.
- [5] CCSDS 850.0-G-1
Spacecraft Onboard Interface Services.
Green Book, Issue 1, June 2007.